Daniel Qu

SOFTWARE DEVELOPER

+1-647-974-1176 | daniel.qu@uwaterloo.ca | danielq987 | danielq987

Skills

Languages	Javascript, Typescript, Python, C, C++, Bash, SQL, HTML, CSS, Rust
Frameworks	React, React Native, Redux, Express, Flutter
Tools	Git, Unix Command Line, Jenkins, REST API

Work Experience

Playstation - Sony Interactive Entertainment
FRONT END SOFTWARE DEVELOPER
Jan. 2022 - Present
React Native, Redux, Python, Jenkins

- Improved working efficiency for QA engineers by developing Python script to monitor status of over 2000 test cases.
- Achieved first prize in global company-wide hackathon by creating a calendar app for the Playstation.
- Supported development of Playstation checkout system by fixing React Native bugs and automating end-to-end tests.

Midnight Sun - University of Waterloo
FIRMWARE DEVELOPER
Nov. 2021 - Present
C, FreeRTOS, Python

- Migrated build system from GNU Make to Python Scons, improving extensibility of linting, formatting, and project setup.
- Ensured continuous voltage during pedal board operation of solar car by writing unit tests in C.

Dreamschools
FULL STACK SOFTWARE DEVELOPER
May. 2021 - Aug. 2021
React, Express, PostgreSQL, Stripe

- Overhauled checkout flow by redesigning React UI, supporting more payment methods, and migrating away from deprecated API's.
- Improved order history API response time by 10x by parallelizing requests in Express and leveraging Stripe API features.
- Designed and implemented complete and flexible notification system with the help of GetStream API.
- Developed an email/SMS communication system between organizations and students using Twilio and PostgreSQL.
- Took initiative to improve code quality by extracting reusable components and reorganizing file structure.

Projects

Hexle - Game - Github
GROUP PROJECT
Feb. - Mar. 2022
TypeScript, React, CSS

- Contributed to hex code guessing game with returning users from over 40 countries worldwide.
- Refactored initial code and React state management to improve maintainability and readability.
- Enhanced UI across devices by using responsive CSS and implementing a "side-by-side" colour comparison feature.

Midnight Sun Driver Dashboard - Github
MIDNIGHT SUN SOLAR CAR STRATEGY TEAM
Apr. - Sep. 2021
Flutter, Dart, Python

- Built digital dashboard for Midnight Sun's solar vehicle, including speedometer, battery life, target speed, and more.
- Performed CAN integration testing using mock CAN messages and Python driver code.
- Effectively used Flutter animations to direct driver attention when appropriate.

Solving SuperGhost - Github
PERSONAL PROJECT
Oct. 2020 - Jan. 2021
ExpressJS, Socket.io, PostgreSQL, Python

- Solved word game 'SuperGhost' algorithmically by constructing 400,000-node directed graph.
- Created a prototype for real-time multiplayer SuperGhost using Javascript and Socket.io.
- Developed a backend server and database to support shareable game codes and game rooms.

Education

University of Waterloo - 92% GPA
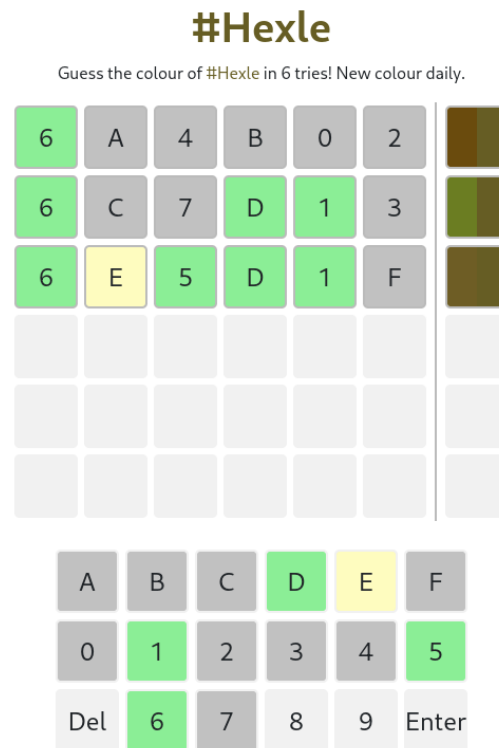B.A.Sc. IN HONOURS MECHATRONICS ENGINEERING
Sep. 2020 - Present
Waterloo, Canada

- Computer Structures and Real Time Systems, Microprocessors and Digital Logic, Data Structures and Algorithms (C++)

Extra: Project Demos

Visit https://tinyurl.com/daniel-qu-portfolio for short video demonstrations.

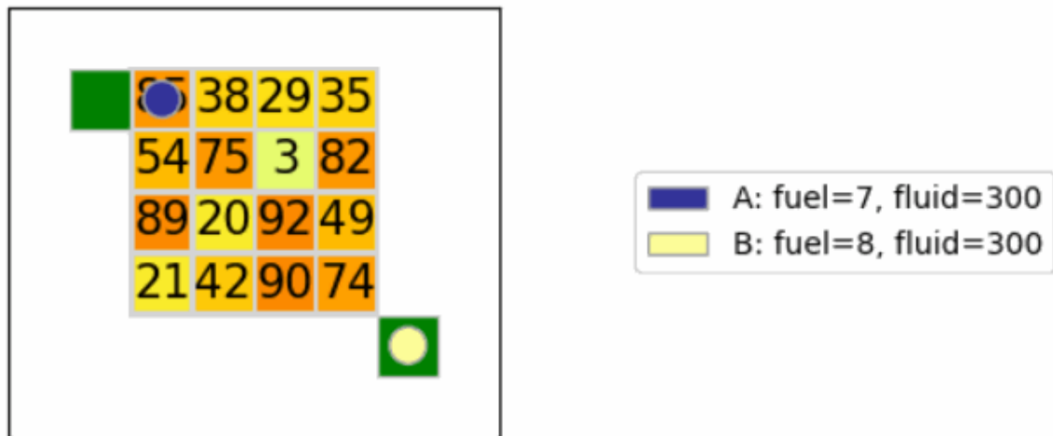**Hexle - Wordle Variant (Typescript, React, CSS)**

- Try the game out at frankljin.github.io/hexle!
- Implementing CSS animations on each guess in the near future for UI polish!

Midnight Sun Driver Dashboard (Flutter)

- Display used as dashboard for Midnight Sun's Solar Car during an international solar car competition.
- Shows a subtle animation to alert driver when recommended speed changes.
- Used gradients effectively to improve UI and meet design specifications.

Waterloo Engineering Competition 2020 - Programming Division (Python)

- Solved programming optimization problem with team, involving giving directions to cleaning robots.
- Carefully considered and developed algorithm to choose robot starting locations (green tiles).
- Maximized competition score by maximizing tiles cleaned, minimizing fuel, fluid, and number of robots used.

Spotify-Track (TypeScript)

- Created this quick Node script since Spotify does not officially support an API for friend listening activity.
- Used open-source API wrapper to track and store friend listening histories in MongoDB.