

JIAWEI WENG

3723 Oleander Ter, West Palm Beach, FL 33404
(917)-244-4903 · EAD card · weng13860@gmail.com

Objective

Seeking a full-stack Software Development Engineer full-time position, where I can leverage my experience in front-end and back-end development to build scalable applications and deliver high-quality user experiences.

Education

University of Florida, FL
B.S. in Computer Science

2021 – 2025

Technical Strengths

| | |
|--------------------|---|
| Computer Languages | C++, C, Java, Python, SQL, TypeScript |
| Frameworks | React, Node.js, Next.js |
| Tools | Git, Clerk, Google Cloud Platform, Supabase, Prisma (ORM), Docker, MongoDB, RESTful API |

Course Project

Cache Simulator | *Simulation Program* | 2023

- Developed a cache simulator using C++ to read a memory access trace from a file, determine whether each memory access is a hit or a miss, and calculate the overall hit rate.
- It simulates different main memory and cache sizes, cache associativity (direct mapped, n-way associative, fully associative), and replacement methods (LRU, FIFO).

Custom Binary Search Tree | *Data Structure Program* | 2022

- Created a custom BST using C++ that can store student names and IDs, providing operations to add, remove, and search students in the tree.
- Supports printing student data in different traversals (in-order, pre-order, post-order), allowing easy organization and retrieval based on IDs.

Side Project

AIDE | *Full-Stack Developer* | September 2024 – Present

- User Authentication: Integrated Clerk for secure sign-ups, logins, and session management, handling identity verification seamlessly.
- RESTful APIs and AI Services: Built a front-end/back-end data flow via RESTful APIs, utilizing third-party LLM endpoints for advanced AI responses. Introduced a *tool call* mechanism (e.g., real-time weather, calculator) to ensure more accurate, context-aware outputs.
- Data Persistence: Employed Supabase (PostgreSQL + Prisma) to store user conversation content and manage version control. This setup maintains robust data integrity while simplifying schema migrations.
- Editing Lock for Collaboration: Implemented a lightweight concurrency lock so that once a user is editing a piece of content, others must wait until it's released—preventing overwrite conflicts and ensuring consistent versioning.