Matthew Rhea

559-300-8930 | m.b.rhea31@gmail.com | https://github.com/Mrhea

Education

University of California Santa Cruz June 2020

Bachelor of Science, Computer Science

Work Experience

Research Assistant *Remote*

UC Santa Cruz Formal Methods Lab June 2020 – Present

- Implemented 3D geometry support in Scenic, a probabilistic programming language for perception systems.
- Extended Scenic's compiler and runtime in Python to support new syntax and semantics.
- Integrated 3D simulators by generalizing Scenic's API to support new types and classes for simulator models.
- Collaborated with other research groups who use Scenic in order to incorporate features they need.

Software Developer *Remote*

Center for Research in Open Source Software June 2020 – Present

- Built a client-layer SQL processing Python application for Skyhook, a data management system that extends Ceph.
- Assisted researchers' analytics by serializing Apache Arrow, Google FlatBuffer, and CSV data frames from queries.
- Aided in the design of a Skyhook SQL dialect used to set custom options for querying.
- Worked on C++ libraries that defined the Skyhook API for applications by abstracting underlying binaries.

Research

Undergraduate Researcher *Remote*

Cornell Programming Language Group January 2020 – May 2020

- Worked on Cornell's geodistributed programming language, Gallifrey.
- Set up distributed testing infrastructure on Linux servers for unit testing Gallifrey's backing store, AntidoteDB.
- Diagnosed and fixed asymmetric network interfaces and data replication errors to prevent lost transactions.
- Researched Microsoft's Z3 theorem prover to verify non-conflicting, concurrent Java method execution.

Projects

Distributed Fault-Tolerant Key-Value Store (Golang)

- A REST-accessible distributed key-value store in Golang that provides causal consistency guarantees.
- Partitions data across multiple replicated shards in a cluster of Docker containers.
- Discovers disconnections and conflicting operations using a peer-to-peer gossiping protocol.

HTTP Server (C/C++)

- Built a multithreaded HTTP server that provides PUT, GET, and, PATCH HTTP request methods.
- Supports an in-memory MRU block cache for commands a client most recently used.
- Provides a name-mapping system to give aliases to objects stored on the server.

Technical Skills

Languages

- Python, C/C++, Golang, Haskell, Java, SQL.

Tools and Methods

- Docker, PostgreSQL, Git, VSCode, Vim, TCP/IP, REST APIs, Agile Development, Continuous Integration, Unit Testing.

Mathematics

- Decision Procedures, Property Testing, Computational Models, Complexity Theory, Algorithm Design and Analysis, Linear Algebra, Probability Theory.