Connor Bernard

Linkedin: https://www.linkedin.com/in/connorbernard/

Github: https://github.com/Connor-Bernard

EDUCATION

University of California, Berkeley

Berkeley, CA

Expected May 2024

Mobile: +1-650-714-4346

Email: connorbernard@berkeley.edu

Bachelor of Computer Science; GPA: 3.93 Courses: Data Structures (CS61B), Efficient Algorithms and Intractable Problems (CS170), Machine Structures (CS61C), Intro to Databases (CS186), Computer Security (CS161), Web Design (CS198), Discrete Mathematics and Probability (CS70)

TECHNICAL SKILLS

- Languages: Python, Java, C, HTML, CSS, JavaScript, PHP, Objective-C, SQL, Unix scripting, Lisp, LaTeX, Markdown
- Tools: GIT, JIRA, XCode, Postman, phpMyAdmin, mySQL, Linux, Github, BitBucket, MAMP, Vim, Figma, WordPress
- Soft Skills: Communication, Self-Awareness, Determination, Time Management, Teamwork, Adaptability, Creativity

EXPERIENCE

Emotewell Inc.

Software Engineering Tech Lead

Berkeley, CA

May 2022 - Present

- Team Lead: Mentored and managed a team of junior developers working on back-end, web-development, and mobile-development through bi-weekly team line-ups and weekly individual meetings to strategize agile approaches to
 - outstanding tickets and pull requests. • REST API Development: Developed a company-wide REST API using Postman and Cloudflare DNS with endpoints for mobile In-App-Purchase verification, server-side user identification and validation, natural language processing (NLP) and
 - Machine Learning (ML), and front-end platform customization. • Website and Mobile Optimization: Decreased website load speed by over 53% and increased maximum server load by over 35% by implementing caching and compression techniques to fit complex computational models generated through load time analysis and offensive load testing.
 - Mobile App Development: Used Xcode and Objective-C to develop and push the first version of the Emotewell App with full website functionality in under two weeks.

Emotewell Inc.

Berkeley, CA

Software Engineering Intern

May 2021 - May 2022

- Website Redesign: Used CSS, HTML, JS, PHP, and Figma to implement and handle a full scale website redesign event.
- SEO Optimization: Applied statistical modeling and advanced regression techniques to increase SEO score by 30%.
- Website Asset Optimization: Implemented loading algorithms such as Lazy-Load, asynchronous and delayed JavaScript loading, and image compression.

Computer Science Tutor

AP CS, CS61A, CS61B, CS61C, CS170

San Mateo, CA | Berkeley, CA

May 2019 - Present

o Tutor and Review Session Host: Tutored peers in one-on-one environments and led several full-class exam-prep sessions on advanced algorithms, data structures, and complex run-time analysis as well as co-taught several classes.

Academic Projects

- Streamlined Git-style VCS: Used Java to create a version-control system that mimics some of the features of Git including active directory moderation, file hashing, merge conflict handling, and remote repository interactions.
- Enigma Machine: Applied advanced applications of data structures to redesign the Enigma machine used in WWII with fully functional file encryption and decryption algorithms.
- Scheme Interpreter: Used the functional programming language Scheme (a Lisp dialect) in conjunction with Python to write a pseudo-self interpreter and Read-Eval-Print Loop (REPL) environment for Scheme.
- Italian Restaurant Website Redesign: Used HTML, CSS, JS, and PHP to create an updated website for a popular restaurant in Berkeley using data structures, prototype design techniques, and HTTP requests to dynamically populate menu items.
- Snake (Game): Used C to recreate the popular game Snake with added functionality for parallel snake instances and a fully operational command-line UI.

Clubs and Affiliations

- The International Honor Society for the Computing Sciences (UPE): Members selected from the top 20% of CS students.
- Computer Science Undergraduate Association (CSUA)
- Institute of Electrical and Electronics Engineers (IEEE)
- National Society for Leadership and Success (NSLS)