Ernesto Ibanez Jr.

602-214-1276 | jribaneze@gmail.com | linkedin.com/in/ernestoibanezjr | github.com/ernestoIJ | ernestoij.github.io

Education

Arizona State University — Tempe, AZ
M.S. Robotics and Autonomous Systems (AI) — *Aug. 2026 – Dec 2027*

Arizona State University — Tempe, AZ
Bachelor of Science in Computer Science — *Jan. 2024 – Dec. 2025*

- **Grades:** 4.0 GPA, Dean's List

Experience

Rubitection Inc. — September 2024 – December 2024
AI/ML Intern — *Remote*

- Developed a **Selenium-based web scraping** tool, **expanding the companies dataset by 2x** for future machine learning applications.
- Designed and implemented a high-accuracy image classification model using **PyTorch** and **transfer learning**, achieving **95.83% accuracy**, **94.03% precision, recall, and F1-scores** to effectively distinguish between clean and invalid images.
- Collaborated on building a **Flask web application** integrated with **AWS SageMaker**, facilitating image uploads and laying the groundwork for an advanced image segmentation model, while gaining proficiency in cloud services and deployment.

Projects

ASU Capstone Management Platform | *Next.js, MySQL, Docker, Node.js, TypeScript*

- Collaborated with a team of 6 students and a faculty member to develop and deploy a CS/CSE Capstone platform **used by 400+ students, 50+ sponsors, and faculty** to manage project proposals, seminar submissions, and student assignments.
- Built admin dashboards for reviewing and approving proposals, managing assignments, and tracking unassigned students — **cutting faculty manual workload by about 86%.**
- Containerized the application using Docker and **deployed it for real-world use on ASU's infrastructure**, achieving stable performance and smooth live testing feedback.

MiniGPT From Scratch | *Python, PyTorch*

- Implemented a custom **Byte Pair Encoding (BPE) tokenizer from scratch**, supporting vocabulary sizes up to 4K and **reducing token count by about 35%** versus raw character encoding.
- Built and trained a **Transformer-based Large Language Model (LLM) from scratch** in **PyTorch**, replicating the core architecture of GPT-style models
- Designed custom training and sampling pipeline, achieving stable convergence (**dropping validation loss from 15.5 to 9.4**), and text generation from scratch on CPU/MPS hardware.

ML-Powered Rock-Paper-Scissors Robot | *Python, C++*

- Engineered a real-time gesture recognition system using Python, OpenCV, and MediaPipe, achieving approximately **90% gesture detection accuracy** for Rock-Paper-Scissors gameplay.
- Implemented and compared three predictive models (**Conditional Probability, Markov Model, Q-Learning**), with the Markov model improving win rate against human players **from 33% to over 50%** after about 50 rounds.
- Integrated machine learning predictions with Arduino-controlled robotic hand, enabling **physical response within 1 second** of user input and **successfully executing over 200+ rounds of interactive**, AI-powered gameplay.

Distributed Model Training | *Python, PyTorch, Flask*

- **Developed a Distributed Data Parallel (DDP) model** to facilitate efficient training of deep learning models across multiple computational nodes, achieving a **42% reduction** in training time compared to traditional single-node setups.
- Implemented robust setup and cleanup procedures for distributed training environments using PyTorch's **torch.distributed** and **torch.multiprocessing** modules, enhancing the **scalability and reproducibility** of machine learning experiments.
- Designed and implemented a simple web interface that allows users to configure training parameters and launch distributed training sessions easily.

Technical Skills

Languages: Java, Python, C/C++, SQL, JavaScript/TypeScript, HTML/CSS
Frameworks: React, Node.js, Flask, Express.js, PyTorch
Developer Tools: Git, Postman, Google Colab, Jupyter Notebook, Firebase, Linux, Docker