|  |  |  |
| --- | --- | --- |
| Phillip Lagoc | | |
|  | San Jose CA ▪ (408) 547 – 7499 ▪ philliplagoc@gmail.com |  |

Team-player: I am working with a 10-person development team, completing tasks involving benchmarking processes and automating tasks together. When I was first hired, experienced lab members were not communicative and inclusive towards new lab members. As I became more experienced, I decided to make the work environment much more welcoming by ensuring new hires had a smoother transition period, taking the initiative to arrange and lead work sessions as well as host recreational events such as board game nights. Through this, I was able to boost productivity over a period of four months. I also took charge in improving communication with the Deputy Director and Director by documenting our progress into a collaborative slide deck, which had not been done when I was first hired. This allowed us to explain in better detail what we had been doing and let us gather our thoughts and concerns more effectively during meetings.

QA: As someone driven to be more time efficient, I employ test-driven development techniques with frameworks such as JUnit and GoogleTest to ensure that each moment spent coding is not wasted. I also strive to efficiently unit-test my code, achieving an average of over 80% in terms of the amount of code being properly tested in these programming projects. I have used these techniques when implementing multiple data structures and algorithms by scratch using Java and C++, such as Huffman trees and Dijkstra’s algorithm.

QA\_2: In several software development projects I have employed test-driven development techniques using JUnit and GoogleTest. For example, I developed a clone of the popular tile game *2048* using Java, unit-testing my implementation of over a dozen game logic functions with JUnit. These game logic functions included calculating the correct score and ensuring each tile’s values were merged correctly. I have also implemented several data structures, including binary search trees and Huffman trees in C++, using the GoogleTest framework to unit-test my progress.

SWE: **talk about coding games (2048, Frogger); implementing graph-search algorithms like A\*, BFS; implemented Huffman tree to compress and decompress files and multi-way trie for autocompleting words**

Data science: **talk about teaching COGS 109 which was about data analysis and modeling, and discussed the mathematics behind… and COGS 108 where I taught students about data visualization, gathering, preprocessing, and how to use standard DSC and ML packages like sklearn and pandas in Python**

ML: **talk about work at MSSL, including learning about weak supervision techniques; talk about participating in Kaggle challenges**

PostgreSQL: Always striving to expand my skillset, I decided to take an online introductory course and learn more about relational databases and how to query them using PostGreSQL After completing this course, I continue to practice these skills by taking online SQL challenges on LeetCode.

Example introduction: After speaking with David In about Generation Veeva and the impact it can make on a recent graduate, I am writing to you as I believe that I would make a great candidate for the Associate Software Development Program.

Sincerely,

A close up of a logo

Description automatically generated

Phillip Lagoc