



## DEPARTMENT OF INFORMATION TECHNOLOGY

**COURSE CODE: DJS22ITL604**

**DATE:**

**COURSE NAME: Full Stack Web Development Laboratory**

**CLASS: TYBTech**

**Name: Anish Sharma**

**Div:IT-1**

**Roll no:I011**

**EXPERIMENT NO. 06**

**CO/LO:** CO1-Develop a full stack web application.

**AIM / OBJECTIVE:** React Router and Navigation Implement React Router for basic navigation between different components/pages within a React application.

### **THEORY:**

#### **Introduction to React Router**

React Router is a standard library for routing in React applications. It enables navigation between different components/pages without requiring a full page reload. This improves user experience by creating a seamless Single Page Application (SPA).

#### **Key Concepts:**

1. Routing - Defines how users navigate within the app.
2. BrowserRouter - Wraps the entire application to enable routing.
3. Routes & Route - Specifies different paths and their corresponding components.
4. Link - Used instead of <a> tags for in-app navigation.
5. 404 Page (Not Found) - Displays when an invalid route is accessed.

#### **Step 1: Set Up the React Project**

1. Create a new React app: npx createreact-app react-router-app cd reactrouter-app



Install React Router: npm

install react-router-dom

### **Step 2: Creating the Main Application File Create**

```
src/App.js import React
```

```
from "react";
```

### **DEPARTMENT OF INFORMATION TECHNOLOGY**

```
import { BrowserRouter as Router, Routes, Route } from "react-router-dom"; import Navbar from "./components/Navbar"; import Home from
```

**COURSE CODE: DJS22ITL604**

**DATE:**

```
"./pages/Home"; import About from "./pages/About"; import Contact from "./pages/Contact";
```

**COURSE NAME: Full Stack Web Development Laboratory**

**CLASS: TYBTech**

```
import NotFound from "./pages/NotFound"; const App = () => { return ( <Router>
```

```
<Navbar />
```

```
<Routes>
```

```
  <Route path="/" element={<Home />} />
```

```
  <Route path="/about" element={<About />} />
```

```
  <Route path="/contact" element={<Contact />} />
```

```
  <Route path="*" element={<NotFound />} />
```

```
</Routes>
```

```
</Router>
```

```
);};
```

```
export default App;
```



### Step 3: Creating the Navigation Bar Create

```
src/components/Navbar.js import React from
"react"; import { Link } from
"reactrouter-dom"; const Navbar = () => {
    return (
        <nav style={{ padding: "10px", background: "#f0f0f0" }}>
            <Link to="/" style={{ margin: "10px" }}>Home</Link>
            <Link to="/about" style={{ margin: "10px" }}>About</Link>
            <Link to="/contact" style={{ margin: "10px" }}>Contact</Link>
        </nav>
    );
};
```

export default Navbar;

### Step 4: Creating Page Components with Functionality Create src/pages/Home.js import

```
React, { useState } from "react";
```

## DEPARTMENT OF INFORMATION TECHNOLOGY

**COURSE CODE: DJS22ITL604**

**DATE:** const Home

```
= () => {
```

**COURSE NAME: Full Stack Web Development Laboratory**

**CLASS: TYBTech**

```
const [count, setCount] = useState(0); return (
```

```
<div>
```



```
<h2>Welcome to the Home Page</h2>
<p>Click the button to increase the counter.</p>
<button onClick={() => setCount(count + 1)}>Count: {count}</button> </div>
);
}; export default Home; Create
src/pages/About.js import React from "react";
import aboutImage from "../assets/about.jpg"; const
About = () => {
return (
<div>
<h2>About Us</h2>
<p>We are a team dedicated to building web applications.</p>
<img src={aboutImage} alt="About Us" width="300" />
</div>
);
};
export default About;
```

**Create src/pages/Contact.js** import React,  
{ useState } from "react"; const Contact = ()



## DEPARTMENT OF INFORMATION TECHNOLOGY

```
=> {  const [message, setMessage] = useState("");  const
handleSubmit = (e) => {
e.preventDefault();  alert("Message sent: "
+ message);  setMessage("");
};
```

**COURSE CODE: DJS22ITL604**

**DATE: COURSE**

**NAME: Full Stack Web Development Laboratory**

**CLASS: TYBTech** return (

<div>

<h2>Contact Us</h2>

<form onSubmit={handleSubmit}>
<textarea value={message} onChange={(e) => setMessage(e.target.value)} placeholder="Enter your message" rows="4" cols="30"></textarea>

<br />

<button type="submit">Send Message</button>

</form>

</div>

); };

export default Contact;



### **Step 5: Implementing a 404 Page (Not Found)**

```

Create src/pages/NotFound.js import React, {
useEffect } from "react"; import { useNavigate } from
"react-router-dom"; const NotFound = () => { const
navigate =
useNavigate(); useEffect(() => { setTimeout(() => {
navigate("/"); }, 5000);
}, [navigate]); return (
<div style={{ textAlign: "center" }}>
<h2> ✗ 404 - Page Not Found</h2>
<p>You will be redirected to Home in 5 seconds...</p>
<button onClick={() => navigate("/")}>Go Back Home Now</button> </div>
);
};export default NotFound;

```

**COURSE NAME: Full Stack Web Development Laboratory**

**CLASS: TYBTech**

### **TESTING THE APPLICATION**

1. Start the project:

**COURSE CODE: DJS22ITL604****DATE:** npm start

2. Open <http://localhost:3000/> in your browser.
3. Navigate between Home, About, and Contact using the Navbar.
4. Test the 404 Page:
  - Enter a random URL like <http://localhost:3000/invalid>.
  - Wait 5 seconds or click 'Go Back Home Now' to be redirected.

```

App.js import {
  BrowserRouter as Router,
  Routes,
  Route
} from 'react-router-dom'
import {
  ToastContainer
} from 'react-toastify'
import 'react-toastify/dist/ReactToastify.css'
import Header from './components/Header'
import PrivateRoute from './components/PrivateRoute'
import Home from './pages/Home'
import Login from './pages/Login'
import Register from './pages/Register'
import NewTicket from './pages/NewTicket'
import Tickets from './pages/Tickets'
import Ticket from './pages/Ticket'

```

// NOTE: Here we have removed the nested routing as the path is the same

```
function App() {
```

**COURSE CODE: DJS22ITL604****DATE:** return (



SHRI VILEPARLE KELAVANI MANDAL'S  
**DWARKADAS J. SANGHVI COLLEGE OF ENGINEERING**  
(Autonomous College Affiliated to the University of Mumbai)  
NAAC ACCREDITED with "A" GRADE (CGPA : 3.18)



<>

<Router>



## DEPARTMENT OF INFORMATION TECHNOLOGY

```
<div className='container'>

<Header />

<Routes>

<Route path='/' element={<Home />} />

<Route path='/login' element={<Login />} />

<Route path='/register' element={<Register />} />

<Route      path='/newticket'
element={

<PrivateRoute>

<NewTicket />

</PrivateRoute>

}

/>
```



SHRI VILEPARLE KELAVANI MANDAL'S  
**DWARKADAS J. SANGHVI COLLEGE OF ENGINEERING**  
(Autonomous College Affiliated to the University of Mumbai)  
NAAC ACCREDITED with "A" GRADE (CGPA : 3.18)



```
<Route      path='/tickets'  
element={
```



<PrivateRoute>

**COURSE CODE: DJS22ITL604**

**DATE:**

<Tickets />

**DEPARTMENT OF INFORMATION TECHNOLOGY**

</PrivateRoute>

}

/>

<Route

path='/ticket/:ticketId'

element={

<PrivateRoute>

<Ticket />

</PrivateRoute>

}

/>

</Routes>



&lt;/div&gt;

&lt;/Router&gt;

&lt;ToastContainer /&gt;

&lt;/&gt;

)

{

## DEPARTMENT OF INFORMATION TECHNOLOGY

export default App

**COURSE CODE: DJS22ITL604****DATE:**

```
Regiter.jsx import { useState } from 'react' import {  
useNavigate } from 'react-router-dom' import { toast }  
from 'react-toastify' import { FaUser } from  
'reacticons/fa' import { useSelector, useDispatch }  
from 'react-redux' import { register } from  
'./features/auth/authSlice' import Spinner from  
'./components/Spinner'
```



```
function Register() { const [formData,  
setFormData] = useState({  
  
    name: "",  
    email: "",  
    password: "",  
    password2: "",  
  
})  
  
const { name, email, password, password2 } = formData
```

#### **DEPARTMENT OF INFORMATION TECHNOLOGY**

```
const dispatch = useDispatch()  
const navigate = useNavigate()  
COURSE CODE:
```

**DJS22ITL604**

**DATE:**



```
const { isLoading } = useSelector((state) => state.auth)
```

```
const onChange = (e) => {
```

```
  setFormData((prevState) => ({
```

```
    ...prevState,
```

```
    [e.target.name]: e.target.value,
```

```
  }))
```

```
}
```

```
// NOTE: no need for useEffect here as we can catch the
```

```
// AsyncThunkAction rejection in our onSubmit or redirect them on the
```

```
// resolution
```

```
// Side effects should go in event handlers where possible
```

#### **DEPARTMENT OF INFORMATION TECHNOLOGY**

```
// source: - https://beta.reactjs.org/learn/keeping-components-pure#where-you-can-causesideeffects
```



```
const onSubmit = (e) => {
```

```
    e.preventDefault()
```

```
    if (password !== password2) {
```

```
        toast.error('Passwords do not match') COURSE
```

**CODE: DJS22ITL604**

**DATE:**

```
} else {     const
```

```
userData = {     name,
```

```
email,     password,
```

```
}
```

```
dispatch(register(userData))
```

```
.unwrap()
```

```
.then((user) => {
```

// NOTE: by unwrapping the AsyncThunkAction we can navigate the user after

**DEPARTMENT OF INFORMATION TECHNOLOGY**



```
// getting a good response from our API or catch the AsyncThunkAction

// rejection to show an error message

toast.success(`Registered new user - ${user.name}`)

navigate('/')

})

.catch(toast.error)

}

}

} if (isLoading)

{

}

COURSE CODE: DJS22ITL604           DATE:      return <Spinner>
/>

}

return (

<>
```



<section className='heading'>

<h1>

<FaUser /> Register

</h1>

## **DEPARTMENT OF INFORMATION TECHNOLOGY**

<p>Please create an account</p>

</section>

<section className='form'>

<form onSubmit={onSubmit}>

<div className='form-group'>

<input type='text'

className='form-

control'

id='name'

name='name'



value={name} **COURSE CODE:**

**DJS22ITL604**

**DATE:**

onChange={onChange}

placeholder='Enter your name'

required

/>

### **DEPARTMENT OF INFORMATION TECHNOLOGY**

</div>

<div className='form-group'>

<input type='email'

className='form-control' id='email'

name='email' value={email}

onChange={onChange}

placeholder='Enter your email'

required



/>

</div>

<div className='form-group'>

    <input type='password'

        className='formcontrol'

**COURSE CODE:**

**DJS22ITL604**

**DATE:**

**DEPARTMENT OF INFORMATION TECHNOLOGY**

    id='password'

    name='password'

    value={password}

    onChange={onChang

    e}

    placeholder='Enter

    password'

    required

/>



&lt;/div&gt;

&lt;div className='form-group'&gt;

&lt;input type='password'

className='form-control'

id='password2'

name='password2'

value={password2}

onChange={onChange}

placeholder='Confirm password'

required

/&gt;

&lt;/div&gt;

### DEPARTMENT OF INFORMATION TECHNOLOGY

&lt;div className='form-group'&gt;

**COURSE CODE: DJS22ITL604****DATE:**

&lt;button className='btn btn-block'&gt;Submit&lt;/button&gt;

&lt;/div&gt;



</form>

</section>

</>

)

}

export default Register

```
 Login.jsx import { useState } from 'react' import { toast
} from 'react-toastify' import { useNavigate } from
'react-router-dom' import { FaSignInAlt } from 'react-
icons/fa' import { useSelector, useDispatch } from
'react-redux' import { login } from
'../features/auth/authSlice' import Spinner from
'../components/Spinner'
```



```
function Login() { const [formData,  
setFormData] = useState({ COURSE
```

**CODE: DJS22ITL604**

**DATE:**

```
email: ",  
password: ",  
})
```

```
const { email, password } = formData
```

```
const dispatch = useDispatch()  
const navigate = useNavigate()
```

```
const { isLoading } = useSelector((state) => state.auth)
```

```
const onChange = (e) => {  
  setFormData((prevState) => ({ ...prevState,
```



[e.target.name]: e.target.value,

### DEPARTMENT OF INFORMATION TECHNOLOGY

}))

}

// NOTE: no need for useEffect here as we can catch the

// AsyncThunkAction rejection in our onSubmit or redirect them on the

// resolution

**COURSE CODE: DJS22ITL604**

**DATE:**

// Side effects shoulld go in event handlers where possible

// source: - <https://beta.reactjs.org/learn/keeping-components-pure#where-you-can-causesideeffects>

```
const onSubmit = (e) => {
```

```
  e.preventDefault()
```

```
  const userData = {
```

```
    email,      password,
```



{}

```
dispatch(login(userData))
```

```
.unwrap()
```

### **DEPARTMENT OF INFORMATION TECHNOLOGY**

```
.then((user) => {
```

```
// NOTE: by unwrapping the AsyncThunkAction we can navigate the user after
```

```
// getting a good response from our API or catch the AsyncThunkAction
```

```
// rejection to show an error message
```

```
toast.success(`Logged in as ${user.name}`)
```

```
navigate('/')
```

```
)
```

```
.catch(toast.error)
```

**COURSE CODE: DJS22ITL604**

**DATE:**

```
}
```



```
if (isLoading) {
```

```
    return <Spinner />
```

```
}
```

```
return (
```

```
<>
```

```
    <section className='heading'>
```

```
        <h1>
```

```
            <FaSignInAlt /> Login
```

## **DEPARTMENT OF INFORMATION TECHNOLOGY**

```
    </h1>
```

```
    <p>Please log in to get support</p>
```

```
    </section>
```

```
    <section className='form'>
```

```
        <form onSubmit={onSubmit}>
```



```
<div className='form-group'>
```

```
    <input
```

```
        type='email'
```

```
        className='form-control'
```

**COURSE CODE:**

**DJS22ITL604**

**DATE:**

```
        id='email'
```

```
        name='email'          value={email}
```

```
        onChange={onChange}
```

```
        placeholder='Enter your email'
```

```
        required
```

```
    />
```

**DEPARTMENT OF INFORMATION TECHNOLOGY**

```
</div>
```

```
<div className='form-group'>
```



```
<input
    type='password'
    className='form-control'
    id='password'
    name='password'
    value={password}
    onChange={onChange}
    placeholder='Enter password'
    required
/>
</div>
<div className='form-group'>
    COURSE CODE: DJS22ITL604
    DATE:
    <button className='btn btn-block'>Submit</button>
</div>
</form>
</section>
</>
```



## DEPARTMENT OF INFORMATION TECHNOLOGY

)

}

export default Login

```
NewTicket.jsx import { useState } from 'react' import {  
useSelector, useDispatch } from 'react-redux' import {  
useNavigate } from 'react-router-dom' import { toast } from  
'react-toastify' import { createTicket } from  
'./features/tickets/ticketSlice' import BackButton from  
'./components/BackButton'
```

```
function NewTicket() { const { user } =
```

```
useSelector((state) => state.auth) COURSE
```

**CODE: DJS22ITL604**

**DATE:** const [name] =



```
useState(user.name) const [email] = useState(user.email) const  
[product, setProduct]
```

### DEPARTMENT OF INFORMATION TECHNOLOGY

```
= useState('iPhone') const [description, setDescription]  
= useState("")
```

```
const dispatch = useDispatch()  
const navigate = useNavigate()
```

```
const onSubmit = (e) => {  
  e.preventDefault() dispatch(createTicket({  
    product, description }))  
  .unwrap()  
  .then(() => {
```



```
// We got a good response so navigate the user
navigate('/tickets')      toast.success('New ticket
created!')
})
.catch(toast.error)
}

return (

```

## DEPARTMENT OF INFORMATION TECHNOLOGY

**COURSE CODE: DJS22ITL604**

**DATE:** ◇

```
<BackButton />

<section className='heading'>
  <h1>Create New Ticket</h1>
  <p>Please fill out the form below</p>
</section>
```



```
<section className='form'>

<div className='form-group'>

<label htmlFor='name'>Customer Name</label>

<input type='text' className='form-control' value={name} disabled />

</div>

<div className='form-group'>

<label htmlFor='email'>Customer Email</label>

<input type='text' className='form-control' value={email} disabled />

</div>

<form onSubmit={onSubmit}>
```

#### **DEPARTMENT OF INFORMATION TECHNOLOGY**

```
<select
  name='product' COURSE

CODE: DJS22ITL604
```



**DATE:**

```
id='product'          value={product}

onChange={(e) => setProduct(e.target.value)}

>

<option value='iPhone'>iPhone</option>

<option value='Macbook Pro'>Macbook Pro</option>

<option value='iMac'>iMac</option>

<option value='iPad'>iPad</option>

</select>

</div>

<div className='form-group'>

<label htmlFor='description'>Description of the issue</label>

<textarea          name='description'

id='description'      className='form-control'>
```



## DEPARTMENT OF INFORMATION TECHNOLOGY

placeholder='Description' value={description}

onChange={(e) => setDescription(e.target.value)}

></textarea>

</div>

**COURSE CODE: DJS22ITL604**

**DATE:**

<div className='form-group'>

<button className='btn btn-block'>Submit</button>

</div>

</form>

</section>

</>

)

}

export default NewTicket



```
Ticket.jsx import { useEffect, useState } from 'react' import { toast
} from 'react-toastify' import Modal from 'react-modal' import {
FaPlus } from 'react-icons/fa' import { useSelector, useDispatch } from
'react-redux' import { getTicket, closeTicket } from
```

#### **DEPARTMENT OF INFORMATION TECHNOLOGY**

```
'../features/tickets/ticketSlice' import { getNotes, createNote } from
'../features/notes/noteSlice' import { useParams, useNavigate } from
'react-router-dom' import BackButton from
'../components/BackButton'
```

**COURSE CODE: DJS22ITL604**

**DATE:** import

```
Spinner from '../components/Spinner' import NoteItem from
'../components>NoteItem'
```

```
const customStyles = { content: {
width: '600px', top: '50%', left: '50%', right: 'auto', bottom: 'auto',
```



marginRight: '-50%', transform:

'translate(-50%, -50%)', position:

'relative',

},

}

Modal.setAppElement('#root')

### **DEPARTMENT OF INFORMATION TECHNOLOGY**

```
function Ticket() { const [modalIsOpen,
  setModalIsOpen] = useState(false) const [noteText,
  setNoteText] = useState("")
```

**COURSE CODE: DJS22ITL604**

**DATE:**

```
const { ticket } = useSelector((state) => state.tickets)
```

```
const { notes } = useSelector((state) => state.notes)
```



```
// NOTE: no need for two useParams
```

```
// const params = useParams() const
navigate = useNavigate() const dispatch =
= useDispatch() const { ticketId } =
useParams()
```

```
useEffect(() => {
dispatch(getTicket(ticketId)).unwrap().catch(toast.error)
dispatch(getNotes(ticketId)).unwrap().catch(toast.error)
}, [ticketId, dispatch])
```

#### **DEPARTMENT OF INFORMATION TECHNOLOGY**

```
// Close ticket const onTicketClose
```

```
= () => {
// NOTE: we can unwrap our AsyncThunkAction here so no need for isError and
// isSuccess state
dispatch(closeTicket(ticketId))
.unwrap()
```



**COURSE CODE: DJS22ITL604**

**DATE:**

```
.then(() => {
```

```
toast.success('Ticket Closed')
```

```
navigate('/tickets')
```

```
)})
```

```
.catch(toast.error)
```

```
}
```

```
// Create note submit const onNoteSubmit
```

```
= (e) => {
```

```
// NOTE: we can unwrap our AsyncThunkAction here so no need for isError and
```

```
// isSuccess state
```

```
e.preventDefault() dispatch(createNote({
```

```
noteText, ticketId }))
```

**DEPARTMENT OF INFORMATION TECHNOLOGY**



```
.unwrap()

.then(() => {

    setNoteText("")

    closeModal()

})

.catch(toast.error)

}

COURSE CODE: DJS22ITL604 DATE:
// Open/close modal const openModal = () =>

setModalisOpen(true) const closeModal = () =>

setModalisOpen(false)

if (!ticket)

{

return <Spinner />

}

return (

```



<div className='ticket-page'>

<header className='ticket-header'>

<BackButton />

## **DEPARTMENT OF INFORMATION TECHNOLOGY**

<h2>

Ticket ID: {ticket.\_id}

<span className={`status status-\$ {ticket.status}`}>

{ticket.status}

</span>

</h2>

<h3>

Date Submitted: {new Date(ticket.createdAt).toLocaleString('en-US')}

</h3>

**COURSE CODE: DJS22ITL604**

**DATE:**

<h3>Product: {ticket.product}</h3>

<hr />



```
<div className='ticket-desc'>  
  <h3>Description of Issue</h3>  
  <p>{ticket.description}</p>  
</div>
```

```
<h2>Notes</h2>
```

```
</header>
```

```
{ticket.status !== 'closed' && (
```

### **DEPARTMENT OF INFORMATION TECHNOLOGY**

```
  <button onClick={openModal} className='btn'>
```

```
    <FaPlus /> Add Note
```

```
  </button>
```

```
)}
```

```
<Modal
```



isOpen={modalIsOpen}

onRequestClose={closeModal}

style={customStyles} contentLabel='Add

Note'

>

**COURSE CODE: DJS22ITL604**

**DATE:**

<h2>Add Note</h2>

<button className='btn-close' onClick={closeModal}>

X

</button>

<form onSubmit={onNoteSubmit}>

<div className='form-group'>

<textarea name='noteText'

id='noteText' className='form-control'

**DEPARTMENT OF INFORMATION TECHNOLOGY**



placeholder='Note text' value={noteText}

onChange={(e) => setNoteText(e.target.value)}

></textarea>

</div>

<div className='form-group'>

<button className='btn' type='submit'>

Submit

</button>

</div>

</form>

**COURSE CODE: DJS22ITL604**

**DATE:**

</Modal>

{notes ? ( notes.map((note) => <NoteItem key={note.\_id}

note={note} />)

) : (

<Spinner />



)}

{ticket.status !== 'closed' &amp;&amp; (

#### DEPARTMENT OF INFORMATION TECHNOLOGY

&lt;button onClick={onTicketClose} className='btn btn-block btn-danger'&gt;

Close Ticket

&lt;/button&gt;

)}

&lt;/div&gt;

)

}

export default Ticket

Tickets.jsx import { useEffect };

from 'react' **COURSE CODE:****DJS22ITL604**

**DATE:**

```
import { useSelector, useDispatch } from 'react-redux' import
{ getTickets } from '../features/tickets/ticketSlice' import
Spinner from '../components/Spinner' import
```

**DEPARTMENT OF INFORMATION TECHNOLOGY**

```
BackButton from '../components/BackButton' import
TicketItem from '../components/TicketItem'
```

```
function Tickets() { const { tickets } =
useSelector((state) => state.tickets)
const dispatch = useDispatch()
// NOTE: only need one useEffect here
```



```
useEffect(() => {  
  dispatch(getTickets())  }, [dispatch])  
  
// NOTE: no need for loading state, we can check for absence of tickets  
// If we don't have tickets we are loading, if we do have tickets we just  
// need to update the tickets with latest tickets in the background if  
(!tickets) {
```

#### **DEPARTMENT OF INFORMATION TECHNOLOGY**

**COURSE CODE: DJS22ITL604** **DATE:** return <Spinner>/>

{}

return (

&lt;&gt;

&lt;BackButton /&gt;

&lt;h1&gt;Tickets&lt;/h1&gt;

&lt;div className='tickets'&gt;



```
<div className='ticket-headings'>
```

```
    <div>Date</div>
```

```
    <div>Product</div>
```

```
    <div>Status</div>
```

```
    <div></div>
```

```
</div>
```

```
{tickets.map((ticket) => (
```

```
    <TicketItem key={ticket._id} ticket={ticket} />
```

```
))}
```

```
</div>
```

```
</>
```

### DEPARTMENT OF INFORMATION TECHNOLOGY

```
)
```

```
}
```

**COURSE CODE: DJS22ITL604**

**DATE:**



export default Tickets

```
Home.jsx import { Link } from 'react-router-dom' import {
```

```
FaQuestionCircle, FaTicketAlt } from 'react-icons/fa'
```

```
function Home() {
```

```
    return (
```

```
    <>
```

```
        <section className='heading'>
```

```
            <h1>What do you need help with?</h1>
```

```
            <p>Please choose from an option below</p>
```

```
        </section>
```

```
        <Link to='/new-ticket' className='btn btn-reverse btn-block'>
```

```
            <FaQuestionCircle /> Create New Ticket
```

```
</Link>
```



## DEPARTMENT OF INFORMATION TECHNOLOGY

<Link to='/tickets' className='btn btn-block'>

<FaTicketAlt /> View My Tickets

## DEPARTMENT OF INFORMATION TECHNOLOGY

**COURSE CODE: DJS22ITL604**

**DATE:**

</Link>

</>

)

}

export default Home

[Support Desk](#) [Login](#)  [Register](#)

# What do you need help with?

Please choose from an option below

 [Create New Ticket](#) [View My Tickets](#)

**DEPARTMENT OF INFORMATION TECHNOLOGY**

**COURSE CODE: DJS22ITL604**

**DATE:**



Support Desk

[Login](#) [Register](#)

## ➡ Login

Please log in to get support

Support Desk

[Login](#) [Register](#)

## 👤 Register

Please create an account

**DEPARTMENT OF INFORMATION TECHNOLOGY**



COURSE CODE: DJS22ITL604

DATE:

Support Desk

[Logout](#)

# What do you need help with?

Please choose from an option below

 [Create New Ticket](#) [View My Tickets](#)

DEPARTMENT OF INFORMATION TECHNOLOGY

COURSE CODE: DJS22ITL604

DATE:

[Back](#)

## Create New Ticket

Please fill out the form below

Customer Name

talok shah

Customer Email

talok2001@gmail.com

Product

iPhone

Description of the issue

Description

**Submit**

**DEPARTMENT OF INFORMATION TECHNOLOGY**

**COURSE CODE: DJS22ITL604**

**DATE:**



Support Desk

[Logout](#)
[Back](#)

## Tickets

Date	Product	Status	
3/20/2025, 2:53:56 PM	Macbook Pro	new	<a href="#">View</a>

Support Desk

[Logout](#)
[Back](#)
**Ticket ID: 67dbdeac3715c4c6f0ed3955**
new
**Date Submitted: 3/20/2025, 2:53:56 PM**
**Product: Macbook Pro**
**Description of Issue**

display not working

**Notes**
[+ Add Note](#)
[Close Ticket](#)



## DEPARTMENT OF INFORMATION TECHNOLOGY

COURSE CODE: DJS22ITL604

DATE:

Support Desk

Logout Ticket Closed

[Back](#)

### Tickets

Date	Product	Status	Action
3/20/2023, 2:53:58 PM	Macbook Pro	Closed	<a href="#">View</a>

Conclusion: We learnt react router and navigations.

### BOOKS AND WEB RESOURCES:

1. React Router Official Docs: <https://reactrouter.com/>
2. MDN Web Docs - SPA Routing: [https://developer.mozilla.org/en-US/docs/Web/API/History\\_API](https://developer.mozilla.org/en-US/docs/Web/API/History_API)
3. React Official Docs: <https://react.dev/>