

# Juhi Bhandari

Mail: [juhibhandari774@gmail.com](mailto:juhibhandari774@gmail.com) | LinkedIn: [LinkedIn](#) | GitHub: [GitHub](#)

## Education

### Diploma in Computer Programming

Seneca Polytechnic

Jan 2024 – Apr 2025

### Bachelor of Science

University of Mumbai

August 2021 – December 2023

## Skills

**Languages:** Java, Python, C++, JavaScript, SQL, HTML, CSS, ES6

**Tools:** Angular, React, MongoDB, PostgreSQL, Oracle SQL, Jira, GitHub, Figma, Docker, Power BI, Tableau, FTM

**Concepts:** RESTful APIs, Agile methods (Scrum), Jira, Unit and Integration Testing

## Experience

### Software Engineer | ScotiaBank

December 2025 - Present

#### Cash Management Technology Team | Toronto, ON

- Traced message flows and analyzed payload mappings to identify transaction-processing errors, then updated ES6 transformations in integration flows to correct data handling.
- Diagnosed environment-specific workflow inconsistencies between local and UAT systems by validating transactional data using SQL queries.
- Updated workflow definitions and state-machine diagrams to resolve incorrect transaction-state transitions in payment workflows.
- Modified existing backend integration flows to support changes in transaction-processing requirements within the Cash Management platform.

## Projects

### Devflow ([GitHub](#))

*React / Vite / Node.js / Express / MongoDB / JWT*

- Built a full-stack journaling application using React, Express, and MongoDB to store and manage user entries.
- Implemented JWT authentication and bcrypt password hashing for secure user login and protected routes.
- Designed RESTful APIs and MongoDB/Mongoose data models for users and daily entries with upsert-based persistence.
- Structured the project into a separate client-server architecture, improving code organization and scalability.

### Web E-Commerce Application ([GitHub](#))

*Node.js / Express / JavaScript / HTML / CSS / Cloudinary*

- Built a REST API using Express to serve product and category data from structured JSON datasets, simulating a lightweight store backend.
- Integrated Cloudinary image hosting with Multer middleware, enabling multipart uploads and server-side validation for product images.
- Implemented a service-layer architecture (store-service.js) to abstract data access, improving modularity and maintainability.
- Developed filtering endpoints for category and publish-state queries, enabling dynamic store data retrieval.

### Superstore Data Visualization ([GitHub](#))

*Python / Pandas / Matplotlib / Excel / Power BI / Tableau*

- Analyzed retail sales data using Pandas to clean, aggregate, and transform transactional datasets for reporting.
- Built Python scripts to generate sales-trend, profit-by-region, and top-product visualizations using Matplotlib, producing reusable reporting outputs.
- Created Power BI and Excel pivot-table dashboards to explore KPIs and validate analysis results across regions and categories.
- Automated data-loading and chart-generation workflows, enabling repeatable analysis without manual spreadsheet processing.

### To-Do List Application (MERN) ([GitHub](#))

*MongoDB / Express / React / Node.js*

- Implemented CRUD endpoints in Express connected to MongoDB, enabling persistent storage of to-do items.
- Connected React frontend components to backend APIs using fetch/axios, supporting dynamic task creation and updates.
- Refined project setup and environment configuration, reducing onboarding friction for collaborators