Evan Guan

Junior Software Engineer
San Francisco, CA 94121
evan.guan.sf@gmail.com
+14159398027

SUMMARY

Hello, my name is Evan. I am a recent SFSU Bachelor in Computer Science graduate.

I…

- consider myself a generalist and fast learner and am able to get up to speed quickly in most subjects and topics.

- am capable of learning and using many programming languages and development technologies.

- have experience with scrum and agile team-based UX-focused software development in a simulated environment for a class taken at San Francisco State University over the course of summer 2019.

- have a strong interest in computer software, hardware, and technology, especially where video games are concerned.

Information on what I consider to be my "best" projects can be found on my website: evanguan.ddns.net

For more information please do not hesitate to contact me directly. I am authorized to work in the US for any employer.

SKILLS

- **C**
- **C++**
- **C#**
- **Java**
- **JavaScript**
- **jQuery**
- **Python**
- **GitHub**
- **HTML**
- **CSS**
- **Bootstrap**
- **Game Design and Programming (C#, Unity)**
- **Agile/Scrum Development Methodology**
- **Amazon Web Services**
- **Node.js**
- **MySQL**
- **Winforms**

EDUCATION

Bachelor's in Computer Science

San Francisco State University - San Francisco, CA August 2017 to August 2019
Notable courses taken:

- **CSc 415: Operating System Principles:**
  - SFSU Description: Operating system concepts: concurrent processes, basic synchronization techniques, deadlock, memory management, file systems, security, networks, and distributed processing.
  - Course assignments included creating basic file copy and shell programs. Also created programs to both generate and safely handle deadlock and producer/consumer scenarios across multiple threads. C with POSIX threads (pthreads).

- **CSc 510: Analysis of Algorithms I:**
  - SFSU Description: Notions of main algorithm design methods. Measures of algorithm complexity in space and time. Algorithms of classic problems including sorting and scheduling and complexity analysis of such algorithms.
  - Explored topics such as complexity functions, complexity categories, limit test ratios, recurrence equations, shortest-path algorithms (Dijkstra, Prim's), and dynamic programming in the pursuit of efficiency.

- **CSc 600: Programming Paradigms and Languages:**
  - SFSU Description: Concepts for high-level programming languages. Procedural, logic, functional, and object-oriented programming paradigms. Comparative study of several languages and an introduction to grammars and parsing techniques.
  - Learned about and utilized languages including Fortran for procedural programming, Prolog for logic programming, Scheme (Racket) for functional programming, and Ruby for multi-paradigm programming (focus on object-oriented programming).

- **CSc 631: Multiplayer Game Development:**
  - SFSU Description: Computer graphics and network characteristics of multiplayer games. Design and development of a game as a team project.
  - Course consisted primarily of a group project that continued development of an already-existing student game project (Sea Divided). Primary goals accomplished included the modifying and expanding of the core game with new game mechanics, art, and audio. Coordinated with sister course SFSU MUS 453 (Scoring for Games II) members to overhaul and replace the sound effects and music with new effects and music.

- **CSc 648: Software Engineering:**
  - SFSU Description: Practical methods and tools for SW engineering including organizational teamwork.
  - Investigated methods, reasoning, and practical solutions to problems faced in organized software engineering such as inter-team communication, goal/milestone planning, and development tracking. The primary focus of the course was using the solutions taught in class in a group class project that utilized scrum and agile software development techniques to create an environmental hazard reporting site using Amazon AWS, Node.js, Bootstrap, MySQL, Google APIs, and other technologies.
    - Personal responsibilities as a front-end developer in the team included designing the use cases and personae for the users of the website along with prototype website layout. Programming work includes Google API integrations and functionality including Maps, Analytics, and reCAPTCHA. Additionally worked on and added form validation to both the front and back end to filter out bad user input.
    - Outside of twice weekly meetings, communications including general group text chat, voip, and information sharing were facilitated by Discord (a Slack analogue) while longer term goals and individual tasks and roles were handled on Trello (a Jira/Asana standin).

- **CSc 665: Artificial Intelligence:**
  - SFSU Description: Overview of algorithms and approaches central to artificial intelligence. Study of fundamental concepts needed to attain human-level intelligence in computer systems, and gain experience in working with these concepts through assignments and programming exercises.
  - Learned about the basics of machine learning, including where, why, and how it is used. Utilized Jupyter Notebook and Python to create and test basic machine learning code from instructor provided code outlines, including random forests, decision tree classifiers, perceptrons, and decision tree regressors.

- **CSc 698: Topics in Computing: Applications of Parallel Computing:**
  - This course was provided to multiple colleges and universities courtesy of UC Berkeley. Topics included an introduction to parallel programming, implementations of parallel computing, challenges to parallel computing, super computing, and examples of its applications. Practical assignments include parallelized CPU password breaking, Nvidia GPU CUDA enabled hashing, nearest-neighbor particle binning simulations with MPI, and implementation and comparison of GPU vs. CPU NxN matrix multiplication runtimes.

Associates equivalent credits in Computer Science

City College of San Francisco - San Francisco, CA August 2010 to May 2017
Notable courses taken:

- **CS 270: Computer Architecture with Assembly Language:**
  - CCSF Description: Basic computer organization including the instruction cycle, parts of CPU (Central Processing Unit), memory hierarchy including caching, pipelining, exception handling, and issues of multiprocessing. Write assembly language programs using a sample architecture. Translation from high-level code is discussed. Basic hardware components and control circuits are designed in logic using combinational and sequential circuits.
  - Course work utilized MARS MIPS to investigate, create, and debug assembly-level programs on a simulated cpu and memory. Final project was a number-guessing game written in MIPS with Java Swing IO.

LINKS

Portfolio site: http://evanguan.ddns.net

GitHub repositories: https://github.com/JammyPajamies

LinkedIn: https://www.linkedin.com/in/evan-guan-72b630194/

WORK EXPERIENCE

Cashier

Safeway Inc./Albertson's Inc - San Francisco, CA June 2017 to August 2017

- Handled returns and exchanges
- Handled cash and credit transactions
- Handled customer complaints and concerns
- Assisted with customer inquiries
- Helped customers find what they are looking for