

InnolabAMS: An Admission Management System for Elementary and Junior High School Institutions in the Philippines

Project Documentation Submitted to the Faculty of the

School of Computing and Information Technologies

Asia Pacific College

In Partial Fulfillment of the Requirements for

Introduction to Systems and Design for CS/IT

M/S NTSDEV

By

|  |  |
| --- | --- |
| *James Albert David* | *Reycel John Emmanuel Carcueva* |
| *Reejay Salinas* | *Althea Noelle Sarmiento* |

2024

Table of Contents

[Executive Summary i](#_Toc181187495)

[List of Figures iv](#_Toc181187496)

[List of Tables v](#_Toc181187497)

[I. Introduction 1](#_Toc181187498)

[1.1 Project Context 1](#_Toc181187499)

[1.2 Statement of the Problem 2](#_Toc181187500)

[1.3 Objectives 2](#_Toc181187501)

[1.4 Significance of the Project 3](#_Toc181187502)

[1.5 Scope and Limitations 4](#_Toc181187503)

[II. Review of Related Literature / Systems 5](#_Toc181187504)

[III. Current Systems 9](#_Toc181187505)

[3.1 Current System 9](#_Toc181187506)

[3.2 Technical Background 9](#_Toc181187507)

[3.3 List of Processes 10](#_Toc181187508)

[3.4 Gap Analysis 12](#_Toc181187509)

[**SWOT Analysis** 13](#_Toc181187510)

[IV. Proposed Solution 13](#_Toc181187511)

[4.2 Lean Canvas 14](#_Toc181187512)

[4.3 Product Vision 16](#_Toc181187513)

[4.4 Technology Specifications 16](#_Toc181187514)

[4.5 Feasibility 18](#_Toc181187515)

[V. Requirements Analysis 22](#_Toc181187516)

[**5.1** **Product Backlog / User Stories** 22](#_Toc181187517)

[**5.2** **User Classes and Characteristics** 23](#_Toc181187518)

[**5.3** **Data Flow Diagram** 24](#_Toc181187519)

[**5.4** **Use Case Diagram** 31](#_Toc181187520)

[**5.5 Test Cases for Fully Dressed Cases** 39](#_Toc181187521)

[**5.5** **Activity Diagram** 43](#_Toc181187522)

[**5.6** **Entity Relationship Diagram with Data Dictionary** 46](#_Toc181187523)

[**5.7** **Prototype/Wireframe** 61](#_Toc181187524)

[**5.8** **Technology Stack** 77](#_Toc181187525)

[**5.9** **GitHub Repository** 78](#_Toc181187526)

[**5.10** **Updates of contents in Project GitHub Repository** 79](#_Toc181187527)

[**5.11** **Cloud hosted prototype** 83](#_Toc181187528)

[**5.12** **Release Plan** 83](#_Toc181187529)

[Reference 84](#_Toc181187530)

[Appendices 87](#_Toc181187531)

[Appendix A: Project Vision 87](#_Toc181187532)

[Appendix B: Schedule 88](#_Toc181187533)

[Appendix C: Product Roadmap 89](#_Toc181187534)

[Appendix D: Teams Meetings 90](#_Toc181187535)

# Executive Summary

InnolabAMS is an admission management system that aims to improve the efficiency of admission processes of private educational institutions in the Philippines. Moreover, InnoLabAMS also features analytics reporting module, providing schools with real-time insights to admission trends and lead reports enabling them to make data-driven decisions and identify the effectiveness of their admission efforts.

The system is cloud-based that provides scalable and accessible solutions for the target market. The Serviceable Addressable Market (SAM) of the project comprises private schools offering elementary and junior high school program located in the Philippines. Specifically, the focus of the team’s startup target market narrows down to private schools within Makati City offering the same program that currently relies on manual processes for managing administrative processes. Additionally, InnolabAMS will operate on the Software-as-a-Service model, providing cost-effective solutions for schools that face budget constraints and cannot afford to invest in internal IT solutions teams or expensive system providers.

The development of InnolabAMS will involve several key steps to ensure the success of its creation. The process will begin with requirements gathering and analysis, where the team gathers information about the specific requirements and data that educational institutions need to collect from the applicants. This will be followed by system planning and design, involving the creation of user interfaces, and database structures. Once the design is finalized, the development phase will commence, in which the core features of InnolabAMS will be implemented. These features involve admission, admission period, system settings and analytics dashboard. Next is the testing phase, which involves testing procedures to identify and fix any bugs or issues, ensuring the system operates smoothly. Finally, in the deployment phase InnolabAMS will be launched on a cloud-based infrastructure, allowing multiple users to use the instance of the system. Security measures will also be implemented such as user authentication and data encryption to ensure adherence to compliance standards related to data protection and privacy.

The project timeline spans about 12 months and aligns with the team’s school curriculum, with each phase averaging approximately three months, corresponding to the usual duration of the school term. This alignment ensures that key milestones, such as system planning, designing, development, testing, and deployment, are executed on par with the given timeframe.

With a team of four members, the project resources are strategically allocated to maximize efficiency and collaboration throughout the development of InnolabAMS. The team’s human resources include four skilled students proficient in various programming languages and development frameworks. The team is in contact with a resource person who has a background in basic education, their knowledge and expertise will contribute to fulfilling this project. Additionally, the team also has access to technological resources such as development tools and devices necessary for coding, testing, and deployment. The team’s financial resources are carefully managed to cover expenses related to web development, cloud subscription, integration, and other project costs. Furthermore, the team prioritizes effective communication and collaboration tools to facilitate coordination among team members and ensure that the project progresses smoothly within the designated timeline.

Furthermore, the expected outcomes of this project include the successful fulfillment of InnolabAMS’ objectives and functionalities delivering a user-friendly, efficient, effective and smart admission management system. For the team members, this project will offer them valuable learning experiences and skill development opportunities. They will gain hands-on experience in design thinking, cloud computing and database management. For their personal development, the team will improve their teamwork, problem-solving, and communication abilities. The team’s contribution to this project will also build their portfolio and opens networking opportunities preparing them for future roles in the industry.

# List of Figures

[Figure 1 Admission Process 11](#_Toc181187453)

[Figure 2 Inquiry 11](#_Toc181187454)

[Figure 3 Fishbone Diagram 12](#_Toc181187455)

[Figure 4 Data Flow Diagram: Context 25](#_Toc181187456)

[Figure 5 Data Flow Diagram: Level 1 26](#_Toc181187457)

[Figure 6 Data Flow Diagram: Level 2 Manage Inquiries 27](#_Toc181187458)

[Figure 7 Data Flow Diagram: Level 2 Manage Application 28](#_Toc181187459)

[Figure 8 Data Flow Diagram: Level 2 Manage Scholarship 28](#_Toc181187460)

[Figure 9 Data Flow Diagram: Generate Admission Reports 29](#_Toc181187461)

[Figure 10 Data Flow Diagram: Level 2 Manage Account 30](#_Toc181187462)

[Figure 11 Use Case Digram 31](#_Toc181187463)

[Figure 12 Manage Account 43](#_Toc181187464)

[Figure 13 Generate Admission Report 44](#_Toc181187465)

[Figure 14 Manage Inquiries 45](#_Toc181187466)

[Figure 15 Manage Scholarship 45](#_Toc181187467)

[Figure 16 Manage Application 46](#_Toc181187468)

[Figure 17 Entity Relationship Diagram 47](#_Toc181187469)

[Figure 18 Entity Relationship Diaram Tables 48](#_Toc181187470)

[Figure 19 Wireframe: Welcome Page 61](#_Toc181187471)

[Figure 20 Wireframe: Sign In 62](#_Toc181187472)

[Figure 21 Wireframe: Register 63](#_Toc181187473)

[Figure 22 Wireframe: Inquiry Form 64](#_Toc181187474)

[Figure 23 Wireframe: Personal Information 65](#_Toc181187475)

[Figure 24 Wireframe: Educational Background 66](#_Toc181187476)

[Figure 25 Wireframe: Family Information 67](#_Toc181187477)

[Figure 26 Wireframe: Documents 68](#_Toc181187478)

[Figure 27 Wireframe: Additional Information 69](#_Toc181187479)

[Figure 28 Wireframe: Scholarship 70](#_Toc181187480)

[Figure 29 Wireframe: Dashboard 71](#_Toc181187481)

[Figure 30 Wireframe: Admissions 72](#_Toc181187482)

[Figure 31 Wireframe: Admissions Full View 73](#_Toc181187483)

[Figure 32 Wireframe: Scholarship 74](#_Toc181187484)

[Figure 33 Wireframe: Inquiries 75](#_Toc181187485)

[Figure 34 Wireframe: Users 76](#_Toc181187486)

[Figure 35 Wireframe: Account Management 77](#_Toc181187487)

[Figure 36 Release Plan 83](#_Toc181187488)

# List of Tables

[Table I List of Process 10](#_Toc181187489)

[Table II Gap Analysis 12](#_Toc181187490)

[Table III SWOT Analysis 13](#_Toc181187491)

[Table IV Product Vision 16](#_Toc181187492)

[Table V Product Backlog 22](#_Toc181187493)

[Table VI User Classes and Characteristics 23](#_Toc181187494)

# Introduction

Admission Management System is a software tool that helps educational institutions automate and optimize their admission processes, leading to a seamless experience for both applicants and administrators [1]. In today's rapidly evolving educational landscape, the efficient management of student information is important for the smooth operation of educational institutions [2]. In response to this demand, our team introduces InnolabAMS: An Admission Management System for K-12 program that aims to improve the efficiency of admission processes of private institutions in the Philippines offering elementary and junior high school program.

## Project Context

Many schools in the Philippines still rely on manual processes in managing administrative processes, these manual processes are time-consuming that often lead to long queues, delays and inaccuracies, impacting the overall efficiency of administration [3]. InnolabAMS is a cloud-based admission management system that will address the administrative inefficiencies of educational institutions by automating the admission process. With InnolabAMS, school administrators can seamlessly receive student applications from the digital admission forms and manage them efficiently on demand. The system also offers analytics reporting module, providing schools detailed insights of their admission trends and lead reports enabling them to make data-driven decisions for optimizing the school’s resources and identify the effectiveness of their admission efforts.

InnolabAMS will be deployed on a cloud infrastructure to reduce the risk of data loss as the student records and other relevant data will be stored securely in a cloud-based storage ensuring availability even in the face of hardware failure and disasters within the school’s premises. The cloud environment provides security features that ensures adherence to security standards regarding data privacy and regulations from educational institutions.

Furthermore, InnolabAMS will be operating on the Software as a Service (SaaS) model offering cost-effective and scalable solution for the target market. The team's target market for this system comprises of two distinct segments, the Serviceable Addressable Market (SAM) and the Startup Target Market. The SAM includes private schools offering elementary and junior high school program located in the Philippines. For the startup target market, the team chose private schools in Makati City offering the same program.

## Statement of the Problem

Based on studies, here are the common problems of private schools that do not have an admission management system:

1. In the admission process, the administrative staff manually enter student data in digital spreadsheets or paper-based records, which are time consuming leading to long queues and longer processing time.

2. Due to reliance on manual processing, data errors may occur which leads to delays in processing a student application.

3. Admission office find it difficult to evaluate the effectiveness of their marketing strategies due to the absence of admission reports.

## Objectives

**Main Objective**

The main objective is to develop an admission management system that will improve the efficiency of the administrative processes to address the identified problems of private institutions.

**Specific Objectives**

1. Improve the efficiency of data entry of the applications, aiming for substantial improvement.

2. Eliminate data errors for every application entirely.

3. Generate timely reports from inquiries and applications.

## Significance of the Project

This project will have a significant benefit to the following stakeholders:

**Administrators.** Administrators, particularly those in school admissions, will greatly benefit from the automated processes offered by the system. By utilizing features such as managing inquiries, applications, scholarships, accounts, and generating admission reports, administrators can improve both the accuracy of data and operational efficiency. The system will eliminate manual data entry errors and allow for quicker, more organized handling of admissions. **Parents.** Parents can submit their child's application anytime, anywhere while also receiving status updates from the school.

**Applicant.** Applicants will experience a smoother transition into the school, as the system facilitates efficient processing of their applications.

## Scope and Limitations

The system only covers main functions such as managing inquiries, application, scholarship, account and analytics reporting. The system will only accommodate private schools offering kindergarten to senior high school programs. It will not address public schools and senior high school programs because the administrative structures of these settings are vastly different.

The system's scope is limited to managing admission process, receiving applicant and lead information such as personal, family and educational background and any other relevant information pertaining to admissions. Reporting data thar are only relevant to admission trends and lead reports from inquiries and applications. Excluding other data management needs within educational institutions such as academic performance, financial management, formulating transcripts and alumni relations. Furthermore, the system also requires adequate internet connectivity and technical proficiency, which may pose challenges in some regions or institutions.

# Review of Related Literature / Systems

**Introduction to the Chapter**

This chapter summarizes local and foreign literature relevant to the study. The first section provides an overview of the functionalities of an enrollment management system. This is followed by an analysis of the already existing admission systems, identifying their strengths and weaknesses in relation to the study’s objectives. The third section compares different student information systems to highlight the gaps that our system aims to address. Lastly, the final section of this chapter summarizes the key findings that will aid in the development of the system.

1. **Theoretical Foundations**

In an article, Why K-12 Schools Should Manage Student Enrollment Online  [4] , the education system is currently on a slow technological growth in comparison to the other industries. As such, student documents are often recorded through pen and paper. Furthermore, the enrollment process is one of the biggest problems in [5] as it usually takes 2-3 weeks due to the manual storage, retrieval and administration of the institution. In a related article [6], the benefits of digitizing the enrollment management system are few of the following: paperwork’s will not easily be lost and misfiled due to the amount of paperwork the administration receives from parents, automation of reminders can be done with online enrollment systems for missing requirements in the process to limit the time and hassle of the administration and lastly, tracking of deadlines are more efficient, unlike having to go through every document one by one.

In the study of [7][8], it is found that many institutions still rely on manual processes in managing school admissions, which are prone to errors, delays, and inaccuracy. As such, different schools have different admission processes, an example would be filling out an online form as a reservation [9], [10] before proceeding to the enrollment process or going to the main campus for both the admission and enrollment process [11]. Manual methods will cause delays in generating accurate and timely analytical reports. Furthermore, some institutions use spreadsheets to manage admissions data, but this method is time-consuming and prone to inaccuracies [12]. In a related article [12], the benefits of an admission management system can provide real-time analytical reports, valuable insights into admission trends, and strategic decision-making.

Multi-Tenancy is a widely used technology in the field of Software as a Service (SaaS) development as it focuses on utilization and allocation of the resources depending on the need of the application, making sure each ‘tenant’ shares the single application and database but also maintaining their own space [13]. Each user is referred to as the process and a multi-tenant application is referred to as an operating system, whereas the applications respond to each of the tenants depending on what they need [14]. Additionally, it enables an easier way to update, maintain and backup the data, requiring only low system requirements to implement [15].

Laravel is an open-source PHP programming language framework established by Taylor Otwell for developing web-based applications [16]. Developing an academic system through Laravel according to [17], provided the following benefits: academic institutions can optimize processes, enhance user experience and improve data management with the use of Laravel’s features.

1. **Related Systems Analysis**

A case study by [18] implemented an online admission and registration system in the Metropolitan International University in Uganda using MySQL and HTML, having a target population of 130 which includes the administrative staff and students. It was found out that having an online admission and registration management system is a huge investment to make with the population of the institution benefiting from it as it is time saving. However, they recommended the system should be well maintained without having unauthorized access from users. In a related study, [19] implemented an online admission system for new students in Surakarta Indonesia using a zoning method. It was said that implementation of the system ran smoothly, having high satisfaction from the Education Department of Surakarta.

Implementation of analytics in an online system such as with [20]. However, it is found out that having analytics in the primary education field is at its early stages and related studies about the implementation of analytics in primary and secondary education are still lacking. Although, the analytics provided a significant effect in the interactions between the students and teachers. Predictive analytics was implemented in a student admission system by [21] to provide support in decisions made by the institution’s staff, provide a probability report in which the student will enroll in a specific course and overall increase the quality of the admission system. It was found that the average accuracy of the implemented model was 72%, recommending to focusing more on improving the accuracy rate of the models.

A related system [22] is an educational management system that automates administrative and academic operations such as real-time student performance tracking, attendance monitoring, and communication among instructors, students, and parents. While WELA offers numerous tools for managing educational activities, its absence of cloud-based infrastructure could limit its efficiency and accessibility. InnolabAMS overcomes these constraints by providing a scalable, cloud-based system with advanced data analytics capabilities, resulting in a more flexible and comprehensive approach to educational administration.

1. **Comparative Analysis**

InnolabAMS is an admission management system for elementary to junior high school program in the Philippines. It will be utilizing cloud systems which are not mentioned by the related admission systems such as [18] and [19]. However, MySQL and HTML will also be utilized in creating the system just like with [18], the team will just add CSS and JavaScript in our creation to provide a more dynamic website but also to be user-friendly. With [19] using a zoning system to create their student admission system, InnolabAMS will be utilizing multi-tenancy to support the system being as a SaaS, providing its users their own separate space in the application while still sharing the single instance of the system.

Analytics will also be implemented in the system just like with [20] and [21], the only difference of the InnolabAMS analytics is how it will be utilized to provide the institution with reports regarding the trends in the admissions stated on the administrators’ dashboard to help them make data driven decisions.

1. **Synthesis and Summary**

With the field of education evolving, the development and implementation of an admission management system (AMS) becomes an important role for the efficiency of the current way of the current admission procedures in schools within the Philippines. The system is essential due to its 24/7 accessibility, acknowledging the difference in time zones and place of the potential enrollees, reduced allocation of materials due to the automotive process of the system as well as its cost effectiveness and scalability to accommodate the student population.

In the context of InnolabAMS, the system highlights potential benefits and challenges that will be addressed to develop an effective solution. Implementation of data analytics and putting the system in the cloud will be included in the development of the system to enhance the overall effectiveness of the system. By utilization of these technologies, the system can provide support in decision making regarding the admission processes of the institutions while also optimizing the process and reducing the cost.

# Current Systems

## 3.1 Current System

**Manual Process**

Administrative processes such as admission and student enrollment are done manually, often involving handwritten notes and physical files.

**Spreadsheet**

In schools that utilize digital spreadsheets, managing administrative processes including, admission and enrollment information are entered and analyzed manually using spreadsheets.

## 3.2 Technical Background

**Hardware**

Client computers serve as workstations for users including administrative staff can input, manage, and analyze data using spreadsheets. On the other hand, servers are utilized in hosting shared spreadsheets, providing centralized storage and backup capabilities.

**Software**

The computers are equipped with standard office hardware with Windows operating systems that can run spreadsheet software such as Microsoft Excel or Google Sheets.

**Network**

Network infrastructure, including local area networks (LAN) and internet connections, enables communication between client computers and server with security measures such as firewalls and authentication to protect data integrity and privacy.

## 3.3 List of Processes

Table 1 contains the list of administrative processes being performed by educational institutions.

Table I List of Process

|  |  |  |
| --- | --- | --- |
| Process ID | Process  Name | Process  Details |
| P1 | Admission | Collecting basic personal information and required documents from the applicant |
| P2 | Inquiries | Collecting basic personal information and inquiries from leads |
| P3 | Scholarship | Collecting scholarship information and assigning scholarship to an applicant. |

Figure 1 Admission Process

This figure shows the step-by-step admission process being performed by educational institutions [20].

Figure 2 Inquiry

This figure shows the step-by-step process of how APC markets and generate leads from school activities.

This figure shows the step-by-step process of how admissions manage a scholarship application.

## 3.4 Gap Analysis

*Gap Analysis*

Table II Gap Analysis

|  |  |  |
| --- | --- | --- |
| Current State | Desired State | Impact |
| P1 | Automated admission process using online forms with document submission and online verification. | Reduced processing time, queues and minimized errors. |
| P2 | Centralized reporting module generated from the dedicated online log sheet. | Obtain timely reports such as admission trends, lead reports and conversion rate without the need to create online forms and separate log sheets. |

Fishbone Diagram

A diagram of a diagram

Description automatically generated

Figure 3 Fishbone Diagram

This diagram illustrates the various potential causes contributing to the prolonged duration of the admission process.

### **SWOT Analysis**

Table III SWOT Analysis

|  |  |
| --- | --- |
| Strength   * Cloud-based * Accessibility * Scalability * User-friendly | Weaknesses   * Dependence on internet connectivity * Resistance to change * Requires training |
| Opportunities   * Growing Demand for Digital Solution * Emerging Technologies | Threats   * Competitive Market * Security |

# Proposed Solution

The core functions of InnolabAMS will revolve around several key module:

**Online Application Portal**– This module will allow applicants and leads to submit their application or inquiries online.  
**Admission Management Module** - This module will allow the administrators to manage submitted applications. This will also allow them to evaluate, update status, set their remarks and send notifications from the applicant via automated email.

**Scholarship Management Module-** This module will allow applicants and administrators to submit and manage scholarship applications.

**Analytics Reporting Module** – This module enables administrators to interact and get insights into the admission trends, lead reports and conversion rate.

## 4.2 Lean Canvas

***Problem***

1. In the admission process, the administrative staff manually enter student data in digital spreadsheets or paper-based records, which are time consuming leading to long queues and longer processing time.
2. Due to reliance on manual processing, data errors may occur which leads to delays in processing a student application.
3. Admission office find it difficult to evaluate the effectiveness of their marketing strategies due to the absence of lead reports.

***Solution***

1. Develop a centralized admission management module that allows administrators efficiently manage and process student applications.
2. Digitalize the current admission processes to improve accuracy of data from each application.
3. Implement an analytics and reporting module that provides administrators with timely and actionable insights from admission.

***Key Metrics***

* Subscription Growth Rate - Measures the rate at which the number of subscribing schools or users is growing over time.
* Monthly Recurring Revenue - Measures the rate at which the number of subscribing schools or users is growing over time.
* User Adoption Rate - Measure of the percentage of users who actively use InnoalabAMS.
* Customer Satisfaction- User feedback scores reflecting overall satisfaction levels and likelihood of recommending InnolabAMS to others.

***Unique Value Proposition***

* Cloud-based platform
* Data analytics
* Cost-efficient solution

***Customer Segment***

* Private schools offer elementary and junior high school program curriculum within Makati City.
* School stakeholders including administrative staff, administrators, students, and parents.

***Channels***

* Contact numbers
* Social Media
* Email
* Zoom

***Revenue Streams***

* Subscription Fees

***Cost Structure***

* Cloud services *-* Costs associated with setting up and maintaining the cloud-based platform where InnolabAMS will be deployed and maintained.

***Unfair Advantage***

* By leveraging cloud technologies, the proposed system can be integrated and further improved for the future. This means that as the educational institution grows or evolves, the system has the flexibility to scale up to accommodate increased demand and new features.

* The team’s adviser is Mr. Jojo F. Castillo who has expertise when it comes to student information systems, he is also the first one to develop the RAMS portal. With his guidance, they will gain invaluable insight and direction to ensure the successful development and implementation of the project.

## 4.3 Product Vision

Table IV Product Vision

|  |  |
| --- | --- |
| For | Private Schools in the Philippines offering Elementary and Junior High School program |
| Who | wants to replace their current manual processes to an automated system |
| InnolabAMS | is a centralized, cloud-based Admission Management System |
| That | enables administrative staff to manage admission process efficiently. |
| unlike | competing software providers |
| Our product | Is a smart Admission Management System, that produces valuable information from the applications and integrates analytics reporting module to provide school administrators detailed insights into the school admission trends, leads and conversion rates. |

## 4.4 Technology Specifications

**Hardware**

* Developer Machines: Each team member will require laptops or computers with sufficient resources to handle development tasks efficiently. Minimum specifications include:
* Processor: Intel i5 or equivalent
* RAM: 8GB or higher
* Storage: 256GB SSD or higher
* Cloud Infrastructure: The team will be utilizing AWS (Amazon Web Services) for hosting the application and managing the database.

**Software**

* UI/UX Design: The team will be utilizing Figma for designing and making prototypes of the system.  
  Database Structure: The team will design the database schema using MySQL Workbench.
* Framework: The team will be using Laravel for developing the backend logic and managing server-side operations. The team can also use its templates that offer pre-built components and layouts ensuring consistency in the user interface.
* **Analytics: The** team will be using Power BI for data visualization and analytics.
* Development Tool: The team will use Visual Studio Code for writing code, using extensions and integrating other tools. GitHub will also be utilized for version control, collaboration and to track changes.

**Peopleware**

* Team Collaboration: This project will be developed by a team of four students, consisting of backend and frontend developers. The team will work collaboratively, scheduling regular meetings, stand-ups, and code reviews to maintain a cohesive development workflow.
* Project Updates: Technical support and guidance will be provided by mentors and instructors to assist the team throughout the development process.
* Agile Methodology: Adopting agile methodologies will provide organized development tasks, prioritize work items, and adaptability to changing requirements efficiently.

**Network**

* **Internet Connectivity:** High-speed internet connection is essential for efficient collaboration and accessing cloud services.
* Collaboration Tools: The team will utilize tools such as Microsoft Teams, SharePoint, Miro and FigJam to communicate, collaborate and coordinate the tasks effectively.

## 4.5 Feasibility

**Operational Feasibility**

Here are the key points to ensure that InnolabAMS can be effectively implemented and utilized by the target market:

*Technical Infrastructure*

InnolabAMS leverages cloud-based technology, which eliminates the need for schools to invest in expensive hardware and IT infrastructure. This enhances accessibility and scalability, given that the environment is Makati City, most private schools already possess reliable internet connectivity, which is essential for the system's smooth operation.

*User Training and Support*

To ensure a smooth transition from manual to automated processes, the system is designed to have a user-friendly interface. On-hand or virtual training programs and ongoing customer support will ensure that administrators and administrative staff can effectively use the system.

*Regulatory Compliance*

InnolabAMS complies with local data privacy laws and educational regulations, ensuring data security and adherence to standards.

*Pilot Testing and Feedback*

Before a full-scale rollout, the team will conduct pilot programs within APC. This approach allows for real-world testing and the gathering of valuable feedback from actual users. So that the team will know what improvements are going to be made.

**Economic Feasibility**

Here are the key benefits of implementing InnolabAMS that can contribute to its economic feasibility:

*Cost of Implementation*

Implementing InnolabAMS involves costs that need to be considered. These expenses are related to hardware and network upgrades such as computers and high-speed internet connections that schools may need to invest in. Subscription fees will also recur, based on the chosen pricing plan which covers the access to the system, maintenance, updates, and support.

*Cost Considerations*

InnolabAMS will operate in SaaS model that provides a cost-effective solution with subscription-based pricing for the schools, making it affordable and predictable, avoiding the need for large upfront investments. The team can also include discounts and free training programs as part of their service package that will benefit both customer and providers as it fosters long-term partnerships.

*Cost Savings*

Schools can significantly reduce their dependence on extensive IT infrastructure and solutions. Cloud services eliminate the need for substantial investments in physical hardware, such as servers and storage devices, as well as the associated costs of maintaining and upgrading these systems. The system can also generate significant cost savings through reduced labor, paperless process, and increased efficiency especially during enrollment peak season. Schools don’t need to invest in infrastructure

*Revenue Generation*

Automated processes can increase customer satisfaction that can potentially lead to higher enrollment numbers and student retention.

**Technical Feasibility**

Here are the factors that can assess the technical feasibility of InnolabAMS:

*Available Resources*

The team have access to standard computing hardware such as laptops or desktops with sufficient processing power and memory.They also have access to development tools including IDE’s, version control systems and frameworks that are completely free to use. Internet access is also available for accessing cloud services, collaborating with team members, and accessing online resources and documentations.

*Capacity and Scalability*

The team will leverage cloud computing technologies, specifically AWS, which provides scalable infrastructure. Their cloud services like Amazon EC2, Amazon RDS, etc., can scale on demand allowing the system to grow and accommodate future needs.

*Prototype*

The team will create an initial prototype with core functionalities as it helps them demonstrate the project's feasibility. It can also be beneficial to validate requirements and gather feedback from potential users to ensure the system meets user needs effectively.

**Scheduling Feasibility**

The team will develop and deploy the project within an acceptable 6-month timeframe, in alignment with the academic calendar. To ensure the successful development of the system, several conditions must be achieved:

*Effective coordination*

The team will prioritize clear communication and collaboration among members. Regular meetings will be held to discuss progress and address any issues.

*Adherence to project milestones*

The team breaks down the project into distinct phases such as planning, designing, development, testing, and deployment which enables regular progress tracking and timely adjustments within the specified timeframe.

*Ongoing stakeholder engagement*

The team will engage with the stakeholders in APC, specifically the school registrar, admissions office, students and adviser. Their feedback will be essential in ensuring the development process of InnolabAMS aligns with their expectations.

*Risks*  
The team will not raise an accelerated schedule because it could introduce potential risks such as increased costs, reduced quality due to rushed development, and higher pressure on the team leading to potential burnout and incomplete features that may outweigh the benefits of the system.

# Requirements Analysis

### **Product Backlog / User Stories**

Table V Product Backlog

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | As a… | I want to be able to | So that… | Priority |
| 1 | Admission Officer | receive new and complete applications and documents seamlessly | I can search and evaluate them efficiently. | Must |
| 2 | Admission Officer | Receive new inquiries from the prospective students seamlessly | I can address concerns and help them get the right information effectively. | Must |
| 3 | Admission Officer | generate admission reports effortlessly | I can assess the effectiveness of the admission process and make data-driven decisions | Should |
| 4 | Admission Officer | receive and manage scholarship applications seamlessly | so that I can evaluate their academic history, financial background in a centralized manner | Must |
| 4 | Applicant | submit my application digitally | so that I can avoid the hassle of paper forms and ensure that the information I submitted is complete and accurate | Must |
| 4 | Applicant | know the status of my application remotely | so that I can stay informed and plan my next steps accordingly. | Must |
| 5 | Lead | to give my questions and concerns regarding the admissions clearly and easily | so that can I have the right information regarding the admissions quickly | Should |

### **User Classes and Characteristics**

Table VI User Classes and Characteristics

|  |  |
| --- | --- |
| *Roles* | *Description* |
| Admission Officer | These users are responsible for managing the admissions process. They evaluate and updates the status of the applicant’s applications, they send details from inquiries and assign scholarships. |
| Applicants | These users are the ones who wants to apply to the school. These users are the central focus of the system, their information is stored and managed within the system. |
| Leads | These users are prospective students that pose interest towards the institution. These users are the ones who inquire about the school. |

### **Data Flow Diagram**

A diagram of a application

Description automatically generated

Figure 4 Data Flow Diagram: Context

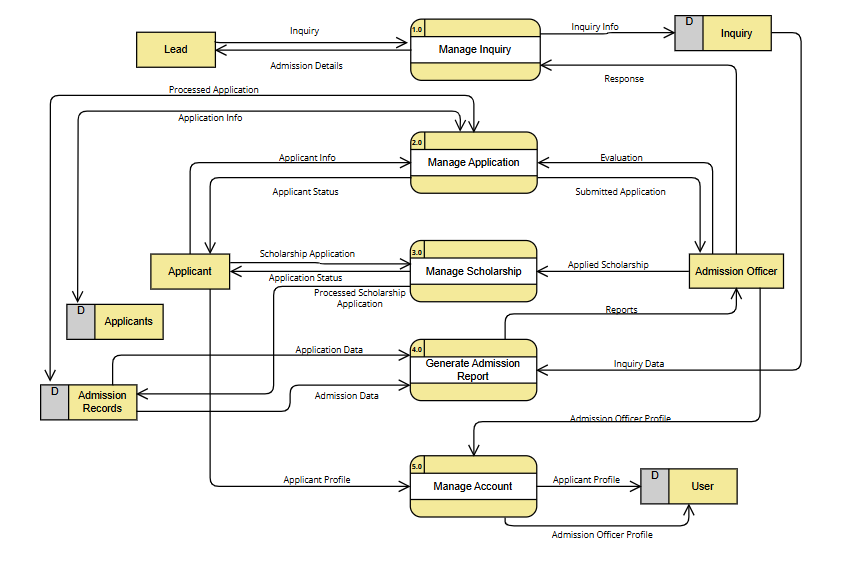


Figure 5 Data Flow Diagram: Level 1

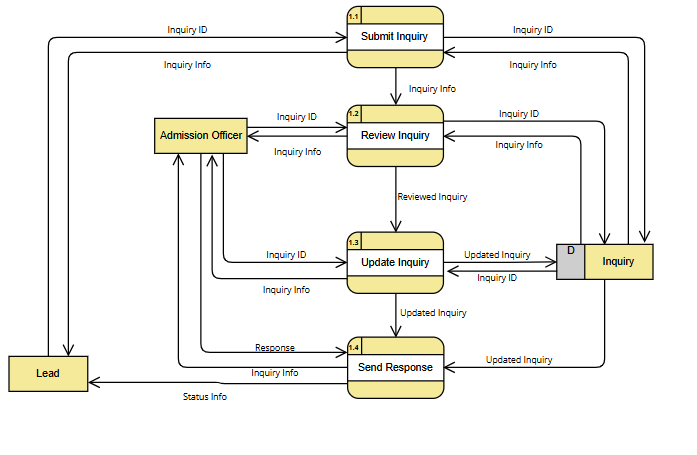


Figure 6 Data Flow Diagram: Level 2 Manage Inquiries

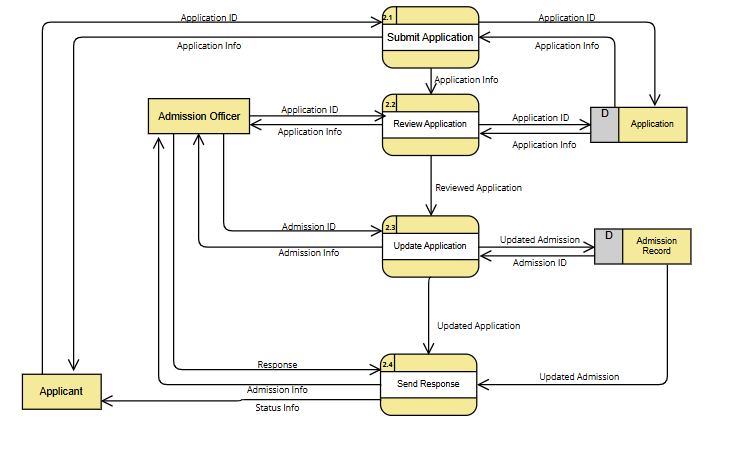


Figure 7 Data Flow Diagram: Level 2 Manage Application

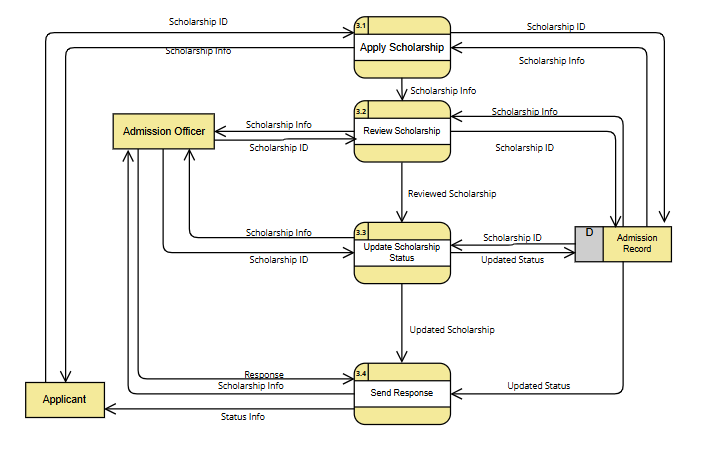


Figure 8 Data Flow Diagram: Level 2 Manage Scholarship

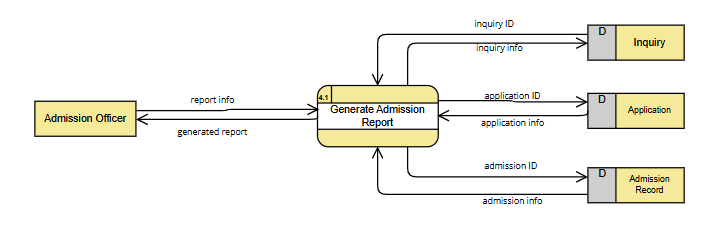


Figure 9 Data Flow Diagram: Generate Admission Reports

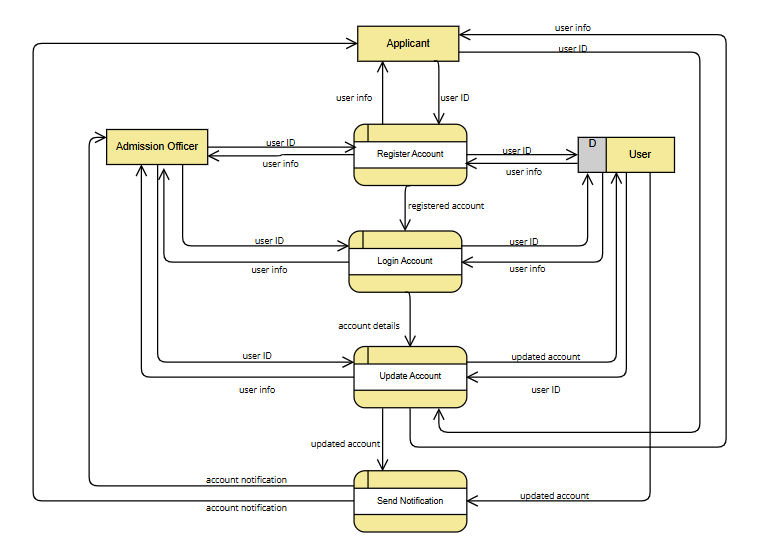


Figure 10 Data Flow Diagram: Level 2 Manage Account

### **Use Case Diagram**

A diagram of a company

Description automatically generated

Figure 11 Use Case Digram

This figure shows the use case diagram of InnolabAMS

|  |  |
| --- | --- |
| **Use Case ID** | **UC\_1** |
| **Use Case Name** | Manage Inquiries |
| **Date created:** | 10-7-2024 |
| **Description:** | This use case allows the Admission Officer to manage inquiries from prospective students or their guardians. They can send information about the admission requirements, tuition fees, scholarships, and school newsletters. |
| **Primary Actor** | 1. Lead 2. Admission Officer |
| **Secondary Actor** | 1. Mail Service |
| **Triggers** | 1. A lead accesses the online admission login page. 2. The Admission Officer logs in to the system and navigates to “Inquiries” module. |
| **Included use cases:** |  |
| **Preconditions:** | 1. The Admission Officer must be logged into the system. 2. An inquiry must be submitted and available in the system. |
| **Postconditions:** | 1. The admission details and newsletter have been sent to the lead. 2. The lead receives the admission details via email. |
| **Main Success Scenario (or Basic Flow):** | 1. The lead clicks the “Submit Inquiry”. 2. The actors fill in all the fields (e.g., personal details, contact information, inquiry) 3. The lead submits the inquiry by clicking the button “Submit” 4. The Admission Officer enters the admission module interface 5. The Admission Officer navigates to the Inquiries item from the admin panel. 6. The system displays all the new inquiries. 7. The Admission Officer selects an inquiry that needs to be processed. 8. The system displays the full details of the inquiry. 9. The Admission Officer can navigate to the lead’s additional information. 10. The Admission Officer clicks the ‘Send Admission Details button’ 11. The Admission Details will pop up and the options are displayed (eg. Admission Requirements", "Tuition Fees", "Financial Aid/Scholarship Opportunities", "Campus Tours", "Programs”). 12. The Admission Officer clicks what admission details need to be sent on the check box. 13. The selected option is highlighted 14. The actor clicks confirm 15. The admission details are confirmed and sent through email to the lead. 16. The system changes the inquiry status to “Responded” |

|  |  |
| --- | --- |
| **Use Case ID** | **UC\_2** |
| **Use Case Name** | Manage Application |
| **Date created:** | 10-7-2024 |
| **Description:** | This use case allows the Admission Officer to process applications by evaluating the application, setting remarks, updating and notifying applicants of their application status. |
| **Primary Actor** | 1. Applicant 2. Admission Officer |
| **Secondary Actor** | 1. Mail Service |
| **Triggers** | 1. An applicant accesses the online admission login page. 2. The Admission Officer needs to process an application. |
| **Included use cases:** |  |
| **Preconditions:** | 1. The Admission Officer must be logged into the system. 2. The Applicant must be logged into the system 3. The application must be submitted and available in the system. |
| **Postconditions:** | 1. The applicant has successfully submitted their application. 2. The application status is updated with the evaluation remarks given. 3. The applicant is notified of their application status via email. |
| **Main Success Scenario (or Basic Flow):** | 1. The applicant accesses the “Personal Information” item in the applicant panel. 2. The system displays the current information that has been saved by the applicant.    1. If the system did not detect an information from the applicant it will display a button. 3. The applicant clicks the button “Add my Personal Information” 4. The system displays the Personal Information form. 5. The applicant fills in all the required fields. 6. The system will validate information completeness. 7. The applicant clicks “Save” to save the information. 8. The system saves the information and display the information as editable. 9. The system redirects the applicant to the next item. 10. The applicant accesses the “Educational Background” item in the applicant panel.     1. The system displays the current information that has been saved by the applicant. 11. If the system did not detect an information, from the applicant it will display a button. 12. The applicant clicks the button “Add my Educational Background”. 13. The system displays the Educational Information form. 14. The applicant fills in all the required fields. 15. The system will validate information completeness. 16. The applicant clicks “Save” to save the information and proceed. 17. The system saves the information and display the information as editable. 18. The system redirects the applicant to the next item. 19. The system saves information. 20. The applicant accesses the “Family Information” item in the applicant panel.     1. The system displays the current information that has been saved by the applicant. 21. If the system did not detect an information, from the applicant it will display a button. 22. The applicant clicks the button “Add my Family Information”. 23. The system displays the family information form. 24. The applicant fills in all the required fields. 25. The system will validate information completeness. 26. The applicant clicks “Save” to save the information and proceed. 27. The system saves the information and display the information as editable. 28. The system redirects the applicant to the next item. 29. The system saves information. 30. The applicant accesses the “Documents” item in the applicant panel. 31. The system displays the current information that has been saved by the applicant.     1. If the system did not detect an information, from the applicant it will display a button. 32. The applicant clicks the button “Add my Documents” 33. The applicant uploads all required supporting documents (eg. Form 137, PSA, Transcript of Record) 34. The system will validate information completeness. 35. The applicant clicks “Save” to save the information and proceed. 36. The system saves the information and display the information as editable. 37. The system redirects the applicant to the next item. 38. The system redirects to the “Additional Information” item in the applicant panel. 39. The system displays the current information that has been saved by the applicant.     1. If the system did not detect an information, from the applicant it will display a button. 40. The applicant clicks the button “Add my Additional Information” 41. The applicant fills in all the required fields. 42. The system will validate information completeness. 43. The applicant clicks “Save” to save the information. 44. The system saves the information and display the information as editable. 45. The system redirects the applicant to the next item. 46. The system detects if all the items are completed. 47. The save button turns into “Submit my Application” 48. The actor reviews and click the button to finalize the application. 49. The system displays a confirmation message on the screen, and a confirmation email is sent to the applicant. 50. The admission Officer enters the admission module interface 51. The admission officer navigates to the Applications item from the system. 52. The system displays all the new applications. 53. The admission Officer selects an application that needs to be processed. 54. The system displays the full details of the application, including the submitted documents and necessary information. 55. The admission Officer evaluates the application to ensure all required documents and information are provided and meet the necessary criteria. 56. The admission Officer can navigate to the applicant’s additional information 57. The admission Officer adds evaluation remarks to the application. 58. The admission Officer updates the application status (e.g, Approved, Rejected, Pending) 59. The admission Officer displays action confirmation. 60. The admission Officer clicks confirm. 61. The system saves the updated application status and remark to the application. 62. The system automatically moves the application to the destined status (Accepted, Rejected, Pending). |
| **Alternate Flow:** | 10, 20, 31, 41, 50a. If the User wants to get back to the previous item, they can click the “Previous Button”  58a. If the admission officer wants to get back to the applications item, they can click the “Back Button” |

|  |  |
| --- | --- |
| **Use Case ID** | **UC\_3** |
| **Use Case Name** | Manage Scholarship |
| **Date created:** | 10-20-2024 |
| **Description:** | This use case describes the process by which the Applicant and Admission Officer manages scholarships, including submitting, assigning, editing, deleting scholarships. |
| **Primary Actor** | 1. Admission Officer 2. Applicant |
| **Secondary Actor** | 1. Mail Service |
| **Triggers** | 1. The Admission Officer accesses the scholarship management module. 2. The Applicant accesses the scholarship management module. |
| **Included use cases:** | None |
| **Preconditions:** | 1. The Admission Officer must be logged into the system. 2. The applicant must be logged into the system. 3. The applicant must be admitted or accepted. |
| **Postconditions:** | 1. The scholarship information is updated, created, or deleted in the system. 2. Applicants receive notifications about the scholarship status |
| **Main Success Scenario (or Basic Flow):** | 1. The applicant accesses the “My Scholarship” module in the applicant panel. 2. The system displays the current scholarship application that has been submitted by the applicant. 3. If the system did not detect a scholarship application, from the applicant it will display a button. 4. The applicant clicks the button “Apply Scholarship” 5. The system displays Scholarship Application form. 6. The applicant can choose if they want to fill out the application form or submit endorsement letters. 7. The system validates information completeness 8. The applicant clicks “Submit my Application” button 9. The system saves the information. 10. The admission officer logs into the system 11. The system retrieves the data from the applicant’s account. 12. The system displays the data from the scholarship application form 13. The admission officer selects an application. 14. The system displays full view of the application 15. The admission officer clicks the dropdown “Assign Scholarship” and choose what scholarship type does the applicant qualify (eg. Public School, Financial-based, Academic Excellence). 16. The system displays the amount of discount for the type of scholarship that was applied 17. The admission officer confirms the action. 18. The system saves the updates. 19. The system will send notification to the applicant information about the scholarship application. |

|  |  |
| --- | --- |
| **Use Case ID** | **UC\_4** |
| **Use Case Name** | Generate Admission Report |
| **Date created:** | 10-7-2024 |
| **Description:** | This use case allows the admission officer to view various analytical reports, charts, and insights generated from the submitted applications. The dashboard provides an overview of key metrics and admission trends. |
| **Primary Actor** | 1. Admission Officer |
| **Secondary Actor** | 1. Data Visualization Tool |
| **Triggers** | 1. The system automatically redirects to the system dashboard when the Admission Officer logs in. 2. The Admission Officer wants to see an overview of the key metrics of the admissions from the system. |
| **Included use cases:** | None |
| **Preconditions:** | 1. The Admission Officer must be logged into the system. 2. The system must have an application data available for analysis. |
| **Postconditions:** | 1. The reports are successfully displayed and viewed by the Admission Officer. |
| **Main Success Scenario (or Basic Flow):** | 1. The system redirects the user to the system’s Dashboard. 2. The system retrieves and displays data visualization of admission reports. 3. The actor views different reports (e.g., total applications, application status, admission trends etc.). 4. The actor can filter the data or adjust the time period to view specific trends. 5. The system updates the dashboard based on the actor’s adjustments. |

|  |  |
| --- | --- |
| **Use Case ID** | **UC\_5** |
| **Use Case Name** | Manage Account |
| **Date created:** | 10/25/2024 |
| **Description:** | This use case describes the process of adding or registering an account and updating profile of the users. |
| **Primary Actor** | 1. Admission Officer 2. Applicant |
| **Secondary Actor** |  |
| **Triggers** | 1. The Applicant needs to register an account. 2. The Admission Officer or Applicant accesses the “Account” module. 3. The admission Officer needs to add an admin account |
| **Included use cases:** | None |
| **Preconditions:** | 1. The Applicant must have an access to the online admission Register page 2. The actors are logged into the system. |
| **Postconditions:** | 1. The applicant’s account is successfully registered 2. The applicant’s account information is successfully updated in the system. 3. The Admission Officer’s account has been added to the system |
| **Main Success Scenario (or Basic Flow):** | 1. The applicant clicks “Submit Application” button 2. The system displays the login form. 3. If the actor doesn’t have an account yet they should click the “Register” link. 4. The system displays the register form. 5. The applicant fills in the required field. 6. The system validates the fields based on its field types and conditions. 7. The applicant clicks “Register” 8. The actor will redirect to the application page. 9. The applicant navigate to the “Account” module at the top right corner of the users dropdown menu. 10. The system displays account details and options for modifications. 11. The users updates the relevant account information, or profile details. 12. The system saves the changes made to the account. 13. The admission officer navigate to the “Users” Item. 14. The system displays all the users. 15. The admission officer clicks “Add account” button 16. The admission officer fills in the credential and assigns the role. 17. The system displays confirmation for the changes. 18. The system saves the changes. |
| **Alternate flow:** | 9a. The users can click their profile picture to redirect them to the account management module. |

### **5.5 Test Cases for Fully Dressed Cases**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test Case ID: | | TC\_001 | | |
| Created By: | | Matthew Paren | | |
| Reviewed By: | | Reycel John Emmanuel Carcueva | | |
| Test Case Description: | | Process of adding or registering an account and updating profile of the users. | | |
| Test Case Scenario: | |  | | |
| Version: | | 1.0 | | |
| Prerequisites | | | | |
| The Applicant must have an access to the online admission Register page. | | | | |
| The actors are logged into the system. | | | | |
| Test Data | | | | |
| name=John Doe | | | | |
| email=johndoe@example.com | | | | |
| Password=Password123 | | | | |
| Step Details | | | | |
| Step Detail | Expected Results | | Actual Results | Status |
| The applicant clicks “Submit Application” button | The system processes the request and displays the login form | |  | Not Executed |
| The system displays the login form. | The login form appears with fields for the applicant to enter their credentials along with a "Register" link for new users. | | The login form appears with fields for entering email and password, along with a “Register” link for new users. | Pass |
| If the actor doesn’t have an account yet they should click the “Register” link. | The system routes registration form | | The system redirects to the registration form, | Pass |
| The system displays the register form. | The register form appears with all necessary fields, | | The system displays all required fields. | Pass |
| The applicant fills in the required field | The fields are fillable and can be seen clearly. | | The system correctly captures input in each field. | Pass |
| The system validates the fields based on its field types and conditions. | The system checks each field’s input | | The system performs validation for each field | Pass |
| The applicant clicks “Register” | The system submits the registration data, | | The registration data is submitted | Pass |
| The actor will redirect to the application page. | The application page appears, confirming the registration is successful and ready for the applicant to proceed with the application process. | | The system successfully redirects to the application page | Pass |
| The applicant navigates to the “Account” module at the top right corner of the users dropdown menu. | The user dropdown menu should open, showing the “Account” option. | | The user dropdown menu opens | Pass |
| The system displays account details and options for modifications. | The system should load the account management page | | The system loads the account management page with the user’s profile information and available options to update fields like email and password. | Pass |
| The users update the relevant account information, or profile details. | The fields should be fillable | | The system allows for editing in the specified fields, | Pass |
| The system saves the changes made to the account | Upon saving, the system should successfully update the user’s profile | | The system updates the user’s profile | Pass |
| The admission officer navigate to the “Users” Item. | The system loads the “Users” page | | The “Users” item has been loaded | Pass |
| The system displays all the users. | The list of users should appear in an organized table or list view | | The list of users appeared. | Pass |
| The admission officer clicks “Add account” button | The system should open an “Add Account” form | |  | Not Executed |
| The admission officer fills in the credential and assigns the role. | The fields should accept valid input | |  | Not Executed |
| The system displays confirmation for the changes. | The system should show a confirmation message | |  | Not Executed |
| The system saves the changes. | The system should save the new user account to the database, updates the “Users” list to reflect the addition. | |  | Not executed |

|  |  |  |
| --- | --- | --- |
| Test Case ID: | TC\_002 | |
| Created By: | Matthew Paren | |
| Reviewed By: |  | |
| Test Case Description: | Process of the submitting and managing an application. | |
| Test Case Scenario: |  | |
| Version: |  | |
| Prerequisites | | |
| The Admission Officer must be logged into the system. | | |
| The Applicant must be logged into the system | | |
| The application must be submitted and available in the system. | | |
| Test Data | | |
| apply\_program= High School | | applicant\_photo= photo1.jpg |
| applicant\_grade\_Level= 10 | | father\_name= Larry |
| applicant\_surname= Doe | | father\_surname= Doe |
| applicant\_given\_name= John | | father\_middle\_name= Frye |
| applicant\_middle\_name= Bean | | father\_date\_of\_birth= 07/21/1988 |
| applicant\_middle\_initial= B | | father\_occupation= Military |
| applicant\_extension= NULL | | father\_contact\_number= 09453651257 |
| applicant\_date\_of\_birth= 02/16/2015 | | mother\_name= Alice |
| applicant\_place\_of\_birth= Quezon City | | mother\_surname= Doe |
| applicant\_ethnicity=NULL | | mother\_middle\_name= Vejano |
| applicant\_gender= Male | | mother\_occupation= Teacher |
| applicant\_civil\_status= Single | | mother\_contact\_number= 096584587475 |
| applicant\_street\_name= 7010 Zapota St | | guardian\_info=”Same as Mother” |
| applicant\_province= Metro Manila | | guardian\_name= Alice |
| applicant\_barangay= Pinagkaisahan | | guardian\_surname= Doe |
| applicant\_city= Makati City | | guardian\_middle\_name= Vejano |
| applicant\_nationality= Filipino | | guardian\_address= 7010 Zapota St |
| applicant\_religion= Catholic | | guardian\_province= Metro Manila |
| applicant\_telephone\_number= NULL | | guardian\_city= Makati City |
| applicant\_mobile\_number= 094533265425 | | guardian\_contact\_number= 096584587475 |
| applicant\_email= johndoe@example.com | | sibling\_full\_name= Ramon Mabuti |
| vaccination\_status= 2nd Dose | | sibling\_date\_of\_birth= 02/15/2017 |
| lrn= 136712120064 | | sibling\_age= 7 |
| sped= NO | | sibling\_gender= Male |
| pwd= NO | | sibling\_grade\_level= 8 |
| last\_school\_name= School Of Makati | | sibling\_school\_attended= School Of Makati |
| last\_school\_address= 1120 South Cembo Makati City | | document1= document1.jpg |
| last\_school\_grade\_level= 9 | | document2= document2.jpg |
| GWA= 90.5 | |  |
| award\_honor= Top 3 | |  |
| Step Details | | |

|  |  |  |
| --- | --- | --- |
| Test Case ID: | TC\_003 | |
| Created By: | Matthew Paren | |
| Reviewed By: |  | |
| Test Case Description: | Verifies the successful creation, viewing, and responding to an inquiry in the "Manage Inquiries" module | |
| Test Case Scenario: |  | |
| Version: |  | |
| Prerequisites | | |
| The Admission Officer must be logged into the system. | | |
| The Applicant must be logged into the system | | |
| The application must be submitted and available in the system. | | |
| Test Data | | |
| lead\_surname=Smith | | detail\_sent= Tuition Fees, Campus Tours |
| lead\_given\_name=John | | inquiry\_status=Responded |
| lead\_extension=N/A | |  |
| lead\_address\_city=Makati City | |  |
| lead\_mobile\_number=09123456789 | |  |
| lead\_email= johnsmith@example.com | |  |
| inquiry\_details=Tuition Fees, Campus Tours | |  |
| lead\_message= | |  |
| inquiry\_submitted= 2024-10-25 | |  |
| lead\_surname=Smith | |  |
| lead\_given\_name=John | |  |
| Step Details | | |

|  |  |
| --- | --- |
| Test Case ID: | TC\_004 |
| Created By: | Matthew Paren |
| Reviewed By: |  |
| Test Case Description: | Verifies the successful creation, viewing, and responding to an inquiry in the "Manage Inquiries" module |
| Test Case Scenario: |  |
| Version: |  |
| Prerequisites | |
| The Admission Officer must be logged into the system. | |
| The applicant must be logged into the system. | |
| The applicant must be admitted or accepted. | |
| Test Data | |
| current\_scholarship= Lani Scholarship | |
| annual\_household\_income= 20,000.00 | |
| applicant\_signature= applicantsignature.jpg | |
| parent\_signature= parentsignature.jpg | |
| scholarship\_document=document.jpg | |
| scholarship\_type= Financial-based | |
| discount\_awarded= 0.30 | |
| Step Details | |

|  |  |
| --- | --- |
| Test Case ID: | TC\_005 |
| Created By: | Matthew Paren |
| Reviewed By: |  |
| Test Case Description: | Admission officer to view various analytical reports, charts, and insights generated from the submitted applications. The dashboard provides an overview of key metrics and admission trends. |
| Test Case Scenario: |  |
| Version: |  |
| Prerequisites | |
| The Admission Officer must be logged into the system. | |
| The system must have an application data available for analysis. | |
| Test Data | |
| academic\_program=ABM | |
| month=june | |
| academic\_year=2024-2025 | |
| age=13 | |
| gender= Male | |
| Step Details | |

### **Activity Diagram**

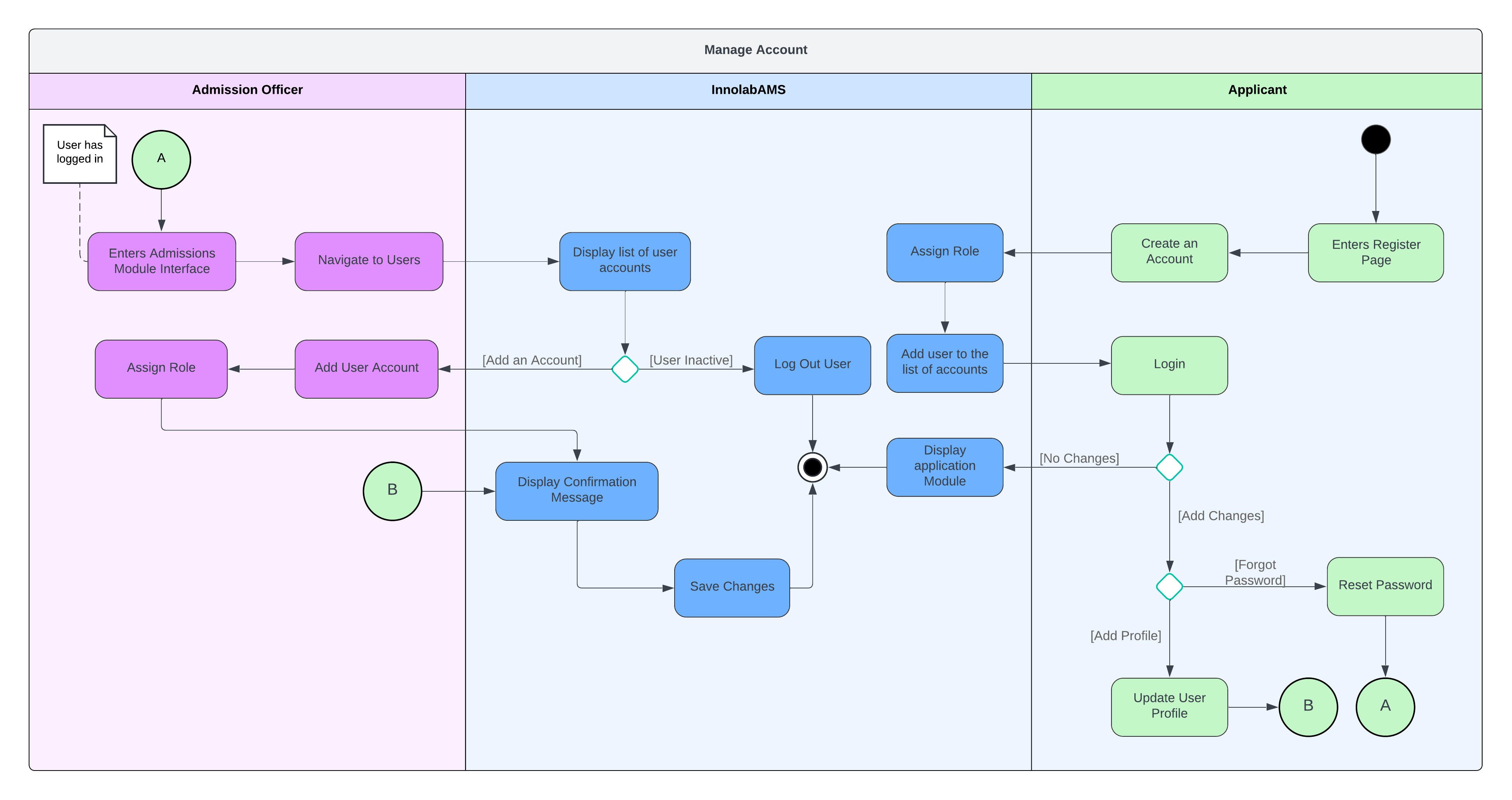


Figure 12 Manage Account

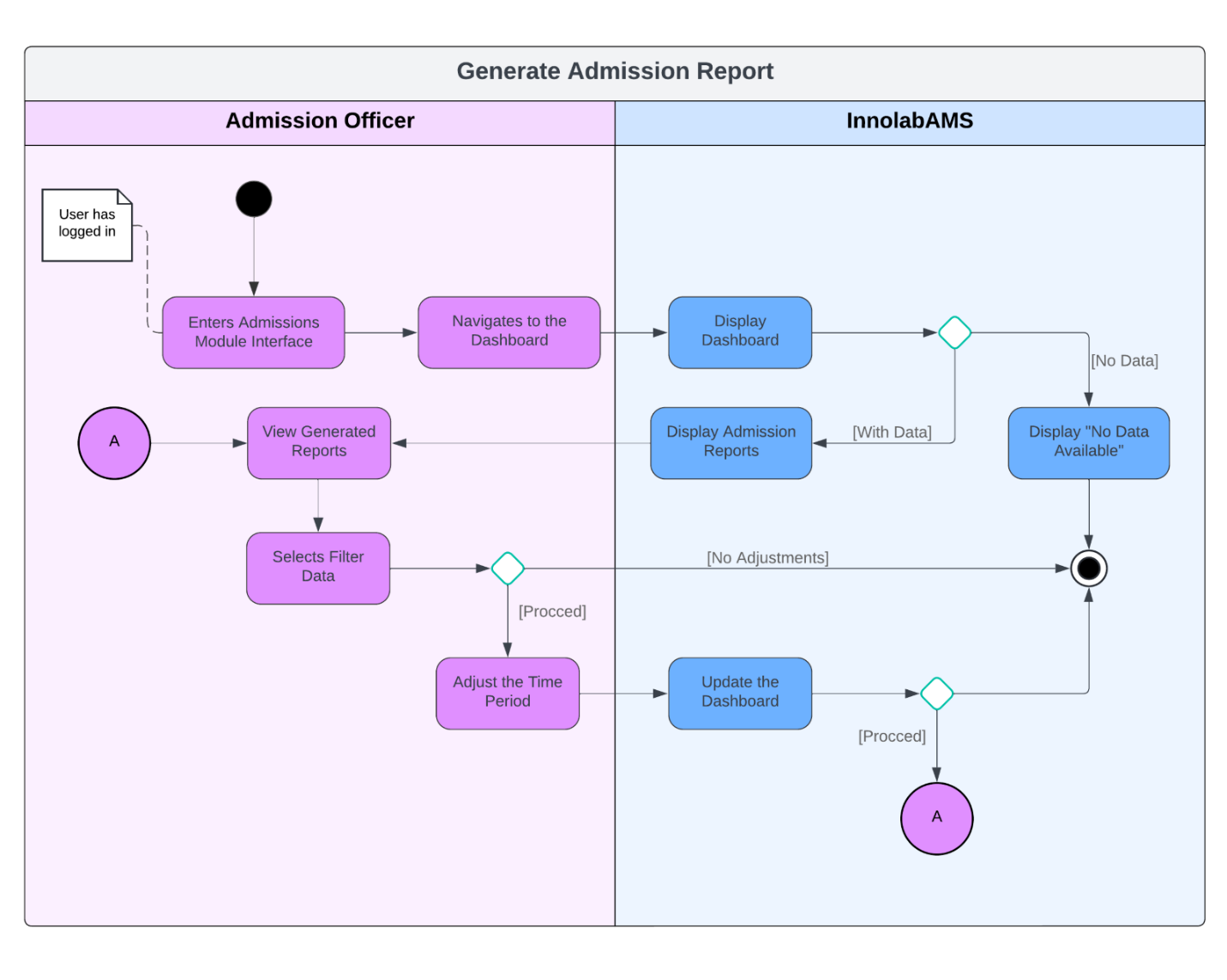


Figure 13 Generate Admission Report

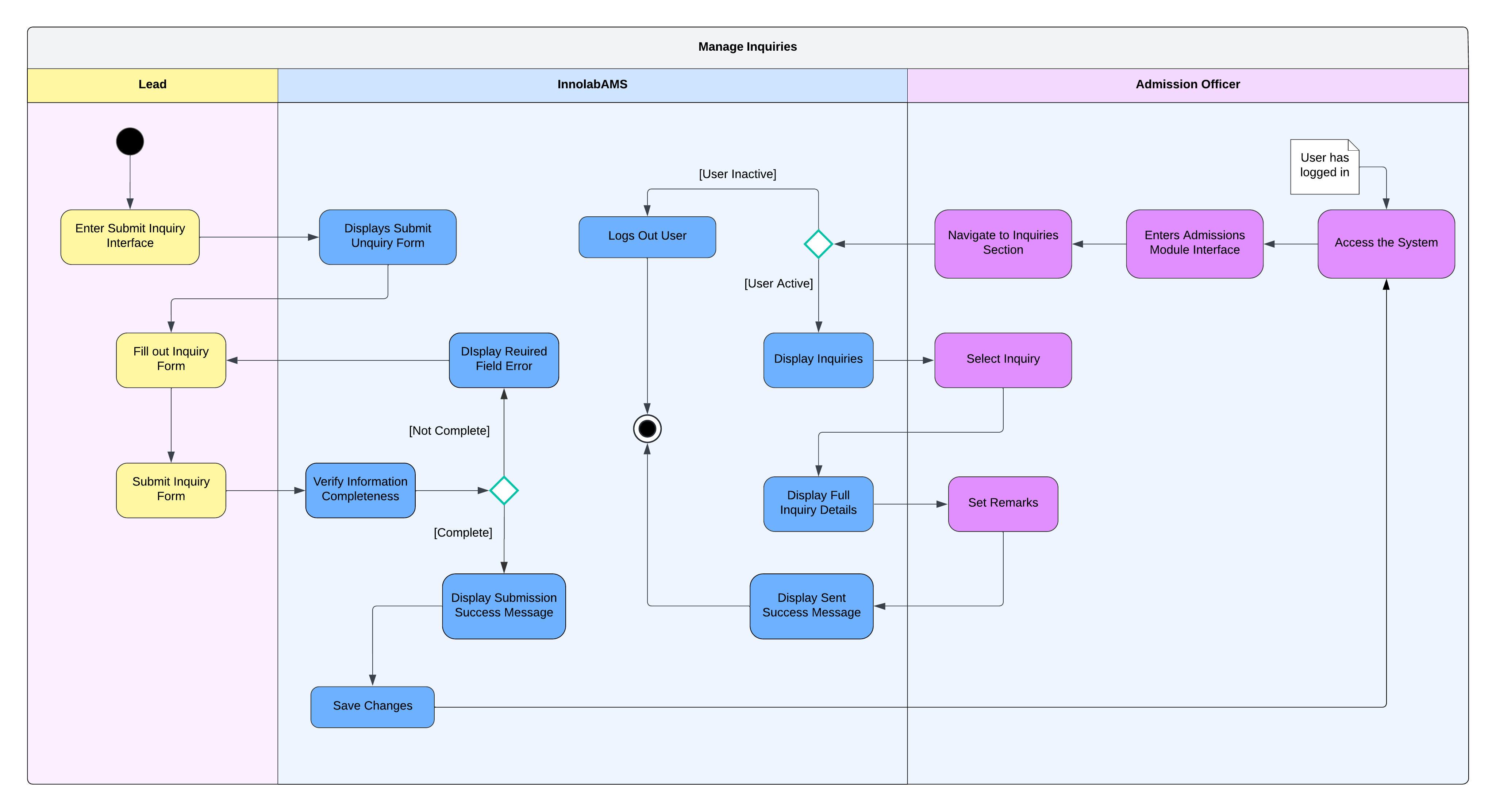


Figure 14 Manage Inquiries

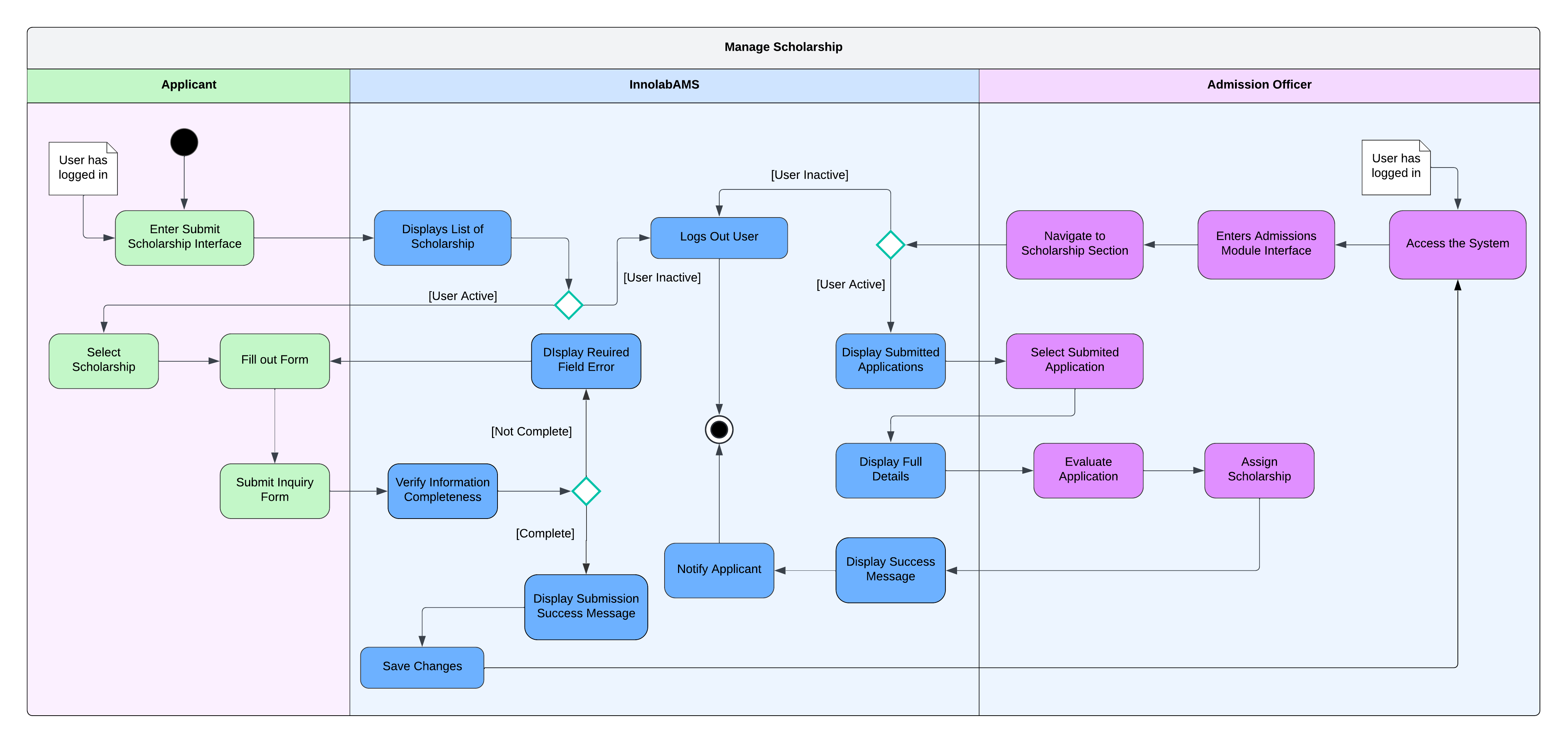


Figure 15 Manage Scholarship

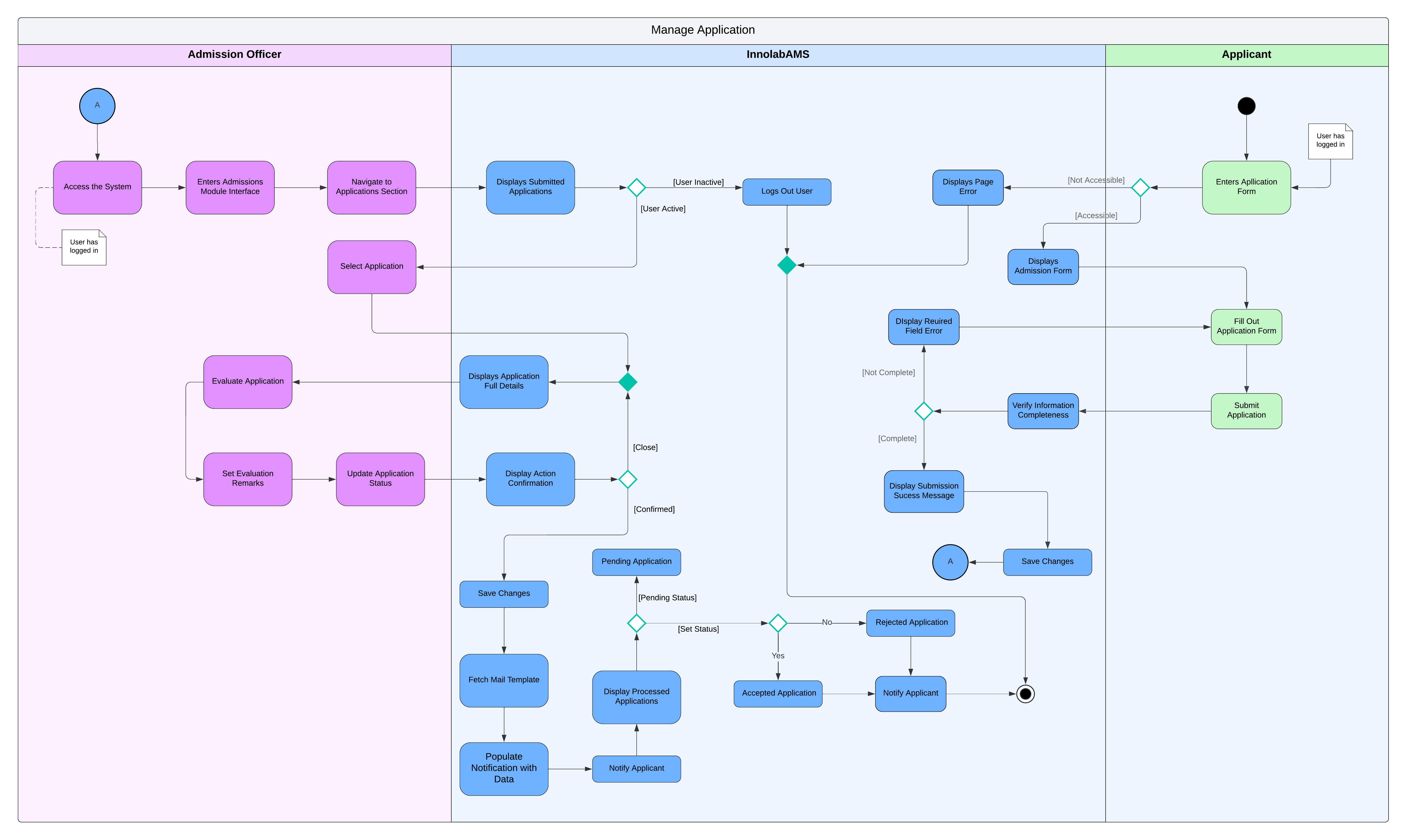


Figure 16 Manage Application

### **Entity Relationship Diagram with Data Dictionary**

A diagram of a computer program

Description automatically generated with medium confidence

Figure 17 Entity Relationship Diagram

A diagram of a diagram

Description automatically generated with medium confidence

Figure 18 Entity Relationship Diaram Tables

**users Table**

|  |  |  |  |
| --- | --- | --- | --- |
| **Column** | **Data Type** | **Constraints** | **Description** |
| id | INT (PRIMARY KEY) | AUTO\_INCREMENT | Unique identifier for each user. |
| name | VARCHAR(255) | NOT NULL | The full name of the user |
| email | VARCHAR(255) | NOT NULL | The email of the user |
| email\_verified\_at | TIMESTAMP | NOT NULL | Time that user has verified their email. |
| password | VARCHAR(255) | NOT NULL | The password of the user. |
| remember\_token | VARCHAR(255) | NOT NULL | The token for users in persistent-login. |
| created\_at | TIMESTAMP | NOT NULL | The exact time and date the account was created. |
| role | ENUM (‘Applicant’, ‘Admission Officer) | NOT NULL | Defines if the user is an applicant or office |
| last\_login\_at | TIMESTAMP | NULL | The exact time and date the user login. |

**applicant\_info Table**

|  |  |  |  |
| --- | --- | --- | --- |
| **Column** | **Data Type** | **Constraints** | **Description** |
| id | INT (PRIMARY KEY) | AUTO\_INCREMENT | Unique identifier for each record |
| user\_id | INT (FOREIGN KEY) | REFERENCES users(id) | Links the applicant\_info to the corresponding user in the users table |
| apply\_program | ENUM (‘Kindergarten’, ‘Elementary’, ‘High School’, ‘Senior High School’ | NOT NULL | The program the applicant is applying fo |
| apply\_grade\_level | ENUM(‘1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12) | NOT NULL | The grade level the applicant is applying to |
| apply\_strand | ENUM(STEM, ABM, TECHVOC, HUMSS, GAS) | NULL | The academic strand for the applicant |
| applicant\_surname | VARCHAR(255) | NOT NULL | Applicant’s last name |
| applicant\_given\_name | VARCHAR(255) | NOT NULL | Applicant’s first name |
| applicant\_middle\_name | VARCHAR(255) | NULL | Applicant’s middle name |
| applicant\_extension | VARCHAR(10) | NULL | Applicant’s extension name (eg. Jr., II, III) |
| applicant\_date\_birth | DATE | NOT NULL | Applicant’s date of birth |
| gender | ENUM('Male', 'Female',) | NOT NULL | Applicant’s gender |
| applicant\_address\_street | |  | | --- | | VARCHAR(255) | | NOT NULL | Applicant’s street address |
| applicant\_address\_province | VARCHAR(255) | NOT NULL | Province where the applicant resides |
| applicant\_address\_city | VARCHAR(255) | NOT NULL | City where the applicant resides |
| applicant\_nationality | VARCHAR(255) | NOT NULL | Applicant’s nationality |
| applicant\_religion | VARCHAR(255) | NULL | Applicant’s religion |
| applicant\_mobile\_number | INT (12) | NULL | Applicant’s mobile phone number |
| applicant\_photo | VARCHAR(255) | NULL | Path or URL to the applicant’s uploaded photo |
| applicant\_id | INT (FOREIGN KEY) | REFERENCES applicant\_info(id) | Links applicant additional info to the applicant\_info table |
| extracurricular\_interest | ENUM('Sports', 'Music', 'Art', 'Drama', 'Debate', 'Science Club', 'Math Club', 'Student Government', 'Volunteering', 'Dance', 'Technology Club') | NOT NULL | The extracurricular activities the applicant is interested in |
| skills | ENUM('Communication', 'Teamwork', 'Leadership', 'Problem-Solving', 'Time Management', 'Creativity', 'Adaptability' ‘Technology-related’) | NOT NULL | Specific skills that the applicant has acquired |
| hobbies | VARCHAR(255) | NULL | Hobbies or personal interests of the applicant |
| participations | VARCHAR(255) | NOT NULL | Notable events or activities the applicant has participated in |
| competitions | VARCHAR(255) | NOT NULL | Competitions the applicant has been involved in (if any) |
| referral\_source | ENUM('Social Media', 'Alumni', 'Online Ad', 'Website', 'School Fair', 'Other') | NULL | How the applicant learned about the program or institution |

**educational\_background Table**

|  |  |  |  |
| --- | --- | --- | --- |
| **Column** | **Data Type** | **Constraints** | **Description** |
| id | INT (PRIMARY KEY) | AUTO\_INCREMENT | Unique identifier for each record |
| applicant\_id | INT (FOREIGN KEY) | REFERENCES applicant\_info(id) | Links educational\_ background table to the applicant\_info table |
| lrn | INT | NULL | Learner Reference Number (LRN) assigned by the educational institution program |
| sped | BOOLEAN | NOT NULL | Indicates if the applicant is a Person with Disability (SPED) program (1 for yes, 0 for no) |
| pwd | BOOLEAN | NOT NULL | Indicates if the applicant is a Person with Disability (PWD) (1 for yes, 0 for no) |
| applicant\_school\_name | VARCHAR(255) | NULL | Name of the school the applicant last attended |
| applicant\_school\_address | VARCHAR(255) | NNULL | Address of the applicant's previous school |
| applicant\_last\_grade\_level | ENUM(‘Kindergarten  1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11) | NULL | Last grade level completed by the applicant |
| applicant\_year\_graduation | DATE | NULL | Year of graduation from the applicant's previous school |
| applicant\_gwa | NUMERIC | NOT NULL | General Weighted Average (GWA) or overall GPA of the applicant |
| applicant\_achievements | VARCHAR(225) | NULL | Any notable achievements of the applicant. |

**family\_info Table**

|  |  |  |  |
| --- | --- | --- | --- |
| **Column** | **Data Type** | **Constraints** | **Description** |
| id | INT (PRIMARY KEY) | AUTO\_INCREMENT | Unique identifier for each record |
| applicant\_id | INT (FOREIGN KEY) | REFERENCES applicant\_info(id) | Links family\_info table applicant\_info table |
| father\_surname | VARCHAR(255) | NULL | Father's last name |
| father\_given\_name | VARCHAR(255) | NULL | Father's first name |
| father\_middle\_name | VARCHAR(255) | NULL | Father's middle name |
| father\_occupation | VARCHAR(255) | NULL | Father's occupation |
| father\_contact\_num | INT | NULL | Father's contact number |
| mother\_surname | VARCHAR(255) | NULL | Mother's last name |
| mother\_given\_name | VARCHAR(255) | NULL | Mother's first name |
| mother\_middle\_name | VARCHAR(255) | NULL | Mother's middle name |
| mother\_occupation | VARCHAR(255) | NULL | Mother's occupation |
| mother\_contact\_num | INT | NULL | Mother's contact number |
| guardian\_info | ENUM (‘Same as Father’, ’Same as Mother’) | NULL | Autofill indicator of the guardian info if it’s the same as the applicant’s mother or father. |
| guardian\_surname | VARCHAR(255) | NOT NULL | Guardian's last name |
| gurdian\_given\_name | VARCHAR(255) | NOT NULL | Guardian's first name |
| guardian\_middle\_name | VARCHAR(255) | NOT NULL | Guardian's middle name |
| guardian\_address\_street | VARCHAR(255) | NULL | Guardian's street address |
| guardian\_address\_city | VARCHAR(255) | NULL | Guardian's city address |
| guardian\_contact\_num | VARCHAR(255) | NOT NULL | Guardian's contact number |
| guardian\_email | VARCHAR(255) | NULL | Guardian's email address |

**siblings\_info**

|  |  |  |  |
| --- | --- | --- | --- |
| sibling\_surname | VARCHAR(255) | NULL | Sibling's last name |
| sibling\_given\_name | VARCHAR(255) | NULL | Sibling's first name |
| sibling\_age | INT | NULL | Sibling's age |
| sibling\_school\_name | VARCHAR(255) | NULL | Name of the school where the sibling is enrolled |
| sibling\_school\_address | VARCHAR(255) | NULL | Address of the sibling's school |
| sibling\_grade\_level | VARCHAR(255) | NULL | Grade level of the sibling |

**application Table**

|  |  |  |  |
| --- | --- | --- | --- |
| **Column** | **Data Type** | **Constraints** | **Description** |
| id | INT (PRIMARY KEY) | AUTO\_INCREMENT | Unique identifier for each record |
| applicant\_id | INT (FOREIGN KEY) | REFERENCES applicant\_info(id) | Links the applicant\_info to the application table |
| admission\_officer\_id | INT (FOREIGN KEY) | REFERENCES admission\_officer(id) | Links the admission\_officer to the application table |
| application\_submitted | TIMESTAMP | NOT NULL | The exact time that the application was submitted |
| application\_status | ENUM (‘Accepted’, ‘Rejected’, ‘Pending’) | NULL | The status of the application that was updated by the admission officer. |
| application\_remarks | TEXT | NULL | The remarks given to the application. |

**admission\_officer**

|  |  |  |  |
| --- | --- | --- | --- |
| **Column** | **Data Type** | **Constraints** | **Description** |
| id | INT (PRIMARY KEY) | AUTO\_INCREMENT | Unique identifier for each record |
| users\_id | INT (FOREIGN KEY) | REFERENCES applicant\_info(id) | Links the admission\_officer to the users table |
| admission\_officer\_name | VARCHAR(225) | NOT NULL | Full name of the admission officer. |
| admission\_officer\_photo | VARCHAR(225) | NULL | Path or URL to the admission officer’s uploaded photo |

**lead\_info**

|  |  |  |  |
| --- | --- | --- | --- |
| **Column** | **Data Type** | **Constraints** | **Description** |
| id | INT (PRIMARY KEY) | AUTO\_INCREMENT | Unique identifier for each record |
| lead\_additional\_info\_id | INT (FOREIGN KEY) | REFERENCES lead\_additional\_info (id) | Links the lead\_additional\_info to the lead\_info table |
| lead\_surname | VARCHAR(225) | NOT NULL | Full name of the admission officer. |
| lead\_given\_name | VARCHAR(225) | NOT NULL | Path or URL to the admission officer’s uploaded photo |
| lead\_middle\_name | VARCHAR(225) | NULL | Lead middle name |
| lead\_extension | VARCHAR(10) | NULL | Lead extension name (eg. II, III, Jr) |
| lead\_address\_city | VARCHAR(225) | NULL | Lead current city |
| lead\_mobile\_number | INT | NULL | Lead current mobile number |
| lead\_email | VARCHAR(225) | NULL | Lead current email address |
| inquired\_details | ENUM | NOT NULL | The details that the lead inquired |
| lead\_message | TEXT | NULL | This is the message of a lead (eg. Questions about admission) |
| extracurricular\_interest\_ lead | ENUM('Sports', 'Music', 'Art', 'Drama', 'Debate', 'Science Club', 'Math Club', 'Student Government', 'Volunteering', 'Dance', 'Technology Club') | NULL | Lead extracurricular interests |
| skills\_lead | ENUM('Communication', 'Teamwork', 'Leadership', 'Problem-Solving', 'Time Management', 'Creativity', 'Adaptability' ‘Technology-related’) | NULL | Specific skills that the applicant has acquired |
| desired\_career | VARCHAR(225) | NULL | Lead desired career |

**inquiry Table**

|  |  |  |  |
| --- | --- | --- | --- |
| **Column** | **Data Type** | **Constraints** | **Description** |
| id | INT (PRIMARY KEY) | AUTO\_INCREMENT | Unique identifier for each record |
| lead\_id | INT (FOREIGN KEY) | REFERENCES lead\_id (id) | Links the lead\_info to the inquiry table |
| admission\_officer\_id | INT (FOREIGN KEY) | REFERENCES admission\_officer\_id (id) | Links the admission\_officer to the inquiry table |
| inquiry\_submitted | VARCHAR(225) | NULL | Lead desired career |
| details\_sent | VARCHAR(225) |  | The details that the admission officer has sent |
| response\_date | TIMESTAMP |  | The exact time when the admission officer sent the response |
| inquiry\_status | ENUM('New', 'Responded') |  | The status of the inquiry |

**application\_table**

|  |  |  |  |
| --- | --- | --- | --- |
| **Column** | **Data Type** | **Constraints** | **Description** |
| id | INT (PRIMARY KEY) | AUTO\_INCREMENT | Unique identifier for each record |
| applicant\_id | INT (FOREIGN KEY) | REFERENCES applicant\_info(id) | Links the applicant\_info to the application table |
| admission\_officer\_id | INT (FOREIGN KEY) | REFERENCES admission\_officer(id) | Links the admission\_officer to the application table |
| application\_submitted | TIMESTAMP | NOT NULL | The exact time that the application was submitted |
| application\_status | ENUM (‘Accepted’, ‘Rejected’, ‘Pending’) | NULL | The status of the application that was updated by the admission officer. |
| application\_remarks | TEXT | NULL | The remarks given to the application. |
| updated\_at | TIMESTAMP | NOT NULL | The exact time that the application was updated. |

**admission\_officer Table**

|  |  |  |  |
| --- | --- | --- | --- |
| **Column** | **Data Type** | **Constraints** | **Description** |
| id | INT (PRIMARY KEY) | AUTO\_INCREMENT | Unique identifier for each record |
| users\_id | INT (FOREIGN KEY) | REFERENCES applicant\_info(id) | Links the admission\_officer to the users table |
| admission\_officer\_name | VARCHAR(225) | NOT NULL | Full name of the admission officer. |
| admission\_officer\_photo | VARCHAR(225) | NULL | Path or URL to the admission officer’s uploaded photo |

**lead\_info**

|  |  |  |  |
| --- | --- | --- | --- |
| **Column** | **Data Type** | **Constraints** | **Description** |
| id | INT (PRIMARY KEY) | AUTO\_INCREMENT | Unique identifier for each record |
| lead\_additional\_info\_id | INT (FOREIGN KEY) | REFERENCES lead\_additional\_info (id) | Links the lead\_additional\_info to the lead\_info table |
| lead\_surname | VARCHAR(225) | NOT NULL | Full name of the admission officer. |
| lead\_given\_name | VARCHAR(225) | NOT NULL | Path or URL to the admission officer’s uploaded photo |
| lead\_middle\_name | VARCHAR(225) | NULL | Lead middle name |
| lead\_date\_birth | DATE | NULL | Lead date of birth |
| lead\_address\_city | VARCHAR(225) | NULL | Lead current city |
| lead\_nationality | VARCHAR(225) | NULL | Lead current nationality |
| lead\_mobile\_number | INT | NULL | Lead current mobile number |
| lead\_email | VARCHAR(225) | NULL | Lead current email address |
| inquired\_details | ENUM | NOT NULL | The details that the lead inquired |

**lead\_additional\_info Table**

|  |  |  |  |
| --- | --- | --- | --- |
| **Column** | **Data Type** | **Constraints** | **Description** |
| id | INT (PRIMARY KEY) | AUTO\_INCREMENT | Unique identifier for each record |
| extracurricular\_interest\_ lead | ENUM('Sports', 'Music', 'Art', 'Drama', 'Debate', 'Science Club', 'Math Club', 'Student Government', 'Volunteering', 'Dance', 'Technology Club') | NULL | Lead extracurricular interests |
| skills\_lead | ENUM('Communication', 'Teamwork', 'Leadership', 'Problem-Solving', 'Time Management', 'Creativity', 'Adaptability' ‘Technology-related’) | NULL | Specific skills that the applicant has acquired |
| desired\_career | VARCHAR(225) | NULL | Lead desired career |

**inquiry Table**

|  |  |  |  |
| --- | --- | --- | --- |
| **Column** | **Data Type** | **Constraints** | **Description** |
| id | INT (PRIMARY KEY) | AUTO\_INCREMENT | Unique identifier for each record |
| lead\_id | INT (FOREIGN KEY) | REFERENCES lead\_id (id) | Links the lead\_info to the inquiry table |
| admission\_officer\_id | INT (FOREIGN KEY) | REFERENCES admission\_officer\_id (id) | Links the admission\_officer to the inquiry table |
| inquiry\_submitted | VARCHAR(225) | NOT NULL | Lead desired career |
| details\_sent | VARCHAR(225) | NOT NULL | The details that the admission officer has sent |
| response\_date | TIMESTAMP | NOT NULL | The exact time when the admission officer sent the response |
| inquiry\_status | ENUM('New', 'Responded') | NOT NULL | The status of the inquiry |

**applicant\_scholarship**

|  |  |  |  |
| --- | --- | --- | --- |
| **Column** | **Data Type** | **Constraints** | **Description** |
| id | INT (PRIMARY KEY) | AUTO\_INCREMENT | Unique identifier for each record |
| applicant\_info\_id | INT (FOREIGN KEY) | REFERENCES applicant\_info\_id (id) | Links the applicant\_info\_id to the applicant\_scholarship |
| current\_scholarship | VARCHAR(225) | NULL | The current scholarship that the applicant has |
| annual\_household\_income | ENUM | NOT NULL | The annual household income that the applicant has |
| applicant\_signature | VARCHAR(225) | NOT NULL | The virtual signature of the applicant |
| parent\_signature | VARCHAR(225) | NOT NULL | The virtual signature of the parent |
| created\_at | TIMESTAMP | NOT NULL | The exact time that the scholarship application is created. |
| scholarship\_document | VARCHAR(225) | NULL | The document that the applicant has (eg. Endorsement letter, Scholarship form) |

**scholarship\_application**

|  |  |  |  |
| --- | --- | --- | --- |
| **Column** | **Data Type** | **Constraints** | **Description** |
| id | INT (PRIMARY KEY) | AUTO\_INCREMENT | Unique identifier for each record |
| applicant\_info\_id | INT (FOREIGN KEY) | REFERENCES applicant\_info\_id (id) | Links the applicant\_info\_id to the scholarship\_application table |
| admission\_officer\_id | INT (FOREIGN KEY) | REFERENCES admission\_officer\_id (id) | Links the admission\_officer\_id to the scholarship\_application table |
| scholarship\_type | ENUM | NOT NULL | The scholarship type that was assigned |
| discount\_awarded | FLOAT | NOT NULL | The discount that was awarded |
| updated\_at | TIMESTAMP | NOT NULL | The exact time that the application save changes. |

### **Prototype/Wireframe**

**A screenshot of a computer

Description automatically generated**

Figure 19 Wireframe: Welcome Page

*Figure 12* shows the initial welcome page of the InnolabAMS system. It prominently features the school’s logo and name. The page has two main actions for users: Apply and Inquire. These options guide applicants and prospective students through the primary functions of the system, ensuring easy navigation for new visitors.

**A screenshot of a login form

Description automatically generated**

Figure 20 Wireframe: Sign In

*Figure 13* shows the sign-in interface of InnolabAMS, where users are prompted to enter their Username and Password. The includes fields for entering credentials. Below the input fields, there are additional options such as Forgot your password for password recovery and Create an account for new users.

A screenshot of a computer

Description automatically generated

Figure 21 Wireframe: Register

*Figure 14* shows the user registration interface for InnolabAMS, where new users can create an account. The form contains fields for Name, Email, Password, and Confirm Password. At the bottom of the form, there is a Register button to complete the registration process and a prompt for users who are Already registered? to navigate to the sign-in page.

**A screenshot of a computer

Description automatically generated**

Figure 22 Wireframe: Inquiry Form

*Figure 15* shows the inquiry form designed for prospective students or their guardians to ask questions or seek information regarding the school and its admission process. The form includes fields for First Name, Last Name, Middle Name, Email, and Contact Number. Additionally, there is a dropdown for selecting What details you would like to know?, followed by a Message box for elaborating on specific inquiries.

**A screenshot of a computer

Description automatically generated**

Figure 23 Wireframe: Personal Information

*Figure 16* shows the personal information form on the applicant panel page of the InnolabAMS system. In this section, applicants can input their personal information, such as Full Name, Date of Birth, Gender, Nationality, and Contact Details. This section ensures that all personal information required for the application is collected accurately.

A screenshot of a computer

Description automatically generated

Figure 24 Wireframe: Educational Background

*Figure 17* shows the educational background form on the applicant panel page of the InnolabAMS system. In this section, applicants provide details about their previous schooling, including the last school attended, grade level completed, and academic achievements. This form gathers essential academic information necessary for processing applications.

**A screenshot of a computer

Description automatically generated**

Figure 25 Wireframe: Family Information

*Figure 18* shows the family information form on the applicant panel page of the InnolabAMS system. In this section, applicants can enter details about their parents or guardians, including names, occupations, and contact information. This section collects key family-related data required during the application process.

**A screenshot of a computer

Description automatically generated**

Figure 26 Wireframe: Documents

*Figure 19* shows the documents upload form on the applicant panel page of the InnolabAMS system. This section allows applicants to upload necessary supporting documents, such as birth certificates, transcripts, and proof of residency. It ensures that all required files are attached to the application efficiently.

**A screenshot of a computer

Description automatically generated**

Figure 27 Wireframe: Additional Information

*Figure 20* shows the additional information form on the applicant panel page of the InnolabAMS system. In this section, applicants can provide details about extracurricular interests, skills, and other activities. This form gathers non-academic information, providing a more comprehensive profile of the applicant.

**A screenshot of a computer

Description automatically generated**

Figure 28 Wireframe: Scholarship

*Figure 21* shows the scholarship application form on the applicant panel page of the InnolabAMS system. Here, applicants can apply for scholarships by entering relevant information and attaching required documents. This section gives applicants the opportunity to apply for financial aid or other support programs.

A screenshot of a computer

Description automatically generated

Figure 29 Wireframe: Dashboard

*Figure 22* shows the dashboard of the InnolabAMS system. This section provides administrators with an overview of key metrics, including Total Submitted Applications, Accepted Applications, and Rejected Applications. The dashboard displays graphical insights like Admission Trends and Acceptance Rates, enabling administrators to analyze the number of applications over different academic years. Additionally, the dashboard includes information on inquiries categorized by channels such as Social Media, Website, and Referrals. A Conversion Rate chart visualizes the proportion of leads converted to applicants, giving administrators a comprehensive view of admission performance and trends.

**A screenshot of a computer

Description automatically generated**

Figure 30 Wireframe: Admissions

*Figure 23* shows the admissions page of the InnolabAMS system. This section allows administrators to manage various stages of the application process, including new applications, accepted applications, and rejected applications. Administrators can access a list of all current applicants and update their status based on the evaluation criteria. This page provides essential features to monitor and control the flow of admission efficiently.

**A computer screen shot of a application

Description automatically generated**

Figure 31 Wireframe: Admissions Full View

*Figure 24* displays the full view of an individual application within the InnolabAMS system. This section shows detailed information about an applicant’s profile, including personal details, uploaded documents, and additional remarks. Administrators can thoroughly review and update the application’s status, ensuring a comprehensive evaluation process.

**A screenshot of a computer

Description automatically generated**

Figure 32 Wireframe: Scholarship

*Figure 25* shows the scholarship management page in the InnolabAMS system. Administrators can review, assign, and update scholarships based on applicants' eligibility. This section includes fields to categorize different types of scholarships, such as financial-based, merit-based, or special grants.

**A screenshot of a computer

Description automatically generated**

Figure 33 Wireframe: Inquiries

*Figure 26* shows the inquiries page of the InnolabAMS system. This section allows administrators to handle incoming inquiries from prospective students and their guardians. Administrators can view inquiry details, respond with relevant information, and track the status of each inquiry to ensure timely communication and engagement.

**A screenshot of a computer

Description automatically generated**

Figure 34 Wireframe: Users

*Figure 27* presents the users management page of the InnolabAMS system. This section lists all the system users, including applicants and administrators. Administrators can add, edit, or deactivate user accounts to maintain appropriate access levels and ensure the integrity of the system.

**A screenshot of a computer

Description automatically generated**

Figure 35 Wireframe: Account Management

*Figure 28* shows the account settings page of the InnolabAMS system. This section enables administrators to update their profile details, modify passwords, and change contact information. The account settings page ensures that users’ profiles remain accurate and up-to-date, providing a secure and personalized system experience.

### **Technology Stack**

This project is currently using the TALL stack—Tailwind CSS, Alpine.js, Laravel, and Livewire. This technology stack supports the system's efficiency, scalability, and user experience, making it an ideal choice for building an admission management system.

The following technologies supports the key aspects of system:

*Laravel*

Backend Logic and Structure - Laravel forms the foundation of the AMS, handling server-side logic, managing user data and application records.

Authentication - Laravel’s built-in authentication simplifies secure access control, with roles of the system.

Notifications and Communication - With Laravel’s notifications system, the system can automatically notify students about important updates via email.

*Livewire*

**Real-Time, Reactive UI** -Livewire enhances the user experience by providing reactive, real-time updates without requiring a JavaScript-heavy setup. Users can interact with various system components like form validation, application tracking, and status changes without page reloads, creating a smooth and engaging experience.

Data-Driven Form Handling - Livewire handles dynamic forms, such as the application form, with real-time validation. This can reduce errors and increase efficiency as applicants progress through the admission process.

*Alipine.js*

Enhanced Interactivity - Alpine.js adds lightweight interactivity to the system, perfect for handling UI elements like dropdowns, modals, and form interactions without needing to add complex JavaScript frameworks.

Simplified UI Logic - With Alpine.js, certain parts of the form can dynamically adjust based on user input (e.g., conditional fields for application types), enhancing usability without impacting performance.

*Tailwind CSS*

Efficient Styling - Tailwind CSS provides utility-first classes that make building the user interface faster and more customizable. The system can maintain a clean, responsive design across devices, ensuring a smooth experience for students and staff

Consistency -Tailwind’s responsive grid system also ensures the AMS is accessible and user-friendly on various screen sizes.

### **GitHub Repository**

<https://github.com/APC-SoCIT/APC-2024-2025-T1-05-Admission-Management-System>

A screenshot of a computer

Description automatically generated

### **Updates of contents in Project GitHub Repository**

Issues:

1. Total Issues Made – 12

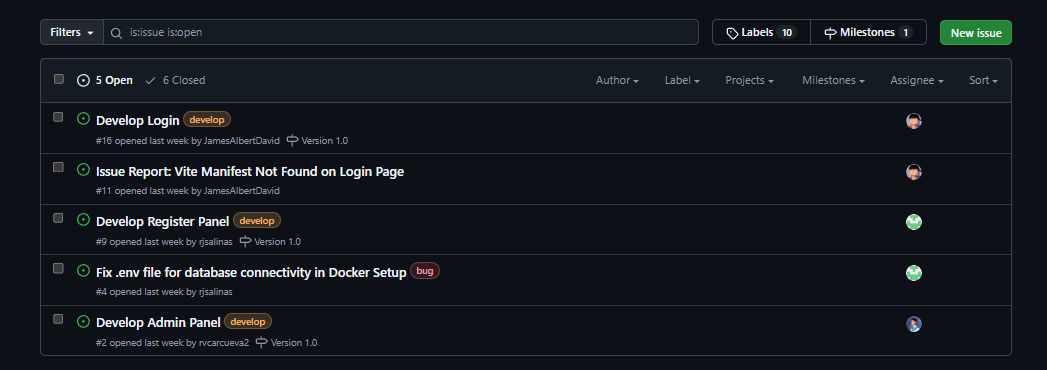
James Albert David – 2

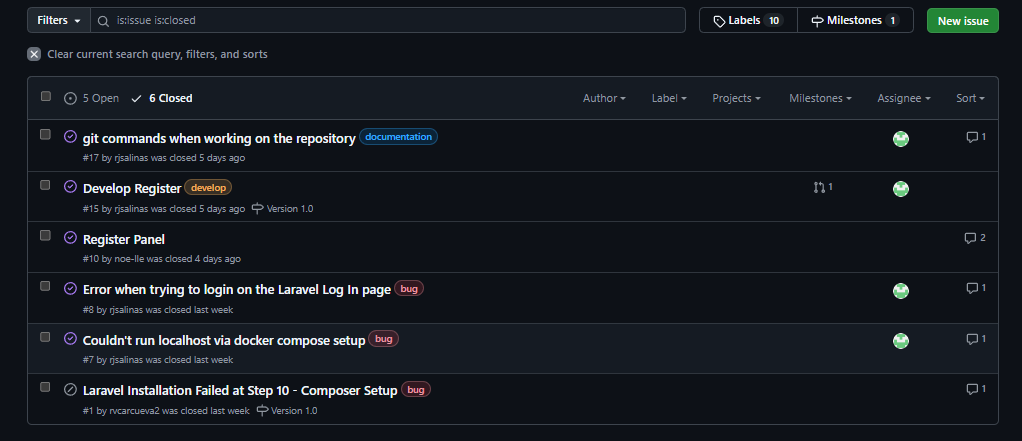
Reycel John Emmanuel Carcueva – 3

Reejay Salinas – 6

Althea Noelle Sarmiento – 1

Matthew Paren -1





Branches:

Total Branches Made - 5

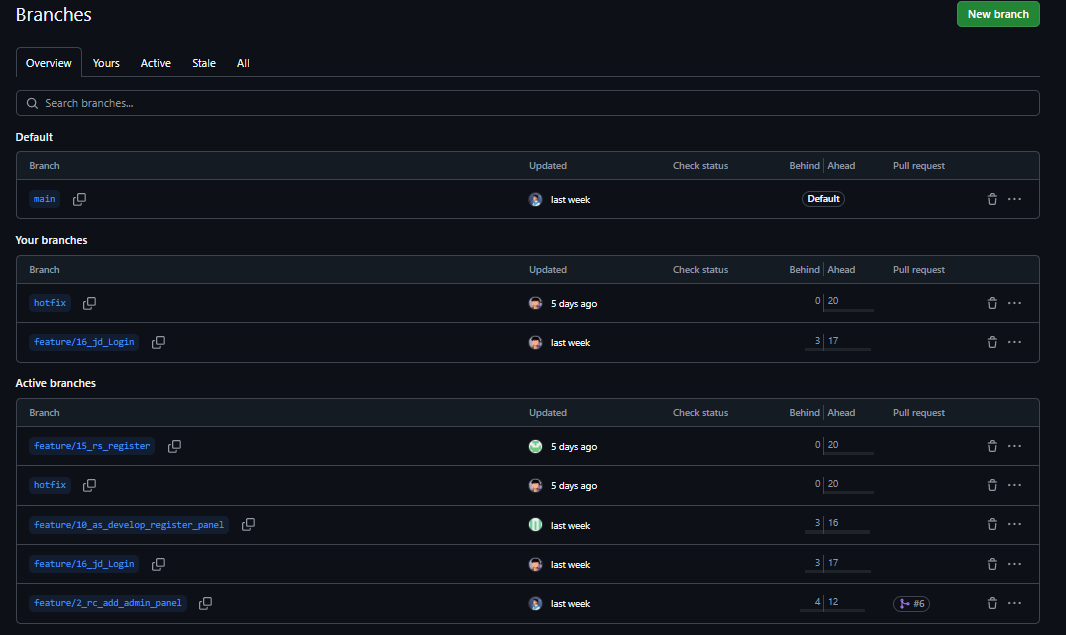
hotfix

feature/2\_rc\_add\_admin\_panel – Reycel John Emmanuel Carcueva

feature/10\_as\_develop\_register\_panel – Althea Noelle Sarmiento

feature/15\_rs\_register – Reejay Salinas

feature/16\_jd\_Login – James Albert David



Pull Requests :

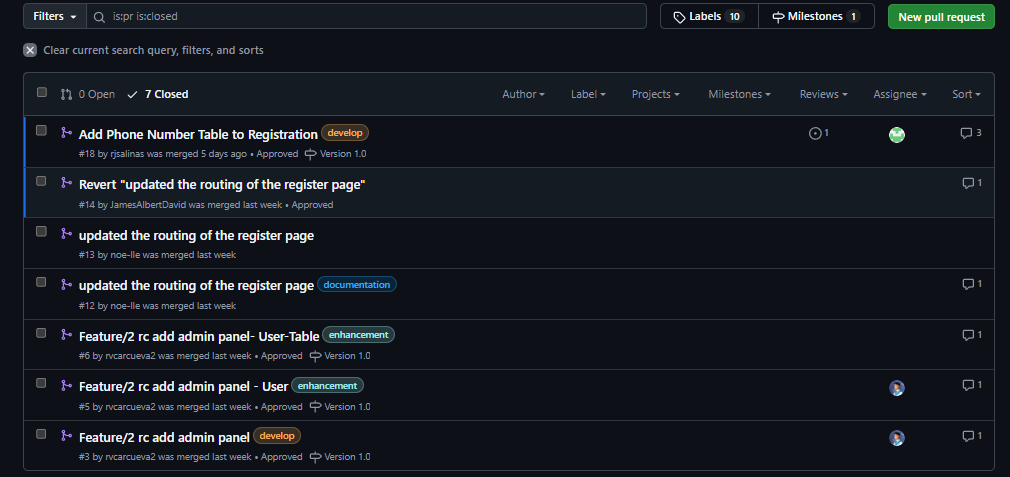
Total Pull Requests Made – 7

Reycel John Emmanuel – 3

James Albert David – 1

Reejay Salinas – 1

Althea Noelle Sarmiento – 2



**Network Graph**

A screenshot of a computer

Description automatically generated

### **Cloud hosted prototype**

<https://squalid-spooky-wraith-g44rqr54r7jq39wg4-80.app.github.dev/>

### **Release Plan**



Figure 36 Release Plan

This figure shows the release plan of InnolabAMS, it is designed with Agile methodologies to ensure flexibility and continuous improvement of the system. The project has a 12-month time frame, initiated in April 2024 and expected to be launched in April 2025. The release plan is divided into four academic terms, each averaging three months. This structured timeline enables the team to focus on specific features and improvements during each term, ensuring a balanced and manageable workload.

# Reference

[1]  “Online Admission Management System Software for School, College & Univ,” www.iitms.co.in. <https://www.iitms.co.in/onlineadmission.html> [Accessed June 25, 2024]

[2] U. A. Shaikh, "Benefits of Student Information System: Transforming Education with QualCampus SIS," Best Education ERP, 31 October 2023. [Online]. Available: https://qualcampus.com/benefits-of-student-information-system-sis/. [Accessed 25 May 2024].

[3] A. B. G. Santos, N. P. Balba and C. B. Rebong, "Attendance Monitoring System of Schools in the Philippines with an Inclusion of Optimization Query Algorithm," June 2021. [Online]. Available: <https://www.ijitee.org/wp-content/uploads/papers/v10i8/H91490610821.pdf>.

[4] S. Thammasane and W. Sattaworn, “THE DEVELOPMENT OF THE FREE ENROLLMENT SYSTEM, UNDERGRADUATE OF SUAN SUNANDHA RAJABHAT UNIVERSITY”, International Academic Multidisciplinary Research Conference in Vienna 2019, March 2019. [Online]. Available: <http://icbtsproceeding.ssru.ac.th/index.php/ICBTSVIENNA/article/view/80>

[5] B. Grepon et’al., “Designing and Implementing e-School Systems: An Information Systems Approach to School Management of a Community College in Northern Mindanao, Philippines” International Journal of Computing Sciences Research, May 2021. Available: Cornell University, <https://arxiv.org/abs/2109.00198>. [Accessed May 30, 2024].

[6] Curacubby Team. “Why Your School Needs Online Enrollment + How to Set It Up,” curacubby.com. Accessed: June 15, 2024. [Online.] Available: <https://www.curacubby.com/resources/your-school-needs-online-enrollment>

[7] U. A. Shaikh, "Benefits of Student Information System: Transforming Education with QualCampus SIS," Best Education ERP, 31 October 2023. [Online]. Available: <https://qualcampus.com/benefits-of-student-information-system-sis/>. [Accessed 25 May 2024].

[8] UNESCO, "The Role of Education Management Information Systems (EMIS) in Supporting Educational Outcomes," [Online]. Available: <https://unesdoc.unesco.org/ark:/48223/pf0000374542>. [Accessed 25 May 2024].

[9] La Immaculada Concepcion School, “Enrollment procedures for new students and returnees”, LICS.edu.ph. Accessed: June 24, 2024. [Online.] Available: <https://lics.edu.ph/enrollment-process-for-new-and-returnees/>

[10] Claret School of Quezon City, “Application for Student Admission for Nursery to Grade 11 for SY 2024-2025”, claretschool.edu.ph. Accessed: June 24, 2024. [Online.] Available: <https://claretschool.edu.ph/admissions/application-admissions.html>

[11] Philippine Pasay Chung Hua Academy, “S.Y. 2024-2025 Enrollment Schedule”, ppcha.edu.ph. Accessed: June 24, 2024. [Online.] Available: <https://ppcha.edu.ph/all_announcements/s-y-2024-2025-enrolment-schedule/>

[12] M. Lynch, "Allocating Resources to Improve Student Learning," The Edvocate, 27 April 2016. [Online]. Available: <https://www.theedadvocate.org/allocating-resources-to-improve-student-learning/>. [Accessed 25 May 2024].

[13] R. Jia, Y. Yang, J. Grundy, J. Keung and L. Hao, “A Systematic Review of Scheduling Approaches on Multi-Tenancy Cloud Platforms.” Information and Software Technology, vol. 132, April 2021. Available: Elsevier, <https://www.sciencedirect.com/science/article/abs/pii/S0950584920302214>. [Accessed: June 02, 2024]

[14] S. Kalra and T. V. Prabhakar, “Multi-tenant Quality Attributes to Manage Tenants in SaaS Applications” 2020 IEEE International Conference on Software Architecture Companion (ICSA-C), March 2020. Available: IEEE Xplore, <https://ieeexplore.ieee.org/document/9095620>. [Accessed: June 15, 2024]

[15] S. Kanade and R. Manza, “A Comprehensive Study on Multi Tenancy in SAAS

Applications.” International Journal of Computer Applications (0975 – 8887), vol. 181, no. 44, March 2019. Available: Academia, <https://www.academia.edu/41292916/A_Comprehensive_Study_on_Multi_Tenancy_in_SAAS_Applications>. [Accessed June 01, 2024]

[16] M. Musyary, A. Kurniati and C. Damarjati, “Laravel Framework-Based Information System of the Department of Information Technology of Universitas Muhammadiyah Yogyakarta” Emerging Information Science and Technology. Vol. 4, no. 2, pp. 48-57, December 2023. Available: ResearchGate, <https://www.researchgate.net/publication/377176843_Laravel_Framework-Based_Information_System_of_the_Department_of_Information_Technology_of_Universitas_Muhammadiyah_Yogyakarta>. [Accessed June 15, 2024].

[17] U. Bashar, M. A. Ibrahim, A. Adedeji and A. Dauda, “Optimizing the Student Application Process with a Laravel-based System of Waziri Umaru Federal Polytechnic Birnin Kebbi: A Case Study” International Journal of Innovative Science and Research Technology, October 2023. Available: International Journal of Innovative Science and Research Technology, <https://ijisrt.com/assets/upload/files/IJISRT23OCT305.pdf>. [Accessed June 01, 2024]

[18] K. Malik and K. Hussein, “ONLINE STUDENT ADMISSION REGISTRATION MANAGEMENT SYSTEM. A CASE STUDY OF METROPOLITAN INTERNATIONAL UNIVERSITY”, Metropolitan Journal of Science and Technology, vol. 2, issue 8, November 2023. [Online]. Available: ResearchGate, <https://www.researchgate.net/publication/375962943_ONLINE_STUDENT_ADMISSION_REGISTRATION_MANAGEMENT_SYSTEM_A_CASE_STUDY_OF_METROPOLITAN_INTERNATIONAL_UNIVERSITY>. [Accessed: June 26, 2024]

[19] R. Istiqomah, Kristiani and B. Wahyono, “The Implementation of Online-Based New Student Admissions in Surakarta”, Journal of Physics: Conference Series, March 2021. [Online]. Available: ResearchGate, <https://www.researchgate.net/publication/350211039_The_Implementation_of_Online-Based_New_Student_Admissions_in_Surakarta>. [Accessed: June 26, 2024]

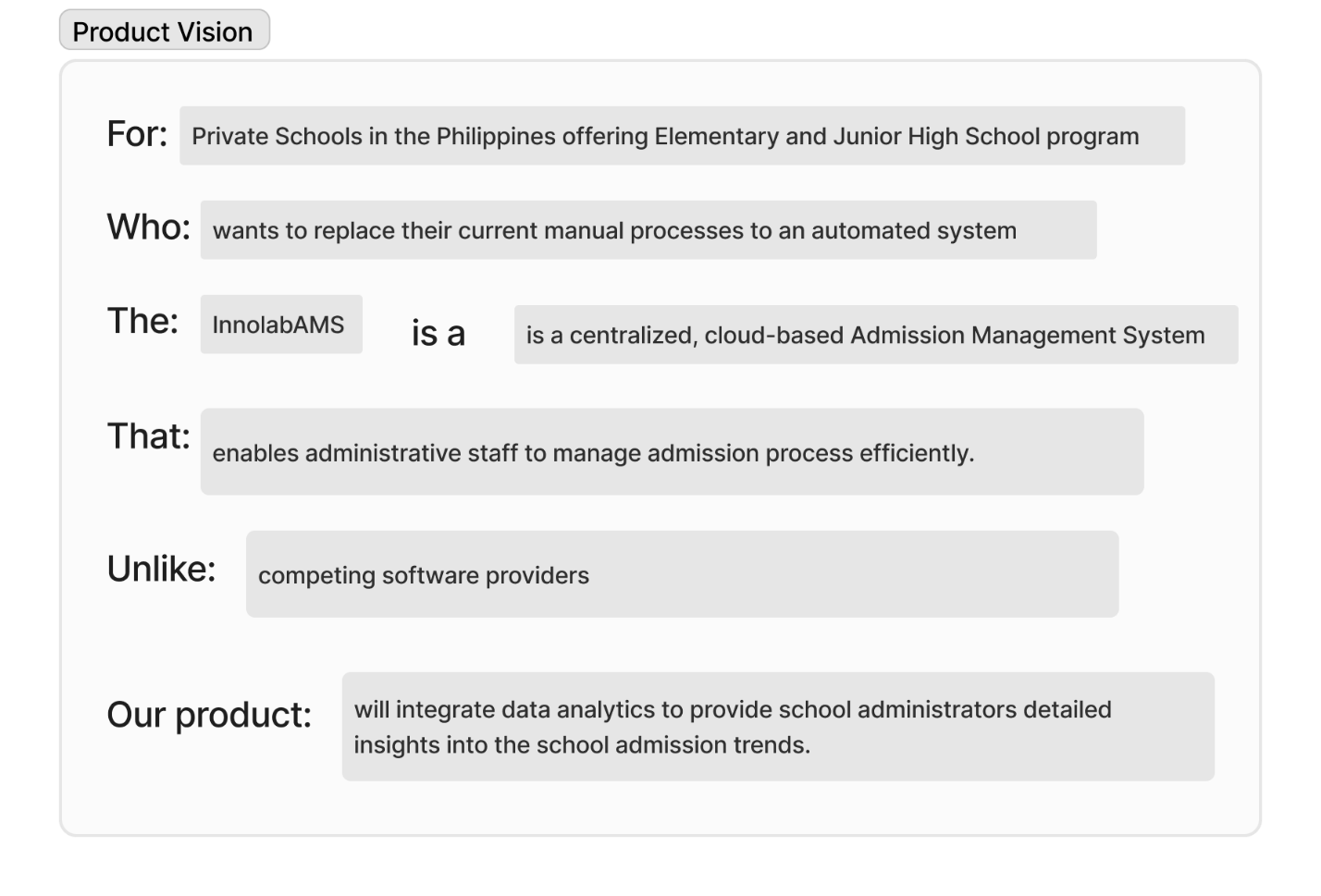
[20] S. H. Abdullah, M. H. M. Puad, M. A. A. Murad and E. Marlisah, “Implementation of Learning Analytics in Primary and Secondary School: A Systematic Literature Review'' International Journal of Academic Research in Progressive Education & Development, vol 11(2), pp. 911-928, May 2022. [Online]. Available: ResearchGate, <https://www.researchgate.net/publication/367682979_Implementation_of_Learning_Analytics_in_Primary_and_Secondary_School_A_Systematic_Literature_Review>. [Accessed June 16, 2024]

[21] J. Cirelli, A. M. Konkol and F. Aqlan, “Predictive Analytics Models for Student Admission and Enrollment”, Proceedings of the International Conference on Industrial Engineering and Operations Management, September 2018. [Online]. Available: ResearchGate, <https://www.researchgate.net/publication/329998695_Predictive_Analytics_Models_for_Student_Admission_and_Enrollment>. [Accessed June 26, 2024]

[22] “Wela online,” wela.online, 2021. <https://wela.online/en/>. [Accessed July 21, 2024]

# Appendices

## Appendix A: Project Vision



## Appendix B: Schedule

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Term** | **Phase** | **Start Date** | **End Date** | **Duration** |
| 3rd | Project Initiation | March 27, 2024 | May 5, 2024 | 40 days |
| 3rd | Project Documentation | May 16, 2024 | June 12, 2024 | 28 days |
| Defense | Defense | June 13, 2024 | June 27, 2024 | 14 days |
| Vacation | Vacation | July 1, 2024 | August 5, 2024 | 35 days |
| 1st | Designing | August 6, 2024 | November 10, 2024 | 97 days |
| 2nd | Development | November 12, 2024 | January 20,2025 | 69 days |
| 2nd | Deployment | January 21, 2025 | February 28, 2025 | 38 days |
| 3rd | Testing | March 3, 2025 | March 20, 2025 | 17 days |
| 3rd | Documentation | March 21, 2025 | April 20, 2025 | 30 days |
| Project Launch | Project Launch | April 30, 2025 (expected) |  |  |

## Appendix C: Product Roadmap

Project Roadmap link:

<https://miro.com/app/board/uXjVKCBHmOg=/?share_link_id=671409884642>

A white grid with colorful boxes

Description automatically generated with medium confidence

## Appendix D: Teams Meetings

**DATE**: MAY 27, 2024   
**TIME STARTED**: 9:00 AM   
**TIME ADJOURNMENT**: 9:30 PM   
**MEDIUM USED**: Microsoft Teams

**ATTENDANCE**

|  |  |
| --- | --- |
| **INNOLAB** | |
| **NAME** | **REMARKS** |
| David, James Albert | Present |
| Carcueva, Reycel John Emmanuel | Present |
| Salinas, Reejay | Present |
| Sarmiento, Althea Noelle | Present |

**AGENDA**  
 Initial discussion with resource person

Here is a summary of how we discussed and finalized our final paper for the Project. (Time Allotted: approximately 30 minutes)

* The group talked with Sir Kimberly about the general problems regarding the current enrollment system of k-12 schools in the Philippines

**PREPARED BY**:

Reejay V. Salinas

**NOTED AND CHECKED BY**:

Leader: James Albert David

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**DATE**: MAY 30, 2024   
**TIME STARTED**: 10:00 AM  
**TIME ADJOURNMENT**: 11:40 PM   
**MEDIUM USED**: Microsoft Teams

**ATTENDANCE**

|  |  |
| --- | --- |
| **INNOLAB** | |
| **NAME** | **REMARKS** |
| David, James Albert | Present |
| Carcueva, Reycel John Emmanuel | Present |
| Salinas, Reejay | Present |
| Sarmiento, Althea Noelle | Present |

**AGENDA**  
 Consultation with Sir Jayvee

Here is a summary of how we discussed and finalized our final paper for the Project. (Time Allotted: approximately 1 hour & 40 minutes)

* The group talked with Sir Jayvee about the whole paper and his opinion about the overall contents of it.
* We clarified what the use case description is
* Finishing and editing of the paper

**PREPARED BY**:

Reejay V. Salinas

**NOTED AND CHECKED BY**:

Leader: James Albert David

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**DATE**: MAY 30, 2024   
**TIME STARTED**: 1:00 PM   
**TIME ADJOURNMENT**: 3:00 PM   
**MEDIUM USED**: Microsoft Teams

**ATTENDANCE**

|  |  |
| --- | --- |
| **INNOLAB** | |
| **NAME** | **REMARKS** |
| David, James Albert | Present |
| Carcueva, Reycel John Emmanuel | Present |
| Salinas, Reejay | Present |
| Sarmiento, Althea Noelle | Present |

**AGENDA**  
 Follow up discussion with resource person

Here is a summary of how we discussed and finalized our final paper for the Project. (Time Allotted: approximately 1 hour)

* The group talked with Sir Kimberly about the general problems regarding the current enrollment system of k-12 schools in the Philippines
* Asked his opinion about the system we will be developing
* Planned the general flow of the system

**PREPARED BY**:

Reejay V. Salinas

**NOTED AND CHECKED BY**:

Leader: James Albert David

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**DATE**: JUNE 14, 2024   
**TIME STARTED**: 1:00 PM   
**TIME ADJOURNMENT**: 2:30 PM   
**MEDIUM USED**: Microsoft Teams

**ATTENDANCE**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **INNOLAB** | | | **NAME** | **REMARKS** | | David, James Albert | Present | | Carcueva, Reycel John Emmanuel | Present | | Salinas, Reejay | Present | | Sarmiento, Althea Noelle | Present | |

|  |  |
| --- | --- |
| **ADVISER** | |
| Castillo, Jojo F. | Present |

**AGENDA**  
 Discussion with Adviser

Here is a summary of how we discussed and finalized our final paper for the Project. (Time Allotted: approximately 1 hour and 30 minutes)

* We talked about how we want to sell the SaaS right after we are done with it, thus having various equations on how much we will sell and make a profit out of it
* Discussion of the innovative aspects of the system and if it is sufficient

**PREPARED BY**:

Reejay V. Salinas

**NOTED AND CHECKED BY**:

Leader: James Albert David

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**DATE**: JUNE 14, 2024   
**TIME STARTED**: 7:00 PM   
**TIME ADJOURNMENT**: 11:00 PM   
**MEDIUM USED**: Microsoft Teams

**ATTENDANCE**:

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **INNOLAB** | | | **NAME** | **REMARKS** | | David, James Albert | Present | | Carcueva, Reycel John Emmanuel | Present | | Salinas, Reejay | Present | | Sarmiento, Althea Noelle | Present | |

**AGENDA**  
 Discussion and Finalization of the Paper

Here is a summary of how we discussed and finalized our final paper for the Project. (Time Allotted: approximately 4 hours)

* The members continue to finalize their parts before we pass the paper on time.
* Use Case and Use Case diagram were constructed.

**PREPARED BY**:

Reejay V. Salinas

**NOTED AND CHECKED BY**:

Leader: James Albert David