

Nama : Prayogi Setiawan
NPM : 22312009
Commit : “yogi: add login & fix panel layout”

Install library next-auth untuk membuat sistem autentikasi di project kita.

```
npm install next-auth
```

Tambahkan konfigurasi untuk next auth di file .env.

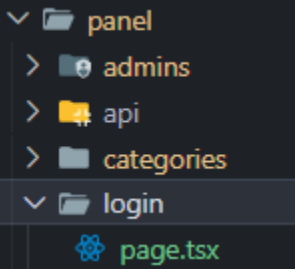
```
1 NEXTAUTH_SECRET="j/1JxRL2WYjTdnSkXKmdfmy5tT5Qaq7AgpFUZ8qBBt8="
2 NEXTAUTH_URL=http://localhost:3000
```

Di dalam folder app, buat folder api/auth/[...nextauth]/route.ts untuk kebutuhan autentikasinya. Isi dari file route.ts adalah sebagai berikut.

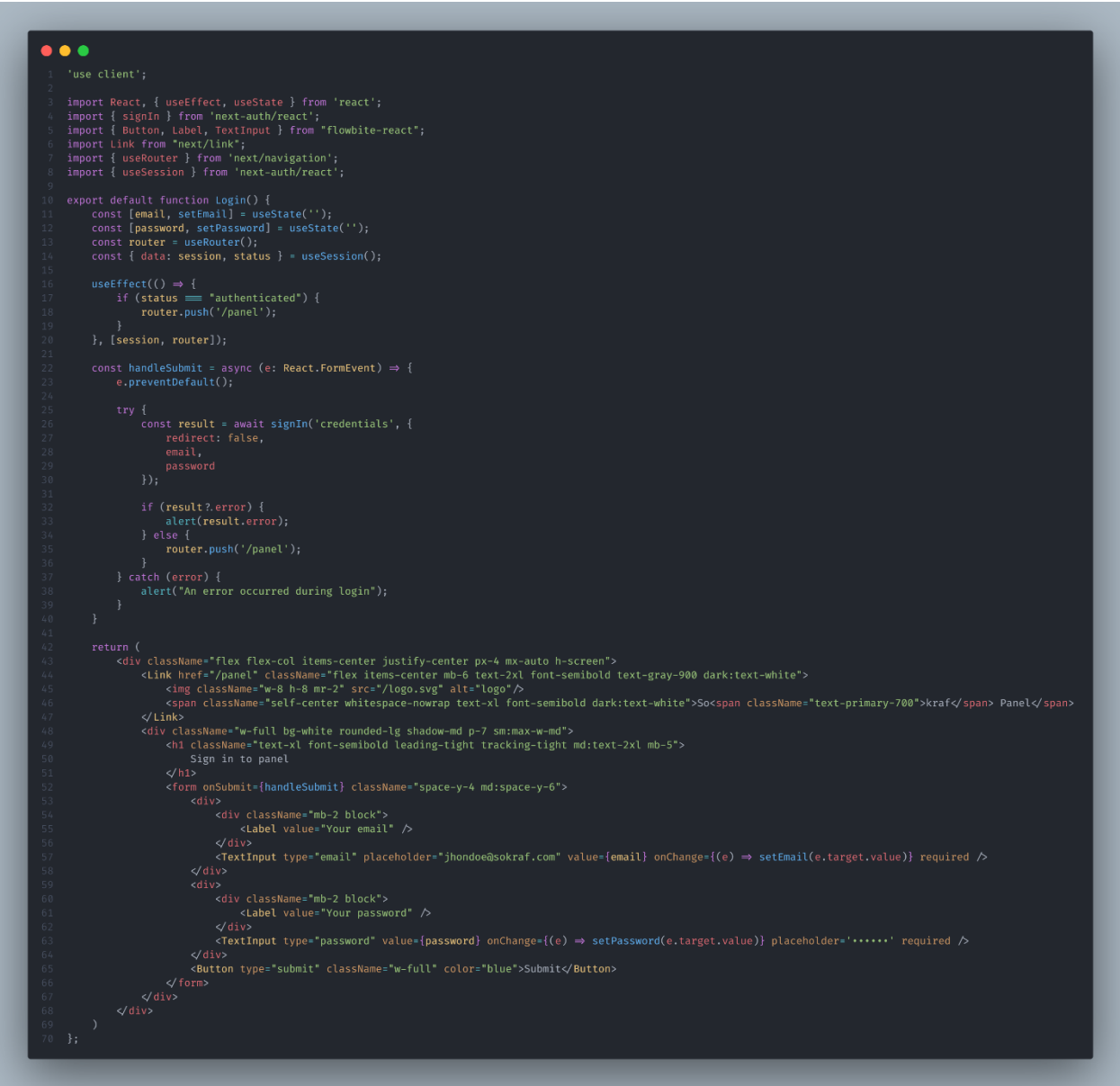
```
1 import NextAuth, { NextAuthOptions } from "next-auth";
2 import CredentialsProvider from "next-auth/providers/credentials";
3 import { PrismaClient } from '@prisma/client';
4 const bcrypt = require("bcryptjs");
5
6 const prisma = new PrismaClient();
7
8 export const authOptions: NextAuthOptions = {
9   secret: process.env.NEXTAUTH_SECRET,
10  session: {
11    strategy: "jwt",
12  },
13  providers: [
14    CredentialsProvider({
15      name: "Credentials",
16      credentials: {
17        email: { label: "Email", type: "email" },
18        password: { label: "Password", type: "password" },
19      },
20      async authorize(credentials) {
21        const { email, password } = credentials as {
22          email: string;
23          password: string;
24        };
25
26        const admin = await prisma.admin.findUnique({ where: { email } });
27        if (!admin) {
28          throw new Error("Invalid email or password!");
29        }
30
31        const isValidPassword = await bcrypt.compare(password, admin.password);
32        if (!isValidPassword) {
33          throw new Error("Invalid email or password!");
34        }
35
36        return {
37          id: admin.id,
38          name: admin.name,
39          email: admin.email
40        };
41      },
42    }),
43  ],
44  callbacks: {
45    async jwt({ token, user }) {
46      return { ...token, ...user };
47    },
48    async session({ session, token, user }) {
49      session.user = token as any;
50      return session;
51    },
52  },
53  pages: {
54    signIn: "/panel/login",
55    error: "/panel/login",
56  }
57 };
58
59 export const handler = NextAuth(authOptions);
60 export { handler as GET, handler as POST };
```

File ini digunakan untuk memproses autentikasi ketika user mengirimkan email dan password pada halaman login. Lalu file ini akan mengautentikasi mencocokkan dengan data di database, ketika berhasil maka akan mengenerate Json Web Token dan menyimpannya ke session.

Buat folder dan file login/page.tsx di dalam folder panel untuk membuat tampilan halaman login.

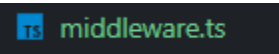


Buat tampilan login seperti ini.



Di sini kita juga membuat pengecekan, jika user sudah melakukan autentikasi atau login, maka tidak bisa masuk lagi ke halaman login. Pengecekan ini memanfaatkan status yang tersimpan di session. Session terbuat setelah user melakukan autentikasi. Jika status adalah “authenticated” artinya user sudah masuk, maka akan di arahkan ke halaman dashboard.

Buat file middleware.ts di root project untuk membatasi halaman agar jika belum melakukan autentikasi, maka user tidak bisa mengakses halaman yang di Batasi.



Isi file middleware.ts adalah sebagai berikut.



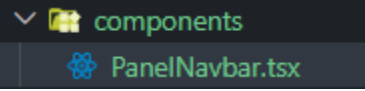
```
1  import { withAuth } from "next-auth/middleware";
2
3  export default withAuth({
4    pages: {
5      signIn: "/panel/login",
6    },
7  });
8
9  export const config = {
10    matcher: [
11      '/panel/api/:path*',
12      '/panel/admins',
13      '/panel/admins/:path*',
14      '/panel/products',
15      '/panel/products/:path*',
16      '/panel/categories',
17      '/panel/categories/:path*',
18      '/panel'
19    ],
20  };
21
```

Kita batasi semua route yang ada di dalam panel, kecuali route /panel/login.

Fix Panel Layout

Karena sekarang user sudah bisa melakukan autentikasi, maka kita akan ubah data yang di tampilkan di navbar panel yang semula adalah data statis.

Memisahkan elemen navbar ke dalam components agar bisa di set menjadi element client. Buat folder dan file components/PanelNavbar.tsx di folder root project.



Dan ambil elemen navbar dari layout.tsx di folder panel ke dalam file ini.

```
1  'use client';
2
3  import { Avatar, Dropdown, Navbar } from "flowbite-react";
4  import { signOut, useSession } from "next-auth/react";
5  import Link from "next/link";
6
7  const PanelNavbar = () => {
8    const { data: session, status } = useSession();
9
10   const handleLogout = async () => {
11     signOut({
12       callbackUrl: "/panel/login",
13     });
14   };
15
16   return (
17     <div className={`border-b border-base-300 sticky top-0 z-50 ${status === "unauthenticated" ? "hidden" : ""}`>
18       <div className="max-w-screen-xl mx-auto">
19         <Navbar fluid rounded>
20           <Navbar.Brand href="/">
21             
22             <span className="self-center whitespace-nowrap text-xl font-semibold dark:text-white">So</span> <span className="text-primary-700">kraf</span> </span> Panel</span>
23           </Navbar.Brand>
24           <div className="flex md:order-2 md:gap-0 gap-2">
25             <Dropdown>
26               arrowIcon={false}
27               inline
28               label={
29                 <Avatar alt="User settings" img={https://ui-avatars.com/api/?background=1A56DB&color=fff&name=${session?.user?.name || "Jhon Doe"}} rounded />
30               >
31                 <Dropdown.Header>
32                   <span className="block text-sm">{session?.user?.name || "Jhon Doe"}</span>
33                   <span className="block truncate text-sm font-medium">{session?.user?.email || "jhondoe@gmail.com"}</span>
34                 </Dropdown.Header>
35                 <Dropdown.Divider />
36                 <Dropdown.Item onClick={handleLogout}>Sign out</Dropdown.Item>
37               </Dropdown>
38             <Navbar.Toggle />
39           </div>
40           <Navbar.Collapse>
41             <Navbar.Link as={Link} href="/panel">Dashboard</Navbar.Link>
42             <Navbar.Link as={Link} href="/panel/categories">Categories</Navbar.Link>
43             <Navbar.Link as={Link} href="/panel/products">Products</Navbar.Link>
44             <Navbar.Link as={Link} href="/panel/admins">Admins</Navbar.Link>
45           </Navbar.Collapse>
46         </div>
47       </div>
48     </div>
49   );
50 };
51
52 export default PanelNavbar;
```

Update juga file layout.tsx di folder panel. Hapus tag html dan body agar tidak duplicate dengan yang ada di root panel. Bungkus semua element menggunakan tag SessionProvider agar semua child root bisa menggunakan session user.

```
1  'use client';
2
3  import "../globals.css";
4  import { SessionProvider } from "next-auth/react";
5  import PanelNavbar from "@components/PanelNavbar";
6
7  export default function PanelLayout({
8    children,
9  }: Readonly<{
10    children: React.ReactNode;
11  }>) {
12    return (
13      <SessionProvider>
14        <PanelNavbar />
15
16        <div className="max-w-screen-xl px-4 mx-auto">
17          {children}
18        </div>
19      </SessionProvider>
20    );
21  }
22
```

Ubah juga nama user di halaman dashboard agar mengambil dari session.

```
1  'use client';
2
3  import React, { useEffect, useState } from "react";
4  import { getTotalAdmins, getTotalCategories, getTotalProducts } from "../api/dashboard";
5  import { useSession } from "next-auth/react";
6
7  export default function Dashboard() {
8    const [admin, setAdmin] = useState(0);
9    const [category, setCategory] = useState(0);
10   const [product, setProduct] = useState(0);
11   const { data: session } = useSession();
12
13   const fetchAdmin = async () => {
14     const totalAdmins = await getTotalAdmins();
15     setAdmin(totalAdmins);
16   }
17
18   const fetchCategory = async () => {
19     const totalCategories = await getTotalCategories();
20     setCategory(totalCategories);
21   }
22
23   const fetchProduct = async () => {
24     const totalProducts = await getTotalProducts();
25     setProduct(totalProducts);
26   }
27
28   useEffect(() => {
29     fetchAdmin();
30     fetchCategory();
31     fetchProduct();
32   }, []);
33
34   return (
35     <div className="pt-2">
36       <div className="text-center flex items-center flex-col mt-5 mb-10">
37         
38         <h3 className="text-lg font-medium mb-3">Welcome, {session?.user?.name}</h3>
39         <h1 className="text-5xl font-bold">SO<span className="text-primary-700">KRAF</span> PANEL</h1>
40       </div>
41       <div className="grid grid-cols-3 gap-6">
42         <div className="sm:col-span-1 col-span-3 p-4 rounded shadow">
43           <div className="text-2xl font-semibold mb-2">{product}</div>
44           <p className="mb-0 text-sm">Total Products</p>
45         </div>
46         <div className="sm:col-span-1 col-span-3 p-4 rounded shadow">
47           <div className="text-2xl font-semibold mb-2">{category}</div>
48           <p className="mb-0 text-sm">Total Categories</p>
49         </div>
50         <div className="sm:col-span-1 col-span-3 p-4 rounded shadow">
51           <div className="text-2xl font-semibold mb-2">{admin}</div>
52           <p className="mb-0 text-sm">Total Admins</p>
53         </div>
54       </div>
55     </div>
56   )
57   };
```