

DAYANANDA SAGAR UNIVERSITY

Devarakaggalahalli, Harohalli
Kanakapura Road,Bangalore South Dt,Ramanagara ,562112, Karnataka, India



**SCHOOL OF
ENGINEERING**

**Bachelor of Technology
in
COMPUTER SCIENCE AND TECHNOLOGY**

Full Stack Development (24C2305)
Mini Project Report

“SMART CAMPUS PORTAL WITH ONLINE REGISTRATION SYSTEM”

BY
Pragathi AB – ENG24CT0015
Khushi NJ – ENG24CT0008
Pallavi B – ENG24CT0051
Anju S – ENG24CT0001
Anikashreya – ENG24CT0029

Under the supervision of
Dr. Santosh Kumar J
Associate Professor
Department of Computer Science and Technology

**DEPARTMENT OF COMPUTER SCIENCE & TECHNOLOGY,
SCHOOL OF ENGINEERING
DAYANANDA SAGAR UNIVERSITY,
(2025-2026)**

DAYANANDA SAGAR UNIVERSITY



SCHOOL OF
ENGINEERING

Department of Computer Science & Technology

Devarakaggalhalli, Harohalli
Kanakapura Road, Ramanagara - 562112, Karnataka, India

CERTIFICATE

This is to certify that the Full Stack Development Mini Project work titled "**SMART CAMPUS PORTA WITH ONLINE REGISTRATION FORM**" is carried out by **Pragathi AB(ENG24CT0015)**, **Khushi NJ (ENG24CT0008)**, **Pallavi B(ENG24CT0051)**, **Anju S(ENG24CT0001)**, **AnikaShreya(ENG24CT0029)**, bonafide students of Third semester of Bachelor of Technology in Computer Science and Technology at the School of Engineering, Dayananda Sagar University, Bangalore in partial fulfillment for the award of degree in Bachelor of Technology in Computer Science and Technology, during the year **2025-2026**.

Dr. Santosh Kumar J

Associate Professor
Dept. of CST,
School of Engineering
Dayananda Sagar University

Date: 04/12/2025

Dr. Shahina Parveen M

Chairman CST
School of Engineering
Dayananda Sagar University

DECLARATION

We, **Pragathi AB(ENG24CT0015)**, **Khushi NJ(ENG24CT0008)**, **Pallavi B(ENG24CT0051)**, **Anju S(ENG24CT0001)**, **Anika Shreya(ENG24CT0029)**, are students of Third semester B. Tech in **Computer Science and Technology**, at School of Engineering, **Dayananda Sagar University**, hereby declare that the Mini Project titled “**Title**” has been carried out by us and submitted in partial fulfilment for the award of degree in **Bachelor of Technology in Computer Science and Technology** during the academic year **2025-2026**.

Name: Pragathi AB

USN : ENG24CT0015

Name: Khushi NJ

USN : ENG24CT0008

Name: Pallavi B

USN : ENG24CT0051

Name: Anju S

USN : ENG24CT0001

Name: Anika Shreya

USN: ENG24CT0029

Place : Bangalore

Date :

ACKNOWLEDGEMENT

It is a great pleasure for us to acknowledge the assistance and support of many individuals who have been responsible for the successful completion of Full Stack Development mini project work.

First, we take this opportunity to express our sincere gratitude to School of Engineering & Technology, Dayananda Sagar University for providing us with a great opportunity to pursue our Bachelor's degree in this institution.

We would like to thank **Dr. Udaya Kumar Reddy K R**, Dean, School of Engineering & Technology, Dayananda Sagar University for his constant encouragement and expert advice.

It is a matter of immense pleasure to express our sincere thanks to **Dr. Shahina Parveen M** , Department Chairperson, Computer Science and Engineering, Dayananda Sagar University, for providing right academic guidance that made our task possible.

We would like to thank our guide **Santosh Kumar J, Associate Professor**, Dept. of Computer Science and Technology, Dayananda Sagar University, for sparing his/her valuable time to extend help in every step of our project work, which paved the way for smooth progress and fruitful culmination of the project.

We are also grateful to our family and friends who provided us with every requirement throughout the course.

We would like to thank one and all who directly or indirectly helped us in the mini Project work.

TABLE OF CONTENTS

LIST OF TABLES

ABSTRACT

	Page
CHAPTER 1 INTRODUCTION.....	1
CHAPTER 2 CHALLENGES FACED.....	2
CHAPTER 3 TOOLS USED IN PROJECT.....	3
CHAPTER 4 OUTCOMES	5
CHAPTER 5 CODE SNIPPETS.....	6
CHAPTER 6 RESULT.....	8
CONCLUSION.....	9

Abstract

This project presents the development of a full-stack **College Website** designed to provide essential institutional information and manage student registrations through a dynamic web interface. The system integrates a responsive frontend, a functional backend, and a reliable database to deliver a complete web application experience.

The frontend of the website offers an intuitive and user-friendly interface that allows users to navigate through various sections such as departments, courses, administration, facilities, and contact details. The backend processes user requests, handles data flow, and ensures smooth interaction between the interface and the server.

A key feature of the project is the **Student Registration Module**, which enables users to submit their information through an online registration form. The submitted data is stored securely in **MongoDB**, a NoSQL database chosen for its flexibility, scalability, and compatibility with JavaScript-based frameworks.

This project demonstrates practical implementation of full-stack development concepts, including UI design, API handling, database connectivity, and data management. The result is a functional, efficient, and interactive college website that highlights the core aspects of modern web development.

1. INTRODUCTION

This mini project was developed as part of our Full Stack Development course, with the goal of creating a fully functional and interactive **College Website** integrated with a database-driven registration system. The primary objective of the project was to design a platform that provides essential information about the college while enabling students to submit their registration details online in a secure and efficient manner.

The project incorporates both frontend and backend technologies, demonstrating the full-stack development lifecycle. The frontend of the website focuses on delivering a clean, responsive, and user-friendly interface using standard web technologies. Various pages were designed to represent different sections of the college, such as academic programs, departments, facilities, and contact information. These pages were structured to make navigation simple and intuitive for students, faculty, and visitors.

For the backend, we implemented server-side logic to handle registration form submissions and communication between the user interface and the database. One of the major components of the project is the **online Registration Form**, which collects personal and academic information from students. To store this data, we used **MongoDB**, a popular NoSQL database known for its flexibility, scalability, and strong compatibility with JavaScript-based applications. MongoDB allowed us to structure the student records efficiently and retrieve them quickly when needed.

Through this project, we gained practical experience in integrating the frontend with the backend, managing APIs, validating user inputs, and establishing a connection between the server and the database. We also learned how to work with modern development tools and frameworks, which enhanced our understanding of real-world web application development.

Overall, this project not only strengthened our technical skills but also provided valuable insights into designing and building a complete full-stack application. It serves as a demonstration of our ability to apply theoretical concepts to create a functional and meaningful web-based solu

2. CHALLENGES FACED

Integrating Frontend with Backend

Connecting the registration form on the frontend with the backend server required understanding of API routes, HTTP methods, and data flow. Initially, handling form submissions and transferring data correctly posed difficulties.

Learning and Implementing MongoDB

Since MongoDB uses a NoSQL structure, designing collections and understanding schema-less data storage was a challenge. Proper configuration of the database and ensuring successful connectivity from the backend also took time.

Handling Data Validation and Errors

Ensuring that the registration form accepted only correct and complete data required implementing validation checks. Managing server-side errors and debugging unexpected issues was also challenging.

Designing a Responsive UI

Making the website visually appealing and ensuring it worked well on different screen sizes (desktop, tablet, mobile) required additional effort in CSS and layout design.

Deploying the Application (if applicable)

Hosting the website and database together, and ensuring stable performance, was initially confusing due to configuration and compatibility issues.

Version Control and Collaboration

If the project was done as a team, coordinating code changes and resolving merge conflicts in version control systems like Git was a challenge.

Managing Time and Tasks

Balancing frontend design, backend logic, testing, and documentation within the project deadline was

3.TOOLS USED

1. HTML (Hyper Text Markup Language)

HTML was used to create the basic structure of the web pages. It defines the layout of the website, including headings, paragraphs, forms, images, and navigation menus. It serves as the foundation of the entire frontend.

2. CSS (Cascading Style Sheets)

CSS was used to style the website and make it visually appealing. It helped in improving the website's design through colors, fonts, layouts, responsiveness, and animations. CSS ensured the website worked well on different devices.

3. JavaScript

JavaScript was used to add interactivity to the website. It handles client-side validations, dynamic content updates, and smooth user interactions. It enhances the overall user experience.

4. Node.js

Node.js served as the backend runtime environment. It allows JavaScript to run on the server side. Node.js was used to build APIs, process form data, and manage server operations.

5. Express.js

Express.js is a lightweight web framework for Node.js. It was used to create routes, manage requests and responses, and simplify backend development. It helps structure the server logically and efficiently.

6. MongoDB

MongoDB is a NoSQL database used to store registration form data. It provides a flexible document-based structure, making it easy to insert, update, and retrieve student details. It integrates smoothly with Node.js applications.

7. Mongoose

Mongoose is an ODM (Object Data Modeling) library for MongoDB. It simplifies database operations by allowing schema creation and easy interaction with collections. It helps maintain clean and structured database code.

8. Git & GitHub

Git was used for version control, enabling team collaboration and tracking code changes. GitHub was used to store the code repository online and manage project updates effectively.

9. VS Code (Visual Studio Code)

VS Code was the main code editor used in this project. It offers extensions, debugging tools, and a clean interface, making development faster and easier.

10. Postman

Postman was used to test backend APIs. It helped verify if data was correctly being sent to the server and stored in MongoDB.

4.OUTCOMES

The development of the full-stack College Website resulted in several valuable technical and practical outcomes. One of the major outcomes was gaining hands-on experience in building a complete web application from scratch using modern web technologies. Through this project, we learned how to design a clean and responsive user interface using HTML, CSS, and JavaScript, ensuring smooth navigation and readability across different devices.

On the backend, we successfully implemented server-side functionalities using Node.js and Express.js. This enhanced our understanding of API creation, routing, request handling, and integrating the frontend with the backend. Another significant achievement was learning how to store and manage data using MongoDB. By working with a NoSQL database, we gained insights into database connectivity, schema design (if using Mongoose), and CRUD operations, which are essential skills for full-stack development.

The successful integration of the registration form with the database demonstrated our ability to implement real-life data processing and validation. In addition to technical knowledge, the project improved our teamwork, problem-solving, and time-management skills. Overall, the project helped us understand the complete workflow of web development and strengthened our confidence to build more advanced full-stack applications in the future.

5.CODE SNIPPETS

The screenshot shows the Visual Studio Code interface. The left sidebar displays a file tree with a project named 'FSD' containing files like server.js, apiex.js, form.html, and form.css. The main area shows the content of form.html, which is an HTML form for student registration. The terminal at the bottom shows the command line running node server.js and outputting that the server is running at http://localhost:3000.

```
<html>
<body>
<div class="container">
<h2>Student Registration Form</h2>
<form id="regForm">
<label>Name</label>
<input type="text" name="name" required>

<label>USN</label>
<input type="text" name="usn" required>

<label>College</label>
<input type="text" name="college" required>

<label>Course</label>
<input type="text" name="course" required>

<label>Course Specialization</label>
<input type="text" name="specialization" required>

<label>Date of Birth</label>
<input type="date" name="dob" required>
```

```
PS C:\Users\Pragathi\Desktop\FSD\Rform> cd ..\Rform\
PS C:\Users\Pragathi\Desktop\FSD\Rform> node server.js
Server running at http://localhost:3000
MongoDB Connected
```

The screenshot shows the Thunder Client interface. It lists recent activity, including a successful GET request to localhost:3000/getRegistrations just now. The main panel shows a successful response with status 200 OK, size 1.12 KB, and time 14 ms. The response body is a JSON array of two student objects.

```
[{"_id": "6927e835c6dabbd0e447a5d6", "name": "Pragathi A B", "usn": "ENG24CT0015", "college": "DSU", "course": "B.Tech", "specialization": "CST", "dob": "2006-06-04", "email": "abpragathi46@gmail.com", "address": "Sannayasi kodamagge hosuru, bhadravathi (tq), shimogga (d)", "__v": 0}, {"_id": "6927e8abc6dabbd0e447a5d8", "name": "KHUSHI NJ", "usn": "ENG24CT0008", "college": "DSU", "__v": 0}]
```

```
PS C:\Users\Pragathi\Desktop\FSD\Rform> cd ..\Rform\
PS C:\Users\Pragathi\Desktop\FSD\Rform> node server.js
Server running at http://localhost:3000
MongoDB Connected
```

The screenshot shows the Thunder Client interface. The left sidebar lists recent activity, including a successful POST request to `localhost:3000/getRegistrations` just now, and another POST request to `localhost:3000/api/message` 5 days ago. The main panel shows a POST request to `http://localhost:3000/register`. The Body tab is selected, displaying the JSON content sent to the server:

```
1 { "name": "Pranav",  
2 "usn": "1DS23CS001",  
3 "college": "DSU",  
4 "course": "B.Tech",  
5 "specialization": "CS(ds)",  
6 "dob": "2025-10-10",  
7 "email": "praghdi56@gmail.com",  
8 "address": "amb"  
9 }
```

The response status is 200 OK, size is 19 Bytes, and time is 41 ms. The response body contains the message "Registration Saved!".

MongoDB Compass - 127.0.0.1:27017/studentDB.students

Connections Edit View Collection Help

Compass

My Queries

Data Modeling

CONNECTIONS (3)

Search connections

127.0.0.1:27017 > studentDB > students

Documents 5 Aggregations Schema Indexes 1 Validation

Type a query: { field: 'value' } or [Generate query](#)

Explain Reset Find Options

25 1 - 5 of 5

Document 1:

```
_id: ObjectId('6927e835c6dabbd0e447a5d6')
name : "Pragathi A B"
usn : "ENG24CT0015"
college : "DSU"
course : "B.Tech"
specialization : "CST"
dob : "2006-06-04"
email : "abpragathi46@gmail.com"
address : "Sannyasi kodamage hosuru, bhadravathi (tq), shimogga (d)"
__v : 0
```

Document 2:

```
_id: ObjectId('6927e8abc6dabbd0e447a5d8')
name : "KHUSHI NJ"
usn : "ENG24CT0008"
college : "DSU"
course : "B.Tech"
specialization : "CST"
dob : "2006-02-09"
email : "khushinj9@gmail.com"
address : "halavagalu , davangere"
__v : 0
```

6. RESULT

The screenshot shows a web browser window titled "Student Registration" at "localhost:3000". The page contains a registration form with fields for Name, Date of Birth, Email, and Address, followed by a "Register" button. Below the form is a "Get Registrations" button. A section titled "All Registrations" displays a list of student names and their corresponding USN and email addresses:

- Pragathi A B - ENG24CT0015 - abpragathi46@gmail.com
- KHUSHI NJ - ENG24CT0008 - khushinj9@gmail.com
- ANIKASHREYA S H - ENG24CT0029 - anikasangameshi@gmail.com
- ANJU S - ENG24CT0001 - anjugowda08@gmail.com
- PALLAVI B - ENG24CT0051 - reddy pallu30@gmail.com

The screenshot shows a web browser window titled "Student Registration" at "localhost:3000". The page displays a "Student Registration Form" with fields for Name, USN, College, Course, Course Specialization, Date of Birth, Email, and Address, each with its own input field. At the bottom is a "Register" button.

GITHUB LINK: https://abpragathi007.github.io/dsu_new2.0/

7.CONCLUSION

In conclusion, the development of our full-stack College Website has been a highly educational and rewarding experience. This project allowed us to apply the concepts learned in full-stack development and understand how frontend, backend, and database systems work together to create a complete and functional web application. By designing an interactive and user-friendly interface, we gained practical skills in structuring web pages, styling them effectively, and ensuring a responsive layout suitable for different devices.

The backend development strengthened our knowledge of server programming using Node.js and Express.js. We learned how to handle form submissions, validate user inputs, manage routes, and implement server-side logic efficiently. Working with MongoDB further enhanced our understanding of database operations and data management, helping us store and retrieve registration details reliably.

Beyond technical skills, the project helped us improve our teamwork, project planning, and problem-solving abilities. We encountered several challenges during development, but overcoming them taught us patience, persistence, and attention to detail. Overall, this project successfully met its objectives and provided us with the confidence to work on more complex full-stack applications in the future. It has strengthened our foundation in modern web development and prepared us for real-world software development tasks.

