|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Arab Academy for Science, Technology & Maritime Transport College of Artificial Intelligence ━━━━━ Alamein Campus.**   |  |  |  |  | | --- | --- | --- | --- | | Course | GN231 | Web Programming |  | | Lecturer | | Dr. Omar Shalash | | | Teaching Assistant | | Eng. Nagy K. Aly | Eng. Aya Abdelhamied |  | |

**Project**

|  |  |
| --- | --- |
| **Team Members** | **Name1: Abdelrahman mamdouh - Reg1: 221010901** |
| **Name2: Ahmed agamy - Reg2: 2210102771** |
| **Name3: Loay waled - Reg3: 221003410** |
| **Name4: omar khaled - Reg4: 221002374** |

|  |  |
| --- | --- |
| **Topic** | **page Number** |
| **Introduction & Objectives** | **2** |
| **Some of Project Images** | **2** |
| **Functional Requirements** | **7** |
| **Non-Functional Requirements** | **12** |
| **Sequence Diagrams** | **16** |
| **Flowchart** | **19** |
| **Conclusion** | **20** |

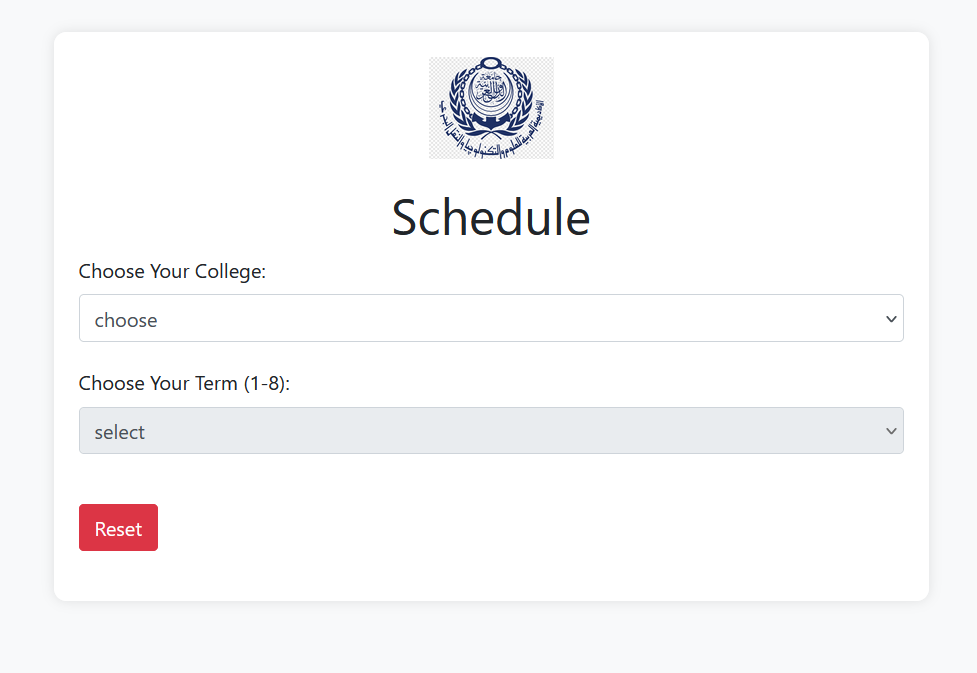
#### **Introduction & Objectives**

This project aims to create a responsive and user-friendly website using HTML and CSS. The primary objectives are to enhance user experience by ensuring that the website is intuitive and easy to navigate. Another objective is to ensure cross-browser compatibility so that the website functions correctly on all popular web browsers. Additionally, The ML web application demonstrates a straightforward approach to dynamically updating content based on user interaction. By using JavaScript to manipulate the DOM and Bootstrap for responsive design, the project achieves its goal of providing an intuitive and interactive user experience. Further enhancements could include server-side integration for persistent data storage and more complex prediction algorithms for educational purposes. And the project aims to implement a clean and modern design to attract and retain users. The website will serve as a platform to showcase various content and features, adhering to best practices in web development.

#### **Some of Project Images**

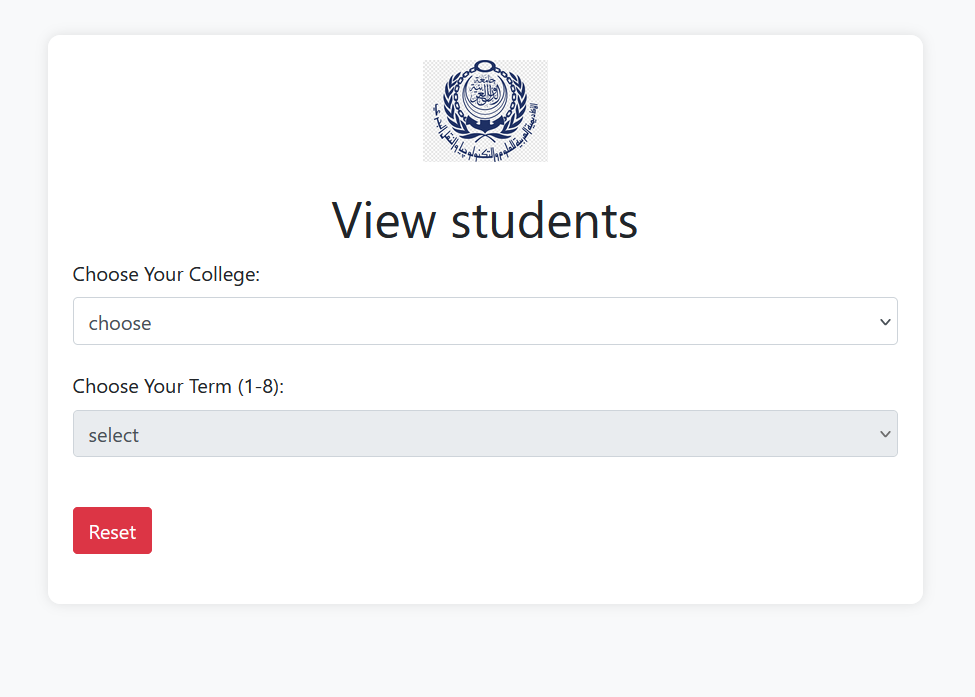
The following images provide visual representations of different parts of the project. These images demonstrate various aspects of the website, including the user interface and key functionalities.

##### **Figure 1: College Schedule Selection Interface**



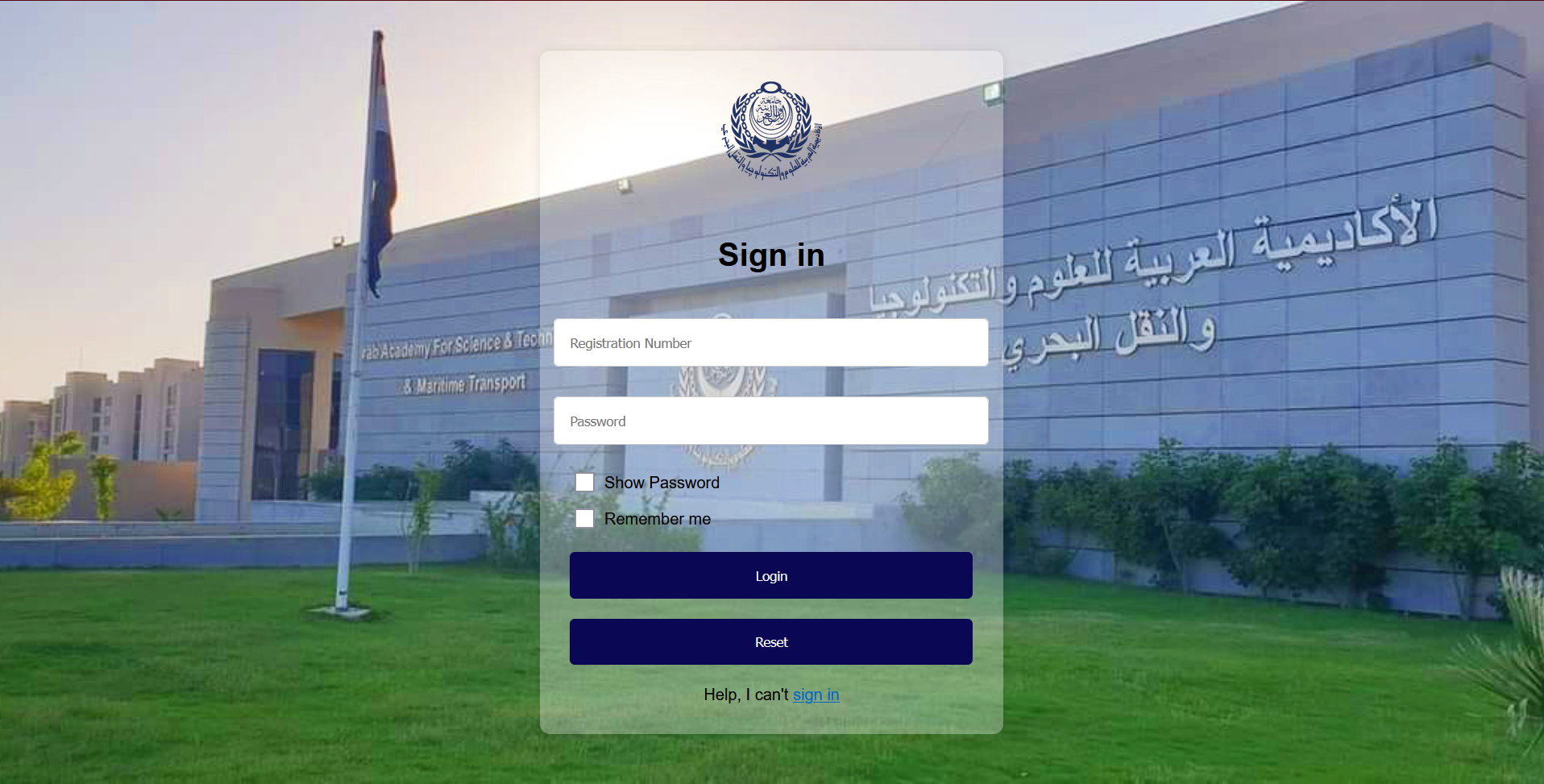
This interface allows users to choose their college and term to view their schedule. The clean and simple design ensures that users can quickly and easily select the appropriate options without confusion. The dropdown menus are clearly labeled, and the reset button allows users to clear their selections if needed.

##### **Figure 2: Student View Interface**



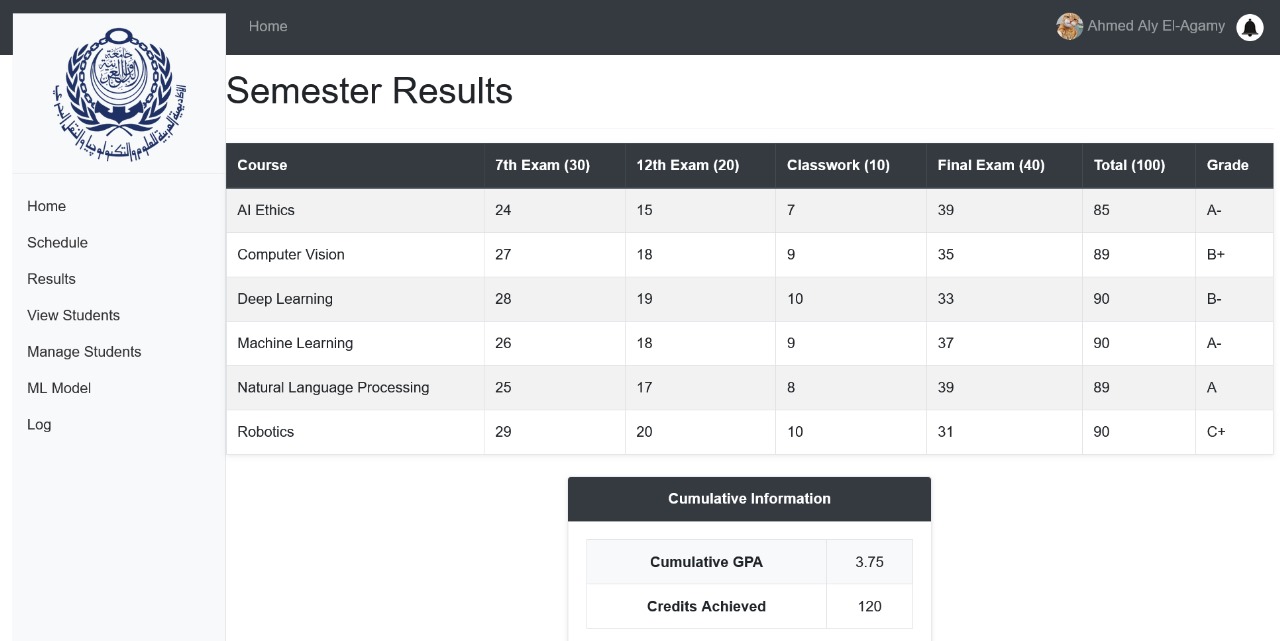
This interface is designed for administrators to view and manage information related to students' courses and schedules. Featuring dropdown menus for selecting the college and term, it ensures a user-friendly design that allows administrators to easily navigate and find the necessary information.

##### **Figure 3: Sign-In Interface**



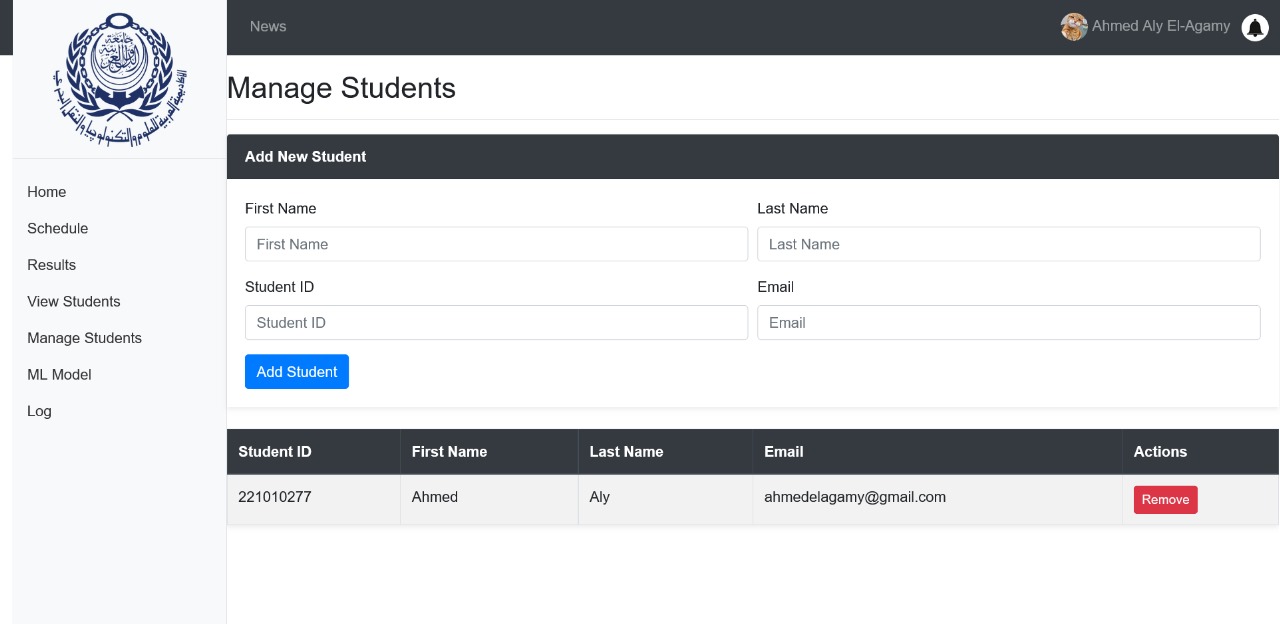
The sign-in interface is the entry point for users to access their accounts. It features input fields for the registration number and password, with options to show the password and remember the user. The design includes a clear and prominent login button, as well as a reset button to clear the input fields. A link is provided for users who need help signing in, ensuring that they have access to support if needed.

##### **Figure 4: College results Interface**



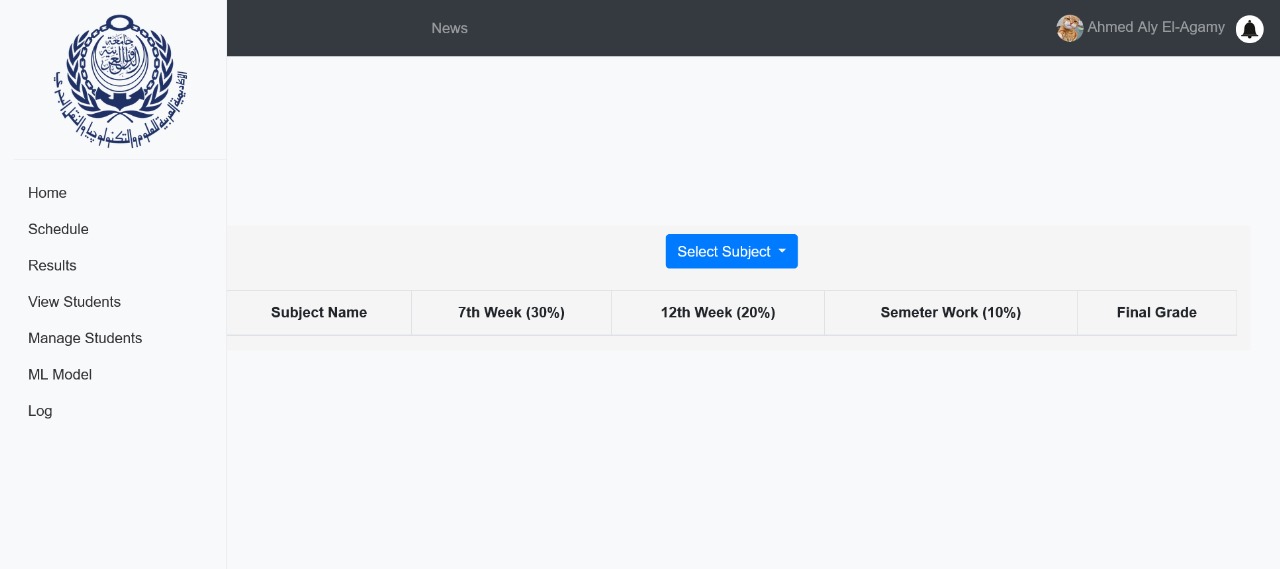
This document details the "Semester Results" interface for the Arab Academy for Science, Technology & Maritime Transport. The interface is designed to display students' course results, including detailed scores and cumulative information.

##### **Figure 5: College manage student Interface**



This document provides an overview of the "Manage Students" interface for the Arab Academy for Science, Technology & Maritime Transport. The interface is designed for administrators to add, view, and manage student records.

##### **Figure 6: College machine learing model Interface**



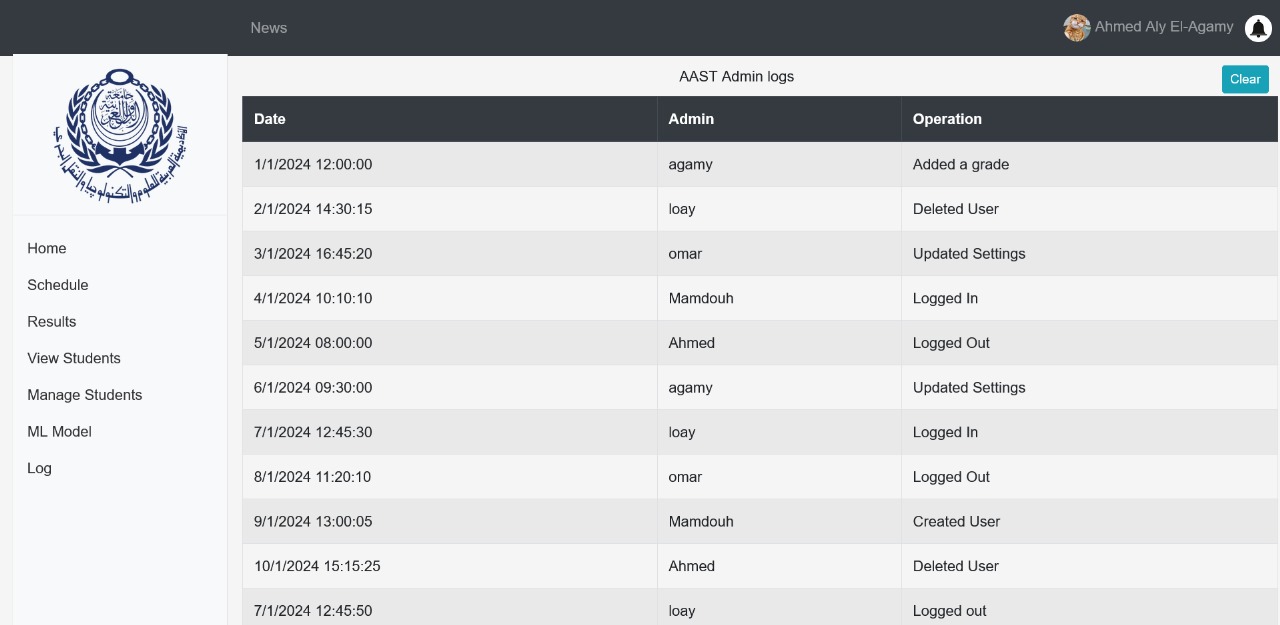
This document provides a comprehensive overview of the "Final Grade Prediction" interface for the Arab Academy for Science, Technology & Maritime Transport. The primary function of this interface is to predict students' final grades based on their performance throughout the semester. This tool is essential for both students and educators, offering a way to estimate final outcomes and make necessary adjustments to study plans and teaching methods.

The prediction model takes into account various assessments and coursework completed during the semester, including:

* 7th Week Exams (30% of the total grade)
* 12th Week Exams (20% of the total grade)
* Semester Work (10% of the total grade)

By integrating these components, the model provides a projected final grade, allowing students to understand their current standing and identify areas for improvement. This proactive approach aims to enhance academic performance and provide a clearer picture of academic progress

##### **Figure 7: College admain log Interface**

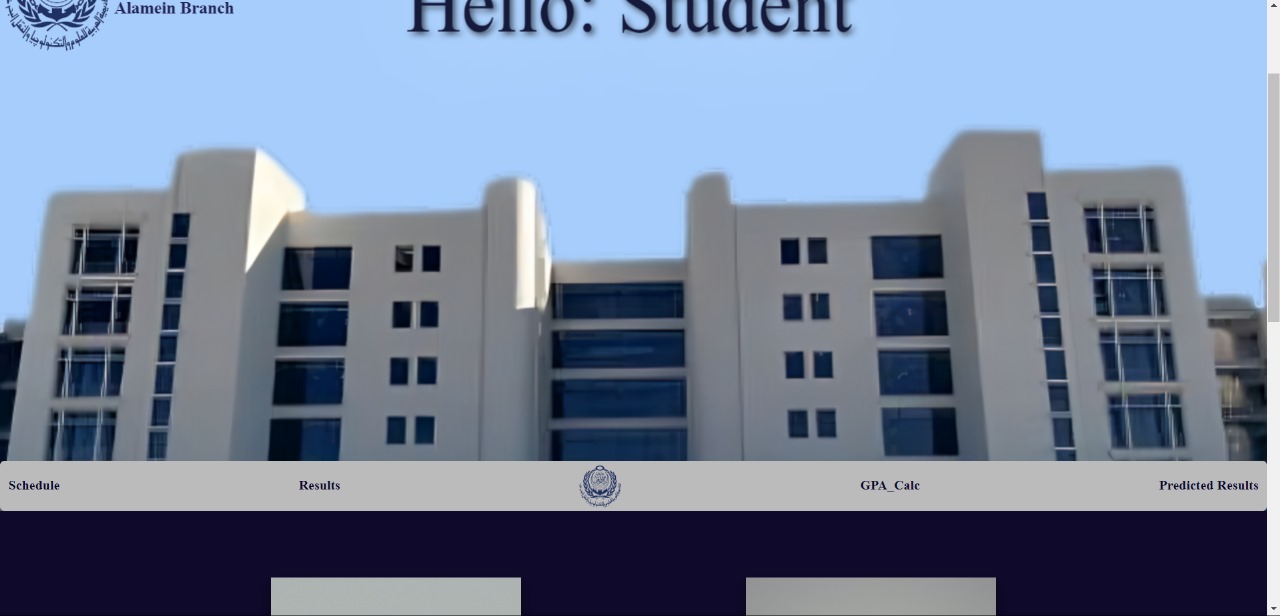


This document details the "Admin Logs" interface for the Arab Academy for Science, Technology & Maritime Transport. The interface is designed to provide a detailed log of administrative actions taken within the system. This tool is crucial for maintaining a record of changes, ensuring accountability, and tracking administrative activities.

The admin logs display a chronological list of actions performed by various administrators, including logging in and out, adding grades, deleting users, and updating settings. This comprehensive record helps in auditing and managing the administrative workflow.

##### **Figure 8: College home page Interface**





This document outlines the "Student Portal" interface for the Arab Academy for Science, Technology & Maritime Transport, Alamein Branch. The portal serves as a centralized hub for students to access various academic resources and information. The home screen greets the student with a welcome message and offers easy navigation to essential functions such as viewing schedules, results, GPA calculations, and predicted results.

The "Student Portal" aims to streamline students' academic experience by providing a user-friendly interface where they can manage their educational activities. The portal integrates multiple functionalities to support students throughout their academic journey, enhancing accessibility to critical academic data and tools.

#### **Functional Requirements**

The functional requirements of this project include several key aspects to ensure the website meets user expectations and technical standards. First, the website must feature a responsive design. This means it should be accessible and visually appealing on devices of all sizes, including desktops, tablets, and mobile phones. To achieve this, the website will use flexible grids, responsive images, and media queries to adjust the layout based on the device's screen size. Additional considerations include touch-friendly design elements and ensuring that all interactive components are accessible via both mouse and touch.

Second, a clear and intuitive navigation structure must be implemented. This includes a well-organized menu, breadcrumbs, and a search feature to help users find content quickly. The navigation should be consistent across all pages, with clear indications of the current page and easy access to related content.

Third, the website should effectively display various types of content, including text, images, and videos. This involves the use of appropriate HTML tags, CSS styles, and possibly JavaScript to enhance the presentation of content. Content should be organized into logical sections with headings and subheadings, and multimedia elements should be optimized for performance.

Lastly, forms should be implemented for user input, such as contact forms and registration forms. These forms should be user-friendly, with clear labels, input validation, and error messages to guide users through the submission process. Form submissions should be handled securely, with data validation on both the client and server sides, and users should receive confirmation of successful submissions

* **College Schedule Selection Interface**

1. **Dropdown for College Selection:**
   * Users should be able to choose their college from a dropdown list labeled "Choose Your College."
2. **Dropdown for Term Selection:**
   * Users should be able to select their term (1-8) from a dropdown list labeled "Choose Your Term (1-8)."
3. **Reset Button:**
   * A reset button should be available to clear the current selections.
4. **Display Schedule:**
   * Upon selecting the college and term, the corresponding schedule should be displayed.

* **Student View Interface**

1. **College Selection Dropdown:**
   * Users can choose their college from a dropdown menu labeled "Choose Your College."
2. **Term Selection Dropdown:**
   * Users can select their academic term (1-8) from a dropdown menu labeled "Choose Your Term (1-8)."
3. **Reset Function:**
   * A reset button is provided to clear the current selections and reset the interface.
4. **Student List Display:**
   * After selecting the college and term, the relevant student list will be displayed.

* **Sign-In Interface**

1. **Registration Number Input:**
   * Users must input their registration number in the designated field.
2. **Password Input:**
   * Users must input their password in the designated field.
   * An option to toggle password visibility is available.
3. **Remember Me Checkbox:**
   * Users can select this option to save their login information for future sessions.
4. **Login Button:**
   * Submits the registration number and password for authentication.
5. **Reset Button:**
   * Clears the input fields for registration number and password.
6. **Help Link:**
   * Provides assistance for users who are unable to sign in, labeled "Help, I can't sign in."

* **College results Interface**

1. **Course Results Table:**
   * Displays a list of courses taken during the semester.
   * Columns include:
     + Course Name
     + 7th Exam Score (out of 30)
     + 12th Exam Score (out of 20)
     + Classwork Score (out of 10)
     + Final Exam Score (out of 40)
     + Total Score (out of 100)
     + Grade
2. **Cumulative Information Section:**
   * Displays overall academic performance including:
     + Cumulative GPA
     + Credits Achieved
3. **Navigation Menu:**
   * Includes links to various sections such as Home, Schedule, Results, View Students, Manage Students, ML Model, and Log.
4. **User Information:**
   * Shows logged-in user's name and profile picture.

* **College manage student Interface**

1. **Add New Student:**

* Fields for inputting the student's first name, last name, student ID, and email.
* An "Add Student" button to save the new student's information.

1. **Student List:**

* A table displaying existing student records with columns for:
  + Student ID
  + First Name
  + Last Name
  + Email

1. An action button for removing a student from the list.

* **College machine learing model Interface**

1. **Subject Selection:**

* A dropdown menu for selecting the subject whose final grade prediction is needed.

1. **Assessment Breakdown:**

* Display the scores for:
  + 7th Week Exam (30% of the final grade)
  + 12th Week Exam (20% of the final grade)
  + Semester Work (10% of the final grade)
* Each component's contribution to the final grade is clearly indicated.

1. **Final Grade Calculation:**

* Calculate and display the predicted final grade based on the input scores.
* **College admain log Interface**

1. **Log Entries Display:**

* Shows a list of log entries with the following columns:
  + Date and Time
  + Administrator Name
  + Operation Performed

1. **Clear Logs Button:**

* A button to clear the displayed logs when necessary.
* **College admain log Interface**

1. **Home Screen:**
   * Displays a welcome message to the student.
   * Provides navigation links to key sections: Schedule, Results, GPA Calculation, and Predicted Results.
2. **Schedule Section:**
   * Allows students to view their current schedule.
3. **Results Section:**
   * Enables students to view their academic results for the semester.
4. **GPA Calculation:**
   * Provides tools for students to calculate their GPA based on their grades.
5. **Predicted Results:**
   * Offers a feature for students to predict their final grades based on current performance.

#### **Non-Functional Requirements**

The non-functional requirements of the project focus on performance, security, usability, and scalability. Firstly, the website should load quickly and perform efficiently, with minimal latency. This includes optimizing images, minifying CSS and JavaScript files, and leveraging browser caching. Performance testing should be conducted to identify and address bottlenecks, and best practices for web performance should be followed.

Secondly, scalability is important for future enhancements. The design and structure should allow for future enhancements and scalability. This includes writing clean, modular code that can be easily maintained and extended, and designing a flexible architecture that can handle increased traffic and additional features. Scalability considerations should include database design, server infrastructure, and the use of scalable technologies such as cloud services.

* **College Schedule Selection Interface**

1. **Usability:**
   * The interface should be easy to use and navigate, with clear labels and instructions.
2. **Performance:**
   * The application should load the schedule quickly upon selection of college and term.
3. **Compatibility:**
   * The application should be compatible with major web browsers.
4. **Accessibility:**
   * The interface should be accessible to users with disabilities, following web accessibility guidelines.
5. **Security:**
   * The application should ensure the security of user data and prevent unauthorized access.

* **Student View Interface**

1. **Ease of Use:**
   * The interface should be user-friendly with clear labels and guidance.
2. **Efficiency:**
   * The application should display the student list promptly upon selection of college and term.
3. **Browser Compatibility:**
   * The application should function smoothly across major web browsers.
4. **Accessibility:**
   * The interface should be accessible to users with disabilities, adhering to web accessibility standards.
5. **Data Security:**
   * The application should protect user data and prevent unauthorized access.

* **Sign-In Interface**

1. **User-Friendly:**
   * The interface is designed to be intuitive and easy to use.
2. **Secure:**
   * Ensures user credentials are protected during the login process.
3. **Efficient:**
   * The system quickly verifies user credentials and grants access.
4. **Cross-Browser Compatibility:**
   * Works seamlessly across all major web browsers.
5. **Accessible:**
   * Complies with accessibility standards to support users with disabilities.

* **College results Interface**

1. **Usability:**
   * The design should be user-friendly, making it simple for students to find and understand their results.
2. **Performance Efficiency:**
   * The system should provide quick access to results to avoid any delays.
3. **Browser Compatibility:**
   * Ensure the interface works well across different web browsers.
4. **Data Security:**
   * Protect students' information from unauthorized access and ensure data integrity.
5. **Accessibility Standards:**
   * The interface must be accessible to all users, including those with disabilities, by following accessibility guidelines.

* **College manage student Interface**

1. **User Interface:**
   * The design should be straightforward and user-friendly, enabling easy management of student records.
2. **System Responsiveness:**
   * The application should respond quickly to inputs and display updates in real-time.
3. **Cross-Platform Compatibility:**
   * Ensure the interface works seamlessly across different devices and web browsers.
4. **Data Integrity:**
   * Maintain the accuracy and consistency of student data throughout the application.
5. **Accessibility Compliance:**
   * Ensure the interface is accessible to all users, including those with disabilities, by adhering to accessibility standards.

* **College machine learing model Interface**

1. **User Experience:**
   * The interface should be intuitive and straightforward, ensuring ease of use for students and educators.
2. **Performance:**
   * The system should quickly process the input scores and generate the predicted final grade without noticeable delay.
3. **Compatibility:**
   * Ensure the interface is compatible with major web browsers and accessible on various devices, including desktops, tablets, and smartphones.
4. **Data Accuracy:**
   * The prediction model should utilize accurate and up-to-date data to provide reliable grade predictions.
5. **Accessibility:**
   * The interface must be accessible to all users, including those with disabilities, complying with relevant accessibility standards and guidelines.
6. **Security:**
   * Protect student data and ensure that the grade predictions are only accessible to authorized users.

* **College admain log Interface**

1. **User Interface:**
   * The interface should be intuitive and easy to navigate, ensuring that administrators can quickly find and review log entries.
2. **Performance:**
   * The system should efficiently display logs without delay, even when handling a large number of entries.
3. **Cross-Browser Compatibility:**
   * Ensure that the interface functions correctly across all major web browsers.
4. **Data Integrity:**
   * Logs should be accurate and tamper-proof to maintain the integrity of the administrative records.
5. **Accessibility Compliance:**
   * The interface should be accessible to all users, including those with disabilities, following accessibility standards.

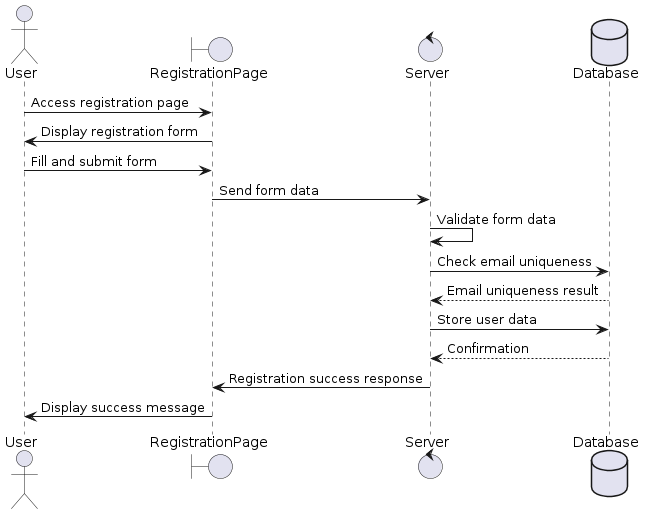
* **College admain log Interface**

1. **User Interface:**
   * The interface should be visually appealing and easy to use, enhancing the overall user experience.
2. **System Responsiveness:**
   * The portal should load pages quickly and respond to user inputs without delay.
3. **Compatibility:**
   * Ensure the portal works seamlessly across various web browsers and devices, including desktops, tablets, and smartphones.
4. **Data Security:**
   * Protect the confidentiality and integrity of student data, ensuring that only authorized users can access sensitive information.
5. **Accessibility:**
   * The interface should be accessible to users with disabilities, complying with relevant accessibility standards.
6. **Reliability:**
   * The portal should be reliable and available at all times, minimizing downtime and ensuring that students can access their information whenever needed.

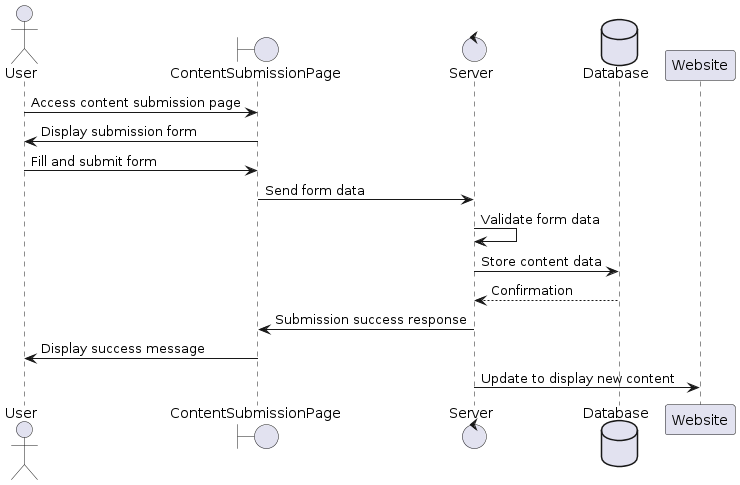
#### 

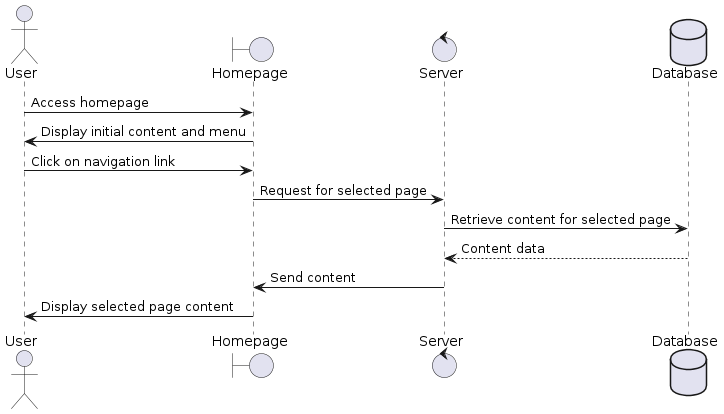
#### **Sequence Diagrams**

Sequence diagrams illustrate the interactions between different components and users of the website. They provide a visual representation of the flow of information and the sequence of events. For instance,



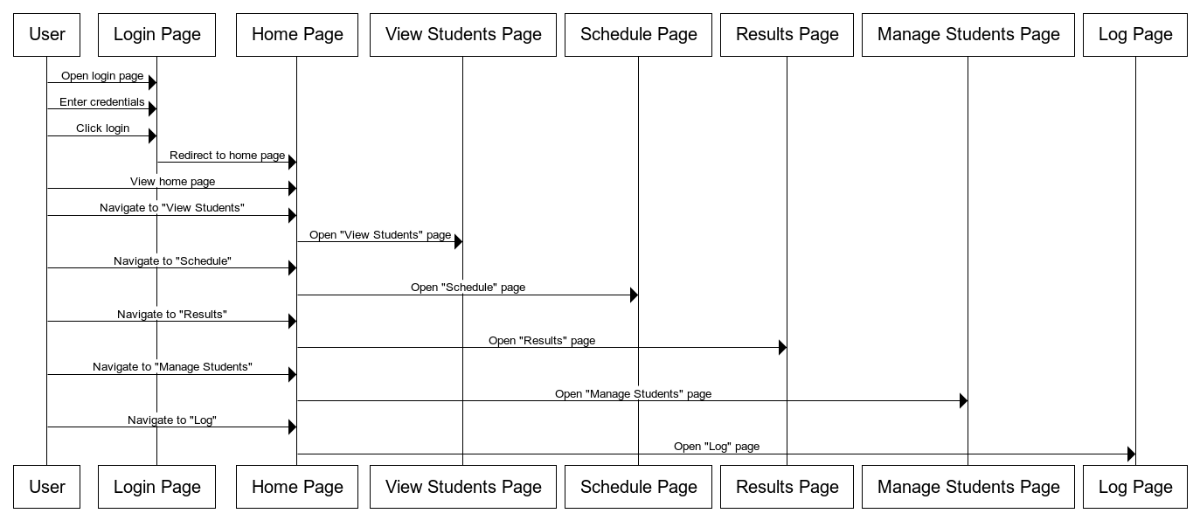
the user registration sequence diagram shows the steps from the user accessing the registration page to successfully creating an account. This includes form submission, server-side validation, database storage, and user feedback.

The content submission sequence diagram demonstrates how users submit content, including form validation, server processing, and feedback to the user. This covers the entire process from user input to content being displayed on the website.

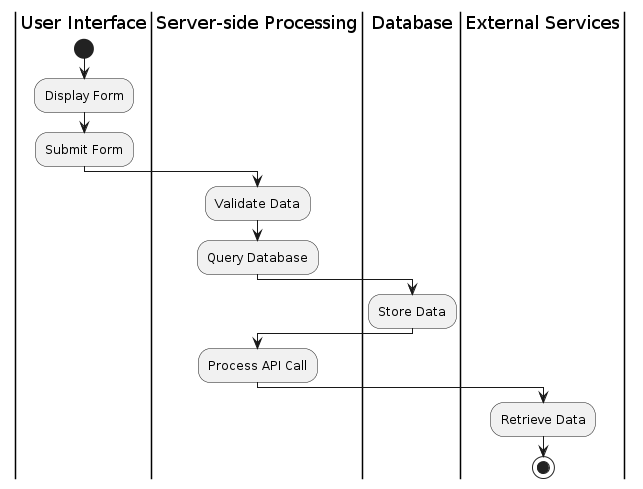


The navigation flow sequence diagram details the user's journey from the homepage to different sections of the website, highlighting key interactions and page transitions. This includes user actions such as clicking on links, using the search feature, and navigating through menus.

#### **Diagrams for all project**



#### **Flowchart**



The user begins by interacting with the user interface, where they fill out and submit a form. The server validates the submitted data and queries the database for necessary information. The validated data is then stored in the database. The server processes an API call to interact with external services, which retrieve the requested data and return it to the server for further processing or display to the user

***Conclusion***The design of the user interface is focused on simplicity and ease of use. The interface features a clean layout with clearly labeled buttons and input fields, ensuring that users can navigate the site without confusion. The use of color and spacing helps to highlight important elements and guide the user's attention to key actions.

Performance optimization is a critical aspect of the project. Techniques such as image compression, lazy loading, and the use of content delivery networks (CDNs) are employed to reduce load times and improve the overall responsiveness of the website. These measures ensure that users have a smooth and fast experience, regardless of their device or internet connection.

Security is integrated into every aspect of the development process. The use of secure coding practices, regular security audits, and the implementation of advanced security features such as multi-factor authentication (MFA) help to protect user data and prevent unauthorized access. Compliance with privacy regulations such as GDPR is also a priority, ensuring that user information is handled with the utmost care and transparency.

Scalability considerations include the use of modular design principles, allowing the website to grow and evolve over time. This modular approach makes it easier to add new features and functionalities without disrupting the existing structure. Additionally, the use of cloud services provides the flexibility to scale server resources up or down based on traffic demands, ensuring optimal performance at all times.

Usability testing is conducted throughout the development process to gather feedback from real users. This feedback is used to make iterative improvements, ensuring that the final product meets the needs and expectations of its audience. Accessibility features such as keyboard navigation, screen reader compatibility, and high-contrast modes are also included to ensure that the website is usable by individuals with disabilities.

The project's architecture is designed to support robust data management and processing capabilities. The use of a relational database ensures that data is stored efficiently and can be retrieved quickly. Server-side processing is optimized to handle complex queries and data transformations, providing a seamless experience for users. The integration with external services, such as payment gateways and third-party APIs, is handled securely and efficiently.

Regular maintenance and updates are planned to ensure the website remains secure and up-to-date. This includes applying security patches, updating dependencies, and adding new features based on user feedback. A comprehensive backup strategy is also in place to protect against data loss and ensure business continuity in the event of an outage or other disruption.