Michaela Murray

murray22@cs.stanford.edu | 480-818-7165 | LinkedIn: linkedin.com/in/murray22/

Education

- **Stanford University** Stanford, CA
  Bachelor of Science in Computer Science *Sept. 2018 - June 2022 (Expected)*
  - **Relevant Courses**: **CS Systems** (Operating Systems, Distributed Systems, Computer Networking, Computer Security), **CS Theory** (Introduction to Algorithms, Randomized Algorithms, Error Correcting Codes, Logic Programming), **Math** (Linear Algebra, Group Theory, Probability and Statistical Physics, Abstract Algebra)

Experience

- **tockOS: Embedded Operating System in Rust** Stanford, CA
  Associate Professor Phil Levis, Stanford Information Networks Group *Jan 2020 - Present*
  - **Remote System Calls**: Developing Rust kernel and userspace code to support a distributed tockOS and creating protocols for securely invoking system calls between microcontrollers. Previously developed userspace applications which sent messages from a controller to a peripheral via SPI to invoke system calls on a different microcontroller.
  - **QDEC Peripheral**: Developed Rust kernel code to support usage of the QDEC (Quadrature DECoder) peripheral, which is present in Nordic nrf5x boards.
- **STEP Intern** Mountain View, CA (Remote)
  Google *June 2020 - September 2020*
  - **UniViz**: Open-source STEP final project focused on creating a platform for potential college students to research information on colleges of interest. Constructed an end-to-end autocomplete feature for the landing page of the STEP Intern Final Project. Used HTML5/CSS3/JS webstack to create a front-end for users to query a REST API with college names. Created a backend in Java which served up autocomplete values in real-time. Created tests for both the front-end and back-end. Used Maven appengine to display and deploy the page. **Github**
- **Esper Video Analytics Project** Stanford, CA
  Associate Professor Kayvon Fatahalian, Stanford Computer Science Department *June 2019 - December 2019*
  - **Clustering Pipeline**: Used Python to create a clustering pipeline with two steps: The first step is a machine learning model that uses k-means clustering to create n clusters of similar pictures of a particular individual. The second step is manual clustering, where a human merges, deletes, or splits clusters accordingly. This step further vets the clusters and ensures the accuracy of the data being displayed.
  - **Data Processing and Visualization**: Used both Matplotlib and Javascript interfaces to display results of the clustering in a human readable form. The interface currently uses a combination of bar charts, scatterplots, heatmaps, and tables to depict the data.
- **Radar System Analysis** Stanford University
  Associate Professor Dustin Schroeder, Stanford Radio Glaciolgy Lab *Jan. 2019 - September 2019*
  - **Software Defined Radio (SDR) Bandwidth Analysis**: Performed a series of analyses on B-200 mini-i SDR to determine whether the advertised and observed bandwidths of the device were the same. Results determined the bandwidths differed, with the experimental bandwidth lower than the advertised one. This analysis was part of a larger investigation to determine the usability of SDRs for radioglaciology research.

Workshops and Publications

- **arXiv 2020**: Analyzing Who and What Appears in a Decade of US Cable TV News. James Hong, Will Crichton, Haotian Zhang, Daniel Y. Fu, Jacob Ritchie, Jeremy Barenholtz, Ben Hannel, Xinwei Yao, **Michaela Murray**, Geraldine Moriba, Maneesh Agrawala, Kayvon Fatahalian.
- **International Glaciology Symposium (IGS)**: Presented the poster *Producing Multi-Decadal Observations of Grounding Line Change in East Antarctica with Archival Radar Data* at the IGS poster sessions
- **CURIS Poster Session**: Presented the poster *Do the Images Used by Cable TV News Channels Introduce Bias?* at Stanford's Computer Science Undergraduate Research Internship (CURIS) Poster Session

Programming Skills

- **Languages (Ordered by Experience)**: C++, C, Rust, Python, Java, JavaScript