Hooks de navigation

Hooks React Router v6

useParams — paramètres d'URL

```
// Route : /users/:id
function UserDetail() {
  const { id } = useParams();
  // id = "42" pour /users/42
  return <p>Utilisateur #{id}</p>;
}
```

useNavigate — navigation programmée

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

function Login() {
  const navigate = useNavigate();

  const handleSubmit = async () => {
    await loginUser();
    navigate('/dashboard');           // Aller à
    navigate(-1);                     // Page précédente
    navigate('/profile', { replace: true }); // Remplacer l'historique
  };
}
```

useLocation — URL courante

```
import { useLocation } from 'react-router-dom';

function Breadcrumb() {
```

```
  const location = useLocation();
  // location.pathname = "/dashboard/stats"
  // location.search = "?page=2"
  // location.state = { from: "/login" }
  return <p>Vous êtes ici : {location.pathname}</p>;
}
```

useSearchParams — query string

```
import { useSearchParams } from 'react-router-dom';

function Recherche() {
  const [searchParams, setSearchParams] = useSearchParams();
  const query = searchParams.get('q') || '';

  return (
    <input
      value={query}
      onChange={(e) => setSearchParams({ q: e.target.value })}
      placeholder="Rechercher..."
    />
  );
  // URL : /recherche?q=react
}
```

Route protégée

```
function PrivateRoute({ children }) {
  const { isAuthenticated } = useAuth();
  const location = useLocation();

  if (!isAuthenticated) {
    return <Navigate to="/login" state={{ from: location }} replace />;
  }

  return children;
}

// Utilisation
```

```
<Route path="/dashboard" element={
  <PrivateRoute><Dashboard /></PrivateRoute>
} />
```

Revision #1
Created 29 March 2026 12:57:38 by Hugo
Updated 29 March 2026 12:57:38 by Hugo