

Avery Rubin

averubin1@gmail.com • <http://www.averyrubin.com/> • <https://github.com/ave233>

EDUCATION

University of Michigan, College of Engineering, Ann Arbor, MI
Bachelor of Science in **Computer Science Engineering**

Relevant Coursework:

- Operating Systems
- Computer Security
- Data Structures & Algorithms
- Web Databases & Information Systems
- Computer Organization
- Programming and Data Structures

LANGUAGES

Proficient in: Python, Java, C++, JavaScript, HTML & CSS, C, Swift, Matlab, & Json

WORK EXPERIENCE

ExodusPoint Capital Management - Software Engineer **Oct 2019- Present**

Exodus Broker Matching – Lead Developer

- Drove the design and development of the funds broker matching and state engine. This is a multi-faceted application which listens to trades in real time that are ready to be sent to our prime brokers for reconciliation, manages the matched state of each trade, send's transformed trade objects to our 3rd party matching platforms (Alpha Omega, Traiana, Omgeo, Icelink) via the FIX engine I built or csv file drops over SFTP, and manages responses.
- Built a Web UI for Operations analysts and traders to view the current state of their trades via a live ticking table as well as a drilldown page for viewing individual trade details such as the state history of each trade, and all FIX messages/Files we sent or received on a trade. Once a match is received, the application flips the status of the trade in the traders portfolio or shows a reject if necessary.
- Tech Used: Java, Python, Docker, Kafka, SQL, React JS, FIX, System Monitor

Fixed Income Firm Wide Monitoring – Lead Developer

- Led the development of funds Fixed Income pre-trade, Risk Survey(RS) monitoring solution. When a new portfolio manager(PM) is onboarded to the platform as a trader they meet with the risk team to develop a RS which outlines all the Instruments they are allowed to trade along with other, more “trade specific rules”
- The application serves as a middle step between traders executing trades, and exchanges or brokers receiving them. When a trade is received it must verify that the trade is within the PM's RS with as little latency as possible and reject any trades that are not within the RS.
- Tech Used: Python, Docker, Kafka, SQL

XSP – Lead Developer

- Designed and built the funds automated Corporate Action(CA) Events Processor. The Application feeds a web UI with real time position, instrument, and Bloomberg CAX information showing traders what positions have eligible CA's available (Stock Split, Merger, Spin Off's, Tender's, etc..)
- Once Traders have made elections on their positions, the application houses all business logic to book the necessary trades directly into the traders portfolio for the new position to reflect the CA. Summaries of the trades booked are then sent off to brokers to match off with
- Tech Used: Java, Python, Docker, Kafka, SQL, React JS, Bloomberg

Goldman Sachs - Software Engineer

Jul 2018 – Oct 2019

MAST STP Project – Lead Developer (Securities Division)

- Spearheaded the development and architecture of a custom Straight Through Pipe (STP) for future/bond spread orders placed by agency traders on the Multi Asset Spread Trader (MAST) software developed by Broadway Technologies.
- Built a live ticking web application, off the back of a FIX drop copy feed with Broadway, where the traders were able to monitor their open orders, modify bond pricing, attach client specific customizations, allocate into sub accounts/book trades automatically, and send fills/allocations to clients over FIX messaging.
- Tech Used: RabbitMQ, FIX, quickfix, java, angular, Hbase, Google guice/Protobuf, dynamic relational algebra (DRA) tables.

INDEPENDENT PROJECTS

BookieBot Sports Betting Software – Software Engineer

Sep. 2018 - Present

- Built a web scraper in python using Selenium to monitor an “open bets” page so that every time a new bet is placed, it automatically records and places that bet on as many other sites as specified to save time and get the same betting line as the original bet.

Algorithmic Crypto Trading – Software Engineer & Developer

Oct. 2017 - Present

- Built a Crypto Currency Arbitrage scanner. Takes in as input Crypto market data and the list of Currencies to scan, transforms it into an internal model, and runs the Bellman Ford Algorithm to try and detect negative weight cycles. Open sourced on my Github

MHybrid Racing – Computer Science Engineering Team

Sep. 2014 - Jun 2016

- Programed the Micro-controller that ran the hybrid system of the race car in C++.
- Through the use of torque vectoring, I was able to improve the handling of the car in corners as well as while cruising in long distance races.