**Muhammad Abdullah**

**SE(6A) | 19F-0916**

Web Engineering Lab

Lab 14: User Authentication

**TASK 1**

**Server.js code**

const express = require('express')

const app = express()

const { users, ROLE } = require('./data')

const {authUser,authRole,authRole2} = require('./basicAuthentication')

const projectRouter = require('./routes/projects')

app.use(express.json())

app.use(setUser)

app.use('/attendance', projectRouter)

app.get('/home', (req, res) => {

  res.send('Home Page')

})

app.get('/markattendance', authUser,authRole(ROLE.ADMIN,ROLE.TEACHER),(req, res) => {

  res.send('Mark Attendance Page')

})

app.get('/viewattendance', authUser, authRole(ROLE.STUDENT,ROLE.ADMIN,ROLE.TEACHER),(req, res) => {

  res.send('View Attendance Page')

})

function setUser(req, res, next) {

  const userId = req.body.userId

  if (userId) {

    req.user = users.find(user => user.id === userId)

  }

  next()

}

app.listen(3000)

**basicAuthentication.js code**

const { ROLE } = require('./data')

function authUser(req,res,next){

    if (req.user == null) {

        res.status(403)

        return res.send("You need to sign in")

    }

    next()

}

function authRole(role,role2,role3){

    return(req,res,next) => {

        if(req.user.role !==  role && req.user.role !== role2 && req.user.role !== role3){

            res.status(401)

            return res.send("Not Allowed")

        }

        next()

    }

}

module.exports={

    authUser,

    authRole

}

**project.js code**

const express = require('express')

const { authUser } = require('../basicAuthentication')

const router = express.Router()

const { attendance } = require('../data')

const {canViewAttendance} = require('../permissions/attendance')

router.get('/', (req, res) => {

  res.json(attendance)

})

router.get('/:attendanceId', setAttendance, authUser, authGetAttendance,(req, res) => {

  console.log(req.attendance);

  res.json(req.attendance)

})

function setAttendance(req, res, next) {

  const attendanceId = parseInt(req.params.attendanceId)

  console.log(attendanceId);

  req.attendance = attendance.find(attendance => attendance.id === attendanceId)

  console.log(req.attendance);

  if (req.attendance == null) {

    res.status(404)

    return res.send('Attendance not found')

  }

  next()

}

function authGetAttendance(req,res,next){

  if(!canViewAttendance(req.users,req.attendance)){

    console.log(req.users)

    console.log(req.attendance)

    res.status(401);

    res.send("Not Allowed");

  }

  next()

}

module.exports = router

**attendance.js code**

const {ROLE} = require('../data');

function canViewAttendance (users,attendance){

    return (

        users.role === ROLE.STUDENT ||

        attendance.userId === users.id

    )

}

module.exports = {

    canViewAttendance

}

**data.js code**

const ROLE = {

  ADMIN: 'admin',

  TEACHER: 'teacher',

  STUDENT: 'student'

}

module.exports = {

  ROLE: ROLE,

  users: [

    { id: 1, name: 'admin', role: ROLE.ADMIN },

    { id: 2, name: 'teacher', role: ROLE.TEACHER },

    { id: 3, name: 'student', role: ROLE.STUDENT }

  ],

  attendance: [

    { id: 3, name: "Abdullah", userId: 1 },

    { id: 3, name: "Jonathan", userId: 2 },

    { id: 3, name: "Zain ul Abideen", userId: 3 },

    { id: 3, name: "Ali Ahmad", userId: 4 },

    { id: 3, name: "Osama", userId: 5 }

  ]

}

**RUNNING CODE OUTPUT FOR ATTENDANCE:**

**Text

Description automatically generated**

**TASK 1 Completed**

**TASK 2**

**Server.js code**

const express = require('express')

const app = express()

const { users, ROLE } = require('./data')

const {authUser,authRole,authRole2} = require('./basicAuthentication')

const projectRouter = require('./routes/projects')

app.use(express.json())

app.use(setUser)

app.use('/attendance', projectRouter)

app.get('/home', (req, res) => {

  res.send('Home Page')

})

app.get('/markattendance', authUser,authRole(ROLE.ADMIN,ROLE.TEACHER),(req, res) => {

  res.send('Mark Attendance Page')

})

app.get('/viewattendance', authUser, authRole(ROLE.STUDENT,ROLE.ADMIN,ROLE.TEACHER),(req, res) => {

  res.send('View Attendance Page')

})

app.get('/deleteattendance', authUser, authRole(ROLE.ADMIN,ROLE.TEACHER),(req, res) => {

  res.send('Delete Attendance Page')

})

function setUser(req, res, next) {

  const userId = req.body.userId

  if (userId) {

    req.user = users.find(user => user.id === userId)

  }

  next()

}

app.listen(3000)

**basicAuthentication.js code**

const { ROLE } = require('./data')

function authUser(req,res,next){

    if (req.user == null) {

        res.status(403)

        return res.send("You need to sign in")

    }

    next()

}

function authRole(role,role2,role3){

    return(req,res,next) => {

        if(req.user.role !==  role && req.user.role !== role2 && req.user.role !== role3){

            res.status(401)

            return res.send("Not Allowed")

        }

        next()

    }

}

module.exports={

    authUser,

    authRole

}

**project.js code**

const express = require('express')

const { authUser } = require('../basicAuthentication')

const router = express.Router()

const { attendance } = require('../data')

const {canViewAttendance} = require('../permissions/attendance')

router.get('/', (req, res) => {

  res.json(attendance)

})

router.get('/:attendanceId', setAttendance, authUser, authGetAttendance,(req, res) => {

  console.log(req.attendance);

  res.json(req.attendance)

})

function setAttendance(req, res, next) {

  const attendanceId = parseInt(req.params.attendanceId)

  console.log(attendanceId);

  req.attendance = attendance.find(attendance => attendance.id === attendanceId)

  console.log(req.attendance);

  if (req.attendance == null) {

    res.status(404)

    return res.send('Attendance not found')

  }

  next()

}

function authGetAttendance(req,res,next){

  if(!canViewAttendance(req.users,req.attendance)){

    console.log(req.users)

    console.log(req.attendance)

    res.status(401);

    res.send("Not Allowed");

  }

  next()

}

module.exports = router

**attendance.js code**

const {ROLE} = require('../data');

function canViewAttendance (users,attendance){

    return (

        users.role === ROLE.STUDENT ||

        attendance.userId === users.id

    )

}

module.exports = {

    canViewAttendance

}

**data.js code**

const ROLE = {

  ADMIN: 'admin',

  TEACHER: 'teacher',

  STUDENT: 'student'

}

module.exports = {

  ROLE: ROLE,

  users: [

    { id: 1, name: 'admin', role: ROLE.ADMIN },

    { id: 2, name: 'teacher', role: ROLE.TEACHER },

    { id: 3, name: 'student', role: ROLE.STUDENT }

  ],

  attendance: [

    { id: 1, name: "Abdullah", userId: 1 },

    { id: 2, name: "Jonathan", userId: 2 },

    { id: 3, name: "Zain ul Abideen", userId: 3 },

    { id: 4, name: "Ali Ahmad", userId: 4 },

    { id: 5, name: "Osama", userId: 5 }

  ]

}

**Admin Can Delete Student:**

**Graphical user interface, application

Description automatically generated**

**Teacher Can Delete Student:**

**Graphical user interface, text, application

Description automatically generated**

**Student Can’t Delete Student:**

**Graphical user interface, text, application

Description automatically generated**

**Particular Students can View Their Attendance Only!**

**A screenshot of a computer

Description automatically generated with medium confidence**

**TASK 3**

**App.js Code for REACT App**

import logo from './National\_University\_of\_Computer\_and\_Emerging\_Sciences\_logo-removebg-preview.png';

import './App.css';

function App() {

  return (

    <div className="App">

      <header className="App-header">

        <img src={logo} className="App-logo" alt="logo" />

        <p>

          Roll: 19F-0916 and Name: M Abdullah.

        </p>

      </header>

    </div>

  );

}

export default App;

**Text

Description automatically generated**

Image Animation Time Reduced to 2 Seconds

**Logo

Description automatically generated**

Please Use Code to View the Moving/Animating Icon for FAST NUCES Icon

**TASK 4**

**App.js Code for REACT App**

import './App.css';

function App() {

  function Table() {

    var i = 1;

    var ans = "";

    var table = +document.getElementById("firstNumber").value;

    var result = document.getElementById("result");

    for (i = 1; i <= 10; i++) {

      ans += ("<br>" + table + " x " + i + " = "  + (table \* i) );

    }

    ans += "";

    result.innerHTML = ans;

  }

  return (

    <div class="container">

      <br></br>

      <div class="format">

        <div class="card-body">

          <h2 >React Table Calculation</h2>

          <form id="final">

            <div >

              <label>Enter the Number to Calculate : </label>

              <input className='field' type="text" id="firstNumber" />

            </div>

            <button className='btn' type="button" onClick={Table} id="submit">Calculate Results</button>

            <p id="result">

            </p>

          </form>

        </div>

      </div>

    </div>

  );

}

export default App;

**CSS File XML**

.format{

  font-size: 30px;

  text-align: center;

  margin-top: 1%;

  margin-left: 5%;

}

.btn{

  font-size: 30px;

  text-align: center;

  margin-top: 1%;

}

.field{

  font-size: 20px;

  text-align: center;

  margin-top: 1%;

}

**Output For Form**

**Table

Description automatically generated**

**TASK 5**

**React App Code**

import './App.css';

function App() {

  return (

    <div className = "form-box">

        <h5 className = "form-step"> REACT FORM </h5>

        <form>

            <div className = "field1">

            <input placeholder="Name"/>

            <input placeholder="UserName"/>

            <input placeholder="Password"/>

            <input placeholder="Confirm Password"/>

            </div>

            <button type = "submit" id= "submitBtn" className = "sbtBtn"> Submit Details</button>

        </form>

    </div>

  );

}

export default App;

**CSS File XML**

\*{

  box-sizing: border-box;

  margin: 0;

  padding: 0;

  border: 3px solid rgb(15, 14, 14);

}

.form-box{

  min-width: 100%;

  height: 100vh;

  background-color: rgb(110, 77, 208);

  margin: 0vw;

  padding: 5vw ;

  position:relative;

  text-align: center;

}

.form-box h5{

  font-size: 35px ;

  text-align: center;

  color: #fdfdfd;

  margin: 0 0 .5vh 0;

}

.form-box label{

  display: block;

  text-align: center;

  font-size: 20px;

  margin: 0 0 2vh 0;

}

.form-box input{

  display: block;

  width: 100%;

  padding: .5rem .8rem .5rem .8rem;

  margin: .9vw 0 ;

  border:0;

  border-radius: 5px;

  text-align: center;

  font-size: 25px;

}

.sbtBtn{

  display: inline-block;

  width: 50%;

  height: 3rem;

  text-align: center;

  color: rgb(2, 1, 1);

  border: 3px solid rgb(15, 14, 14);

  bottom: 0;

  cursor: pointer;

  font-size: 30px;

  border-radius: 20px 20px 20px 20px;

}

**Output For Form**

**Graphical user interface, application

Description automatically generated**