Andrew Nawat Nuisud

Boston, MA | anuisud@bu.edu | linkedin.com/in/andrewnuisud | github.com/andynuisud

EDUCATION

Boston University Expected May 2027

Bachelor of Arts in Computer Science and Mathematics *Boston, MA*

Relevant Coursework: Linear Algebra, Discrete Math, Computer Organization, Data Structures, Combanatoric Structures

Technical Skills

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

Frameworks & Tools: Spring Boot, Next.js, React.js, Postgres, Docker, AWS, Firebase, GraphQL, Jenkins, Git

Python Libraries: Pandas, NumPy, TensorFlow, OpenCV, Matplotlib

Concepts: Software Engineering, Frontend, Backend, Distributed Systems, Low-Latency Processing, REST APIs

EXPERIENCE

Teaching Assistant – Engineering Programming Jan 2025 – May 2025

University of California *Remote*

- Evaluated 700+ Python lab submissions for 90+ students using Gradescope + Jupyter, achieving 100% on-time grading turnaround across two weekly sections.
- Improved average lab scores by 18% by giving targeted debugging feedback and introducing standardized rubrics that reduced grading variance by 22%.

Software Development Intern Aug 2023 – Jan 2024

Bay Valley Tech *Remote*

- Built and deployed an internal Flask-based blog, improving load time from 2.6s to 1.5s by adding server-side rendering, pagination, and image thumbnailing.
- Reviewed 31 PRs in a 4-dev team, cutting turnaround time from 26h to 15h and raising merge success from 77% to 88% with automated style/lint checks.
- Increased test coverage from 24% to 71% across ~300 LOC, reducing defects per release from 7 to 2 by writing 42 unit + integration tests with pytest.

PROJECTS

Backend Banking System | ***Java, Spring Boot, SQL (Basic Queries, H2)***

- Designed and exposed RESTful API endpoints (POST, GET) with clear request/response validation DTO classes.
- Processed 50 simulated transactions with a 98.7% success rate by designing RESTful APIs and Spring Data JPA queries with schema auto-generation.
- Reduced transaction latency to 84 ms avg by implementing concurrent fund transfers and in-memory H2 indexing.

Automated Game Asset Trading System | ***Python, Firebase, REST APIs***

- Executed 5,326 trades over 11 days, generating ~$3,000 in in-game value with an 8.43% average margin by designing valuation algorithms and arbitrage detection.
- Increased throughput from 182 to 294 trades/hr by introducing multithreaded API calls and built a CLI dashboard displaying live trade logs with <100ms latency.

Self-Driving Line-Following Robot | ***C++, Embedded Systems, IR Sensors***

- Completed 2 of 2 course trials in under 15s avg by integrating IR sensor thresholds with PWM-tuned motor control and direction memory logic.
- Increased lost-line recovery success from 61.2% to 79.5% by adding boost recovery and stuck-turn detection, reducing course failures by 22.4%.
- Implemented boost recovery with stuck-turn detection improved line re-acquisition consistency by 30%.

Outfit Matcher AI | ***Python, OpenAI API, Selenium***

- Achieved 96.4% attribute extraction accuracy across 57 diverse test cases by combining GPT-4 Vision with perceptual hashing and Hamming distance.
- Scraped 1,482 product listings from Depop in 42 minutes using Selenium with dynamic waits, returning structured metadata for 12 clothing categories from Depop.

ACHIEVEMENTS

Hewlett Packard Enterprise CodeWars Programming Competition *March 2023*

3rd Place Winner - Annual Hackathon (Top 1.5% of 200+ participants)

- Solved competitive programming challenges using graph algorithms (DFS/BFS traversal), sorting algorithms (quicksort, mergesort), and dynamic programming in Java with 3-member team.
- Implemented data structures including binary trees, hash tables, and adjacency lists to optimize performance.