JinLin Cen

437-989-1893 | jinlin.cen@mail.utoronto.ca | https://github.com/simoncen

OBJECTIVE

- Extreme passionate full stack developer.
- Full stack project and game design program from the ground up – design, development, unit testing, delivery and maintenance;
- Experienced in app and game development tools with MERN stack.

PROJECT EXPERIENCE

To-do App Full stack Developer
April 2020 – August 2020

https://github.com/simoncen/todo-app.git

- A **MERN** built Trello-like mini to-do list full stack website application.
- Developed the **React**-built frontend with features that allowed users to create their own to-do lists after they logged in their own user accounts with the use of **passport-jwt** for authorization and authentication through **JSON Web Token**.
- Developed **Rest APIs** with **Express.js** and tested them with **Postman.**
- Developed the database with **MongoDB** and managed user's to-do lists' data
- Applied the components from **Material-UI**, a React UI framework, to assist the front-end development of the website.
- Utilized **Mongoose** library for **Node.js** to access MongoDB.

Programmer for Reversi-Game
Feb. 2020 – Mar. 2020

https://github.com/simoncen/Reversi-Game.git

- A Reversi game individual coding project developed in **C language**.
- Developed a Reversi game that could be played on the Terminal with features that allowed users to play the game after they configurate the dimension of the board and choose their chess color.
- Applied the **Greedy Algorithms** which looked for the tile with the highest score to place at according to calculation backed by research.
- Beat professor's "smarter" AI program on an 8 by 8 board in a best of two game:
  49 vs. 15 pieces, 36 vs. 28 pieces.

Programmer for Tic-Tac-Toe Game
Sept. 2020 – Oct. 2020

- A Tic-Tac-Toe individual coding project developed in **C++ language.**
- Developed a tic-tac-toe game that could be played on the Terminal with features that reminded players about the state of the game including the validity of the move, the game-over state, and the wincode of the game.
- Utilized the **Model-View-Controller** design pattern to have the **game controller** to interact with the **game logic** using the **game state** and display the game to the user on Terminal.

TECHNICAL SKILLS

- Computer Language Skills: C, ES6 JavaScript, C++, Java, Python, HTML, CSS, MATLAB, Verilog
- Frameworks/Libraries: Express.js, React, Redux, Material UI, Mongoose, Validator.js, node.bcrypt.js, jwt-decode
- Database/SW Tools: MongoDB, Node.js, Postman, Heroku, Google Analytics, Git, Namecheap, Passport.js, body-parser, CORS

EDUCATION

University of Toronto — **Toronto, Canada**

Bachelor of Applied Science in Computer Engineering — Expected Graduation: April 2024