**Start Project:**

1. Install Node.js

Check nodes: node -v

1. Install NPM

Cmd install: npm install -g npm

Check npm: npm -v

1. Globally React install

Cmd install: npm install -g create-react-app

Check: create-react-app –version

1. Globally install react creating app

Cmd: create-react-app xxxxxxx

Cmd: npm init react-app xxxxxxx

1. Direct Install or Create your react app

Cmd: npx create-rect-app xxxxxxxx (npx means the nove module will be above the version)

Cmd: npx create-react-app . (to create app in the same directory without creating a new folder)

1. Run the react app

Cmd: npm start

**Configure project**

1. Delete all files inside src except Apps.js, Index.css y index.js
2. Delete all content of index.css.
3. Write the following codes:

In index.js write the following:

**import** React **from** 'react';

**import** ReactDOM **from** 'react-dom';

**import** './index.css';

**import** App **from** './App';

ReactDOM.render(

    <**App** />,

  document.getElementById('root')

);

In App.js write the following:

**function** App() {

**return** (

    <**div**>

      <**h1**>App</**h1**>

    </**div**>

  );

}

**export** **default** App;

1. Check if the project still runs:

Cmd: npm start

Imagen que contiene Forma

Descripción generada automáticamente

**Add dependencies:**

1. Open package.json to check the dependencies loaded.

Interfaz de usuario gráfica, Texto

Descripción generada automáticamente

1. Add the dependency React Router DOM <https://reactrouter.com/> <https://www.npmjs.com/package/react-router> by writing in the command line: npm install react-router-dom@6
2. To include the social media icons, you’ll use a package called react-font-awesome, which provides Font Awesome support for React. You’ll need these three packages:

* @fortawesome/react-fontawesome
* @fortawesome/fontawesome-svg-core
* @fortawesome/free-brands-svg-icons

Install these with the following command:

npm install --save @fortawesome/react-fontawesome @fortawesome/fontawesome-svg-core @fortawesome/free-brands-svg-icons

1. Add the Google Maps API dependency to insert maps in the react application.

npm install --save google-maps-react

1. Let's install emailjs package to send automatic emails

npm i emailjs-com

1. Install EmailJS SDK like a module. This SDK fully supports TypeScript.

npm install @emailjs/browser

1. Let’s install react-hook-form and yup libraries to validate form fields and also a yup branch @hookform/resolvers to allow us to create a schema base for validation

npm install react-hook-form yup @hookform/resolvers

**Add Bootstrap styles:**

There are 3 options: To add bootstrap from https://getbootstrap.com; to get the react bootstrap <https://react-bootstrap.github.io/> o el bootstrap cdn <https://www.bootstrapcdn.com/>

1. Install bootstrap from https://getbootstrap.com/docs/5.1/getting-started/download/

npm install bootstrap

1. Go to the rute of node\_modules\bootstrap\dist\css\bootstrap.min.css, click with mouse left button and click on copy path. Then past this path into the import sentence on index.js file.

**import** 'node\_modules\bootstrap\dist\css\bootstrap.min.css';

1. On the get starting, we will find more links to include as the poopers, bungle, Js, etc. Go to find those files into the installation files, and import the link.

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

**import** 'node\_modules\bootstrap\dist\js\bootstrap.bundle.min.js';

1. Change the direction on the bars of both importations.

**import** 'bootstrap/dist/css/bootstrap.min.css';

**import** 'bootstrap/dist/js/bootstrap.bundle.min.js';

**Structure the project:**

1. Build the following structure inside src folder:

Interfaz de usuario gráfica

Descripción generada automáticamente con confianza baja

1. Go into each file: Home.js; About.js and Contacts.js and construct the component writing rfc + enter to get a template.

**import** React **from** 'react'

**export** **default** **function** Home() {

**return** (

        <**div**>

        </**div**>

    )

}

1. Inside the return sentence, write the following bootstrap template:

        <**div** className**=**"container">

            <**div** className**=**"card mt-4">

                <**div** className**=**"card-body">

                    <**h2**>Home Page</**h2**>

                </**div**>

            </**div**>

        </**div**>

1. Repete this process for each of the pages.

**Build the Navbar**

1. Go to bootstrap webpage and look for the code of Navbar component and copy it. Paste this code into the Navbar.js page inside the div tags of function return sentence.

<https://getbootstrap.com/docs/5.1/components/navbar/>

Check if there are some syntax errors in the code.

**import** React **from** 'react'

**export** **default** **function** Navbar() {

**return** (

        <**div**>

            <**nav** class**=**"navbar navbar-expand-lg navbar-light bg-light">

                <**div** class**=**"container-fluid">

                    <**a** class**=**"navbar-brand" href**=**"#">Navbar</**a**>

                    <**button** class**=**"navbar-toggler" type**=**"button" data-bs-toggle**=**"collapse" data-bs-target**=**"#navbarSupportedContent" aria-controls**=**"navbarSupportedContent" aria-expanded**=**"false" aria-label**=**"Toggle navigation">

                        <**span** class**=**"navbar-toggler-icon"></**span**>

                    </**button**>

                    <**div** class**=**"collapse navbar-collapse" id**=**"navbarSupportedContent">

                        <**ul** class**=**"navbar-nav me-auto mb-2 mb-lg-0">

                            <**li** class**=**"nav-item">

                                <**a** class**=**"nav-link active" aria-current**=**"page" href**=**"/">Home</**a**>

                            </**li**>

                            <**li** class**=**"nav-item">

                                <**a** class**=**"nav-link" href**=**"#">Link</**a**>

                            </**li**>

                            <**li** class**=**"nav-item dropdown">

                                <**a** class**=**"nav-link dropdown-toggle" href**=**"#" id**=**"navbarDropdown" role**=**"button" data-bs-toggle**=**"dropdown" aria-expanded**=**"false">

            Dropdown

                                </**a**>

                                <**ul** class**=**"dropdown-menu" aria-labelledby**=**"navbarDropdown">

                                    <**li**><**a** class**=**"dropdown-item" href**=**"#">Action</**a**></**li**>

                                    <**li**><**a** class**=**"dropdown-item" href**=**"#">Another action</**a**></**li**>

                                    <**li**><**hr** class**=**"dropdown-divider"/></**li**>

                                    <**li**><**a** class**=**"dropdown-item" href**=**"#">Something else here</**a**></**li**>

                                </**ul**>

                            </**li**>

                            <**li** class**=**"nav-item">

                                <**a** class**=**"nav-link disabled">Disabled</**a**>

                            </**li**>

                        </**ul**>

                        <**form** class**=**"d-flex">

                            <**input** class**=**"form-control me-2" type**=**"search" placeholder**=**"Search" aria-label**=**"Search">

                            <**button** class**=**"btn btn-outline-success" type**=**"submit">Search</**button**>

                        </**form**>

                    </**div**>

                </**div**>

            </**nav**>

        </**div**>

    )

}

1. Delete part of the code of Navbar that is not needed and keep only what you need.

**import** React **from** 'react'

**export** **default** **function** Navbar() {

**return** (

        <**div**>

            <**nav** class**=**"navbar navbar-expand-lg navbar-light bg-light">

                <**div** class**=**"container-fluid">

                    <**a** class**=**"navbar-brand" href**=**"#">Navbar</**a**>

                    <**button** class**=**"navbar-toggler" type**=**"button" data-bs-toggle**=**"collapse" data-bs-target**=**"#navbarSupportedContent" aria-controls**=**"navbarSupportedContent" aria-expanded**=**"false" aria-label**=**"Toggle navigation">

                        <**span** class**=**"navbar-toggler-icon"></**span**>

                    </**button**>

                    <**div** class**=**"collapse navbar-collapse" id**=**"navbarSupportedContent">

                        <**ul** class**=**"navbar-nav me-auto mb-2 mb-lg-0">

                            <**li** class**=**"nav-item">

                                <**a** class**=**"nav-link active" aria-current**=**"page" href**=**"/">Home</**a**>

                            </**li**>

                            <**li** class**=**"nav-item">

                                <**a** class**=**"nav-link" href**=**"#">Link</**a**>

                            </**li**>

                        </**ul**>

                    </**div**>

                </**div**>

            </**nav**>

        </**div**>

    )

}

1. Apply router-dom to the Navbar by importing the Link and change all a tags to Link tags.

**import** React **from** 'react'

**import** { Link } **from** 'react-router-dom';

**export** **default** **function** Navbar() {

**return** (

        <**div**>

            <**nav** class**=**"navbar navbar-expand-lg navbar-light bg-light">

                <**div** class**=**"container-fluid">

                    <**Link** to**=**"/" class**=**"navbar-brand">Navbar</**Link**>

                    <**button** class**=**"navbar-toggler" type**=**"button" data-bs-toggle**=**"collapse" data-bs-target**=**"#navbarSupportedContent" aria-controls**=**"navbarSupportedContent" aria-expanded**=**"false" aria-label**=**"Toggle navigation">

                        <**span** class**=**"navbar-toggler-icon"></**span**>

                    </**button**>

                    <**div** class**=**"collapse navbar-collapse" id**=**"navbarSupportedContent">

                        <**ul** class**=**"navbar-nav me-auto mb-2 mb-lg-0">

                            <**li** class**=**"nav-item">

                                <**Link** to**=**"/" class**=**"nav-link active">Home</**Link**>

                            </**li**>

                        </**ul**>

                    </**div**>

                </**div**>

            </**nav**>

        </**div**>

    )

}

1. Change to ms-auto to put the navbar on the right
2. Add the Links to the other pages:

**import** React **from** 'react'

**import** { Link } **from** 'react-router-dom';

**export** **default** **function** Navbar() {

**return** (

        <**div**>

            <**nav** class**=**"navbar navbar-expand-lg navbar-light bg-light">

                <**div** class**=**"container-fluid">

                    <**Link** to**=**"/" class**=**"navbar-brand">ACE - Aire Comprimido Eficiente</**Link**>

                    <**button** class**=**"navbar-toggler" type**=**"button" data-bs-toggle**=**"collapse" data-bs-target**=**"#navbarSupportedContent" aria-controls**=**"navbarSupportedContent" aria-expanded**=**"false" aria-label**=**"Toggle navigation">

                        <**span** class**=**"navbar-toggler-icon"></**span**>

                    </**button**>

                    <**div** class**=**"collapse navbar-collapse" id**=**"navbarSupportedContent">

                        <**ul** class**=**"navbar-nav ms-auto mb-2 mb-lg-0">

                            <**li** class**=**"nav-item">

                                <**Link** to**=**"/" class**=**"nav-link active">Home</**Link**>

                            </**li**>

                            <**li** class**=**"nav-item">

                                <**Link** to**=**"/about" class**=**"nav-link active">About Us</**Link**>

                            </**li**>

                            <**li** class**=**"nav-item">

                                <**Link** to**=**"/contact" class**=**"nav-link active">Contact Us</**Link**>

                            </**li**>

                        </**ul**>

                    </**div**>

                </**div**>

            </**nav**>

        </**div**>

    )

}

**Route Pages**

1. Go to Apps.js and include these pages

**import** React **from** 'react';

**import** { BrowserRouter **as** Router} **from** 'react-router-dom';

**import** Navbar **from** './components/inc/Navbar';

**import** Home **from** './components/pages/Home';

**import** About **from** './components/pages/About';

**import** Contact **from** './components/pages/Contact';

**function** App() {

**return** (

    <**Router**>

        <**div**>

            <**Navbar** />

            <**Home** />

            <**About** />

            <**Contact** />

        </**div**>

    </**Router**>

  );

}

**export** **default** App;

1. Press ctrl + space to import the component **import** Contact **from** './components/pages/Contact';
2. Go again to App.js and create some routes like the following:

**import** React **from** 'react';

**import** { BrowserRouter **as** Router, Route, Routes} **from** 'react-router-dom';

**import** Navbar **from** './components/inc/Navbar';

**import** Home **from** './components/pages/Home';

**import** About **from** './components/pages/About';

**import** Contact **from** './components/pages/Contact';

**function** App() {

**return** (

    <**Router**>

        <**div**>

            <**Navbar** />

            <**Routes**>

                <**Route** path**=**"about" element**=**{<**About** />} />

                <**Route** path**=**"contact" element**=**{<**Contact** />} />

                <**Route** path**=**"/" element**=**{<**Home** />} />

            </**Routes**>

        </**div**>

    </**Router**>

  );

}

**export** **default** App;

**Footer :**

1. Go to Footer.js and introduce this code :

**import** React **from** 'react';

**import** { Link } **from** 'react-router-dom';

**export** **default** **function** Footer() {

**return** (

        <**div**>

            <**section** className**=**"section footer bg-dark text-white">

                <**div** className**=**"container">

                    <**div** className**=**"row">

                        <**div** className**=**"col-md-4">

                            <**h6**>Company Information</**h6**>

                            <**hr**/>

                            <**p** className**=**"text-white">

                                Bla bla bla

                            </**p**>

                        </**div**>

                        <**div** className**=**"col-md-4">

                            <**h6**>Quick Links</**h6**>

                            <**hr**/>

                            <**div**><**Link** to**=**"/">Home</**Link**></**div**>

                            <**div**><**Link** to**=**"/about">About Us</**Link**></**div**>

                            <**div**><**Link** to**=**"/contact">Contact</**Link**></**div**>

                            <**div**><**Link** to**=**"/">Home</**Link**></**div**>

                        </**div**>

                        <**div** className**=**"col-md-4">

                            <**h6**>Contact Information</**h6**>

                            <**hr**/>

                            <**div**><**p** className**=**"text-white mb-1">Direccion</**p**></**div**>

                            <**div**><**p** className**=**"text-white mb-1">Telf</**p**></**div**>

                            <**div**><**p** className**=**"text-white mb-1">Diremobile</**p**></**div**>

                            <**div**><**p** className**=**"text-white mb-1">email</**p**></**div**>

                        </**div**>

                    </**div**>

                </**div**>

            </**section**>

        </**div**>

    )

}

**Social Icons :**

1. Look for the social media icons in <https://fontawesome.com/v6.0/icons?m=free>
2. Write the code:

**import** React **from** "react";

**import** { FontAwesomeIcon } **from** "@fortawesome/react-fontawesome";

**import** {

  faYoutube,

  faFacebook,

  faTwitter,

  faInstagram,

  faLinkedinIn,

  faWhatsapp,

  faSkype

} **from** "@fortawesome/free-brands-svg-icons";

**import** { Link } **from** 'react-router-dom';

**export** **default** **function** SocialFlow() {

**return** (

    <**div**>

        <**div** className**=**"section bg-dark">

            <**p** className**=**"social-container">

                <**p** className**=**"text-white">Siganos en las redes sociales:  </**p**>

                <**Link** to**=**"#" className**=**"youtube social">

                    <**FontAwesomeIcon** icon**=**{faYoutube} size**=**"2x" />

                </**Link**>

                <**Link** to**=**"https://www.facebook.com/ACE-Aire-Comprimido-Eficiente-103723305063187" className**=**"facebook social">

                    <**FontAwesomeIcon** icon**=**{faFacebook} size**=**"2x" />

                </**Link**>

                <**Link** to**=**"#" className**=**"instagram social">

                    <**FontAwesomeIcon** icon**=**{faInstagram} size**=**"2x" />

                </**Link**>

                <**Link** to**=**"#" className**=**"twitter social">

                    <**FontAwesomeIcon** icon**=**{faTwitter} size**=**"2x" />

                </**Link**>

                <**Link** to**=**"https://www.linkedin.com/company/aceaire" className**=**"linkedin social">

                    <**FontAwesomeIcon** icon**=**{faLinkedinIn} size**=**"2x" />

                </**Link**>

                <**a** href**=**"https://api.whatsapp.com/send?phone=648808368&text=Gracias%20por%20contactar%20a%20ACE%20-%20Aire%20Comprimido%20Eficiente.%20%C2%BFEn%20que%20te%20puedo%20ayudar?" className**=**"whatsapp social">

                    <**FontAwesomeIcon** icon**=**{faWhatsapp} size**=**"2x" />

                </**a**>

                <**a** href**=**"#" className**=**"skype social">

                    <**FontAwesomeIcon** icon**=**{faSkype} size**=**"2x" />

                </**a**>

            </**p**>

        </**div**>

    </**div**>

  );

}

1. Connect with whatsapp API. <https://api.whatsapp.com/send?phone=648808368&text=Gracias%20por%20contactar%20a%20ACE%20-%20Aire%20Comprimido%20Eficiente.%20%C2%BFEn%20que%20te%20puedo%20ayudar>?

**Contact Form:**

1. Write the form code the following way

**import** React **from** 'react'

**export** **default** **function** Contact() {

**return** (

        <**div**>

            <**section** className**=**"section">

                <**div** className**=**"container">

                    <**div** className**=**"card shadow">

                        <**div** className**=**"card-body">

                            <**div** className**=**"row">

                                <**div** className**=**"col-md-6 border-left">

                                    <**div**>

                                        <**h6**>Contact Form</**h6**>

                                    </**div**>

                                    <**hr** />

                                    <**div** data-form-alert**=**"true"></**div**>

                                    <**form** action**=**"https://formsubmit.co/info@ACE-Aire.com" method**=**"POST" data-form-title**=**"CONTACT FORM">

                                        <**div** className**=**"form-group">

                                            <**label** className**=**"mb-1">Full Name</**label**>

                                            <**input** type**=**"text" className**=**"form-control" name**=**"name" required**=**"" placeholder**=**"Nombre\*" data-form-field**=**"Name" />

                                        </**div**>

                                    <**div** className**=**"form-group">

                                        <**label** className**=**"mb-1">Phone Number</**label**>

                                        <**input** type**=**"tel" class**=**"form-control phone" name**=**"phone" placeholder**=**"Telefono" data-form-field**=**"Phone" />

                                    </**div**>

                                    <**div** className**=**"form-group">

                                        <**label** className**=**"mb-1">Email Adress</**label**>

                                        <**input** type**=**"email" class**=**"form-control email" name**=**"email" required**=**"" placeholder**=**"Email\*" data-form-field**=**"Email" />

                                    </**div**>

                                    <**div** className**=**"form-group">

                                        <**label** className**=**"mb-1">Message</**label**>

                                        <**textarea** class**=**"form-control message" name**=**"message" placeholder**=**"Type your message..." rows**=**"7" data-form-field**=**"Message"></**textarea**>

                                    </**div**>

                                    <**div** className**=**"form-group py-3">

                                        <**label** className**=**"mb-1">Message</**label**>

                                        <**button** type **=**"submit" className**=**"btn btn-primary shadow w-100">Send Message</**button**>

                                    </**div**>

                                    </**form**>

                                </**div**>

                                <**div** className**=**"col-md-6 border-start">

                                    <**h5**>Address Information</**h5**>

                                    <**div** className**=**"underline"></**div**>

                                    <**p**>

                                        Direction

                                    </**p**>

                                    <**p**>

                                        Phone

                                    </**p**>

                                    <**p**>

                                        Email

                                    </**p**>

                                </**div**>

                            </**div**>

                        </**div**>

                    </**div**>

                </**div**>

            </**section**>

        </**div**>

    )

}

1. Configure the email sent on <https://formsubmit.co/>
2. Activate the email sent following the instructions gotten from <https://formsubmit.co/>
3. Create a thanks page to get feedback from the form filled in.
4. Send an automatic email confirming you are taking care of the message sent.

**Insert Map**

1. Open Google Maps. <https://www.google.es/maps/?hl=es>
2. Go to the map (or Street View) of your location.
3. Click Menu (top left).
4. Click Share or embed map.

Mapa

Descripción generada automáticamente con confianza media

1. Click Embed map.

Una captura de pantalla de una computadora

Descripción generada automáticamente

1. Pick the size you want by clicking the down arrow to the left of the text field.
2. Copy the displayed HTML code. Paste it into your web app.

Interfaz de usuario gráfica

Descripción generada automáticamente

1. Now change the style="border:0;" into style={{ border: 0 }}

Insert email client

1. Setup EmailJS. Let's first create a free account in EmailJs <https://dashboard.emailjs.com/sign-up>

In this case, we will create account with info@AppsLeanIT.com

1. On 'Email Services' tab. After clicking on the 'Add New Service' button you should be seeing something similar to the photo below. Click on Gmail (this is what we'll use in this case).Interfaz de usuario gráfica, Texto, Aplicación, Word

   Descripción generada automáticamente
2. To connect the service with your Gmail account click on the 'Connect Account' button. Also, remember your Service ID because we'll need it later (in this case it’s service\_iwzpjtq).

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

1. Associate the gmail account you want to link. Click on the checkbox "Send email on your behalf" to allow us to send the emails through your Gmail account. The new Gmail access popup leaves all permissions disabled by default

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

1. Finally, click on the 'Create Service' and check your inbox to see if you got the test email.Interfaz de usuario gráfica, Aplicación

   Descripción generada automáticamente
2. Check if you received the test email.
3. Let's now create a template. Go to the 'Email Template' tab on the side menu and click on the 'Create New Template' button. For testing purposes, we will use this default one. A caveat, the variables in double curly brackets are dynamic variables that will be replaced with the data we'll provide in the method emailjs.send, in our case, in React. Click 'Save' and we're good to proceed.

Interfaz de usuario gráfica, Texto, Correo electrónico

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación, Chat o mensaje de texto, Correo electrónico

Descripción generada automáticamente

1. Now, we import it (don’t forget to grab the User ID). <https://dashboard.emailjs.com/admin/integration>

import './App.css';

import { useState } from 'react';

import { send } from 'emailjs-com';

function App() {

return (

<div className="App">

...

</div>

);

}

export default App;

1. Now let’s create the form inside the div tags on return statement

            <**form** onSubmit**=**{onSubmit} data-form-title**=**"CONTACT FORM">

                <**div** className**=**"form-group">

                    <**label** className**=**"mb-1">Full Name</**label**>

                    <**input**

                        type**=**"text"

                        className**=**"form-control"

                        name**=**"from\_name"

                        required**=**""

                        placeholder**=**"Nombre\*"

                        data-form-field**=**"Name"

                        value**=**{toSend.from\_name}

                        onChange**=**{handleChange}

                    />

                </**div**>

                <**div** className**=**"form-group">

                    <**label** className**=**"mb-1">Phone Number</**label**>

                    <**input**

                        type**=**"tel"

                        className**=**"form-control phone"

                        name**=**"from\_phone"

                        placeholder**=**"Telefono"

                        data-form-field**=**"Phone"

                        value**=**{toSend.from\_phone}

                        onChange**=**{handleChange}

                    />

                </**div**>

                <**div** className**=**"form-group">

                    <**label** className**=**"mb-1">Email Adress</**label**>

                    <**input**

                        type**=**"email"

                        className**=**"form-control email"

                        name**=**"from\_email"

                        required**=**""

                        placeholder**=**"Email\*"

                        data-form-field**=**"Email"

                        value**=**{toSend.from\_email}

                        onChange**=**{handleChange}

                    />

                </**div**>

                <**div** className**=**"form-group">

                    <**label** className**=**"mb-1">Message</**label**>

                    <**textarea**

                        className**=**"form-control message"

                        name**=**"from\_message"

                        placeholder**=**"Escribe tu mensaje aquí..."

                        rows**=**"7"

                        data-form-field**=**"Message"

                        value**=**{toSend.from\_message}

                        onChange**=**{handleChange}

                    >

                    </**textarea**>

                </**div**>

                <**div** className**=**"form-group py-3">

                    <**button** type **=**"submit" className**=**"btn btn-primary shadow w-100">

                        Enviar Mensaje

                        <**Link** to**=**"/gracias"></**Link**>

                    </**button**>

                </**div**>

            </**form**>

1. Now, let’s create the variables inside the component function but before the return statement.

we used useState() Hook that lets you add React state to function components. We initialized the state with the 'toSend' object with the same instance names as the dynamic ones we have in the emailjs template. We also created two methods are for manipulating form data. handleChange to update the state, and onSubmit to handle the submission.

**const** [toSend, setToSend] **=** useState({

        from\_name:'',

        from\_email:'',

        from\_message:'',

        reply\_to:'',

        to\_name:'',

    });

**const** onSubmit **=** (e) **=>** {

        e.preventDefault();

        {*/\* --- METHOD TO SEND THE EMAIL --- \*/*}

    };

**const** handleChange **=** (e) **=>** {

        setToSend({ **...**toSend, [e.target.name]: e.target.value});

    };

1. The method to send the email onSubmit variable, to send data via emailjs.send method should look like this:

**const** onSubmit **=** (e) **=>** {

        e.preventDefault();

        {*/\* --- METHOD TO SEND THE EMAIL --- \*/*}

        send(

            'SERVICE ID',

            'TEMPLATE ID',

            toSend,

            'User ID'

        )

            .then((response) **=>** {

                console.log('SUCCESS!', response.status, response.text);

            })

            .catch((err) **=>** {

                console.log('FAILED...', err);

            });

    };

1. Add your unique IDs required in the send method that you can find on emailjs dashboard.

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico, Sitio web

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación, Chat o mensaje de texto, Correo electrónico

Descripción generada automáticamente

Interfaz de usuario gráfica, Aplicación, Sitio web

Descripción generada automáticamente

1. The result should be:

        send(

            'service\_iwzpjtq',

            'template\_safus0a',

            toSend,

            'user\_qN1EDLF3yczpYrH5HSbtE'

        )

1. The final code should like the following:

**import** React **from** 'react';

**import** { Link } **from** 'react-router-dom';

**import**{ init } **from** '@emailjs/browser';

**import** { send } **from** 'emailjs-com';

**import** { useState } **from** 'react';

**export** **default** **function** Form() {

**const** [toSend, setToSend] **=** useState({

        from\_name:'',

        from\_email:'',

        from\_message:'',

        reply\_to:'',

        to\_name:'',

    });

**const** onSubmit **=** (e) **=>** {

        e.preventDefault();

        {*/\* --- METHOD TO SEND THE EMAIL --- \*/*}

        send(

            'service\_iwzpjtq',

            'template\_safus0a',

            toSend,

            'user\_qN1EDLF3yczpYrH5HSbtE'

        )

            .then((response) **=>** {

                console.log('SUCCESS!', response.status, response.text);

            })

            .catch((err) **=>** {

                console.log('FAILED...', err);

            });

    };

**const** handleChange **=** (e) **=>** {

        setToSend({ **...**toSend, [e.target.name]: e.target.value});

    };

**return** (

        <**div**>

            <**form** onSubmit**=**{onSubmit} data-form-title**=**"CONTACT FORM">

                <**div** className**=**"form-group">

                    <**label** className**=**"mb-1">Full Name</**label**>

                    <**input**

                        type**=**"text"

                        className**=**"form-control"

                        name**=**"from\_name"

                        required**=**""

                        placeholder**=**"Nombre\*"

                        data-form-field**=**"Name"

                        value**=**{toSend.from\_name}

                        onChange**=**{handleChange}

                    />

                </**div**>

                <**div** className**=**"form-group">

                    <**label** className**=**"mb-1">Phone Number</**label**>

                    <**input**

                        type**=**"tel"

                        className**=**"form-control phone"

                        name**=**"from\_phone"

                        placeholder**=**"Telefono"

                        data-form-field**=**"Phone"

                        value**=**{toSend.from\_phone}

                        onChange**=**{handleChange}

                    />

                </**div**>

                <**div** className**=**"form-group">

                    <**label** className**=**"mb-1">Email Adress</**label**>

                    <**input**

                        type**=**"email"

                        className**=**"form-control email"

                        name**=**"from\_email"

                        required**=**""

                        placeholder**=**"Email\*"

                        data-form-field**=**"Email"

                        value**=**{toSend.from\_email}

                        onChange**=**{handleChange}

                    />

                </**div**>

                <**div** className**=**"form-group">

                    <**label** className**=**"mb-1">Message</**label**>

                    <**textarea**

                        className**=**"form-control message"

                        name**=**"from\_message"

                        placeholder**=**"Escribe tu mensaje aquí..."

                        rows**=**"7"

                        data-form-field**=**"Message"

                        value**=**{toSend.from\_message}

                        onChange**=**{handleChange}

                    >

                    </**textarea**>

                </**div**>

                <**div** className**=**"form-group py-3">

                    <**button** type **=**"submit" className**=**"btn btn-primary shadow w-100">

                        Enviar Mensaje

                        <**Link** to**=**"/gracias"></**Link**>

                    </**button**>

                </**div**>

            </**form**>

        </**div**>

    )

}

1. Optimize the templates to automatically send a message to you and to the contact who filled in the message.

Interfaz de usuario gráfica, Texto

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

1. Validate fields in the form to make sure you avoid send it empty or with lack of filled fields.

    {*/\* --- VALIDATE FORM FIELDS --- \*/*}

**const** schema **=** Yup.object().shape({

        from\_name: Yup.string().required('Introduce el nombre.'),

        from\_phone: Yup.number().min(8).positive().integer().required('Introduce un numero de telefono valido.'),

        from\_email: Yup.string().email().required('Introduce un email valido.'),

        from\_message: Yup.string().required('Introduce un mensaje.'),

    });

**const**{register,handleSubmit,formState: { errors }} **=** useForm({

        resolver:yupResolver(schema),

    });

**const** submitForm **=** (e) **=>** {

        e.preventDefault();

        {*/\* --- DISPLAY VALIDATION ERRORS --- \*/*}

        console.log(e);

1. On the form, insert the functions handleSubmit and submitForm to action when the form is summited.

          <**form** onSubmit**=**{handleSubmit, submitForm} data-form-title**=**"CONTACT FORM">

1. In each input, introduce the validation with:

{**...**register("name\_of\_the\_field\_to\_validate")}

<**p**>{errors.name\_of\_the\_field\_to\_validate?.message}</**p**>

Here an example:

                <**div** className**=**"form-group">

                    <**label** className**=**"mb-1">Nombre</**label**>

                    <**input**

                        type**=**"text"

                        className**=**"form-control"

                        name**=**"from\_name"

                        required

                        placeholder**=**"Nombre\*"

                        data-form-field**=**"Name"

                        {**...**register("from\_name")}

                        value**=**{toSend.from\_name}

                        onChange**=**{handleChange}

                    />

                    <**p**>{errors.from\_name?.message}</**p**>

                </**div**>

1. Redirect to another page, once the form is validated and the button sumited by importing the history through useNavigate

import { useNavigate } from "react-router-dom";

1. Started the useNavigate in the code

**const** navigate **=** useNavigate();

1. Introduce the navigation path on the submitForm function when the validation is correct

            .then((response) **=>** {

                console.log('SUCCESS!', response.status, response.text);

                navigate('/gracias');

            })

1. Now that we have everything working, here is the final code for the form:

**import** React **from** 'react';

**import** { init } **from** '@emailjs/browser';

**import** { send } **from** 'emailjs-com';

**import** { useState } **from** 'react';

**import** {

   Link,

   useNavigate

} **from** "react-router-dom";

**import** { useForm } **from** 'react-hook-form';

**import** { yupResolver } **from** '@hookform/resolvers/yup';

**import** \* **as** Yup **from** 'yup';

**export** **default** **function** Form() {

**const** navigate **=** useNavigate();

    {*/\* --- SET INITIAL DATA FIELDS --- \*/*}

**const** [toSend, setToSend] **=** useState({

        from\_name:'',

        from\_phone:'',

        from\_email:'',

        from\_message:'',

        reply\_to:'',

        to\_name:'',

    });

    {*/\* --- VALIDATE FORM FIELDS --- \*/*}

**const** schema **=** Yup.object().shape({

        from\_name: Yup.string().required('Introduce el nombre.'),

        from\_phone: Yup.number().min(8).positive().integer().required('Introduce un numero de telefono valido.'),

        from\_email: Yup.string().email().required('Introduce un email valido.'),

        from\_message: Yup.string().required('Introduce un mensaje.'),

    });

**const**{register,handleSubmit,formState: { errors }} **=** useForm({

        resolver:yupResolver(schema),

    });

**const** submitForm **=** (e) **=>** {

        e.preventDefault();

        {*/\* --- DISPLAY VALIDATION ERRORS --- \*/*}

        console.log(e);

        {*/\* --- METHOD TO SEND THE EMAIL --- \*/*}

        send(

            'service\_iwzpjtq',

            'template\_safus0a',

            toSend,

            'user\_qN1EDLF3yczpYrH5HSbtE'

        )

            .then((response) **=>** {

                console.log('SUCCESS!', response.status, response.text);

                navigate('/gracias');

            })

            .catch((err) **=>** {

                console.log('FAILED...', err);

            });

    };

**const** handleChange **=** (e) **=>** {

        setToSend({ **...**toSend, [e.target.name]: e.target.value});

    };

**return** (

        <**div**>

            <**form** onSubmit**=**{handleSubmit, submitForm} data-form-title**=**"CONTACT FORM">

                <**div** className**=**"form-group">

                    <**label** className**=**"mb-1">Nombre</**label**>

                    <**input**

                        type**=**"text"

                        className**=**"form-control"

                        name**=**"from\_name"

                        required

                        placeholder**=**"Nombre\*"

                        data-form-field**=**"Name"

                        {**...**register("from\_name")}

                        value**=**{toSend.from\_name}

                        onChange**=**{handleChange}

                    />

                    <**p**>{errors.from\_name?.message}</**p**>

                </**div**>

                <**div** className**=**"form-group">

                    <**label** className**=**"mb-1">Telefono</**label**>

                    <**input**

                        type**=**"tel"

                        className**=**"form-control phone"

                        name**=**"from\_phone"

                        required

                        placeholder**=**"Telefono"

                        data-form-field**=**"Phone"

                        {**...**register("from\_phone")}

                        value**=**{toSend.from\_phone}

                        onChange**=**{handleChange}

                    />

                    <**p**>{errors.from\_phone?.message}</**p**>

                </**div**>

                <**div** className**=**"form-group">

                    <**label** className**=**"mb-1">Correo electronico</**label**>

                    <**input**

                        type**=**"email"

                        className**=**"form-control email"

                        name**=**"from\_email"

                        required

                        placeholder**=**"Email\*"

                        data-form-field**=**"Email"

                        {**...**register("from\_email")}

                        value**=**{toSend.from\_email}

                        onChange**=**{handleChange}

                    />

                    <**p**>{errors.from\_email?.message}</**p**>

                </**div**>

                <**div** className**=**"form-group">

                    <**label** className**=**"mb-1">Mensaje</**label**>

                    <**textarea**

                        className**=**"form-control message"

                        name**=**"from\_message"

                        required

                        placeholder**=**"Escribe aquí el mensaje ..."

                        rows**=**"7"

                        data-form-field**=**"Message"

                        {**...**register("from\_message")}

                        value**=**{toSend.from\_message}

                        onChange**=**{handleChange}

                    >

                    </**textarea**>

                    <**p**>{errors.from\_message?.message}</**p**>

                </**div**>

                <**div** className**=**"form-group py-3">

                    <**button** className**=**"btn btn-primary shadow w-100">

                        Enviar Mensaje

                    </**button**>

                </**div**>

            </**form**>

        </**div**>

    )

}

**Caroussel Component:**

1. Search for carousel into <https://getbootstrap.com/docs/5.1/components/carousel/> and look for the template code.
2. In this case, let’s choose the 3rd code and after creating the template react code with rfc+enter, paste this code inside the return sentence of the function.
3. Align everything and correct whatever syntax error exist.

**import** React **from** 'react'

**export** **default** **function** Slider() {

**return** (

        <**div**>

            <**div** id**=**"carouselExampleIndicators" class**=**"carousel slide" data-bs-ride**=**"carousel">

                <**div** class**=**"carousel-indicators">

                    <**button** type**=**"button" data-bs-target**=**"#carouselExampleIndicators" data-bs-slide-to**=**"0" class**=**"active" aria-current**=**"true" aria-label**=**"Slide 1"></**button**>

                    <**button** type**=**"button" data-bs-target**=**"#carouselExampleIndicators" data-bs-slide-to**=**"1" aria-label**=**"Slide 2"></**button**>

                    <**button** type**=**"button" data-bs-target**=**"#carouselExampleIndicators" data-bs-slide-to**=**"2" aria-label**=**"Slide 3"></**button**>

                </**div**>

                <**div** class**=**"carousel-inner">

                    <**div** class**=**"carousel-item active">

                        <**img** src**=**"..." class**=**"d-block w-100" alt**=**"..."/>

                    </**div**>

                    <**div** class**=**"carousel-item">

                        <**img** src**=**"..." class**=**"d-block w-100" alt**=**"..."/>

                    </**div**>

                    <**div** class**=**"carousel-item">

                        <**img** src**=**"..." class**=**"d-block w-100" alt**=**"..."/>

                    </**div**>

                </**div**>

                <**button** class**=**"carousel-control-prev" type**=**"button" data-bs-target**=**"#carouselExampleIndicators" data-bs-slide**=**"prev">

                    <**span** class**=**"carousel-control-prev-icon" aria-hidden**=**"true"></**span**>

                    <**span** class**=**"visually-hidden">Previous</**span**>

                </**button**>

                <**button** class**=**"carousel-control-next" type**=**"button" data-bs-target**=**"#carouselExampleIndicators" data-bs-slide**=**"next">

                    <**span** class**=**"carousel-control-next-icon" aria-hidden**=**"true"></**span**>

                    <**span** class**=**"visually-hidden">Next</**span**>

                </**button**>

            </**div**>

        </**div**>

    )

}

1. Import the images and link them in slider

**import** React **from** 'react';

**import** hospital **from** '../images/hospital.jpg';

**import** wood **from** '../images/wood.jpg';

**import** repair **from** '../images/repair.jpg';

**export** **default** **function** Slider() {

**return** (

        <**div**>

            <**div** id**=**"carouselExampleIndicators" className**=**"carousel slide" data-bs-ride**=**"carousel">

                <**div** className**=**"carousel-indicators">

                    <**button** type**=**"button" data-bs-target**=**"#carouselExampleIndicators" data-bs-slide-to**=**"0" className**=**"active" aria-current**=**"true" aria-label**=**"Slide 1"></**button**>

                    <**button** type**=**"button" data-bs-target**=**"#carouselExampleIndicators" data-bs-slide-to**=**"1" aria-label**=**"Slide 2"></**button**>

                    <**button** type**=**"button" data-bs-target**=**"#carouselExampleIndicators" data-bs-slide-to**=**"2" aria-label**=**"Slide 3"></**button**>

                </**div**>

                <**div** className**=**"carousel-inner">

                    <**div** className**=**"carousel-item active">

                        <**img** src**=**{hospital} className**=**"d-block w-100" alt**=**"hospital"/>

                    </**div**>

                    <**div** className**=**"carousel-item">

                        <**img** src**=**{wood} className**=**"d-block w-100" alt**=**"wood"/>

                    </**div**>

                    <**div** className**=**"carousel-item">

                        <**img** src**=**{repair} className**=**"d-block w-100" alt**=**"repair"/>

                    </**div**>

                </**div**>

                <**button** className**=**"carousel-control-prev" type**=**"button" data-bs-target**=**"#carouselExampleIndicators" data-bs-slide**=**"prev">

                    <**span** className**=**"carousel-control-prev-icon" aria-hidden**=**"true"></**span**>

                    <**span** className**=**"visually-hidden">Previous</**span**>

                </**button**>

                <**button** className**=**"carousel-control-next" type**=**"button" data-bs-target**=**"#carouselExampleIndicators" data-bs-slide**=**"next">

                    <**span** className**=**"carousel-control-next-icon" aria-hidden**=**"true"></**span**>

                    <**span** className**=**"visually-hidden">Next</**span**>

                </**button**>

            </**div**>

        </**div**>

    )

}

1. Import the slider into Home page

**import** React **from** 'react'

**import** Slider **from** '../inc/Slider';

**export** **default** **function** Home() {

**return** (

        <**div**>

            <**Slider** />

            <**div** className**=**"container">

                <**div** className**=**"card mt-4">

                    <**div** className**=**"card-body">

                        <**h2**>Home Page</**h2**>

                    </**div**>

                </**div**>

            </**div**>

        </**div**>

    )

}

**Content and Design:**

1. Navbar:

* Change all “class” to “className”
* Changing into dark navbar:

1. Home Page:

* Create 3 sections with titles and content:

            <**section** className**=**"section">

                <**div** className**=**"container">

                    <**div** className**=**"row">

                        <**div** className**=**"col-md-12 text-center">

                            <**h3** className**=**"main-heading">Our Company</**h3**>

                            <**div** className**=**"underline mx-auto"></**div**>

                            <**p**>bla bla bla</**p**>

                            <**Link** to**=**"/about" className**=**"btn btn-warning shadow">Read More</**Link**>

                        </**div**>

                    </**div**>

                </**div**>

            </**section**>

The App.css file has the following styles:

*/\* \*\*\*\*\*\*\*\*\*\*\*Content styles\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* \*/*

**p**{

    font-size: 14px;

    color: #262626;

}

.section{

    padding-top: 40px;

    padding-bottom: 40px;

}

.main-heading{

    color: rgb(204, 18, 18);

    font-size: 26px;

}

.underline{

    height: 04px;

    width: 4rem;

    background-color: #f76c2f;

    margin-top: 10px;

    margin-bottom: 10px;

}

.bg-c-light{

    background-color: #f2f2f2;

}

#imagencarda{

    height: 100;

    width: 100;

}

*/\* \*\*\*\*\*\*\*\*\*\*\*Footer styles\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* \*/*

.footer **a**{

    color: #fff;

    text-decoration: none !Important;

}

.footer **a**:hover{

    color: grey;

}

*/\* \*\*\*\*\*\*\*\*\*\*\*SocialFollow styles\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* \*/*

.social-container {

    text-align: center;

    font-size: 12px;

    color: #fff;

    font-weight: bold;

    text-decoration: none !Important;

    bottom: 0;

    width: 100%;

    display: flex;

    flex-direction: line;

    position: sticky;

    justify-content: center;

    margin: 0 auto;

    margin-bottom: 1px;

    padding: 0px 100px

  }

**a**.social {

    margin: 0 1rem;

    transition: transform 250ms;

    display: inline-block;

  }

**a**.social:hover {

    transform: translateY(-2px);

  }

**a**.youtube {

    color: #eb3223;

  }

**a**.facebook {

    color: #4968ad;

  }

**a**.twitter {

    color: #49a1eb;

  }

**a**.instagram {

    color: white;

  }

**a**.linkedin {

    color: #4968ad;

  }

**a**.whatsapp {

    color: green;

  }

**a**.skype {

    color: #4968ad;

  }

**a**.web {

    color: #4968ad;

  }

*/\* \*\*\*\*\*\*\*\*\*\*\*Slider styles\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* \*/*

#carouselExampleIndicators {

    display: block;

    margin-left: auto;

    margin-right: auto;

    width: 500px;

}

**img**{

    border-radius: 8px;

    padding: 5px;

}

**Save into a Git repository**

1. Initialize the git, so write in the command line: git init
2. Now we want to add all the files, so we write: git add . (. To add all files)
3. Now write: git commit -m “initial commit” (m refers to message)
4. Go to <https://github.com/solangejmendes?tab=repositories> and login into the repositories. In this case I logged in with solangejmendes@gmail.com
5. Click in create new repository
6. Give a name to the repository. In this case we are going to call it: React\_aceaire
7. Click on Create Repository
8. Once is created the repository, go to the code button and copy the route for push propose

Captura de pantalla de computadora

Descripción generada automáticamente

1. Go into the command line again and paste the remote configuration command

Git remote add origin https://github.com/solangejmendes/React\_aceaire.git

1. Now we can push the application into github in a remote way by writing the following in the command line: git push -u origin master
2. Update your local repo from the central repo (git pull upstream master).
3. Make edits,
4. save,
5. git add, (git add .)
6. and git commit all in your local repo. (git commit -m “buildvs1”)
7. Push changes from local repo to your fork on github.com (git push -u origin master)
8. Update the central repo from your fork (Pull Request)

**Deploy de React Application**

1. Write on the command line: npm run build
2. It’s going to give a couple of options.
3. We’ll need top install server writing in the command line: npm install -g serve
4. Now we need to create a new folder into the application so we write: serve -s build
5. A build folder is created with all information needed there
6. Ctr + c to close the server
7. Get into <https://app.netlify.com/signup/start> and login. In this case I logged in with [info@AppsLeanIT.com](mailto:info@AppsLeanIT.com)
8. Click on import from git

Interfaz de usuario gráfica, Texto, Sitio web

Descripción generada automáticamente

1. Click in Github

Captura de pantalla de computadora

Descripción generada automáticamente

1. Click in authorize Netlify
2. Give authorization to all repositories

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

1. Search for the project you want to deploy

Captura de pantalla de computadora

Descripción generada automáticamente

1. Fill in the fields

Captura de pantalla de computadora

Descripción generada automáticamente

1. Click deploy site
2. It will be depoyed successfully if it’s in production

Interfaz de usuario gráfica, Sitio web

Descripción generada automáticamente

1. Go into deploys tab and deploys settings and select domain management

Interfaz de usuario gráfica, Sitio web

Descripción generada automáticamente

1. Click on add custom domain and write your domain. In this case it is [www.ACE-aire.com](http://www.ACE-aire.com). Click on verify.

Interfaz de usuario gráfica, Texto, Aplicación, Sitio web

Descripción generada automáticamente

1. If it’s registered in another place, click in add domain.
2. Configure the DNS. So login into <https://www.godaddy.com/> in this case with the account [info@LeanSixSigmaSis.com](mailto:info@LeanSixSigmaSis.com).

Interfaz de usuario gráfica, Texto, Correo electrónico

Descripción generada automáticamente

1. Go to Administrar DNS

Interfaz de usuario gráfica, Texto, Aplicación, Word

Descripción generada automáticamente

1. Change the configuration according with the information provided by Netlify.

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto

Descripción generada automáticamente

1. Configure the your DNS in your domain provider to also point ALIAS, ANAME, or flattened CNAME record as well. So on Netlify, get the configuration:

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

1. Now go to your domain provider (in my case, goddady) and introduce those configurations in DNS configuration.

Una captura de pantalla de una computadora

Descripción generada automáticamente

1. Select the right option

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

1. To secure the site, go to <https://letsencrypt.org/>

    <**div** className**=**"container">

      <**div** class**=**"card shadow">

        <**h1**>App</**h1**>

        <**div** class**=**"card-body">

          <**button** className**=**"btn btn-primary">Click Me</**button**>

        </**div**>

      </**div**>

    </**div**>

**Link y NavLink**

1. Create a folder inside src called components
2. Create a files inside this folder called: Navbar.js
3. In AppRouter.js file and between Router and Switch, we introduce the tag Navbar. <**Navbar** />
4. We come back to Navbar.js
5. Let’s import Link from react-router-dom: **import** { Link } **from** 'react-router-dom';
6. Now we can create the list of links in a way that each page doesn’t load all content and we have a one-single-application

**import** { Link } **from** 'react-router-dom';

**export** **default** **function** Navbar() {

**return** (

        <**nav**>

            <**ul**>

                <**li**>

                    <**Link** to**=**"/">Home</**Link**>

                </**li**>

                <**li**>

                    <**Link** to**=**"/about">About</**Link**>

                </**li**>

                <**li**>

                    <**Link** to**=**"/contact">Contact</**Link**>

                </**li**>

            </**ul**>

        </**nav**>

    )

}

1. Now we can go to index.css and put some styles in our page. We can add the style:

.active {

    background-color: red;

    color: white;

}

1. To make sure our Navbar.js component receives the styles from index.css, we should change “Link” by “NavLink” and add the class in the tag by activeClassName=”active”
2. To make sure each path has the correspondent style we should put “exact” before “to”.

**import** { Link, NavLink } **from** 'react-router-dom';

**export** **default** **function** Navbar() {

**return** (

        <**nav**>

            <**ul**>

                <**li**>

                    <**NavLink** exact to**=**"/" activeClassName**=**"active">Home</**NavLink**>

                </**li**>

                <**li**>

                    <**NavLink** exact to**=**"/about" activeClassName**=**"active">About</**NavLink**>

                </**li**>

                <**li**>

                    <**NavLink** exact to**=**"/contact" activeClassName**=**"active">Contact</**NLink**>

                </**li**>

            </**ul**>

        </**nav**>

    )

}

**Parámetros**

1. Create a file inside the pages folder called: ProfilePage.js
2. In AppRouter.js file lets add the ProfilePage route which will be variable according with the user so we put :username in front of the path.

<**Route** exact path**=**"/profile/:username" component**=**{ProfilePage}/>

**Querys**

1. Create a file inside the pages folder called: CategoriesPage.js
2. We the page writing rfc + enter, deleting the import sentence and adding a h1 tag sentence with the name of the page.
3. Add a new link in the navbar for categories.

   <**li**>

     <**NavLink** exact to**=**"/categories" activeClassName**=**"active">Categories</**NavLink**>

   </**li**>

1. Go to AppRouter.js and add a Route with the categories.

<**Route** exact path**=**"/categories" component**=**{CategoriesPage}/>

1. Go to CategoriesPage and write the following

**import** { useLocation, useHistory } **from** 'react-router-dom'

**export** **default** **function** CategoriesPage() {

**const** location **=** useLocation();

**const** history **=** useHistory();

**const** query **=** **new** URLSearchParams(location.search)

*// parseInt para pasar el valor a entero y así poder sumarlos más adelante*

**const** skip **=** parseInt(query.get("skip")) **||** 0

**const** limit **=** parseInt(query.get("limit")) **||** 15

**const** handleNext **=** () **=>** {

        console.log("Next")

*// if 2 parameters (skpi and limit)*

*//query.set("skip", 100)*

*//query.set("limit", 200)*

        query.set("skip", skip **+** limit)

        history.push({pathname: "/categories", search:query.toString()})

    }

**return** (

        <**div**>

            <**h1**>CategoriesPage</**h1**>

            <**h2**>Skip: {skip}</**h2**>

            <**h2**>Limit: {limit}</**h2**>

            <**button** onClick**=**{handleNext}>

                Next

            </**button**>

        </**div**>

    )

}

1. This way you can query 2 parameters: skip and limit

**Autenticación**

1. Create 4 pages: LoginPage; RegisterPage; DashboardPage ; and PaymentsPage
2. Build a rute to those pages in AppRouter.js
3. Create a link in Navbar for each of the pages
4. Now lets create a PrivateRoute.js fle at same level as AppRouter.js file change Route by PrivateRoute in AppRouter.js file for those pages which we want to limit the access. We will limit access in 2 pages:

<**PrivateRoute** exact path**=**"/dashboard" component**=**{DashboardPage}/>

<**PrivateRoute** exact path**=**"/payments" component**=**{PaymentsPage}/>

1. We come back to PrivateRoute.js and return the following propreties.

**import** { Route } **from** 'react-router-dom';

**export** **default** **function** PrivateRoute(props) {

**return** (

*<Route exact={props.exact} path={props.path} component={props.component} />*

    );

}

1. Now we can desestructure the props by putting the following:

**import** { Route } **from** 'react-router-dom';

**export** **default** **function** PrivateRoute(props) {

**return** (

        <**Route** {**...**props} />

    );

}

1. If user is loged in should keep at DashboardPage, else should redirect to LoginPage.

**export** **default** **function** PrivateRoute({ component: Component, **...**rest }) {

**return** <**Route** {**...**rest}>{user **?**  **:** <**Redirect** to**=**"/login" />}</**Route**>;

}

1. Now we check if the user is logged in. If it’s logged ingo to component, else go to login page.

**import** { Route, Redirect } **from** 'react-router-dom';

**const** user **=** null;

*//const user = {id: 1, username: "luis50"}*

**export** **default** **function** PrivateRoute({ component: Component, **...**rest }) {

**return** (

        <**Route** {**...**rest}>

            {user **?**  <**Component** /> **:** <**Redirect** to**=**"/login" />}

        </**Route**>

    );

}

1. Si le metemos un usuario quitando el comentario de *const user = {id: 1, username: "luis50"}* ya nos lleva al DashboardPage.