

Ava Hamedi

US Citizen | @Hamedi.ava74@gmail.com | ☎ (209)312-2837 | in [Ava-Hamedi](#) | 📄 [Personal Website](#)

📍 San Diego, CA |

EDUCATION

University of California, San Diego (UCSD)

Bachelor of Science in Computer Science; GPA: 3.69

San Diego, CA

Sep 2020 - Mar 2023

RELEVANT COURSES

Data Structure (I, II, Advance)
Software Engineering
Software Web Development
Machine Learning

Software Tools in Linux
Algorithm Design & Analysis
Network Service

Computer Organizing & System Programming
Data Base System Principles
Computer Security (Base)

SKILLS

- **Programming Languages and Scripts:** Java, C/ C++, Java Script, SQL, Go, Bash/Linux
- **Software Web Development Frameworks:** HTML, CSS, React, Node.js, Express.js
- **Software Engineering Techniques & Skills:** Unit Testing, Integration Testing, UI Testing, GDB, Junit, Data Structure
- **Personal Skills:** Analytical Reasoning, Creative problem-solving, decision-making, Ability to work under pressure, Awareness of trends in business and technology, Collaboration / Teamwork, Time management, Detail-oriented, Comfortable multi-tasking
- **Other skills:** Mathematics, Microsoft Office

WORK EXPERIENCE

Marshall's

Store Associate

San Diego, CA

Sep 2018 – Jun 2019

- **i:** Trained new employees in money transactions and more effective ways of communication
- **ii:** Worked at a rapid pace to meet daily store opening deadlines
- **iii:** Balanced the needs of multiple customers simultaneously in a fast-paced environment

Balboa International Market

Front End Associate

San Diego, CA

Oct 2016 – Aug 2018

- **i:** Worked and managed the store front by answering customer phone calls and conducting money transactions
- **ii:** Communicated effectively with customers and assessed their immediate needs
- **iii:** Directed and trained new employees to help create a team environment

PROJECT HIGHLIGHTS

- **Task Manager:** [Task Manager](#)
 - Implemented a full-stack web application that allows users to manage their to-do lists efficiently. Utilized React for the front-end and Express.js for the back-end. Additionally, the app includes a unique feature that generates random motivational quotes.
 - Created a responsive and user-friendly front-end using React, ensuring a smooth and intuitive user interface
 - Integrated Express.js to build a robust and secure back-end, handling data retrieval, storage, and authentication
 - Implemented user authentication with Passport.js, enabling users to securely sign up, log in, and manage their personal to-do lists
 - Employed Axios to handle asynchronous HTTP requests, enabling smooth communication between the front-end and back-end
 - Implemented dynamic rendering of to-do items, allowing users to add and remove tasks effortlessly
 - Main Tools: React, HTML, CSS, Axios, React Router, Express.js, Node.js, Passport.js, MongoDB
- **File compressor:**
 - implemented a program to compress and decompress a variety of file types
 - Used different data structures and the Huffman encoding algorithm
 - Saved the header of the file while compressing it for the purpose of being able to decompress it again

- Main Tools: C++, GDB, Unit test, Integration test
- **Maze Solver:**
 - Checked all the ways from the beginning to the ending point, and saved the shortest one in each comparison by implementing BFS & DFS
 - Recognized the obstacle path and removed them by using BFS and DFS Algorithms
 - Main Tools: Java, Junit, Unit test, Integration test
- **Sending notification to devices for exposure:**
 - Implemented a program to send notifications to devices indicating the number of COVID exposures in the target area
 - Notifying the people in the target area and updating exposure numbers by sending warning
 - Main Tools: Java, Unit test, Integration test
- **HTTP request and response:**
 - Developed a web server that implements a subset of the HTTP/1.1 protocol specification called TritonHTTP
 - TritonHTTP implementation is a client/server HTTP protocol that is layered on top of the reliable stream-oriented transport protocol TCP
 - Main Tools: Go Language, TCP, HTTP
- **Spam Checker:**
 - Implemented an interactive program to check if an email address is on a list of spam addresses
 - Examined the file with the list of all spam emails and filtered all of them at the time of checking for the validity of emails
 - Main Tools: C, GDB, Unit test, Integration test