

Daniel Wan Rosli

LinkedIn: <https://www.linkedin.com/in/danielwanrosli>

danwr@stanford.edu

+1-262-744-9648

Skills Summary

- **Software and Tools:** Python, C++, C, C#, JavaScript, TypeScript, Unix, Linux, Git, SQL, LLM+GPT
- **Web Development:** HTML, CSS, React, Angular, Express, MongoDB, Node, Flask, Django
- **Mobile Development:** React Native, Expo, Firebase, Supabase
- **Human-Computer Interaction:** UI/UX Design (Figma), HMW/POV Statements, Accessible Design, Needfinding, Usability Testing, Iterative Design
- **Methodologies and Practices:** Reactive Web Design, Functional Programming, MVC, OOP, A/B Testing, Agile/Scrum, Code Reviews

Education

Stanford University

M.S. Candidate in Computer Science

Jan. 2023 - Jun. 2024

B.S. Candidate in Computer Science

Sep. 2019 - Jun. 2024

Specializing in Human-Computer Interaction

Relevant Coursework: Design and Analysis of Algorithms; Principles of Computer Systems; Web Applications; Data Management and Data Systems; Designing for Accessibility; Data Visualization; Human-Centered Product Management; Machine Learning with Graphs

Work Experience

Course Assistant (CA) — Cross-Platform Mobile Development

Sep. 2023 - Dec. 2023

- As one of four CAs, co-designed and helped lead Stanford's first cross-platform mobile development course, focusing on functional programming and full-stack development, for ~120 students.
- Provided detailed feedback for ~25 students' programming assignments each week, assessing both functionality and coding style.
- Conducted weekly office hours to mentor students in React Native development through code debugging and conceptual help.

Software Engineer Intern — Tarjimly

Jun. 2023 - Aug 2023

- Achieved up to a 10x improvement in recall by reducing pings sent from hundreds to 30 and attained a 90% match rate by developing an improved recommender system in Python using OOP, in collaboration with a team of three interns.
- Using Serverless Framework, worked with lead engineer to transition the recommender system to an event-driven system with AWS (RDS, SQS, SNS, Lambda)
- Leveraged Python, pandas, and SQL in a Jupyter Notebook to analyze translator retention and response rates, guiding feature extraction and algorithm design.
- Created unit tests for the recommender system components and validated AWS Lambda endpoints using Postman.

Frontend Engineer Intern — Ample

Jul. 2022 - Sep. 2022

- Spearheaded frontend development as the primary engineer for a cross-platform mobile app utilizing TypeScript and React Native.
- Worked with the founder to use Figma to design task-flows and enhance usability through design thinking principles.

Software Engineer Intern — Direct Supply

Apr. 2021 - Sep. 2021

- Enhanced web app components with TypeScript and React, adapting to new customer workflows.
- Optimized UI and email processes using C# and OOP, migrated tasks to Nomad servers, and ensured robustness via SQL validations.

Research Experience

CS Research Assistant — Stanford Machine Learning Group

Sep. 2023 - Present

- Working to improve web app displaying methane emissions data using Angular, TypeScript, and MapBox GL JS.
- Using a web accessibility testing tool (WAVE) to improve web app accessibility and usability.

LLM Research Assistant and Consultant — Stanford GSB

Apr. 2023 - Jun. 2023

- Created a full-stack Flask web app to summarize PDF articles using GPT-4, iteratively improving features based on PI's feedback.
- Researched trends and new insights in large language models to provide consulting and prompting insights.

CS Research Assistant — ReactGenie Lab

Jan. 2023 - Jun. 2023

- Developed a React Native prototype with ReactGenie and GPT-4 that enabled multimodal interaction through voice commands.
- Conducted user studies and developed an elicitation study; synthesized results culminating in paper submission to UIST 2023.

CS Research Assistant — GPTeach Lab

Jan. 2023 - Mar. 2023

- Helped develop React prototype for GPTeach: including data retrieval and logging with Firebase, user interface development, and back-end with GPT-3; eventually released to 2000+ users as part of Code in Place.
- Conducted user studies with the prototype, using design thinking to identify needs and refine app features.