

# Yeng Chien Wong

☎ (+60) 111-528-6176 | ✉ yengchien@hotmail.com | [in](#) LinkedIn | [WongYC-66](#) | [website](#) | [Kuala Lumpur, Malaysia](#)

## EDUCATION

The Odin Project Curriculum - Full Stack JavaScript	2023-2024
FreeCodeCamp Curriculum	2023
Bachelor of Engineering (Mechanical) (Honours) - Swinburne University of Technology	2015-2019

## WORK EXPERIENCE

Manufacturing Engineer	Western Digital Malaysia	04/2019 - 12/2022
1. Created GUI and developed new motion-based feature on Automation Robot(Omron Cobra Robot)		
2. Increased machine production rate by 25% through motion study and software optimization (control flow change and efficient data transfer)		

## PROJECTS

**Mini Threads** Full Stack  
A simple clone of Meta's Threads, offering functionality such as post creation, commenting, user following, and profile editing. The front-end provides a smooth user experience built with Next.js and React, while the back-end handles authentication with Github and JWT. The RESTful API is tested using Jest, and image uploads are managed via Supabase. Data modeling is handled using Prisma ORM for efficient database management. Created with next.js, react, tailwindcss, shadcn/ui, node.js, passport.js, express, postgresSQL, prisma, supabase.

**MessageMe Messenger** Full Stack  
A simple messaging application built with HTTP, featuring fast client-side routing as a single-page application (SPA) using React Router. Backend supports a RESTful API with JWT and Passport.js for authentication. It features basic group chat support, powered by PRISMA ORM. APIs tested with JEST. Created with react, react-router, node.js, express, prisma, postgresSQL.

**Game - Where's Waldo** Full Stack  
A simple Where's Waldo game. Client fetches click result from backend API. Upon game completion, username entered by client, and time used are recorded as leaderboard, stored in MongoDB. Created with react, vite, bootstrap, node.js, express, mongoDB.

**Cloudy Drive** Full Stack  
A simple clone of Google Drive, with server-sided rendering using EJS. Users can upload, delete, and manage files (CRUD), with shareable links enabled. The app is powered by middleware like Multer for file handling and Passport.js for authentication. Files are stored in Supabase for secure storage and retrieval. Created with ejs, bootstrap, node.js, express, prisma, supabase.

**Blog** Full Stack  
A simple blogging platform with a single-page application (SPA) built using React and Bootstrap. It features JWT-based authentication and supports basic CRUD operations for posts and comments via a RESTful API. The backend follows the MVC pattern, with data stored in MongoDB. Created with react, vite, bootstrap, node.js, express, mongoDB.

**Inventory Web Apps** Full Stack  
A simple inventory management application with backend functionality following the MVC pattern and server-side rendering using PUG. The UI is designed with Bootstrap, allowing users to perform CRUD operations on items, categories, and suppliers. Data is stored in a PostgreSQL database, and uploaded images are managed via Cloudinary. Created with pug, bootstrap, node.js, express, postgresql, cloudinary.

**Game - Memory Card** Front End  
An interactive browser game where user continue to select card without repeating. Built with react.js. Images used are fetched from pokeAPI. Created with react, vite.

**CV - Creator** Front End  
An interactive web application that allows users to create a CV by filling in their personal information. Created with react, vite.

**Maple Library** Front End  
A fan-made wiki for old school *MapleStory* featuring smooth client-side routing built with React, React Router, and Bootstrap. It incorporates fuzzy search and dynamic sorting based on user selection. Images are sourced from multiple locations, including local files, maplestory.io, and other websites. The data is fully static, exported as JSON and served to the client. Created with react, react-router, vite, bootstrap, JavaScript.

**Game - Tic Tac Toe** Front End  
A browser-based Tic Tac Toe game where you can play against another local player or the computer (random move). Created with html, css, JavaScript.

**To-do-list** Front End  
A web-based to-do list application where user data is stored locally in the browser. Created with html, css, JavaScript.

**Baaaaaattle ship** Front End  
A browser-based Battleship game where you play against a computer that initially makes random moves, then guesses intelligently based on possible locations.

Created with html, css, JavaScript.

### YUMEKO's Clothes

A simulated online shopping cart where users can add items, adjust quantities, and store the data locally in the browser.

Created with react, react-router, vite.

Front End

### Admin Dashboard

A simple user interface for an admin dashboard with basic layout and design only.

Created with html, css.

Front End

### Discord Bot

A fully functional Discord bot, which integrates with the Discord API and third-party APIs to deliver interactive features.


Features : slash command, external API integration, image rendering, scheduled reminders via cron-based GET request.

Created with node.js, netlify function















Serverless Back End

## SKILLS

---

**Programming Languages:**  JavaScript,  Python,  Ruby,  sql

**Tools and Frameworks:**

-  Git,  Node.js,  React,  React Router,  Next.js,  Jest,  Rails
-  Bootstrap,  Sass,  tailwindcss,  daisyUI
-  mongoDB,  PostgreSQL,  Prisma

**Languages:** English (proficient), Mandarin (native), Malay (proficient)

## CERTIFICATIONS

---

JavaScript (Intermediate) - HackerRank

Problem Solving (Basic) - HackerRank

JavaScript Algorithms and Data Structures - FreeCodeCamp

Data Visualization - FreeCodeCamp

Back End Development and APIs - FreeCodeCamp

Scientific Computing with Python - FreeCodeCamp

Information Security - FreeCodeCamp

College Algebra with Python - FreeCodeCamp

Problem Solving (Intermediate) - HackerRank

Responsive Web Design - FreeCodeCamp

Front End Development Libraries - FreeCodeCamp


Relational Database - FreeCodeCamp

Quality Assurance - FreeCodeCamp

Data Analysis with Python - FreeCodeCamp

Machine Learning with Python - FreeCodeCamp

Foundational C# with Microsoft - FreeCodeCamp

 Scan for softcopy

