

Shubham Kumar

+91 9546366774 | suvmsharma121@gmail.com | linkedin.com/in/shubham-kumar | github.com/suvm-sharma

EDUCATION

Lakshmi Narain College of Technology (LNCT)

Bachelor of Engineering in Electronics and Communication Engineering (ECE)

Bhopal, MP

April. 2016 – May 2020

EXPERIENCE

Blue Infinity Tech LLP

Node.js Developer, Internship

July 2023 – Oct 2023

Ahmedabad, Gujarat

- Developed a robust image upload API using Multer, enabling users to upload photos that are stored in both the database and a dedicated folder. Implemented features for deleting and updating photos with corresponding database and folder management.
- Designed and implemented a Product and Cart API using Mongoose, establishing various relationships such as one-to-one, one-to-many, and many-to-many, enhancing data organization and retrieval.
- Developed a comprehensive restaurant API, gaining insights into the workings of the restaurant industry behind the scenes. Designed access control for staff members, ensuring only valid users can access specific sections within the restaurant.

Impact Infotech Pvt. Ltd.

Node.js Developer

Feb 2023 – Mar 2025

Gurugram, Haryana

EC Portal Backend & Avocado

- Managed and uploaded a categorized Excel product list (food, drinks, etc.) with high-quality images for better visual presentation.
- Modeled complex relational data in Mongo DB/Mongoose.
- Focused on backend logic and validation layers using Node.js and Joi.

Agastya (Medical Document Portal)

- Built a secure portal with 5 user roles to help over 100 doctor's access DICOM medical imaging reports.
- Integrated a DICOM Viewer to allow doctors to view patient imaging reports in standard medical formats.
- Implemented role-based access control (RBAC) via PostgreSQL + Sequelize, blocking 95% + unauthorized requests.
- Automated document review workflows, cutting report processing time by 30%.

Genoroot

- Built a multi-role platform with 5 user roles (Super-Admin, Doctor, Lab-Admin, Clinic-Admin, and PCC), allowing secure access control for over 100 doctors and staff.
- Designed and implemented role-based access control (RBAC) using PostgreSQL and Sequelize, blocking over 95% of unauthorized API calls.
- Used PostgreSQL to store key data like patients, appointments, user roles, and lab orders; and Redis to cache sessions and speed up access control checks.
- Created a document review workflow that automated status updates and approvals, cutting report processing time by about 30%.
- Integrated a web-based DICOM viewer, enabling doctors to seamlessly view and interact with patient imaging reports in standard medical formats.
- Managed image and DICOM file storage via AWS S3, ensuring secure, scalable, and cost-effective asset management.

PROJECTS

Dev Tinder | [Link](#) | JavaScript, Mongo DB

- Created a Mongo DB schema to support dynamic portfolios and user matching.
- Built a Node.js/Express backend with JWT authentication, using Mongoose instance methods and middleware (pre and post hooks).
- Designed a Mongo DB schema for dynamic portfolios with match request management (accept/reject), handling all edge cases.
- Developed a robust API for managing user portfolios, enabling real-time updates and dynamic matching based on portfolio data and preferences.
- Implemented custom Mongoose queries and aggregation pipelines to efficiently retrieve and filter match data, improving user experience and reducing response times.

Natours Application | [Link](#) | *JavaScript, Mongo DB*

- Developed a Restful API (Express.js) with filtering/pagination.
- Implemented advanced querying with filtering, sorting, field limiting, and pagination for tour records.
- Designed Mongo DB schemas with Mongoose virtual for computed properties and population for relational data.
- Secured authentication using JWT tokens and cookie-based sessions, reducing unauthorized access by 100%
- Built error handling middleware for centralized error catching, reducing debugging time by 30%.
- Automated email notifications via Node mailer for bookings and password resets, cutting reset latency.

TECHNICAL SKILLS

Languages: JavaScript, Typescript, C++

Frameworks: Node.js, Express.js, Rest Api, Socket.IO

Developer Tools: Git, Docker

Data: *PostgreSQL*, Mongo DB, Redis