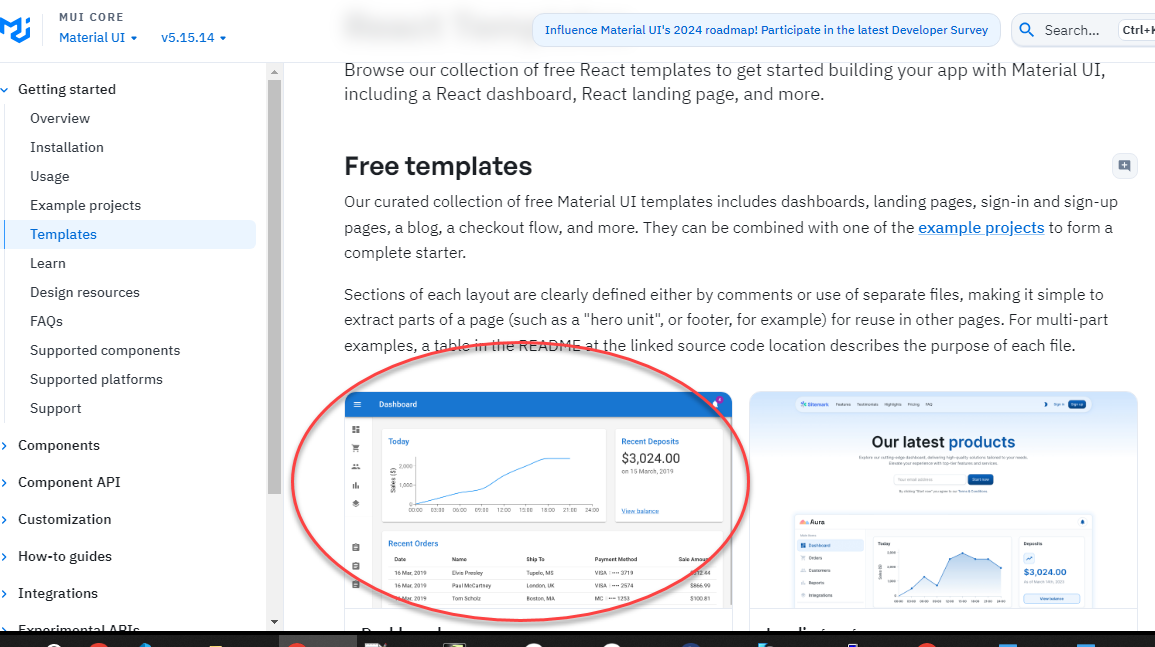
# 6. Using Material UI for Web Apps Dashboard

To shorten the time, we can use an interface library such as Material UI

You can see example here : <https://mui.com/material-ui/getting-started/templates/>



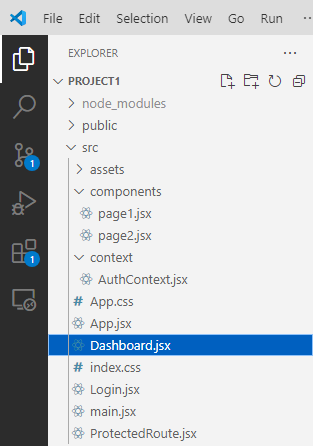
Open last project (5. Add Login Protection in Web Apps)

In last tutorial we give the component folder name with **routes**. The aim is to clarify the function of the folder.  
But now we must rename **routes** folder name to **components**, to conform with standards used by many people.

First, rename “**src/routes**” to “**src/components**”

Then rename “**src/components/root.jsx**” to “**src/components/Dashboard.jsx**”

**Move** “**src/components/Dashboard.jsx**” to “**src/Dashboard.jsx**”

This is files structure of our project  


Because routes folder has been changed. We must update code in App.jsx

Open and replace **src/App.jsx** with this code:  
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 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: "login",

        element: <Login />,

    },

]);

export default function App() {

    return (

        <AuthProvider>

            <RouterProvider router={router} />

        </AuthProvider>

    );

}

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 Paper from "@mui/material/Paper";

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),

            },

        }),

    },

}));

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>

                                <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={3}>

                            <Grid item xs={12}>

                                <Paper

                                    sx={{

                                        p: 2,

                                        display: "flex",

                                        flexDirection: "column",

                                    }}

                                >

                                    <Outlet />

                                </Paper>

                            </Grid>

                        </Grid>

                    </Container>

                </Box>

            </Box>

        </ThemeProvider>

    );

}

Run this project:

**npm run dev**

