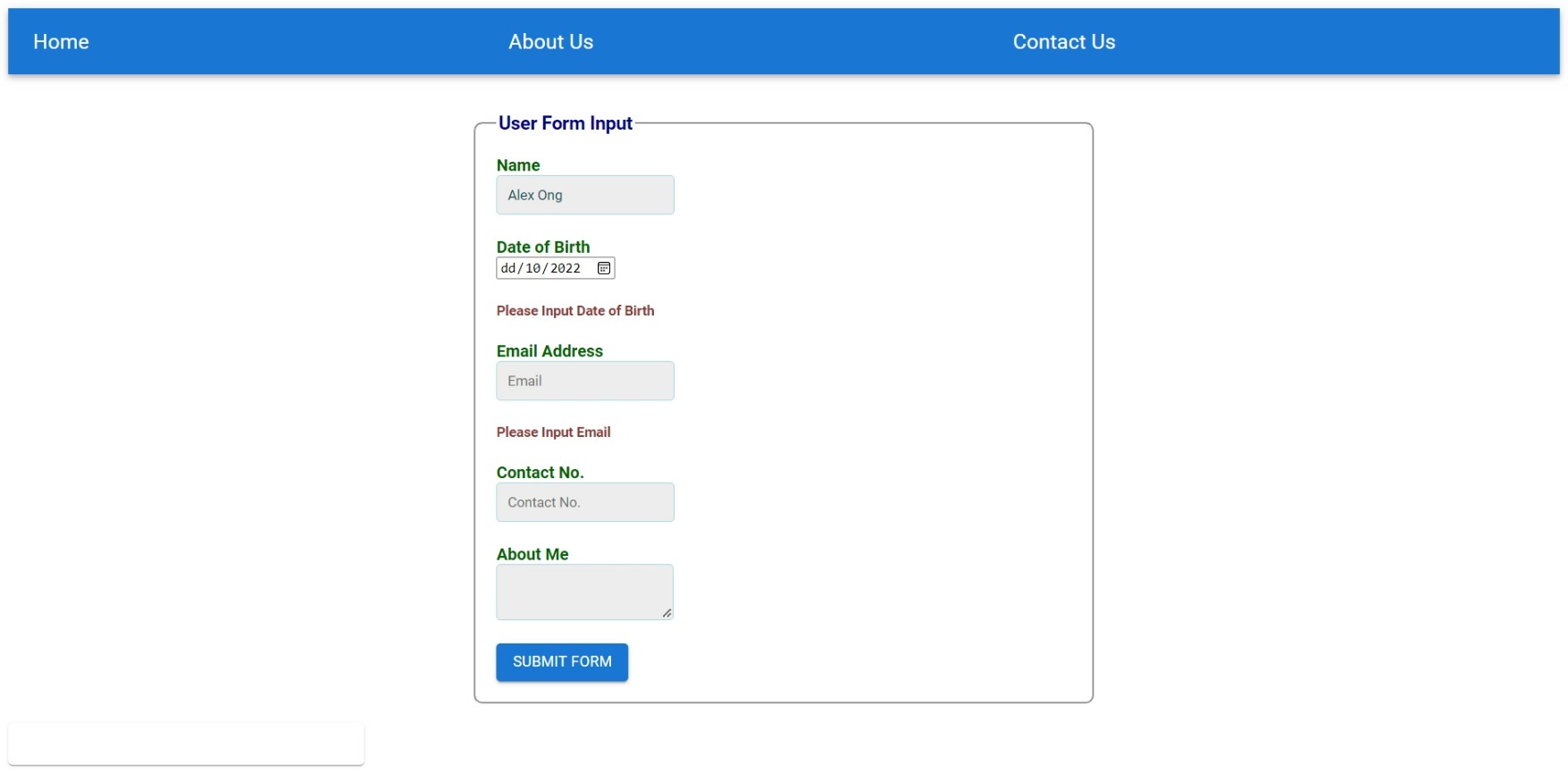
**Geco ReactJS Training**

|  |  |
| --- | --- |
| ReactJS Assignment |  |
| Name: | Ho Choo Geok |
| Date of Issue: | 21st October 2022 |
| Date of Submission: | 24th October 2022 |
| Cohort No.: | 27 |
| Week No.: | 2 |
| Weekly Task No.: | 2 |
| Session No.: | 10 |

1. Create a webpage using ReactJs
2. Create a Navbar (add home, about us, contact menu)
3. Create a form that will take following input from the user:

Name, Date of birth, Email Id, Contact Number, Tell me about yourself

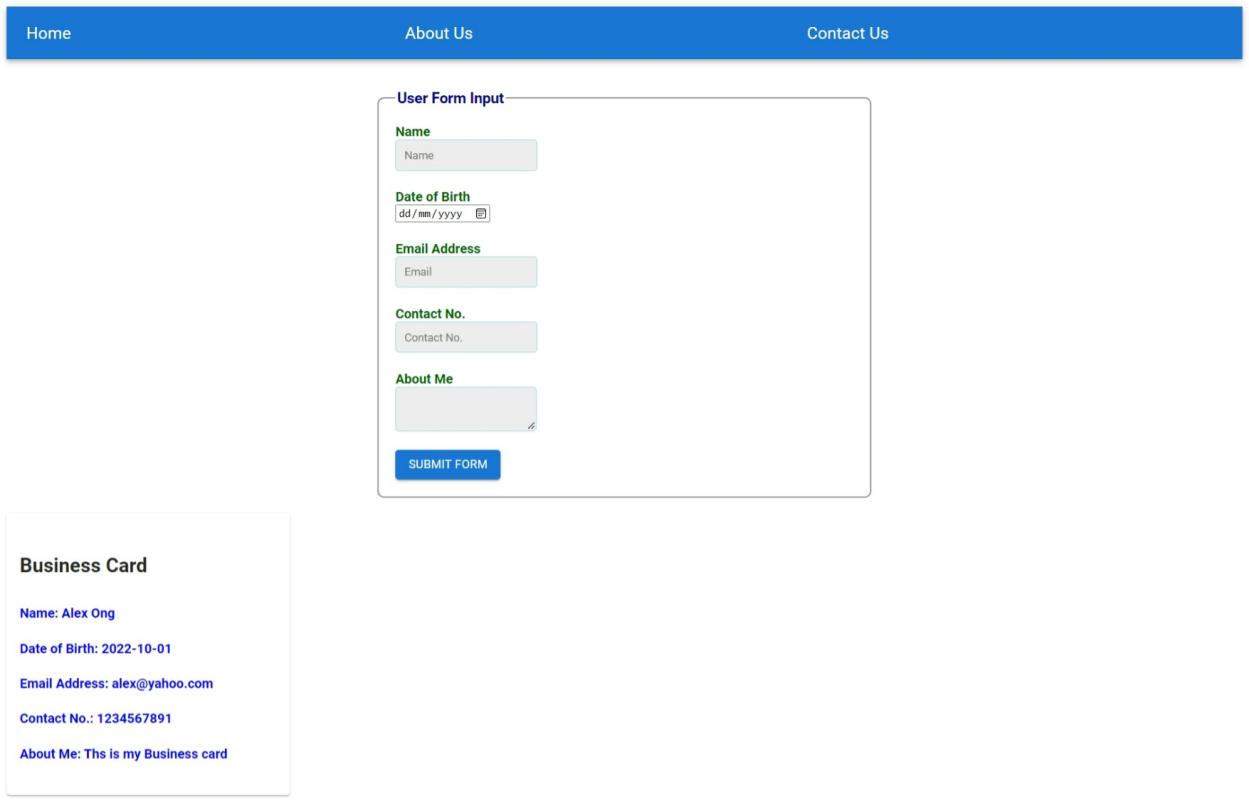
**Solution**

The form with the required input fields are created. The screen capture as follow.

Button created using MUI

Nav Bar created using MUI

**1c) Display user input in a card just below the form (when a form is submitted).**

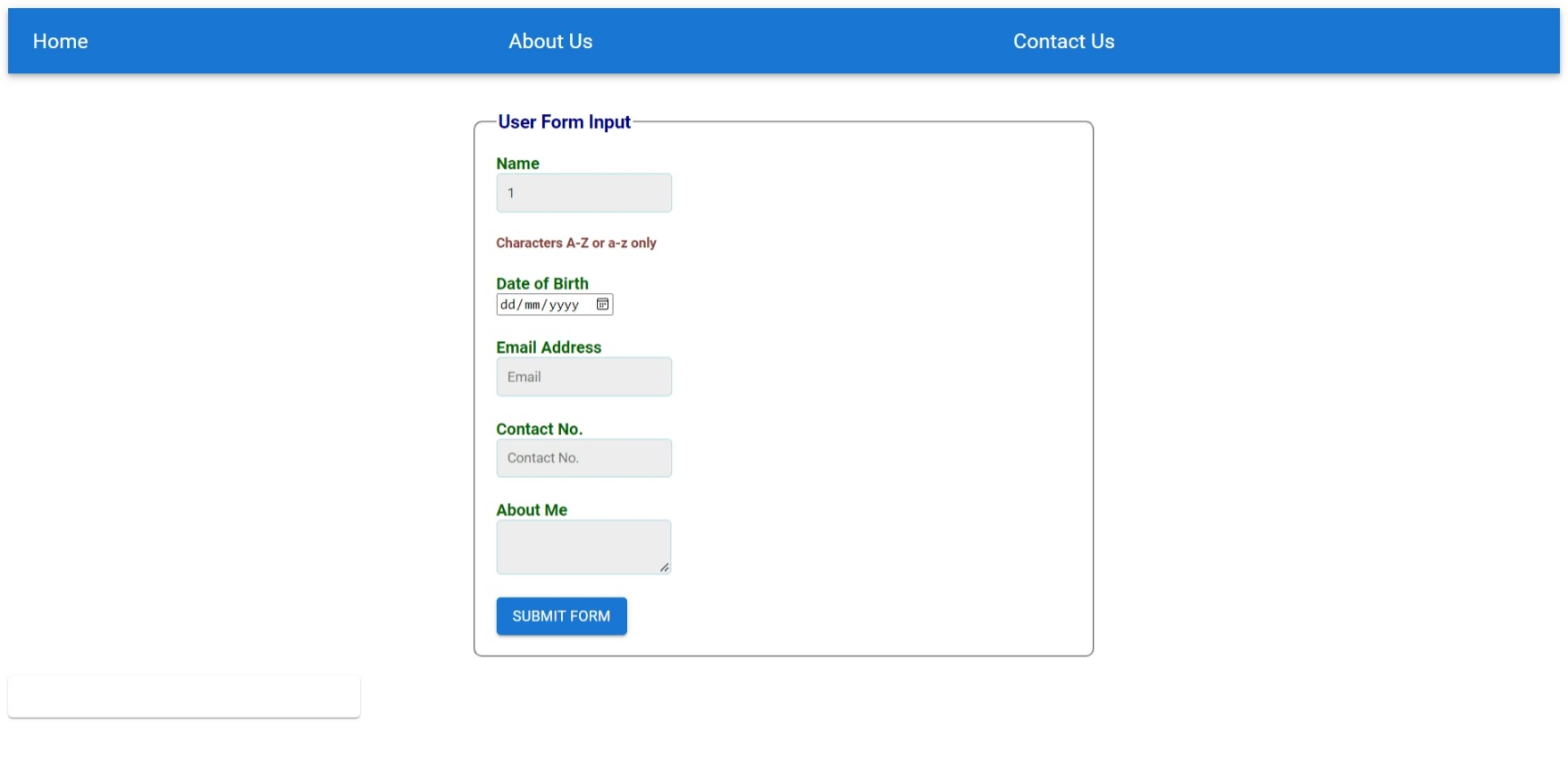
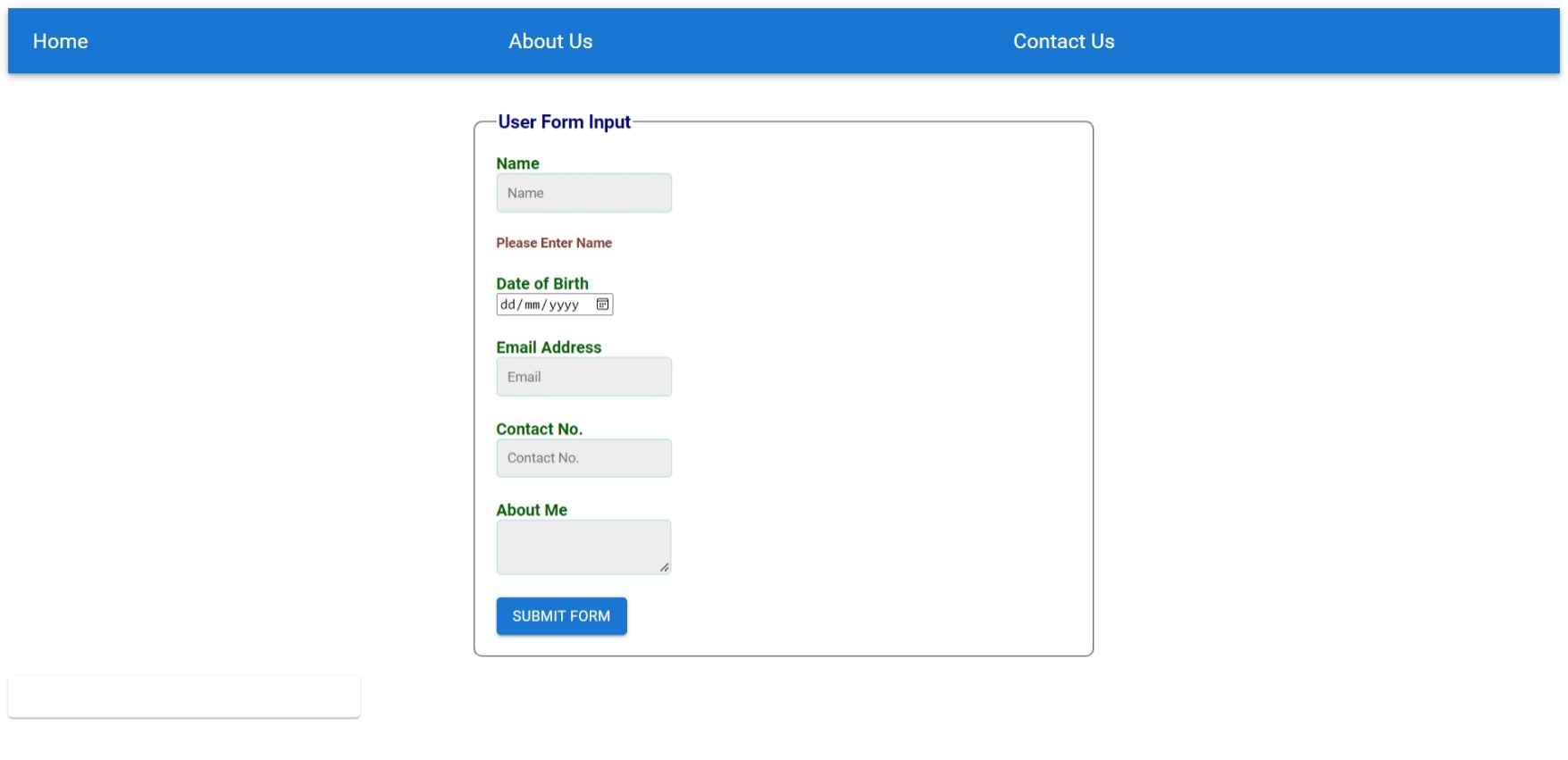


Input Details is displayed at the card created using MUI

**1d) Validation of Input Fields**

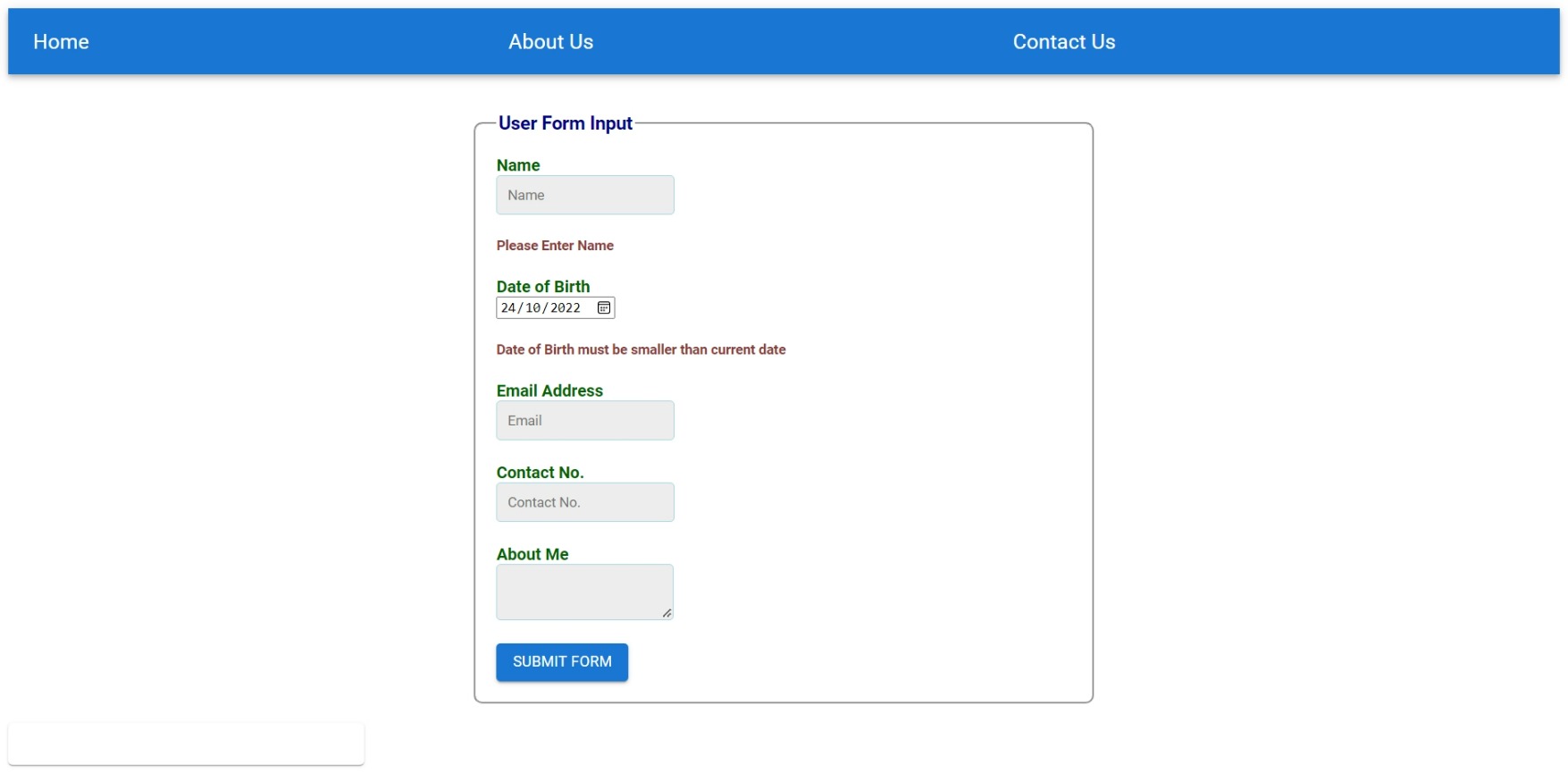
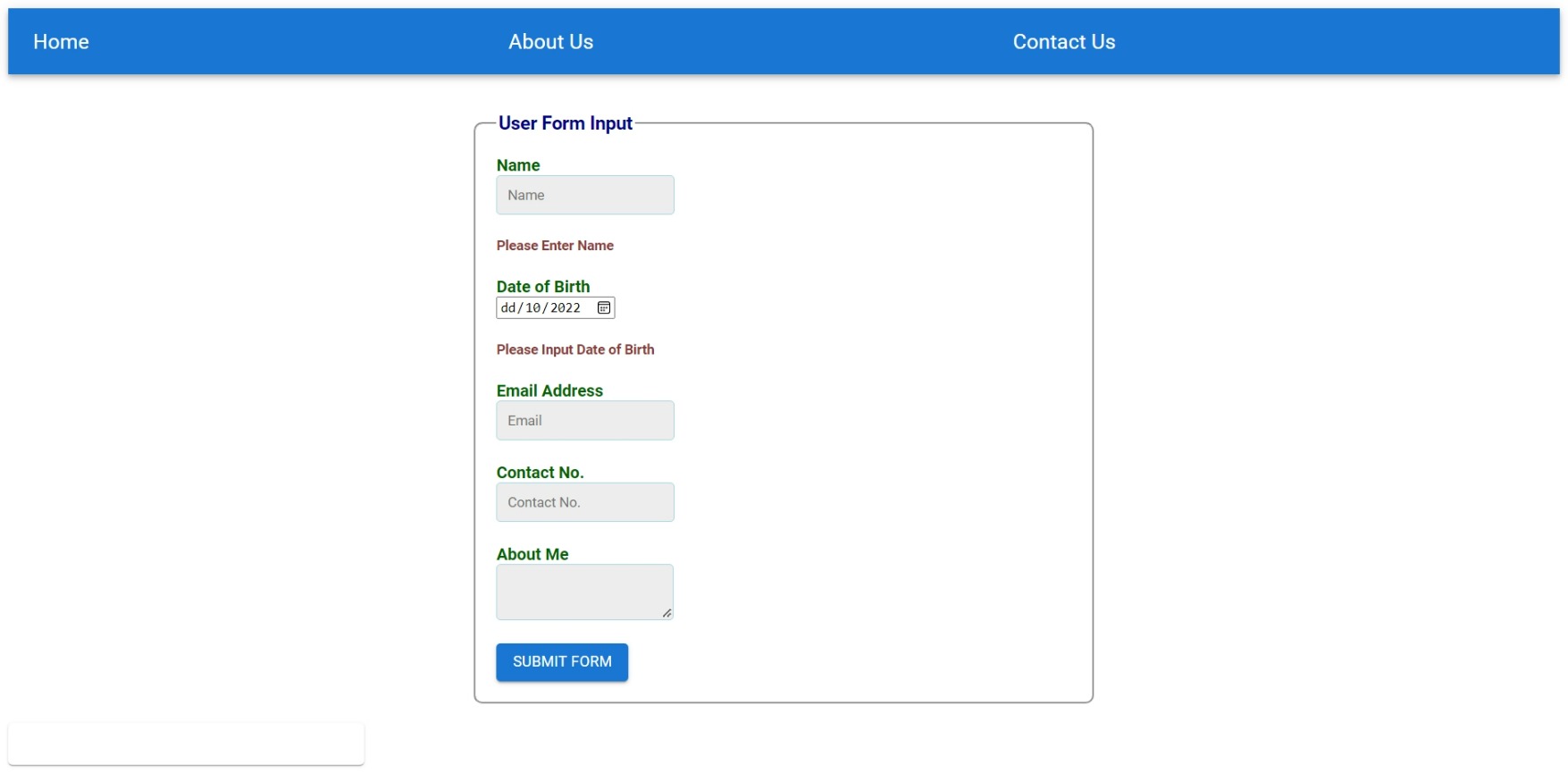
**Name Validation**

1) Validation for empty string 2) Validation for characters input only



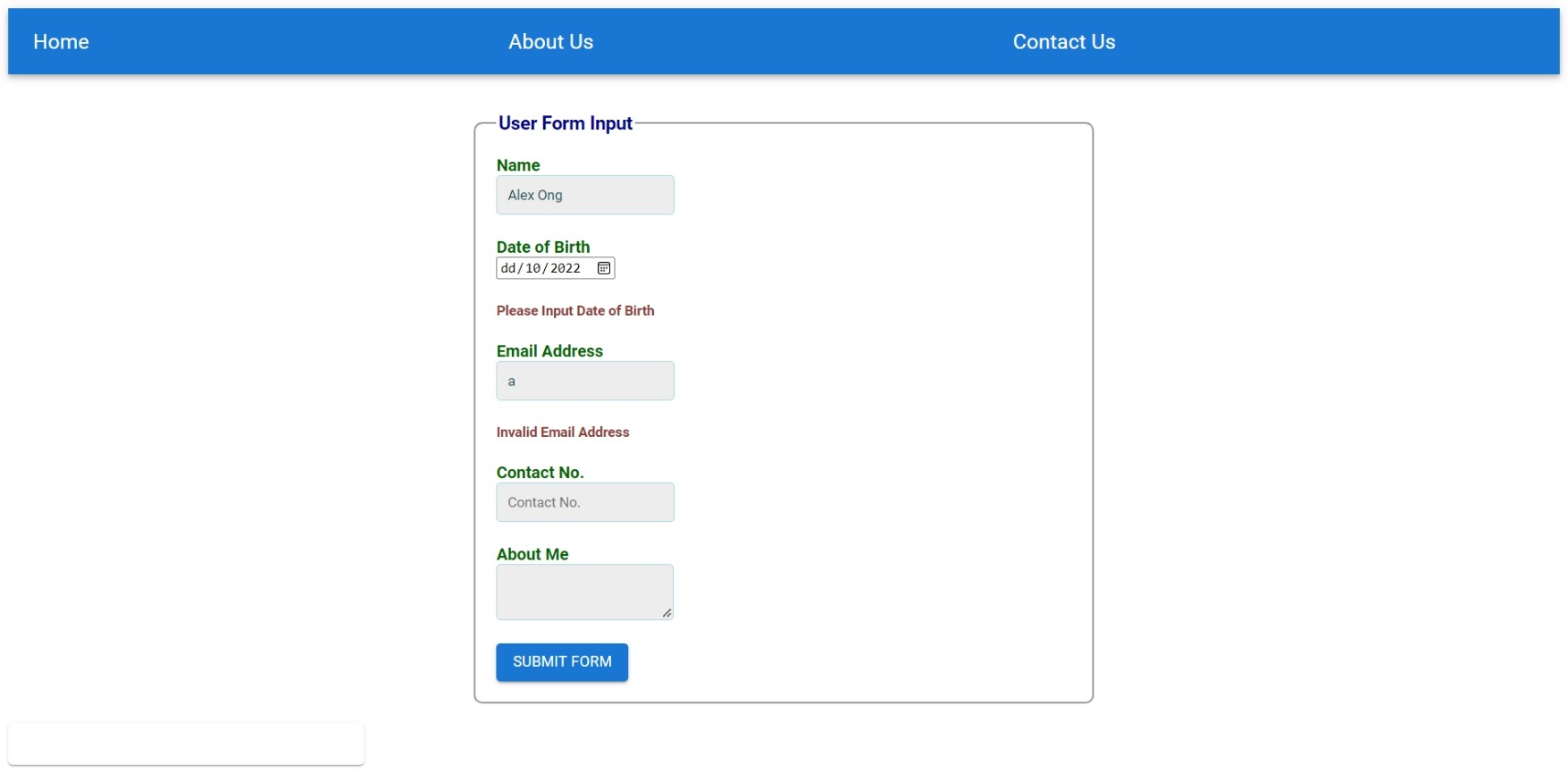
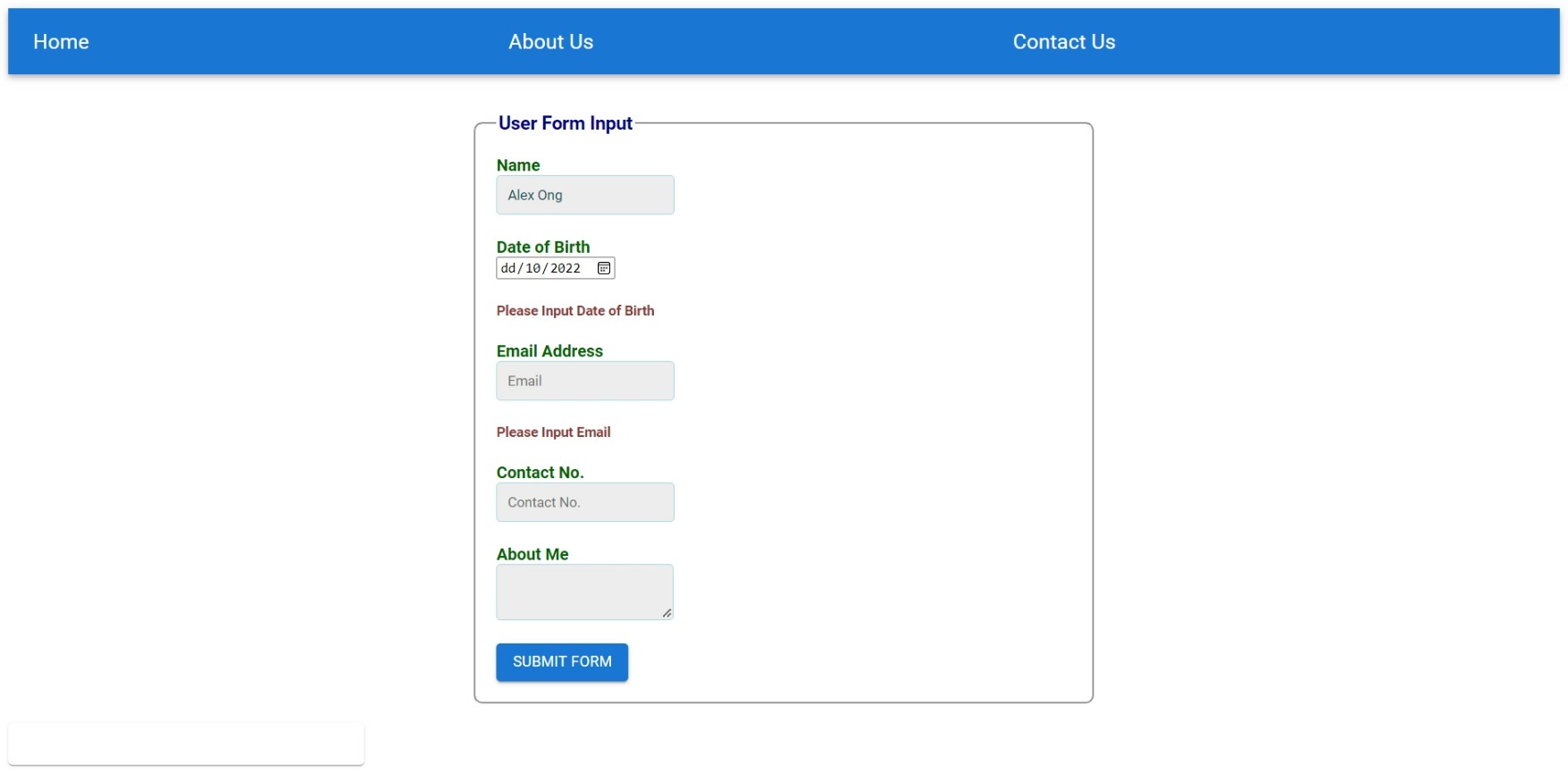
**Date of Birth Validation**

1. Validation for invalid date 2) Validation for Date greater than current Date



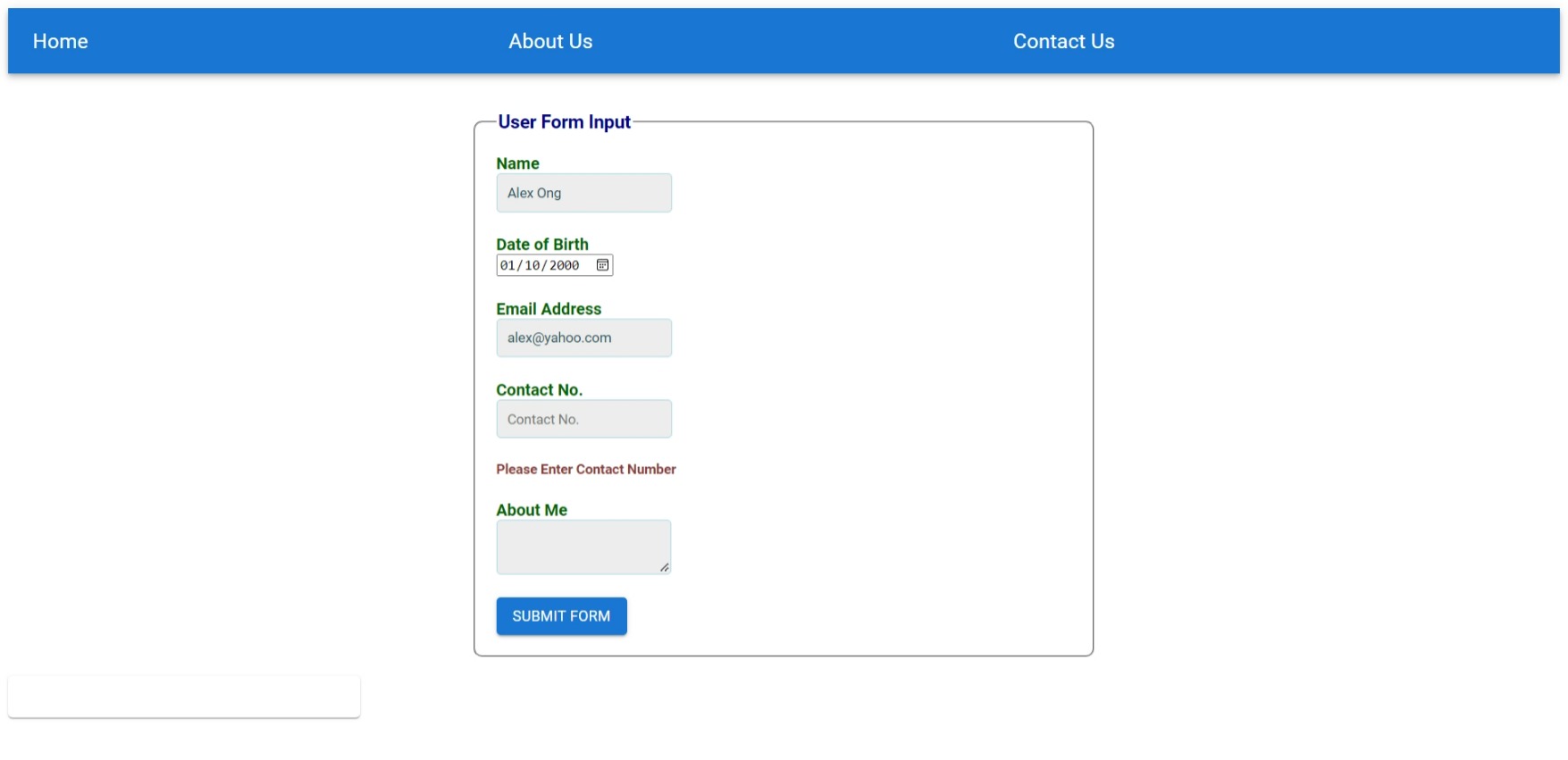
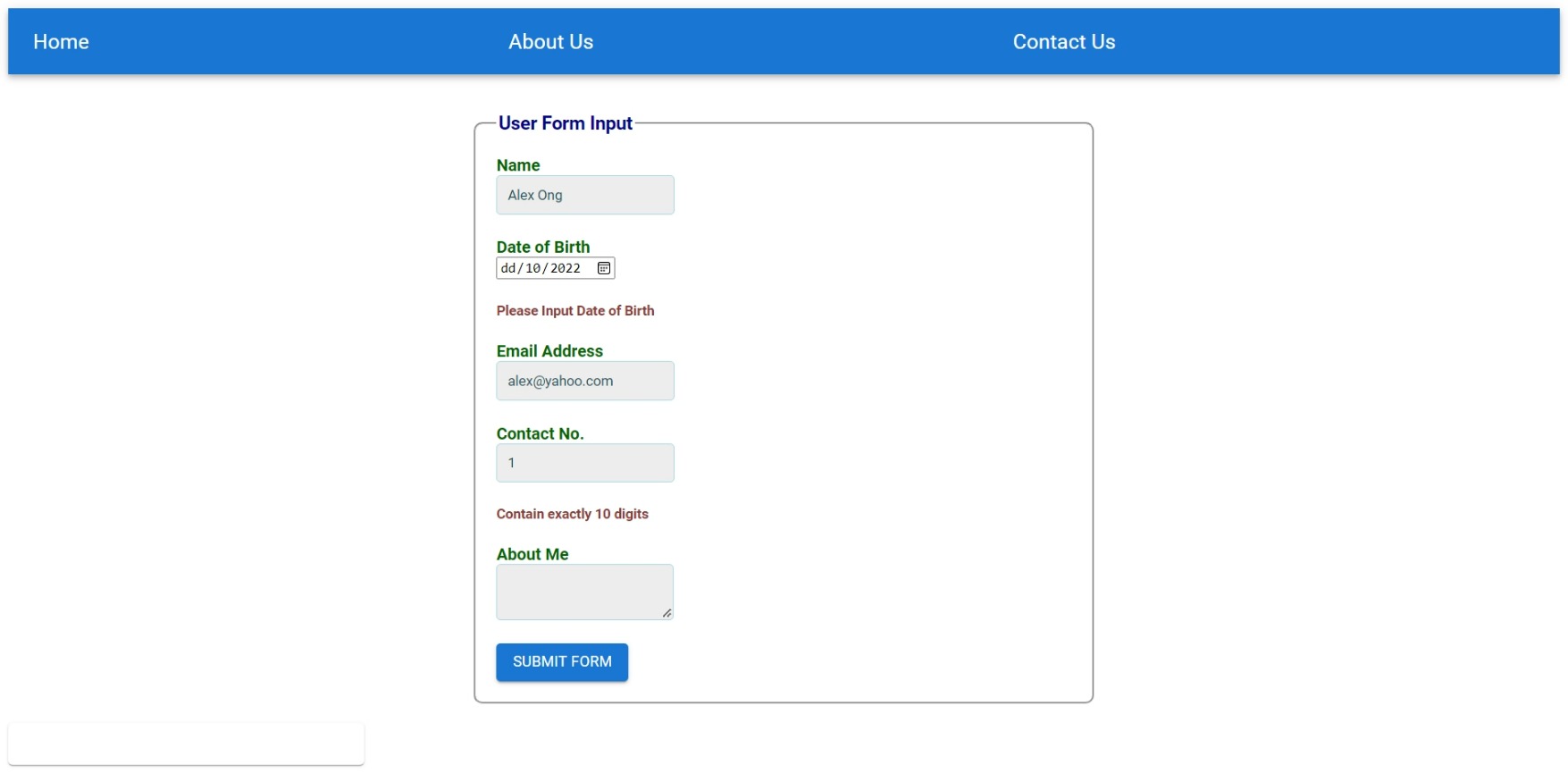
**Email Validation**

1. Validation for empty field 2) Validation for invalid email format



**Contact Validation**

1. Validation for 10 digits input 2) Validation for empty field



Source Code for App.js

import React, { Component } from 'react';

import FormInput from './FormInput';

import Display from '../Display';

import NavBar from '../NavBar';

class App extends Component{

state = {userData:[]};

AddData =(user) =>{

  console.log(user);

    this.setState({

      userData: [...this.state.userData,user]

    })

}

  render() {

    return (

       <div>

        <NavBar/> <br/><br/>

       <FormInput data={this.AddData} /> <br />

       <Display details={this.state.userData} />

       </div>

    );

  }

};

export default App;

Source Code for Display.js

import React from 'react';

import Card from '@mui/material/Card';

import CardContent from '@mui/material/CardContent';

import Typography from '@mui/material/Typography';

import { CardActionArea } from '@mui/material';

import "./src/App.css";

const Display = (props) => {

    const {details} = props;

    const dataList = details.map( (ele) => {

        return(

            <div className='card' key={ele.contact}>

            <h4> Name: {ele.name} </h4>

            <h4> Date of Birth: {ele.dob} </h4>

            <h4> Email Address: {ele.email} </h4>

            <h4> Contact No.: {ele.contact} </h4>

            <h4> About Me: {ele.description} </h4>

            </div>

        )

    });

    return (

        <Card sx={{ maxWidth: 345 }}>

          <CardActionArea>

              <CardContent>

              <Typography gutterBottom variant="h5" component="div">

               <h4> Business Card </h4>

               <p> {dataList} </p>

              </Typography>

          </CardContent>

          </CardActionArea>

        </Card>

      );

    };

      export default Display;

Source Code for NavBar.js

import React from 'react';

import AppBar from '@mui/material/AppBar';

import Box from '@mui/material/Box';

import Toolbar from '@mui/material/Toolbar';

import Typography from '@mui/material/Typography';

const NavBar = () =>{

return (

    <Box sx={{ flexGrow: 1 }}>

      <AppBar position="static">

        <Toolbar>

          {/\* <IconButton

            size="large"

            edge="start"

            color="inherit"

            aria-label="menu"

            sx={{ mr: 2 }}

          >

            <MenuIcon />

          </IconButton> \*/}

          <Typography variant="h6" component="div" sx={{ flexGrow: 1 }}>

            Home

          </Typography>

          <Typography variant="h6" component="div" sx={{ flexGrow: 1 }}>

            About Us

          </Typography>

          <Typography variant="h6" component="div" sx={{ flexGrow: 1 }}>

            Contact Us

          </Typography>

        </Toolbar>

      </AppBar>

    </Box>

  );

};

export default NavBar;

Source Code for FormInput.js

import React, { Component } from 'react';

import Button from '@mui/material/Button';

import "./App.css";

class FormInput extends Component{

    state = {

         name:'',

         dob:'',

         email:'',

         contact:'',

         description:'',

        errormsg:{

            nameError:'',

            dobError:'',

            emailError:'',

            contactError:'',

           },

        formCheck:false,

    };

    // To handle the change in the User Input

    handleChange = (e) => {

        if(e.target.id == 'name'){

            this.checkName(e.target.value);

        } else if(e.target.id == 'dob'){

                this.checkDob(e.target.value);

                } else if(e.target.id == 'email'){

                    this.checkEmail(e.target.value);

                    }else if(e.target.id == 'contact'){

                        this.checkContact(e.target.value);

                    }else if(e.target.id == 'description'){

                        this.setState({description:e.target.value})

                    }

                };

    // Validation of Name Field

    checkName = (nameUpdate) => {

        let nameError = this.state.errormsg.nameError;

        let formCheck = this.state.formCheck;

        let regex = /^[a-zA-Z\s]\*$/ ;

        if(nameUpdate.trim() === ''){

            nameError = "Please Enter Name";

            formCheck = false;

        }else if (!regex.test(nameUpdate)){

            nameError = "Characters A-Z or a-z only";

            formCheck = false;

        } else {

            nameError ='';

            formCheck = true;

        }

        this.setState({

            name: nameUpdate,

            errormsg:{...this.state.errormsg,nameError}

        });

        return formCheck;

    }

    // Validation of DOB Field

    checkDob = (dobUpdate) => {

        let dobError = this.state.errormsg.dobError;

        let formCheck = this.state.formCheck;

        let date = new Date() ;

        let checkdate = date.getFullYear() +"-" + (Number(date.getMonth())+Number(1)) + "-" + date.getDate();

        if(dobUpdate === ''){

            dobError = "Please Input Date of Birth";

            formCheck = false;

        }else if (dobUpdate >= checkdate){

            dobError = "Date of Birth must be smaller than current date ";

            formCheck = false;

        } else {

            dobError ='';

            formCheck = true;

        }

        this.setState({

            dob: dobUpdate,

            errormsg:{...this.state.errormsg,dobError}

        });

        return formCheck;

    }

    // Validation of Email Field

    checkEmail = (emailUpdate) => {

        let emailError = this.state.errormsg.emailError;

        let formCheck = this.state.formCheck;

        let regex = /^(([^<>()[\]\\.,;:\s@"]+(\.[^<>()[\]\\.,;:\s@"]+)\*)|(".+"))@((\[[0-9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}\])|(([a-zA-Z\-0-9]+\.)+[a-zA-Z]{2,}))$/ ;

        if(emailUpdate.trim() === ''){

            emailError = "Please Input Email";

            formCheck = false;

        }else if (!regex.test(emailUpdate)){

            emailError = "Invalid Email Address";

            formCheck = false;

        } else {

            emailError ='';

            formCheck = true;

        }

        this.setState({

            email: emailUpdate,

            errormsg:{...this.state.errormsg,emailError}

        });

        return formCheck;

    }

    // Validation of Contact Field

    checkContact = (contactUpdate) => {

        let contactError = this.state.errormsg.contactError;

        let formCheck = this.state.formCheck;

        let regex = /^[0-9]{10}$/ ;

        if(contactUpdate.trim() === ''){

            contactError = "Please Enter Contact Number";

            formCheck = false;

        }else if (!regex.test(contactUpdate)){

            contactError = "Contain exactly 10 digits";

            formCheck = false;

        } else {

            contactError ='';

            formCheck = true;

        }

        this.setState({

            contact: contactUpdate,

            errormsg:{...this.state.errormsg,contactError}

        });

        return formCheck;

    }

    //To handle the Submit Button Event

    handleSubmit = (e) => {

        e.preventDefault();

        let nameCheck = this.checkName(this.state.name);

        let dobCheck = this.checkDob(this.state.dob);

        let emailCheck = this.checkEmail(this.state.email);

        let contactCheck = this.checkContact(this.state.contact);

        if (nameCheck && dobCheck && emailCheck && contactCheck){

            alert("form submitted");

            this.props.data(this.state);

            this.setState({

                name:'',

                dob:'',

                email:'',

                contact:'',

                description:'',

            });

        }else{

            alert("Please Check Your Details")

        }

    }

    render() {

        return (

            <div>

                <form className='formInput' onSubmit={this.handleSubmit}>

                <fieldset>

                <legend> User Form Input </legend>

                <label htmlFor='name'> Name </label>

                <input type="text" id="name" name="name" placeholder='Name' value={this.state.name} onChange={this.handleChange}/>

                <h5>{this.state.errormsg.nameError}</h5>

                <label htmlFor='dob'> Date of Birth </label>

                <input type="date" id="dob" name="dob" placeholder='Date of Birth' value={this.state.dob} onChange={this.handleChange}/>

                <h5>{this.state.errormsg.dobError}</h5>

                <label htmlFor='email'> Email Address </label>

                <input type="email" id="email" name="email" placeholder='Email' value={this.state.email} onChange={this.handleChange}/>

                <h5>{this.state.errormsg.emailError}</h5>

                <label htmlFor='contact'> Contact No. </label>

                <input type="number" id="contact" name="contact" placeholder='Contact No.' value={this.state.contact} onChange={this.handleChange}/>

                <h5>{this.state.errormsg.contactError}</h5>

               <label htmlFor='description'> About Me </label>

               <textarea id="description" name="description" value={this.state.description} onChange={this.handleChange}/>

               <br/><br/>

                <Button variant="contained" type="submit">Submit Form </Button>

                </fieldset>

                </form>

                </div>

        );

    }

};

export default FormInput;