

Tarun Ajjarapu

972-515-7877 | tarun.ajjarapu@utexas.edu | [linkedin.com/in/tarunajj](https://www.linkedin.com/in/tarunajj) | github.com/tarunajjarapu | tarunajjarapu.github.io/

EDUCATION

University of Texas at Austin

Austin, TX

Bachelor of Science in Computer Science, Minor in Business

Expected Graduation: May 2026

GPA: 3.8/4.0

Relevant Coursework: Data Structures and Algorithms, Operating Systems, Software Engineering, Mobile Computing, Computer Architecture, CS Statistics, Multi-variable Calculus, Competitive Programming

Honors and Awards: University Honors

EXPERIENCE

Incoming Software Engineer Intern

June 2024 – August 2024

JPMorgan Chase & Co.

Plano, TX

Software Engineer Intern

June 2023 – August 2023

Trend Micro

Austin, TX

- Developed Python-based REST APIs using Chalice Framework, Marshmallow, and Boto3 to streamline the Business Intelligence (BI) scan process for retrieving Stack (Security VPC) settings and NSVA (Network Security Virtual Appliances) information from AWS resources, resulting in a 40x reduction in API response time.
- Leveraged the boto3 library to interact with AWS services and efficiently collect and aggregate data from AWS resources in production, consolidating over 100 individual calls every 30 minutes into a streamlined process for retrieving stack information in a single request.
- Utilized the Chalice framework to handle API routing, request validation, and integration with AWS services, ensuring seamless communication and data retrieval from the NSHI (Network Security Hosted Infrastructure).
- Employed the marshmallow library for data serialization and deserialization, ensuring efficient and consistent handling of data structures within the APIs.
- Conducted thorough unit testing for the developed APIs using Pytest to verify the functionality, performance, and reliability of the REST APIs and simulate endpoints to be used by prospective clients.

PROJECTS

Unshakable Japan | *Python, Flask, React.JS, Docker, MySQL, AWS EC2*

January 2024 – March 2024

- www.UnshakableJapan.me: Developed a Flask-based web application to display real-time information about earthquakes in Japan, integrating Google Maps API for geolocation and providing features for sorting and searching earthquake attributes, prefectures, and resources.
- Implemented REST APIs using Flask, Python, and MySQL to scrape data from the internet and create relational connections between instances, while deploying the app using Docker on AWS EC2 for scalable online access.

Web App for JPMorgan Chase CFG | *Node.JS, Express, MongoDB, React Native*

October 2023

- As the backend developer in a team at JPMorgan Chase's Code for Good Hackathon, built APIs for a full-stack web application for Mothers Against Drunk Driving (MADD), featuring a gamified score system, educational content, and user rewards to incentivize responsible driving.
- Constructed backend REST APIs using Node.js, Express, and MongoDB for user profile customization and driving score tracking, and implemented secure user authentication using JSON Web Tokens and bcrypt for data security.

Dynamic Memory Allocator | *C*

January 2023 – March 2023

- Developed a dynamic memory allocator in C, using a memory address sorted policy for free memory block insertion and a first-fit allocation strategy for efficient block assignment, averaging 2200 operations per millisecond.
- Implemented memory block splitting to utilize available space within allocated blocks and employed coalescing techniques to merge adjacent free blocks, successfully attaining an average memory utilization of over 82% during stress tests.

TECHNICAL SKILLS

Languages: Java, Python, C/C++, JavaScript, ARM, HTML/CSS, Kotlin, XML, Swift

Frameworks: React.JS, Node.JS, Flask, Chalice, Boto3, Marshmallow, Pytest, Botocore, Express, Requests

License and Certifications: AWS Certified Cloud Practitioner

Databases: MySQL, SQL (Postgres), SQLAlchemy, MongoDB, DynamoDB, Amazon RDS, Firebase

Other Skills: RESTful API, AWS CF / EC2 / Lambda / RedShift, Git, Docker, Postman, Agile/Scrum methodology