

Tanner J. Berman

berman.tanner@gmail.com | [linkedin.com/in/tanner-berman](https://www.linkedin.com/in/tanner-berman) | github.com/bermantanner

Education

University of California San Diego

Bachelor of Science in Computer Science

GPA: 3.73

Expected Graduation: Fall 2026

Technical Skills: C++, C, Go, Python, HTML/CSS/JavaScript, React, PostgreSQL, MongoDB, Express, n8n, TypeScript, Docker

Relevant Coursework: Operating Systems, Computer Networks, Parallel Programming, Data Structures & Algorithms, Computer Architecture, Wireless Embedded Systems, Software Engineering, Linear Algebra, Statistical Methods

Experience

Parsley360, Software Engineer

San Diego, CA

September 2025 - Present

- Built an automated ETL pipeline (n8n, Apify) with custom JavaScript modules that parse and sanitize unstructured data into PostgreSQL.
- Eliminated out-of-memory crashes and ECONNRESET timeouts on large binary payloads via custom execution loops and chunked batching.
- Designed a fault-tolerant Supabase PostgreSQL schema using SQL and JSONB parsing to manage queue state, deduplication, and relational storage mapping.
- Improved extraction reliability by reverse-engineering request patterns and generating dynamic HTTP headers to get past Cloudflare anti-bot.

Parsley360, SWE Intern

San Diego, CA

June 2025 - September 2025

- Built a multimodal data-ingestion pipeline (Python, C++) that scrapes, OCRs, and cleans documents, cutting manual preprocessing ~90%.
- Modeled a DuckDB/MotherDuck corpus database (S3 URI links, SHA-256 dedup); delivered clean topics/embeddings for the demo's reasoning engine.
- Applied Gensim LDA and BERTopic for topic-driven RAG and ontology mapping; wrote a C++ tokenizer.

Software Lead and Event Host at ACM UCSD

San Diego, CA

January 2025 - June 2025

- Led a 6-person Agile team to architect a MERN stack web app matching UCSD students for rides to airports.
- Built MongoDB schemas via Mongoose/Atlas, integrated Firebase auth, and implemented post filtering.
- Facilitated Agile ceremonies, mentored junior developers, and oversaw backend PR reviews.
- Led and organized multiple events, including an intern panel with 50+ attendees.

Projects

Reverse-Engineered Smart-Bulb BLE Remote (C++, ESP32, BLE, Cryptography)

July 2026 - Present

- Reverse-engineered the undocumented BLE protocol of AiDot/Leedarson smart bulbs (jadx APK decompilation, Wireshark packet analysis, cryptanalysis), reconstructing its session-key derivation and AES command cipher.
- Found and coordinated responsible disclosure of a cryptographic weakness in the protocol, withholding specifics until the vendor's remediation window closes.
- Built ESP32 firmware in C++ (NimBLE, mbedTLS) that performs the authentication handshake to control multiple bulbs locally with no app, cloud, or Wi-Fi dependency.

FinDoc-Eval: Financial Document RAG Pipeline (Python, FastAPI, pgvector, OpenAI)

May - June 2026

- Built an async SEC 10-K RAG pipeline with table-aware PDF parsing, semantic chunking, and pgvector similarity search, streaming source-grounded answers (custom retrieval/synthesis, no LangChain).
- Designed an LLM judge harness to evaluate retrieval and answer accuracy on a curated NVIDIA 10-K dataset.

Broccoli Game Platform (Go, WebSockets, JavaScript)

November 2025 - April 2026

- Developed a local-authoritative multiplayer game engine in Go, using WebSockets for real-time state sync between a desktop host and mobile clients.
- Streamlined deployment with Ngrok TCP tunnels that bypass NAT, letting remote players join via browser without port-forwarding.
- Packaged cross-compiled binary executables for macOS and Windows, abstracting local server initialization and networking into single-click launch scripts.

Low-Power BLE Tracker (C, STM32L4, I2C)

January - April 2025

- Pair-programmed firmware for a BLE-based tracker on STM32L4 hardware, designed to detect when an item is left behind and broadcast a lost signal via BLE.
- Implemented custom drivers for I2C sensors, timers, accelerometer (LSM6DSL), and BLE communication.
- Optimized system power consumption from 110 mA to ~80 μ A using STOP2 mode, LSI-driven timers, and aggressive peripheral gating for multi-month battery life.