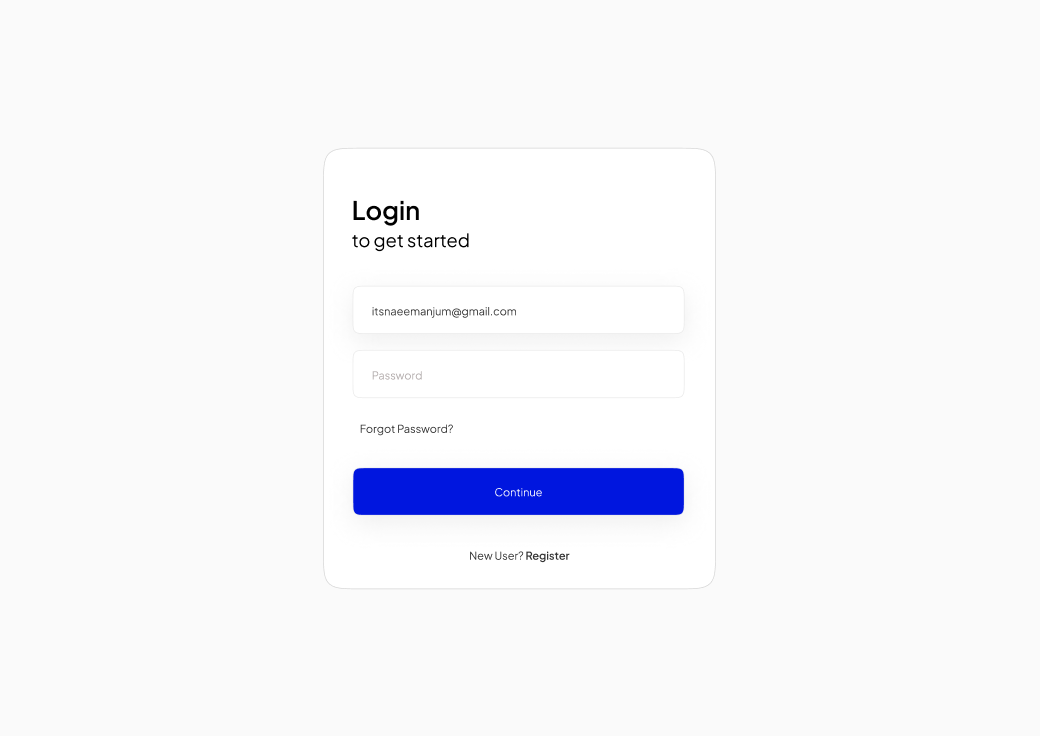
**Task**

**Aim:** Create only frontend using React for below Images

**Login Page:**

****

**Registration Page:**

**A screen shot of a login form

Description automatically generated**

**Description:**  
React is a JavaScript library for building user interfaces, providing a declarative and efficient way to create interactive and dynamic UI components. Developed and maintained by Facebook, React simplifies the process of building scalable and reusable UI components, facilitating the creation of robust and efficient web applications.

**Source Code:**

**Login.js**

import React, { useState } from 'react';

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

import './AuthForm.css'; // Import your CSS file

const Login = () => {

  const [formData, setFormData] = useState({

    email: '',

    password: '',

  });

  const [loginStatus, setLoginStatus] = useState(null);

  const navigate = useNavigate();

  const handleChange = (e) => {

    setFormData({ ...formData, [e.target.name]: e.target.value });

  };

  const handleSubmit = (e) => {

    e.preventDefault();

    // Simulate login success for demonstration purposes

    // In a real application, you would perform authentication with a backend

    console.log('Login Data:', formData);

    // Simulate a successful login for demonstration purposes

    // In a real application, you would check user credentials with a backend

    const isLoginSuccessful = true;

    if (isLoginSuccessful) {

      setLoginStatus('success');

      setTimeout(() => {

        navigate('/dashboard');

      }, 2000);

    }

  };

  const handleForgotPassword = () => {

    // Handle the "Forgot Password" functionality

    alert('Forgot Password: Implement your password recovery logic.');

  };

  const handleCreateAccount = () => {

    // Redirect to the registration page

    navigate('/register');

  };

  return (

    <div className="auth-container">

      <h2>Login</h2>

      {loginStatus === 'success' && (

        <p className="success-message">Login successful! Redirecting...</p>

      )}

      <form onSubmit={handleSubmit}>

        <label>

          Email:

          <input type="email" name="email" value={formData.email} onChange={handleChange} />

        </label>

        <label>

          Password:

          <input type="password" name="password" value={formData.password} onChange={handleChange} />

        </label>

        <div className="form-actions">

          <button type="submit">Login</button>

          <span className="forgot-password" onClick={handleForgotPassword}>

            Forgot Password?

          </span>

        </div>

      </form>

      <p className="create-account-link">

        Don't have an account? <span onClick={handleCreateAccount}>Create an Account</span>

      </p>

    </div>

  );

};

export default Login;

**Register.js**

import React, { useState } from 'react';

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

import './AuthForm.css'; // Import your CSS file

const Register = () => {

  const [formData, setFormData] = useState({

    username: '',

    email: '',

    password: '',

    confirmPassword: '',

    agreeTerms: false,

  });

  const [registrationStatus, setRegistrationStatus] = useState(null);

  const navigate = useNavigate();

  const handleChange = (e) => {

    setFormData({ ...formData, [e.target.name]: e.target.type === 'checkbox' ? e.target.checked : e.target.value });

  };

  const handleSubmit = (e) => {

    e.preventDefault();

    if (!formData.agreeTerms) {

      alert("Please agree to the terms and conditions.");

      return;

    }

    console.log('Registration Data:', formData);

    const isRegistrationSuccessful = true;

    if (isRegistrationSuccessful) {

      setRegistrationStatus('success');

      setTimeout(() => {

        navigate('/login');

      }, 2000);

    }

  };

  return (

    <div className="auth-container">

      <h2>Register</h2>

      {registrationStatus === 'success' && (

        <p className="success-message">You have registered successfully!</p>

      )}

      <form onSubmit={handleSubmit}>

        <label>

          Username:

          <input type="text" name="username" value={formData.username} onChange={handleChange} />

        </label>

        <label>

          Email:

          <input type="email" name="email" value={formData.email} onChange={handleChange} />

        </label>

        <label>

          Password:

          <input type="password" name="password" value={formData.password} onChange={handleChange} />

        </label>

        <label>

          Confirm Password:

          <input

            type="password"

            name="confirmPassword"

            value={formData.confirmPassword}

            onChange={handleChange}

          />

        </label>

        <label className="checkbox-label">

          <input

            type="checkbox"

            name="agreeTerms"

            checked={formData.agreeTerms}

            onChange={handleChange}

          />

          {' '}

          I agree to the terms and conditions

        </label>

        <button type="submit">Register</button>

      </form>

      <p className="login-link">

        Already Registered? <span onClick={() => navigate('/login')}>Login</span>

      </p>

    </div>

  );

};

export default Register;

**Dashboard.css**

/\* src/Dashboard.css \*/

.dashboard-container {

    text-align: center;

    margin-top: 50px;

  }

  .success-message {

    font-size: 24px; /\* Increase the text size of the success message \*/

    color: green; /\* Change the text color if desired \*/

  }

**AuthForm.css**

body {

  margin: 0;

  font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;

  background-color: #f4f4f4;

}

.auth-container {

  max-width: 300px;

  margin: auto;

  padding: 20px;

  border: 1px solid #ccc;

  border-radius: 5px;

  box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);

  margin-top: 50px;

  background-color: #fff;

}

form {

  display: flex;

  flex-direction: column;

  gap: 10px;

}

label {

  display: block;

  margin-bottom: 5px;

}

input {

  padding: 8px;

  font-size: 16px;

}

/\* Add the following styles for aligning email and password input fields \*/

label {

  display: flex;

  flex-direction: column;

}

input {

  margin-top: 5px;

}

button {

  padding: 10px;

  background-color: #007bff;

  color: #fff;

  border: none;

  cursor: pointer;

  font-size: 16px;

}

button:hover {

  background-color: #0056b3;

}

p {

  margin-top: 10px;

  font-size: 14px;

}

/\* src/AuthForm.css \*/

body {

  margin: 0;

  font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;

  background-color: #f4f4f4;

}

.auth-container {

  max-width: 300px;

  margin: auto;

  padding: 20px;

  border: 1px solid #ccc;

  border-radius: 5px;

  box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);

  margin-top: 50px;

  background-color: #fff;

}

form {

  display: flex;

  flex-direction: column;

  gap: 10px;

}

label {

  display: flex;

  flex-direction: column;

}

input {

  padding: 8px;

  font-size: 16px;

}

/\* Add the following styles for the checkbox \*/

label.checkbox-label {

  display: flex;

  align-items: center;

}

.checkbox-label input {

  margin-top: 0;

  margin-right: 5px;

}

/\* src/AuthForm.css \*/

/\* ... (your existing styles) \*/

.login-link {

  margin-top: 10px;

  font-size: 14px;

  text-align: center;

}

.login-link span {

  color: #007bff;

  cursor: pointer;

}

/\* src/AuthForm.css \*/

/\* ... (your existing styles) \*/

.forgot-password {

  margin-top: 10px;

  font-size: 14px;

  color: #007bff;

  cursor: pointer;

}

.form-actions {

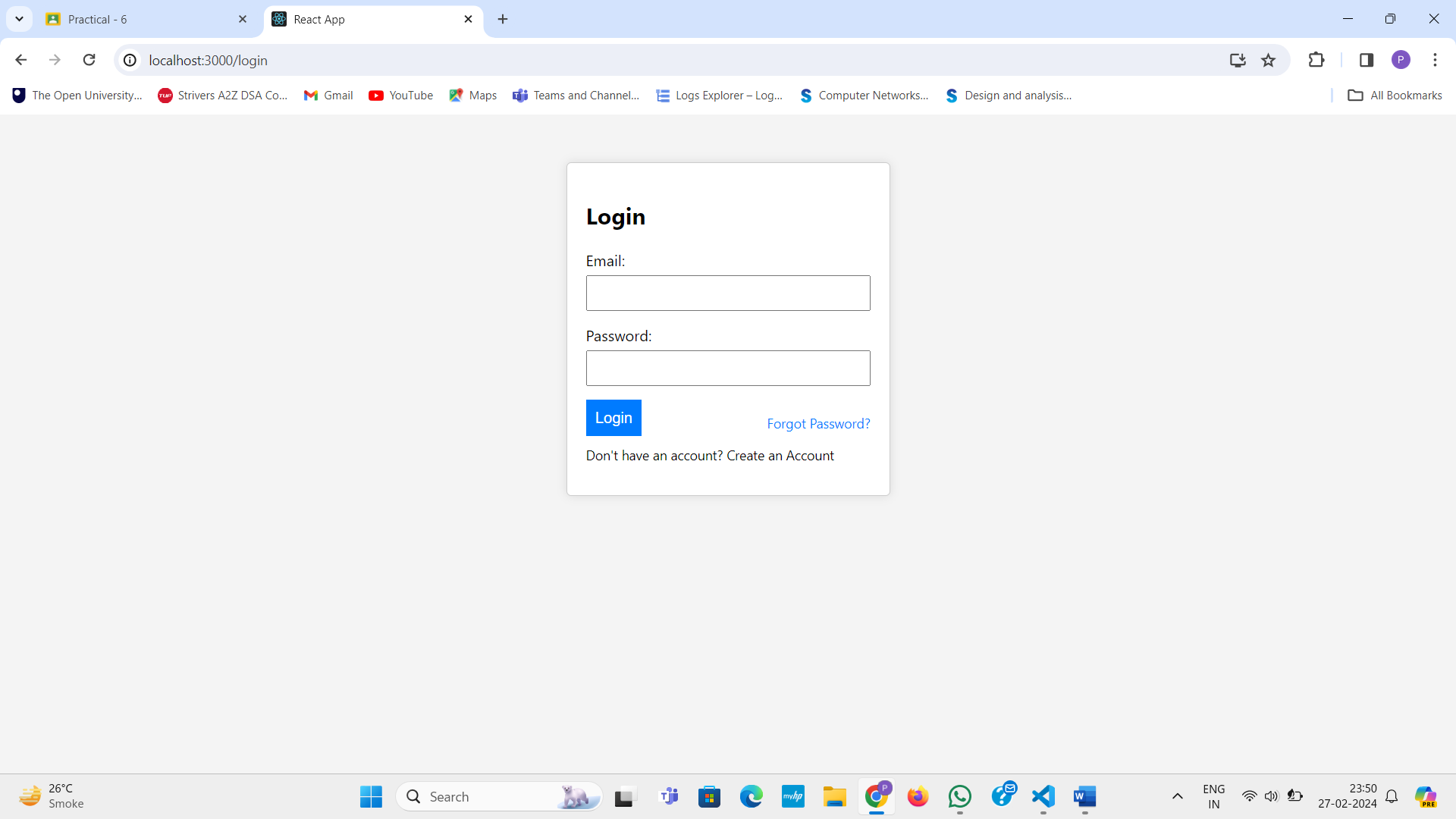
  display: flex;

  justify-content: space-between;

  align-items: center;

}

**Output:**

****

**A screenshot of a computer

Description automatically generated**

**A screenshot of a computer

Description automatically generated**

**A screenshot of a computer

Description automatically generated**

**Learning Outcome:**

In this task we learn about how React js.