# EDUCATION

**University of California, Irvine** *Sept 2018 - June 2022*

Bachelor of science in Computer Science

Bachelor of science in Computer Game Science

GPA: 3.84

# TECHNICAL SKILLS

**Languages** HTML, CSS, JavaScript, Typescript, Python, Java, C#, C, C++, SQL

**Frameworks** Node.js, React.js, React native, Jasmine.js, Spring boot

**Tools** Git, Visual Studio Code, Visual Studio, Unity

**Systems** Windows (10,7), Linux (Ubuntu)

# PROJECTS

**Fabflix** *May 2022*

Mock movie storefront where users can search and purchase digital copies of movies.

Libraries/tools used: Java, SQL, JavaScript, ReactJS, React native, Stripe, Spring boot

* Set up and developed the Fabflix back end via the process of Test Driven Development.
* Performs 2-5 queries in the backend per search to populate movie information in the frontend.
* Utilized the Stripe credit card vendor API to imitate payment functionality.

**Object recognition project** *March 2022*

Object detection software that can be trained with images to detect and match objects in other images.

Library/tools used: Python, NumPy, Matplotlib, SciPy, Juptyer Notebook

* Implemented object recognition with image processing, using HOG (histogram of oriented gradients).
* Create a template for the software to match by passing in positive and negative training images.

**Checkers AI** *December 2020*

This AI simulates and backtracks via the use of search trees to play checkers.

Libraries/tools used: C++, CMake

* Utilized Monte Carlo tree search, and backtracking to empower the checkers AI to make good moves.
* Improved AI’s effectiveness by increasing its simulations per turn from 80 simulations to 1000 simulations.

**DiscordActivityBot** *October 2020*

Discord Bot to track and notify user of their activities and the server’s activity.

Libraries/tools used: Discord.py, Matplotlib, SQLite, APScheduler, pytz, asyncio.io

* Co-developed the bot alongside a fellow student and hosted the bot on AWS for 5-6 months.
* Deployed a Discord bot that scraped fellow server occupants’ online activities in a server of 30-40 people.
* Stored hashed data to SQLite database which allowed the bot to retrieve the users’ activity later on.