

JERRY HE

jerryhe@utexas.edu | [linkedin.com/in/jerryhehq](https://www.linkedin.com/in/jerryhehq) | jerryhehq.com | github.com/jerryhehq

EDUCATION

The University of Texas at Austin (Austin, TX)

Aug 2022 - May 2026

Bachelor of Science in Computer Science, Minor in Robotics

- **GPA:** 4.0, Distinguished College Scholar
- **Relevant Coursework:** Data Structures, Algorithms, Statistics, OS, Architecture, Linear Algebra, Discrete Math
- **Ongoing Coursework:** Software Engineering, Machine Learning, Gateway to Robotics

SKILLS

Collaborative and dedicated third-year undergraduate with 6+ years of web, mobile, and systems development experience.

- **Languages:** Java, Python, C, Javascript, HTML, CSS, Kotlin, Dart, LaTeX
- **Frameworks/Technologies:** ReactJS, Bootstrap, Flutter, Firebase (NoSQL), MySQL, Flask, PyUnit, Linux/Unix
- **Tools:** Git, GitHub, Visual Studio Code, Android Studio, Pandas, Numpy, Scikit learn, PyUnit, Gradle, Figma

EXPERIENCE

Human-Computer Interaction Lab (Austin, TX)

Aug 2024 - Present

HCI Independent Researcher

- Researching the effects of mouse sensor position on the 3D aiming abilities of flicking, tracking, and switching

C S 313E Elements Of Software Design (Austin, TX)

Aug 2024 - Present

Undergraduate Course Assistant

- Guided 300+ students in understanding data structures through the establishment of rubrics and timely feedback
- Improved student comprehension by leading 30+ sessions to address misunderstandings and coding issues
- Helped students progress their programming skills with the construction of coding tutorials and PyUnit tests

ECLAIR Robotics (Austin, TX)

Jan 2024 - Present

Autonomous Vehicle Engineer (Ongoing Project)

- Developing a small-scale autonomous vehicle capable of navigating a track and adapting to human instructions
- Implementing autonomous navigation through bifocal computer vision and integrating NLP instruction processing
- Enabled remote operation and live video streaming using a Raspberry Pi control system programmed in Python

PROJECTS

DotFlare (<https://github.com/jerryhehq/dotflare-v2>)

Jan 2024 - May 2024

- Developed a React web app to help users provide rapid and targeted feedback with token-based sessions
- Implemented features for users to upload images and annotate them with feedback dots in the formal analysis format, select feedback subject areas, and categorize the feedback as positive or negative for quick critiques
- Designed an intuitive interface using Bootstrap for senders to visualize feedback concentration and details, enhancing design iteration based on user insights and allowing for rapid prototyping

PintOS Implementation

Aug 2023 - Dec 2023

- Collaborated with a team of four to implement the functionalities of PintOS as part of an operating systems course
- Programmed critical system features, including system calls, process scheduling algorithms, thread synchronization mechanisms, deadlock avoidance, virtual memory management, and file system operations
- Ensured optimal and seamless integration of system components in C through extensive planning and discussion

ARM System Emulator

Jan 2023 - May 2023

- Pair-programmed an ARM instruction simulator for a custom ISA in C as part of a computer architecture course
- Constructed a 5-Stage FDEMW instruction pipeline with signal/hazard control and a two-level memory hierarchy, resulting in improved simulation correctness and throughput by more than 5x compared to a single stage simulator
- Consulted the ARM Reference Manual to optimize development processes and ensure adherence to specifications

Sertinary Application (<https://github.com/jerryhehq/Sertinary>)

Jan 2022 - Jun 2022

- Developed a cross-platform mobile app using Flutter and Firebase to help users achieve their health goals
- Deployed features such as a food intake tracker, nutritional information display, real-time custom recipe sharing, and persistent alarms each with an intuitive and user-friendly interface
- Achieved a streamlined integration of Firebase Firestore backend services with fast and reliable data updates