Albert Felix

505-603-3072 | afelix13@ucsc.edu | www.linkedin.com/in/albert-felix | https://github.com/Alb111 | Portfolio Link

Education

University of California, Santa Cruz — Santa Cruz, CA
Bachelor of Science in Computer Engineering — *Sept. 2023 – June 2026*

Experience

Undergraduate Research Assistant - Agents and Compositionality Group — Dec 2023 – Jul 2024
University of California, Santa Cruz — *Santa Cruz, CA*

- Led and contributed to the design of the front-end for a full-stack web application using Svelte-Kit, Tailwind, and TypeScript, enabling researchers to better document and standardize multi-model AI architectures.
- Worked on methods to store and build upon these multi-model systems efficiently, fostering greater collaboration in research.
- Explored approaches for AI agents to optimize these multi-model architectures.

Projects

Slug Assistant *(Cruz Hacks Winner)* | *Svelte-Kit, Supabase, Tailwind-CSS, TypeScript, Skeleton UI* — Feb 2024

- Led Front-end Development to create a school mascot-themed AI assistant to help incoming college students better handle their daily schedules and responsibilities.
- Effectively implemented OpenAI's API and simple prompting to facilitate a welcoming tone of communication.
- Repo: https://github.com/Sluggish-Solutions/Slug-Mommy
- Devpost: https://devpost.com/software/slug-mommy

Pomato | *Svelte-Kit, Supabase, Tailwind-CSS, TypeScript, Skeleton UI* — Mar 2025 – Present

- Developing a full-stack web application using Supabase and Svelte-Kit to help users implement the Pomodoro technique for more efficient task completion.
- Implementing OAuth to securely store user study sessions and track progress.
- Creating features to display past user data, study sessions, and completed tasks, providing progress tracking and motivation.
- Repo: https://github.com/Alb111/Pomato_V1

CLI-based Hangman | *C* — May 2024

- Developed a command-line interface (CLI)-based Hangman game with ASCII art, providing a visually engaging user experience.
- Implemented game logic for word selection, tracking player guesses, and managing game states such as wins/losses.
- Created a robust input validation system to handle incorrect guesses and provide real-time feedback.
- Repo/Src: Can be shown on request, currently private for academic purposes.

Dino Run Game (Offline Chrome Clone) | *Verilog, FPGA, Verible, Vivado* — July 2024

- Designed and implemented a hardware-based side-scrolling runner game inspired by Chrome's offline Dino Run using Verilog on an FPGA.
- Built real-time game logic including obstacle generation, collision detection, jump mechanics, and scoring on a VGA display.
- Integrated user input, sprite animation, and frame-based timing to simulate smooth gameplay on FPGA hardware.
- Repo/Src: Can be shown on request — currently private for academic purposes.

Technical Skills

Languages: Java, Python, C/C++, JavaScript, HTML/CSS, Verilog, Bash
Frameworks: Node.js, Svelte-Kit, PyTorch (learning)
Developer Tools: Git, VS Code, Vim-motions, Linux
Libraries: pandas (learning), NumPy (learning)
CAD Tools: AutoCAD Fusion, Blender