

# Abinash Mohanty

(+91)9711242241 | [abinashkummar33@gmail.com](mailto:abinashkummar33@gmail.com) | [LinkedIn](#) | [GitHub](#) | [Portfolio](#) | Delhi, India.

## SUMMARY

Software Engineer with 1.5+ years of experience building scalable frontend applications and API automation frameworks in a product-driven environment. Experienced in Angular-based UI architecture, REST API integration, and performance optimization, with working knowledge of backend systems.

## TECHNICAL SKILLS

**Languages:** Java, TypeScript, SQL

**Frontend:** Angular (v17+), RxJS, Angular Material, SCSS, HTML5

**Backend:** Spring Boot (REST APIs), Node.js (Basics)

**Testing:** Karate, JUnit, Jasmine, Karma

**DevOps & Tools:** Docker, Jenkins, Maven, GitHub, Postman, Jira

**Core Concepts:** RESTful APIs, System Design Basics, OOPS, Debugging, Agile (Scrum)

## WORK EXPERIENCE

### Fidelity Investments | Chennai

Sept 2024 - Sept 2025

**Problem Statement:** To design and implement a custom reporting engine while refactoring the existing product's legacy codebase.

**Software Engineer – UI Development**

- Engineered Angular(v17) UI components with shared API services, enabling real-time reporting features and achieving a **50%** improvement in component responsiveness; completed project three weeks ahead of schedule.
- Increased code reusability and reduced onboarding time by **50%** by refactoring the monolithic component logic into a **modular parent-child architecture**.
- Reduced selection errors by **40%** by implementing default template logic with reactive forms and conditional rendering.
- Highlighted project impact measured by presenting the **\$4M** cost-saving migration plan to VP and stakeholders.
- Refined dropdown load performance by **98%**, reducing time from **230s** to under **5s** by optimizing CUSIP **fetch size** and **memory handling**, enabling smooth rendering of over 15,000 entries without leaks.

**Test Automation Engineer**

- Lowered manual configuration effort by **40%** and accelerated execution cycles by **30%** by implementing automated environment selection in **Karate** and **Bruno** for SIT, UAT, and DEV.
- Improved test coverage by **35%** and lowered post-release defects by **20%** through design and execution of parameterized API test scenarios.
- Increased data validation accuracy by **45%** and minimized external dependencies by integrating direct database connectivity to execute stored procedures within test scripts.
- Ensured system stability and defect detection rate by **25%** by implementing structured negative test cases in automated API validations.

### Adyogi | Delhi

**Frontend Developer Intern**

Aug 2023 - Nov 2023

- Improved web application stability, reducing system downtimes by **15%** through effective maintenance.
- Increased mobile traffic by **30%** and decreased bounce rates by **20%** by utilizing responsive design principles.
- Conducted routine code reviews to enforce coding standards and optimize best practices.

## PROJECTS

### WorkForce Analytics Dashboard : Angular 19 | RxJS | Chart.js | PapaParse

[GitHub](#) | [Live Demo](#)

- Built a client-side analytics dashboard to process and visualize workforce datasets (2000–2025) using CSV ingestion.
- Implemented dynamic KPI calculations including YoY growth, company ranking, and trend aggregation.
- Optimized large dataset parsing using PapaParse with efficient transformation pipelines in TypeScript.

### Soil-FDD Reporting application(CRUD) : Angular 19 | Angular Material | Chart.js

[Github](#) | [Live Demo](#)

- Developed a structured CRUD application to manage soil density test records.
- Implemented validation logic for engineering threshold comparisons.
- Designed reporting UI to visualize compaction metrics against required standards.

### Quote Flow – RFQ Management System(Backend) : Spring Boot | REST APIs | JWT | JPA

[GitHub](#) | [Case Study](#)

- Built Spring Boot REST APIs for RFQ creation, modification, and approval workflows with request validation and error handling.
- Developed backend logic for pricing checks, approval conditions, and RFQ state transitions.
- Implemented role-based access control (RBAC) to restrict RFQ actions (create, approve, reject) based on user roles.

## ACHIEVEMENTS

- Smart India Hackathon (2023) pre-qualifiers.
- CodeChef starters division 4 contest **global rank 360**.
- Achieved **97% test coverage across API and UI endpoints** as part of the QA team