

Tarun Krishna Dasari

(813-748-5756) • tarunkdasari@gmail.com • linkedin.com/in/tarun-krishna-dasari-61842519b/

EDUCATION

University of Florida | Gainesville, FL

Expected Graduation: May 2025

Bachelor of Science in Computer Science at Herbert Wertheim College of Engineering

GPA: 3.63

2024 University Research Scholar for the College of Engineering

Relevant Coursework: Data Structures & Algorithms, Artificial Intelligence Fundamentals, Intro to Software Engineering, Generating Expressiveness in Intelligent Agents and Avatars, Human-Computer Interaction, Competitive Programming, Operating Systems, User Experience Design, Algorithm Abstraction and Design, Enterprise Software Engineering Practices, Data Analytics

Certificates: Artificial Intelligence Fundamentals & Applications Undergraduate Certificate, MTA: Intro to Programming Using Java, MTA: Intro to Programming Using JavaScript

TECHNICAL SKILLS

Languages | C++, Python, Java, JavaScript, CSS, MATLAB, R, C, Assembly, MySQL

Technologies | Git, RESTful API, Unit-Testing, Makefile, Node.js, Linux, pip, Jira, Microsoft Office

EXPERIENCE

Undergraduate Research Assistant | Virtual Experiences Research Group Lab at University of Florida

2023-24

- University Research Scholars Program Study: Increasing Knowledge & Intentions to Use HIV PrEP medication among Emerging African American Adults
 - Tools Used: Nodejs, Express, JavaScript, CSS, MySQL, Python, BeautifulSoup, Tiktoken, Pandas, NumPy, ChatGPT API, Synthesia AI Video Generator, Figma
 - Contributed significantly to research aimed at bolstering awareness and utilization of HIV Pre-Exposure Prophylaxis (PrEP) medication among emerging African American adults
 - Conceptualized and developed virtual African American human avatars to deliver HIV PrEP content to users
 - Developed mockup frontend designs in Figma and translated them into functional code using JavaScript and CSS
 - Employed web scraping techniques via BeautifulSoup to curate HIV PrEP embedded content, dynamically filtering the most relevant sections based on users' inquiries.
 - Integrated a prompt-engineered ChatGPT chatbot for personalized interactions and tailored responses to user queries
 - Enhanced user engagement and immersion by converting GPT's responses into AI audio via the OpenAI API

PROJECTS

Tweet Finder Twitter Bot | Python, REST API for Twitter (PyTwitter), OpenCV, Tesseract OCR, Git

2022

Bot capable of scanning a screenshot image of a Tweet and sending the corresponding Tweet link to user

- Implemented OpenCV to scan the image, identify margins and profile pictures, and reformat it to highlight text exclusively
- Utilized Tesseract OCR algorithms to extract text from the reformatted image and convert it into string data, containing tweeter and tweet information, ready to be processed by PyTwitter
- Integrated the PyTwitter library and REST API to query users from the screenshot, scan timelines to locate the tweet, and subsequently reply to the requester with the appropriate tweet link

WeClean Web Application | JavaScript, MongoDB, Express, React, Nodejs, Bootstrap, Google API, Jira, Git

2022-23

Group-collaboration website that enables clean-up volunteers to organize clean-up events on a global scale

- Leveraged the MERN stack technology: incorporated MongoDB for the document database, Express for the Node.js web framework, React with Bootstrap for the front-end, and Node.js as the web server
- Integrated Google Calendar API to track users' clean-up event information and facilitate event creation with dates.
- Created Interactive map using React and Google API that's populated with clean-up events globally
- Utilized Jira as an agile project management tool for task assignment, progress tracking, and managing bug issues within the development team

File Directory Management System | C++, FUSE, Linux, Unit-Testing, Shell Script, Makefile, Git

2023

C++ Project that implemented a userspace filesystem daemon using Filesystem in UserSpace (FUSE) API

- Implemented features such as directory and file structure addition/removal by reading and updating descriptor file data
- Utilized stack data structure and Depth-First Search algorithm to parse for efficient directory traversal and management
- Employed memory allocation techniques to effectively track directory structures and their respective sub-directories/files
- Implemented a userspace daemon via FUSE, enabling users to access, navigate, and manipulate directories and files within the system