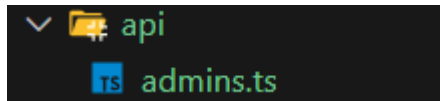


Nama : Siti Aspita Fitri  
NPM : 22312046  
Commit : pita : crud admins

Membuat file admins.ts di dalam folder api yang sudah dibuat sebagai controller untuk memproses semua CRUD dari admins.



Lalu isikan file admins.ts dengan sebagai berikut.

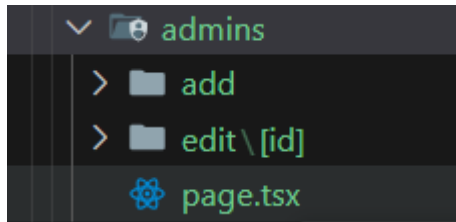
```
1 'use server';
2
3 import { PrismaClient } from '@prisma/client';
4 const bcrypt = require('bcryptjs');
5
6 const prisma = new PrismaClient();
7
8 export async function createAdmin(name: string, email: string, password: string) {
9   return prisma.admin.create({
10     data: {
11       name,
12       email,
13       password: await bcrypt.hashSync(password, 8)
14     }
15   });
16 }
17
18 export async function getAllAdmins() {
19   return prisma.admin.findMany({
20     orderBy: { id: 'desc' },
21   })
22 }
23
24 export async function getAdminById(id: number) {
25   return prisma.admin.findUnique({
26     where: { id },
27   })
28 }
29
30 export async function updateAdmin(id: number, name: string, email: string, password: string) {
31   const data: { name: string, email: string, password?: string } = {
32     name,
33     email
34   };
35
36   if (password) {
37     data.password = await bcrypt.hashSync(password, 8);
38   }
39
40   return prisma.admin.update({
41     where: { id },
42     data
43   });
44 }
45
46 export async function deleteAdmin(id: number) {
47   return prisma.admin.delete({
48     where: { id },
49   });
50 }
```

Kode diatas tersebut saya membuat 5 fungsi untuk melakukan CRUD. createAdmins() untuk membuat data Admins baru, getAllAdmins() untuk mengambil semua data admins. getAdminsById() untuk mengambil admins berdasarkan id, updateAdmins() untuk handle mengubah data Admins , dan deleteAdmins() untuk menghapus data Admins.

Instal juga library untuk mengenskripsi password

```
npm install bcryptjs
```

Membuat folder Admins di dalam folder panel untuk membuat route/admins. Di dalamnya buat juga file page.tsx untuk membuat tampilan halamannya.



Membuat isi tampilan Admins

```
1 'use client';
2
3 import React, { useEffect, useState } from "react";
4 import { deleteAdmin, getAllAdmins } from "../api/admins";
5 import Link from "next/link";
6 import { useRouter } from 'next/navigation';
7 import {
8   Breadcrumb,
9   BreadcrumbItem,
10  Button,
11  Table,
12  TableBody,
13  TableCell,
14  TableHead,
15  TableHeadCell,
16  TableRow,
17 } from "flowbite-react";
18 import { HiHome, HiOutlinePencilAlt, HiOutlineTrash, HiPlus } from "react-icons/hi";
```

Menyiapkan kebutuhan pengambilan dan wadah data.

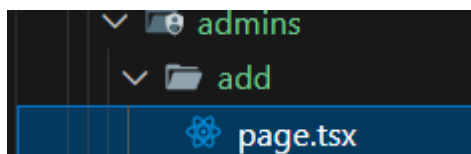
```
1 export default function Admins() {
2   const [getAdmins, setAdmins] = useState({});
3   const router = useRouter();
4
5   async function fetchData() {
6     setAdmins(await getAllAdmins());
7   }
8
9   useEffect(() => {
10    fetchData();
11  }, []);
12
13  const handleDelete = async (id: number) => {
14    if (confirm('Are you sure you want to delete this admin?')) {
15      await deleteAdmin(id);
16      fetchData();
17    }
18  }
19 }
```

Buat tampilan table untuk pengelolaannya juga tombol action deletenya.

```
1 return (  
2   <div>  
3     <Breadcrumb aria-label="Default breadcrumb example" className="mt-3 mb-5">  
4       <BreadcrumbItem href="/panel" icon={HiHome}>  
5         <Link href="/panel">  
6           Dashboard  
7         </Link>  
8       </BreadcrumbItem>  
9       <BreadcrumbItem>Admins</BreadcrumbItem>  
10    </Breadcrumb>  
11    <div className="flex justify-between items-center mb-4">  
12      <h2 className="text-xl font-medium">Admins</h2>  
13      <Button as={Link} href="/panel/admins/add" color="blue">  
14        <HiPlus className="mr-2 h-5 w-5" />  
15        Add Admin  
16      </Button>  
17    </div>  
18    <div className="shadow p-5 rounded">  
19      <div className="overflow-x-auto">  
20        <Table>  
21          <TableHead>  
22            <TableHeadCell>#</TableHeadCell>  
23            <TableHeadCell>Full Name</TableHeadCell>  
24            <TableHeadCell>Email</TableHeadCell>  
25            <TableHeadCell className="text-center">Action</TableHeadCell>  
26          </TableHead>  
27          <TableBody className="divide-y">  
28            {Object.values(getAdmins).map((admin: any, index: number) => (  
29              <TableRow key={admin.id} className="bg-white dark:border-gray-700 dark:bg-gray-800">  
30                <TableCell>{index + 1}</TableCell>  
31                <TableCell className="whitespace-nowrap font-medium text-gray-900 dark:text-white">  
32                  {admin.name}  
33                </TableCell>  
34                <TableCell>{admin.email}</TableCell>  
35                <TableCell>  
36                  <div className="flex gap-1 justify-center">  
37                    <Button as={Link} href={`/panel/admins/edit/${admin.id}`} color="blue" size="xs">  
38                      <HiOutlinePencilAlt className="h-5 w-5" />  
39                    </Button>  
40                    <Button color="failure" size="xs" onClick={() => handleDelete(admin.id)}>  
41                      <HiOutlineTrash className="h-5 w-5" />  
42                    </Button>  
43                  </div>  
44                </TableCell>  
45              </TableRow>  
46            </TableBody>  
47          </Table>  
48        </div>  
49      </div>  
50    </div>  
51  )  
52  </div>  
53  );
```

## Tambah Admins

Buat folder add di dalam folder Admins, lalu buat juga page.tsx nya

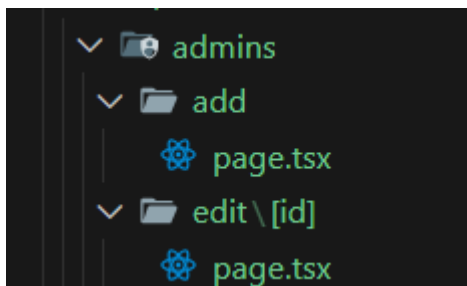


Untuk tampilannya yaitu :

```
1 'use client';
2
3 import { createAdmin } from '../api/admins';
4 import React, { useState } from 'react';
5 import { useRouter } from 'next/navigation';
6 import Link from 'next/link';
7 import {
8   Breadcrumb,
9   BreadcrumbItem,
10  Button,
11  Label,
12  TextInput,
13 } from 'flowbite-react';
14 import { HiHome, HiArrowDownLeft, HiOutlineSave } from 'react-icons/hi';
15
16 export default function AddAdmin() {
17   const [name, setName] = useState('');
18   const [email, setEmail] = useState('');
19   const [password, setPassword] = useState('');
20   const [confirmPassword, setConfirmPassword] = useState('');
21   const router = useRouter();
22
23   const handleSubmit = async (e: React.FormEvent) => {
24     e.preventDefault();
25
26     if (password !== confirmPassword) {
27       alert('Passwords do not match');
28       return;
29     } else {
30       await createAdmin(name, email, password);
31       router.push('/panel/admins');
32     }
33   }
34
35   return (
36     <div>
37       <Breadcrumb aria-label="Default breadcrumb example" className="mt-3 mb-5">
38         <BreadcrumbItem href="/panel/" icon={HiHome}>
39           <Link href="/panel/">
40             Dashboard
41           </Link>
42         </BreadcrumbItem>
43         <BreadcrumbItem href="/panel/admins">
44           <Link href="/panel/admins">
45             Admins
46           </Link>
47         </BreadcrumbItem>
48       </Breadcrumb>
49       <div className="flex justify-between items-center mb-4">
50         <div className="text-xl font-medium">Add Admin</div>
51         <Button as={Link} href="/panel/admins" color="blue">
52           <HiArrowDownLeft className="mr-2 h-5 w-5"/>
53           Back
54         </Button>
55       </div>
56       <div className="shadow p-5 rounded">
57         <Form onSubmit={handleSubmit}>
58           <div className="grid grid-cols-2 gap-4">
59             <div className="mb-3">
60               <div className="mb-2 block">
61                 <Label value="Full Name" />
62               </div>
63               <TextInput type="text" value={name} onChange={(e) => setName(e.target.value)} required />
64             </div>
65             <div className="mb-3">
66               <div className="mb-2 block">
67                 <Label value="Email" />
68               </div>
69               <TextInput type="email" value={email} onChange={(e) => setEmail(e.target.value)} required />
70             </div>
71             <div>
72               <div className="mb-3">
73                 <div className="mb-2 block">
74                   <Label value="Password" />
75                 </div>
76                 <TextInput type="password" value={password} onChange={(e) => setPassword(e.target.value)} required />
77               </div>
78               <div>
79                 <div className="mb-3">
80                   <div className="mb-2 block">
81                     <Label value="Confirm Password" />
82                   </div>
83                   <TextInput type="password" value={confirmPassword} onChange={(e) => setConfirmPassword(e.target.value)} required />
84                 </div>
85               </div>
86             <div>
87               <Button type="submit" color="blue">
88                 <HiOutlineSave className="mr-2 h-5 w-5"/>
89                 Save
90               </Button>
91             </div>
92           </div>
93         </Form>
94       </div>
95     </div>
96   );
97 }
```

## Edit Admins

Buat folder edit di dalam folder Admins, lalu folder [id] untuk menampung id Admins nya, dan file page.tsx nya.



Buat isi halamannya seperti berikut. Ambil data Admins berdasarkan id menggunakan fungsi `getAdminsById()`.

```
1 'use client';
2
3 import { getAdminById, updateAdmin } from "../../api/admins";
4 import Link from "next/link";
5 import { useParams, useRouter } from "next/navigation";
6 import React, { useEffect, useState } from "react";
7 import {
8   Breadcrumb,
9   BreadcrumbItem,
10  Button,
11  Label,
12  TextInput,
13 } from "flowbite-react";
14 import { HiHome, HiArrowLeft, HiOutlineSave } from "react-icons/hi";
15
16 export default function EditAdmin() {
17   const { id } = useParams();
18   const router = useRouter();
19   const [name, setName] = useState('');
20   const [email, setEmail] = useState('');
21   const [password, setPassword] = useState('');
22   const [confirmPassword, setconfirmPassword] = useState('');
23
24   const fetchData = async () => {
25     const admin = await getAdminById(Number(id));
26     if (admin) {
27       setName(admin.name);
28       setEmail(admin.email);
29     }
30   };
31
32   useEffect(() => {
33     fetchData();
34   }, [id]);
35
36   const handleSubmit = async (e: React.FormEvent) => {
37     e.preventDefault();
38
39     if (password !== confirmPassword) {
40       alert("Passwords do not match");
41       return;
42     } else {
43       await updateAdmin(Number(id), name, email, password);
44       router.push("/panel/admins");
45     }
46   };
47
48   return (
49     <div>
50       <Breadcrumb aria-label="Default breadcrumb example" className="mt-3 mb-5">
51         <BreadcrumbItem href="/panel" icon={HiHome}>
52           <Link href="/panel">
53             Dashboard
54           </Link>
55         </BreadcrumbItem>
56         <BreadcrumbItem href="/panel/admins">
57           <Link href="/panel/admins">
58             Admins
59           </Link>
60         </BreadcrumbItem>
61         <BreadcrumbItem>Edit Admin</BreadcrumbItem>
62       </Breadcrumb>
63       <div className="flex justify-between items-center mb-4">
64         <h2 className="text-xl font-medium">Edit Admin</h2>
65         <button as={Link} href="/panel/admins" color="blue">
66           <HiArrowLeft className="mr-2 h-5 w-5" />
67           Back
68         </button>
69       </div>
70       <div className="shadow p-5 rounded">
71         <form onSubmit={handleSubmit}>
72           <div className="grid grid-cols-2 gap-4">
73             <div className="mb-3">
74               <div className="mb-2 block">
75                 <Label value="Full Name" />
76               </div>
77               <TextInput type="text" value={name} onChange={(e) => setName(e.target.value)} required />
78             </div>
79             <div className="mb-3">
80               <div className="mb-2 block">
81                 <Label value="Email" />
82               </div>
83               <TextInput type="email" value={email} onChange={(e) => setEmail(e.target.value)} required />
84             </div>
85           </div>
86           <div className="grid grid-cols-2 gap-4">
87             <div className="mb-3">
88               <div className="mb-2 block">
89                 <Label value="Password" />
90               </div>
91               <TextInput type="password" value={password} onChange={(e) => setPassword(e.target.value)} />
92             </div>
93             <div className="mb-3">
94               <div className="mb-2 block">
95                 <Label value="Confirm Password" />
96               </div>
97               <TextInput type="password" value={confirmPassword} onChange={(e) => setconfirmPassword(e.target.value)} />
98             </div>
99           </div>
100          <Button type="submit" color="blue">
101            <HiOutlineSave className="mr-2 h-5 w-5" />
102            Save
103          </Button>
104        </form>
105      </div>
106    </div>
107  );
108 }
```