Mustahid Ahmed

last updated: July 30, 2025

Online Version

Residence
Tokyo, Japan

Nationality
Bangladesh

Git Profile
https://github.com/ahmedmustahid

Email
amustahid25@gmail.com

Website
ahmedmustahid.netlify.app

Python	++++	C++20	++++	C	++++	Dart/Flutter	++++	Swift	+++	Javscript	+++
AWS	++++	Statistics	++++	Image Processing	++++	NLP	++++	Algorithms	++++	Data Structure	++++

Professional Experiece

IBM System Engineering

AI Engineer

March 2025 - Present

Pioneer Electronics

Machine Learning Engineer

July 2023 - Feb 2025

Sensyn Robotics

Machine Learning Engineer

May 2022 - June 2023

HyperCube Ltd.

Machine Learning Engineer

April 2020 - April 2022

Batton Ltd.

Machine Learning Engineer (Freelance)

Aug 2023 - Dec 2023

AI Idea Lab Ltd.

Machine Learning Engineer (Freelance)

April 2023 - Dec 2023

Open Source Contributor

Software Engineer

April 2020 - Present

Projects

Financial AI Multi-Agent System,IBM System Engineering

April 2025 - Present

Developed a sophisticated multi-agent AI system for financial data analysis and document processing using LangGraph orchestration framework. Built a scalable FastAPI backend integrating specialized AI agents for SQL query execution, PDF document retrieval, data visualization, and intelligent request routing. Implemented agentic capabilities with multilingual support (English/Japanese) for financial document analysis.

Key Contributions:

- Architected state-based agent workflow system using LangGraph for coordinated AI agent interactions
- Built RESTful API endpoints with session management and CORS support for web application integration
- Implemented automated financial data visualization and chart generation using Matplotlib
- Developed PDF processing pipeline for extracting and analyzing financial documents
- Set up comprehensive development environment with code quality tools (Black, Ruff, isort, Basedpyright)

`Python` `FastAPI` `LangChain` `LangGraph` `OpenAI API` `PostgreSQL` `Pandas` `Matplotlib` `Docker`

PostgreSQL MCP Server: Dual Transport & Database Integration,Open Source Contributor

January 2025

Developed a production-ready PostgreSQL MCP (Model Context Protocol) server with dual transport support (HTTP and Stdio).

Key Contributions

- Architected a **dual-transport system** (HTTP & Stdio) and designed a **RESTful API** using **FastAPI**, complete with automatic **OpenAPI documentation**.
- Engineered a robust **PostgreSQL integration** featuring **connection pooling**, stateful **session management**, and secure, **parameterized queries** to prevent SQL injection.
- Developed a flexible **configuration system** using **environment variables** and **command-line arguments** to enhance deployment adaptability.
- Streamlined deployment by creating production-ready **Docker containers** and **Docker Compose** configurations.
- Implemented comprehensive **error handling**, graceful **shutdown logic**, and a full **testing suite** to ensure system reliability and stability.

`Python` `PostgreSQL` `MCP Protocol` `HTTP` `Stdio` `Docker` `FastAPI`

View on GitHub

xiyan_mcp_server: Azure OpenAI, llama-cpp-python, & SQLite Integration, Open Source Contributor

January 2025

Contributed to xiyan mcp server project by implementing comprehensive Azure OpenAI integration and enhancing local model support through llama-cpp-python. Enabled Streamable HTTP transport protocol and integrated SQLite database support.

Key contributions:

- Implemented Azure OpenAI SDK integration with configurable model endpoints and API versions
- Added Streamable HTTP transport support with configurable host/port parameters
- Enabled llama-cpp-python local model support for running models locally
- Integrated SQLite database backend with configurable dialect and database path
- Enhanced configuration system with comprehensive YAML-based setup
- Added comprehensive error handling and logging capabilities
- Created detailed documentation and usage examples for all new features

Python | Azure OpenAI | llama-cpp-python | SQLite | Streamable HTTP | MCP Protocol

View on GitHub

Implemented mcp server for static error correction with basedpyright, Open Source Contributor

May 2025 - Present

Contributed to quack-mcp-server project by adding BasedPyright static analysis support as an MCP tool alongside existing mypy integration. BasedPyright provides faster type checking with enhanced features compared to standard pyright.

Key contributions:

- Implemented automatic installation and configuration detection for basedpyright
- Added severity filtering and top-N limiting capabilities for diagnostic output
- Created comprehensive test suite with full coverage for edge cases and error handling
- Integrated with existing diagnostic filtering utilities for consistent output format

Python | Streamable HTTP | MCP Protocol

View on GitHub

Implemented RAG pipeline with LLM for Qualcomm SoC, Pioneer Electronics

April 2024 - August 2024

Designed and implemented end to end RAG pipeline from scratch using **faiss** and **llama.cpp**.

- Created separate library for RAG after deriving llama and embeddings from llama.cpp
- Implemented build system using **Conan**
- Set up CI/CD using **GitHub Actions**

C++ | Conan | git | faiss | llama.cpp

Designed end to end middleware system for automotive radar, Pioneer Electronics

August 2023 - March 2024

Implemented API that obtains sensor data streams and transfers them to machine learning module to deliver inference results back into an alarm system.

- Implemented **lock-free ring buffer system** that stopped data loss and enabled extremely fast inference
- Employed comprehensive unit tests and CI/CD pipeline
- Delivered real-time inference results to alarm system

C++ CMake git CI/CD

Crack segmentation from tiles, Sensyn Robotics

Feb 2023 - April 2023

Implemented model for crack segmentation using novel **transformer-based segmentation architecture** inspired from retinal vessel segmentation in medical literature.

- Implemented **continual learning system** based on labelling→training cycle
- Achieved **high precision of 0.9** and **recall of 0.8**
- Applied cutting-edge medical imaging techniques to civil engineering problems

Python Pytorch OpenCV mmcv

Analog gauge reading by computer vision, Sensyn Robotics

November 2022 - April 2023

Constructed model for accurately reading analog gauges by determining dial position, minimum and maximum values from the scale face after applying perspective transformation.

- Minimized expected error rate to **0.15%**
- Compressed model size through **quantization in C++ using TVM**
- Successfully deployed to edge device (**NVIDIA Jetson Nano**)
- Applied advanced computer vision techniques for precise measurement

Python Pytorch OpenCV onnx mmcv Apache TVM ncnn

People detection from fish eye images, Sensyn Robotics

May 2022 - July 2022

Developed new algorithm for people detection in fisheye images, as commonly used algorithms are not suitable for such distorted perspectives.

- Incorporated **angular values of slanted bounding boxes** for improved detection accuracy
- Implemented end-to-end pipeline from **labelling to cloud storage** using **CoCo Annotator** and **AWS S3**
- Achieved **high mAP score** comparable to state-of-the-art YOLO algorithms
- Solved unique challenges posed by fisheye lens distortion

Python Pytorch OpenCV AWS

NLP based Chatbot, HyperCube Ltd.

June 2021 - October 2021

Full stack Development: Frontend and backend development using Flutter/Dart/Swift. Model deployment, system design and maintenance using AWS services:

- AWS Amplify for frontend hosting
- AWS API Gateway for API management

- AWS Lambda for serverless functions
- AWS SageMaker for ML model deployment
- Frontend and backend development using Flutter/Dart/Swift

AWS | Dart/flutter | Swift | Docker | Git

View on GitHub

Database design and deployment, HyperCube Ltd.

March 2021 – May 2021

Designed MySQL database schema and deployed it in Azure.

- Created comprehensive database schema design
- Deployed database on Azure cloud platform
- Developed web application API for database connectivity
- Implemented secure data access patterns

MySQL | Azure | Python | Git

Designed novel algorithm for OCR of Japanese fax documents,Batton Ltd.

Aug 2023 - Dec 2023

Collected and labelled dataset of fax documents containing tables. Implemented novel algorithm to extract its data and obtained 97% accuracy.

- Collected and labeled dataset of fax documents with tables
- Developed novel OCR algorithm for table data extraction
- Achieved 97% accuracy on test dataset
- Dockerized the ML model for deployment
- Created HTTP endpoint using TorchServe
- Deployed to GCP Artifact Registry
- Created inference endpoint through Vertex AI

Python | Pytorch | torchserve | OCR | Docker | GCP

Japanese kanji generation from a specific font,AI Idea Lab Ltd.

April 2023 - Sep 2023

Used controlnet and prompt engineering to create Kanjis from novel fonts.

- Applied ControlNet for Japanese kanji generation
- Used prompt engineering for font-specific character creation
- Built interactive frontend using Gradio
- Developed FastAPI backend endpoints
- Integrated Gradio frontend with FastAPI backend

Python | Image Generation | FastApi | Gradio

GPT based AI Callerbot,AI Idea Lab Ltd.

Oct 2023 - Dec 2023

Converting caller's speech into text and using prompt engineering to obtain the best output from Open AI API.

- Convert caller's speech to text using speech recognition
- Apply prompt engineering for optimal OpenAI API responses

- Convert API responses back to voice using text-to-speech
- Create WebRTC bridge between Python and Node.js
- System currently being deployed in company application

Python Node.js WebRTC

Memorize: A game for training memory,Open Source Contributor

June 2021 - Sep 2021

Designed memory training game with engaging user interface.

- Designed frontend using Swift programming language
- Implemented game logic for memory training mechanics
- Created intuitive user experience for cognitive exercises

Swift Git CI/CD

View on GitHub

amusta-chain: A distributed peer to peer blockchain,Open Source Contributor

June 2021 - Sep 2021

Developed distributed peer-to-peer blockchain system with comprehensive testing.

- Employed end-to-end Test-Driven Development (TDD)
- Implemented peer-to-peer mining functionality
- Created secure currency transfer system to user wallets
- Developed transaction validation by miners
- Built distributed consensus mechanism

javascript Express Node.js Git CI/CD

View on GitHub

Education

Tohoku University
MSc, Particle Physics [2018 - 2020]

Tohoku University
BSc, Physics [2014 - 2018]

Tokyo University of Foreign Studies
Associate Degree, Japanese [2013 - 2014]

Research

Masters Thesis□Machine Learning, Deep Learning, Data Analysis

Simulation of particle collisions and their reaction in particle detectors using C++.

- Simulated particle collisions and detector reactions using C++
- Designed particle detectors using C++ framework
- Analyzed detector data using machine learning and deep learning algorithms
- Applied advanced statistical methods for particle physics research

Python OpenCV PyTorch C++ Git CMake

Conference Presentation: Japan Physical Society Annual Meeting

```
  Full Detector Simulation of Pair Monitor and Application of Machine Learning to Determine
Determine Beam Size
  Nagoya University, Nagoya, Japan [March ’20]
```

Conference Presentation International Workshop on Future Linear Colliders

```
  Search for weakly interacting dark matter in the International Linear Collider
  University of Texas, Arlington, Texas, USA. [Oct ’18]
```

Languages

```
  English, Business Level, TOEFL iBT 110
  Japanese: Business Level
  Bengali: Native Level
```

MOOCs

- Deep Learning Specialization : Coursera, Stanford Online
- Algorithm Specialization : Coursera, Stanford Online
- GAN Specialization : Coursera, Stanford Online
- NLP Specialization : Coursera, Stanford Online

Categories: Python C++ Algorithms Deep Learning