THESIYA JEEL RAMESHBHAI

M.Tech – Machine Learning, DA-IICT

Tel +91 94295 59098 @ jeelthesiya01@gmail.com @ 202511065@dau.ac.in

in linkedin.com/in/jeel-thesiya-8a2331369 **GitHub** github.com/jeel00dev

Dhirubhai Ambani University
Technology
Formerly DA-IICT

EDUCATION

Dhirubhai Ambani University (DA-IICT) 2025 – Present
M.Tech – Machine Learning ***CPI**: 7.67 (1st Year)* *Gandhinagar, Gujarat*

MBIT College, CVM University 2021 – 2025
B.E. – Computer Engineering ***CPI**: 7.7* *Gujarat*

Krishna Royal School (GHSEB) 2019 – 2021
12th – Science Stream ***Percentage**: 70%* *Gujarat*

Krishna Science School (GSEB) 2017 – 2019
10th ***Percentage**: 91%* *Gujarat*

PROJECTS

AlphaZero Chess Engine | *C++, Python, Cython, PyTorch, MCTS, Reinforcement Learning* **GitHub**

- Built an AlphaZero-style self-play chess engine from scratch with a C++ chess environment and Cython bindings for high-performance Python-C++ interop, eliminating all handcrafted evaluation functions.
- Implemented Monte Carlo Tree Search (MCTS) coupled with a PyTorch neural network trained entirely via self-play; automated a full pipeline — game generation, replay buffer storage, network training, and model checkpointing.
- Developed an ELO rating system and arena evaluation framework to benchmark successive model checkpoints and track strength progression across training iterations.

C++ Machine Learning Library (AMD GPU) | *C++, Vulkan, OpenCL, GPGPU, Linear Algebra* **GitHub**

- Engineered a from-scratch C++ ML library targeting AMD GPU acceleration via Vulkan compute shaders and OpenCL kernels, optimizing tensor operations and matrix multiplication for non-CUDA GPU architectures.
- Implemented core ML primitives including forward/backward passes, activation functions, and gradient descent optimizers with hardware-aware memory management and GPU workgroup tuning.
- Designed a clean API for building and training neural networks entirely in C++, achieving significant speedups over CPU-only execution by leveraging parallel GPU compute pipelines.

Custom 5×6_5 Handwired Keyboard (RP2040 Zero) | *C, QMK Firmware, RP2040, Python, CAD/3D Print* **GitHub**

- Designed and hand-wired a full Dactyl Manuform-style ergonomic split keyboard (5×6_5 layout) around the RP2040 Zero MCU, including custom PCB-free wiring, diode matrix soldering, and TRRS interconnect.
- Authored a custom QMK keymap and firmware configuration for the RP2040 target; set up the full QMK toolchain with Python-based build automation and UF2 flashing workflow for the RP2040 bootloader.
- Produced STL models for 3D-printed case components and documented the full build process — wiring diagrams, soldering reference photos, and flashing instructions — for reproducibility.

TECHNICAL SKILLS

Languages: C++, C, Python, Java, Go, JavaScript, Bash
Developer Tools: Git, Docker, Linux, QMK Firmware, VS Code
ML / **AI**: PyTorch, NumPy, Cython, Reinforcement Learning, MCTS, Neural Networks
GPU / **Systems**: Vulkan Compute, OpenCL, GPGPU, RP2040 / Embedded C, 3D Printing (STL/CAD)
Coursework: Data Structures & Algorithms, Operating Systems, Computer Networks, DBMS, Machine Learning
Areas of Interest: Systems Programming, GPU Computing, AI/ML, Embedded Systems, Compiler Design