# Michelle Cheng

| *Phone* | *Email* | *Location* | *Github* | *Portfolio* |
|---|---|---|---|---|
| (562) 253-8021 | chengmic@oregonstate.edu | San Diego, CA | github.com/chengmic | chengmic.vercel.app |

## OBJECTIVE

Recent graduate seeking an entry-level role in a computer science-related field. Aiming to advance to positions such as QA Engineer or Software Engineer while continuously developing and expanding skillsets.

## EDUCATION

**Oregon State University |** Corvallis, OR

B.S. Computer Science, December 2023

GPA: 3.98

## SKILLS

**Languages:** Python, C++, C#, SQL, Javascript, HTML, CSS

**Frameworks and Technologies:** Unity, MySQL, Flask, Next.js, Python unittest, Git

## PROJECTS

**ML Breakout |** https://github.com/chengmic/ml_breakout

**Description:** A class project replicating Atari's Breakout with an added Machine Learning element in versus mode for player vs. AI gameplay. Built using Unity.
- Unity and Unity ML Agents: Applied Unity, Unity ML Agents, and C# to create a functional and engaging gaming experience.
- Collaboration and Teamwork: Led and participated in agile team meetings, working closely with the team to ensure the game met project requirements and functional specifications.
- Quality Assurance: Actively engaged in bug tracking and reporting, covering functional, usability, and performance aspects.

**Dark Brew Café Database Website |** https://github.com/chengmic/Dark-Brew-Cafe-Database-Website

**Description:** A web application for the database management of a hypothetical company, Dark Brew Cafe.
- Database Design: Designed a database structure for Dark Brew Cafe, ensuring effective data management.
- Web Development: Utilized Flask to design and implement the web application, providing interface for CRUD operations.
- SQL: Implemented SQL queries and procedures to manage and retrieve data effectively.

**PokePY |** https://github.com/chengmic/PokePy

**Description:** A Python desktop Pokedex application, Uses CustomTkinter for the graphical user interface and PokeAPI for dynamic Pokemon data retrieval.

- API Integration: Utilized external API to dynamically fetch and display Pokemon data.
- Microservice Integration: Collaborated with a teammate to successfully implement and showcase the utilization of a microservice architecture.
- Error Handling: Developed error-handling mechanisms to maintain main program functionality in the absence of the microservice.