iJRASET
International Journal For Research in Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 12 **Issue:** III **Month of publication:** March 2024

DOI: https://doi.org/10.22214/ijraset.2024.59305

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 12 Issue III Mar 2024- Available at www.ijraset.com

Design and Implementation of Educonnect: A Cloud Native Application for Educational Institutions

Srinivasa Rao Pendela[1], Gnana Chandra Vutukuri[2], Himana Ruttula[3], Venkata Sumathi Yadala[4], Sri Lakshmi Pujitha Tarigopula[5]
[1]*Associate Professor,* [2, 3, 4, 5]*UG Students, Department of CSE, Vasireddy Venkatadri Institute of Technology (Autonomous), Guntur, AP*

Abstract: ***In modern software environments, a paradigm known as the MERN stack has emerged as an important force due to its breadth of development and management capabilities. Leveraging native MERN (MongoDB, Express.js, React.js, Node.js) software development, deployment, and management techniques with Cloud services, "Educonnect" emerges as an innovative solution for educational institutions seeking efficient and cost-effective data management. This paper deals with the implementation of a cloud native web application Educonnect after a dexterous development strategy to create a cloud native web application. The main focus of this cloud native application is to provide a solution for data organization and collection along with an effective way of sending communication to the students by using cloud services***
Keywords: MERN stack, Cloud Native, Google Cloud Platform, CI/CD, Educational Institutions

I. INTRODUCTION

In the current educational landscape, cloud-native applications created using the MERN stack have gained more popularity and are considered more reliable. Recognizing this trend, we are introducing the Educonnect application, a cloud-native, MERN stack-powered solution tailored specifically for educational institutions for maintaining students personal and academic records at the same place. This application reduces the burden of managing student details, courses, certifications, announcements, internships, and job opportunities. It also assists college management in efficiently retrieving specific student details, such as those who have completed particular certifications or internships with specific skill sets. Manual data collection, which is prone to errors and requires significant effort, is thereby minimized. Moreover, the traditional methods of communication, such as WhatsApp messages or class announcements, are replaced by utilizing notifications from the application, facilitated by Firebase Cloud Messaging service.

II. LITERATURE REVIEW

Various studies had been done on the limitations, impacts of using on premises application and MERN stack web application using cloud services. On premises applications need to be installed in the systems of institutions though we have control over the systems and information of the students and handling of the data we need a lot of staff for maintenance. Also, backup storage is required for the on-premises application but using MERN with cloud services we can overcome these limitations.

1) Vaishali Gentyal, Ritesh pati, Vaishnavi, Gauri, and Devyani ambi presented the web application for college using MERN stack. To boost productivity and to increase the efficiency of the educational system MERN stack is used to develop the quality college management system.
2) Spoorthi M S, Kavana V, Koushik S N, Veena M proposed a paper on A Review On Placement Management System. This System provides the facility of viewing both the personal and academic information of the student and company.
3) Muhammed Kuliya, Ashok Sharma, Abdulkadir and Tarandeep kaur presented the application of cloud computing in education. They provided the solution for resource availability, cost control through cloud computing in the education institution.
4) Saijal Gupta, Gaurav Kumar, Akshaya Renganathan, Pooja and Akshay Jain developed an online college portal using MERN stack. This system provides information to all levels of management in the college. The students and staff can use the system to access the information related to the college.

©IJRASET: All Rights are Reserved

We preferred this architecture and added some additional features by integrating cloud services. MERN stack applications, which are cloud-native, inherently offer superior scalability. This means that as the educational institution grows or experiences varying workloads, the application can adapt to handle increased data and user demands. Unlike on-premises solutions that may require hardware upgrades, Cloud Native applications can scale dynamically in response to evolving requirements.

We can also observe many successful examples of applications that uses MERN stack and cloud services, Spotify extensively uses Google Cloud (GCP) for cloud services, React.js for the front end, Node.js for the backend, and MongoDB for data storage. WhatsApp, owned by Facebook, is known to use Facebook's infrastructure for cloud services, React.js for the web interface, Node.js for the server, and MongoDB for data storage.

Coming to the cost considerations an on-premises solution can have a high total cost of ownership (TCO) than a cloud system. Large initial purchases are typically required for on-premises systems, which can often lead to higher needed capital expenditures in terms of support and functional improvements so maintenance costs should also be included. You are responsible for server hardware, software, data backup, storage, and disaster recovery with in-house systems. Dubbed more than on-premises for flexibility, reliability and security, cloud eliminates the hassle of maintaining and updating systems, enabling you to invest your time, money and resources into enjoying your central business processes and providing real-time access to policy settings and records.

III. METHODOLOGY

This application is developed using the MERN (MongoDB, Express.js, React.js, Node.js) Stack, which is a collection of technologies commonly used together for building full-stack web applications. It also includes the integration of various Google Cloud Services (such as Google Cloud Storage, App Engine, etc) to enhance its functionality. Alongside the MERN Stack, the application follows the MVC (Model-View-Controller) architecture, a widely adopted design pattern in software development. It follows a 3-tier architecture, where:

1) The Model represents the data of the application.
2) The View presents the user interface to the end user.
3) The Controller serves as an intermediary between the Model and the View and contains business logic.

This architectural approach enhances modularity, maintainability, and scalability of the application by separating concerns and promoting code reusability.

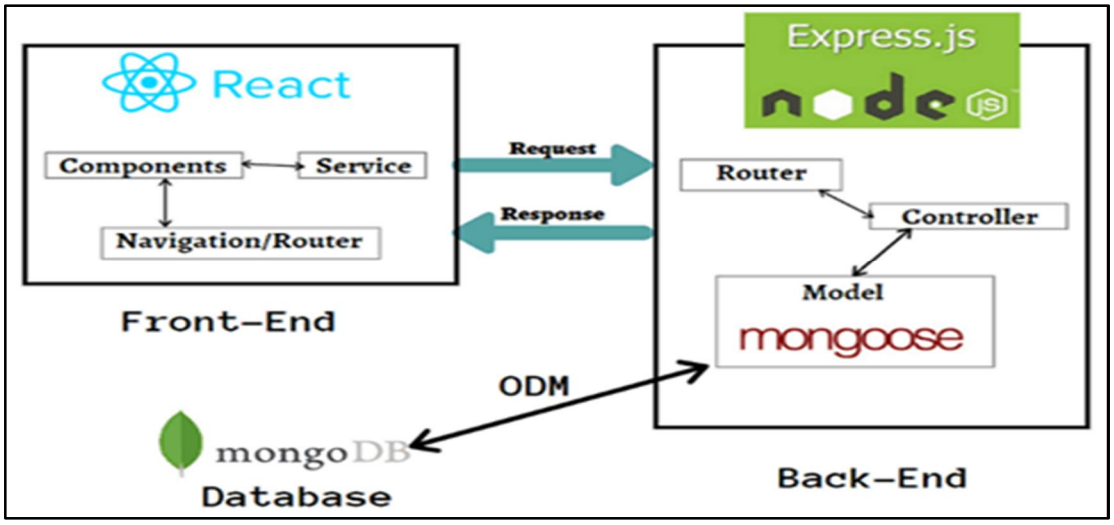


Fig.1 Communication Flow in MERN Stack's MVC Architecture.

A. *Client Side*

At the client side, users can access the web application through any device connected to the internet. The client-side of the application is developed using React.js, Tailwind CSS, Shadcn UI, Redux Toolkit. React.js allows us to create SPAs.

1) *React.js:* React.js, commonly referred to as React, is an open-source JavaScript library developed by Facebook for building user interfaces (UIs) for web applications. It is often used for creating single-page applications (SPAs) and dynamic web interfaces. React allows developers to build reusable components that can be composed together to create complex UIs.

2) *Tailwind CSS and Shadcn UI:* Tailwind CSS is a utility-first CSS framework that enables developers to build custom designs without writing custom CSS. It offers a set of low-level utility classes representing specific CSS properties. Shadcn UI is a modern and feature-rich UI library providing pre-designed components and styles.
3) *Redux Toolkit:* Redux Toolkit is used to maintain and update data across your applications for multiple components to share, all while remaining independent of the components.

B. *Server Side*

The server-side contains the business logic of the application, developed using Node.js and Express.js. A NoSQL database MongoDB is used for storing user data. Redis is used for caching the frequently accessed data.

1) *Node.js and Express.js:* Node.js is a JavaScript runtime environment that executes JavaScript code outside of a browser. It is built on Chrome's V8 Javascript engine. Express.js is a minimalist web framework for Node.js, simplifying server-side application development. It provides robust features for handling HTTP requests, routing, and middleware.
2) *MongoDB:* MongoDB is a popular NoSQL database management system, known for its flexibility, scalability, and performance. It stores data in a flexible, JSON-like format called BSON (Binary JSON) and is designed to handle large volumes of data and high throughput applications.
3) *Redis:* Redis is an open-source, in-memory data structure store, utilized primarily as a high-performance cache and message broker. It is used for storing frequently accessed data to reduce the read operations on the database thereby increasing the performance.

C. *Google Cloud*

Google Cloud is a suite of cloud computing services offered by Google, providing a range of infrastructure and platform services for building, deploying, and managing applications and services on Google's infrastructure. Various google cloud services are used in this application to provide certain functionalities.

1) *Google Cloud Storage:* It is a cloud-based object storage service offered by Google Cloud Platform. It allows users to store and manage data in a highly available, secure, and scalable manner.
2) *Google Cloud Build:* It is a fully managed continuous integration/continuous deployment (CI/CD) platform provided by Google Cloud Platform. It automates the process of building, testing, and deploying applications in the cloud.

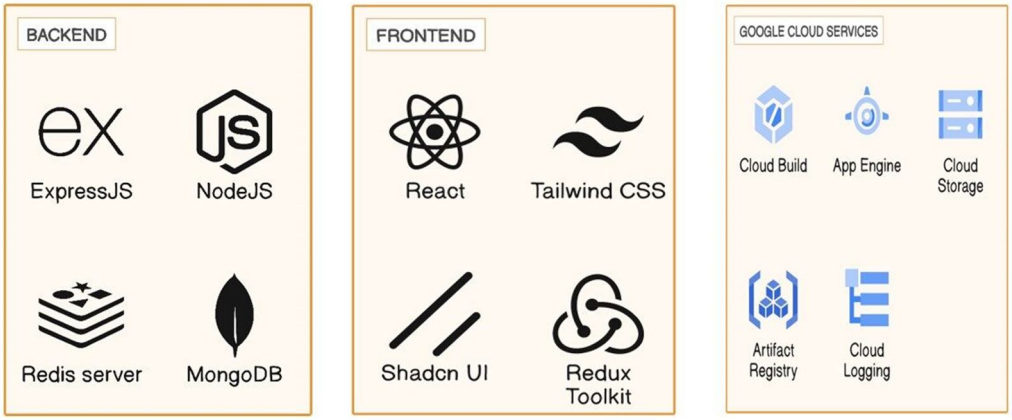


Fig.2 Tech Stack of Frontend, Backend and Google Cloud Services

IV. IMPLEMENTATION

A. *Client Side*

The application is developed for all devices which are connected to the internet and is accessible from anywhere and anytime. Based on the role of the user different screens will be displayed and is permitted to perform actions based on the permissions available.

1) *Secure User Authentication and Authorization using JWT:* In the designed authentication workflow, users interact with a responsive UI to log in using their email, password, and selected role. Alternatively, students can register by providing necessary details such as name, email, roll number, and an image. The user data, including email and hashed passwords, is securely stored in a MongoDB database.

Authentication and authorization processes are managed through JSON Web Tokens (JWT), which securely transmit user information in encrypted form. When a user attempts to log in, the authentication service verifies their credentials against the database. Upon successful authentication, a JWT containing relevant user data like ID, roles, and permissions is generated and issued to the client for storage. Subsequent requests requiring authentication involve the client presenting this JWT to the server. The server verifies the token's integrity to ensure it hasn't been tampered with, then authenticates the user based on its contents. Additionally, the server evaluates the user's permissions stored within the JWT to determine their access rights. Based on this evaluation, the server responds accordingly, either granting access to requested resources or denying it. This structured approach ensures a secure and seamless authentication and authorization process, safeguarding user data while facilitating access to authorized resources.

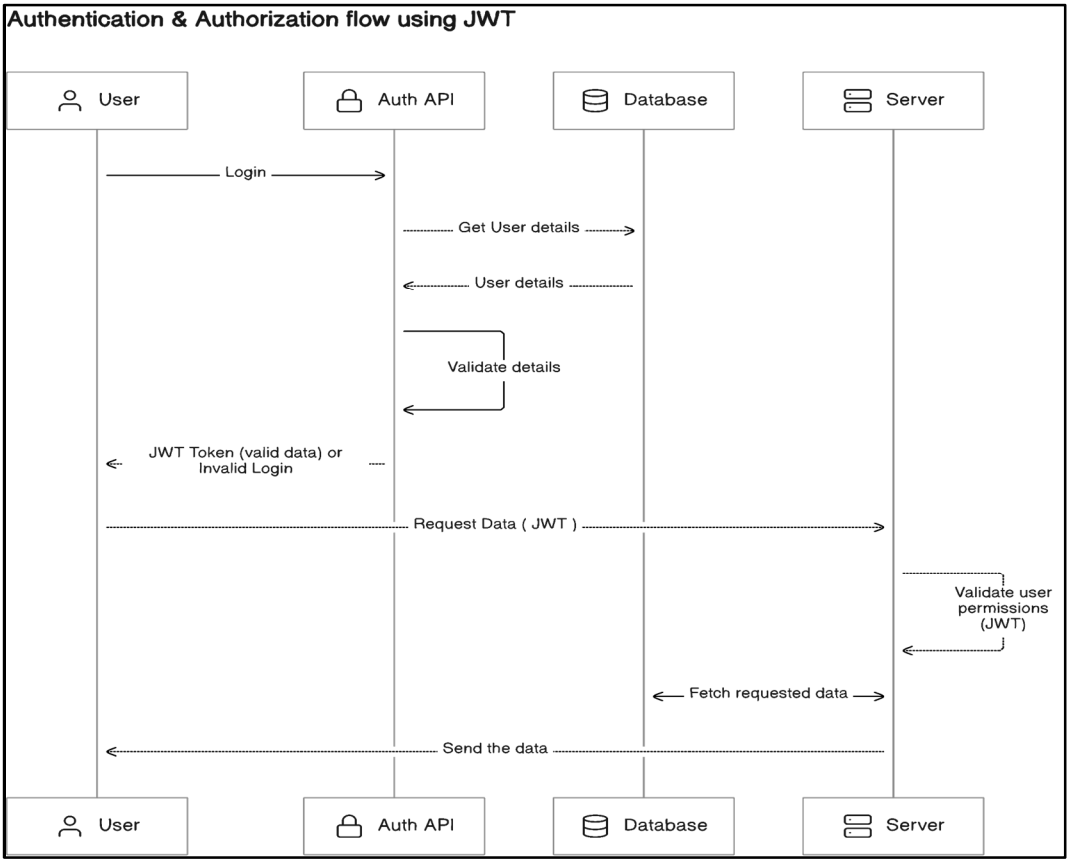


Fig.3 Authentication and Authorization using JWT

2) *Administrator and Coordinator Module:* Administrators and Coordinators are designated users with specific access privileges within the application. The Administrator oversees the comprehensive management of student data, encompassing personal and academic information. Whereas, Coordinators possess access limited to student data within a particular branch. Admin tasks involve managing department coordinators, issuing alerts to students through push notifications, organizing job drives, overseeing student data management, generating customized reports, verifying internship permission requests, and collating data of students interested in posted job opportunities. Coordinators, on the other hand, have access solely to student data within their respective departments, enabling them to send alerts, verify internships, and generate department-specific reports.
3) *Student Module:* Students register by completing a form with required details like name, email, roll number, profile photo, branch, and year of passing. Upon successful submission, this data is stored in MongoDB. They then log in using their registered email and password to access the application. Once logged in, they can input information such as courses completed, certifications earned, achievements, and internships. Students can also view announcements from admins and coordinators, receiving push notifications for important alerts and opting into posted job drives. Additionally, they receive email notifications for critical alerts and specific job opportunities.

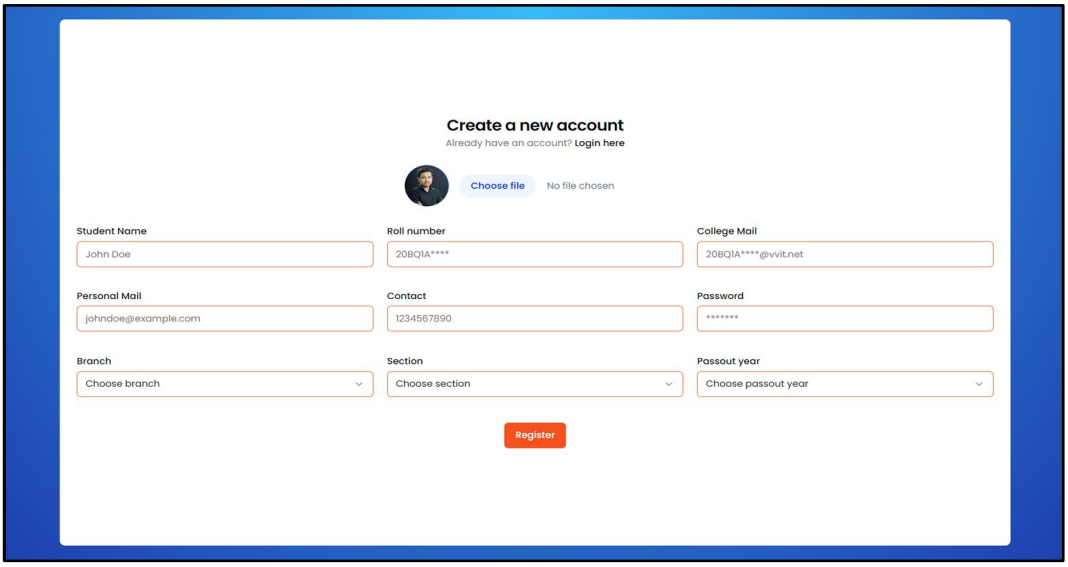


Fig. 4 Student Registration Screen

B. Server Side & Google Cloud

The server side of the application is developed by using Node.js as runtime and Express.js as a framework for creating robust and reliable web server. It contains the business logic and it contains the interaction with the database to read and write the data. The major functionalities such as integration with Google Cloud Services (Cloud Storage, Cloud Logging etc) are done.

1) *File Upload Module:* This application utilizes Google Cloud Storage to store all application files, including student profile images, certifications, internship offer and completion letters, courses, and related files. Each file is stored in specific folders within a single bucket. Google Cloud Storage offers high durability and reliability of 99.999999999%. When a user uploads a file, it is sent to the server for validation, including format acceptance and size limitations. If validation fails, an appropriate error message is sent to the user. Valid files are uploaded to Cloud Storage using multer and the Google Cloud Storage package. Upon successful upload, a unique URL is generated for accessing the file, which is then stored in the database. Files are renamed to ensure each file in a folder has a unique name.

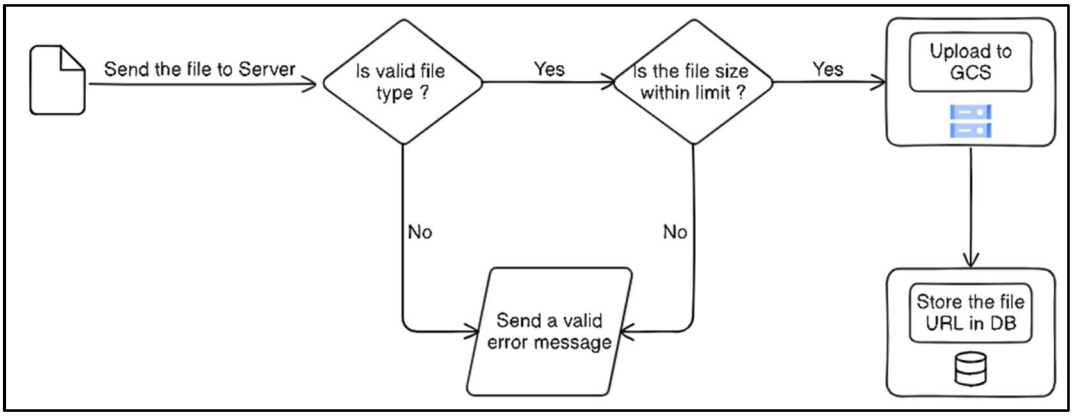


Fig.4 Flow diagram of file upload to Google Cloud Storage

2) *Cloud Services Integration:* This application is integrated with various cloud services, including Google Cloud Storage, Google Cloud Logging, and Firebase Cloud Messaging. Google Cloud Storage is utilized for storing user files in buckets within respective folders. Google Cloud Logging records all user interactions within the application, such as logins, profile updates, data requests, and modifications, enabling security checks when necessary. Firebase Cloud Messaging facilitates push notifications to students, allowing them to receive alerts sent by admins and respond accordingly. The integration of these services enhances the application's functionality and user experience.

3) *Database and Redis:* MongoDB, a NoSQL database, is utilized for storing application data. Data is organized into collections, with each collection containing related documents. Seven collections are used in this application: student, user, certifications, courses, internships, job drives, and announcements. To enhance performance and reduce read operations on frequently accessed data, Redis, an in-memory caching solution, is employed. Redis stores data in RAM, allowing for fast retrieval and reducing database operations.

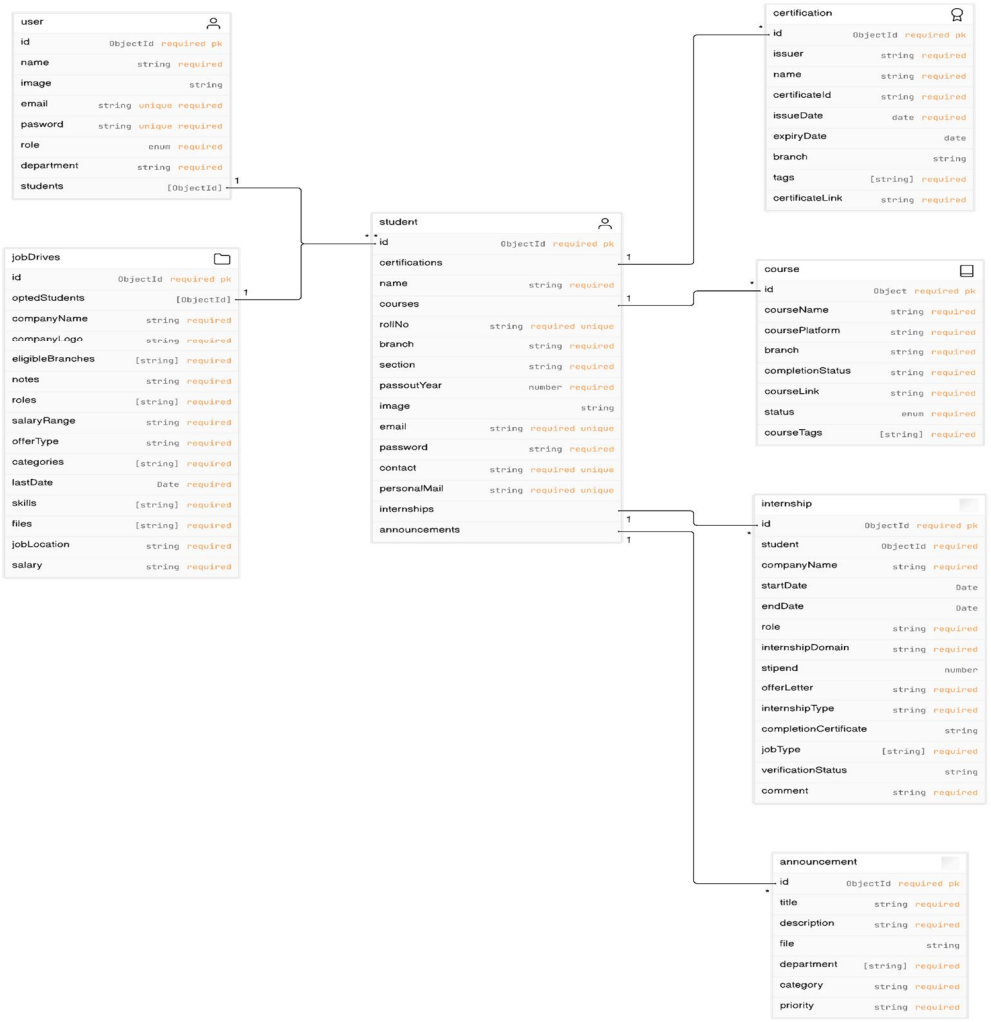


Fig.5 ER Diagram of the Database

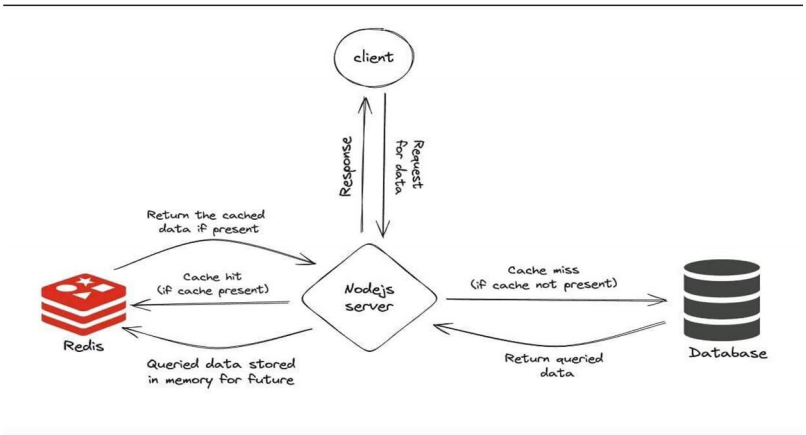


Fig.6 Interaction between Client, Server, Database and Redis

C. CI/CD Pipeline

Google Cloud offers a CI/CD pipeline with services like Cloud Build for building and testing, Artifact Registry for storing artifacts, and Cloud Deploy or Cloud Run for deployment. This lets you automate building, testing, and deploying your applications with every code change.

1) *Cloud Build:* In Cloud Build, the build process is initiated by events like code pushes or merge requests, defined through Cloud Triggers. These triggers specify when Cloud Build should commence its automated deployment steps. Within Cloud Build, these steps typically include tasks such as building Docker images, pushing them to artifact repositories, and deploying them to target services. This deployment capability extends to various Google Cloud services such as Cloud Run, App Engine, and GKE, offering flexibility in deployment options. By automating deployment processes, Cloud Build significantly reduces manual effort, streamlining the continuous delivery pipeline and enabling faster and more reliable software deployments.
2) *Artifact Registry:* In Google Cloud, Artifact Registry is the service utilized for storing project artifacts, including packages and Docker images, essential for application development and deployment. The steps for building a Docker image are typically provided in the form of a .yaml file or Dockerfile within Cloud Build configurations. These steps facilitate the generation of a Docker image, which can then be deployed across various Google Cloud Services. Within Artifact Registry, users can create and manage images, allowing for multiple versions of an application with distinct tags. The specified version, often labeled as "latest," is typically deployed to the target service, ensuring the latest iteration of the application is readily available for use. Additionally, Artifact Registry offers features for managing artifact versions, access control, and integration with CI/CD pipelines, enabling streamlined artifact management throughout the development lifecycle.
3) *App Engine:* Google Cloud's App Engine is a Platform-as-a-Service (PaaS) offering that abstracts away infrastructure management, relieving developers of the burdens associated with server maintenance. It operates in a serverless manner, automatically scaling resources to match application demand, and scaling down during periods of low traffic. App Engine boasts support for multiple programming languages and frameworks, providing developers with flexibility in their development choices. Positioned as the final step in the CI/CD pipeline, App Engine deployment occurs seamlessly via Cloud Build, utilizing Docker images obtained from the Artifact Registry. Upon successful deployment, the application is accessible via a provided URL, simplifying the process of launching and accessing web applications.

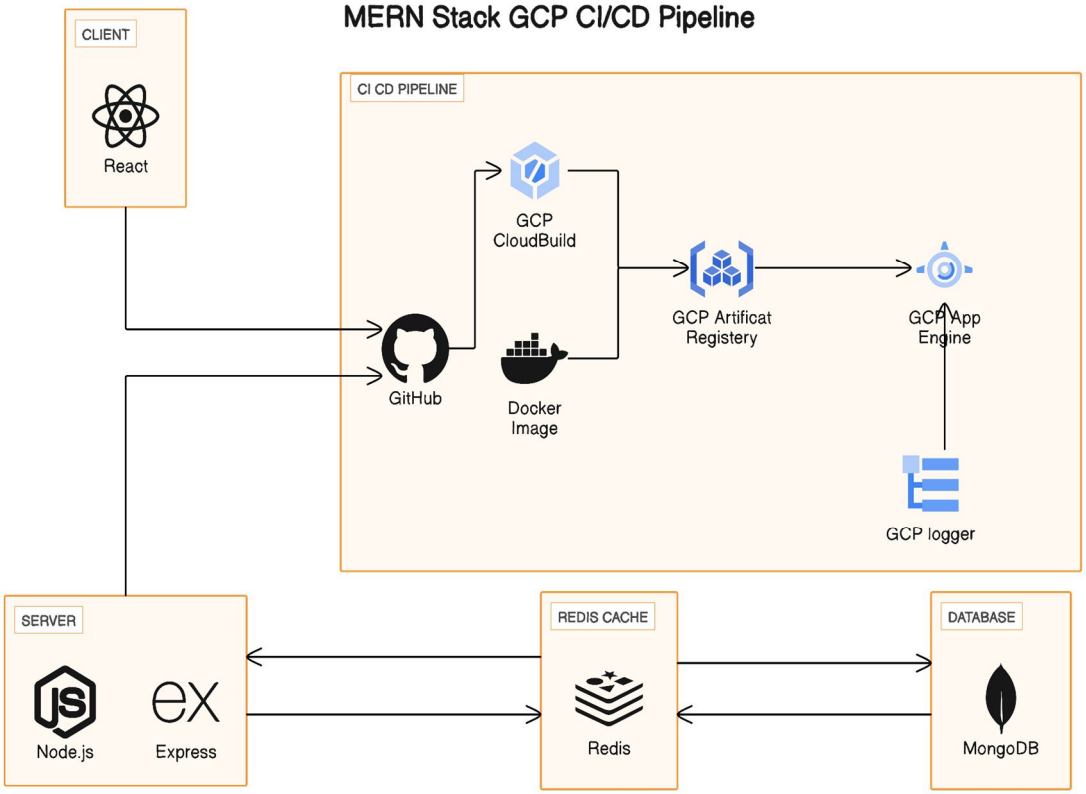


Fig. 7 Implementation of CI/CD Pipeline using Google Cloud Services

V. RESULTS

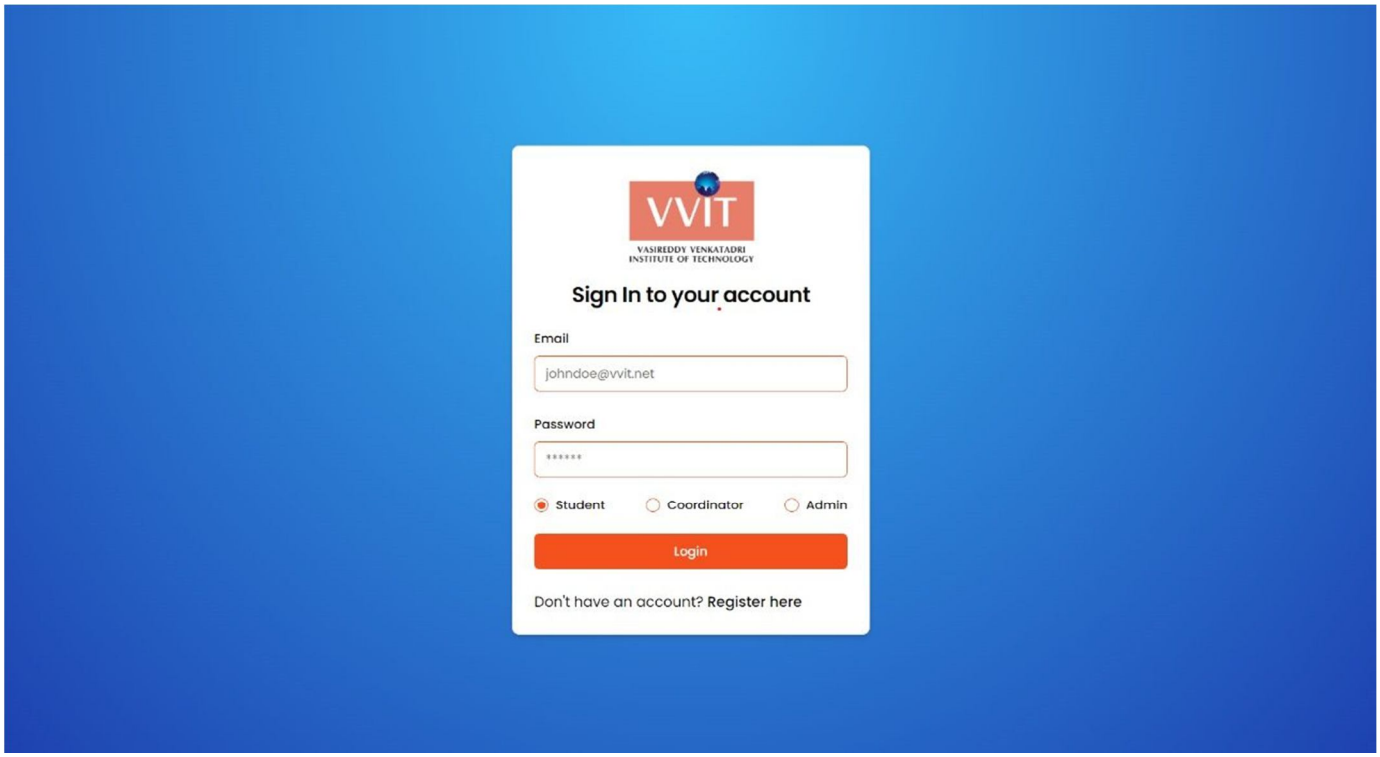


Fig.1 Login Screen for Users.

1) *Admin and Coordinator Dashboard:* This page contains graphs related to the placement statistics of the college, the details can be filtered department wise and gives an overview about placements.

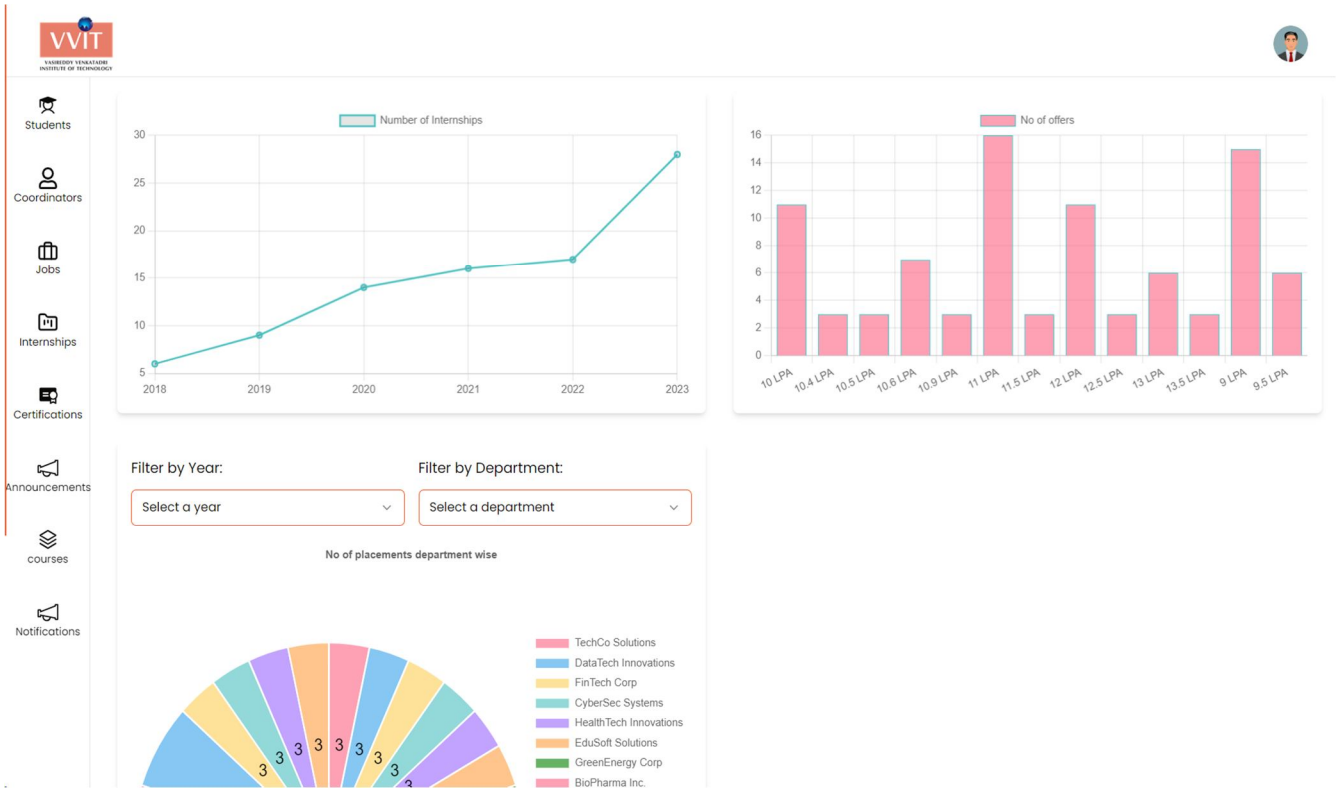


Fig. 2 Admin and Coordinator Dashboard View

2) *Students View:* This is the students view for both admin and coordinators, admin will be able to access the details of all the students whereas coordinator will be able to access the details of students of the department.

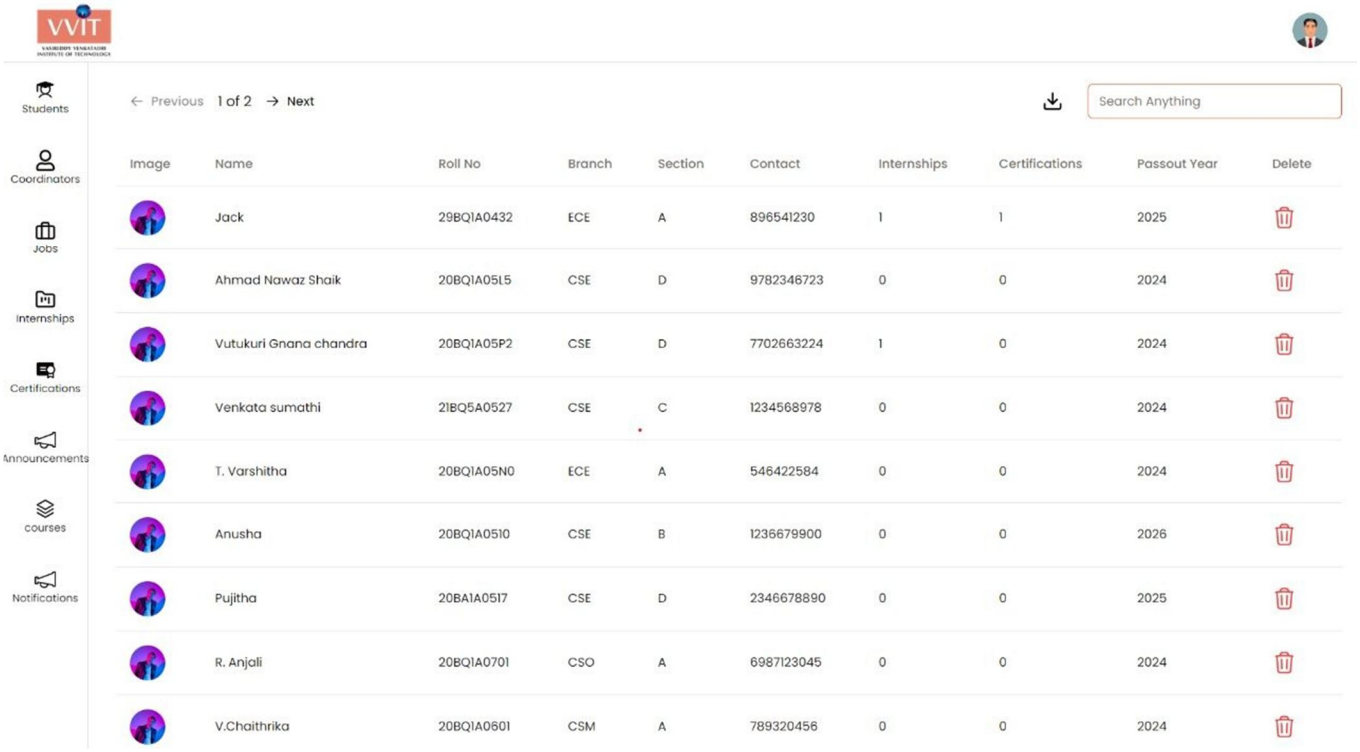


Fig. 3 Admin and Coordinator Students View

3) *Individual Student View:* Individual student details page displays both the academic and personal details of student and the details such as courses, certifications, internships etc related to the student can be accessed from this page.

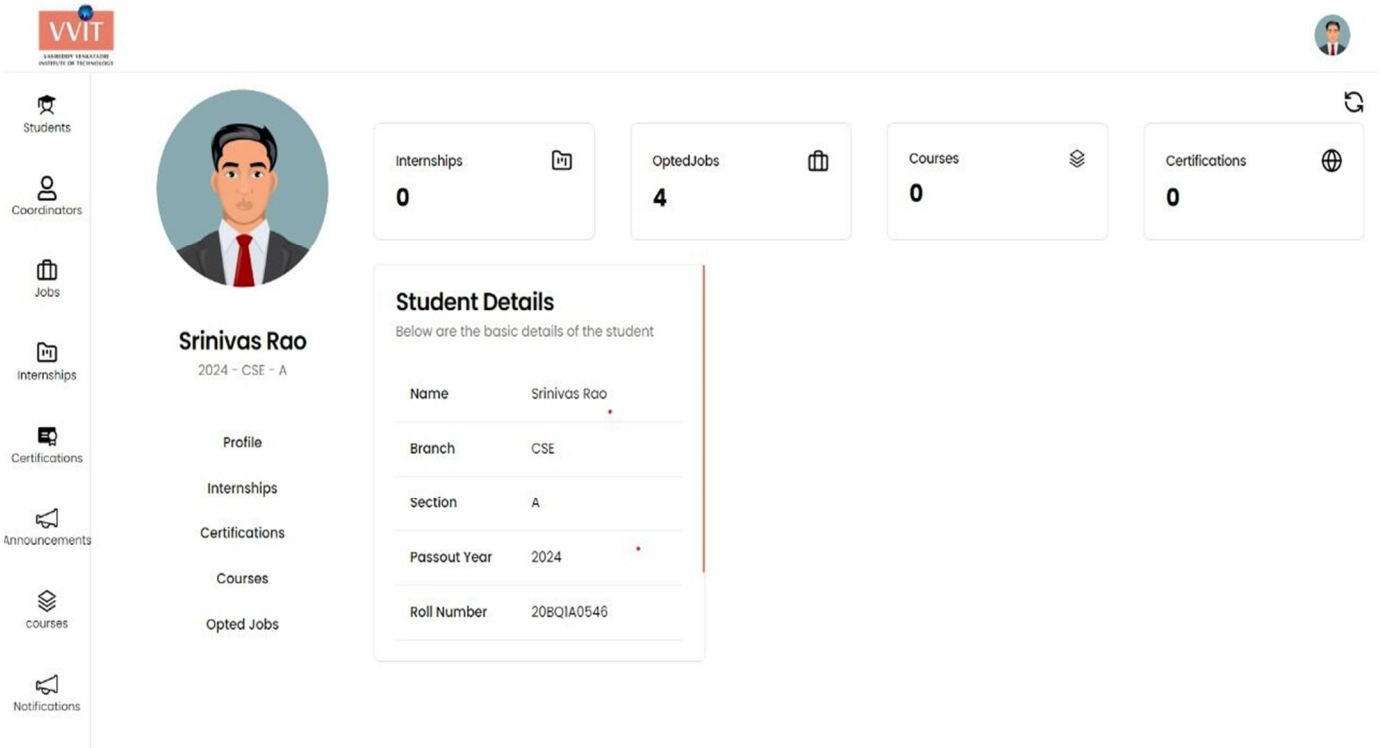


Fig. 4 Individual View of Student

4) *Coordinators View for Admin:* Admin can manage the list of coordinators, add, modify, delete coordinators based on the requirements.

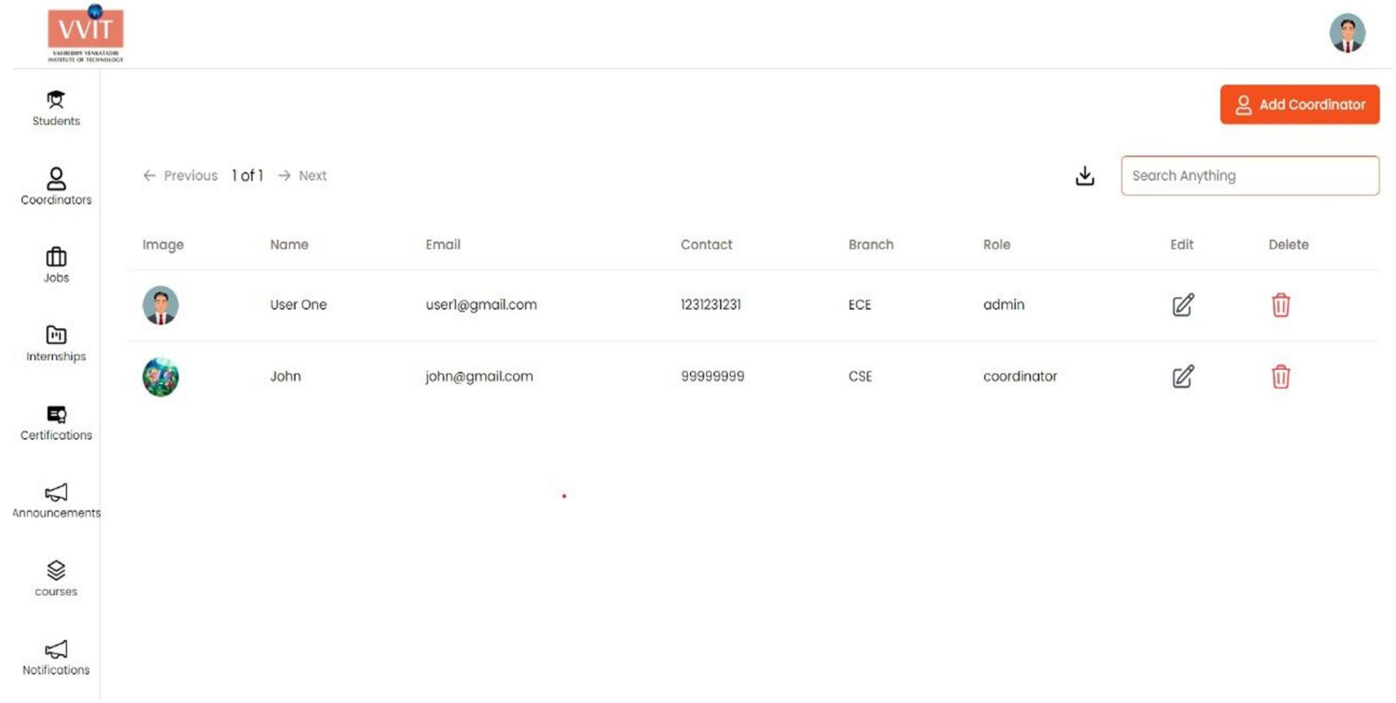


Fig. 5 Coordinators View of Admin.

5) *Internships View for Admin and Coordinator:* All the internships entered by students will be displayed here, details of each internship such as company name, stipend, offer letter, completion certificate, duration, student details will be displayed, admin or coordinator can verify the details and mark the internship as verified or rejected.

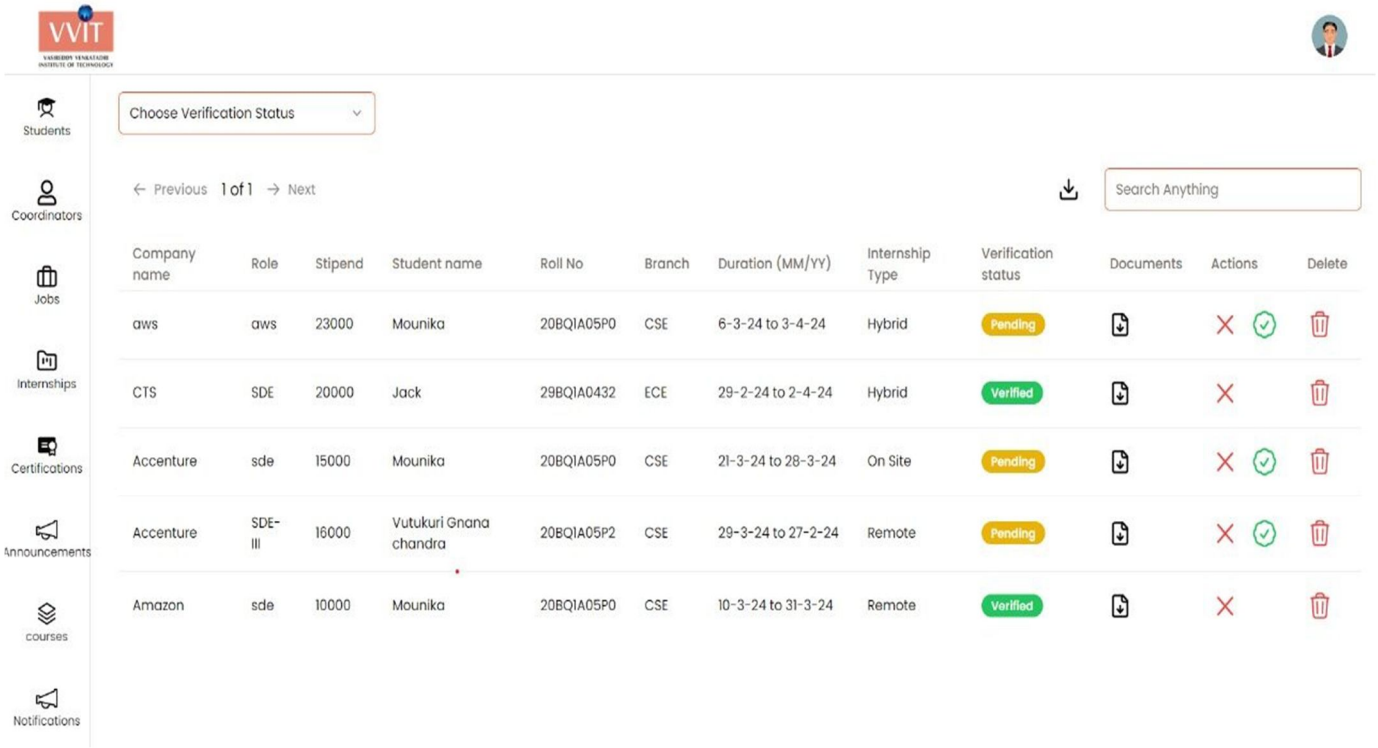


Fig. 6 Internships View for Admin and Coordinator

6) *Certification and Courses View:* The courses or certifications that are being done by students can be viewed here and the filtered data of students doing a particular certification can be downloaded in the form of excel file.

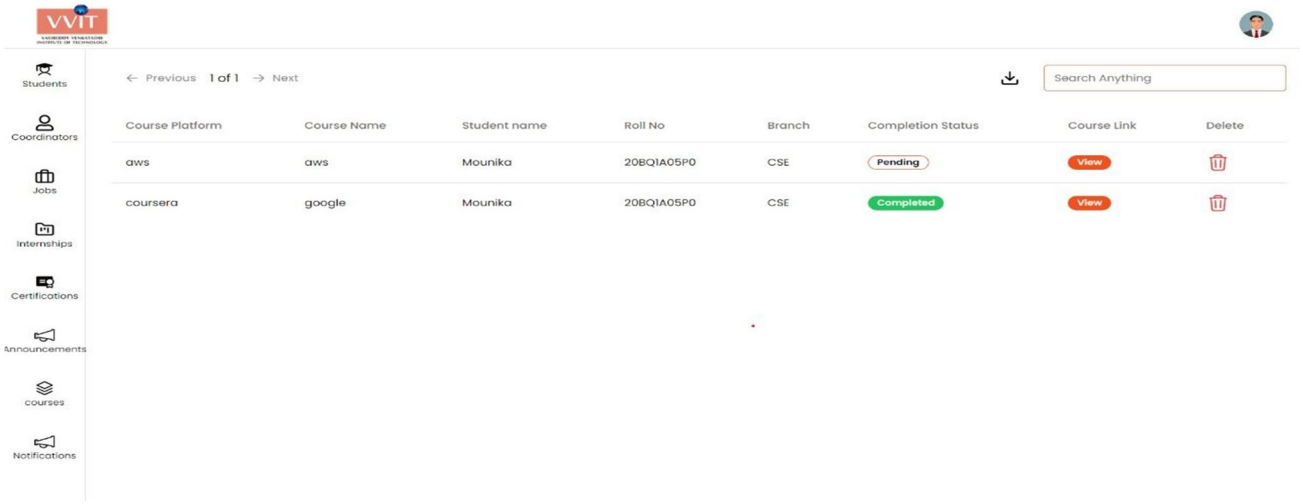


Fig. 7 Courses view

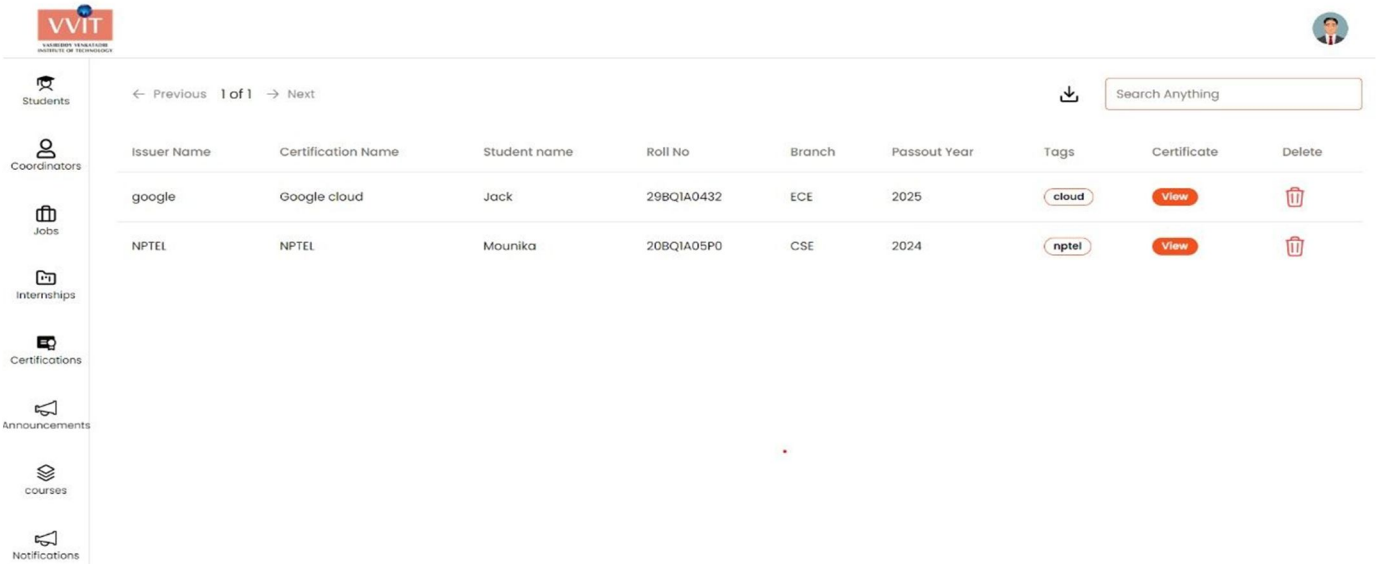


Fig. 8 Certifications View

VI. CONCLUSIONS

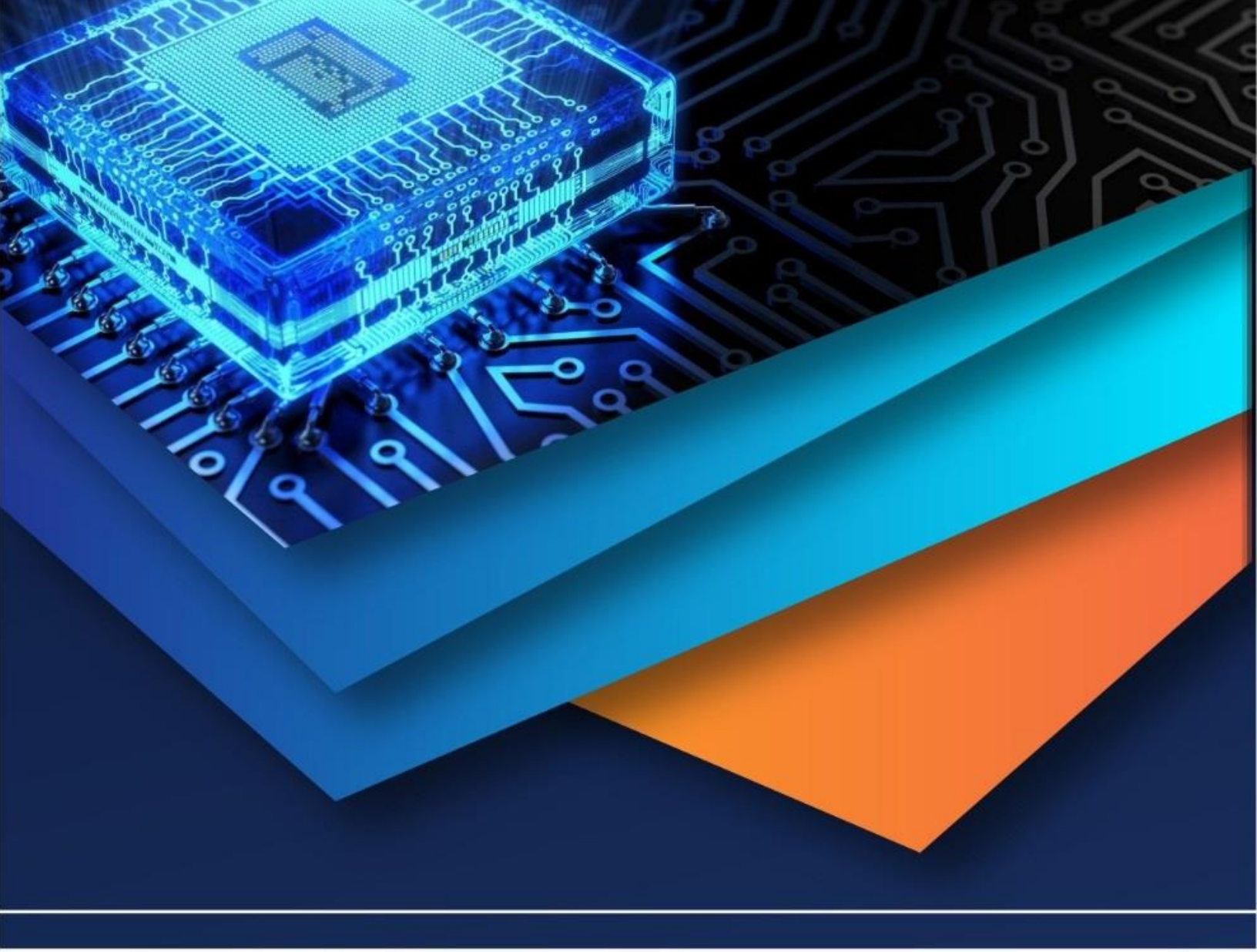
In conclusion, Educonnect stands out as a highly advantageous platform for educational institutions, facilitating efficient management of academic and personal student data. Its personalized features, such as direct alerts to students, replace reliance on WhatsApp and classroom announcements, enhancing communication within the educational community. Built using cutting-edge technologies like the MERN Stack and leveraging Google Cloud Services such as Cloud Storage and App Engine for file storage and deployment, as well as Firebase Cloud Messaging for notifications, Educonnect ensures reliability and scalability. Administrators and coordinators benefit from advanced data filtering and downloading capabilities, further enhancing administrative efficiency. Overall, integration with Google Cloud Services enhances reliability, minimizing downtime and ensuring uninterrupted access for users as needed.

REFERENCES

[1] Dr. Santosh Kumar Shukla, Shivam Dubey, Tarun Rastogi and Nikita Srivastava. Application using MERN Stack. International Journal for Modern Trends in Science and Technology 2022, 8(06), pp. 102-105.

[2] Yenduri Harshitha Lakshmi, Kuncham Pushpa Sri Vyshnavi, Shaik Fyzulla, Dr Yalamanchili Sangeetha. A Website for a Consultancy using Mern Stack. 2023 3rd International Conference on Smart Data Intelligence (ICSMDI)
[3] Neenad Sahasrabuddhe, Sanskar Khandelwal, Neha Purohit, Rishika Jain, Sonali Kalekar, Dr. Ashish K Sharma. Print–OnTheGo Remote Printing Application using MERN Stack. 2023 4th International Conference for Emerging Technology (INCET) Belgaum, India. May 26-28, 2023.
[4] Spoorthi M S, Kavana V, Koushik S N, Veena M. A Review On Placement Management System. International Journal of Creative Research Thoughts (IJCRT), | Volume 9, Issue 7 July 2021, ISSN: 2320-2882
[5] Anjali.V, Jeyalakshmi.PR, Anbubala.R, Sri Mathura devi.G, Ranjini.V. Web Based Placement Management System. International Journal of Computer Science and Information Technologies, Vol. 7 (2), 2016, 760-763
[6] Fiza Kousar, Gandharva V Hegde, Prof. Saravanan C. Design and Development of Department Placement Portal using MERN Technology. International Advanced Research Journal in Science, Engineering and Technology Impact Factor 7.105, Vol. 9, Issue 1, January 2022
[7] Ajeena Sunny, Aneena Felix, Angelin Saji, Christina Sebastian, Praseetha V.M. Placement Management System for Campus Recruitment. International Journal of Innovative Science and Research Technology ISSN No: -2456-2165, Volume 5, Issue 5, May – 2020
[8] A R Sri Nandhini, Alaine Joseph, Ajay S. Impact of Implementing Cloud Native Applications in Replacement to on-Premise Applications. International Journal of Engineering Research & Technology (IJERT)ISSN:22780181, Vol. 9 Issue 06, June-2020
[9] Tuncay Ercan. Effective use of Cloud Computing in educational Institutions. Procedia - Social and Behavioral Sciences, Volume 2, Issue 2, 2010, Pages 938-942
[10] Muhammad Mannir Ahmad Getso and Riyaz Ahmed A. H. Applications of Cloud Computing in Academic Institutions. International Journal of Information System and Engineering Volume 2, November2014

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)