

# Jonathan P. Hang

[jonathan.hang@outlook.com](mailto:jonathan.hang@outlook.com) | [github.com/jnthnhng](https://github.com/jnthnhng)

## EDUCATION

---

### Oregon State University

*Bachelor of Science in Computer Science*

Corvallis, OR

*September 2022*

### University of Texas at Dallas

*Bachelor of Arts in Biology*

Richardson, TX

*March 2020*

## PROJECTS

---

### Fitopolis | *JavaScript, React Native, GitHub Actions, Stream's Activity Feed API, Firebase, Cloud Functions*

- Collaborated with a team to develop a cross-platform crowd-sourced fitness challenge application.
- Initiated the project by setting up project files, configuring GitHub Repository, and managing GitHub Actions.
- Implemented in-app search and display functionality, incorporating React Chip components to enhance keyword searches.
- Integrated Stream's Activity Feed API to create a global activity feed, allowing users to post text and images.
- Orchestrated the deployment of a server-less token retrieval function on Google's Cloud Functions.

### Harry Potter Movies Database | *JavaScript, Node.js, Express.js, AWS MariaDB*

- Fostered productive collaboration within the team, contributing to the design of a comprehensive project proposal, ER diagram, and database schema.
- Developed RESTful API for performing CRUD operations, enabling users to interact with the database through POST, GET, PUT, and DELETE actions.
- Led the backend development, leveraging Node.js, Express.js, and MySQL to build robust endpoints.
- Configured Amazon RDS to support the MariaDB backend for efficient data storage.

### Content Generator | *Python, Tkinter, RabbitMQ's Pika, Wikipedia API*

- Owned the design and development of a user-friendly GUI, allowing users to input primary and secondary keys, query Wikipedia via an API, and display a paragraph containing both keys.
- Implemented a solution to process CSV input and generate an output CSV file, enhancing the application's versatility.
- Developed a microservice with RabbitMQ's Pika module, functioning as both a producer and consumer, enabling efficient communication between components.

### Small Shell | *C*

- Spearheaded the implementation of built-in commands, including exit, cd, and status, enhancing the shell's functionality.
- Pioneered the support for additional commands by invoking the 'exec' family of functions, configuring input and output direction, and enabling foreground and background command execution.

## TECHNICAL SKILLS

---

**Languages:** JavaScript, Python, C, SQL, HTML/CSS

**Frameworks:** React Native, Node.js, Express.js

**Databases:** NoSQL, MySQL

**Developer Tools:** Git, Docker, Google Cloud Platform, Firebase, VS Code, Figma, Trello