

Hammed Olatunde

Front-end Developer

PROFILE

[LinkedIn](#) [Github](#)

CONTACT



[Portfolio](#)



Timmortaltech@gmail.com



[+2348107175197](tel:+2348107175197)



Address

Lagos, Nigeria.

SKILLS

React JS

Javascript

Next JS

Typescript

Git & Github

Tailwind CSS

HTML5

CSS3

API's

EDUCATION

2022

JAVASCRIPT DATA

STRUCTURES & ALGORITHMS

Freecodecamp

2022

REACT DEVELOPER

Zero To Mastery

2018

WEB DESIGN

Classic Systems Infotech

EXPERIENCE

Contributing to Open Source & Self Employed - Frontend Developer

11/2022 - Present

- I launched an Artificial Intelligence image generator MERN (MongoDB, Express JS, Node) application
- Consumed the Dall-E & Open AI API
- Implemented accessible, responsive and functional user interfaces which delivered a perfect user experience for all visitors regardless of the device they use to access the web app
- Debugged landing page carousel & was responsible for fixing overflowing layouts.

Mp4Hdd - Frontend Developer

09/2020 - 11/2022

- Designed, Implemented & Released, Layout Designs that resulted in a 23% increase in User Engagements
- Coded using HTML, CSS and JavaScript to develop features for both mobile and desktop platforms
- Developed Landing Page while implementing the best industry practices.
- Designed and Contributed to various layouts to meet usability and performance requirements.

Naijacrawl - Website Manager

09/2018 - 08/2020

- Troubleshooting Web Responsiveness
- Configured, purchased and had the domains made secure for websites.
- Utilized analytics and site metrics to determine areas of improvement.
- Uploaded, reviewed and edited Website Content

PROJECTS

EXPELIMUS AI Image Generator, MERN App

[Github](#) | [Web-App](#)

- My Favorite App yet, AI Image Generation App, where your imagination is the only limit to what can be created, it uses Dall-E & Open AI API to instantly fetch your prompts in real-time
- Built the optimal approach for the database and file storage for the AI generated Images.
- Implemented the App image storage to scale well, before creating instances of documents, I uploaded the AI generated images straight up to cloudinary, which stores it in the cloud and simply gives me a photo URL without needing to store them on users device
- Based on the Image URL I then create a new post in the Database by sharing only the Cloudinary URL (Image URL) , instead of using Base 64 URL which is good for a few images but is not ideal for lots of images

Graphevilla Social Media App

[Github](#) | [Web-App](#)

- Graphevilla is where users share images and discuss what the images inspires in them with other users who have similar interests.
- Implemented Google Auth, for users to login and have secured and private conversations
- Used Sanity Database for user posts storage
- Designed different categories so that users can find out what category of images they would like and meet others who have similar preferences.

Petsu Ecommerce Store App

[Github](#) | [Web-App](#)

- Credit Card functionality via Stripe, which sends customers a confirmation of their order into their inbox
- Managing the global state of the Cart across the entire application with Redux, to keep track of customers booked items, so they can comeback later to it and continue their shopping where they stopped
- I used MySQL as the chosen database for Strapi headless CMS for the store inventory management