Zeping He

zepinghe@umd.edu | https://github.com/finallyegg | 716-580-0767

Education

Master of Science, Computer Science *Expect Dec 2021*
University of Maryland College Park, College Park, MD

Bachelor of Science, Geographical Information Science *May 2019*
University at Buffalo, The State University of New York, Buffalo, NY

Skills

Programming Languages: *Java, Python, JavaScript, Go, C/C++, Swift* Support Tools: *Docker, Git, Bash*
Back-end: *Flask, Express.js, AWS Lambda, DynamoDB, Google Firebase* Front-end: *React.js, Node.js*

Experiences

Software Engineer Intern | *Yahoo!* *June 2021 – August 2021*
- Designed and implemented text parser for configuration in JSON and ECMAScript format using ANTLR library as a **Java Maven** project to extract desired information into CSV
- Developed recommendation system based on user visiting pattern using ALS, KMeans Clustering and Binary Classification model in **Spark** and achieved root mean squared error of 0.3

Graduate Research Assistant | *University of Maryland Software Engineering Group* *January 2021 – Present*
- Worked on constrained group meeting projects and actively converting software application into board business logic
- Optimized client updating logic by replacing REST APIs with **webSocket** architectures, improved user experience and set up as code base for other components
- Developed reference utility using **React.js** and **Flask**, served as tech stacks and code sample base

Software Engineer Intern | *Jasper Financial* *May 2020 – August 2020*
- Built user registration and fraud verification APIs in **AWS** API-gateway, Lambda and DynamoDB, served as a critical part in pipeline
- Added serval functions by integrated third party APIs into pipeline and documented usage and revising based on demand of product engineering team

Projects

BitTorrent Client – *Python based BitTorrent client able to corporate with other BitTorrent clients* *May 2021*
- Designed and built python-based application supports downloading files and feeding others from torrent file
- Used multiple threads to download file chunks simultaneously from peer clients
- Supported both **TCP** and **UDP** protocol to maximize operation efficiency
- Implemented **optimistic unchoking**, **rare-first** strategy and **endgame** mode to have the best performance
- Achieved the highest project score among the Computer Network class

My storage - *Distributed key-value file system with versioning over local network* *Dec 2020*
- Implemented distributed file system using **Raft** consensus algorithm over local network
- Supported **CURD** operation through messaging between clients using **HTTP** request
- Constructed file tree at server-side then added **sync** and **comparison** operation to replicate at lower cost
- Added security layer by using **RSA key pair** to verify client's signature before writing
- Extended to file system via **FUSE** interface. Supported file archive and time travel by chaining

Post - Comment App - *Experimental project using **React.js**, **Express.js** and **Firebase*** *Sept 2020*
- Built user services, handled authentication via middleware and local store which uses **JWT token** and use **middleware** to embed into POST request to identify themselves
- Stored posts and comments in separate tables and used scanning to fetch all comments in post from **Google Firebase**
- Wrote frontend using **React.js** and sent updates through **React Hook**

Relevant Coursework

Computer Network, Mobile Programming, Distributed System, Database Architecture and Implementation