

Michael Capriotti

in [LinkedIn](#) [Github](#) [Portfolio](#) ✉ michaelcapriotti2028@u.northwestern.edu

Education

Northwestern University - Bachelor's degree in **Computer Science & Mathematics**

Evanston, IL

- **Coursework:** Data Structures & Algorithms, MENU Linear Algebra & Multivariable Calculus

2024 - 2028

Work Experience

Hudson Design House

Oswego, IL / Remote

Software Development Intern — *JavaScript, React, Node.js, Express, MongoDB, Tailwind*

June 2025 – Present

- Developed a full-stack e-commerce platform with React frontend and Node/Express backend, integrated Google Places API for streamlined checkout via address autofill and implemented a custom AI chatbot with React Router providing customer support.
- Integrated the Square API for dynamic category/product management, order processing, and secure checkout with the Web Payments SDK which prevents overselling with concurrent users via a custom session-based reservation system.
- Preparing the application for deployment through error testing, performance optimization, and scalability improvements.

Research Projects

Los Alamos National Laboratory

Remote

Quantum Research — *Python, QAOA, NumPy, Scipy, Matplotlib, Qiskit, CVXPY, Jupyter Notebook*

- Applied the Quantum Approximate Optimization Algorithm (QAOA) to solve combinatorial optimization problems (portfolio optimization, travelling salesman, and maximum independent set), by transforming QUBO problems into Max-Cut instances.
- Utilized Semidefinite Programming (SDP)-based “warm-start” strategies, leveraging classical optimization to accelerate and improve the initialization of QAOA; currently revising manuscript for submission to ACM.

Northwestern Kellogg School of Management

Evanston, IL

Financial Research — *Python, Pytesseract OCR, LLMs, Openpyxl*

- Developed scalable automated obituary classification by transitioning from manual labeling (500 samples) to OCR-based data extraction and automatic labeling using sentence embeddings, Logistic/XGBoost models, and large language models (LLMs).

Software Projects

Stock Forecast — *FastAPI, React, TypeScript, Python, NumPy, Pandas, TensorFlow, Sqlite3*

- Engineered a machine learning web application predicting next-day stock closing prices for 400+ companies using individual LSTM models trained on 5 years of historical market data, achieving an average mean absolute error (MAE) of 2.76%.
- Built an end-to-end system with a Python/FastAPI backend to fetch, clean, and process stock data via Yahooquery, train per-stock LSTM models, store results in Sqlite3, and serve them to a React dashboard with Chart.js visualizations for interactive analysis.

Golf Swing Analyzer — *Flask, Random Forest, Python, MediaPipe, OpenCV, NumPy*

- Created and deployed a full-stack web app enabling users to upload, trim, and analyze golf swing videos, classifying them as Pro or Amateur with interactive annotated pose overlays while minimizing backend memory usage.
- Collected and processed a dataset of 100 golf swing videos, extracting and normalizing 3D coordinates, velocities, and joint angles for 13 key body landmarks using MediaPipe Pose, and trained a Random Forest Classifier achieving around an 80% accuracy.

Trading Simulation — *C++, Python, Matplotlib, CSV*

- Developed a single-ticker market simulator in C++ modeling an order book with price-sorted matching and market dynamics, featuring five distinct trading bot strategies (Value Investor, Trend Follower, Stop Loss, Market Maker, Noise).
- Engineered bot logic to submit orders each tick with price, quantity, and time-to-live, tracked portfolios across 2,000 ticks, and computed real-time metrics for best, worst, and average performance per bot type, visualized with Matplotlib.

Evolutionary Algorithm Flappy Bird — *Unity, C#, Evolutionary Algorithm*

- Developed a physics-based Flappy Bird clone in Unity with multiple gameplay modes, including training, watching, and playing against an AI agent, with interactive UI controls and real-time visualization of the training process.
- Implemented a custom NEAT-inspired neural network in C# with elitism, mutation, and multi-cycle fitness evaluation, enabling progressive AI improvement across generations and automatic saving of best models for replay and gameplay.

Technical Skills

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

Frameworks/Platforms: React, Node.js, Express, Flask, FastAPI, Unity, MongoDB.

Libraries: Pandas, NumPy, TensorFlow, Scikit-learn, Matplotlib, Qiskit, Mediapipe, Pytesseract.

Additional Information

Programs: MIT Introduction to Engineering, and Science (MITES), Goldman Sachs Engineering Possibilities Summit.

Activities: Institute for Student Business Education, Consultants Advising Student Enterprises, Table Tennis Club.