# Mohamed Zainudeen V A

## Experience Summary

- 0.9 years of experience in backend and full-stack development.
- Basic knowledge of Machine Learning including classification, regression, and clustering.
- Solved 200+ LeetCode problems with strong DSA skills (Graphs, Trees, Linked Lists, Binary Search).
- Built and trained ML models using PyTorch for regression and classification tasks.
- Skilled in Java, Spring Boot, Angular, and Microservices with solid design principles.

## Skills Summary

| | |
|---|---|
| Domain | Java Full Stack |
| Programming Languages | Java,Python,SQL,HTML/BootStrap/CSS,Javascript |
| Operating System / ERP Version | Windows 11 |
| Tools / DB / Packages / Framework / ERP Components | Pytorch,SpringBoot,MicroServices,Angular,Github,Hibernate, |

## Professional Certifications/ Trainings

1. Ignite Certification in the domain of Java Developer.
2. Orchard Training in the domain of Java Full Stack Developer.
3. Completed Oracle Certified Java SE 11 Developer (1Z0-819) Certification.

# Work Experience

## Project 1

| | | | |
|---|---|---|---|
| **Project Name** | MNIST Digit Classification | Team Size | 1 |
| **Start Date** | Self-Initiated | End Date | Ongoing |
| **Project Description** | Built a digit recognition system using logistic regression on the MNIST dataset to classify handwritten digits with high accuracy. | | |
| **Role & Contribution** | • **Implemented data loading**, preprocessing, and visualization using PyTorch and Matplotlib<br>• **Built and trained a logistic regression** model using a custom neural network class<br>• **Applied softmax activation** and cross-entropy loss for multi-class classification<br>• **Evaluated model performance** using accuracy metrics and validation datasets | | |
| **Technology & Tools** | Python, PyTorch, MNIST, Matplotlib, Google Colab, Jupyter Notebook | | |

## Project 2

| | | | |
|---|---|---|---|
| **Project Name** | Linear Regression with PyTorch | Team Size | 1 |
| **Start Date** | Self-Initiated | End Date | Ongoing |
| **Project Description** | Developed a regression model using PyTorch to predict numerical outcomes from structured input data, simulating real-world scenarios like crop yield estimation. | | |
| **Role & Contribution** | • **Built a custom linear regression** model using PyTorch's nn.Module<br>• **Implemented training loop** with gradient descent and manual parameter updates<br>• **Used Mean Squared Error (MSE)** as the loss function for optimization<br>• **Handled data batching** using DataLoader for efficient training<br>• **Visualized and validated** predictions against target values | | |
| **Technology & Tools** | Python, PyTorch, NumPy, Google Colab, Jupyter Notebook | | |

## Project 3

| | | | |
|---|---|---|---|
| **Project Name** | Online Examination System | Team Size | 5 |
| **Start Date** | Dec 2024 | End Date | Jan 2025 |
| **Project Description** | Online Exam System with secure authentication, question handling, answer submission, and scoring via scalable REST APIs. | | |
| **Role & Contribution** | • **Built Online Exam System** using Java, Spring Boot & Angular.<br>• **Secured Access** with JWT authentication.<br>• **Enabled Exam Scheduling** with time-bound access.<br>• **Handled Answer Submission & Scoring** via JSON input.<br>• **Tracked Exam History** with scores and pass/fail status.<br>• **Optimized Performance** using pagination for questions.<br>• **Implemented Error Handling** for invalid or unauthorized actions. | | |
| **Technology & Tools** | Java, Spring Boot, Angular, MySQL, JWT, Maven, Postman | | |

## Project 4

| | | | |
|---|---|---|---|
| **Project Name** | Forest Fire Detection using Random Forest Algorithm | Team Size | 4 |
| **Start Date** | May 2023 | End Date | Jun 2023 |
| **Project Description** | A system to detect forest fires from images using machine learning. Trained on labeled image data (Fire/No Fire) and used Random Forest for classification. | | |
| **Role & Contribution** | • **Collected and preprocessed image** dataset (resizing, normalization, feature extraction)<br>• **Extracted features** using color histograms, texture analysis (GLCM), and edge detection<br>• **Trained Random Forest** model for fire classification using majority voting<br>• **Implemented prediction pipeline** for new images with fire/no fire output<br>• **Generated safety reports** with emergency contacts and evacuation tips upon fire detection | | |
| **Technology & Tools** | Python, Scikit-learn, OpenCV, NumPy, Pandas, Matplotlib | | |

## Project 5

| | | | |
|---|---|---|---|
| **Project Name** | Call Taxi Booking System | Team Size | 1 |
| **Start Date** | Self-Initiated | End Date | Ongoing |
| **Project Description** | A console-based taxi booking system simulating real-time ride allocation using location mapping, cost calculation, and admin profit tracking. | | |
| **Role & Contribution** | • **Designed & implemented** a modular system in Java for booking, ride tracking, and earnings management<br>• **Developed ride allocation logic** based on shortest distance and taxi availability<br>• **Integrated fare calculation** with admin profit split and driver earnings<br>• **Enabled user roles**: Customer (booking), Driver (ride history), Admin (profit reports)<br>• **Handled data structures** for location mapping, ride logs, and user authentication<br>• **Implemented exception handling** and input validation for robust user interaction | | |
| **Technology & Tools** | Java, OOP, Collections Framework, HashMaps, Scanner I/O | | |

## Project 6

| | | | |
|---|---|---|---|
| **Project Name** | Railway Reservation System | Team Size | 1 |
| **Start Date** | Self-Initiated | End Date | Ongoing |
| **Project Description** | A console-based railway ticket booking system simulating real-world scenarios like berth preferences, senior citizen priority, RAC, and waiting list management. | | |
| **Role & Contribution** | • **Developed a dynamic ticket booking system** in Java with real-time berth allocation<br>• **Implemented priority logic** for senior citizens and minors<br>• **Handled berth preferences** (Lower, Middle, Upper, Side Lower, Side Upper) with fallback options<br>• **Managed RAC and Waiting List** queues when preferred berths are unavailable<br>• **Enabled ticket cancellation** with automatic reallocation from RAC and WL<br>• **Built reporting features** to print current bookings and availability | | |
| **Technology & Tools** | Java, OOP, Collections Framework, Queue & List Structures, Scanner I/O | | |

# Educational Qualification

| | |
|---|---|
| **Education & Credentials** | Bachelor of Engineering and Computer Science |

LTIMindtree

# Let’s get to the future, faster. Together.