Jay Himanshu Mistry

Full-Stack Developer

✉ jay.mistry2208@gmail.com   📞 +91 9313021049

📍 Dindoli,Surat   📅 22-08-2002

in linkedin.com/in/jay-mistry-jhm22082002   ⊙ github.com/jay-228

✏ DESCRIPTION

To obtain a challenging position as a Full Stack Developer where I can utilize my skills in HTML & CSS, Bootstrap, C language, JavaScript & jQuery, React Js, Node Js, Express Js and API Integration to contribute to the success of the organization and gain valuable experience in the field.

🎓 EDUCATION

2021 – 2024 Surat, India	**Bachelor of Computer Applications** *C.B patel Computer collage, vesu*
04-2020 Surat, India	**Higher Secondary Education Board** *Vidhya Vihar Snkul School*

📚 COURSES

06-2023 – 11-2024 Surat, India	**Red and white multimedia education** *Full-Stack Devloping*

🧠 SKILLS

- HTML
- Bootstrap
- JavaScript
- React js
- MongoDB

- CSS
- c & c++
- jquery
- Node js
- PHP

✏ LANGUAGE

- English
- Gujrati
- Hindi

🏅 AWARDS

HackLoom:Weaving Teech Marvels(2024) 🔗
Runner up in Website Design.

Best Student of The Month(2023) 🔗
An appreciation for excellence in academic performance.

</> PROJECTS

React-Data_table ✐

Develop a web application using React.js that serves as a Product Management System. The system should allow users to perform CRUD operations (Create, Read, Update, Delete) on products. Additionally, users should be able to sort, search, paginate, and filter products. Axios will be used for making HTTP requests, and JSON Server will be used to simulate a backend REST API.

React-Navigator ✐

Build a web application using React.js that showcases a product catalog. Users should be able to navigate through different pages such as Home, About, and Products. The Products page will display a list of products fetched from a JSON Placeholder API. Additionally, there will be a Project page where users can view details of a specific product. The Products page should be private and accessible only to authenticated users. Axios will be used for making network requests.

Node-Login_With_Blog ✐

Develop a Blog Management System with a React.js frontend and a Node.js, Express, MongoDB backend. Implement JWT-based authentication, bcrypt for password encryption, and cookie-based session management. Users can sign up, log in, and manage blog posts with role-based access (admin/user). Admins manage all posts, while users handle their own. Features include CRUD operations, responsive UI, and secure API communication using Mongoose, React Router, and Axios.