Kinshuk Phalke

+1-(404)-519-7018 | phalkekinshuk@gatech.edu | linkedIn/kinshukphalke | github/xinslu | kinshukphalke.com

EDUCATION

GEORGIA INSTITUTE OF TECHNOLOGY Atlanta, GA | Aug 2021 - May 2024
Bachelors of Science in Computer Science **GPA:** 4.0

Coursework: Data Structure and Algorithms, Honors Discrete Math
Clubs: GreyHat (CyberSec Club), dependently-typed (Programming Language Club)

WORK EXPERIENCE

NCR CORPORATION | Software Engineering Intern Atlanta, GA | May 2022 - Aug 2022

- Worked in the Retail innovation team in building a micro-service in Go to enable Cart service enabling shopping, checkout, transactions, taxes and payments.
- Worked on making API endpoints, integrating with other APIs to do other tasks like tax and get price details of an item.
- Worked on modelling data and made a CI/CD pipeline that performs a series of checks and publishes the Kubernetes Cluster onto Google Cloud Platform.
- Made a Bash Script with 12 presets to help change environment variables for different stages in production, enabled simple deployments, creating and updating internal CA Certificates and managing kubectl pods and deployments.

NOTITIA | Software Engineer Atlanta, GA | Nov 2021 -

- Working with Full-Stack and NLP in building a MVP for a startup at Georgia Tech.
- Built a LDA model in Python and worked with tuning UMAP parameters and implemented a clustering algorithm using FAISS. Worked on UI of a WordCloud using React Flow.
- Optimized the front-end by 60% and back-end by 40%.

PROJECTS

SPEECHTOCODE Python, ReactJS, Node.js
Using AssemblyAI API, ReactJS and Node, SpeechToCode allows you to use your voice to write code in multiple languages to better help you understand and get started in learning a new language. Built a meta language parser that enables you to speak in a language close to English and creates a class based abstraction based on it, which is then easily translated to the language selected. Worked on designing the frontend and connecting the Parser to the frontend using a flask server.

DEUSCL Rust, Common Lisp, Yew Framework
A memory-safe REPL built in Rust. Built a handwritten recursive descent parser and a visitor pattern dependent evaluator. Added a basic type inference, type checking and error handling system. Added dynamic evaluation of functional environment and parameters. Wrote a simple garbage collector and environment cleaner. Building a frontend in Web Assembly in Rust to enable users to use a web based simulation of REPL.

PUBLIXMON MongoDB, Express.JS, React Native, Node.JS, Flask, Node.js
Publixmon is an app to boost retail engagement. Built a smart contract allowing users to mint NFTs based on Order History of each user. Integrated with the NCR BSP layer to pull order details, catalog and inventory details. Built and tested mining NFTs on the Ganache Blockchain and built a node.js server to facilitate blockchain actions. Project Won the First Place at the NCR API Challenge at HackGT 8.

MENTAL HEALTH CHAT Redis, Docker, MongoDB, Express.JS, ReactJS, Node.JS, Socket.IO
A timed chat for people to share and talk about issues their having in relation with their mental health and help them overcome it. A message in the Redis database is set to expire in about 6 hours in order to encourage people to come out and discourage users calling each other out. Created a Dockerized App with a Redis Instance during development.

HACK-A-LANG Python
Worked with a team of 3 for the "hack-a-lang" event, to make the python implementation of the Lox Learning Language, complete with an implementation of dynamic functional environments, type checking, error handling, recursive descent parser and a visitor patter dependent evaluator. Also, spoke at the event about the evaluator, visitor patterns, environment creation and variable handing. Also, created exercise for the participants to work on to increase their knowledge.

ENHANCE Python, Tensorflow, Keras
Created a Res-Net Implementation of the Super Resolution GAN Paper (https://arxiv.org/pdf/1609.04802.pdf) for Single Image Super Resolution. I worked with tensorflow in creating a Residual Blocks and Skip Connections and used the VGG-19 Model to calculate the Content Loss of the Model. The model achieved perceptible improvements in image quality on the DIV2K dataset with a downscale factor 4.

SKILLS

Languages: Python, Rust, Go, JavaScript, Java, C, C++, C#, , Solidity
Full-Stack: ReactJS, Express.js, Node.js, Flutter, Ionic Framework, React Native, Socket.IO, Redux, ReactFlow, Web3.js
Machine Learning: NumPy, Scikit-learn, Open AI Gym, Matplotlib, Pandas, Keras, NLTK, Gensim, FAISS, UMAP
Technologies: Git, Docker, Postman, LaTeX, SQL, MongoDB, Redis, Kubernetes