International Journal of Research Publication and Reviews

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

University Web Portal using MERN Stack

Lakshmi Prasanna Pidaparthi[1], Dr. V.Aruna[2]

[a]Lakshmi Prasanna Pidaparthi, Student, MCA, Aurora Deemed to be University, PG student, Aurora Higher Education and Research Academy, (Deemed to be University),Hyderabad, Telangana.

[b]Dr.V.Aruna Dean of Academics, Aurora Deemed to be University, Faculty, Aurora Higher Education and Research Academy, (Deemed to be University)Hyderabad, Telangana.

ABSTRACT:

General university websites are static, have minimal basic info, and are incapable of processing real-time student interactions. That is, there is a delay in communication, old announcements, and bad handling of queries. The University Website proposed built on the MERN (MongoDB, Express.js, React.js, Node.js) stack addresses these issues as it provides an interactive, responsive, and cloud-based system.The site offers a student portal for education via which courses, program details, activities, and notices are published, and also a safe admin control panel via which administrators process student queries, modify course materials, and download data reports. Role-based login, auto-filtering of the questions, and CSV export make administrators more efficient and the students more active.Testing and evaluation reveal that the system is efficient, scalable, and responsive. It effectively minimizes paperwork, makes everything transparent, and enhances communication between the students and the university. The model suggested can be adopted by other institutions to apply their digitalization plan.

Keywords: MERN stack, University Website, Digital Transformation, Admin Dashboard, Automation.

Introduction:

Universities nowadays require websites not only for the presentation of the academic material but also for facilitating communications and operational effectiveness. Websites today are static and isolated where admissions rely on ancient PDFs or hand-written admissions for admission details, fee policy, and event notifications.The University Web Portal was created to bridge these loopholes. The portal is an MERN stack-based centralized interactive application. The portal provides:Students to look for courses, programs, and events.Admins to add academic content and act suitably on complaints brought forth by students via one assigned dashboard.Protected login-based access for guarding sensitive information.This system, unlike standard websites, provides real-time updations, cloud-readiness for deployment, and mobile-device compatibility. It provides higher transparency and facilitates proper communication among students and teachers

Review of Literature:

Previous research into the history of university site development shows that the earliest sites built with HTML and PHP were limited to providing static information alone without any interactive real-time content. Later, the combination of MySQL and PHP introduced primitive content management capabilities but systems still failed to incorporate the aspect of automation and scalability. The majority of the other studies, particularly the Search Engine Optimization (SEO) studies, focused extensively on website ranking and visibility but did not include the component of backend query management or administration efficiency. New trends provide the use of the MERN stack as the modern alternative to making modular and scalable web portals. Features such as role-based authentication, dynamic dashboards, and query filtering with analytics are evolving Launched early university websites using HTML and PHP had no way to interact in real time except for delivering static content. Subsequently, the combination of MySQL along with PHP facilitated easy content management, albeit the systems were manual and non-scalable. The majority of the research, especially that with a focus on Search Engine Optimization (SEO), prioritized optimization of website rank and visibility but did not consider using backend query processing or administrative effectiveness. Current trends see utilizing the MERN stack as a contemporary solution to creating web portal modules in a scalable and modulable manner. Role-based authentication, dynamic dashboards, and analytics-based filtering of queries are becoming the norm in higher education systems. But all the previous solutions have either been frontend-based or concerned with nothing more than content rendering, without considering higher-level features. The University Website described here bridges this gap through the combination of MERN stack technologies with live responsiveness, automation of query management, CSV export for analysis, and mobile-first capabilities, thus providing an end-to-end as well as future-proof solution. However, existing solutions have either focused primarily on frontend development or only restricted themselves to mere content distribution, not providing advanced functionalities. The hinted University Website addresses this gap through the incorporation of MERN stack technologies with real-time interactivity, query auto-handling, CSV export for analysis, and mobile-first responsive design, hence delivering an end-to-end and future-proof solution.

Methodology:

Existing Methodology

In current event planning, most teams continue to use disparate tools such as spreadsheets, emails, phone conversations, and messaging apps to coordinate with clients and vendors. This dispersed method causes significant inefficiencies. It takes teams precious time to reconcile data, manually update multiple records, and correct conflicting versions of event information.Errors are frequent—duplicated or missing details can result in scheduling issues, incorrect vendor orders, and last-minute issues on the event day. Customer information is usually held in various locations, making it difficult to offer personalized services or communicate effectively.Due to this, communication between stakeholders is inconsistent, causing confusion, delays, and lost opportunities to provide additional services. Such challenges not only drive operating expenses up but also decrease clients' trust and satisfaction levels. Thus, manual processes are now no longer feasible in managing or growing contemporary event planning.

Proposed Methodology Using Software Testing

The University Website presents a centralized online platform with the following modules:

Student Portal: View programs, ask queries, and see events, submit the responses and they can get the required answer for their queries

Admin Dashboard: Password-protected login, filter queries, reply to students.

Query Management: Issues saved in MongoDB with timestamps.

Filtering System: Admin filters queries by issue category or course.

Response System: Admin responses saved in database; future potential includes email notification.

CSV Export: Export student queries for records/analytics.

Scalability: Cloud-ready deployment with MongoDB Atlas.

Software Testing Approach

The backend for all the server requests and data was developed using Node.js and Express.js. We used secure admin login with JSON Web Tokens (JWT). The backend handles submitting queries, course information, and responses by the admin.We used MongoDB as the database because it is highly flexible and can store data in a plain JSON-like format. MongoDB stores all the queries, responses, and course information securely.We tested at all points. We tested small pieces of the system (unit testing) first, and we tested how the pieces interacted. with each other (integration testing).Then the backend was established with Node.js and Express.js to manage all of the server requests and information. Secure login as an admin was applied with JSON Web Tokens (JWT). The backend also manages sending queries, course data, and admin responses.MongoDB was employed for the database since it can be dynamic and store information in plain JSON-like format. Response, query, and course data are all stored securely in MongoDB.

Lastly, the project was hosted on a cloud platform such as Render or AWS so it is accessible at any time. Passwords and database links were kept safe using environThe University Website was created with the MERN stack, which encompasses MongoDB for data storage, Express.js and Node.js for creating the backend, and React.js for the frontend. It was chosen because it is light, scalable, and manageable. Incremental development was performed. Frontend was initially developed using React.js to implement a minimum mobile-first UI. Students can simply see courses, ask questions, and see announcements. Tailwind CSS was used to style the look and feel of neat and responsive on all devices. Secondly, Express.js and Node.js were utilized to develop the backend to manage all requests from servers as well as data. Secure admin login was also implemented with JSON Web Tokens (JWT). Backend also includes submission of queries, course information, and admin feedback. MongoDB was utilized as a database because it is flexible and can hold data in plain JSON-like data type. All regarding responses, queries, and courses' data is safely stored in MongoDB. We had tested each step. We had tested small fragments of the system individually first before doing anything else (unit testing), then how they fit together (integration testing). Tested how quickly the system was with large groups of people at once and tested security to make sure only people who are supposed to be able to view sensitive information can.Lastly, the project was hosted on a cloud host such as Render or AWS in order to make it accessibleClick to see full answer The MERN stack was used to make the University Website with MongoDB storing data, Express.js and Node.js for organizing the backend, and React.js for the frontend.

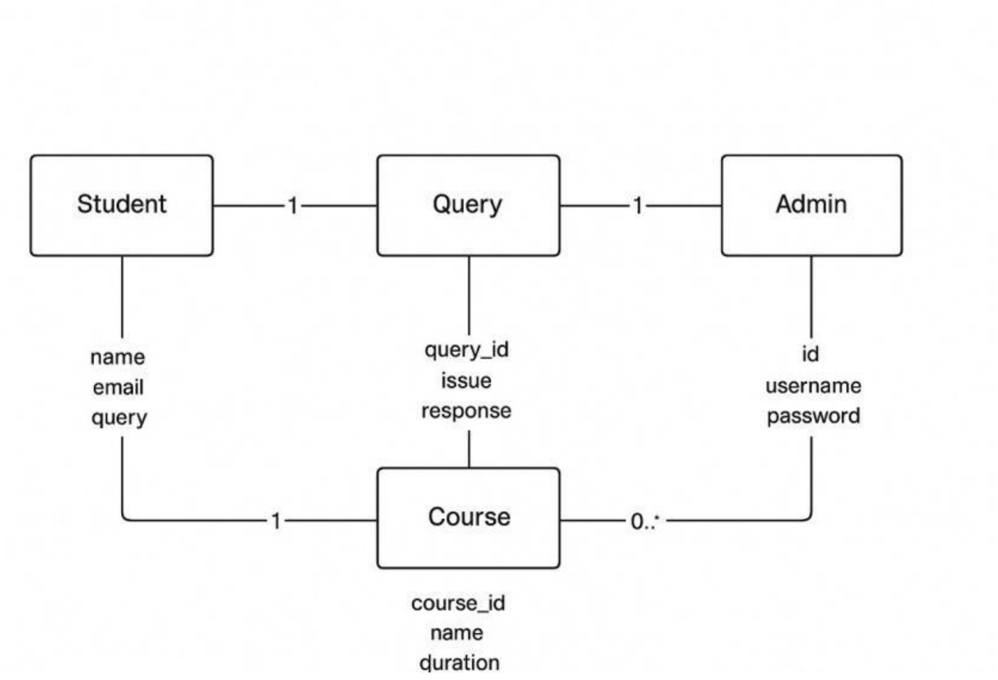
The query, response, and course information are safely stored in MongoDB.We attempted at all levels. We attempted small elements of the system first (unit testing) and then how the elements cooperated (integration testing). We also attempted how fast the system ran with large numbers of people using it simultaneously and security so that only the right individuals could see private information.Finally, the project was deployed on a cloud platform like Render or AWS so that it could be accessed at any time. Database links and passwords were also encrypted using environment variables.Through this simple means, the site was made secure, accessible, mobile, and scaleable for use in the future.This simple solution made the website secure, easy to use, mobile friendly, and scaleable to meet future needs.ment variables.This simple strategy made the site secure, intuitive, easy to use, and scaleable to future needs. Environment variables have been used in order to keep passwords and database connections secure.This simple strategy made the site secure, easy to use, mobile friendly, and scaleable to future needs

system architecture

The University Website is built with the MERN stack to ensure it is scalable and responsive. The frontend is written with React.js and styled using Tailwind CSS, offering a modular and mobile-friendly interface for navigability. The backend is Node.js with Express.js, offering RESTful APIs to enable data operations and client-server interactions. MongoDB serves as the database, securely storing queries, course data, and admin feedback. For secure login and restricted access, JWT authentication is used to protect sensitive information in this project

The architecture is modular, following a workflow model: Home Page → Course Listing → Contact Form → Query Submission → Admin Login → Dashboard → Filter/Respond → Export Reports.

Fig:1 (sequence Diagram)



Results

Student questions posted through the portal were securely stored in MongoDB in real-time with proper data handling. Questions could be easily filtered by course or problem type through the admin panel, and answers could be saved without wastage. CSV export functionality was also fine with data analysis offline and records kept. The portal was end-to-end responsive with the same user experience on desktops, tablets, and mobile phones. Load testing ensured the system's performance even under concurrent concurrent submission and demonstrated it to be scalable and resilient against high traffic without experiencing any performance loss.

Advantages of EMS

The university website offers the below advantages of different functionalities. It offers one-stop solution for handling students' queries, course details, and announcements, reducing the need for manual systems. Automated query management system saves students' and administrators' time by offering instant responses. Role-based secure login safeguards sensitive data and makes it accessible only to the concerned. The platform is mobile-friendly and responsive, easy to navigate on any device. Offline storage and examination are eased through its CSV export capability. Cloud-deployability makes the system scalable, secure, and suitable for institutions seeking modern digital platforms for communication and management.

Fig:2(Flow chart)

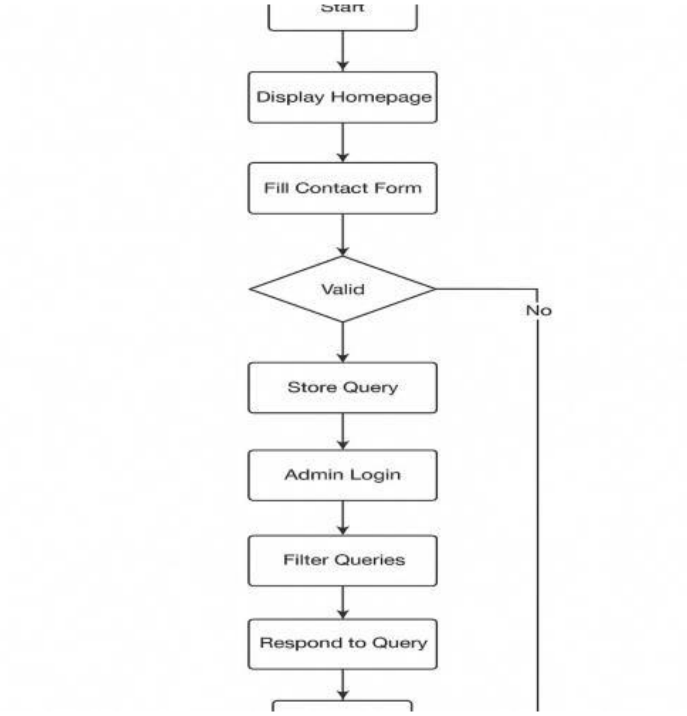


Fig:3 (use case Diagram)

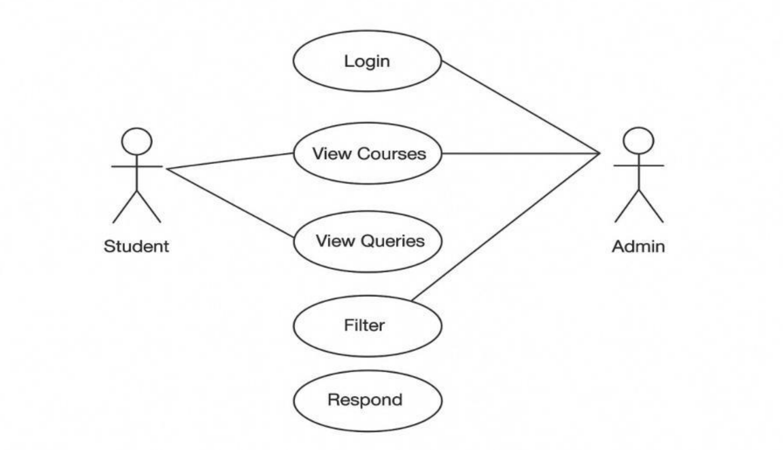
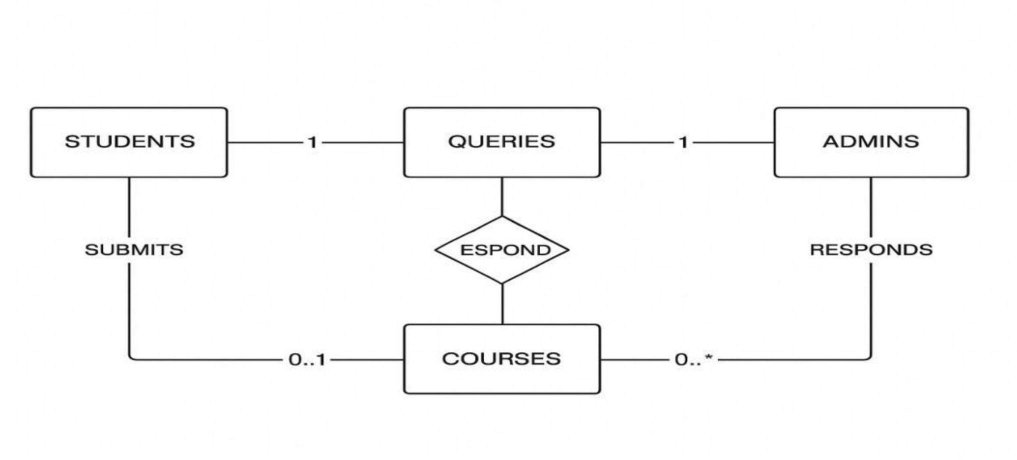


Fig:4(E-R diagram)



Conclusions:

The University Website based on MERN stack successfully counters the drawbacks of static university websites that, hitherto, have been a limitation by offering a dynamic, secure, and responsive website. It facilitates smooth communication among students and administrators with real-time submission, filtering, and handling of queries and alleviates the data management load through CSV export facility. The system offers role-based authentication, thereby both secure and reliable for institutional application. With mobile-first and cloud-deployment architecture, the website is scalable and future-proof. This work illustrates how new web technology can revolutionize education management systems to position them as a launchpad for further development and integration of advanced features.

Acknowledgement

The authors wish to thank everyone who has contributed to the success of this research work.

REFERENCES:

1. Wang Bin, Gao Bingyun, Liu Peishun, Li Xiaoqing, "A Study on Tactics for College Website at Search Engine Optimization", IEEE 3rd International Conference on Big Data Analysis, 2018.

2. Rohan Shantaram Padwal et al., "College Website Development", IRJMETS, Vol. 5, Issue 8, 2021.

3. Md. Faiz Anwar et al., "Online College Management System", IJERT, Vol. 10, Issue 10, 2021.

4. Mozilla Developer Network (MDN Web Docs) – HTML, CSS, JavaScript Documentation.

5. React Documentation – ReactJS Official Website.

6. MongoDB Documentation – MongoDB Official Website. these are the refrences which i refered during this project