https://\*\*\*.com **Healthcare project**

My recent project was a booking application for patients to book appointments with doctors. Patients can choose specific doctors, time, clinic with Google map. Also, by filling information in this application, they can introduce themselves and their medical history. By doing this, the doctor can prepare the meeting with the patient, and let them know more about patients.

In regard to the **technical part**, I used React, Redux for global state management, AWS amplify and Lambda for backend, AWS SNS, SES for sending SMS and email, IVS for video streaming, S3 bucket for store image files, Twilio for send MMS, and used bootstrap and styled component for styling components.

I worked in the React frontend team for this project, and mainly worked on select Appointment, card scan and OCR, Medical history, and review appointment part. My team has 12 team members and one team leader.

The biggest challenge in this project was implementing OCR. We have to implement OCR functionality to get the information from ID card, Insurance card, and Driving license card. There are many services to scan cards, and we decided to use **microblink for this**, but there's no one for insurance card. Even if there's one, we couldn't spend money for this service, because we need 3 different OCR functionalities. Because of this, I had to implement the OCR for Insurance card, and succeed. I am really proud of this.

To do this, I used tesseract-ocr NPM to read card data, and tried to detect card data manually. It was really difficult to scan several Insurance cards. Any way, I've succeeded to detect all data from Insurance card.

[www.ocean.com](http://www.ocean.com/) **Next.js**

It's my latest project built with Next.js. It's a booking application for holiday visitors to book appointments with ocean picnic. This website lets visitors explore incredible vacation experiences across their nine World's Leading Cruise Line brands.

In regard to the **technical part**, I used Redux for global state management, and used Ant design and styled component for styling components.

I worked independently as a frontend developer for this project.

### Most challenging:

We have to prevent Cross-Site Scripting cyberattack for this Project.

In summer vacation period, uncountable users visits this website. It reached to 30K last summer. The biggest challenge for this project was to make it secure and optimized website. We are using SSR in Next.js, and SSR opens a way for attackers to exploit vulnerabilities in third-party NPM packages.

To prevent this, I used

* Regular-Expressions for each data transformation,
* serialize-javascript package.

Serialize JavaScript to a superset of JSON that includes regular expressions, dates and functions.

By using this, we can prevent XSS Cross-Site Scripting cyberattack

By doing this, we succeed to maintain security and optimize the website.

[https://\*\*\*.com](https://petrolprices.com/)

This project is the ecommerce project that sells mens wear. The tech stacks are gatsby+strapi+datoCMS+firebase+react Gatsby is mainly used for shopping pages as it supports best loading speed and SSR. The customer wants the fast Speed on priority one so gatsby was the best solution as ecommerce data is not so changeable.

But the client came up with the new idea. This website also has the community area where customers can design and develop their own products as favorites. At that time we faced at one challenge. It is that gatsby did not work well for the very changeable data. So we came up with hybrid solution of gatsby and react and strapi instead of datoCMS for real time database change and show. it uses firebase realtime database and notification service. As the project contains so many images in the page, for optimization we did many things for image loading like lazy loading. Also as optimization we used local graphql request so it will be run within build time. It means we can save time for api request and help site performance. This projects was the first community led ecommerce project. Main services are deployed in GCP and netlify and algoria search for better search performance. It also has AI and live customer support service. As hybrid app, we used gatsby dynamic routing and client side routing for community pages for routing. It is also implementing live video sharing system using mediasoup. It is for users only. And that is based on Google cloud platform.

<https://petrolprices.com>

The goal of this project was to provide petrol service to millions of drivers of UK. including petrol station service, petrol balance, etc In this project I worked as a frontend developer. We use React graphql and AWS serverless framework. We use graphql as it is good fit for complex systems and microservices. The good points are as follows.

* Fetching data with a single API call. ...
* No over- and under-fetching problems. ...
  + Good fit for complex systems and microservices. ...
* Tailoring requests to your needs. ...
* Validation and type checking out-of-the-box. ...
* Autogenerating API documentation. ...
* API evolution without versioning. ...
* Code-sharing. About css we use styled components. The techs I implemented was google map service for best reasonable petrol stations, calendar integration, payment systems, etc. One of the important goal was the scalability of the project as it should provide service to millions of drivers at the same time. So we used AWS services like load balancer, [VPC](https://aws.amazon.com/vpc/faqs/), [ECS](https://aws.amazon.com/ecs/), etc. They were good for removing heavy load for backend. That is it.

<https://mightyplants.com/>

I would like to explain about my project with vue.js.

jsut sec. let me share my project url and codebase.

This is my project based on vue. as you can see, this is a ecommerce site.

My clients want Mighty Plants to be a platform that showcases the amazing plant-based products from lesser-known brands, alongside the well established favourites.

They always focus on frozen products as it retains its freshness and quality and has a big impact on reducing food waste

So we placed **great emphasis** on making frontend design more beautiful and fresh.

1. Project Introduction:

Mighty Plants is the UK's food industry business and ecommerce platform that selects the best plant-based products from independent brands.

1. Modules description:

Basically it has two modules - consumer facing and admin portal.

1. Advantages and the main functionality of your application:

Mighty Plants is the retailer business, so admin portal allows to administrators can upload products, logo, pricing from various brands so that they can update business categories, invite collaborators in organization. Consumers can see product details and make orders, set payment with Stripe also support google and apple pay, deliver ordered items with UK's shipping service.

1. Tools, Technologies, and Platform used:

We used Vue, SCSS for frontend, Netlify for frontend hosting, Algolia for search engine solution. And we used Segment and Google Adwords for digital marketing. Backend is designed with Go-microservice architecture and hosted by Google Cloud Platform.

1. Personal contribution and your role in the project:

I contributed this project as the frontend lead, there are 6 of our team. I've managed frontend development in collaboration with cross-functional and Scrum teams, including participation in daily Scrum calls, Git version control, task management on Jira, and communications via Slack. I always try to follow and maintain a good coding architecture before starting any sprint, I usually brainstormed my idea with other developers. The reason is that, rather than just coding and meeting criteria, we should give our efforts to make code runs faster and better.

1. Challenges in the project:

The hard part was to setup checkout flow and upload page in admin dashboard.

1. Amount of time it took for the project:

I had worked for 6 months on Mighty Plants.

1. Improvements in the future for the present system:
2. Drawbacks:

Mighty Plants is the UK's food industry business and ecommerce platform that selects the best plant-based products from independent brands.

Basically it has two modules - consumer facing and admin portal.

Now I am working in OURFIRES company as a frontend developer in contract part time position. In comcate company I have worked as a front end developer in remote position. I was a member of talented software team that works on technologically cutting-edge application in a software industry startup. my Roles and responsibilities include managing Angular(Gatsby,Vue) & Node.JS application development while providing expertise in the full software development lifecycle, from concept and design to assisting in testing.

**Remote creative work project**

1. **Project Introduction:**

Our client objectives is to bridge the divide between art, branding, design and technology to make digital products and services that are both memorable and usable.

As a part of the company strategies, my client decided to build a social job site during pandemic, which helps clients and developers can find a relevant employees and relevant jobs according to their company strategy and personal abilities.

During the pandemic, remote working has been an essential and popular part of the employment system. So we couldn't spend much time to prepare. All designers and developers have become creators for setting a strategy and plan.

According to project characteristics, we thought as a JavaScript based React framework Gatsby is the most suitable framework for our project and choose it.

Gatsby is an open source front-end development framework. It helps us build fast, reliable static websites using React and other JavaScript tools. These sites generally do not require a traditional back end. Instead they use APIs and CDNs to deliver content to browsers or mobile apps.

My clients wanted to be a platform that showcases the reliable social job-posting products from lesser-known brands, alongside the well established favorites.

So we placed **great emphasis** on gathering all digital product and services as well as making frontend design more simple.

1. **Modules description:**

This project consists of two parts. job post part for client who is finding developers, and the other part which developers and employees can apply in the job matched for their skill set.

so client portal allows to client can post its hiring process, logo, contact information and set payment with Stripe so that they can update business categories, invite collaborators in organization. on the other hand, developers can see jobs details from job categories and choose one or more jobs which is matched for their ability and skills.

1. **Advantages and the main functionality of your application:**

As a social job site, this project is similar to other job site. But the main advantages of this site is to involve all kind of digital product and services which requires employees remotely.

And even though remote working has been popular in employment system, but many clients and developers are still strange for remote working. so we attached tutorials and blogs for them.

How to become a good remote colleague, how to become a good remote manager and the other kinds of tips for helps them work remotely.

1. **Tools, Technologies, and Platform used:**

We used serverless framework Gatsby, SCSS for frontend, Netlify for frontend hosting, Algolia for search engine solution.and we used s3 bucket as a AWS services And we used Segment and Google Adwords for digital marketing. Backend is designed with Go-microservice architecture and hosted by Google Cloud Platform.

1. **Personal contribution and your role in the project:**

I contributed this project as the frontend lead, there are 6 of our team. I've managed frontend development in collaboration with cross-functional and Scrum teams, including participation in daily Scrum calls, Git version control, task management on Jira, and communications via Slack. I always try to follow and maintain a good coding architecture before starting any sprint, I usually brainstormed my idea with other developers and engineers. The reason is that, rather than just coding and meeting criteria, we should give our efforts to make code runs faster and better.

The hard part was to setup checkout flow and upload page in admin dashboard.

I had worked for 6 months on Mighty Plants.

Mighty Plants is the UK's food industry business and ecommerce platform that selects the best plant-based products from independent brands.

Basically it has two modules - consumer facing and admin portal.