Name : Sanjay Khanna S

Emp ID : 12124

Intermediate Assessment – React Js Lab

Question: Laptop Allocation Management System

Required Properties are : Laptop Id, Laptop Make, Configuration, Allocated User, Allotted On, Allocation Period.

1. Define a web application for the given scenario using React as a front end.
2. Configure JSON server and consider JSON file as your database.
3. Perform all the CRUD operations using React Http Client Service.
4. Perform the validations for all the fields.
5. Application must be the Responsive Page (Including Mobile View) and Single Page Application (SPA).
6. UI must be a Rich UI and look & Feel.
7. Maintain the Code Quality .

Answer:

Laptop Allocation Management

My Web Application Consist of

new-app

-node_modules

-Public

-Src

-App.css

-App.js

-Home.js

-Index.css

-Index.js

-LaptopDetail.js

-LaptopCreate.js

-LaptopEdit.js

-LaptopListing.js

-LaptopLogin.js

-LaptopRegister.js

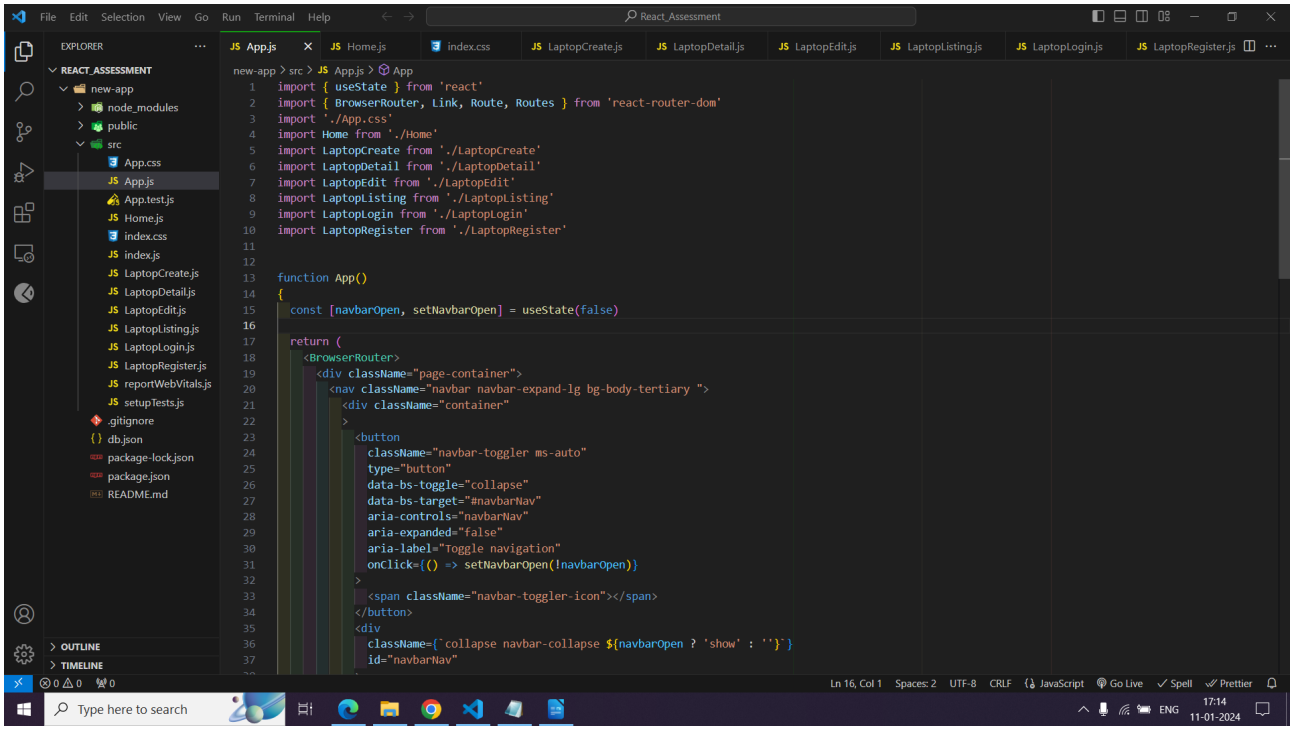
-db.json

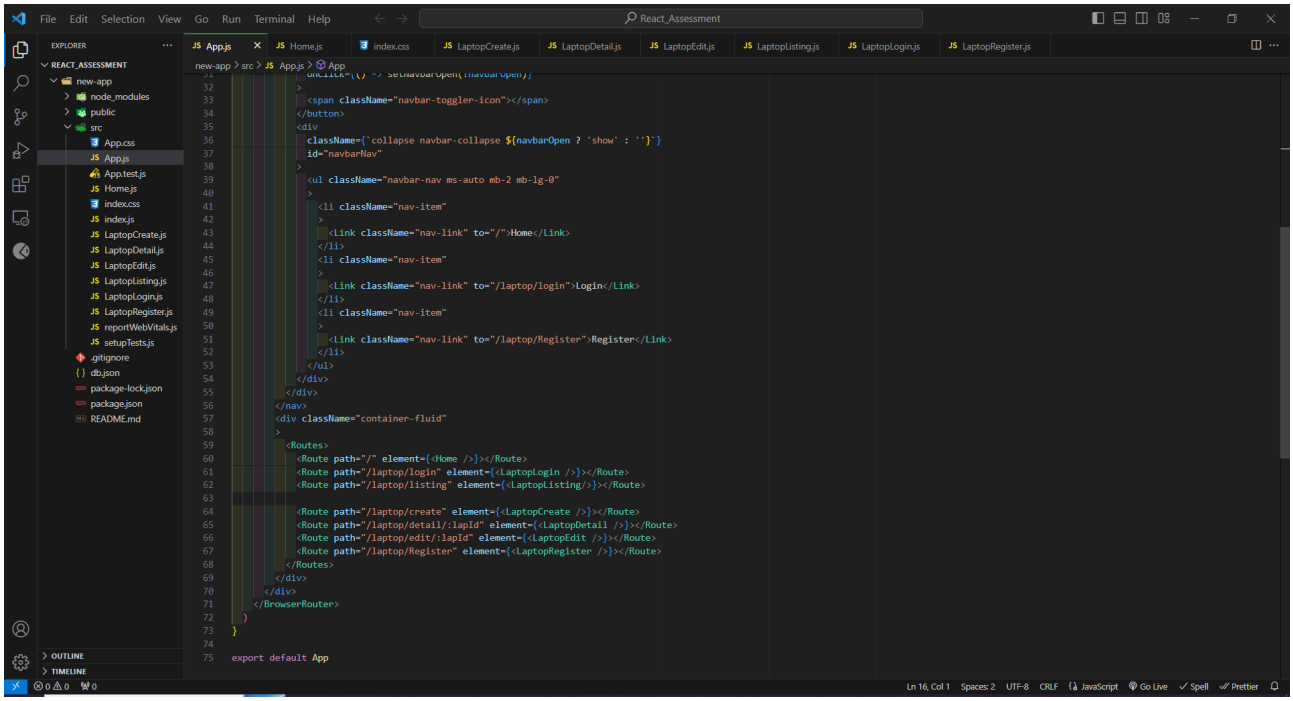
-Package.lock.json

-Package.json

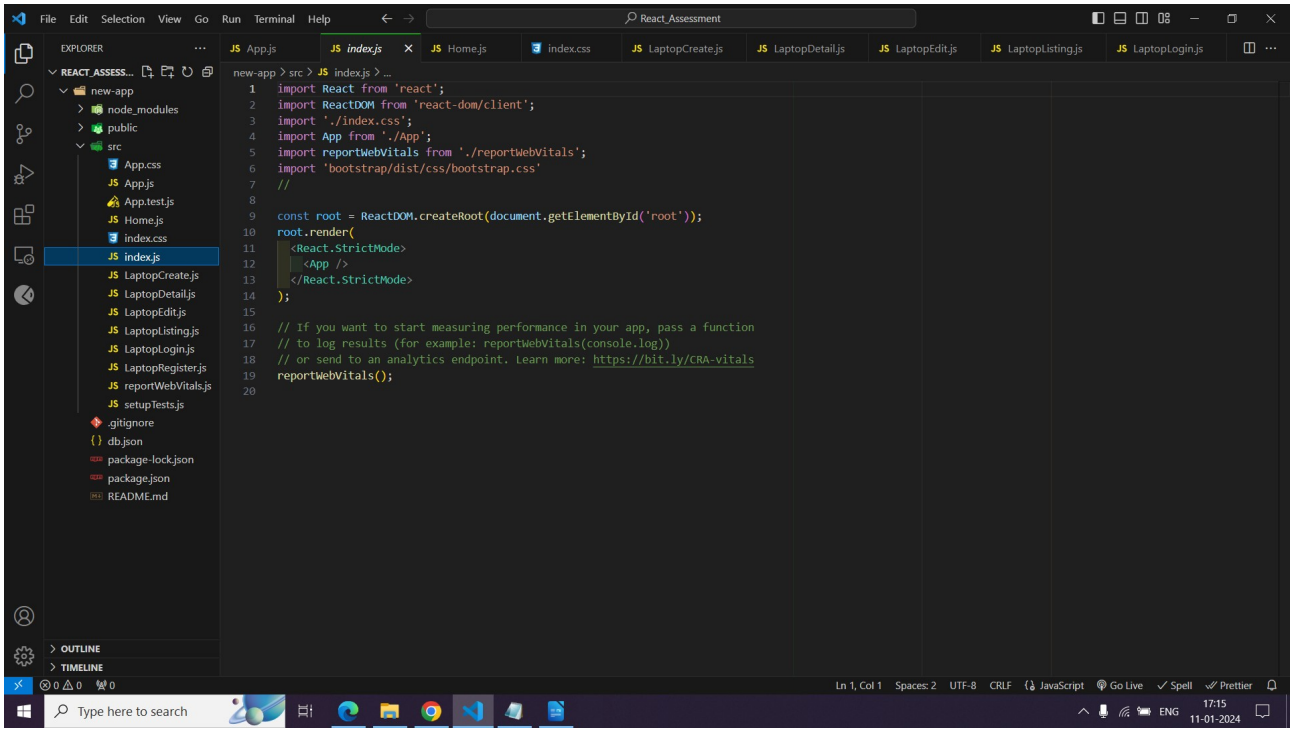
Coding

App.js

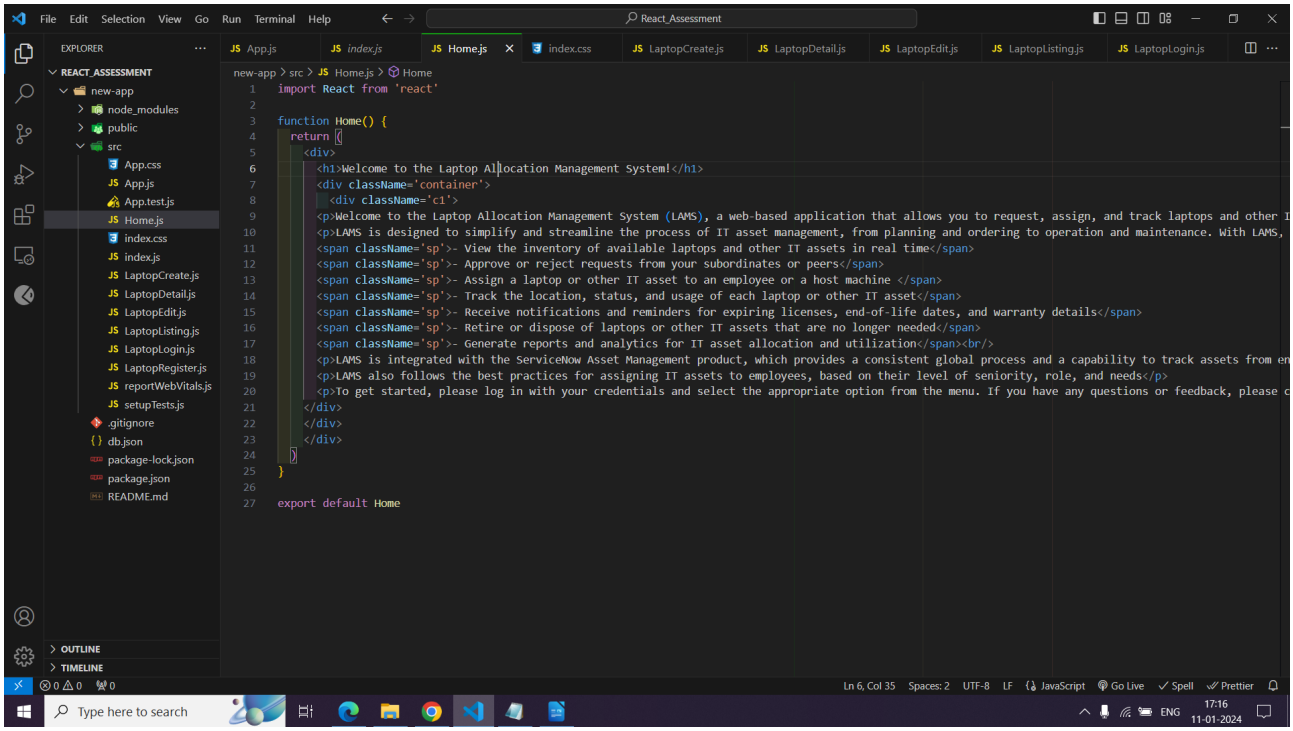




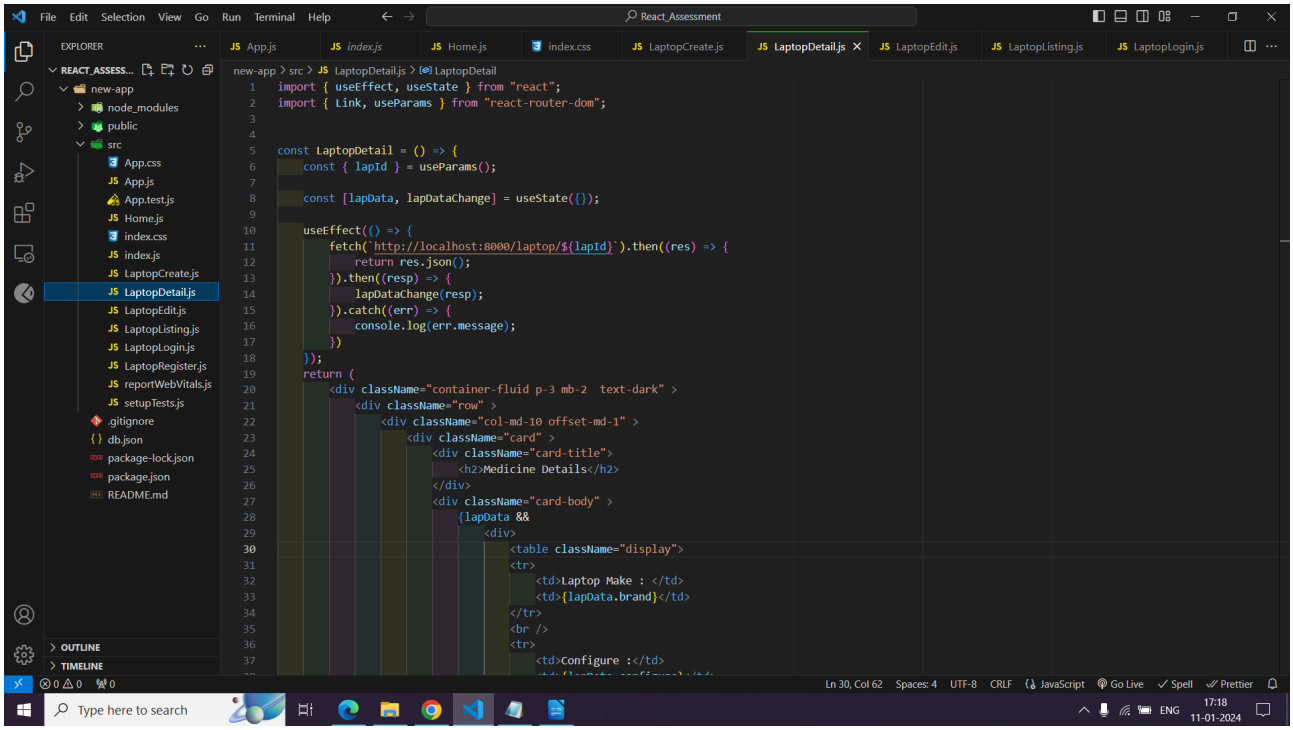
Index.js



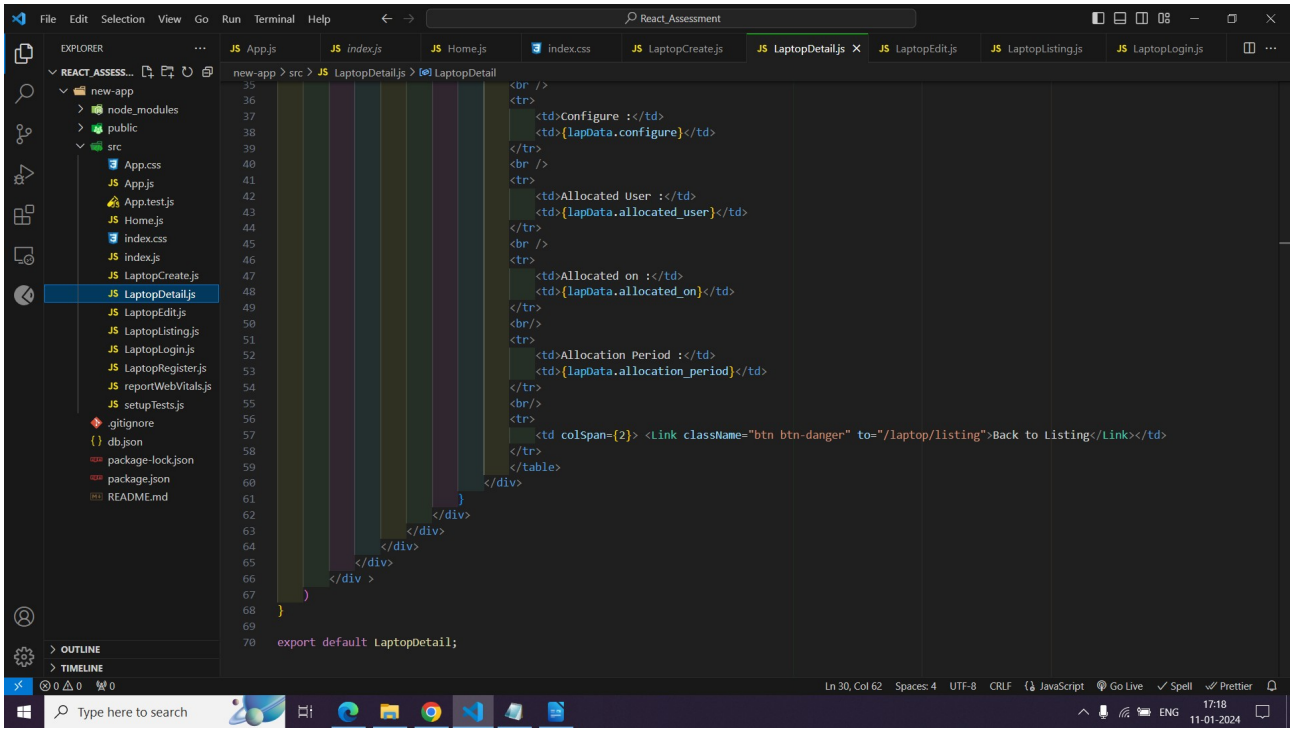
Home.js



LaptopDetail.js

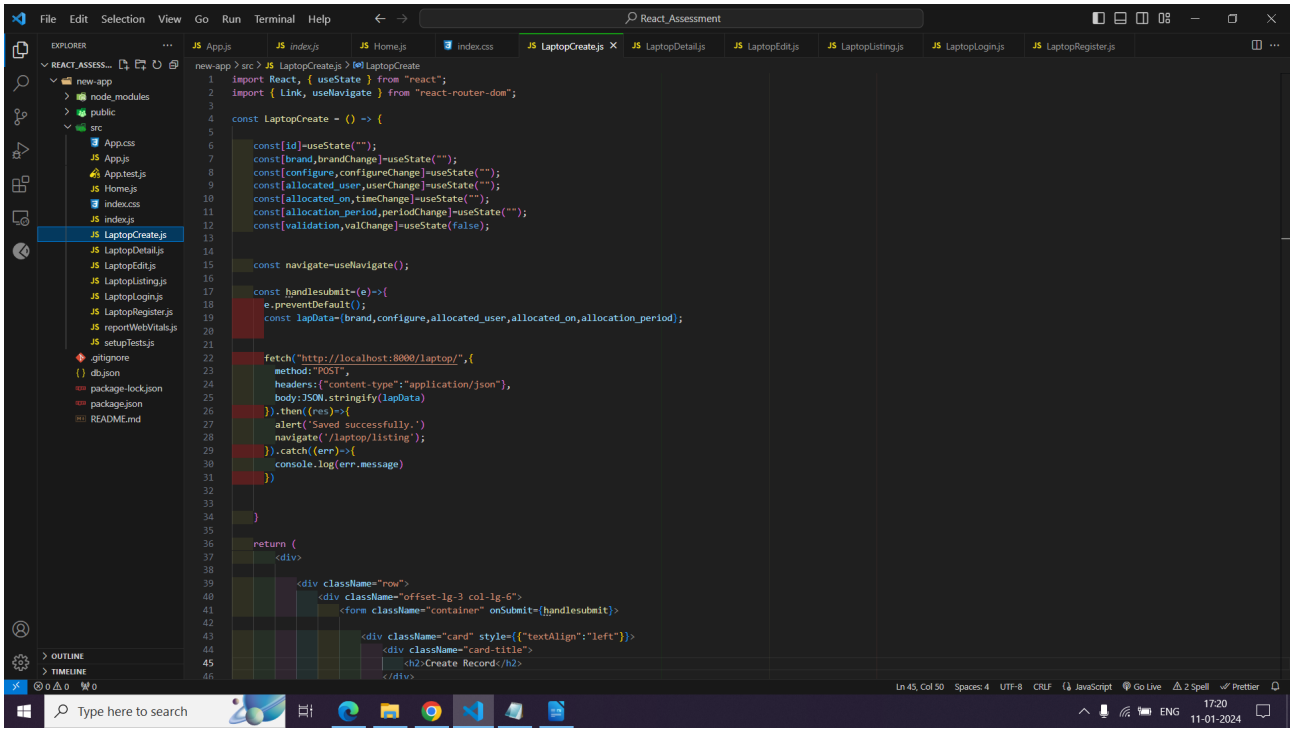


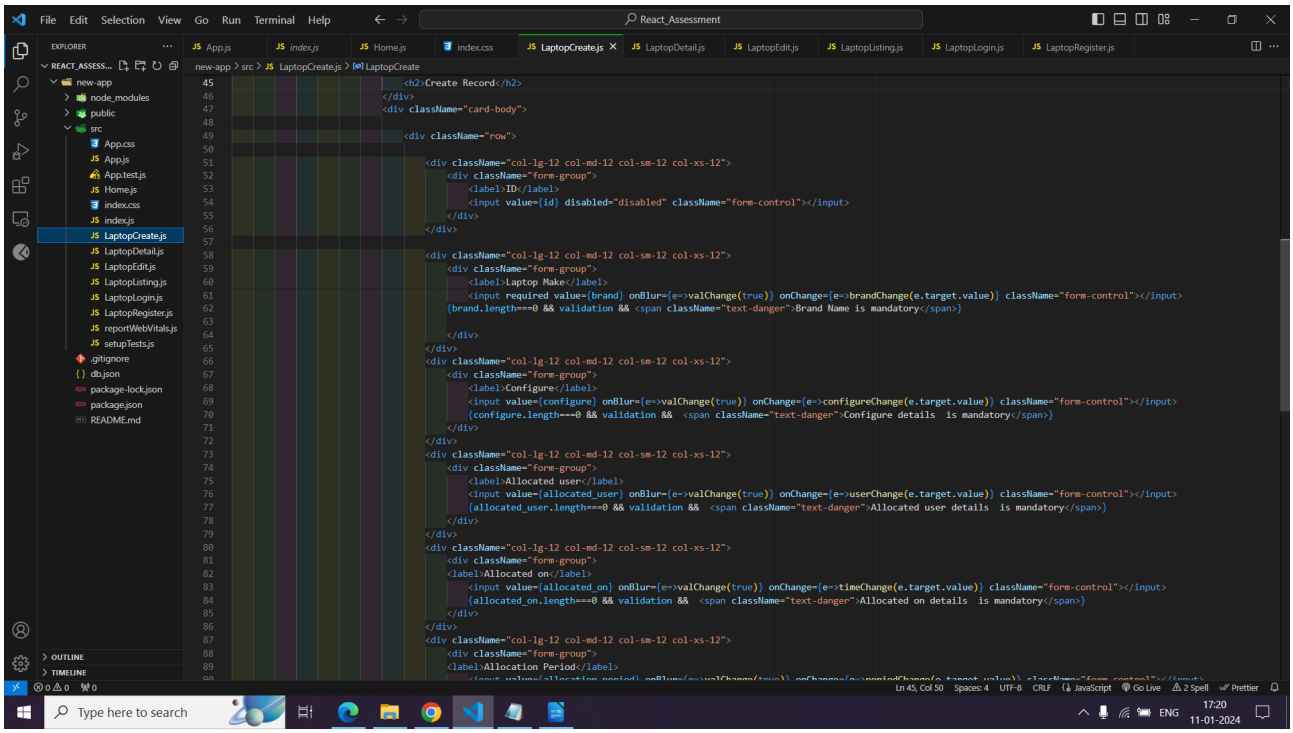
```
import { useEffect, useState } from "react";
import { Link, useParams } from "react-router-dom";


const LaptopDetail = () => {
    const { lapId } = useParams();

    const [lapData, lapDataChange] = useState({});

    useEffect(() => {
        fetch(`http://localhost:8000/laptop/${lapId}`).then((res) => {
            return res.json();
        }).then((resp) => {
            lapDataChange(resp);
        }).catch((err) => {
            console.log(err.message);
        })
    });
    return (
        <div className="container-fluid p-3 mb-2  text-dark" >
            <div className="row" >
                <div className="col-md-10 offset-md-1" >
                    <div className="card" >
                        <div className="card-title">
                            <h2>Medicine Details</h2>
                        </div>
                        <div className="card-body" >
                            {lapData &&
                                <div>
                                    <table className="display">
                                    <tr>
                                        <td>Laptop Make : </td>
                                        <td>{lapData.brand}</td>
                                    </tr>
                                    <br />
                                    <tr>
                                        <td>Configure :</td>
```

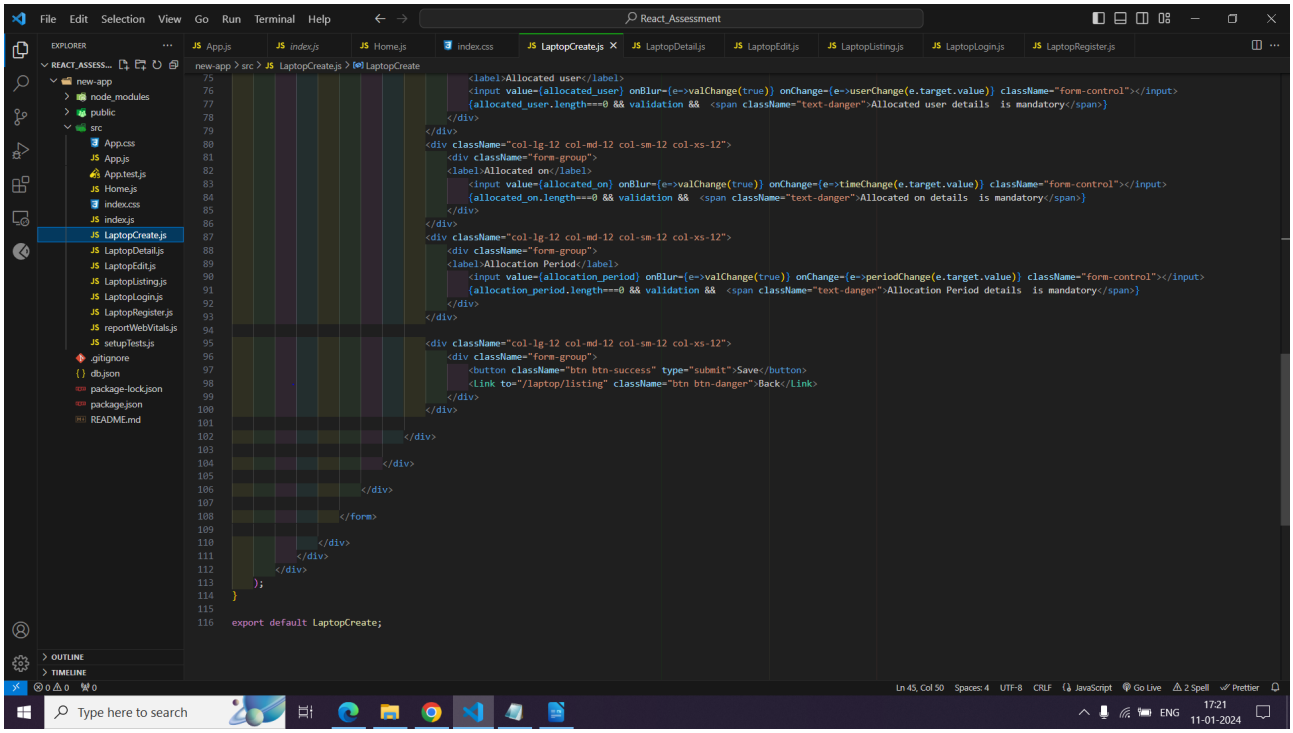


```
                                    <br />
                                    <tr>
                                        <td>Configure :</td>
                                        <td>{lapData.configure}</td>
                                    </tr>
                                    <br />
                                    <tr>
                                        <td>Allocated User :</td>
                                        <td>{lapData.allocated_user}</td>
                                    </tr>
                                    <br />
                                    <tr>
                                        <td>Allocated on :</td>
                                        <td>{lapData.allocated_on}</td>
                                    </tr>
                                    <br/>
                                    <tr>
                                        <td>Allocation Period :</td>
                                        <td>{lapData.allocation_period}</td>
                                    </tr>
                                    <br/>
                                    <tr>
                                        <td colSpan={2}> <Link className="btn btn-danger" to="/laptop/listing">Back to Listing</Link></td>
                                    </tr>
                                    </table>
                                </div>
                            }
                        </div>
                    </div>
                </div>
            </div>
        </div >
    )
}

export default LaptopDetail;
```

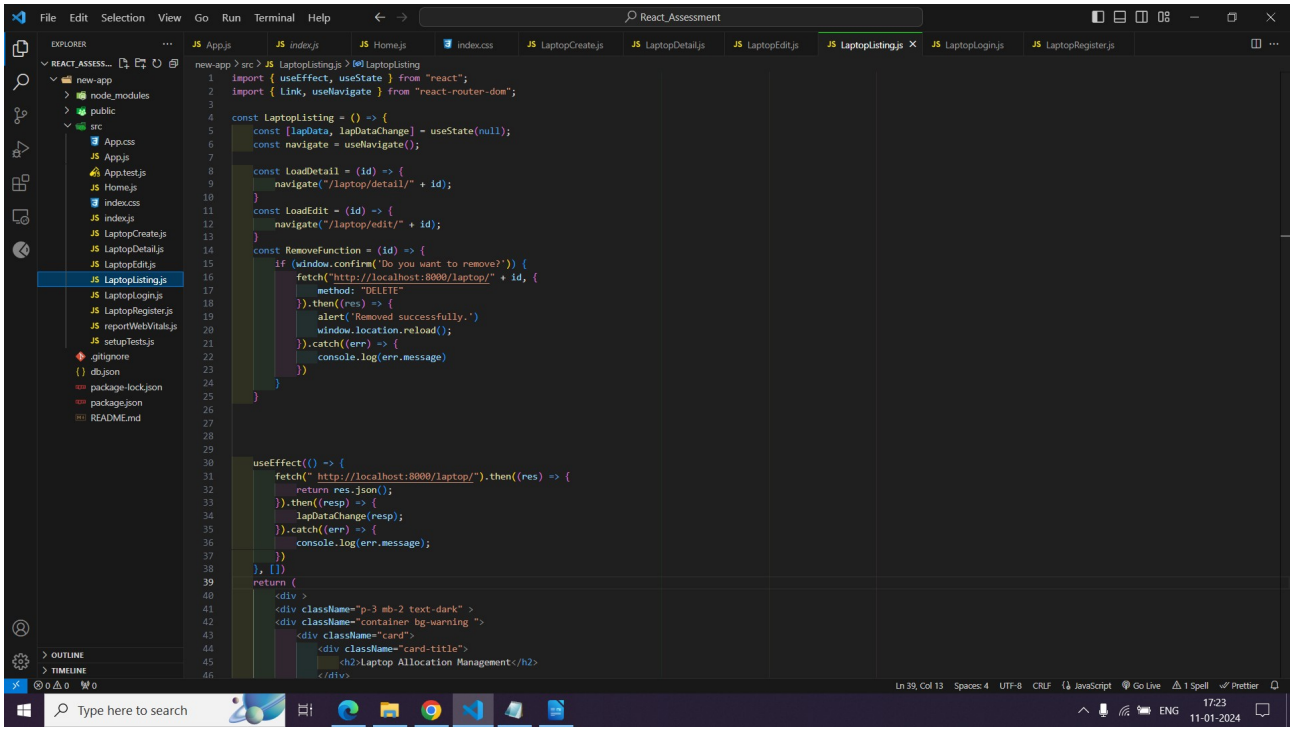
LaptopCreate.js

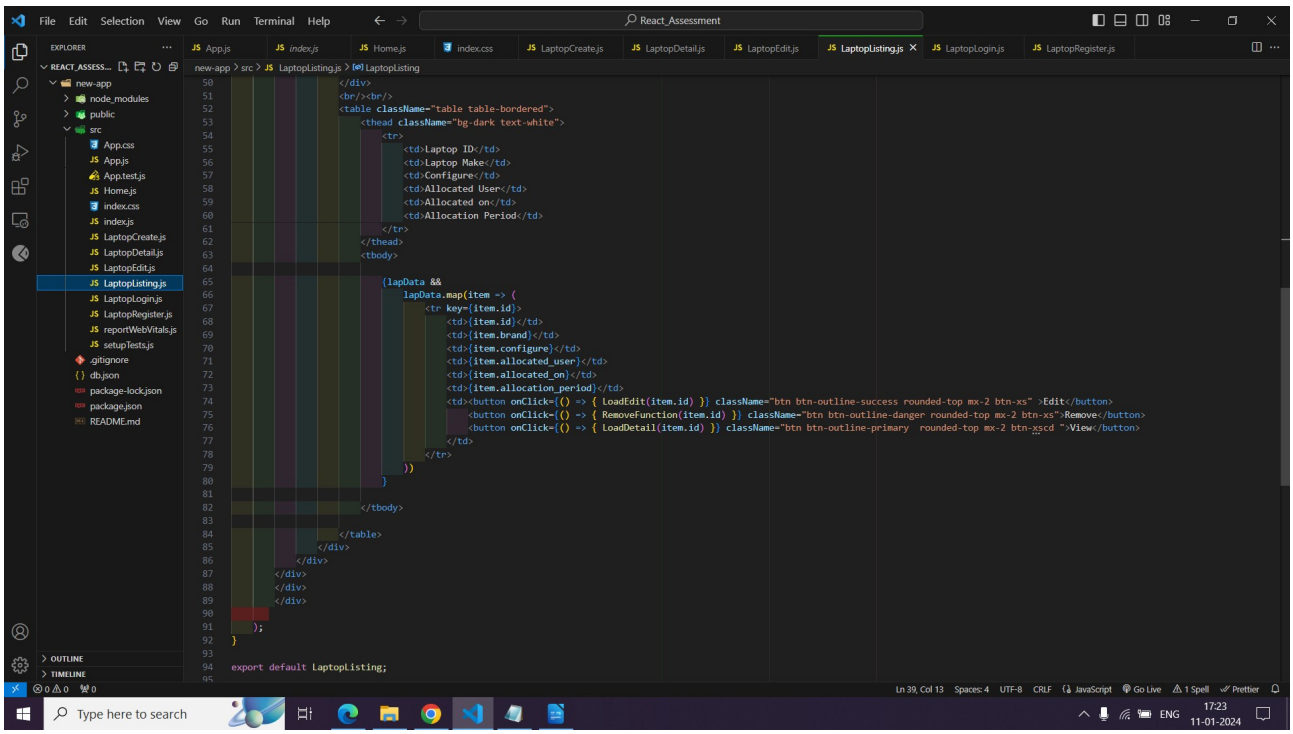




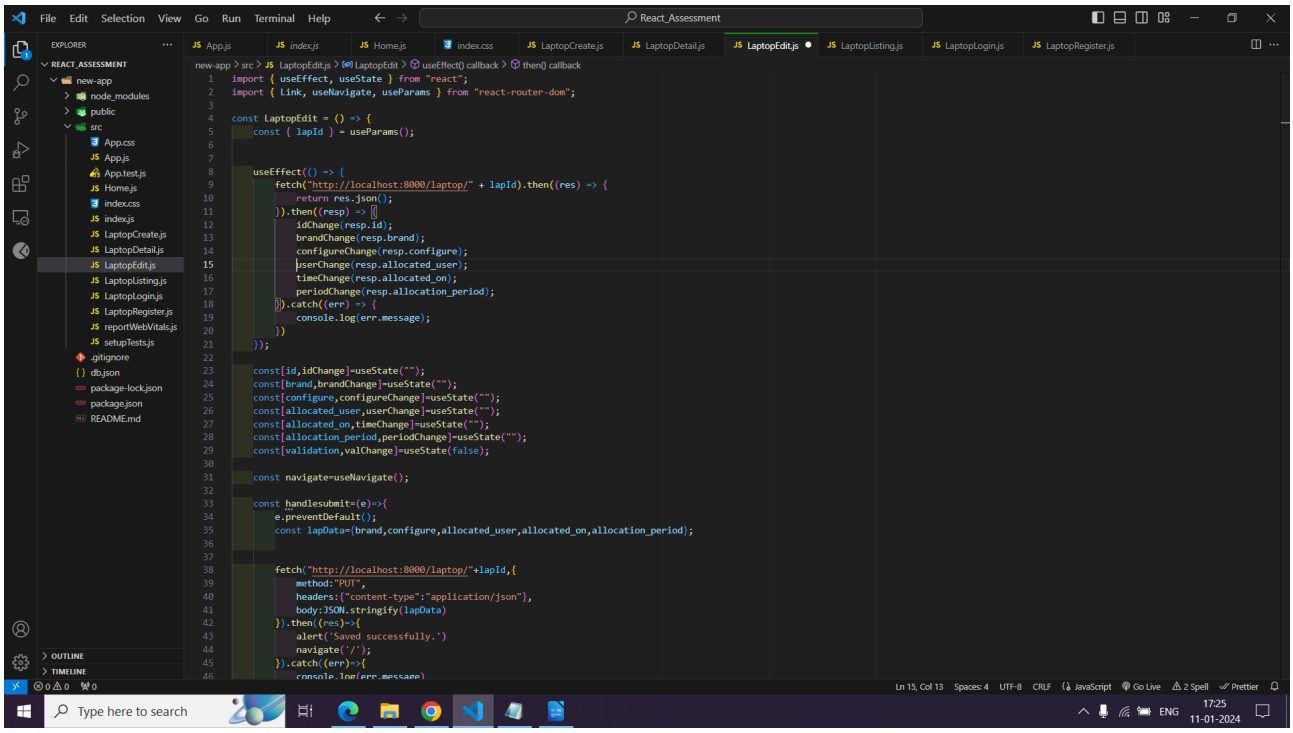


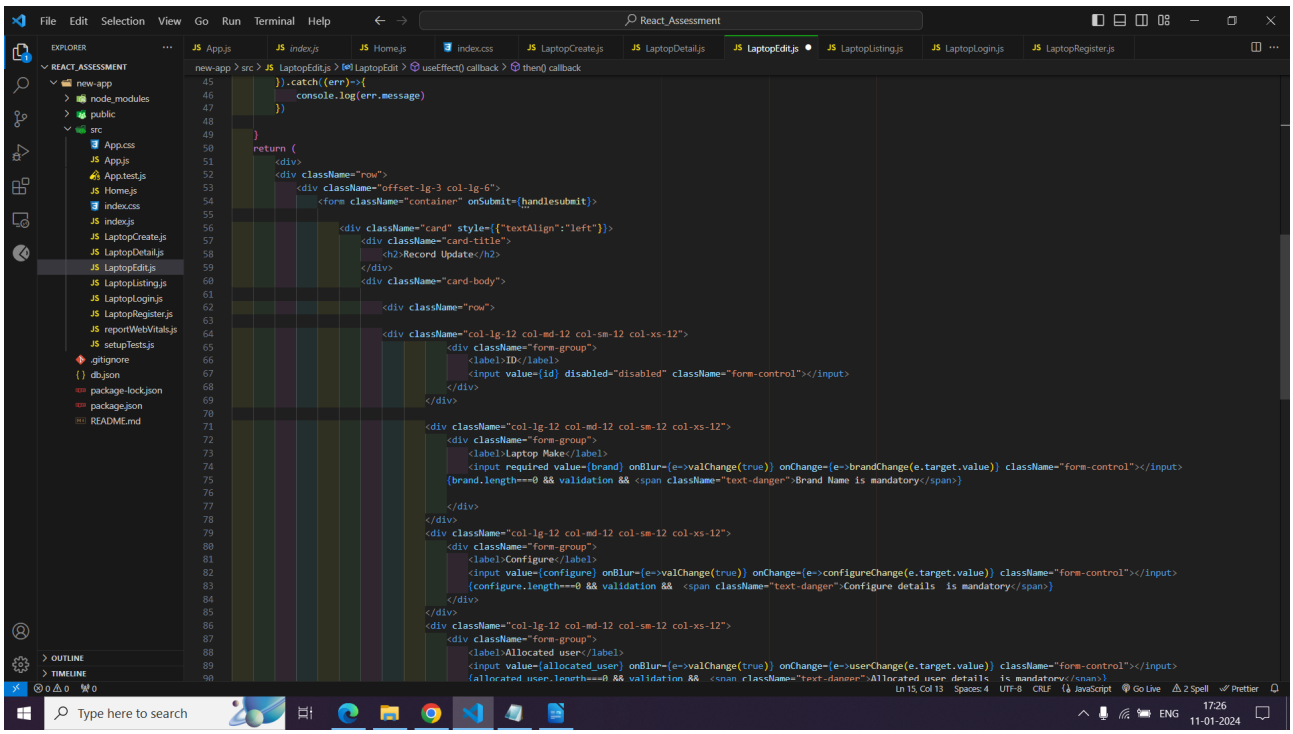
LaptopListing.js



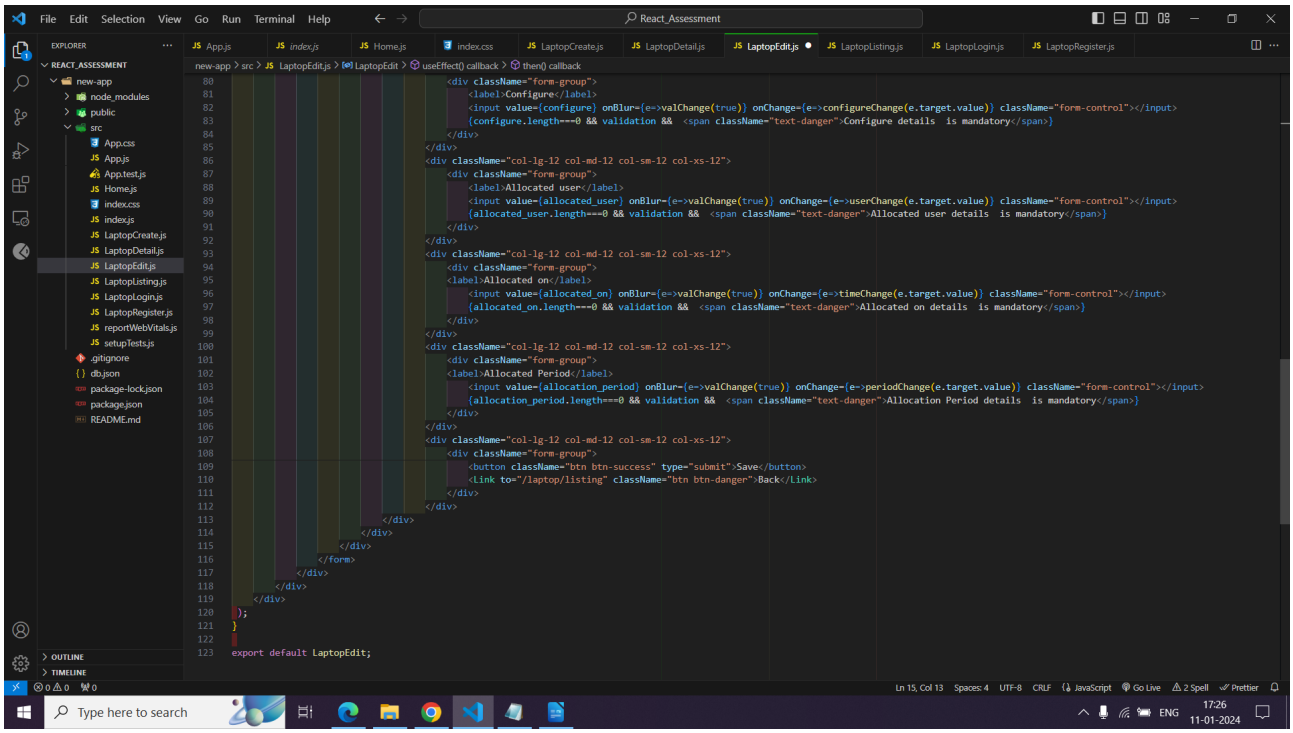


LaptopEdit.js

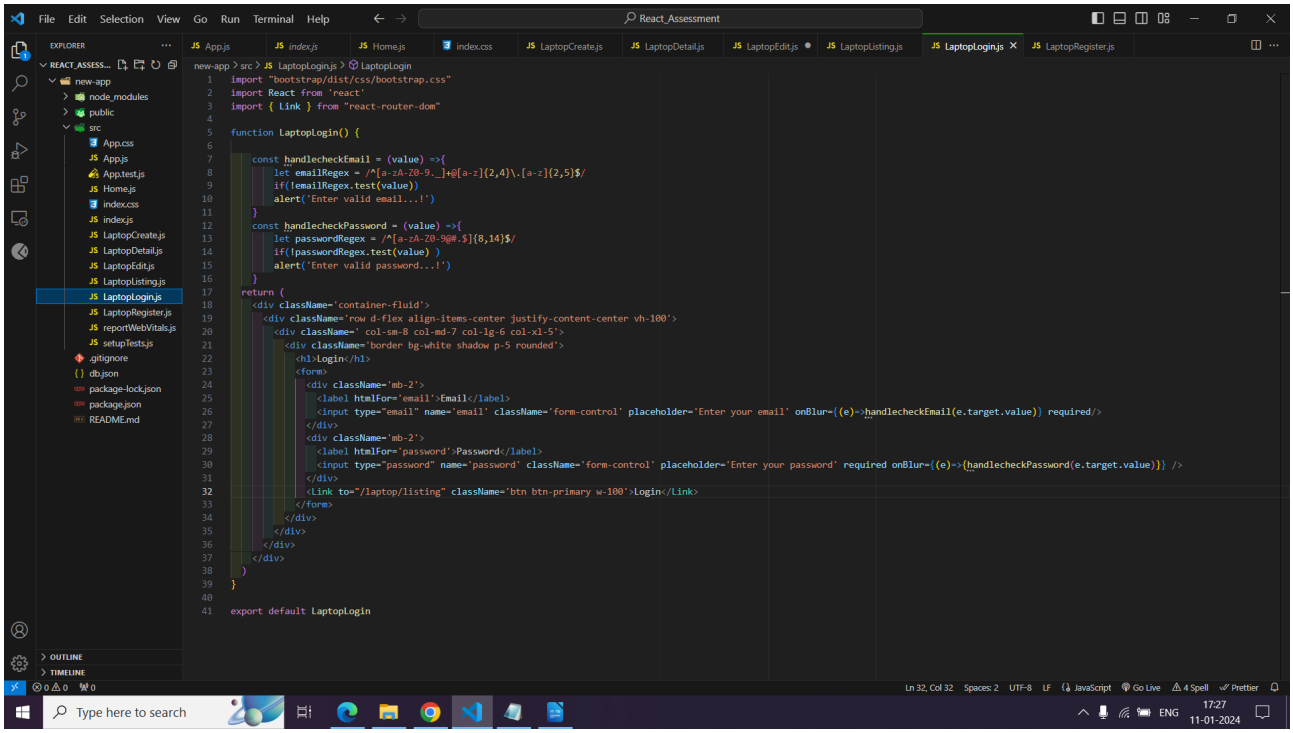




```
        }).catch((err)=>{
            console.log(err.message)
        })

    }
    return (
        <div>
        <div className="row">
            <div className="offset-lg-3 col-lg-6">
                <form className="container" onSubmit={handlesubmit}>

                    <div className="card" style={{"textAlign":"left"}}>
                        <div className="card-title">
                            <h2>Record Update</h2>
                        </div>
                        <div className="card-body">

                            <div className="row">

                            <div className="col-lg-12 col-md-12 col-sm-12 col-xs-12">
                                    <div className="form-group">
                                        <label>ID</label>
                                        <input value={id} disabled="disabled" className="form-control"></input>
                                    </div>
                                </div>

                                <div className="col-lg-12 col-md-12 col-sm-12 col-xs-12">
                                    <div className="form-group">
                                        <label>Laptop Make</label>
                                        <input required value={brand} onBlur={e=>valChange(true)} onChange={e=>brandChange(e.target.value)} className="form-control"></input>
                                        {brand.length===0 && validation && <span className="text-danger">Brand Name is mandatory</span>}

                                    </div>
                                </div>
                                <div className="col-lg-12 col-md-12 col-sm-12 col-xs-12">
                                    <div className="form-group">
                                        <label>Configure</label>
                                        <input value={configure} onBlur={e=>valChange(true)} onChange={e=>configureChange(e.target.value)} className="form-control"></input>
                                        {configure.length===0 && validation &&  <span className="text-danger">Configure details  is mandatory</span>}
                                    </div>
                                </div>
                                <div className="col-lg-12 col-md-12 col-sm-12 col-xs-12">
                                    <div className="form-group">
                                        <label>Allocated user</label>
                                        <input value={allocated_user} onBlur={e=>valChange(true)} onChange={e=>userChange(e.target.value)} className="form-control"></input>
                                        {allocated_user.length===0 && validation &&  <span className="text-danger">Allocated user details  is mandatory</span>}
                                    </div>
                                </div>
                                <div className="col-lg-12 col-md-12 col-sm-12 col-xs-12">
                                    <div className="form-group">
                                    <label>Allocated on</label>
                                        <input value={allocated_on} onBlur={e=>valChange(true)} onChange={e=>timeChange(e.target.value)} className="form-control"></input>
                                        {allocated_on.length===0 && validation &&  <span className="text-danger">Allocated on details  is mandatory</span>}
                                    </div>
                                </div>
                                <div className="col-lg-12 col-md-12 col-sm-12 col-xs-12">
                                    <div className="form-group">
                                    <label>Allocated Period</label>
                                        <input value={allocation_period} onBlur={e=>valChange(true)} onChange={e=>periodChange(e.target.value)} className="form-control"></input>
                                        {allocation_period.length===0 && validation &&  <span className="text-danger">Allocation Period details  is mandatory</span>}
                                    </div>
                                </div>
                                <div className="col-lg-12 col-md-12 col-sm-12 col-xs-12">
                                    <div className="form-group">
                                        <button className="btn btn-success" type="submit">Save</button>
                                        <Link to="/laptop/listing" className="btn btn-danger">Back</Link>
                                    </div>
                                </div>
                            </div>
                        </div>
                    </div>
                </form>
            </div>
        </div>
    </div>
);
}

export default LaptopEdit;
```

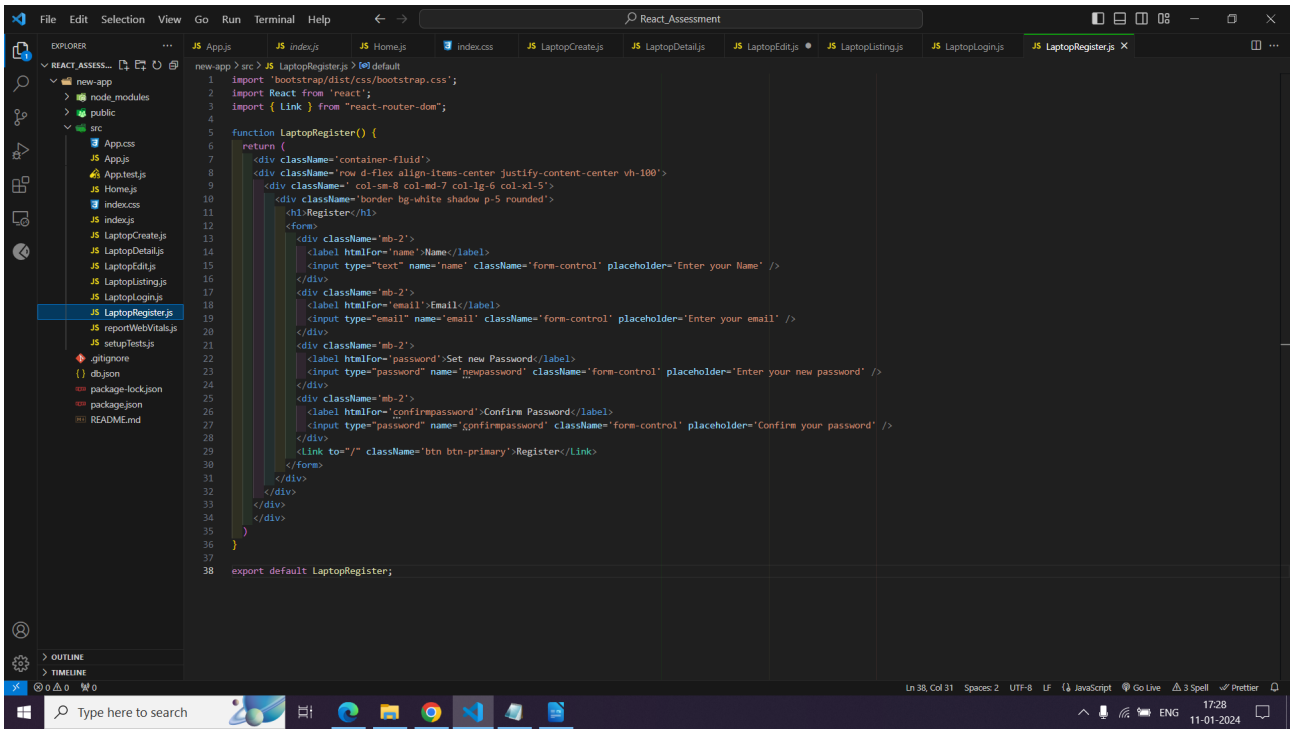


LaptopLogin.js



```
import "bootstrap/dist/css/bootstrap.css"
import React from 'react'
import { Link } from "react-router-dom"

function LaptopLogin() {

    const handlecheckEmail = (value) =>{
        let emailRegex = /^[a-zA-Z0-9._]+@[a-z]{2,4}\.[a-z]{2,5}$/
        if(!emailRegex.test(value))
        alert('Enter valid email...!')
    }
    const handlecheckPassword = (value) =>{
        let passwordRegex = /^[a-zA-Z0-9@#.$]{8,14}$/
        if(!passwordRegex.test(value) )
        alert('Enter valid password...!')
    }
  return (
    <div className='container-fluid'>
      <div className='row d-flex align-items-center justify-content-center vh-100'>
        <div className=' col-sm-8 col-md-7 col-lg-6 col-xl-5'>
          <div className='border bg-white shadow p-5 rounded'>
            <h1>Login</h1>
            <form>
              <div className='mb-2'>
                <label htmlFor='email'>Email</label>
                <input type="email" name='email' className='form-control' placeholder='Enter your email' onBlur={(e)=>handlecheckEmail(e.target.value)} required/>
              </div>
              <div className='mb-2'>
                <label htmlFor='password'>Password</label>
                <input type="password" name='password' className='form-control' placeholder='Enter your password' required onBlur={(e)=>{handlecheckPassword(e.target.value)}} />
              </div>
              <Link to="/laptop/listing" className='btn btn-primary w-100'>Login</Link>
            </form>
          </div>
        </div>
      </div>
    </div>
  )
}

export default LaptopLogin
```

LaptopRegister.js

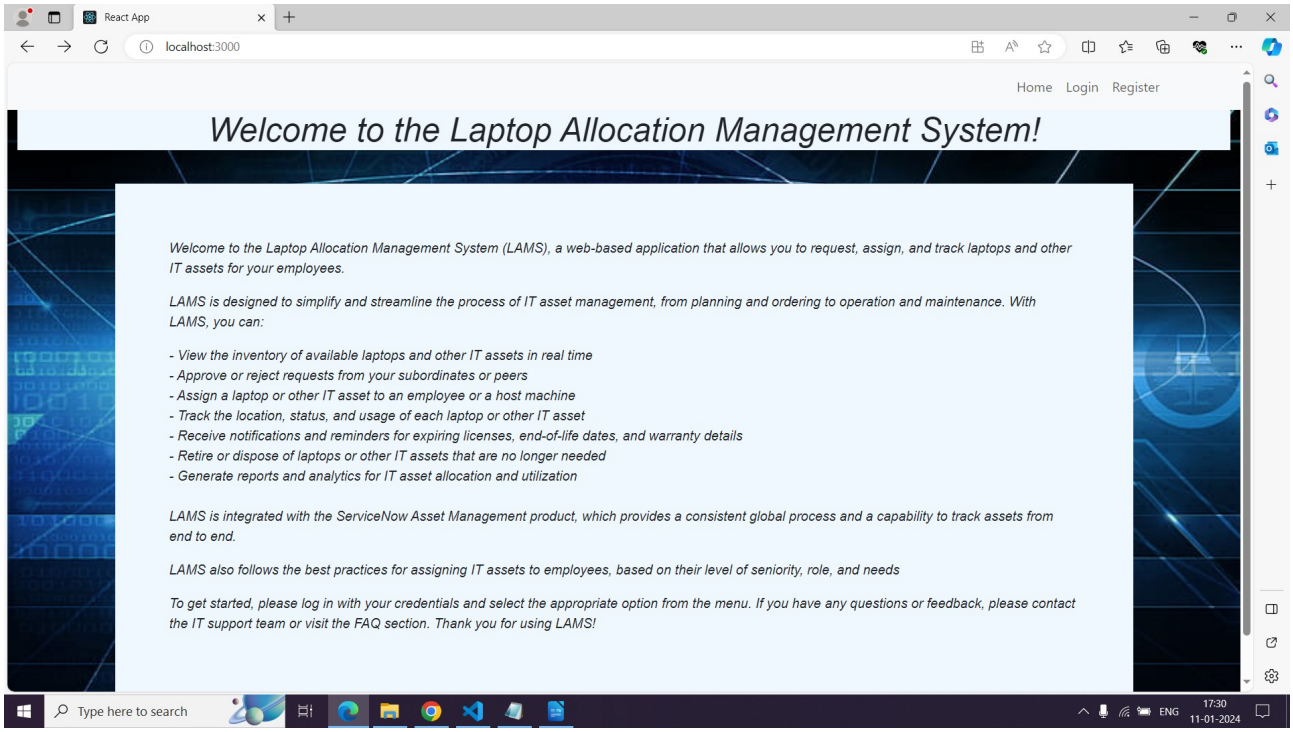


```
import 'bootstrap/dist/css/bootstrap.css';
import React from 'react';
import { Link } from "react-router-dom";

function LaptopRegister() {
  return (
    <div className='container-fluid'>
    <div className='row d-flex align-items-center justify-content-center vh-100'>
      <div className=' col-sm-8 col-md-7 col-lg-6 col-xl-5'>
        <div className='border bg-white shadow p-5 rounded'>
          <h1>Register</h1>
          <form>
            <div className='mb-2'>
              <label htmlFor='name'>Name</label>
              <input type="text" name='name' className='form-control' placeholder='Enter your Name' />
            </div>
            <div className='mb-2'>
              <label htmlFor='email'>Email</label>
              <input type="email" name='email' className='form-control' placeholder='Enter your email' />
            </div>
            <div className='mb-2'>
              <label htmlFor='password'>Set new Password</label>
              <input type="password" name='newpassword' className='form-control' placeholder='Enter your new password' />
            </div>
            <div className='mb-2'>
              <label htmlFor='confirmpassword'>Confirm Password</label>
              <input type="password" name='confirmpassword' className='form-control' placeholder='Confirm your password' />
            </div>
            <Link to="/" className='btn btn-primary'>Register</Link>
          </form>
        </div>
      </div>
    </div>
    </div>
  )
}

export default LaptopRegister;
```

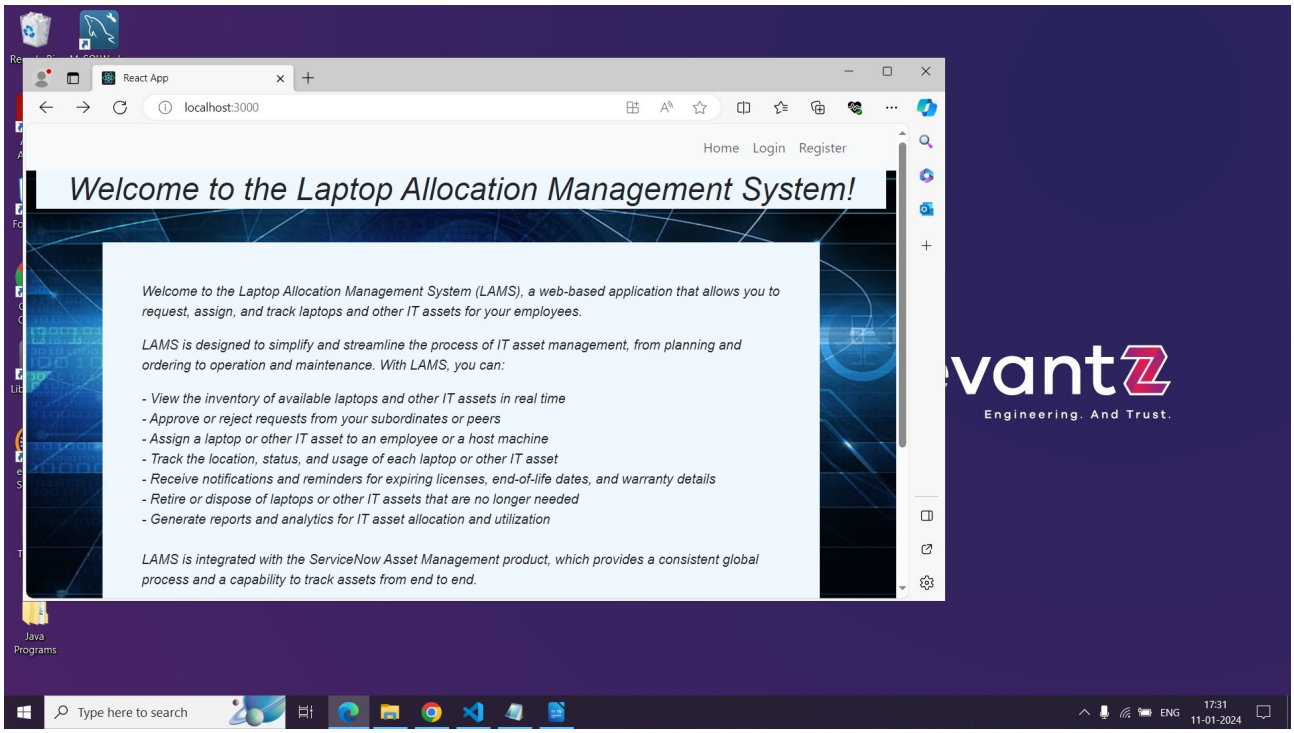

Output:

Home page

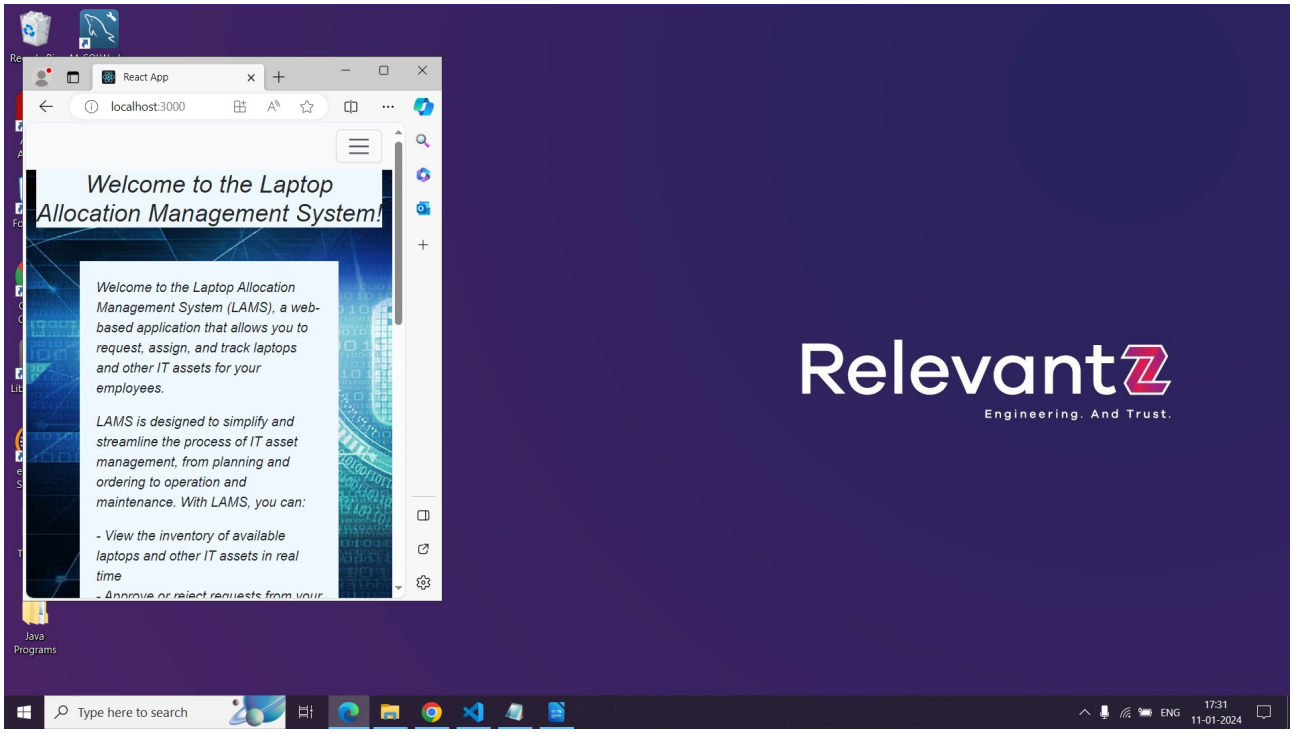
Larger:

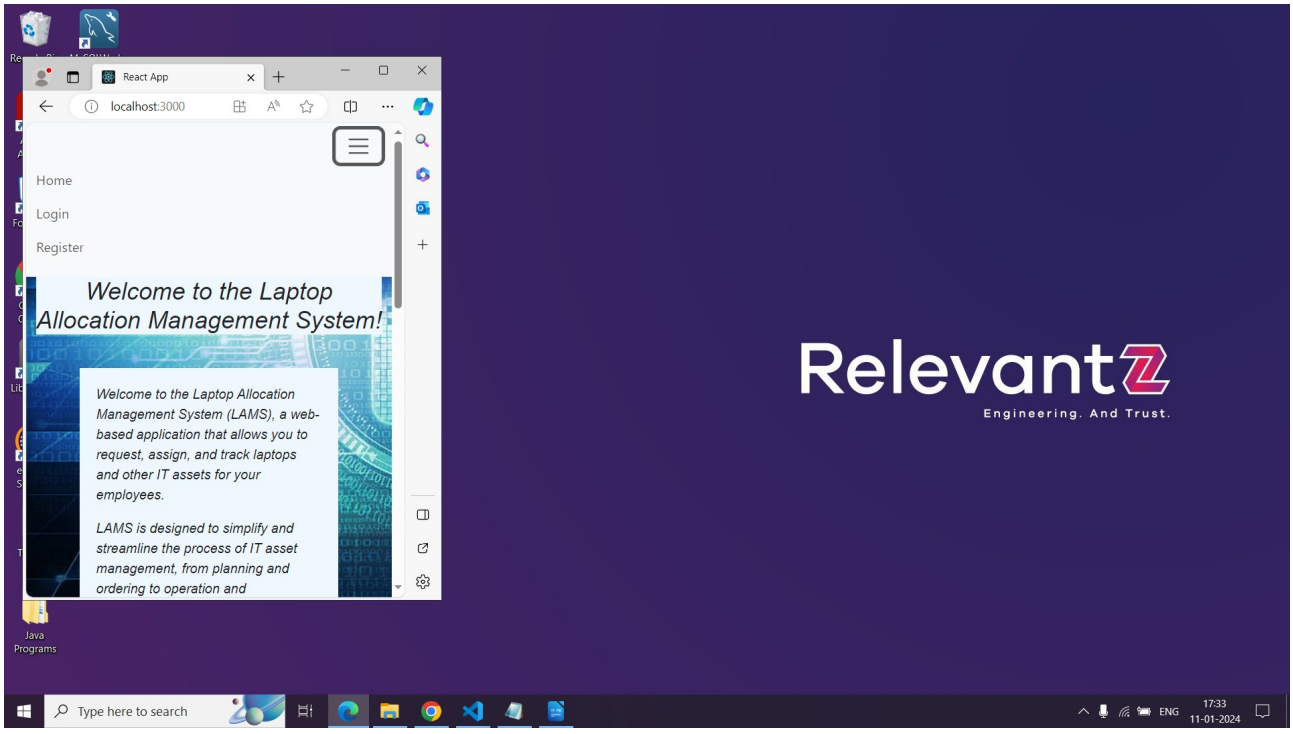


Medium:



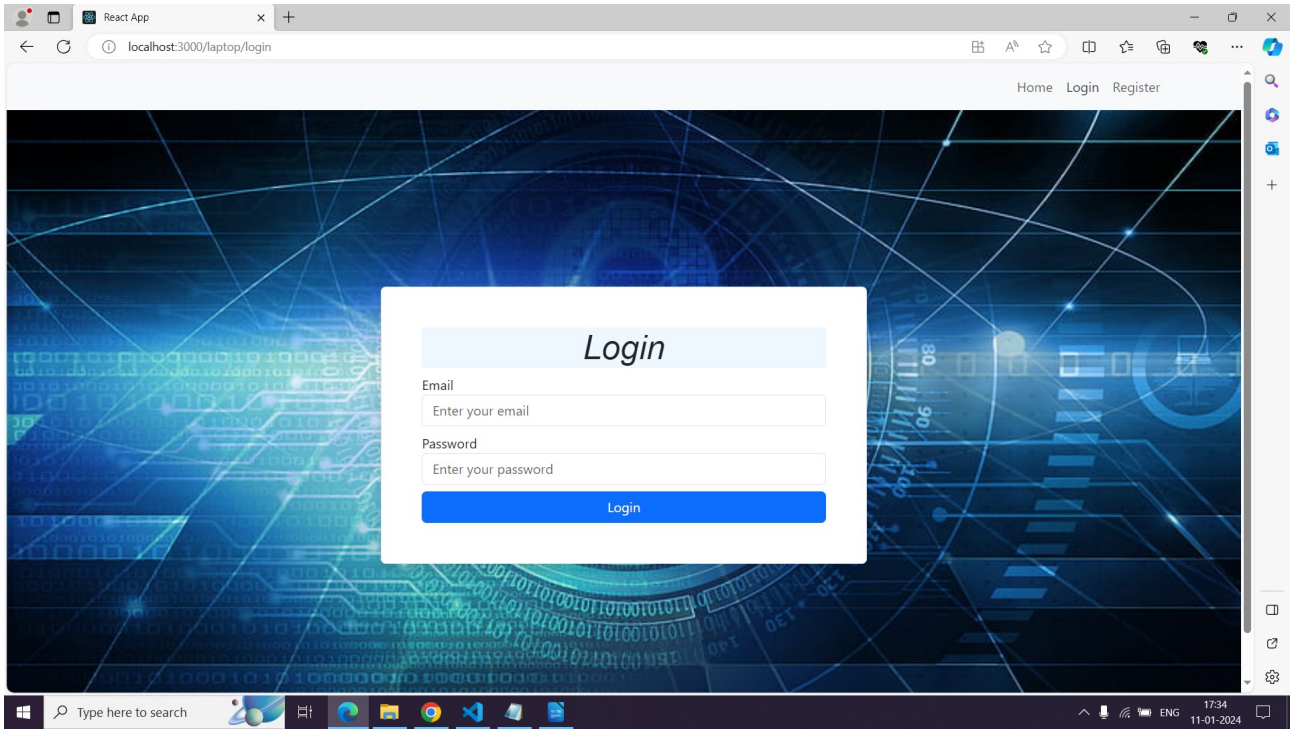
Small:


React App
localhost:3000
Welcome to the Laptop Allocation Management System!
Welcome to the Laptop Allocation Management System (LAMS), a web-based application that allows you to request, assign, and track laptops and other IT assets for your employees.
LAMS is designed to simplify and streamline the process of IT asset management, from planning and ordering to operation and maintenance. With LAMS, you can:
- View the inventory of available laptops and other IT assets in real time
Java Programs
Type here to search
ENG
17:31
11-01-2024
Relevantz
Engineering. And Trust.

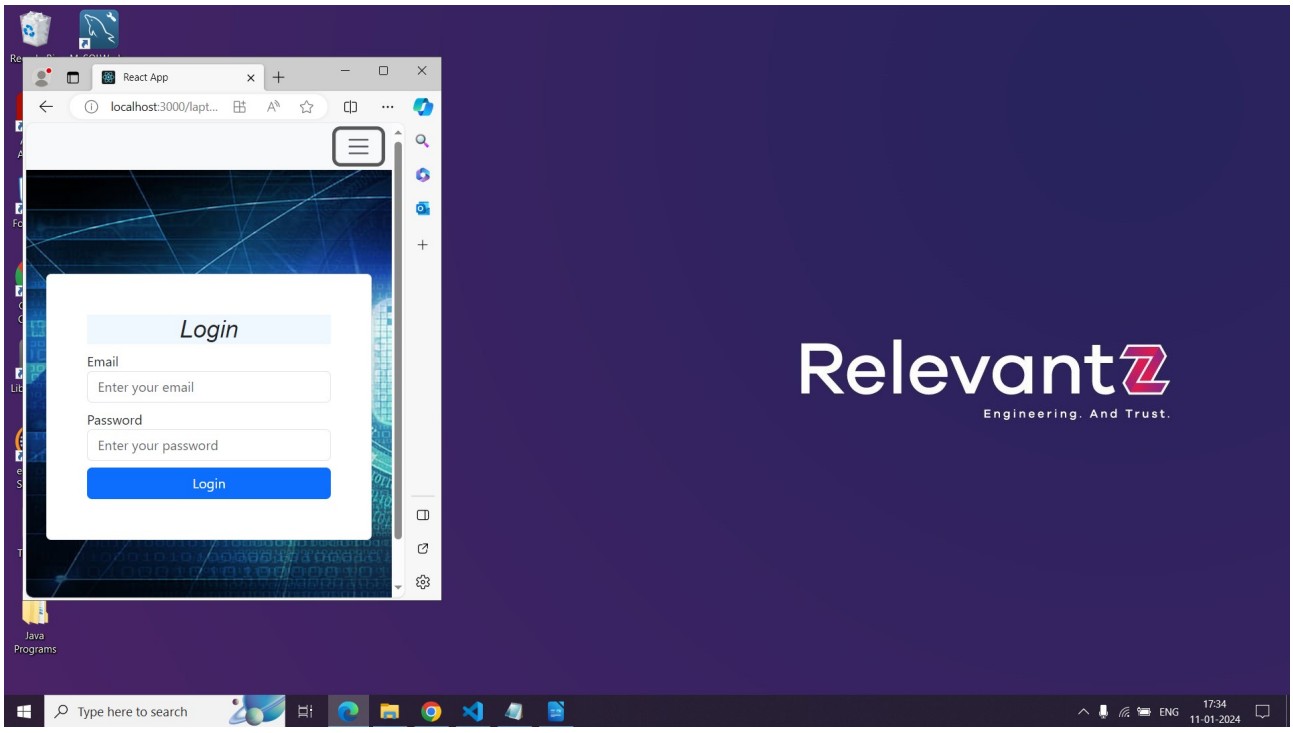

React App
localhost:3000
Home
Login
Register
Welcome to the Laptop Allocation Management System!
Welcome to the Laptop Allocation Management System (LAMS), a web-based application that allows you to request, assign, and track laptops and other IT assets for your employees.
LAMS is designed to simplify and streamline the process of IT asset management, from planning and ordering to operation and
Java Programs
Type here to search
ENG
17:33
11-01-2024
Relevantz
Engineering. And Trust.

Login page:

Large:

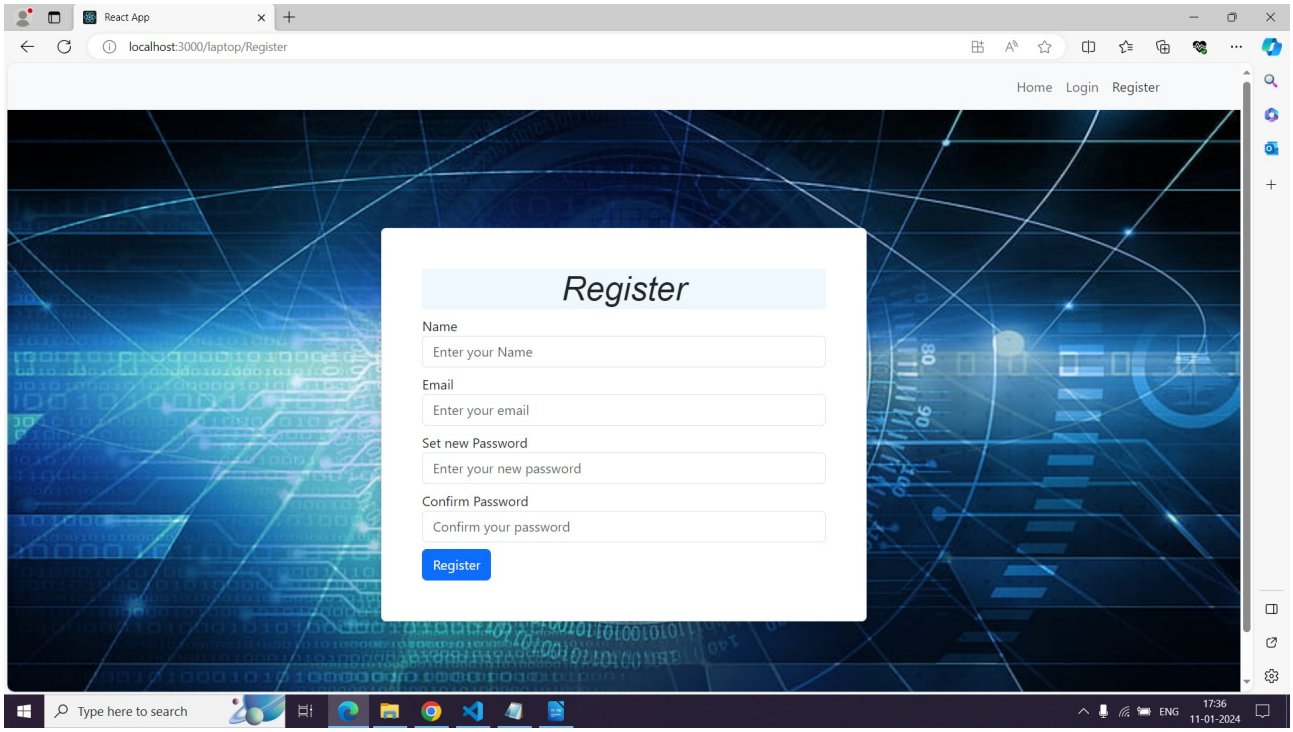


Small:

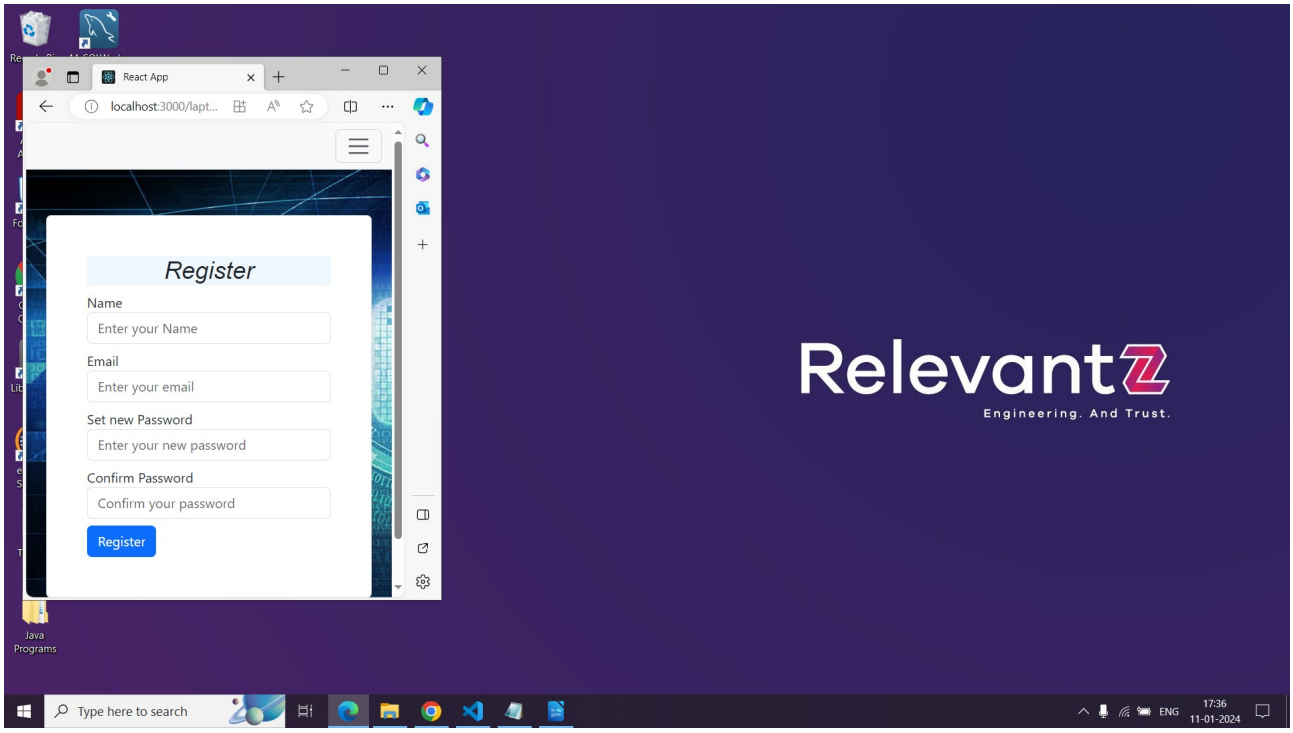


Register page:

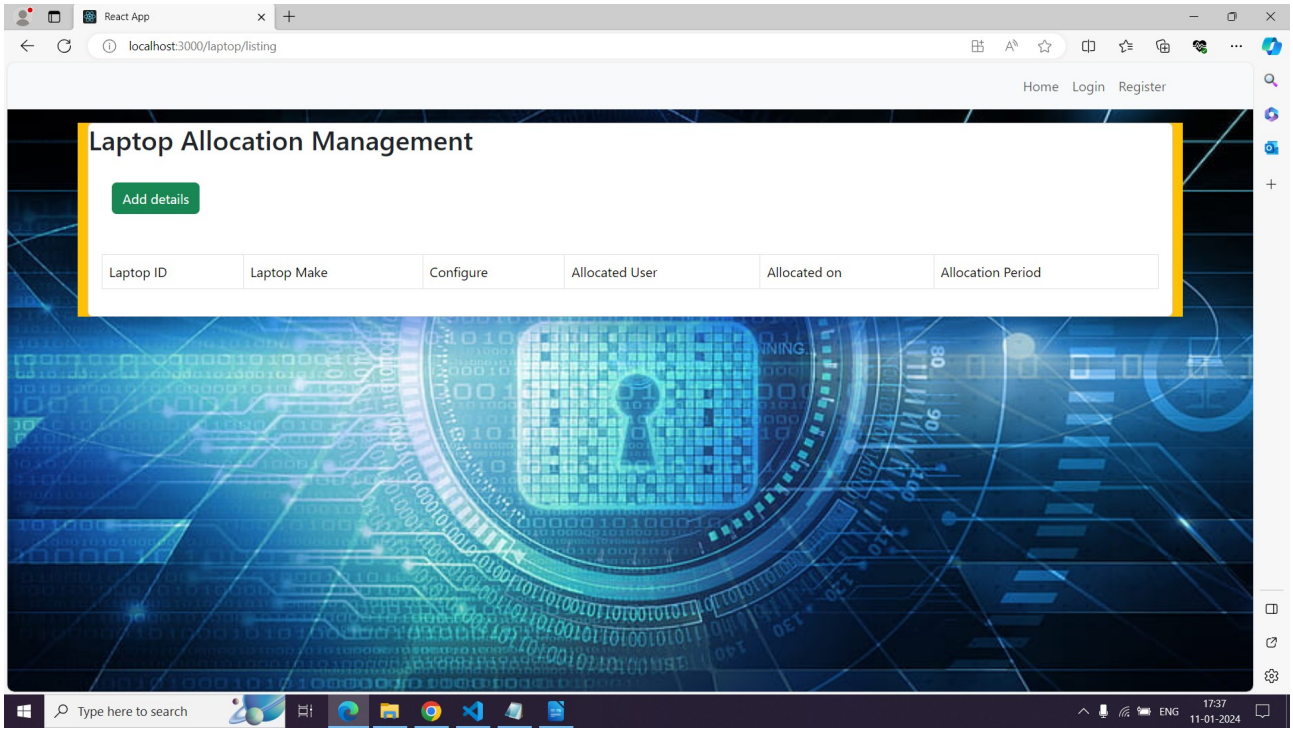
Large:



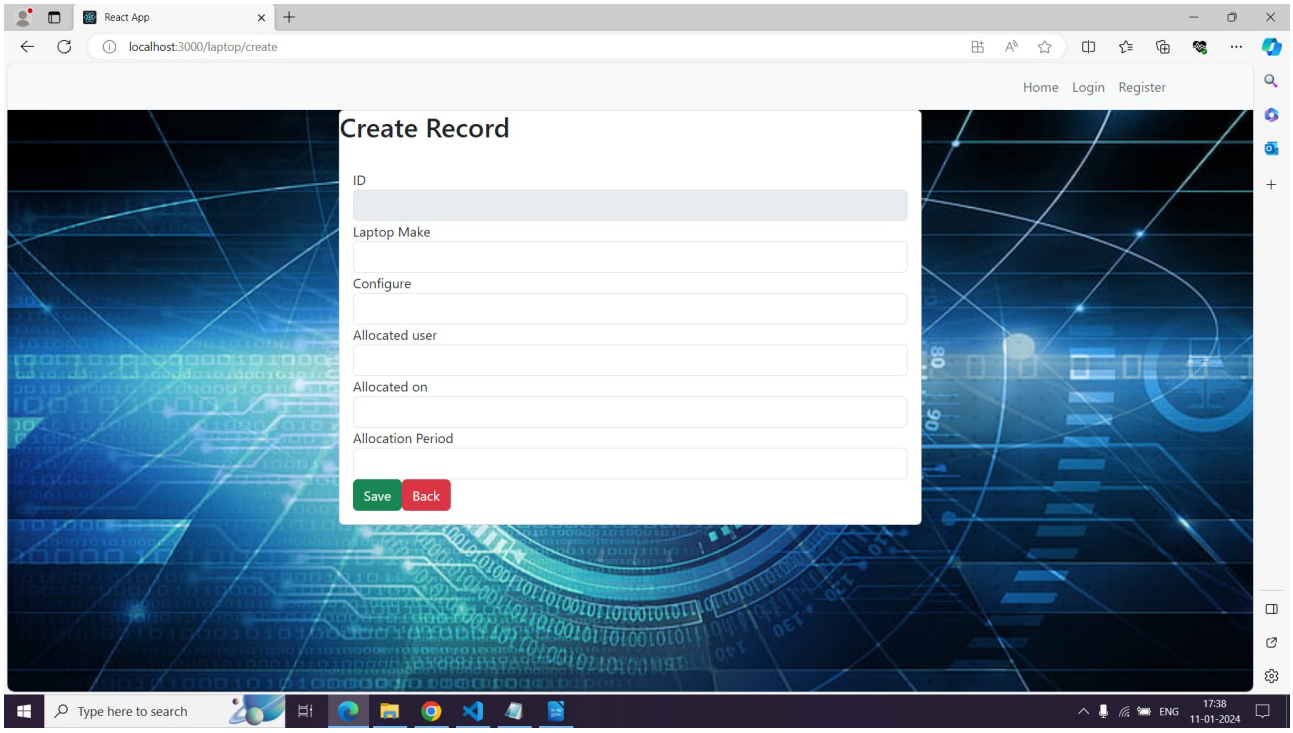
Small:



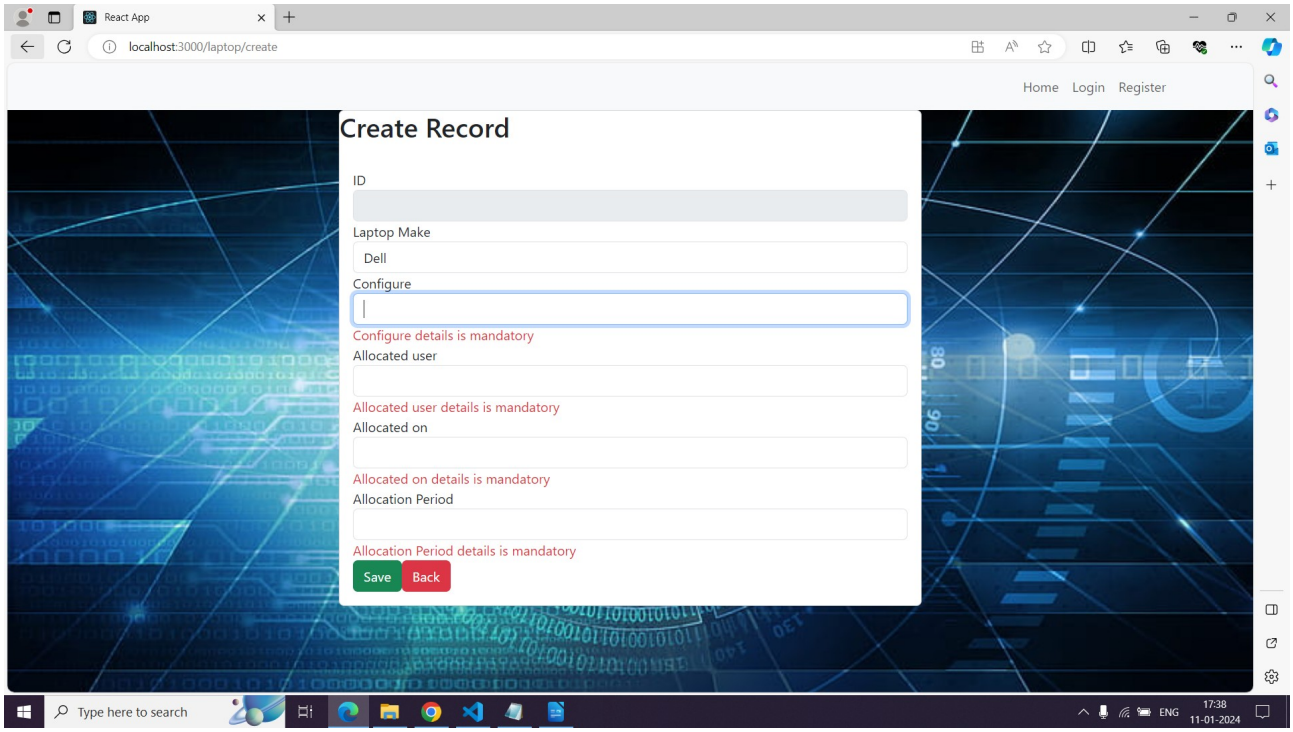
Main Page:



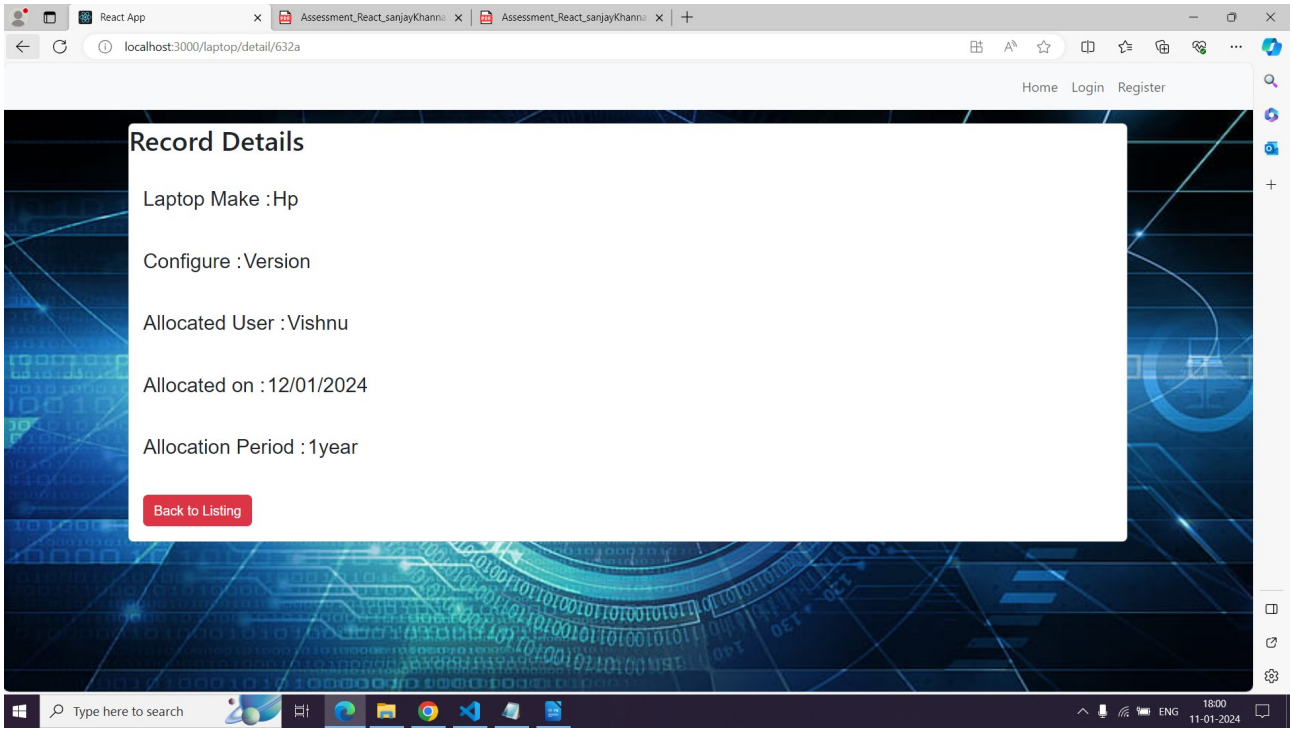
Add Operation:



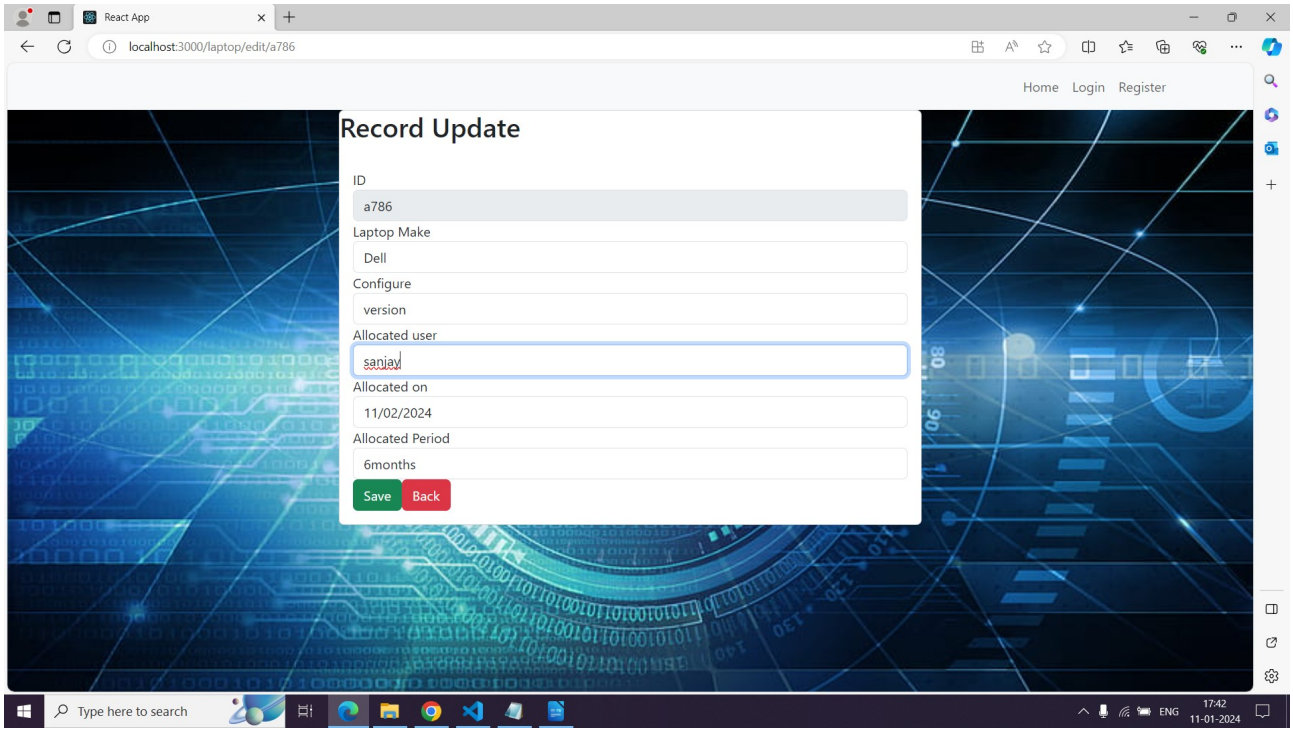
Add Operation with Validation:



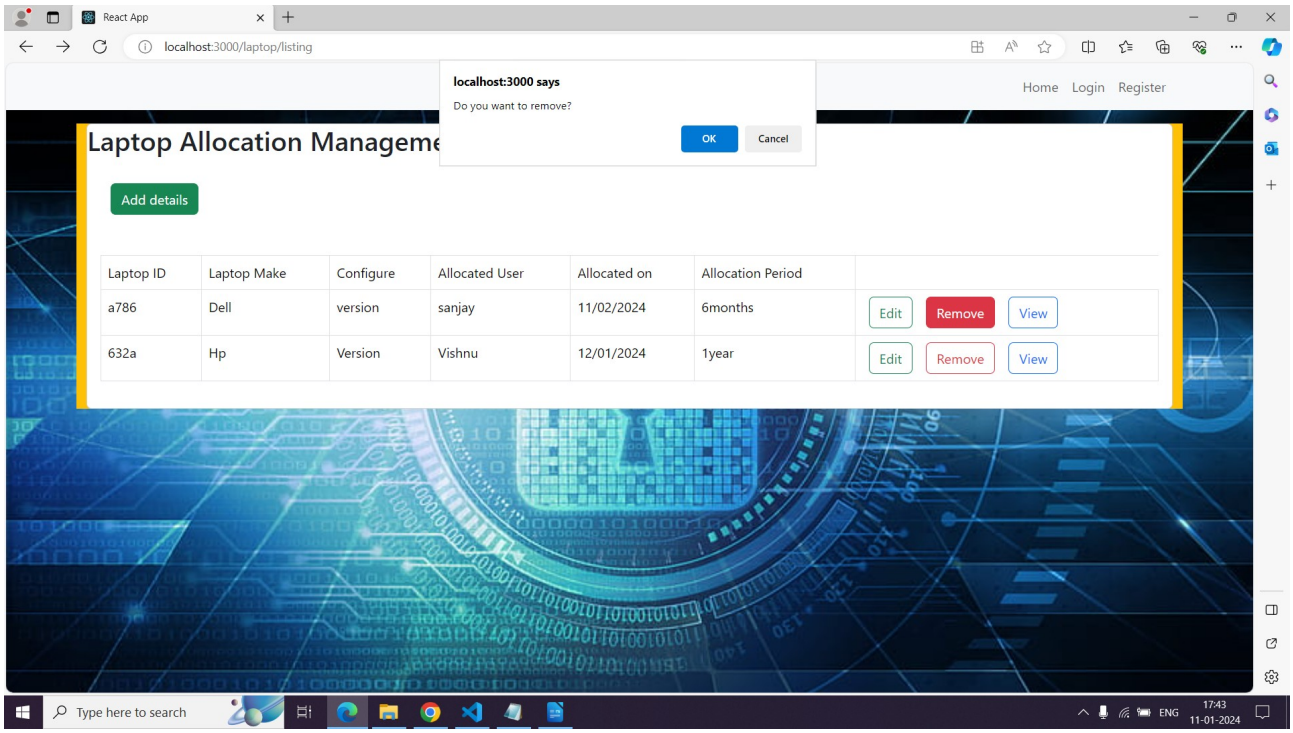
View Operation:

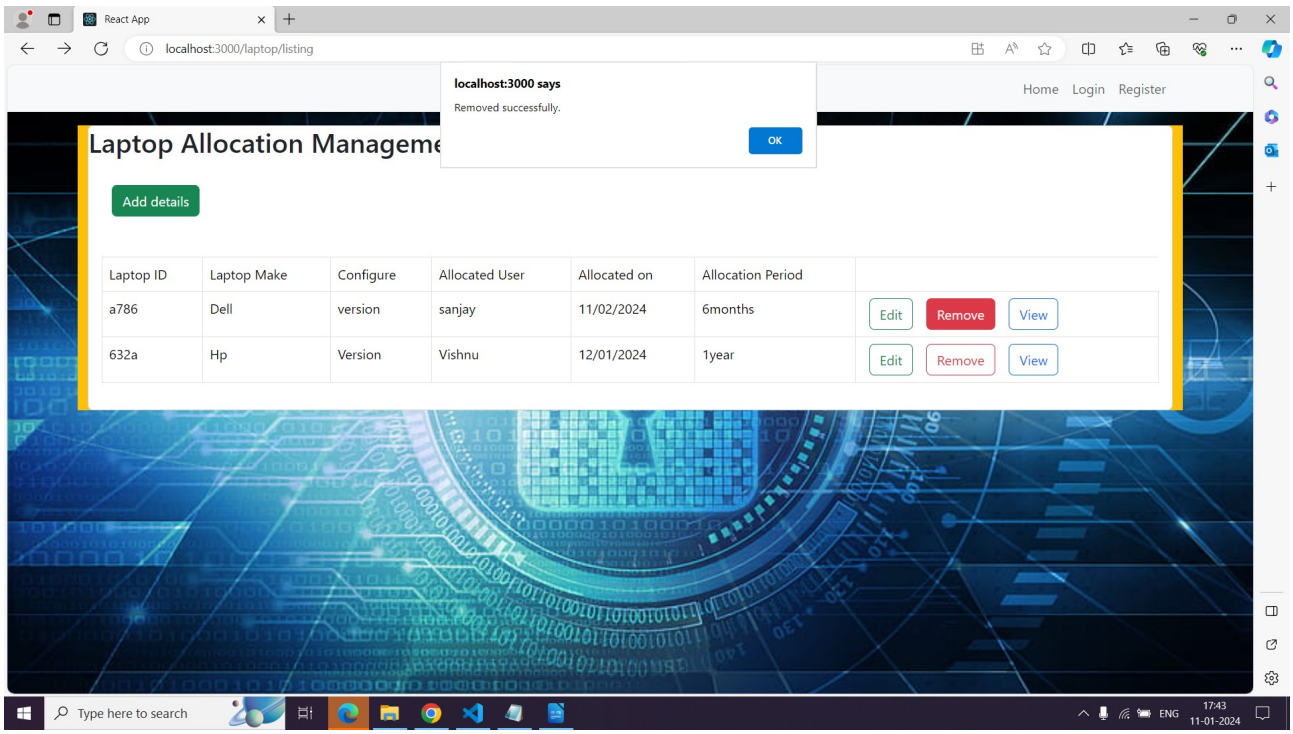


Edit Page:



Delete Page:




React App
localhost:3000/laptop/listing
localhost:3000 says
Removed successfully.
OK
Home
Login
Register
Laptop Allocation Manageme
Add details
Laptop ID
Laptop Make
Configure
Allocated User
Allocated on
Allocation Period
a786
Dell
version
sanjay
11/02/2024
6months
Edit
Remove
View
632a
Hp
Version
Vishnu
12/01/2024
1year
Edit
Remove
View
Type here to search
ENG
17:43
11-01-2024