Educational Institute Management System

Information Technology Project (IT2080)

Project Proposal Report

Submitted on: 02/03/2023
Group: ITP_B02_T23

Submitted by:

1. IT21388248 Fernando B.K.M.
2. IT21559136 Wijethunge W.L.D.N.N.
3. IT21704970 Lokuge P.P.
4. IT21786464 Nisal V.P.T.
5. IT21271328 Withana J.W.J.
6. IT21273094 Ediriwickrama E.A.K.V.
7. IT21360978 Serasinghe G. P. G. Y.
8. IT 21228926 Perera G.R.S.

Table of Contents

Introduction

Client Background

Our client is a leading educational institute in Hanwella, Colombo. This institution started in 2005 with one classroom and a few students. Now there are over 500 students and 15 teachers with 5 staff members. They have Sinhala & English medium classes for Ordinary Level & Advanced Level classes. In this Institution, specific teachers teach specific subjects. As an example, Mr. Nuwan teaches Mathematics to grades 10 & 11. And in this institution, students can select subjects they prefer to learn. In this institute, they maintain all the documents & classes in physical mode. In this institute, students should pay their class fees monthly, and calculate and pay the salaries of the teachers at the end of the month. Salaries are calculated according to the payments done by students. In this institution, they mark students' attendance regularly. They conduct exams, and extra classes.

Problems

After analyzing the existing travel management systems and the requirements of our client we were able to identify the existing errors in the current functioning travel management systems and also totally new complications which are mentioned below.

- Data Management – Problems may arise in the storage and management of information such as users, study materials, and subjects.
- Security and Privacy - Take care of the security of data such as financial data, user data, and corporate data. Therefore, this management system must ensure that the security of the data is secure.
- Communication – Effective communication between various stakeholders is very important. However, communication can be a challenge without a reliable method.
- Scalability – As educational institutions grow; this management system must also be scalable.
- Technical challenges: The possibility of technical difficulties in the implementation and improvement of technologies in educational institutional management systems.

Motivations

All the above problems can be addressed and successfully resolve by using the proposed web application which will meet the requirements of the client. The client will receive below mentioned benefits from the IT solution.

- Better Communication – The system helps improve communication between different parties, thereby creating an integrated learning environment.

- Enhanced Security and Privacy – Improving the security of data improves system performance and reliability.
- Increased efficiency – Time and effort are minimized as teachers or staff members can perform critical tasks automatically through this system.
- Improved Student Experience – Students can increase academic performance by providing access to study materials and subjects through this system.

Aims and Objectives

The aim of our Institute Management System is to provide an efficient and user-friendly platform for all the students, staff members and parents in the institute. This system covers all aspects of the institute like student registration, financial management, attendance management, assignment management, exam management, study materials management, and support services. We think this system will streamline the administrative processes of an educational institute, ensuring optimal utilization of resources and enhancing the overall performance of the institute.

- identify and analyze the existing administrative processes of the institute and determine the areas that need improvement.
- design and develop a user-friendly institute management system that integrates all administrative processes such as student registration, attendance management, examination management, and finance management.
- Improve accuracy to minimize errors.
- Improve the student's time management through smart application features.
- ensure that the institute management system is easily accessible to authorized personnel, such as teachers, staff, and students.
- To ensure that the institute management system is secure and reliable, with adequate measures to protect confidential information.
- test and evaluate the effectiveness of the institute management system, making necessary adjustments and modifications as required.
- assist administration and academic staff to monitor student achievements.
- provide guidance to teachers, students, and staff on how to use the institute management system effectively.
- ensure that the institute management system is scalable, flexible, and adaptable to future technological advancements.
- The system should support online learning and provide a learning platform.

-

System Overview

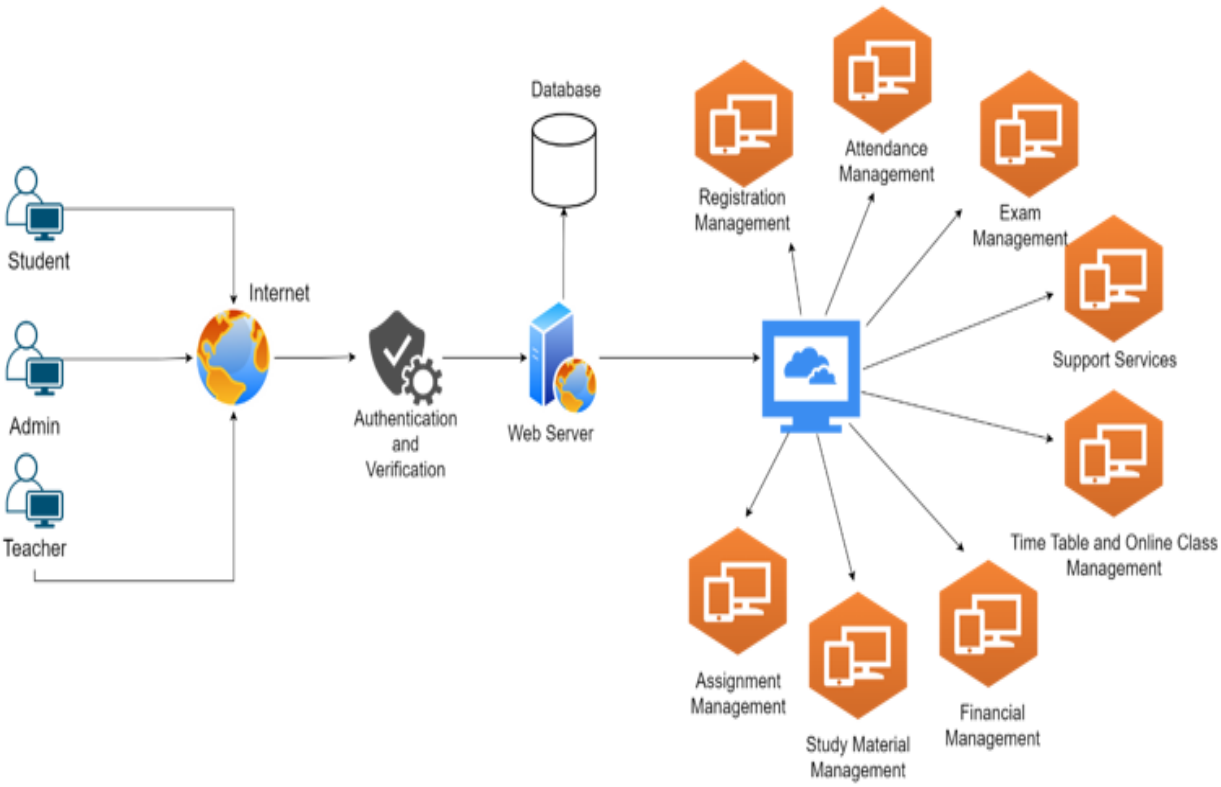

Figure 1: System Overview

Functional & Non-functional Requirements

1. Registration Management

The registration management function is responsible for managing the process of enrolling students and registering teachers into the Thilina Educational Institute system. Creating and maintaining student and teacher profiles, checking their information, and allowing access to the proper system features and resources are the parts of this role.

Functional Requirements

- **Allow students to register for the system.**
  The system should allow students to fill out the registration form and pay the registration fee and submit the payment receipt with it.

- **Allow admin to register teachers into the system.**
  By entering their personal information, teaching experience, and desired subject areas, teachers should be able to register with the system.
- **Verify student details and send a verification email.**
  The management of the institute should be able to verify the information provided by the students and should give them a confirmation email. The system login information should be included in the email
- **Allow students and teachers to access their profiles.**
  Users should be able to access and edit their own profiles on the system, including updating their personal information and, if necessary, removing their profile.
- **The system should generate unique QR codes for each student.**
  Each student should receive a special QR code from the system that can be used to track their development and confirm their attendance.
- **Allow students to request a free card.**
  If their any student facing financial problems they can request fora free card through the profile.
- **Allow students to search and select the subjects.**
  Then they can also view the timetable and search their grade and select the classes they prefer to learn in the institute
- **Allow students to download the reports.**
  Students should be able to download their ID card and payment summery report from their profile.

Non-Functional Requirements

- **Security**: Only registered students can log in to the system. The system should be secure and protect the personal information and payment details of students and teachers**.**
- **Scalability**: As the institute expands, the system should be scalable and capable of supporting an increasing number of users and courses.
- **Availability**: The system should function properly and always be accessible. Any time students can register, log in for the system.
- **Usability**: Both students and teachers should find it simple to use and navigate the system. It should feature an easy-to-use interface and detailed instructions for registering, choosing courses, and managing profiles**.**

2. Attendance Management

Attendance management systems are essential tools for educational institutions to track student attendance and manage their academic progress. They are equipped with a user-friendly interface and provide a variety of reports to analyses attendance patterns over time. Additionally, the system provides a report about each student's attendance details, allowing students to keep

track of their attendance status and take appropriate actions if they face any issues with their attendance. The system is an effective tool that helps educational institutions manage student attendance efficiently, saving time, minimizing errors, and providing accurate information about student attendance.

Functional Requirements

- **Marking attendance using QR or ID**
  QR (Quick Response) and ID (Identification) are two methods used to mark student attendance.
- **Generate student attendance reports by using search by student ID.**
  If a student request to see their attendance time to time, he or she able to generate a attendance report through our system, addition to this download report as PDF if they needed. So, students are able to see their performance on their attendance throughout the time period.
- **Using a user-friendly UI**
  The program is designed to be user-friendly and intuitive, making it easy for teachers and administrators to use.

Non-Functional Requirements

- **Security**: The system should have strong security measures to protect student data.
- **Scalability**: The system should be scalable to accommodate future growth and expansion of the institute and the number of students.
- **Compatibility**: The system should be compatible with different devices and platforms, including desktop and mobile devices and various web browsers.
- **Reliability**: The system must be available 24/7.

3. Financial Management

This function is responsible for all financial-related things in the institute. This function implements getting students' payments online and physically, searching & showing payments records, calculating salaries, generating financial reports.

Functional Requirements

- **Getting students' class payments**
  The system should allow students to select the grades and subjects they wish to Pay for. Also, the system should enable students to pay class fees either at the class premises or online. When

paying at the class premises the system should allow admins to enter the students' payments into the system

- **Sending payment verification**
  The system should send the payment confirmation to the student's email and generate invoices for students according to the payments.
- **Searching previous payment records**
  When paying at the class premises, the system should be able to search previous payment records by using a student ID. The system should allow students to see their payment records from their student profiles.
- **Calculating total fees for each student**
  Also, the system should calculate and display the monthly fees of each student based on the selected classes and subjects.
- **Storing all the data in the database**
  And the system should store all the payment data in the database.
- **Updating students' records**
  The system should be able to update or delete students' payment records if mistakes happen.
- **Teachers' Salary Calculation**
  The system should calculate the teachers' salaries when selecting the relevant grades & subjects teachers teach according to the payments made by students. The system should subtract the owner's commission when calculating salaries and the system should allow the admin to set the owner's commission.
- **Generate reports.**
  The system should generate salary slips for the teachers. At the end of the month, the system should also be able to generate all kinds of income reports for the owner.

Non-Functional Requirements

- **Security:** The system must be secure to prevent unauthorized access and ensure the privacy of students' and teachers' data.
- **Usability:** The system should be user-friendly for students, teachers, admins, and the institute owner.
- **Performance:** The system must be able to handle a large number of transactions and calculations.
- **Reliability:** The system must be available 24/7 with minimal downtime.
- **Scalability:** The system must be able to accommodate future growth in the number of students, teachers, and classes.

4. Timetable Management

Having a clear class schedule for an educational institution can indeed be very useful for teachers, students, and the management of the institute. It helps everyone to ensure that classes are running on time. It provides a convenient way for students and teachers to access their class schedules.

Functional Requirements

- **Displaying the timetable**
  Users should be able to view three types of timetables in the system. They are Whole class timetable, Personal timetable, and the Exam timetable. Including unregistered users, everyone can view the whole class timetable and it shows the time schedule of all the classes. Personal timetable shows their specific time schedules of the classes, for students that they have enrolled and for teachers that they are conducting. Exam timetable shows the specific time schedules of the upcoming exams. It should be visible only for students.
- **Enable real time updates.**
  Only the administrator should be able to modify the timetable in case of changes in class timings, class cancellations and class additions. The system allows admin to add, edit and delete classes in the system, specifying details such as the subject, grade, time, date, and teacher name.
- **Online Class Management**
  The system should allow teachers to upload meeting links for extra online classes, which will only be displayed to students who have paid their class fees for the relevant month.
- **Class reminders and Exam reminders**
  Students and teachers should be able to see the remaining time for their next class in the class reminder. The remaining days and hours for the upcoming exam would be visible in the Exam reminder. Those reminders are specific to each other.
- **Notification Management**
  The system should be able to send notifications to students regarding changes to the class schedule, such as cancellations or postponements.
  When an already registered student registers for another class the system checks whether there is a time clashing. If the classes are clashing students should be notified at the same time.
- **Search Functionality**
  The system should allow users to search for specific class schedules in the timetable by entering the subject and grade.
- **Enable to download timetables.**
  Students and teachers should be able to download their personal timetables.
- **Integration with Google Calendar**
  The system should be able to integrate with the Google Calendar app, allowing users to sync their personal timetables across devices and platforms.

Non-Functional Requirements

- **Usability**: The system must be user-friendly for students, teachers to view the timetable and for admin to make modifications.
- **Performance:** The system must be able to handle a large number of class details.
- **Reliability:** The system must be available 24/7 with minimal downtime and with quick response.
- **Scalability**: The system should be built on a scalable infrastructure that can be easily expanded as the institution grows.
- **Compatibility:** The system should be compatible with a wide range of devices and browsers

5. Exam Management

exam management is very important for our institute. There are two separate UIs for teachers and students. Teachers can publish exams on the exam page and students can attend them. Students can check exam information and their marks after it is completed. exams are timed and after the time is up exam will automatically close and automatically submitted student answers. Generated reports are Automatically send student's parent.

Functional Requirements

- **Create Exam**
  Teachers should be able to create exams by adding questions and answers through a form-based UI. Teachers can set time limits and the number of questions for each exam. They should be able to select the type of question, such as multiple-choice or structured
- **Edit Exam**
  Teachers should be able to edit exams and update questions and answers, and also exam's time limit and the number of questions.
- **Publish Exam**
  After creating an exam, teachers should be able to publish the exam so that it is available for students to attempt.
- **View Reports**
  Teachers should be able to view detailed reports on each student's exam performance, including comparisons with past exams.
- **Student UI**
- **Search for Exam**
  Students should be able to search for exams that are available to attempt.
- **Attempt Exam**
  Students should be able to attempt exams by answering questions using the relevant UI. The order of questions should be different for each student.

- **Submit Exam**
  Students should be able to submit their exams after completing them, and their marks should be displayed along with the correct and incorrect answers.
- **Request Other Exam**
  If a student is unable to attend an exam, they should be able to request another exam from the teacher, and if approved, a new exam should be created.

Other Functional Requirements

- **Automated Reminders**: After an exam is published, reminders should be automatically sent to students via Gmail and their profile to attempt the exam.
- **Automated Mark Report**: After a student submits their exam, a mark report should be automatically generated and sent to their parent's email, and marks should be stored in the database.
- **Automated Progress Report**: After the term is finished, an automated progress report should be generated for each student and sent to their parent's email.

Non-Functional Requirements

- **Performance**: The system should be able to handle a large number of users and simultaneous exam attempts without any delays or interruptions.
- **Usability**: The system has a user-friendly interface for both teachers and students to create, manage, and attempt exams.
- **Reliability**: The system should be available 24/7 without any downtime
- **Security**: The system should have strong security measures to protect exam materials and student data.
- **Scalability**: The system should be scalable to accommodate future growth and expansion of the institute and the number of students and teachers.
- **Accessibility**: The system should be accessible to every user in the Institute.
- **Compatibility**: The system should be compatible with different devices and platforms, including desktop and mobile devices and various web browsers.

6. Study Material Management

Study material management is a system for organizing and managing educational resources such as notes and other materials. It helps teachers and students easily access and share resources and manage the distribution of materials.

Functional Requirements

- **Access to this service**
  To access this service, the system will ask for user ID. user should enter their valid id number to this system. Then System automatically validate user entered id number is verified or not. IF the identity is verified, they will be directed to the respective dashboard. Otherwise, cannot access this service.
- **Adding and Organize Study materials.**
  Only teachers can add and organize study materials. Teacher should select specific material category such as class recording, notes, research papers. When they selected category, they should have to fill out form with their name, teaching grade and subject as well. Teachers also can edit and delete what they uploaded materials in to the system. In additionally, teachers can view student feedbacks for study materials.
- **Availability of study materials**
  When accessing the student dashboard, the student dashboard shows what subjects they enrolled to our institute and study material categories. Students should search study materials using their voice or text input. Students get study materials what they need, they can add feedbacks for the relevant study materials and they have ability to edit and delete feedbacks. Students can download study materials using this service.
- **Report Generation**
  This service generates two reports.one is for students, if students download study material using this service, system automatically generate report what they downloaded material. And other report is, when teacher adding study materials to the system, then system automatically generate report what they added into our system.

Non-Functional Requirements

- **Security** - Access should be controlled based on permissions.
- **Availability:** The system should be available 24/7 to ensure access to study materials.
- **Usability**: This Service should be easy to use and User friendly
- **Reliability**: The service should be reducing the errors occur using this service and implementing

7. Assignment Management

Assignment management is one of the fundamental activities in education. In traditional assignment management schemes, assignments are recorded on paper. They must be delivered or organized manually. This is inconvenient and inefficient, and may cause many problems due to material limits and human errors, etc. Furthermore, instructors and students usually are unaware of the ongoing academic information, which can be used to help them find out problems and improve their teaching and learning. Therefore, it becomes important to implement an

assignment management method, which can provide both instructors and students with quality educational services. This system helps the user to complete work on time as user can check the status of assigned work any time. System is developed to manage the work assigning process online.

Functional Requirements

- Student should have a valid username and password.-Without having a valid username or password students can not log into the system.
- System should validate the user.
- It facilitates the teachers to view the status of the assigned tasks.
- System should display a countdown for each assignment informing the deadline.
- Upcoming assignments will be remaindered by the system.
- Teachers should generate evaluation reports.
- Student can search assignments using search bar.
- Teachers should create, read, update, and delete assignments.
- Student should create, read, update, and delete submissions.
- Teachers should generate feedback reports.

Non-Functional Requirements

- **Usability**: The system should be easy to use and navigate, with an intuitive interface that minimizes the need for user training.
- **Security**: The system should provide security measures to protect sensitive student data, including strong authentication, encryption, and access controls.
- **Performance:** The system should be able to handle many simultaneous users and assignments, with fast response times and minimal delays.
- **Scalability:** The system should be able to handle growth and increased usage over time, without requiring significant modifications or upgrades.
- **Maintainability**: The system should be easy to maintain and update, with clear documentation and support from the vendor or development team.
- **Compatibility:** The system should be compatible with a variety of hardware and software platforms, allowing users to access the system from different devices and locations.
- **Accessibility:** The system should be accessible to users with disabilities, including support for assistive technologies and compliance with accessibility guidelines.
- **Interoperability:** The system should be able to integrate with other systems and tools used in the educational environment, such as learning management systems and student information systems. assigned task and generate reports accordingly.
- **Reliability:** The system should be dependable and operate without errors or downtime, ensuring that assignments can be submitted and graded on time.

8. Support Services Management

The Support services module in Institute Management System is a critical component that provides a centralized platform for students, parents, and guests to report issues, request help, or request various services from the institution. Also, these support services are helpful to ensuring the success of students and the effective operation of the institution.

Functional Requirements

- **Raise a ticket.**
  Students can raise tickets through the Support service. To access this service, firstly students want register to the system. These tickets can be used to report issues, request assistance, or request various services from the institution. If a student needs to change their ticket, they can update it, and if they no longer need the ticket, they can delete it. The support services module is designed to ensure that all issues reported by students are addressed and resolved in a timely manner.
- **Reply for tickets and report generation.**
  The module is managed by the admin who has access to all the tickets raised by the students. The admin can access the ticket and provide a solution for the issue raised. If the Admin needs to update the Reply message, they can search the ticket id and do it. The admin can also generate reports about tickets and download them for further analysis. This helps the institution to identify recurring issues and address them in a proactive manner.
- **Ask questions on the FAQ page.**
  The support services module also provides an FAQ page, which is a valuable resource for anyone who has questions about the institution's policies, procedures, or services. The admin has the ability to reply to questions and delete the question.
- **Use chatbot.**
  In addition, the module provides a chatbot that can answer common questions and provide quick replies to students who need assistance. The chatbot is available 24/7 and can be accessed from anywhere, providing students with a convenient and efficient way to get the help they need.

- **Watch user guide videos.**
  Furthermore, the Support services module provides user guide videos, which are a valuable resource for students who want to learn more about the registration process and other facilities in the system. These videos provide a clear and concise overview of various features and functionalities of the institute Management system, helping students to navigate the system more effectively.

Non-Functional Requirements

- **Performance**: respond quickly to user requests and provide a smooth user experience.
- **Usability**: easy to use and navigate for both teacher and students, with clear instructions and user-friendly interfaces.
- **Availability**: The support service chatbot should be available 24/7, with minimal downtime for maintenance.
- **Security**: Only the authorized people should be able to access the system using usernames and passwords.

Technical Requirements

1. Development stack: Specify the development stack to be used, such as the MERN (MongoDB, Express.js, React.js, Node.js) stack.
2. REST API: Use a RESTful API to connect the Frontend and Backend.
3. Database: Use a cloud-based database service MongoDB to store data and enable scalability, availability, and data security.
4. Back-end development: Develop the back-end of the application using a framework like Express.js, which provides a wide range of features and flexibility for creating scalable web applications.
5. Front-end development: Develop the front-end of the application using React.js. React.js provides a wide range of features, tools, and community support for developing modern web applications.
6. Mobile application development: Develop the QR scanner mobile application using an Android studio. It helps to develop simple and fast Android application.
7. Version control: Use Git and GitHub for version control to enable efficient collaboration among team members, track code changes, and ensure code quality.
8. Runtime environment: Use Node.js and npm as the runtime environment for the application, enabling this we can use thousands of open-source packages to add functionality and features to our application.
9. Payment gateway: We use a payment gateway to enable online payments for our application. it will allow our application to process payments securely and efficiently, making it easy for users to pay for services or products.

Literature Review

Education institutes play a vital role in shaping the future of society, and in recent years, education technology has become a crucial part of their daily operations. The implementation of an education institute management system can revolutionize the way an educational institute operates by automating processes, streamlining operations, and improving communication between students, teachers, and administrative staff. This literature review aims to explore the various benefits and challenges of implementing an education institute management system. [1] [2]

Benefits of implementing an education institute management system

- Automation of processes:
  An education institute management system can automate many of the administrative tasks, such as attendance tracking, grading, and scheduling. This automation can reduce the workload on teachers and administrative staff and allow them to focus on more critical tasks.
- Improved communication:
  An education institute management system can improve communication between students, teachers, and administrative staff. Features such as messaging systems, forums, and chat rooms can facilitate communication and collaboration between different stakeholders.
- Centralized data management:
  An education institute management system can provide a centralized platform for managing student data, including attendance records, grades, and exam results. This centralized platform can reduce the chances of data errors and ensure that data is accurate and up-to-date.
- Enhanced student experience:
  An education institute management system can enhance the student experience by providing them with access to a range of online resources such as course materials, study guides, and online forums. This online access can enable students to learn at their own pace and on their own schedule.
- Increased operational efficiency:
  An education institute management system can increase operational efficiency by automating many administrative tasks, reducing the workload on teachers and administrative staff, and improving communication between different stakeholders. This increased efficiency can lead to cost savings and improved productivity.

Challenges of implementing an education institute management system

- Resistance to change:
  One of the most significant challenges of implementing an education institute management system is resistance to change. Teachers and administrative staff may be resistant to the new technology and may require additional training and support to adopt the system fully.

- Cost:
  Implementing an education institute management system can be expensive, and institutes may need to allocate a significant amount of funds to implement the system fully.
- Technical issues:
  Implementing an education institute management system can also present technical challenges, such as software compatibility issues, server capacity, and cybersecurity concerns.
- Data privacy and security:
  An education institute management system contains sensitive data, such as student records and financial information. Therefore, data privacy and security must be a top priority when implementing the system.

Data sources that we used.

1."Design and Implementation of an Institute Management System" by Oluwaseun Ogunsola and Emmanuel Adegbola (2018): The study aims to design and implement an Institute Management System for a Nigerian tertiary institution. The system includes features such as student registration, attendance management, timetable scheduling, and result management. The authors reported that the system was able to improve the efficiency of administrative tasks and reduce the workload of staff.

2."Institute Management System: A Case Study of A Technical College" by Emmanuel Ayobami Olagunju and Adeola Afolabi (2019): The study conducted a case study of a technical college in Nigeria to evaluate the effectiveness of an Institute Management System. The authors reported that the system was able to streamline administrative tasks, reduce errors, and improve data accuracy. The study also found that the system improved communication between staff and students.

3."A Comparative Study of Institute Management Systems in India" by Hemalatha Palanisamy and Dr. J. Jayasudha (2019): The study compared three different Institute Management Systems used in Indian educational institutions. The authors found that the systems were effective in managing administrative tasks, but they differed in terms of the features offered and ease of use. The study recommended that educational institutions should carefully evaluate their needs and choose a system that meets their requirements.

4."Evaluation of Institute Management System in Indian Higher Education: A Case Study" by Parul Gupta and Dr. Sanjay Singh (2020): The study conducted a case study of an Institute Management System implemented in an Indian higher education institution. The authors reported that the system was able to improve the efficiency of administrative tasks and reduce errors. The study also found that the system improved communication between staff and students and provided timely access to academic records.

5. Impact of Institute Management System on Educational Institutions: A Study of Nigerian Universities" by Gbadebo Olusegun Ogundele and Joseph Oluwafemi Akinyemi (2021): The study aimed to evaluate the impact of Institute Management Systems on educational institutions in Nigeria. The authors reported that the systems were able to improve administrative efficiency, reduce errors, and improve data accuracy. The study also found that the systems improved communication between staff and students and provided timely access to academic records.

Conclusion

In conclusion, implementing an education institute management system can have numerous benefits, such as automation of processes, improving the efficiency of administrative tasks, reducing errors, improving data accuracy, improved communication, centralized data management, enhanced student experience, and increased operational efficiency. However, there are also challenges associated with implementing the system, such as resistance to change, cost, technical issues, and data privacy and security concerns. Institutes should carefully evaluate the benefits and challenges of implementing an education institute management system before deciding to implement it fully. Additionally, institutes should ensure that they have the necessary resources, such as funding and technical expertise, to implement the system successfully.

Methodology

Benefits of implementing an education institute management system

- **Automation of processes**:
  An education institute management system can automate many of the administrative tasks, such as attendance tracking, grading, and scheduling. This automation can reduce the workload on teachers and administrative staff and allow them to focus on more critical tasks.
- **Improved communication**:
  An education institute management system can improve communication between students, teachers, and administrative staff. Features such as messaging systems, forums, and chat rooms can facilitate communication and collaboration between different stakeholders.
- **Centralized data management**:
  An education institute management system can provide a centralized platform for managing student data, including attendance records, grades, and exam results. This centralized platform can reduce the chances of data errors and ensure that data is accurate and up-to-date.
- **Enhanced student experience**:
  An education institute management system can enhance the student experience by providing them with access to a range of online resources such as course materials, study guides, and online forums. This online access can enable students to learn at their own pace and on their own schedule.

- **Increased operational efficiency**:
  An education institute management system can increase operational efficiency by automating many administrative tasks, reducing the workload on teachers and administrative staff, and improving communication between different stakeholders. This increased efficiency can lead to cost savings and improved productivity.

Challenges of implementing an education institute management system

- Resistance to change:
  One of the most significant challenges of implementing an education institute management system is resistance to change. Teachers and administrative staff may be resistant to the new technology and may require additional training and support to adopt the system fully.
- Cost:
  Implementing an education institute management system can be expensive, and institutes may need to allocate a significant amount of funds to implement the system fully.
- Technical issues:
  Implementing an education institute management system can also present technical challenges, such as software compatibility issues, server capacity, and cybersecurity concerns.
- Data privacy and security:
  An education institute management system contains sensitive data, such as student records and financial information. Therefore, data privacy and security must be a top priority when implementing the system.

Design Method

Wireframes are a useful tool in the design process that allows one to visually portray the layout and functionality of a website or application before it is developed. Once wireframes are created, they should be evaluated to ensure they are user-friendly, satisfy client requirements, and are operational and functional.

Sprint Plan

Sprint plan for developing our Institute Management System using the Agile Scrum framework:

Sprint 1 (2 weeks)

- User stories and requirements gathering
- Wireframing and prototyping of core features (e.g., student management, course management, attendance tracking)
- Set up development environment and version control.
- Implement basic authentication and authorization for different user roles.
- Develop database schema and start implementing database functionality.
- Create initial UI mockups.

Sprint 2 (2 weeks)

- Build out student management functionality (e.g., add, edit, delete, search)
- Build out course management functionality (e.g., add, edit, delete, search)
- Build out attendance tracking functionality.
- Integrate initial UI mockups with functionality.
- Start testing and bug fixing.

Sprint 3 (2 weeks)

- Build out user management functionality (e.g., add, edit, delete, roles)
- Build out enrollment functionality (e.g., add, edit, delete)
- Build out grade management functionality.
- Refine UI and UX based on feedback and testing.
- Continue testing and bug fixing.

Sprint 4 (2 weeks)

- Build out billing and payment functionality.
- Build out reporting functionality (e.g., attendance, grades, billing)
- Build out email notifications and alerts.
- Continue refining UI and UX based on feedback and testing.
- Continue testing and bug fixing.

Sprint 5 (2 weeks)

- Finalize UI and UX design.
- Conduct comprehensive testing and bug fixing.
- Optimize performance and scalability.
- Write documentation and user manuals.
- Prepare for deployment and production release.

Software Development

Agile development

Project Initiation:

Hold a gathering with the entire team to go over the project's aims, targets, and scope.

Establish user personas, identify stakeholders, and order user stories according to their business value.

Sprint planning: Before the start of each sprint, have a meeting to define the goals, estimate the number of user stories, and assign tasks to team members.

Rank the user tales in order of importance based on their business worth and interdependencies.

Sprint:

Carry out the sprint, which includes daily standup meetings, development, testing, and sprint review meetings.

Work closely with stakeholders to ensure that the product meets their expectations.

User Stories:

Create user stories that encapsulate the system's characteristics and capabilities.

Break the user experiences down into jobs and subtasks.

Testing:

Perform unit testing, integration testing, and acceptance testing throughout the development process.

When possible, use automated testing tools to accelerate testing.

Continuous Integration and Deployment: Automate the build and deployment processes by implementing continuous integration and deployment practices.

To automate testing, building, and deployment, use tools like Jenkins.

Retrospective:

At the end of each sprint, hold a retrospective meeting to reflect on progress, identify areas for improvement, and make changes to the development process.

Collaboration:

Work closely with stakeholders such as faculty, students, and employees to ensure that the system fulfills their requirements.

To facilitate communication, use Agile communication tools such as Jira, Trello, and Slack.

The team can deliver a high-quality web-based educational institute management system that meets the needs of its users,

is delivered on time and within budget and can be adapted to future changes by following this Agile development methodology.

Integration

Methods:

- API integration: Exposing APIs is a great way to integrate different software systems. RESTful APIs can be used for this project to allow for seamless communication and data exchange between different modules and components of the system.

- Integration tools: Tools like Apache Camel, MuleSoft, and Dell Boomi provide a variety of integration options and support different patterns for integrating systems.
- Service-oriented architecture (SOA): SOA enables networked data to flow between various software systems by defining services as self-contained modules that provide APIs. This method can produce a scalable, adaptable system that is simple to combine with other systems.
- ETL (Extract, Transform, Load) tools: These tools can extract data from various sources, transform it into a common format, and load it into the target system. This approach can integrate data from different systems into a common database or data warehouse.
- Alternatives: Message-oriented middleware (MOM): This approach can be used to integrate different systems, but it requires a more complex infrastructure and is less flexible than API integration or SOA.
- Custom integration using programming languages and frameworks: This approach can be used, but it requires more development effort and is not as scalable as using integration tools or SOA.
- Justifications: Since they provide a flexible and scalable method of integrating many systems and are extensively used in the industry, API integration and SOA are excellent approaches for integrating the MERN system. RESTful APIs can be used with a variety of programming languages and frameworks and are simple to create. The modular architecture offered by SOA makes it possible for various components to be created and implemented separately. Integration tools are chosen because they offer a variety of integration possibilities and support various integration patterns. ETL tools are preferred because they may be used for data warehousing and business intelligence and are efficient at combining data from several sources into a single format.

Development Tools

Many technologies such as LAMP stack, MEAN stack, Django, and ASP.NET can be used to develop web applications. Among them, We are planning to use the MERN stack to develop our web application.

MERN is a full-stack JavaScript stack, meaning that it uses the same programming language (JavaScript) for both front-end and back-end development. java script is the most developed, famous, and most demanded programming technology. To keep up with the modern-day sophisticated technology we are using the MERN stack as its one of the cutting-edge development methods.

The MERN stack, which consists of MongoDB, Express, React, and Node.js, is a popular choice for web development due to its flexibility, scalability, and ease of use.

MongoDB:-We use MongoDB to store and manage the application's data.

Express.js:- We use Express.js as a framework to manage the application's routing, middleware, and other back-end tasks. It is a back-end web application framework that runs on top of Node.js, providing a set of tools and features for building robust and scalable web applications.

React.js:-We use React.js to create dynamic and interactive user interfaces that can respond to user actions in real-time. React can simplify the process of building complex user interfaces by breaking them down into smaller, reusable components.

Node.js: Node.js serves as a back-end runtime environment that can manage server-side logic and communicate with databases and other services. This event-driven, non-blocking server-side environment is what we use to create web applications because it makes the process quick and effective.

Visual Studio Code: VS Code is a free and open-source source code editor that provides features such as debugging. Syntax highlighting, intelligent code completion, and source control integration. It is highly customizable with various extensions and themes available, making it suitable for a wide range of programming languages and frameworks. We chose VS Code for its versatility, ease of use, and strong community support.

REST API :- To securely connect the back end to the frontend, we use REST API. Use this API in specific because it allows simple integration with third-party services and offers a standardized and scalable approach for communication between various software systems.

npm: ‘npm’ is a package manager for Node.js and JavaScript, widely used for managing dependencies and packages for web development projects, and it offers a vast collection of packages and easy installation process, making it a popular choice among web developers.

Software Testing

Both white-box and black-box testing methods can be used to test software for a management system for educational institutions. Here's how you could go about it: [3]

Registration Management:

a. Black box testing: Verify that the system enables students to register for courses, collects accurate student data, and sends the proper confirmation messages following registration.
b. White box testing: Test if the registration module interacts correctly with the database and if the input validation is performing as expected. Test the system's capacity to manage several registrations concurrently and whether any race-related conditions exist.

Attendance Management:

a. Black box testing: Check whether the system enables administrators to take attendance for students, whether accurate attendance data is gathered, and whether the system produces an appropriate attendance report following the taking of attendance.
b. White box testing: Check the input validation and the way the attendance module communicates with the database. Additionally, examine if the system can manage multiple sorts of attendance techniques (such as manual or electronic).

Financial Management:

a. Black box testing: Check whether the system enables students to pay fees, whether the right fee data is gathered, and whether the system generates the right confirmation message after payment.
b. White box testing: Check the functionality of the payment gateway integration and the way the financial module communicates with it. Test the system's ability to accept various payment methods, including credit card, debit card, and payment at the class premises, as well as its security.

Timetable Management:

a. Black box testing: Test whether the system displays a schedule for each class and grade, and whether the schedule is accurate and updated.
b. White box testing: Check to see if the timetable module communicates with the database properly, generates class-clashing warnings, and performs as intended. Test the system's ability to manage various schedule restrictions (such as extra classes or exams)

Exam & Quizzes Management:

a. Black box testing: Analyze whether the system enables teachers to make examinations and quizzes, whether the right question and answer data are gathered, and whether the system generates acceptable tests when exams are created.
b. White box testing: Verify that the input validation is operating as intended and that the exam and quiz module interacts with the database correctly. Check to see if the system can handle various question formats (such as multiple-choice or essay questions).

Study Materials Management:

a. Black box testing: Check to see if the system enables teachers to submit that kind of study material, if the proper material data is gathered, and if the system generates the proper confirmation message after upload.
b. White box testing: Check the input validity and the way the study materials module communicates with the database. Also, make that the system can handle various file formats (such as Word or PDF documents) and that there are no security flaws

Assignment Management:

a. Black box testing: Test whether the system enables teachers to create assignments for each class and grade, provided that the right assignment data is gathered and that the system creates an appropriate display for students after creating the assignment.
b. White box testing: Check the assignment module's interaction with the database and the accuracy of the input validation. Test the system's ability to handle various assignment formats as well.

Support Services Management:

a. Black box testing: Test whether the system can manage student requests for assistance, such as difficulties with academic login, enrolling in exams, etc. Test whether the system can allocate assistance to each student request and monitor the status of each request. Check to see if the system can produce reports on the different support service requests and the results of each one.
b. White box analysis: Check to see if the database's request mechanism for support services functions properly. Test if the system can assign support workers to each request correctly and update the status of each request. Check to see sure the reports are generated appropriately and show the correct data.

Software Maintenance

Software maintenance for an institute management web application involves the ongoing process of updating, enhancing, and fixing issues in the application after its initial development and deployment. The following are some of the essential steps involved in software maintenance for an institute management web application:

Bug Fixes: The first step in software maintenance is to fix any bugs or issues that may arise in the application. This may involve debugging the code, identifying the root cause of the problem, and implementing a fix.

Security Updates: As security threats continue to evolve, it's essential to keep the application up-to-date with the latest security patches and updates to ensure the safety of the data.

Performance Optimization: Regular performance monitoring and optimization are necessary to ensure that the application runs smoothly and quickly. This may involve analyzing and optimizing the code, database, and other components of the application.

Documentation: As the application evolves, it's important to update the documentation to reflect any changes in the application's functionality or architecture. This can help ensure that the application remains maintainable over time.

User Support: Finally, ongoing user support is essential to ensure that users can use the application effectively and efficiently. This may involve providing training or documentation, troubleshooting issues, and answering questions.

Overall, software maintenance for an institute management web application is an ongoing process that requires regular attention and effort to ensure that the application remains effective and efficient over time.

Evaluation of our system

- Our approach seeks to offer a whole educational experience that satisfies the requirements and preferences of both teachers and students. A user-friendly interface is among the most crucial elements in attaining this goal. Our system's user interface is created to be simple and straightforward so that both teachers and students can easily manage it. The system offers a straightforward and structured approach to manage and access course materials, exams, and grades.
- Our system's capability to effortlessly generate, manage, and distribute course content is another crucial feature. This covers text, tests, and homework. The creation and organization of course materials are made simple by our system's user-friendly interface, which also makes it simple for teachers to develop and deliver content to their students.
- Our system's primary components also include teamwork and communication. We offer a variety of communication and collaboration tools for teachers and students, such as message boards, texting, and video conferencing. With the use of these technologies, students may communicate with their peers and professors, ask questions, and get feedback. In today's quick-paced, technologically advanced environment, collaboration and communication are crucial for efficient learning. Our system provides a variety of tools for collaboration and communication that help students and teachers communicate and work together effectively.
- Another resource we provide is texting, which gives students and teachers a quick and effective method to interact. Our technology also provides video conferencing, which enables teachers to hold online meetings, give real-time feedback, and hold virtual classrooms.
- Additionally, our system offers teamwork features so that students can collaborate on projects and tasks. Students can cooperate on group tasks, share resources, and critique each other's work, for instance. This not only improves their educational experience but also fosters critical thinking and teamwork.

- In conclusion, our system offers a range of communication and collaboration tools that enable students and instructors to communicate effectively, collaborate on projects, and engage with each other. By using these tools, students can learn more effectively and build the skills they need to succeed in the modern workplace.
- Furthermore, essential to our system are assessment and grading. In addition to a straightforward grading system, we provide a variety of assessment types, such as exams, quizzes, and assignments. Together with progress reports, attendance records, and grade sheets, our system also offers analytics and statistics on student achievement. To keep parents updated about their child's progress, we also provide exam results to them. Compatibility and integration are additional crucial components of our system. Our solution is compatible with a wide range of devices and browsers and is built to interface with other systems, such as student information systems. Along with extensive support and training materials, we also provide documentation, tutorials, recordings, and support services for both teachers and students.
- Overall, our system prioritizes the user experience, with an emphasis on ease of use and accessibility. We believe that our system provides a comprehensive learning experience that meets the needs and preferences of both students and teachers.

Gantt Chart

ID	Task	Week 1							Week 2							Week 3							Week 4						
		06 Feb	07 Feb	08 Feb	09 Feb	10 Feb	11 Feb	12 Feb	13 Feb	14 Feb	15 Feb	16 Feb	17 Feb	18 Feb	19 Feb	20 Feb	21 Feb	22 Feb	23 Feb	24 Feb	25 Feb	26 Feb	27 Feb	28 Feb	01 Mar	02 Mar	03 Mar	04 Mar	05 Mar
	Analysis																												
01	Requirement Gathering																												
02	Requirement Analysis																												
03	Preparing charter document																												
04	SCRUM Activity																												
05	Proposal presentation																												
06	Preparing proposal document																												
07	Preparing SRS																												

ID	Task	Week 5							Week 6							Week 7							Week 8						
		06 Mar	07 Mar	08 Mar	09 Mar	10 Mar	11 Mar	12 Mar	13 Mar	14 Mar	15 Mar	16 Mar	17 Mar	18 Mar	19 Mar	20 Mar	21 Mar	22 Mar	23 Mar	24 Mar	25 Mar	26 Mar	27 Mar	28 Mar	29 Mar	30 Mar	31 Mar	01 Apr	02 Apr
	Design																												
08	Proposal Submission and Evaluation																												
09	Wireframe drawing																												
10	User interface design																												
11	Database design (ER)																												
12	Database development																												
13	Document design specifications																						MID EXAM						

ID	Task	Week 9							Week 10							Week 11						
		03 Apr	04 Apr	05 Apr	06 Apr	07 Apr	08 Apr	09 Apr	10 Apr	11 Apr	12 Apr	13 Apr	14 Apr	15 Apr	16 Apr	17 Apr	18 Apr	19 Apr	20 Apr	21 Apr	22 A0r	23 Apr
	Development																					
14	Develop system modules																					
15	Integrate system modules																					
16	Perform initial testing																					
17	Progress Evaluation										New Year Holiday											
	Testing																					
18	Perform system testing																					
19	Debugging																					

ID	Task	Week 12							Week 13						
		24 Apr	25 Apr	26 Apr	27 Apr	28 Apr	29 Apr	30 Apr	01 May	02 May	03 May	04 May	05 May	06 May	07 May
	Implementation														
20	Final report writing														
21	Final presentation and viva														
22	Final report submission														
23	System launching														

Work Breakdown

No	Task	Sub-Task	Members
01	Requirement Gathering & Analysis	• Interview with clients • Gather requirements details. • map out project timeline -Identify exactly which team members will be involved in our project. • Create charter document. • Create an agile document and project. • Understanding project risks	Kavindi Nipuna Janani Pamitha Tharuka Gihan Kushan Ranush
02	Develop UI	**Login and Signup** • Student Signup & Login UI • Admin Login UI • Teacher Login UI • Paper class student Login UI **User view UI** • Home page • Profile page • Exam page • Service page • Payment page • Assignment page • Study material page • Timetable page **Admin View UI** • Registration page • Attendance page • Payment page • Timetable admin page • Exam creating page • Assignment page • Support Service page • Assignment page	Kavindi Nipuna Janani Pamitha Tharuka Gihan Kushan Ranush

03	Developing Database & Backend Developing	• Draw Diagrams • Create database • Insert relevant data • Create backend. • Connect tables with the frontend. • Develop the backend	Kavindi Nipuna Janani Pamitha Tharuka Gihan Kushan Ranush
04	Testing & launching	• Check validation. • Black box Testing. • White box Testing	Kavindi Nipuna Janani Pamitha Tharuka Gihan Kushan Ranush

References

[1] F. D. A. V. A. L. and L. I. , "ScienceDirect," 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2405844020306691.

[2] S. . J. and P. P. , "Emerald insight," 2010. [Online]. Available: https://www.emerald.com/insight/content/doi/10.1108/14676371011058532/full/html.

[3] ibm, "www.ibn.com," IBM, [Online]. Available: https://www.ibm.com/topics/software-testing.

[4] I. a. o. e. c. StrongLoop, "Expressjs.com," 2017. [Online]. Available: https://expressjs.com/.

[5] T. S. C. (TSC), "nodejs.org," Technical Steering Committee (TSC), [Online]. Available: https://nodejs.org/en/about/.

[6] Microsoft, "Visual Studio Code," MIcrosoft, 2015.

[7] npm, "npm Docs," [Online]. Available: https://docs.npmjs.com/.

[8] E. H. K. R. and D. M. , "MongoDB.com," company DoubleClick, 2007. [Online]. Available: https://www.mongodb.com/docs/.

[9] I. Meta Platforms, "React.js," Meta Platforms, Inc, 2023. [Online]. Available: https://reactjs.org/.