CHUXI WANG

(215) 356-6516 | wangchuxi94@gmail.com |Fremont, CA, 94536

EDUCATION

University of San Francisco *Aug. 2020 – Dec. 2022*
- Master of Science in Computer Science (**GPA: 4.0/4.0**)

University of Pennsylvania *Aug. 2016 – May. 2018*
- Master in Biotechnology

Nanjing University (Nanjing, China) *Sep. 2012 – Jun. 2016*
- Bachelor of Science in Biological Sciences

WORKING EXPERIENCE

Meta Platforms, Inc. *[Flask, Socket.IO, React, AWS S3, ML/NLP]* **Menlo Park, CA**
Software Engineering Intern (FAIR Labs) *Jun. 2022 – Aug. 2022*
- Developed a dashboard application with Flask and React for the Human-in-the-loop learning (HiTL) system of the *Droidlet*, where users can easily monitor job progress, view experiment results, and manage assets (datasets, models, etc.) interactively. *Droidlet* is a NLP project for AI researchers to explore ideas around grounded dialogue, interactive learning, and human-computer interfaces. A strong pillar of our approach is HiTL, which allows people to evaluate ML models and provide human signals to improve the model outcomes.
- Updated HiTL pipeline logic to be able to store pipeline metadata and HiTL job logs on AWS S3 bucket. Developed a backend server with *Flask* and *Socket.io* to serve that data to the frontend through streaming.
- Implemented the frontend with React, *React-Router-Dom*, and *Antd* library. Created reusable React components that supported viewing pipeline metadata, viewing the list of jobs belonging to a specific pipeline, searching in and filtering the job lists, monitoring job detail information, and managing pipeline assets.
- Created interactive components for visualizing the model retraining loss and accuracy changes with D3-based library *Recharts*, which helps the user to analyze experiment results efficiently.
- Wrote Python scripts for extracting error tracebacks in past logs and relative interactive job commands, a highly requested feature by the FAIR Lab research team.

Salesforce.com, Inc. *[Java, Lightning Framework, Spring, MySQL, MVC, RESTful API]* **San Francisco, CA**
Software Engineering Intern (Record Access Control) *Jun. 2021 – Aug. 2021*
- Developed the Admin UI for Restriction Rules based on RESTful API for admins to manage rules for accessing different salesforce records. Implemented a highly interactive and accessible list view and detail view using the Lightning Aura framework, which is an MVC framework with a markup view, a JavaScript controller, and a Java model layer. This Admin Restriction Rule UI is a highly requested feature by users and resolves a critical functionality gap as Restriction Rule is an API-only feature in the previous release.
- Worked with cross-functional teams to refactor a copious amount of lightning aura components, added more interfaces to the components, and made them more accessible and forward compatible.
- Maintained and improved the stability of the backend RESTful APIs to create, delete and update Restriction Rule based on Java *Spring*.
- Designed and developed JavaScript unit testing using a host-agnostic JS test framework xUnit.js for the JavaScript controllers and helpers.

University of San Francisco Information Technology Services *[Apex, Python, Unit Testing]* **San Francisco, CA**
Part-time Student Developer (Salesforce Team) *Apr. 2022 – May. 2022*
- Maintained the USF Student Hub web application which is a Salesforce Application and wrote Apex tests to improve the application's test coverage. Developed easy-to-use utility scripts to visualize duplications in charity donation records of USF using Python and Salesforce APIs.

SKILLS

- Java, JavaScript, Python, C, SQL, HTML, CSS, Go, Swift, C++
- Databases & Cloud: MySQL, AWS EC2, AWS S3, Mephisto DB
- Web: Java Servlet, Node.js, HTML, CSS, React, Spring, Hibernate

PROJECTS

Agami: Live-updating & interactive visualization of streaming data. *[Spark, Streaming, Distributed System]*
- Implemented a Jupyter Notebook client for real-time querying of the steaming data and animated visualization of the queried streaming data with *PySpark* & *Altair*. Wrote scripts to compare the querying and plotting performance of running Spark with various worker count, thread count, and master machine combinations in a cluster.