Jorge Camarena

Software Developer | UC Berkeley Grad

• +1 510.904.2828 • camarena.jorge127@gmail.com •github.com/jorge-camarena
• linkedin.com/in/jorge-camarena • https://jorge-camarena.github.io/

PROFILE

Graduating senior majoring in Data Science with a concentration in Applied Mathematics & Modeling. Ambitious and versatile developer seeking a full-time job / internship in a software engineering role involving data-driven and impactful projects.

Education

University of California, Berkeley
Berkeley, CA
B.A. Data Science

Technical Skills

Programming Languages

Python, Java, C, C++, C#, Golang, Swift, Objective C, SQL, MatLab, Bash/Shell, HTML, CSS, Javascript

Frameworks & Technologies

Node.js, React.js, Angular.js, GoGin, Django, .NET/ASP.NET, Docker, Git, Numpy, Pandas, Scikit-learn, Scipy, PyTorch, MatPlotLib, PostgreSQL,

Open Source Contributions

Twelve Data Client API

- Created and actively maintain a .NET/C# Client package for TwelveData API
- Client allows user to fetch important stock data from twelvedata.com in an efficient and concise way
- Published Package on NuGet.com

Projects

PacMan Reinforcement Learning **Python** · 2022

- Utilized methods and algorithms such as policy iteration, policy extraction, value iteration and Q-Learning to train a PacMan agent to optimize the actions it takes to win the game

Encrypted File Sharing System **Golang** · 2022

- Designed the client and server of a secure file sharing system that allows user to create, append, share files with other users, and revoke access previously granted.
- The file sharing system was implemented to support the following security guarantees: confidentiality, authenticity, and integrity.
- The underlying encryption is proven to be robust against a variety of attacks (under the assumption that the data storage servers are untrusted and potentially “malicious”)

Stock Market Management System **ASP.NET** · 2024

- Designed and implemented a fully functional back-end API for managing and monitoring stock market portfolios for users to try and test different trading strategies.
- REST API fully supports creating accounts, portfolios, “buying” stocks, and endpoints to assess their performance over time.
- Used industry level MCV architecture for high-scalability and ease-of-use micro-service invocation, and fully Dockerized

Bear Maps **Java** · 2019

- Built a simplistic form of Google Maps for the city of Berkeley, CA
- Supports map rastering with the appropriate image resolution based on window size of browser
- Supports routing: given a source and destination, gives detailed instructions on how to get there in the shortest route possible.
- Implemented using A* search algorithm, along with the appropriate data structures (such as min-heap priority queue, kd-trees, etc) for computational efficiency