

Timothy Medewase

medewaset@gmail.com | linkedin.com/in/timothy-medewase | github.com/TimothyMedewase | timothy-medewase.com

EDUCATION

University of Montevallo

Bachelor of Science in Computer Science, Minor in Mathematics

Montevallo, AL

Aug. 2021 – May 2025

TECHNICAL SKILLS

Languages: Java, TypeScript/JavaScript, SQL, Python, C++, HTML/CSS

Frameworks & Libraries: Spring Boot, Maven/Gradle, React, Redux, Node.js, Next.js, Express.js, Vite, Jest, d3.js, recharts, Django, Flask, FastAPI, Pandas, NumPy, Celery, TensorFlow, PyTorch, Scikit-learn, Material-UI, TailwindCSS

Tools, Databases & Deployment: Git, GitHub, Bitbucket, LaTeX, Prisma, SQLAlchemy, SQLite, PostgreSQL, MySQL, Microsoft SQL Server, MongoDB, REST APIs, GraphQL, Docker, AWS, Heroku, Vercel

EXPERIENCE

Software Engineer (AI/ML)

May 2025 - Present

Figbox Technologies, Inc

Remote, US

- Designed **Django**-based machine-learning pipelines running in **Docker**, serving about **20,000+** users with **sub-200 ms** inference latency
- Engineered a **scikit-learn** recommendation engine backed by **PostgreSQL** and **REST APIs** that increased engagement with suggested connections by **20%**
- Developed natural-language search using **spaCy** and **OpenAI** within a **React**, **TypeScript**, and **Node** stack, returning relevant profiles in **under one second** for **95%** of queries

Software Development Intern

May 2023 – Jul. 2023

Solera, Inc

Westlake, TX

- Created modular **React** and **d3.js** dashboards that increased user engagement by **30%** and reduced feature development time by **40%**
- Optimized **Microsoft SQL Server** stored procedures and containerized services in **Docker**, reducing data-retrieval latency by **50%**
- Automated **REST API** integrations and **Bitbucket CI/CD** pipelines, reducing integration bugs by **40%** and improving deployment frequency by **2x**

PROJECTS

SCOUR | Swift, RevenueCat, Supabase (Auth/Postgres/RLS), FastAPI | Link

Oct. 2025

- Shipped an **iOS “Tinder for recruiting”** app where high-school, transfer, and international athletes swipe through more than **15,000+** college programs, filtering by sport, division, and location with results **under 300 ms**
- Implemented an AI-assisted outreach and CRM backend with **FastAPI**, **Supabase**, and **RevenueCat** to manage paid tiers, track coach contacts, and generate compliant emails, syncing data for hundreds of athlete and coach records each month while handling about **500 API** requests per day with **under 300 ms** average latency

MNIST Digit Classification CNN | Python, PyTorch, Computer Vision | Link

Aug. 2025

- Trained a **PyTorch** Convolutional Neural Network achieving **99.41%** accuracy on the **MNIST** handwritten digit classification task
- Designed and trained a **3.3 million** parameter model with normalization and early stopping, reaching peak validation accuracy in **14 epochs** and visualizing learned filters on test images

ExplainAI | Next.js, React, Python, FastAPI, FAISS, OpenAI, Heroku | Link

Apr. 2025

- Built a **retrieval-augmented generation** system that improved document-answer quality by **40%** over baseline search
- Implemented secure multi-file uploads and **FAISS**-based semantic search, reducing query latency by **60%**
- Deployed modular **React** interfaces and **Heroku**-hosted APIs that handled **50%** more traffic without performance degradation

Movie Management System | Java, Spring Boot, Spring JDBC | Link

Nov. 2024

- Designed a layered backend architecture using **Java** and **Spring** to support scalable CRUD functionality for thousands of movie records
- Applied **Spring Boot** auto-configuration and **Spring JDBC** to reduce setup time by **40%** and eliminate boilerplate query logic
- Improved processing performance by **20%** through optimized in-memory data structures

SPAZ (Spotify Music Analyzer) | Next.js, React, TypeScript, TailwindCSS, Spotify | Link

Feb. 2024

- Launched a scalable analytics system generating personalized **Spotify** insights for users
- Integrated the **Spotify API** using **OAuth 2.0** to aggregate thousands of listening events per day