Jake Lee

1625 Buckthorn Dr. Hoffman Estates, Illinois 60192
(847) 257-3871 | jaketlee@bu.edu | GitHub: jaketlee07 | LinkedIn: jaketlee | jaketlee.tech

EDUCATION

Boston University, College of Engineering   *Expected Graduation: May 2025*
Bachelor of Science in Computer Engineering
Relevant Coursework: *Applied Algorithms, Software Engineering, Smart and Connected Systems, Computer Networking, Statistics Probability and Data Science, Engineering Design, Discrete Mathematics, Computational Linear Algebra, Logic Design*

EXPERIENCE

BostonHacks   **Boston, MA**
Head of Technology   *February 2023 - Present*
- Led a team of 10 engineers to develop and deploy a successful hackathon website, registration portal, and check-in system for 850+ applications, utilizing React.js, Next.js, and Postgres for a seamless user experience before and during the hackathon
- Developed an administrative portal connected to for streamlined management of applicants, enhancing operational efficiency

PROJECTS

Smart Document Analyzer   *March 2024 - Present*
- Developed a Flask-based web application, Smart Document Analyzer, leveraging Natural Language Processing with spaCy for document analysis, including entity recognition and sentiment analysis, and implemented RESTful APIs.
- Engineered a secure, scalable backend with PostgreSQL and SQLAlchemy, integrating a queue-based system for asynchronous task management and employing user authentication, enhancing data security and application performance
- Utilized Docker for efficient deployment and Python's unit test for rigorous testing, applying MVC architecture and OOP principles to ensure high code quality and maintainability, significantly improving user experience and application reliability

Smart Voting System   *November 2023 - December 2023*
- Built an IoT device system using MQTT5 protocols that communicate with our ESP32 firmware to provide real-time communication within the system of voting booths to track voters through publishers and subscribers
- Implemented an authentication server to verify voting counts and user IDs to prevent multi-voting among all booths
- Designed replicable device systems for the fob and voter booths for scalability to all voting centers

Venus Rover   *October 2023 - November 2023*
- Built a smart self-driving rover for trips between points A and B, that utilizes LIDAR and ultrasonic sensors for collision avoidance along with a remote control through UDP commands for start, stop, and turn on a DDNS
- Implemented a PID algorithm for precise self-correction during turns, ensuring consistent distance from guiding walls
- Utilized the ESP32 WiFi capability to implement a fully self sufficient system

"Carmin" Smart Watch and "Straba" Social Media Hub   *September 2023 - October 2023*
- Collaborated with a team to create an IoT device featuring real-time display, step tracking, and body temperature monitoring
- Developed C code on an ESP32 board utilizing hardware interrupts, RTO tasks, state design, and data visualization
- Utilized a Raspberry Pi and a Router to centralize the data collection and display the data and step leaders among watches

MEALME: Tinder for Food   *November 2022 - December 2022*
- Developed a React-based website that takes user inputs for location and price point, generates randomized restaurant recommendations on Tinder-inspired cards, and employs an algorithm to match users with their ideal dining options
- Integrated the Yelp API with JavaScript to efficiently organize restaurant data and develop a sophisticated matching algorithm, enhancing the website's accuracy and personalization of restaurant recommendations

Pokemon Masters: Victory Road   *October 2022 - December 2022*
- Developed a simulated 2-dimensional grid using C++ which involves the user controlling multiple Trainers to battle Pokemon to defeat the Gyms, go to the Pokemon Center to heal their Pokemon, and interact with wild Pokemon
- Leveraged object oriented programming, user defined types, and operator overloading to track locations of objects in order to properly perform actions at appropriate times based on a finite-state system with error handling

SKILLS

Programming Languages: C, C++, C#, Java, Javascript, Python, MATLAB, Verilog, MIP AL
Web Development: HTML, CSS, React.js, Next.js, Node.js, Tailwind, Firebase, SQL, CanvasJS, DDNS, .NET
Embedded Systems: ESP32, RTOS Task, UART, I2C, Hardware Interrupt, UDP, GPIO, MQTT5
Development Tools: Unix, Github, Visual Studio, Docker, CI/CD, ArduinoIDE, ESP-IDF
Languages: English (Native), Korean (Fluent)