

Abdelgadir Osman

o.abdelgadir32@gmail.com | [linkedin.com/in/abdelgadir-osman](https://www.linkedin.com/in/abdelgadir-osman) | abdelgadirosman.com

EDUCATION

The University of Texas at Austin

Electrical & Computer Engineering – Software Engineering & Design – GPA: 3.6

Coursework: Software Design & Implementation II, Algorithms, Digital Logic Design, Embedded Systems, Operating Systems, Probability, Circuit Theory, Linear Systems & Signals

Austin, TX

Anticipated May. 2027

SKILLS

- **Languages:** Python, C, C++, C#, HTML/CSS, Java, JavaScript, Swift, TypeScript, Verilog, MATLAB
- **Frameworks/Libraries:** Node.js, Next.js, PyTorch, TensorFlow, OpenCV, Playwright, NumPy, Vitest, FastAPI
- **Tools & Platforms:** Git, GitHub Actions, Docker, PostgreSQL, CI/CD, Spinnaker, Datadog, Splunk, Jetson Nano, Maven
- **Certifications:** JPMC Software Engr. Job Sim, Ford Digital Advanced & EV Engineer Job Sim

EXPERIENCE

Expedia

Software Development Engineering Intern

Austin, TX

May 2026 – July 2026

- Built an agentic TypeScript and Node.js security pipeline that scanned application dependencies, embedded Go binaries, and container OS/distribution packages for known vulnerabilities and prioritized actionable findings.
- Automated dependency remediation through root-package tracing, validated version upgrades, CI checks, re-scanning, and reviewer-ready pull request generation, fixing 50+ production vulnerabilities.
- Developed Playwright-based runtime testing and deterministic safety controls to catch broken user flows, unsafe dependency upgrades, validation failures, and production regressions before engineer review.

Flex

Software Engineering Intern

Austin, TX

May 2025 – August 2025

- Developed and deployed an AI visual inspection system for Mac Pro motherboards, leveraging Anomalib Patchcore and transferring learning on extensive datasets to accurately detect over 10 distinct critical defects.
- This AI solution saves hours and ~\$27,405 quarterly by identifying defects much earlier in the production cycle, drastically cutting rework costs, and eliminating significant manual inspection time.
- Built a VMI/Receiving automation system on the NVIDIA Jetson Nano using Google Vision API and Python to detect and scan incoming boxes and auto-populate part, box, location, etc. speeding up operations by hours daily

UT Austin

ML Research Engineer Assistant

Austin, TX

Jan 2024 – Apr 2024

- Developed and trained AI models using Python and TensorFlow to analyze medical imaging and patient datasets, enhancing diagnostic accuracy and efficiency through advanced machine learning techniques.
- Conducted data preprocessing, feature extraction, and statistical analysis to optimize model performance, ensuring robust and reliable predictive outcomes.
- Collaborated with a multidisciplinary team to design and validate AI-driven solutions for personalized medicine and improved patient care.

PROJECTS

Helm - helmpm.app — AI-Native Product Management Platform

Dec. 2025

Next.js, FastAPI, PostgreSQL, OpenAI

- Built a “Cursor for PMs,” replacing Jira and the like with natural-language ticket creation, automated daily briefings, and real-time meeting capture that converts discussions directly into actionable tickets.
- Architected an agentic LLM system with 25+ callable tools using OpenAI function calling, enabling contextual reasoning across tickets, GitHub pull requests, commits, and meeting transcripts to answer complex product questions in chat.
- Designed and implemented a hybrid retrieval system combining SQL-based filtering with vector embeddings across project data, powering automated briefings that summarize activity, identify blockers, and recommend prioritized next steps.

Space Invaders Embedded Video Game

Mar. 2025

Embedded C – Circuit Design

- Developed a handheld video game on the TI MSPM0+ microcontroller, integrating an LCD display, slide potentiometer, and buttons for comprehensive user interaction.
- Implemented an interrupt-driven software architecture, leveraging ADC for real-time potentiometer input and DAC for sound generation.
- Engineered a modular codebase with custom sprite rendering, emphasizing scalability and adaptability for future growth.