Worksheet-6

Student Name: Harsh **UID: 24MCA20045**
Branch: MCA (General) **Section/Group: 1/A**
Semester: 2nd **Date of Performance: 11-04-25**
Subject Name: Front End Technologies **Subject Code: 24CAH-655**

1. Aim/Overview of the practical:

Create change password, and forget password forms using different React controls.

2. Objective:

- The Change Password form typically includes input fields for:

1. Current Password
2. New Password
3. Confirm New Password

- The form performs validation to ensure:

1. The current password is provided.
2. The new password meets security requirements (e.g., length, special characters).

Coding :

(app.js)

```
import React, { useState } from 'react'; import Login from './component/login'; import Signup from './component/signup';

function App() {
```

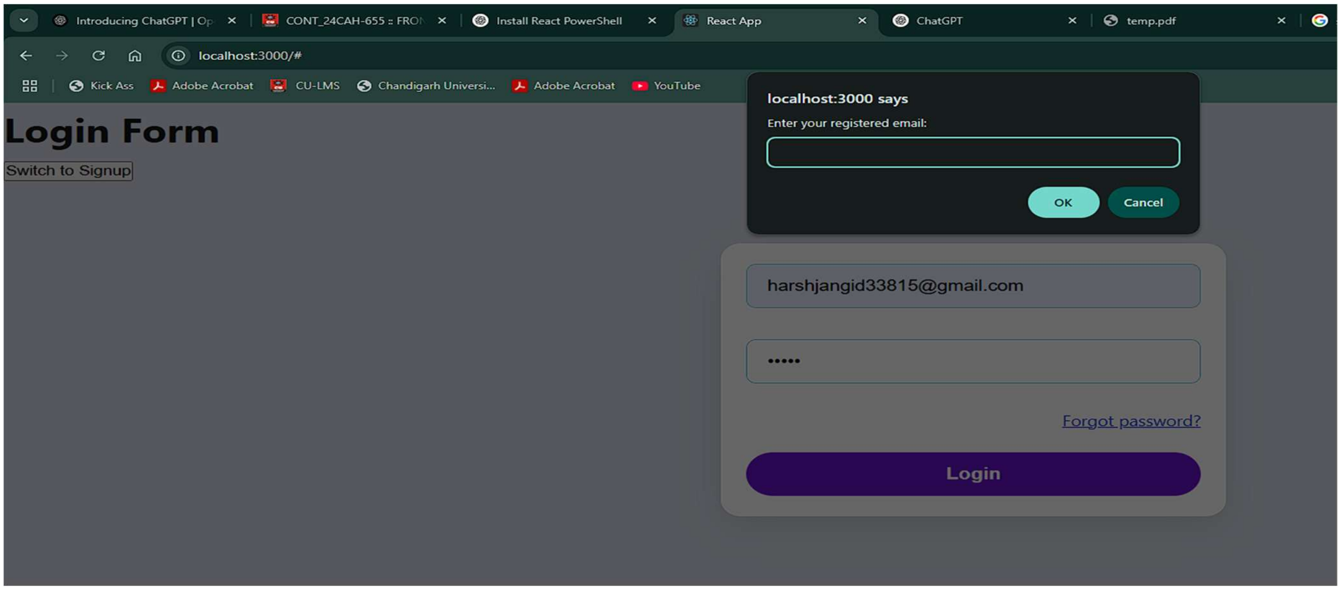
```
const [showLogin, setShowLogin] = useState(true);

return (

<div className="App">

<h1>{showLogin ? 'Login' : 'Signup'} Form</h1>

<button onClick={() => setShowLogin(!showLogin)}> Switch to {showLogin ? 'Signup' : 'Login'}

</button>

{showLogin ? <Login /> : <Signup />}

</div>

);

}

export default App;
```

(signup.js)

```
import React from 'react';

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

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

import * as yup from 'yup';

import './login.css';


// Validation schema

const loginSchema = yup.object().shape({

  email: yup.string().email('Invalid email').required('Email is required'),

  password: yup.string().min(4).required('Password is required'),

});


const Login = () => {
```

```
const {
  register,
  handleSubmit,
  formState: { errors },
} = useForm({ resolver: yupResolver(loginSchema) });

const onSubmit = (data) => {
  const storedUser = JSON.parse(localStorage.getItem('user'));

  if (storedUser && data.email === storedUser.email && data.password === storedUser.password) {
    alert('Login successful!');
  } else {
    alert('Invalid credentials!');
  }
};

const handleForgotPassword = () => {
  const email = prompt('Enter your registered email:');

  if (!email) return;

  // Simulate checking if user exists
  const storedUser = JSON.parse(localStorage.getItem('user'));
  if (storedUser && storedUser.email === email) {
    alert(`A password reset link has been sent to ${email}.`);
  } else {
      alert('No account found with this email.');
    }
  };

  return (
    <form onSubmit={handleSubmit(onSubmit)}>
      <input placeholder="Email" {...register('email')} />
      <p>{errors.email?.message}</p>

      <input type="password" placeholder="Password" {...register('password')} />
      <p>{errors.password?.message}</p>

      <div className="forgot-password">
        <a href="#" onClick={handleForgotPassword}>Forgot password?</a>
      </div>

      <button type="submit">Login</button>
    </form>
  );
};

export default Login;
```

Output:



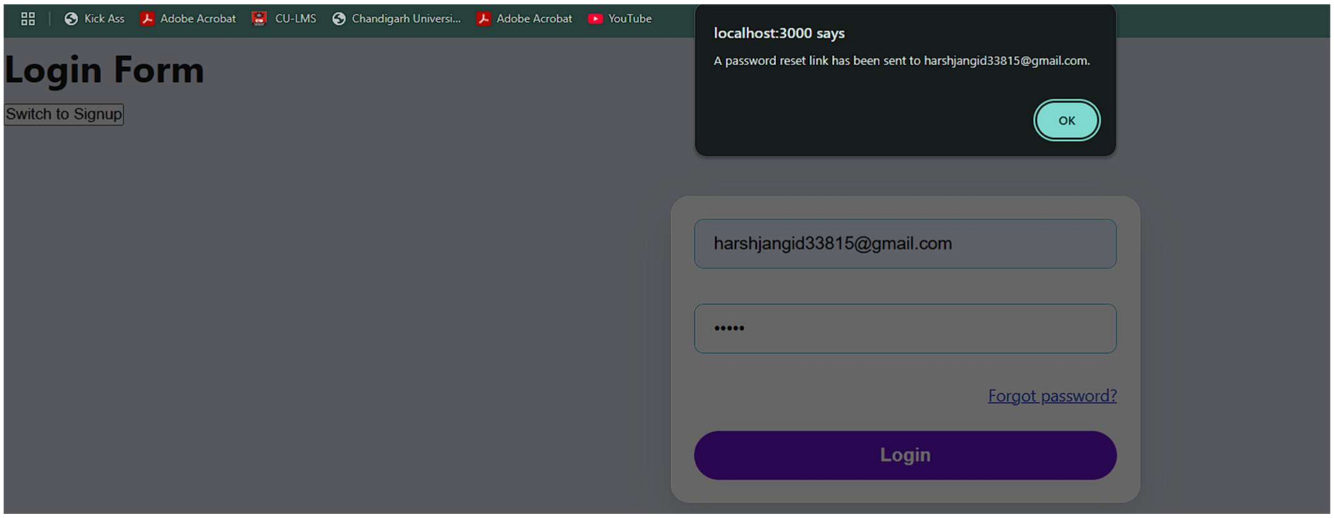
Login Form

localhost:3000 says

Enter your registered email:

harshjangid33815@gmail.com

•••••



Login Form

localhost:3000 says

A password reset link has been sent to harshjangid33815@gmail.com.

harshjangid33815@gmail.com

•••••

3. Learning outcomes (What I have learnt):

- Understand form handling in React (useState, form events).
- Implement password validation logic.
- Learn secure password update workflows.
- Connect React with backend APIs using `fetch` or `axios`.
- Implement controlled components and conditional rendering for form messages (e.g., error, success).

5. Evaluation Grid:

Sr. No.	Parameters	Marks Obtained	Maximum Marks
1.	Worksheet Completion		08 marks
2.	Conduct		12 marks
3.	Viva Voice		10 marks
	Total		30 marks