

# MICHAEL SPARRE

michaelsparre@berkeley.edu  
linkedin.com/in/michaelthorsparre  
github.com/micsparre

/\* Looking for a full-time position in a full-stack Software Engineering role \*/

---

## Education

University of California, Berkeley

Graduated May 2022

- B.S. Electrical Engineering and Computer Science

GPA: 3.8

---

## Skills

**Programming Languages:** C, Go, Java, JavaScript, Python, SQL, Typescript

**Other:** CSS, HTML, Git, React, PyTorch, AWS: Lambda, API Gateway, S3, CloudFront, QLDB

---

## Experience

**Full-Stack Software Engineer**, Gilead Sciences

Aug. 2021 - Current

- Developing PySpark ETL scripts to interact with AWS Glue to transform Master Data sources
- Syncing data between on-prem and cloud data sources with Python ETL scripts
- Researching blockchain use cases with AWS QLDB technology as a back-end/auditing-edge database

**Software Engineer Intern**, Gilead Sciences

May 2021 - Aug. 2021

- Developed internal web apps in React using AWS: Lambda, API Gateway, S3, CloudFront services
- Architected a free, private component library as an alternative to the npm registry to improve code reusability across the company and promote easier access to code documentation

---

## Projects

**ANikan**, [link](#)

Apr. 2022 - Current

- Utilizing an anime API to develop a web app that finds anime shows/movies
- Building features that allow you to track current/past shows watched and recommend what to watch next based on previous selections

**Facial Keypoint Detection**, [link](#)

Oct. 2021 - Nov. 2021

- Engineered a neural network with PyTorch on Google Colab to identify facial keypoints on personal photos
- Trained a model with >6000 images, utilizing data augmentation and hyper-parameter tuning to increase validation accuracy

**NP Problem Contest**, [link](#)

Apr. 2021 - May 2021

- Worked in a 3-person team to produce the best outputs for a NP hard problem, placing 29th out of 239 teams
- Implemented an algorithm that ran 4 unique, greedy algorithms on graph inputs and used parallelism to run the algorithms on 4 separate cores to speed up runtime

**NumC**, Great Ideas in Computer Architecture

Nov. 2020 - Dec. 2020

- Extended matrix operations written in C to a Python-C interface
- Accelerated efficiency of matrix multiplication by 125x and exponentiation by 1900x with SIMD, OpenMP, cache efficient memory structure, loop unrolling, and repeated squares exponentiation

---

## Leadership

**Mens Club Soccer Coach**, UC Berkeley

Aug. 2021 - May 2022

- Lead a 30-man team with 2 practices a week and 1 game a week
- Handled logistics and training sessions of a highly competitive soccer team