Matt **Gottlieb**

Software Engineer · Computer Science B.S.

(818)-325-7776 | mattgottlieb96@gmail.com | www.mattgottlieb.me | Mgla96 | matt-gottlieb

Education

University of California, Santa Barbara — *Santa Barbara, CA*

B.S. in Computer Science — *Class of 2020*

- Relevant Courses: Advanced App Programming (CS56), Mobile Application Development (CS184), Introduction to Computational Science (CS111), Data Structures and Algorithms I & II (CS130A/CS130B), Computer Communication Networks (CS176A), Human-Computer Interaction (CS185), Operating Systems (CS170), Fundamentals of Database Systems (CS174A), Artificial Intelligence & Machine Learning (CS165A/B).

Experience

Happy Cows — *Santa Barbara, CA*

Lead Software Engineer | github.com/Mgla96/HappyCows — *Jan. 2020 - Apr. 2020*

- Implemented a multiplayer simulation game web application which is used in Chem 123 classes for Professor de Vries at UC Santa Barbara as an interactive learning tool for his students. This application uses Node.js for the backend with the assistance of the Express.js framework to handle routing and middleware, mySQL for the database, and Embedded Javascript for the Front-end.

Ocean Recoveries Lab — *Santa Barbara, CA*

Software Engineer | github.com/Mgla96/OceanRecoveryLabScripts — *Jan. 2020 - Mar. 2020*

- Created Python scripts to assist researchers at the Ocean Recoveries Lab at UC Santa Barbara. These scripts are used with Metashape to automate the process of converting datasets of photos researchers have taken of coral into accurately scaled 3D models in order to quantifiably measure the growth or deterioration of coral reefs.

Gomoku AI — *Santa Barbara, CA*

Artificial Intelligence | github.com/Mgla96/GomokuAI — *Jun. 2020 - Jun. 2020*

- Developed an AI in Python to play Gomoku. This AI uses the Minimax algorithm and a custom point evaluation system to determine the best move. It incorporates Alpha-Beta Pruning and an optimized method of searching in order to reduce computation time.

Fitness Equipment Scraper — *Oak Park, CA*

Web Application | github.com/Mgla96/GymEquipmentFinder — *Jul. 2020 - Jul. 2020*

- Constructed a Python Flask web application hosted on Heroku that utilizes cron jobs to scrape equipment information with Beautiful Soup from popular fitness websites and store them in a Postgres database. Due to COVID-19, fitness equipment is hard to find, so this application displays all available equipment in one place to save people time in their search.

Author Predictor — *Santa Barbara, CA*

Machine Learning | github.com/Mgla96/AuthorPredictor — *May. 2020 - May. 2020*

- Assembled a Machine learning Multinomial Naive Bayes Multiclass classification model with a bag-of-words text representation to predict the author of an anonymous article. Used lemmetization, bigrams, and trigrams as methods to improve accuracy.

Diabetic Retinopathy Predictor — *Santa Barbara, CA*

Machine Learning | github.com/Mgla96/DiabeticRetinopathy — *May. 2020 - May. 2020*

- Manufactured a model that predicts whether a patient has a medical condition known as Diabetic Retinopathy based on 18 total features comprised of categorical(integer) and continuous(float) features. I implemented a decision tree machine learning model for this data with random forests to increase accuracy and threshold splits to prevent overfitting.

LocNes — *Santa Barbara, CA*

Web Developer — *Mar. 2019 - Oct. 2019*

- Designed and developed the website and e-commerce platform for a startup called LocNes. This website was written in HTML, CSS, and JavaScript and incorporates the Shopify API.

Walter Reed National Military Medical Center — *Bethesda, MD*

Hardware Engineer Intern — *Jul. 2018 - Aug. 2018*

- Interned at Walter Reed's state-of-the-art 3D Printing facility designing and printing devices for medical applications. I also built my own custom 3D printer prior to this internship.

First-Person-View Quadcopter — *Oak Park, CA*

Hardware & Firmware — *Oct. 2018 - Dec. 2018*

- Built a quadcopter that is controlled with a first-person-view real time video feed to better understand how all the hardware components such as the electronic speed controllers, PDB, radio and video receivers/transmitters, brushless DC motors, and gyroscope interact with firmware placed on the flight controller.