

```

import React, { useEffect, useState } from "react";
import { ProSidebar, Menu, MenuItem } from "react-pro-sidebar";
import "react-pro-sidebar/dist/css/styles.css";
import { Box, IconButton, Typography, useTheme } from "@mui/material";
import { Link, useLocation } from "react-router-dom";
import { tokens } from "../../theme";
import HomeOutlinedIcon from "@mui/icons-material/HomeOutlined";
import BarChartOutlinedIcon from "@mui/icons-material/BarChartOutlined";
import MenuOutlinedIcon from "@mui/icons-material/MenuOutlined";
import MapOutlinedIcon from "@mui/icons-material/MapOutlined";
import DevicesOutlinedIcon from "@mui/icons-material/DevicesOutlined";
import ExitToAppIcon from "@mui/icons-material/ExitToApp";

```

```

const Item = ({ title, to, icon, selected, setSelected, isCollapsed }) => {
  const theme = useTheme();
  const colors = tokens(theme.palette.mode);
  return (
    <MenuItem
      active={selected === title}
      style={{
        color: colors.gray[100],
      }}
      onClick={() => setSelected(title)}
      icon={icon}
    >
      {!isCollapsed && <Typography>{title}</Typography>}
      <Link to={to} style={{ textDecoration: "none", color: "inherit" }} />
    </MenuItem>
  );
};

```

```

const Sidebar = ({ onLogout }) => {
  const theme = useTheme();
  const colors = tokens(theme.palette.mode);
  const [isCollapsed, setIsCollapsed] = useState(true);
  const [selected, setSelected] = useState(
    localStorage.getItem("selectedItem") || "Dashboard",
  );
  const location = useLocation();

```

```

  useEffect(() => {
    const pathname = location.pathname;
    if (pathname.includes("/dashboard")) {
      setSelected("Dashboard");
    } else if (pathname.includes("/device-config")) {
      setSelected("Device Configuration");
    }
  });

```

```

    } else if (pathname.includes("/device-location")) {
      setSelected("Device Location");
    } else if (pathname.includes("/historical")) {
      setSelected("History and Analytics");
    }
  }, [location.pathname]);

  useEffect(() => {
    setIsCollapsed(true);
  }, []);

  useEffect(() => {
    localStorage.setItem("selectedItem", selected);
  }, [selected]);

  if (location.pathname === "/") {
    return null;
  }

  return (
    <Box
      sx={{
        "& .pro-sidebar-inner": {
          background: `${colors.primary[400]} !important`,
        },
        "& .pro-icon-wrapper": {
          backgroundColor: "transparent !important",
        },
        "& .pro-inner-item": {
          padding: "5px 15px 20px !important",
        },
        "& .pro-inner-item:hover": {
          color: `${colors.orangeAccent[300]} !important`,
        },
        "& .pro-menu-item.active": {
          color: `${colors.orangeAccent[400]} !important`,
        },
      }}
    >
    <ProSidebar
      collapsed={isCollapsed}
      width={isCollapsed ? "65px" : "270px"}
      collapsedWidth="65px"
    >
      <Menu iconShape="square">
        <MenuItem

```

```

onClick={() => setIsCollapsed(!isCollapsed)}
icon={isCollapsed ? <MenuOutlinedIcon /> : undefined}
style={{
  margin: "10px 0 20px 0",
  color: colors.gray[100],
}}
>
{!isCollapsed && (
  <Box
    display="flex"
    justifyContent="space-between"
    alignItems="center"
    ml="15px"
  >
    <Typography variant="h3" color={colors.gray[100]}>
      ADMIN
    </Typography>
    <IconButton onClick={() => setIsCollapsed(!isCollapsed)}>
      <MenuOutlinedIcon />
    </IconButton>
  </Box>
)}
</MenuItem>

{/* user */}
{!isCollapsed && (
  <Box mb="25px">
    <Box display="flex" justifyContent="center" alignItems="center">
      <img
        alt="gutterguard-logo"
        width="100px"
        height="100px"
        src={"../assets/ggutterguard-icon.png"}
      />
    </Box>

    <Box textAlign="center">
      <Typography
        variant="h2"
        color={colors.gray[100]}
        fontWeight="bold"
        sx={{ m: "10px 0 0 0" }}
      >
        GutterGuard
      </Typography>
      <Typography variant="h5" color={colors.orangeAccent[400]}>

```

```

        Admin Access
      </Typography>
    </Box>
  </Box>
)}

<Box paddingLeft={isCollapsed ? undefined : "10%"}>
  <Item
    title="Dashboard"
    to="/dashboard"
    icon={<HomeOutlinedIcon />}
    selected={selected}
    setSelected={setSelected}
    isCollapsed={isCollapsed}
  />
  <Item
    title="Device Configuration"
    to="/device-config"
    icon={<DevicesOutlinedIcon />}
    selected={selected}
    setSelected={setSelected}
    isCollapsed={isCollapsed}
  />
  <Item
    title="Device Location"
    to="/device-location"
    icon={<MapOutlinedIcon />}
    selected={selected}
    setSelected={setSelected}
    isCollapsed={isCollapsed}
  />
  <Item
    title="History and Analytics"
    to="/historical"
    icon={<BarChartOutlinedIcon />}
    selected={selected}
    setSelected={setSelected}
    isCollapsed={isCollapsed}
  />
</Box>
<Menu iconShape="square">
  <MenuItem
    style={{
      color: colors.gray[100],
      position: "absolute",
      bottom: 0,

```

```

        width: "100%",
      }}
      icon={<ExitToAppIcon />}
    >
      {!isCollapsed && <Typography>Logout</Typography>}
      <Link
        to="/"
        onClick={onLogout}
        style={{ textDecoration: "none", color: "inherit" }}
      />
    </MenuItem>
  </Menu>
</Menu>
</ProSidebar>
</Box>
);
};

```

```

export default Sidebar;

```