WWW.IJRPR.COM

International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

rEMS - revised Education Management System

Aayush Umesh Tiwari

dakshumeshtiwari@gmail.com

ABSTRACT :

The project aims to result as a final outcome for all education management system. It is a creative implementation of education/enterprise management system. This platform has been tailored to simplify and enhance the management of administrative, academic, and student-faculty related operations in college organizations. This project integrates several essential features like Admit Card Generation, Announcements, Attendance, Course Information, Hostel Details, Grievances, Fees Management, Event Managements, and more such things. All these are targeted to provide a comprehensive solution to take the needs of modern educational institutions to next level. This project utilizes the basic concepts of web development for front end such as HTML, CSS, Bootstrap and intermediate back end development such as Python, Django and database programming skills such as SQL to deliver a large scale, efficient, and user friendly solution. By putting all these features into a single project to build up an efficient operational system with real time data access to foster better communication among the organization. This project is made of basic implementation skills as the project is aimed for college organizations, so viewing it from as a programmer, this project represents that basics are used to build up a basic need for basic management implementation.

Ultimately, this project serves as a stepping stone towards modernizing educational management practices, leveraging technology to meet the dynamic demands of contemporary educational institutions. It underscores the potential of basic development skills in creating meaningful solutions that align with the needs of the digital age.

INTRODUCTION :

In this rapidly evolving digital era, college organizations require efficient management systems to manage administration, academic and student-faculty operations. rEMS aims to provide a platform that simplifies the processes while enhancing productivity and communication among the insides of organization. By combining basic web development technologies and database programming, this project should be able to deliver an efficient and user friendly solution tailored to the needs of education management.

BACKGROUND :

Managing the various tasks for an educational organizations were time consuming and were prone to be errors when they were handled manually. There has been implementation of education management solutions in various organizations to streamline the operations into a unified system. The day by day growing demand for digital solutions in education and desire to improve operational efficiency has shown many improvements and announced new features.

PROBLEM STATEMENT :

Educational Management often challenges in managing various administrative and academic operations. Some limitations leads to inefficiencies and delays which puts a negative impact on overall productivity. The users also faces issues while using these solutions and some implementations uses new complex ways. This project aims to resolve these issues by developing a solution which would overcome all these to provide a user satisfaction.

OBJECTIVES :

Develop an integrated platform for management of administrative, academic, and student-faculty related operations in educational organizations.

Implementing features that are most essential such as admit card generation, attendance tracking, grievance record, fee management, and event organization.

Leverage basic web technologies like HTML, CSS, Bootstrap and back end technology such as Python, Django and database language such as SQL to ensure scalability, robustness, and user- friendliness.

Provide real time data access to improve communication and decision making among the inside of the organization.

Symbolize that basic development and programming skills can effectively take a huge part in setting up the core management needs for educational organizations.

RELEVANCE :

The relevance of this project comes to address the difficulty in many educational organizations. There are selected few solutions regarding this solution and the ones who are existing in the market. To build this project the implementation is done solely on personal experience as a student on just a single platform as other requires a demo for organization but the demo access is limited to organizational use. By integrating key operational features into a single platform this solution revised EMS simplifies complex processes which results in highly practical and impacting solution. This solution highlights the potential of basic development skills to create meaningful management system, emphasizing the importance of practical implementations in addressing daily challenges face in the educational organization.

LITERATURE REVIEW :

For any educational institution to work effectively it requires a lot of properly processed and managed information. Data and Information comes in various forms in a continuous manner. Educational Institutions uses various tools to process on these. Nowadays educational organizations are solely based on this data to effectively run an institution. In [1] phrases like SIS - Student Information Systems and AIS - Academic Information Systems have been used to study about the information systems that are used to manage students related data. Few more terms such as EMIS - Education Management Information System, MIS - Management Information System, ISHE - Information System for Higher Education have been used too. These terms are used to integrate information from different educational organizations to properly analyze them on a common platform. Talking about success of a Education Management System does not depends on one single reason itself. Various factors such as effectiveness of design, operations, user friendliness are some of the factors that make an EMS a successful product. As different organizations have different ways of operating, so it is a challenging task to create a commercial scale software that would satisfy the client requirements on a common scale.

There are lot of features personalized for sections assigned to them such as Personal Information, Attendance, Fees and Finances, Results, etc. To manually operate on these it would be ton of work load and time consuming. EMS - Education Management System is a designated solution which integrates the services required for smooth functioning of facilities while keeping the performance in mind. Ease of services such as user management systems, admission process, attendance systems and much more. EMS provides a bird's eye view on the entire functioning of the education institution. Management System manages the different processes insides the educational organization like administration, staff management, academics, student management, accounts, etc.

The [3] analyzes and compares the commonly used client/server architecture and browser/server architecture. From the software engineering point of view, cross parallel method is proposed based on client/server and browser/server design idea of genetic algorithms for course scheduling. With the seamless and continuous expansion of students in educational institutions the amount of data present in these organizations is increasing on a large scale. And that data is directly proportional to workload of staff that results in lower work efficiency. Using EMS as a solution to further assist education management is to ensure the efficiency. To ensure the effectiveness of teaching management, a good educational management platform should be built. The education management system used in colleges is a platform that can efficiently manage the services in combination with modern technology. To meet the demands of the management of educational institutions, it is necessary to have a complete and proper education management system that is suitable for the specific institution.

METHODOLOGY :

The systematic approach adopted in designing and developing the rEMS, the tools, frameworks, and technologies implemented, - a comprehensive overview of the project architecture and a detailed description of the project is mentioned ahead.

TOOLS, FRAMEWORKS & TECHNOLOGY USED :

1. **Front end Development**

HTML (Hyper Text Markup Language) is the structure language used to build web pages designed on World Wide Web. It defines elements such as headings, paragraphs, links, images, and other multimedia. HTML elements are the building blocks of HTML pages, and they are represented by tags, enclosed in angle brackets. HTML5 introduced semantic elements like <article>,
<section>, <header>, and <footer>, which enhance accessibility and SEO by clearly defining the purpose of different parts of a web page. HTML is supported by all modern browsers, ensuring consistent rendering of web pages across different platforms. HTML allows the integration of Cascading Style Sheets (CSS) for styling and JavaScript for dynamic behaviors, creating interactive and visually appealing web applications. HTML supports the creation of forms, enabling user input and data collection, which is essential for interactive web applications. HTML makes it easy to embed multimedia content such as images, videos, and audio files, enriching the user experience.

CSS (Cascading Style Sheets) is a stylesheet language used to control the presentation of web pages written in HTML. It allows developers to apply styles such as colors, fonts, and layouts, making the web content more visually appealing and user-friendly. CSS enables the separation of content (HTML) from its presentation, allowing for cleaner code and easier maintenance. CSS uses selectors to target HTML elements and applies styles through properties like color, font-size, margin, and padding. CSS media queries facilitate responsive design by allowing different styles for different devices and screen sizes, ensuring a consistent user experience across various platforms. CSS provides powerful layout models like Flexbox and CSS Grid, which simplify the creation of complex, flexible, and responsive layouts. CSS supports animations and transitions, enabling smooth and engaging user interactions without the need for JavaScript.

Bootstrap is an open-source framework for development of responsive and mobile websites. Developed by Twitter, Bootstrap provides a collection of HTML, CSS, and JavaScript components that simplify the process of creating modern, responsive web applications. Bootstrap's grid system uses a flexible layout grid that adjusts to different screen sizes and resolutions, ensuring consistent design across devices. Bootstrap offers a wide array of reusable components, such as navigation bars, buttons, forms, modals, and carousels, which help in rapid UI development. Developers can customize Bootstrap's styles and components by modifying variables or using custom CSS, allowing for tailored designs that meet specific project needs. Bootstrap ensures consistent rendering of web pages across all major browsers, improving user experience. Bootstrap includes JavaScript plugins (using jQuery) that provide additional interactivity.

2. Back end Development

Python is a high level, general purpose programming language famous for its readability, simplicity, and versatility. It supports multiple programming paradigms, including procedural, object- oriented, and functional programming, making it a popular choice for developers in various domains. Python's syntax is clean and easy to learn, emphasizing readability, which reduces the cost of program maintenance. Python does not require explicit declaration of variable types, as the interpreter infers the type at runtime. Python comes with a vast standard library that provides modules and functions for a wide range of tasks, from file I/O to web development. Python is able to run on various OS, including Windows, macOS, and Linux that allows cross-platforms. Python has a large and active community, contributing to an extensive ecosystem of third-party libraries and frameworks, such as NumPy for numerical computing, Django for web development, and TensorFlow for machine learning. Frameworks like Django and Flask make it easy to develop web applications. Libraries such as Pandas, NumPy, and scikit-learn facilitate data analysis, machine learning, and scientific computing. Python is widely used for automating repetitive tasks, such as file handling, data scraping, and testing.

Django is a high-level, open-source web framework for building robust and scalable web applications. Written in Python, Django encourages rapid development and clean, pragmatic design. Django follows the Model- Template-View (MTV) architecture,
which separates business logic (Model), user interface (Template), and the controller logic (View). Django provides an auto-generated administrative interface to manage application data, significantly reducing the time to build admin dashboards. It includes a robust authentication system with support for user management, session handling, and permissions. Django's ORM allows developers to interact with databases using Python objects, abstracting SQL queries and database schema management. Django is designed for high-traffic websites, with built-in features to handle common security issues such as SQL injection. Django simplifies form processing with its form classes, supporting validation, CSRF protection, and error handling. The Django template engine allows dynamic HTML generation using template tags and filters. Django supports middleware, which is a lightweight, low-level plugin system for globally altering Django's input or output. Django provides built-in support for translating text and formatting dates, times, and numbers to different locales.

3. Database Management

SQL (Structured Query Language) is a standardized language used for managing and manipulating relational databases. It is essential for performing tasks such as querying data, updating records, and managing database schema. SQL allows users to retrieve data from databases using SELECT queries, which can filter, sort, and aggregate data. SQL provides commands like INSERT, UPDATE, and DELETE to manage and modify database records. SQL includes CREATE, ALTER, and DROP commands to define and manage the structure of database objects such as tables, indexes, and views. SQL offers commands like GRANT and REVOKE to control access to the database and ensure security. SQL supports transactions through commands like BEGIN, COMMIT, and ROLLBACK, ensuring data integrity and consistency. SQL supports various types of joins (INNER, LEFT, RIGHT, FULL) to combine data from multiple tables based on related columns. SQL allows embedding queries within queries to perform complex data retrieval tasks. Indexes improve the speed of data retrieval operations on a database table by providing quick access to rows. SQL enables the creation of reusable stored procedures and functions to encapsulate complex business logic. SQL allows the creation of virtual tables (views) that represent the result of a stored query.

PROJECT ARCHITECTURE & DESIGN :

This EMS project can be classified into three categories from an architecture's point of view. The architecture overview and its components-

1. Presentation Layer - (Front end) Handles user interactions and displays dynamic content. Technologies: HTML, CSS, Bootstrap. The presentation layer includes various UI elements like buttons, forms, menus, and dialog that enable users to interact with the application. It formats and displays data retrieved from the business logic layer in a way that users can understand, such as tables, charts, and reports. The presentation layer captures and validates user input before passing it to the business logic layer for processing. By separating the UI from the business logic, the presentation layer allows for independent development, testing, and maintenance, improving code organization and

scalability. Modern presentation layers focus on creating responsive designs that adapt to different screen sizes and ensure accessibility for users with disabilities.

2. Business Logic Layer - (Back end) Processes user inputs and enforces business rules. Technologies: Python, Django. The BLL receives input data from the presentation layer, such as user-submitted form data. It applies business rules, such as checking for required fields, calculating totals, or applying discounts. The BLL queries the data access layer to retrieve additional data or update records in the database. After processing, the BLL sends the results back to the presentation layer, such as a confirmation message or a list of search results.

3. Data Layer - (Database) Stores and retrieves data using SQL for structured storage and manipulation. The data layer abstracts the underlying database schema and query language, allowing the business logic layer to interact with data without concerning itself with the specifics of the database technology. It provides Create, Read, Update, and Delete (CRUD) operations to manage data in the database. The data layer ensures data integrity by enforcing database constraints and handling transaction management, such as commits and rollbacks. It manages database connections, ensuring efficient use of resources and maintaining performance. To improve performance, the data layer may implement caching mechanisms to store frequently accessed data temporarily. The data layer receives requests from the business logic layer to fetch or manipulate data. It translates these requests into SQL queries or appropriate commands for the database. The data layer fetches the results from the database and returns them to the business logic layer, possibly transforming the raw data into domain objects.

STEP BY STEP APPROACHES :

Requirement Analysis

Identifying the features required for an education management system by educational organizations.

Front end Development

Building responsive web pages using HTML, CSS and Bootstrap.

Implementing creative as well as dynamic for an enhanced user experience.

Back end Development and Integration

Setting up Django as the web framework.

Setting up the back end to connect with the database for data exchange.

Integrating front end components with back end.

Database Implementation

Design and create tables using SQL/MySQL.

Testing and Debugging

Performing unit and integration testing to ensure efficient interaction across different features.

Deployment and Maintenance Deploy the application on server.

Collect feedback for improvements and possible enhancements.

DIAGRAMS :

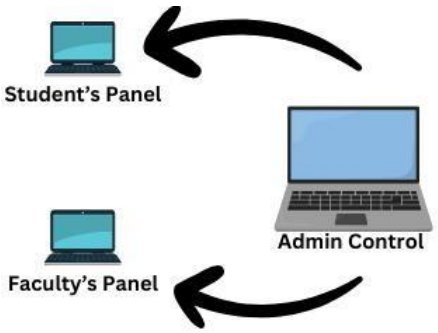

Fig.1. Panels of rEMS

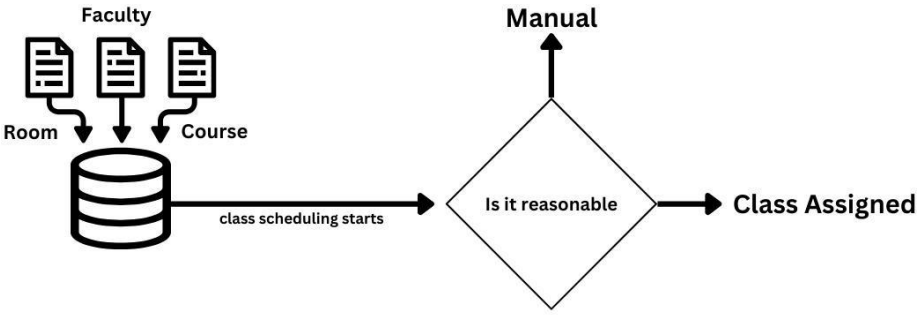

Fig.2. Class Schedulement

RESULT :

So far the results that have been achieved are as follows:
Constructed the Login page for the application.
Worked on the front end of the student's panel.
Assigned and planned the features that would be included in the application. Completed the back end for the Grievance feature of the project and connected the component for it's both
front as well as back end.
Planned for the database for the application.

CHALLENGES FACED :

Working upon this project was quite challenging. The resources required to build this project were quite limited or close to null. The research paper for this field are also available in limit. The existing solutions are also problematic to explore. This project is build upon by observing the one only solution i.e. icloud ems. Furthermore, to explore the faculty side panel is also proving to become quite challenging.

CONCLUSION & FUTURE WORK CONCLUSION :

The results so far have been quite satisfying. In addition this project covers the every aspect of learning which could be implemented on each feature from data handling to data manipulation. This project can also be used from a business point of view.
As this kind of project is not looked by many so to build this project and to take it to huge success would itself be quite an accomplishment.

FUTURE WORK :

Completing all the planned features.
After completion, to enhance the user experience.
Improve this web application to be available as an application for android as well as ios.

REFERENCES :

1. K. S. Sastry Musti, Research on Management Information, https://orcid.org/0000-0003-4384-7933, Namibia University of Science and Technology, Namibia
2. Systems for Higher Education Institutions: Challenges and Opportunities
3. MD Maruf Billah, Research on Easy Education Management System, International Journal of Scientific & Engineering Research, Volume 12, Issue 11, November-2021 ISSN 2229-5518
4. Zidi Chen, Juan Gao Measurement: Sensors 33 (2024) 101201
5. T. Phetmalaikul, Needs assessment in education and educational administration, J. Educ. Adm. 14 (26)
6. (2021) 118–129.
7. L. Lu, J. Zhou, Research on mining of applied mathematics educational resources based on edge computing and data stream classification, Mobile Inf. Syst. 2021 (7) (2021) 1–8.
8. S. Anardani, S. Riyanto, D. Setiawan, Perancangan knowledge management system berbasis web pada tenaga kependidikan fakultas teknik universitas pgri madiun, Jurnal Teknologi Informasi dan Ilmu Komputer 8 (1) (2021) 77–84. W3C. (2025). HTML & CSS: The
9. Complete Reference. Retrieved from W3C. Mozilla Developer Network. (2025). HTML: Hypertext Markup Language. Retrieved from MDN Web Docs
10. W3C. (2025). CSS: Cascading Style Sheets. Retrieved from W3C.
11. Mozilla Developer Network. (2025). CSS: Cascading Style Sheets. Retrieved from MDN Web Docs
12. Bootstrap Documentation. (2025). Introduction to Bootstrap. Retrieved from Bootstrap Docs
13. Mozilla Developer Network. (2025). Bootstrap. Retrieved from MDN Web Docs
14. Python Software Foundation. (2025). Python Documentation. Retrieved from Python Docs
15. Lutz, M. (2023). Learning Python. O'Reilly Media.
16. Mozilla Developer Network. (2025). Python. Retrieved from MDN Web Docs
17. Django Software Foundation. (2025). Django Documentation. Retrieved from Django Docs
18. Holovaty, A., & Kaplan-Moss, J. (2023). The Definitive Guide to Django: Web Development Done Right. Apress.
19. Mozilla Developer Network. (2025). Django Web Framework. Retrieved from MDN Web Docs
20. ISO/IEC. (2023). ISO/IEC 9075:2023
21. Information technology — Database languages — SQL. International Organization for Standardization.
22. W3Schools. (2025). SQL Tutorial. Retrieved from W3Schools
23. Microsoft. (2025). SQL Server Documentation. Retrieved from Microsoft Docs