

# Jordan Fleming

github.com/JordanPCF ■ jordanpflem@gmail.com

**Summary:** Quick-learning and highly analytical. Looking to leverage my experience building data pipelines as a physicist and my passion for implementing software design patterns in my personal projects.

---

## EDUCATION

### University of California, Berkeley

May 2019

- M.A. Applied Physics (Astrophysics)
- GPA: 4.00/4.00

### Duke University

May 2017

- B.S. Physics
  - GPA: 3.86/4.00
  - CS Coursework: Data Structures and Algorithms (using Java)
- 

## WORK EXPERIENCE

### PhD Research Scientist, UC Berkeley

August 2017 – June 2020

- Streamlined data reduction and analysis by upgrading a pipeline from IDL to Python
- Reduced uncertainty in calculations by 90% using Markov Chain Monte Carlo (MCMC) sampling
- Minimized the impact of irregular sensitivity variations in the telescope's hardware with decorrelation models
- Optimized performance tradeoffs such as minimizing photon vs 'red' noise
- Improved model reliability by jointly fitting data sets from 2 telescope surveys that differed in sampling rates and observed wavelength

### Consulting Intern, Jaffe Holden Acoustics

May – July 2016

- Created the firm's first database of acoustic wall, ceiling, and roof systems in Excel
  - Developed firm-wide organization standards for data management
- 

## PROJECTS

### iOS Mobile Application with Serverless Backend, *personal project*

Search song lyrics within Spotify playlists

- Created a low-latency API to access Python back-end code with Amazon API Gateway Lambda proxy integration
- Coordinated asynchronous tasks with WebSocket API routes and AWS Step Functions
- Achieved consistent search efficiency with parallel search methods optimized for different use cases
- Expanded functionality by integrating third-party REST APIs
- Currently building the user interface in Swift

### Web Application with 3-D Data Visualization of Minecraft Diamond Locations, *personal project*

- Created a data dashboard with an interactive 3-D visualization of user-provided data
- Reduced the number of database queries by pre-joining relational data in a single NoSQL table
- Locally developed and tested the database in a Docker container
- Utilized: JavaScript, HTML/CSS, Amazon DynamoDB, Docker, Three.js, dat.GUI

### Computational Model of Theoretical Astrophysics Problem, *Master's coursework*

- Improved agreement between a simulation and data by expanding the analysis of the magnetic field
  - Derived equations of motion and computationally solved differential equations with the Runge-Kutta method
  - Investigated the effects of 9 degrees of freedom on the simulation results
- 

## SKILLS

**Proficient:** Python, Amazon Web Services (Lambda, API Gateway, DynamoDB, Step Functions, Simple Queue Service, Amplify, CloudWatch Logs), Git

**Familiar:** JavaScript, jQuery, AJAX, HTML, CSS, Bootstrap, Docker, SQL, Swift/Xcode, Bash Scripting, C