

7. Add Product Module to Web Apps

Open Last project with VS Code (6. Using Material UI for Web Apps Dashboard)

We will adjust `context/AuthContext.jsx`:

```
import React, { createContext, useContext } from "react";

const AuthContext = createContext();

export const useAuth = () => useContext(AuthContext);

export const baseUrl = "https://learning2.pt-mine.id";
export const AuthProvider = ({ children }) => {
  const handleLogin = async (res) => {
    localStorage.setItem("appuser", JSON.stringify(res));
  };

  const handleLogout = () => {
    localStorage.removeItem("appuser");
    window.location.href = "/login";
  };

  const checkLogin = () => {
    try {
      const token = localStorage.getItem("appuser");
      if (token !== null) {
        const appuser = JSON.parse(token);
        return appuser.email;
      } else {
        return false;
      }
    } catch (error) {
      return false;
    }
  };

  return (
    <AuthContext.Provider value={{ checkLogin, handleLogin, handleLogout }}>
      {children}
    </AuthContext.Provider>
  );
};

export const getToken = () => {
  try {
    const token = localStorage.getItem("appuser");
    if (token !== null) {
      const appuser = JSON.parse(token);
      const bearerToken = "Bearer " + appuser.token;
      return bearerToken;
    } else {
```

```

        return null;
    }
} catch (error) {
    return null;
}
};

```

Create file **src/components/productList.jsx** and put this code:

```

import { useState, useEffect } from "react";
import axios from "axios";
import Grid from "@mui/material/Grid";
import Typography from "@mui/material/Typography";
import Card from "@mui/material/Card";
import CardActionArea from "@mui/material/CardActionArea";
import CardContent from "@mui/material/CardContent";
import CardMedia from "@mui/material/CardMedia";
import Button from "@mui/material/Button";
import Paper from "@mui/material/Paper";
import Box from "@mui/material/Box";
import TextField from "@mui/material/TextField";
import { baseUrl, getToken } from "../context/AuthContext";

export default function ProductList() {
    const [products, setProducts] = useState([]);
    const [searchTitle, setSearchTitle] = useState("");
    const [searchDescription, setSearchDescription] = useState("");

    const getProducts = () => {
        axios
            .get(`${baseUrl}/product/list`, {
                params: {
                    title: searchTitle,
                    description: searchDescription,
                },
                headers: {
                    Authorization: getToken(),
                },
            })
            .then((res) => {
                setProducts(res.data);
            })
            .catch((error) => {
                console.error("Error sending data: ", error);
            });
    };

    useEffect(() => {
        getProducts();
    }, []);

    const handleSubmit = (event) => {
        event.preventDefault();
        getProducts();
    };

```

```

};
return (
  <>
    <Grid container spacing={2}>
      <Grid item xs={12}>
        <Paper sx={{ p: 2 }}>
          <Box
            component="form"
            noValidate
            onSubmit={handleSubmit}
            sx={{ mt: 3 }}
          >
            <Grid container spacing={2}>
              <Grid item xs={12}>
                <TextField
                  required
                  fullWidth
                  id="title"
                  label="Title"
                  name="title"
                  value={searchTitle}
                  onChange={(e) =>
                    setSearchTitle(e.target.value)
                  }
                />
              </Grid>

              <Grid item xs={12}>
                <TextField
                  required
                  fullWidth
                  id="description"
                  label="Description"
                  name="description"
                  value={searchDescription}
                  onChange={(e) =>
                    setSearchDescription(e.target.value)
                  }
                />
              </Grid>
            </Grid>
            <Button
              type="submit"
              variant="contained"
              sx={{ mt: 3, mb: 2 }}
            >
              Search
            </Button>
          </Box>
        </Paper>
      </Grid>

      <Grid item xs={12}>
        <Paper
          sx={{ p: 2, display: "flex", flexDirection: "column" }}
        >
          <Grid container spacing={2}>

```

```

        {products.map((row) => (
            <Grid item xs={12} md={6}>
                <CardActionArea component="a" href="#">
                    <Card sx={{ display: "flex" }}>
                        <CardContent sx={{ flex: 1 }}>
                            <Typography
                                component="h2"
                                variant="h5"
                            >
                                {row.title}
                            </Typography>
                            <Typography
                                variant="subtitle1"
                                color="text.secondary"
                            >
                                {row.time}
                            </Typography>
                            <Typography
                                variant="subtitle1"
                                paragraph
                            >
                                {row.description}
                            </Typography>
                            { /*<Button variant="contained">Edit</Button>
                                */ }
                        </CardContent>
                        <CardMedia
                            component="img"
                            sx={{
                                width: 160,
                                display: {
                                    xs: "none",
                                    sm: "block",
                                },
                            }}
                            image={row.image}
                            alt={row.title}
                        />
                    </Card>
                </CardActionArea>
            </Grid>
        )))
    </Grid>
</Paper>
</Grid>
</Grid>
</>
);
}

```

We must include **productList.jsx** in **App.jsx**

Open and put this code in **src/App.jsx**:

```
import * as React from "react";
import { createBrowserRouter, RouterProvider } from "react-router-dom";
import Dashboard from "./Dashboard";
import Page1 from "./components/page1";
import Page2 from "./components/page2";
import ProductList from "./components/productList";
import Login from "./Login";
import { AuthProvider } from "./context/AuthContext";
import { ProtectedRoute } from "./ProtectedRoute";

const router = createBrowserRouter([
  {
    path: "/",
    element: (
      <ProtectedRoute>
        <Dashboard />
      </ProtectedRoute>
    ),
    children: [
      {
        path: "page-one",
        element: <Page1 />,
      },
      {
        path: "page-two",
        element: <Page2 />,
      },
      {
        path: "product-list",
        element: <ProductList />,
      },
    ],
  },
  {
    path: "login",
    element: <Login />,
  },
]);

export default function App() {
  return (
    <AuthProvider>
      <RouterProvider router={router} />
    </AuthProvider>
  );
}
```

To add menu in sidebar,

Open and put this code in **src/Dashboard.jsx**:

```
import * as React from "react";
import { Outlet, Link } from "react-router-dom";
import { useAuth } from "../context/AuthContext";
import { styled, createTheme, ThemeProvider } from "@mui/material/styles";
import CssBaseline from "@mui/material/CssBaseline";
import MuiDrawer from "@mui/material/Drawer";
import Box from "@mui/material/Box";
import MuiAppBar from "@mui/material/AppBar";
import Toolbar from "@mui/material/Toolbar";
import List from "@mui/material/List";
import Typography from "@mui/material/Typography";
import Divider from "@mui/material/Divider";
import IconButton from "@mui/material/IconButton";
import Container from "@mui/material/Container";
import Grid from "@mui/material/Grid";

import MenuIcon from "@mui/icons-material/Menu";
import ChevronLeftIcon from "@mui/icons-material/ChevronLeft";
import ListItemButton from "@mui/material/ListItemButton";
import ListItemIcon from "@mui/material/ListItemIcon";
import ListItemText from "@mui/material/ListItemText";
import LogoutIcon from "@mui/icons-material/Logout";
import ListAltIcon from "@mui/icons-material/ListAlt";

const drawerWidth = 240;

const AppBar = styled(MuiAppBar, {
  shouldForwardProp: (prop) => prop !== "open",
})(({ theme, open }) => ({
  zIndex: theme.zIndex.drawer + 1,
  transition: theme.transitions.create(["width", "margin"], {
    easing: theme.transitions.easing.sharp,
    duration: theme.transitions.duration.leavingScreen,
  }),
  ...(open && {
    marginLeft: drawerWidth,
    width: `calc(100% - ${drawerWidth}px)`,
    transition: theme.transitions.create(["width", "margin"], {
      easing: theme.transitions.easing.sharp,
      duration: theme.transitions.duration.enteringScreen,
    }),
  }),
}));

const Drawer = styled(MuiDrawer, {
  shouldForwardProp: (prop) => prop !== "open",
})(({ theme, open }) => ({
  "& .MuiDrawer-paper": {
    position: "relative",
    whiteSpace: "nowrap",
    width: drawerWidth,
    transition: theme.transitions.create("width", {
      easing: theme.transitions.easing.sharp,
      duration: theme.transitions.duration.enteringScreen,
```

```

    })),
    boxSizing: "border-box",
    ...(!open && {
      overflowX: "hidden",
      transition: theme.transitions.create("width", {
        easing: theme.transitions.easing.sharp,
        duration: theme.transitions.duration.leavingScreen,
      }),
      width: theme.spacing(7),
      [theme.breakpoints.up("sm")]: {
        width: theme.spacing(9),
      },
    })),
  },
});
});

```

```

// TODO remove, this demo shouldn't need to reset the theme.
const defaultTheme = createTheme();

```

```

export default function Dashboard() {
  const [open, setOpen] = React.useState(true);
  const toggleDrawer = () => {
    setOpen(!open);
  };

  const { handleLogout } = useAuth();

  const logout = () => {
    handleLogout();
  };

  return (
    <ThemeProvider theme={defaultTheme}>
      <Box sx={{ display: "flex" }}>
        <CssBaseline />
        <AppBar position="absolute" open={open}>
          <Toolbar
            sx={{
              pr: "24px", // keep right padding when drawer closed
            }}
          >
            <IconButton
              edge="start"
              color="inherit"
              aria-label="open drawer"
              onClick={toggleDrawer}
              sx={{
                marginRight: "36px",
                ...(!open && { display: "none" }),
              }}
            >
              <MenuIcon />
            </IconButton>
            <Typography
              component="h1"
              variant="h6"
              color="inherit"

```

```

        noWrap
        sx={{ flexGrow: 1 }}
      >
        Dashboard
      </Typography>
    </Toolbar>
  </AppBar>
  <Drawer variant="permanent" open={open}>
    <Toolbar
      sx={{
        display: "flex",
        alignItems: "center",
        justifyContent: "flex-end",
        px: [1],
      }}
    >
      <IconButton onClick={toggleDrawer}>
        <ChevronLeftIcon />
      </IconButton>
    </Toolbar>
    <Divider />
    <List component="nav">
      <ListItemButton>
        <ListItemIcon>
          <ListAltIcon />
        </ListItemIcon>
        <Link
          to={` /page-one`}
          style={{ textDecoration: "none" }}
        >
          Go to Page 1
        </Link>
      </ListItemButton>
      <ListItemButton>
        <ListItemIcon>
          <ListAltIcon />
        </ListItemIcon>
        <Link
          to={` /page-two`}
          style={{ textDecoration: "none" }}
        >
          Go to Page 2
        </Link>
      </ListItemButton>
      <ListItemButton>
        <ListItemIcon>
          <ListAltIcon />
        </ListItemIcon>
        <Link
          to={` /product-list`}
          style={{ textDecoration: "none" }}
        >
          Product
        </Link>
      </ListItemButton>
    </List>
  </Drawer>
</MainContent>
</PageContent>
</Page>
</App>

```



```

        <ListItemButton>
          <ListItemIcon>
            <LogoutIcon />
          </ListItemIcon>
          <ListItemText onClick={logout} primary="Log Out" />
        </ListItemButton>
      </List>
    </Drawer>
    <Box
      component="main"
      sx={{
        backgroundColor: (theme) =>
          theme.palette.mode === "light"
            ? theme.palette.grey[100]
            : theme.palette.grey[900],
        flexGrow: 1,
        height: "100vh",
        overflow: "auto",
      }}
    >
      <Toolbar />
      <Container maxWidth="lg" sx={{ mt: 4, mb: 4 }}>
        <Grid container spacing={2}>
          <Grid item xs={12}>
            <Outlet />
          </Grid>
        </Grid>
      </Container>
    </Box>
  </Box>
</ThemeProvider>
);
}

```

Run this project: **npm run dev**

