Sejal Amancha

sejal.amancha.5057@gmail.com
+91-79848 44580

5+ years | Full Stack Developer

Summary:

5+ years of experience as a Full Stack Developer with a strong focus on building robust, scalable web applications using modern frontend and backend technologies. On the frontend, I specialize in **React.js with Vite**, with expertise in **JavaScript**, **TypeScript**, modern UI design, API integration, and libraries like **Material UI** and **Ant Design**. I have hands-on experience with Figma and Balsamiq for wireframing and UI/UX workflows. On the backend, I have worked extensively with **C#**, **.NET 8.0**, **ASP.NET Core**, and the **ABP Framework (abp.io)**. I also have deep experience with **MongoDB**, handling schema design, CRUD operations, and performance optimization. I'm proficient in integrating RESTful APIs, managing version control with GitHub, BitBucket, and AWS CodeCommit, and following Agile methodology with tools like JIRA and Trello. I thrive in collaborative environments, bring a proactive mindset, and always aim to deliver clean, well-documented, and high-performing code.

Technical Skills:

Language	JavaScript, Typescript, ES6, HTML, CSS, CSS3, ECMAScript (ES6),
Expert Tech Stack	React.js, Next.js, Gatsby.js, Express.js, Angular.js, Vue.js, .NET 8.0, ASP.NET Core, ABP Framework (abp.io)
UI Tools	Figma, AutoCAD & Balsamiq Wireframes
React JS Tools	Material UI, Saga, Redux, Thunk, Axios, Postman, Hooks, Fetch API
React JS UI Libraries	Material-UI, Ant Design, Bootstrap, Less, SASS, Reactstrap, Paper UI
Project Management Tools	JIRA, Mantis Bug Tracker, Clickup, Trello
Version Control	AWS Codecommit, GitHub, BitBucket, GitLab, SVN, AWS CodeCommit
Amazon Web Services	AWS S3, AWS Lambda, AWS EC2, API Gateway
Database	MySQL and PostgreSQL
SDLC	Agile Methodology (SCRUM)
Web Services	Rest API
IDE	Visual Studio Code
Unit Testing	Jest, Familiarity with ES-Lint, Mocha.

Project Experience:

Project Name	**Mikado**
Description:	I spearheaded the Mikado Project, which focuses on managing pipeline and pressure circuits to optimize fluid flow operations and enhance overall system efficiency. This comprehensive management system integrates front-end technologies such as Angular JS, Material, and Redux with a backend built on Node.js and Express.js. The project aims to streamline fluid management processes, improve operational effectiveness, and provide a robust solution for handling complex pipeline and pressure circuit requirements.
Responsibilities and Role:	In the Mikado Project, I led the development and implementation of both front-end and back-end components. On the front end, I used Angular JS, Next.js for building dynamic user interfaces, Material for design consistency, and Redux for effective state management. On the back end, I worked with Node.js and Express.js to create a robust server-side infrastructure. My role involved coordinating the integration of these

	technologies to ensure a cohesive system that optimized pipeline and pressure circuit management, ultimately improving fluid flow operations and overall system efficiency.
Technical Stack	The Mikado Project utilizes React.js for dynamic front-end development, complemented by Material for design components and Redux for state management. The backend is powered by Node.js and Express.js, providing a scalable and efficient server-side solution. This tech stack ensures a seamless integration between the front-end and back-end, optimizing fluid flow operations and enhancing system efficiency.
Team Size	6
Duration	1 year

Project Name	**OTG OAP Assessment Player**

Description:	The OTG OAP Assessment Player project was a key initiative that demonstrated my technical expertise and problem-solving abilities. I was instrumental in designing, implementing, and optimizing this assessment tool, which was crucial for evaluating and enhancing user skills. The project focused on creating a robust and efficient platform to assess various competencies effectively.
Responsibilities and Role:	I was responsible for designing an intuitive user interface that ensured a seamless and engaging user experience. My role involved rigorous testing to guarantee the accuracy and reliability of the assessment player. I worked closely with cross-functional teams to ensure the tool aligned with organizational goals and met user requirements. Additionally, I utilized emerging technologies and best practices to continuously improve and update the assessment player, contributing significantly to its success and effectiveness.
Technical Stack	The OTG OAP Assessment Player project employed a variety of technologies to build a comprehensive and effective assessment tool. The front end was developed using Vue.js, Gatsby.js modern frameworks and libraries to create an intuitive and responsive user interface. On the back end, robust technologies were used to handle data processing and ensure seamless functionality. The stack also included advanced tools for rigorous testing and optimization, along with emerging technologies and best practices to enhance the tool's performance and reliability.
Team Size	5
Duration	7 months

Project	**Easy On Services**

Description	Easy On Services is a comprehensive service-based application designed to facilitate customer access to various repair and installation services, including washing machine repair, computer repair, AC repair, and CCTV camera installation. The platform provides users with the flexibility to book services online or visit a physical shop, tailored to their current location. This ensures a convenient and efficient service experience, addressing a wide range of household and technical needs through a user-friendly interface and streamlined booking process.
Responsibilities and Role	As a MEAN Developer for this project, I was responsible for implementing and maintaining the application's front-end and back-end components. My role included developing user interfaces with Angular JS and Chakra UI, managing state with Redux, and ensuring reliable API interactions using Axios. I also handled database management with MySQL, integrated testing frameworks like Jest, and used Postman for API testing. I tracked project progress and managed tasks using JIRA.
Team Size	5
Technical Stack	The project utilized Angular JS for front-end development, Nuxt3, JavaScript, HTML, CSS, Tailwind, CSS Chakra UI Axios for user interface components, and Redux for state management. The back end was supported by MySQL for database operations, with Hooks and Axios managing data fetching and state. JavaScript was used throughout the application, with Jest for unit testing and Postman for API testing. JIRA was employed for project
Duration	1 year

Project	**DNAS**
Description	The Daily News & Analysis System (DNAS) is a comprehensive web application designed to deliver the latest updates in finance, government jobs, banking, railway news, IPOs, and COVID-19 developments. The platform provides users with an enriched news experience, featuring detailed editorial coverage and timely updates. DNAS integrates various sources of information to offer a consolidated view of important news and market insights, ensuring users stay informed with relevant and in-depth content.

Responsibilities	As the Angular JS Developer for the DNAS project, I was responsible for designing and implementing the front-end components of the web application. My role included developing a user-friendly and responsive interface using Angular JS, ensuring smooth integration with backend services via Axios. I handled real-time data updates through Firebase and conducted thorough API testing with Postman to ensure data accuracy and reliability. Additionally, I used Trello for managing tasks and tracking project progress, ensuring timely delivery and coordination with the development team.
Team Size	7
Technical Stack	The DNAS project utilizes Angular JS for building a dynamic and responsive front-end interface. Firebase is employed for real-time data management and authentication, while Axios handles API requests for fetching news and updates. Postman is used for API testing and validation, and Trello is utilized for project management and task tracking. This stack ensures a robust and efficient development environment, supporting a seamless news and analysis experience.
Duration	8 months

Project Name	**IPG – Invoice Generator**
Description:	Worked on back-end side to implement employee's management portal. Implemented cron jobs to generate employee's salary on monthly basis. Where user can register as company and create employees with admin

	access. Admin user can create and manage user profiles as company and employee. user can manage/create invoices and payslips for particular user and stock. Admin user can generate the payslip from admin panel and sent as attachment mail to particular user

Responsibilities and Role:	In the IPG – Invoice Generator project, I was responsible for both front-end and back-end development. On the front end, I utilized React and Redux to build a user-friendly interface for managing employee profiles, generating invoices, and handling administrative tasks. On the back end, I worked with Node.js and Express.js to implement server-side logic, including the creation and management of employee data and invoice processing. I also set up **cron jobs** to automate the generation of monthly salaries and integrated MySQL for database management. My role involved ensuring seamless functionality across both front-end and back-end components, providing a comprehensive solution for employee management and financial documentation.
Technical Stack:	The IPG – Invoice Generator project utilizes React and Redux, ECMAScript (ES6), for front-end development, providing a dynamic and responsive user interface. The back-end is built using Node.js and Express.js for handling server-side logic and API interactions. MySQL is used for database management, storing employee and company data, as well as invoice and payslip information. Cron jobs are implemented to automate the monthly generation of employee salaries. This stack ensures a scalable and efficient system for managing employee profiles and financial documentation.
Team Size	6
Duration	1 year

Project Name	**Employing Portal**
Description:	The Employing Portal is a comprehensive job management system designed to facilitate job postings and applications. Admin users can post job listings with specific locations and manage these listings, including overseeing online payments. Job seekers can browse available job opportunities, apply

	for positions, manage their profiles, create and update resumes, and send messages to potential employers. The portal streamlines the job application process and enhances communication between job seekers and employers, providing an efficient platform for job management and application.

Responsibilities and Role:	In the Employing Portal project, my primary role was front-end development. I utilized React to build a responsive and interactive user interface, incorporating Material-UI for consistent design and visual appeal. Redux was employed to manage the application state, ensuring efficient data handling and seamless user interactions. My responsibilities included developing features for job postings, application management, user profile management, and resume creation. I focused on creating a user-friendly experience for job seekers and administrators, integrating functionalities to streamline job applications and communications within the portal.
Technical Stack:	The Employing Portal utilizes React ,Vue js for building a dynamic and responsive front-end user interface. Material-UI is employed for a consistent and modern design system. Redux is used for state management, ensuring smooth and efficient data handling across the application. This stack supports a robust and user-friendly experience for both job seekers and administrators, facilitating job postings, applications, and profile management.
Team Size	11
Duration	6 months

Project Name	**AMS – Asset Management System.**
Description:	The Asset Management System (AMS) is a real-time vulnerability scanning platform designed to enhance site security by identifying potential vulnerabilities. Users can scan their sites using integrated scanners and receive detailed reports on security issues. The system displays a comprehensive overview of available assets, including allocation details for each user. As the team lead and primary developer, I was responsible for integrating the Accunetix scanner, developing middleware, managing scan frequencies for different products, and visualizing data on dashboards. The

	project aims to provide users with actionable insights to safeguard their assets and improve overall security posture.

Responsibilities and Role:	As the team lead and primary developer for the AMS project, I was responsible for overseeing the development and integration of key features. I led the front-end development using **React** and **Material-UI**, creating an intuitive and visually appealing user interface. **Redux** was utilized to manage application state, ensuring smooth and efficient data handling. I integrated the **Accunetix** scanner into the system, developed middleware for seamless communication, and managed scan frequencies for various products. Additionally, I designed and implemented dashboards to present vulnerability reports and asset allocation data, providing users with actionable insights and enhancing overall security management
Technical Stack:	The AMS project employs Reac**t** for creating a dynamic and interactive front-end user interface. Material-UI is used for consistent and modern design elements, while Redux manages the application state to ensure efficient data handling and smooth user interactions. The back end integrates with the Accunetix scanner for vulnerability scanning, and middleware is used to handle communication between the scanner and the application. Data is displayed on dashboards to provide users with actionable insights on asset allocation and security vulnerabilities.
Team Size	10
Duration	5 months