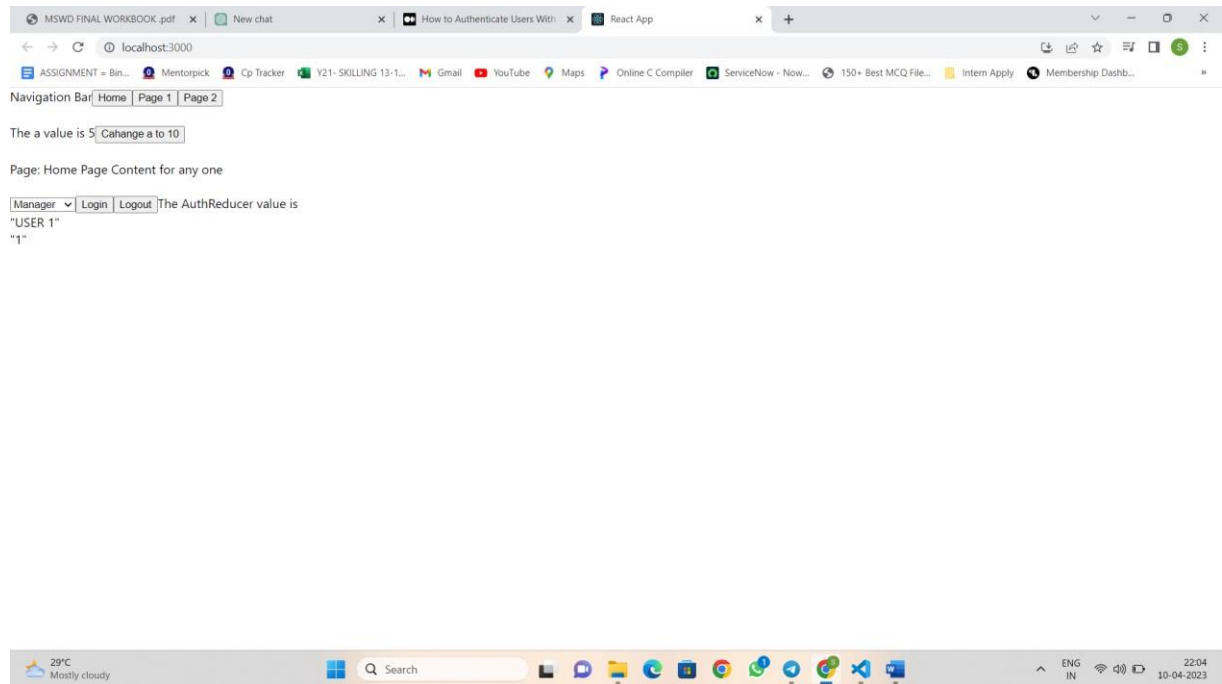
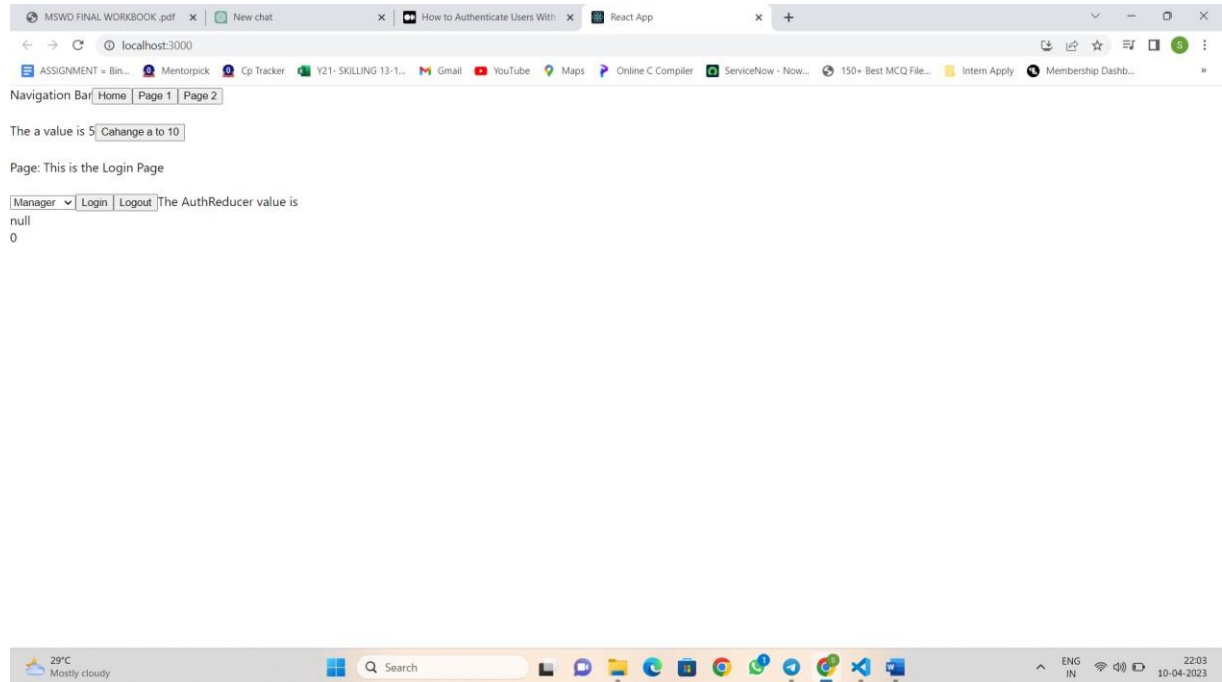


SKILL-12

Id: 2100031023 Name: S
Likith Sai Reddy Output:



APP.js:

```
// import {useState} from "react";  
// import axios from 'axios';
```

```
// function App({store}) {

//   const [a,setA]=useState(5);

//   function changeA()
//   {
//     setA(10);
//     console.log(a);
//   }

//   function loginHandle() {
//     axios.post('http://localhost:8082/login',{
//       un: "USER 1",
//       pw: "PASS 1"
//     }).then((response)=>{
//       console.log(response);
//       localStorage.setItem("token", response.data.token);
//       console.log(localStorage.getItem("token"));
//     });

//     store.dispatch({type: "login", data:{un:"USER 1",
// role:document.getElementById("role").value}});
//     store.dispatch({type:"change", data:"Home Page Content For AnyOne"})
//   }

//   function logoutHandle() {
//     store.dispatch({type: "logout"});
//     store.dispatch({type:"change", data:"This is the Login page"})
//   }

//   function homeHandle() {
//     if(store.getState().AuthReducer[1] ==1 ||
store.getState().AuthReducer[1] == 2)
//     {
//       store.dispatch({type:"change", data:"Home Page Content For AnyOne"})
//     }
//     else
//     {
//       store.dispatch({type:"change", data:"Error Page"})
//     }
//   }

//   function page1Handle() {
//     if(store.getState().AuthReducer[1] ==1 ||
```

```

store.getState().AuthReducer[1] == 2)
//    {
//        store.dispatch({type:"change", data:"Page1 Content For Manager and
Employee"})
//    }
//    else
//    {
//        store.dispatch({type:"change", data:"Error Page"})
//    }
// }

// function page2Handle() {
//     if(store.getState().AuthReducer[1] == 1) //
//     {
//         store.dispatch({type:"change", data:"Page2 For Manager Only"})
//     }
//     else
//     {
//         store.dispatch({type:"change", data:"Error Page"})
//     }
// }

// return(
//     <div>

//         Navigation Bar
//         <button onClick={homeHandle}> Home </button>
//         <button onClick={page1Handle}> Page1 </button>
//         <button onClick={page2Handle}> Page2 </button>
//         <br/>
//         <br/>

//         The a Value is: {a}
//         <button onClick={changeA}> Change a to 10</button>
//         <br/>
//         <br/>

//         Page: {store.getState().NavReducer}

//         <br/>
//         <br/>

//         {/* total select is an element */}
//         <select id="role">
//             {/* <option value={0}>Select</option>  */}
//             <option value={1}>Manager</option>

```



```

//      <option value={2}>Employee</option>
//    </select>

//    <button onClick={loginHandle}> Login </button>
//    <button onClick={logoutHandle}> Logout </button>
//    The AuthReducer value is {store.getState().AuthReducer.map(p => (
//    <div>
//      {JSON.stringify(p)}
//    </div>
//    ))}

//  </div>
// );
// }

// export default App;

// //JWT and bcrypt
import axios from "axios";
import { useState } from
"react";
function App ({store})
{
  const [a, setA] =
useState(5);
  function changeA()
{
    setA(10);
    console.log(a);
  }
  function loginHandle() {
    axios.post('http://localhost:8082/login', {
      un: "USER 1",          pw: "PASS 1"
    }).then((res)=> {
      console.log(res)
      localStorage.setItem("token", res.data.token);
      console.log(localStorage.getItem("token"));
    })
    store.dispatch({type:"login", data:{un:"USER 1",
role:document.getElementById("role").value}})
    store.dispatch({type:"change", data:"Home Page Content for any one"})  }
  function logoutHandle() {
    store.dispatch({type:"logout"})
  }
}

```

```
store.dispatch({type:"change", data:"This is the Login Page"})
```

```

    }    function homeHandle() {
if(store.getState().AuthReducer[1] == 1 ||
store.getState().AuthReducer[1] == 2) {
store.dispatch({type:"change", data:"Home Page Content for any one"})
}    else {    store.dispatch({type:"change", data:"Error
Page"})    }
    }    function page1Handle() {
if(store.getState().AuthReducer[1] == 1 ||
store.getState().AuthReducer[1] == 2) {
    store.dispatch({type:"change", data:"Page 1 for Manager and
Employee"})    }    else {
store.dispatch({type:"change", data:"Error Page"})
    }
    }    function page2Handle() {
if(store.getState().AuthReducer[1] == 1) {
store.dispatch({type:"change", data:"Page 2 for Manager Only"})
    }    else {
store.dispatch({type:"change", data:"Error Page"})
    }
    }
    return (
<div>
    Navigation Bar
    <button onClick={homeHandle}> Home </button>
    <button onClick={page1Handle}> Page 1 </button>
    <button onClick={page2Handle}> Page 2 </button>
    <br/>
    <br/>

    The a value is {a}
    <button onClick={changeA}>Cahange a to 10</button>

```



```
<br/>
```

```
<br/>
```

```
Page: {store.getState().NavReducer}
```

```
<br/>
```

```
<br/>
```

```
<select id="role">
```

```
  <option value={1}>Manager</option>
```

```
  <option value={2}>Employee</option>
```

```
</select>
```

```
<button onClick={loginHandle}> Login </button>
```

```
<button onClick={logoutHandle}> Logout </button>
```

```
The AuthReducer value is {store.getState().AuthReducer.map(p => (
```

```
<div>
```

```
    {JSON.stringify(p)}
```

```
</div>
```

```
))}
```

```
</div>
```

```
);
```

```
} export default
```

```
App
```

Server.js:

```
// const express = require('express')
// const cors = require('cors')
// const bcryptjs = require('bcryptjs');
// const jwt = require('jsonwebtoken')
// // const { MongoClient } = require('mongodb')
// // const { request, response } = require('express')

// const app = express()
// app.use(cors()) // cros orgin resource sharing
// app.use(express.json())

// // const uri =
```

```
"mongodb+srv://admin:rocky9886@cluster0.nczlwja.mongodb.net/?retryWrites=true&w=majority"
// // const client = new MongoClient(uri)
// // client.connect()
// // var db = client.db("ex5")
// // var col = db.collection("c207")

// var key="dhiuldh";
// var algorithm=""
```

```
// app.get('/',(req,res) => {
//   res.send("This is server page");
// })

// app.post('/login', async (req,res) => {                                //npm i bcryptjs ---->
//   to encrypt data
//   console.log(req.body);
//   var pw = null;
//   var pass_check=false;
//   var token= null;

//   await bcryptjs.hash("PASS 1", 5).then(response => {
//     console.log(response);
//     pw = response;
//   })
//   console.log(pw);
//   token =await jwt.sign(req.body, key, {
//     algorithm: algorithm,
//     expiresIn: '2m'
//   })
//   bcryptjs.compare(req.body.pw, pw).then(res => {
//     pass_check=res;
//   })
//   if(req.body.un == "USER 1" && pass_check == true)
//   {
//     res.status(200).json({
//       message: "Sucessfull",
//       token: token
//     });
//   }
//   else{
//     res.status(401).json({
//       message: "Failed"
//     });
//   }
// })

// app.listen(8082)
// console.log("server started")

const express = require('express');
const cors = require('cors'); const
bcryptjs = require('bcryptjs')
```

```
const jwt =  
require('jsonwebtoken');
```

```
const app = express();  
app.use(express.json())  
;
```

```
app.use(cors());
var key = "dhiuldh";
var algorithm =
"HS256";
app.get('/',(req, res) => {
res.send("This is a Server Page");
}) app.post('/login', async (req, res) => {
console.log(req.body);    var pw = null;    var
pass_check = false;    var token = null;
await bcryptjs.hash("PASS 1", 5).then(res => {
pw = res;
    })    console.log(pw);    await
bcryptjs.compare(req.body.pw, pw).then(res => {
pass_check = res;
    })    console.log(pass_check);
token = await jwt.sign(req.body, key, {
algorithm: algorithm,    expiresIn:
'2m'
    })    if(req.body.un == "USER 1" && pass_check ==
true) {        res.status(200).json({
message: "success",        token: token
    });    }    else {
res.status(401).json({
message: "fail"
    });
    }
}) app.listen(8082);
console.log("Server
Started");
```