

Documentation for StackFood React Web App

Introduction

Thank you for your interest in StackFood.

StackFood is a complete Multi Restaurant food delivery system developed using Laravel and Flutter Framework. Laravel (PHP Framework for Web Artisans) is a web application framework with an expressive, elegant syntax. We've already laid the foundation – freeing you to create without sweating the small things. Flutter framework created by Google is open-source mobile application development. It is used to develop applications for Android and iOS, as well as being the primary method of creating applications.

StackFood comprises of 3 Mobile Applications, 2 Web Panel, 1 website, 1 landing page and 1 web-app(React) –

- Customer application for Android, iOS and web developed using Flutter.
- Delivery Boy application for Android developed using Flutter
- Restaurant application for both Android and iOS developed using Flutter.
- Super Admin web panel developed using Laravel (admin can provide role base access).
- Restaurant web panel developed using Laravel (owner can provide role base access).
- Landing page developed using Laravel blade.
- Web app build with React and Next.js

Prerequisite (React Web App)

StackFood Admin Panel minimum version 6.0.1 should be pre installed on your server. If you didn't purchased the StackFood Main system yet, [Click here](#) to purchase StackFood main system.

- [Vs code](#) or [Webstorm](#).

- [Nodejs](#)
- [Npm](#) or [Yarn](#)

Environment Setup

You have to download and setup node js from nodejs.org. You can follow documentation with your own device.

Installing Node js:

Recommended link is here -

<https://nodejs.org/en/download>

Great!! As now you have installed the [NodeJs](#) successfully, now you just have to install **NPM or Yarn**. Below we have given the easy documentation.

Installing npm:

You can check the official documentation -

<https://docs.npmjs.com/downloading-and-installing-node-js-and-npm>

To download the latest version of npm, run the following command on the command line:

```
npm install -g npm
```

Or,

Installing Yarn:

You can check the official documentation -

<https://classic.yarnpkg.com/lang/en/docs/install/#debian-stable>

```
npm install --global yarn
```

Mandatory setup

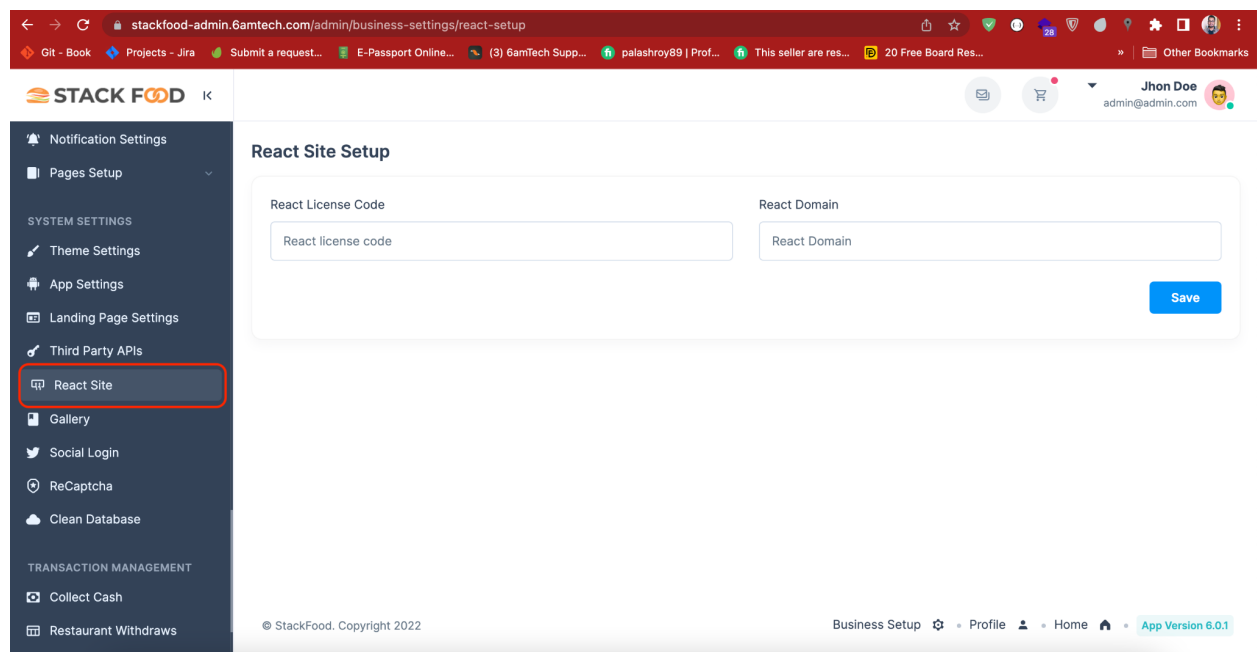
Note: This item doesn't include any admin panel, as we mention before in prerequisites section you need to have pre installed the StackFood Admin Panel on your server to use this React Web App. If you haven't purchased the StackFood main item yet, [click here](#) to purchase and install the admin panel first on your server first.

Activate React Web App in the Admin Panel:

Go to your StackFood Admin Panel,

- Go to system settings section from side menu bar.
- Go to In the submenu called React site.
- Set your purchased license code and your registered domain (Here registered domain means where you want to host your react web app, for example here I want to host my react web app here <https://stackfood-react.6amtech.com> so I need to put this domain/sub-domain here as a react domain).

If you failed to get your purchased license code [check this](#).



Active from your React project:

Note: Download the react web app source code from Codecanyon. If you don't know how to download the source code follow the following steps -

- Log into your Envato Market account.
- Hover the mouse over your username at the top of the screen.
- Click 'Downloads' from the drop-down menu.'
- Click 'All files and documentation.

Now unzip the zip files and go to StackFood React folder and the project in your IDE ([Vs code](#) or [Webstorm](#)).

- Go to project **.env.development** file and **.env.production**
- Change your NEXT_CLIENT_HOST_URL variable with the same domain that you registered in the admin panel. For example: My admin panel is hosted in <https://stackfood-admin.6amtech.com> and I will host the react web app in <https://stackfood-react.6amtech.com> so I changed the following variable like below, please note don't add / at the end.

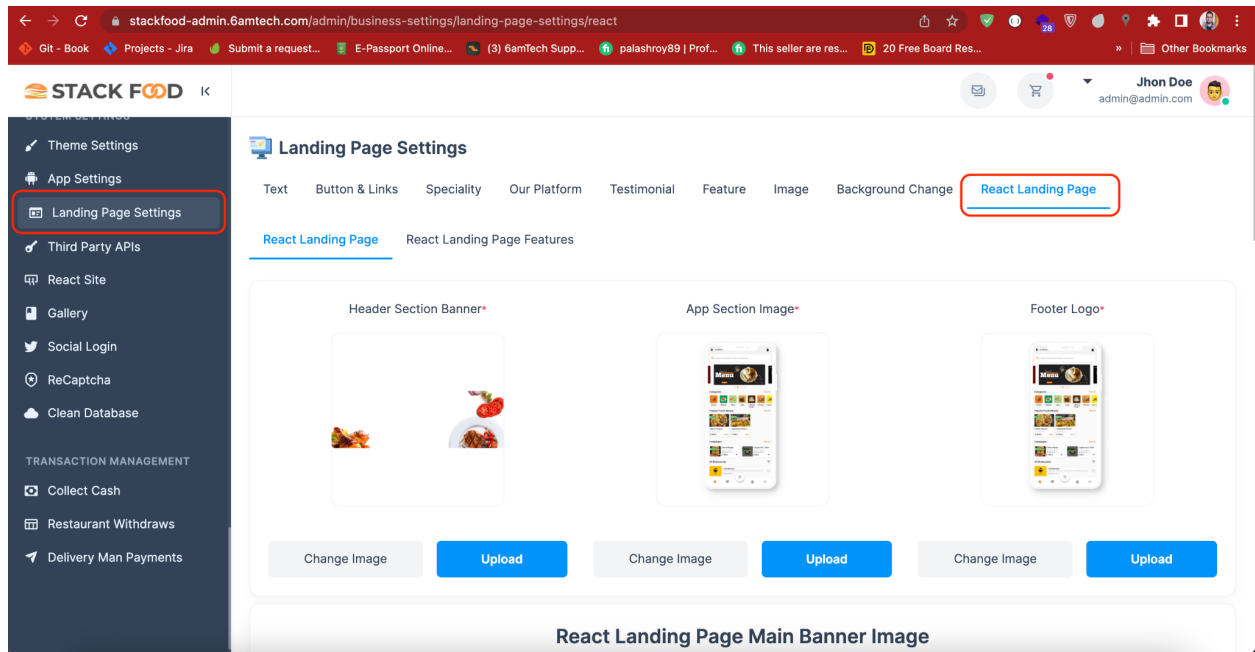
NEXT_PUBLIC_BASE_URL= https://stackfood-admin.6amtech.com

NEXT_CLIENT_HOST_URL= https://stackfood-react.6amtech.com

Landing Page Settings in the Admin Panel:

Again go to your StackFood Admin Panel,

- Go to the section named SYSTEM SETTINGS from side menu bar.
- From SYSTEM SETTINGS go to the menu called **Landing page settings**
- From landing page settings , go to the submenu named **React landing page**
- Setup all your preferred information. Like, Top Header Title,Sub Title,About Section Title,Footer Article and many more things.



Mandatory Setup - In your React Code

Run an existing react project on IDE:

- Run the purchased and downloaded NextJS project on your preferred IDE.
- Before running the project in your local machine,
 - You have to set your activated domain url and purchased license key in your admin panel.
 - To set up your domain in the admin panel please follow below instruction.
 - In the admin panel we have a menu called system settings. In the system settings we have a submenu called React site where you give your purchased license code and your activated domain url to the form and submit it.
 - You also have to change your project .env.development and .env.production->NEXT_CLIENT_HOST_URL and NEXT_PUBLIC_BASE_URL variables with your activated domain url.

Recommended tutorial is below 🖱️

<https://youtu.be/g3jf6nvnJRM>

Change Base URL:

To change the base URL follow the steps given below -

- Must remember that don't put slash(/) at the end of your base url. Already mentioned in **Active from your React project** section.
- Use your admin url as base url.
- First you have to install your admin panel. For example: If your admin url is `https://your_domain.com/admin` then base url will be `https://your_domain.com`.
- Create `.env.production` and `.env.development` file in your project
- Add `NEXT_PUBLIC_BASE_URL` and `NEXT_CLIENT_HOST_URL` variable in both env files. Then assign your URL as a variable value. For example:
`NEXT_PUBLIC_BASE_URL="Your URL"`
- `NEXT_CLIENT_HOST_URL:"Your host URL"`

Add Google Map API Key:

- You need to generate the google API key. Visit this link- <https://developers.google.com/maps/documentation/embed/get-api-key>
- You need to enable mention APIs: Direction API, Distance Matrix API, Geocoding API, , Place API.
- You have to enable billing account. Visit this url for activating: <https://support.google.com/googleapi/answer/6158867?hl=en>
- After generating API key, you have to put it on your project `.env` file.
- Open `<project>.env.production` and `<project>.env.development` and add your api key inside those files at `NEXT_PUBLIC_GOOGLE_MAP_KEY`. For example:
`NEXT_PUBLIC_GOOGLE_MAP_KEY = YOUR_MAP_API_KEY_HERE`

Recommended tutorial is below 🖱️

<https://youtu.be/sSuXJTEuZsk>

Setup Firebase for Push Notification

To get push notification you need to setup firebase in your react web app project, But please note don't create a new firebase project for react web app, you need to use one firebase project for all apps and web of StackFood system.

Recommended tutorial is below 🖱️

<https://youtu.be/bnUHffkVI1Q>

Customization Setup

Country Filter:

To change the country filter, follow the steps below-

- If you want your country only in country choosing dialog which we saw in login, registration and forget password page, you have to set your default country first from admin panel Business Setup section. Then open `<project>src/component/CustomPhoneInput.js` file and search `onlyCountries`. Now add a parameter with value like this: `onlyCountries={[defaultCountry]}`
- If you want to disable the country choosing dialog then just change `disableDropdown="false"` to `disableDropdown="true"`.

Recommended tutorial is below 🖱️

<https://youtu.be/6nNu-wA115U>

Add New Local Language:

To add a new language, follow the steps below-

- Go to project `/src/language` and press the right button on the language folder and create a new file and name it with your language code (.js). For example if your language is Bengali then you have to name your file as `bn.js`. You have to name it with proper and valid language code otherwise app won't work. For getting

language and country code you can visit this url:

https://docs.oracle.com/cd/E13214_01/wli/docs92/xref/xqisocodes.htm

- Copy all data from en.js and paste it in your created file.
- Translate all English text placed here after colon(:) to your local language. Their texts are in key-value format. You have to translate the value only, not the key , otherwise it won't work. For example: "home": "Home" -> "home": "বাড়ি"*
- Import your language file i18n.js which is placed in the same folder.after that in i18n.js file line 10(resources object) add your language as like other.
- You can set initial language by following way I18n.js file line 25 set lang: "your initial language" and line 28 set fallbackLng:"your initial language".

```
/src/language?/i18n.js
```

```
const resources = {  
  en: {  
    translation: english,  
  },  
  bn: {  
    translation: bengali,  
  },  
  ar: {  
    translation: arabic,  
  },  
}
```

```
i18n.use(initReactI18next) // passes i18n down to react-i18next
```

```
.init({  
  resources,  
  lng: 'en',  
  
  fallbackLng: 'en',  
  interpolation: {  
    escapeValue: false,  
  },  
})
```


Recommended tutorial is below 🙌

<https://youtu.be/Boozbjrw7xY>

Change Web App color:

To change the app color, follow the steps below-

- Open `<project>/src/theme/light-theme-options.js` file and set preferred primary colors for light themes.
- In the same way open `<project>/src/theme/dark-theme-options.js` file and set preferred primary, accent and etc. colors for dark theme.

Recommended tutorial is below 

<https://youtu.be/jwVw8HH1QRY>

Change Web App Font:

To change website font you have to go to `<project>src/theme/base-theme-options`. then line 299 `fontFamily: "Signika Negative", "sans-serif"`, replace `fontFamily`. Path: `project/src/theme/base-theme-options` Example : `fontFamily:"your font family"`

Site Build and Deploy

Deployed in your own server:

Next.js can be deployed to any hosting provider that supports Node.js. If you've followed the instructions so far, your `package.json` should have the following build and start scripts:

```
/src/language?/i18n.js
{
  "scripts": {
    "dev": "next",
    "build": "next build",
```

```
"start": "next start"
}
}
```

In your own terminal, run the build script once, which builds the production application in the .next folder.

npm run build

To start the deployed site, use the command below

npm run start

INFO

For more information, follow this [official documentation](#)


For Local Build and deploy:

For local build -

- you have to run the following command on your project directory.

npm run build

- After the successful build, make sure your directory has the .next folder then deploy the whole project contained with .next to the server.

Recommended tutorial is below 

<https://youtu.be/xBh6HGXIqKc>

Deploy to Vercel:

The easiest way to deploy Next.js to production is to use the Vercel platform developed by the creators of Next.js.

Vercel is a serverless platform for static and hybrid applications built to integrate with your headless content, commerce, or database. We make it easy for frontend teams to develop, preview, and ship delightful user experiences, where performance is the default. You can start using it for free — no credit card required.

Create a Vercel Account First, go to <https://vercel.com/signup> to create a Vercel account. Choose Continue with GitHub and go through the sign up process.

Recommended documentation is below 🙋

link:<https://nextjs.org/learn/basics/deploying-nextjs-app/deploy>

----- Best of luck with Fast, Powerful and Modern Technology -----