

Albert Felix

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

EDUCATION

University of California, Santa Cruz

Bachelor of Science in Computer Engineering

Santa Cruz, CA

Sept. 2023 – June 2026

EXPERIENCE

Undergraduate Research Assistant - Agents and Compositionality Group

Dec 2023 – Present

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.
- Currently working on methods to store and build upon these multi-model systems efficiently, fostering greater collaboration in research.
- Exploring approaches for AI agents to optimize these multi-model architectures.

PROJECTS

Pomato | *Svelte Kit, Supabase, Tailwind-CSS, Typescript, Skeleton UI*

Sept 2024 – 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

Slug Assistant *Cruz Hacks Winner* | *Svelte Kit, Supabase, Tailwind-CSS, Typescript, Skeleton UI* Feb 2024

- Lead 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>

Stop it Game | *Verilog, FPGA, Verible, Vivado*

July 2024

- Developed a real-time timing game, Stop It, on an FPGA using Verilog, involving precise timing and control logic.
- Implemented digital logic circuits to handle game state, timing, score, and user input for start/stop functionality on an LED display and a set of score LED's.
- Tested and debugged the system on FPGA hardware, ensuring reliable timing precision and responsive gameplay.
- Repo/Src: Can be shown on request currently private for academic purposes.

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.

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)