Bixia Deng

San Francisco, CA

Email : bixiadd@gmail.com

Website: kayleedeng.com

EDUCATION

University of California, Irvine — Irvine, CA
Bachelor of Science in Computer Science GPA: 3.84/4.0 — *2019-Present*

City College of San Francisco — San Francisco, CA
Associate Degree for Transfer in Mathematics GPA: 3.80/4.0 — *2017-2019*

- **She Who Codes Club Treasurer**:
  - **Events**: Organized technology field speaker events on campus and participated in SF Hacks and Flutter WWC hackathons and conferences (e.g., Oracle OpenWorld 2018, Developer Weeks).
  - **Club Webpage**: Implemented club webpage with information about past events and contact information, meeting time, and mission of our club. Website: https://shewhocodesccsf.github.io/club/

PROJECTS

Fabflix — Full Stack Web Application
https://fabflix.shop — *Apr 2021 - Jun 2021*

- **RESTful API**: Implemented the RESTful API with JavaServlet, Javascript with Ajax for client-side, MySQL database management, deployed web app on Apache Tomcat, and used AWS and GCP instances as servers

- **Android and Search Integration**: Integrated to Android platform using Java, App and Web include fuzzy search, full-text search, and auto-complete

- **Security**: Enabled HTTPS, encrypted password and reCATPCHA

- **Scalability**: Improved application performance by using connection pooling with JDBC, Matser-Slave Replication, sticky session, and load balancer

Let's Fika — Full Stack Capstone Project
https://letsfika.today/ — *Jan 2021 - Jun 2021*

- **MERN Stack Development**: Implemented the RESTful API with NodeJS and ExpressJS, MongoDB Atlas for database management, and deployed on Heroku and Netlify

- **React**: Implemented front-end with React for both user flow and administration content management

- **OAuth2.0**: Pull contents from Spotify and Youtube, use OAuth2.0 authorizations for calling their API

Pixel Jump — Machine Learning Project
https://bikaylee.github.io/Pixel-Jump/final.html — *Oct 2020 - Dec 2020*

- **Deep Reinforcement Learning**: Jumping game simulation using Malmo. The agent learns from the reward system based on its actions. The environment is difficult with enormous action space in which the agent can choose the initial velocity and degree from a range of continuous data points.

- **Proximal Policy Optimization**: Trained the agent with PPO algorithm that makes updates based on the transitions that were obtained by the current policy and is used in the agent's decision for better performance

- **Classical Mechanics**: Based on the 3D projectile motion calculation to simulate jumps

EXPERIENCE

CCSF CS Department — San Francisco, CA
Teacher Assistant — *Aug 2018 - May 2019*

- **Python Tutoring**: Assisted 50 students with basic Python syntax

- **Java Tutoring**: Held office hours for tutoring students in Java fundamental concepts like inheritance and polymorphism

- **Assignment Grading**: Graded 200 student's homework assignments both in Java and Python courses

SKILLS

Languages: Java, Python, C++, C, SQL   **Technologies**: NodeJS, React, MongoDB, Flask, PPO, RLlib