**Users component – slide 41-46**

import React, { useState, useEffect } from "react";

import {

  Paper,

  List,

  ListItem,

  ListItemAvatar,

  ListItemText,

  ListItemSecondaryAction,

  IconButton,

  Avatar,

  Typography,

  Link,

} from "@mui/material";

import ArrowForward from "@mui/icons-material/ArrowForward";

import { list } from "./api-user.js";

import { Link as RouterLink } from "react-router-dom";

export default function Users() {

  const [users, setUsers] = useState([]);

  useEffect(() => {

    const abortController = new AbortController();

    const signal = abortController.signal;

  list(signal).then((data) => {

      if (data?.error) {

        console.log(data.error);

      } else {

        setUsers(data);

      }

    });

    return () => abortController.abort();

  }, []);

  return (

    <Paper

      elevation={4}

      sx={{

        maxWidth: 600,

        mx: "auto",

        mt: 5,

        p: 3,

      }}

    >

      <Typography variant="h6" sx={{ mb: 2, color: "text.primary" }}>

        All Users

      </Typography>

      <List dense>

        {users.map((item, i) => (

          <Link

            component={RouterLink}

            to={`/user/${item.\_id}`}

            underline="none"

            key={item.\_id}

            sx={{ color: "inherit" }}

          >

            <ListItem button>

              <ListItemAvatar>

                <Avatar />

              </ListItemAvatar>

              <ListItemText primary={item.name} />

              <ListItemSecondaryAction>

                <IconButton edge="end">

                  <ArrowForward />

                </IconButton>

              </ListItemSecondaryAction>

            </ListItem>

          </Link>

        ))}

      </List>

    </Paper>

  );

}

**Signup component code – slide 47 – 59**

import React, { useState } from "react";

import {

  Card,

  CardContent,

  Typography,

  TextField,

  CardActions,

  Button,

  Dialog,

  DialogTitle,

  DialogContent,

  DialogContentText,

  DialogActions,

} from "@mui/material";

import { Link } from "react-router-dom";

import { create } from "./api-user";

export default function Signup() {

  const [values, setValues] = useState({

    name: "",

    password: "",

    email: "",

    error: "",

  });

  const [open, setOpen] = useState(false);

  const handleChange = (name) => (event) => {

    setValues({ ...values, [name]: event.target.value });

  };

  const handleClose = () => {

    setOpen(false);

  };

  const clickSubmit = () => {

    const user = {

      name: values.name || undefined,

      email: values.email || undefined,

      password: values.password || undefined,

    };

    create(user).then((data) => {

      if (data.error) {

        setValues({ ...values, error: data.error });

      } else {

        setOpen(true);

      }

    });

  };

  return (

    <div>

      <Card

        sx={{

          maxWidth: 400,

          margin: "0 auto",

          mt: 3,

          p: 2,

          textAlign: "center",

        }}

      >

        <CardContent>

          <Typography variant="h6" sx={{ fontSize: 18 }}>

            Sign Up

          </Typography>

          <TextField

            id="name"

            label="Name"

            sx={{ width: "100%", mb: 2 }}

            value={values.name}

            onChange={handleChange("name")}

            margin="normal"

          />

  <TextField

            id="email"

            label="Email"

            sx={{ width: "100%", mb: 2 }}

            value={values.email}

            onChange={handleChange("email")}

            margin="normal"

          />

          <TextField

            id="password"

            label="Password"

            sx={{ width: "100%", mb: 2 }}

            value={values.password}

            onChange={handleChange("password")}

            type="password"

            margin="normal"

          />

          {values.error && (

            <Typography color="error" sx={{ mt: 1 }}>

              {values.error}

            </Typography>

          )}

  </CardContent>

        <CardActions>

          <Button

            color="primary"

            variant="contained"

            onClick={clickSubmit}

            sx={{ margin: "0 auto", mb: 2 }}

          >

            Submit

          </Button>

        </CardActions>

      </Card>

      <Dialog open={open} onClose={handleClose}>

        <DialogTitle>New Account</DialogTitle>

        <DialogContent>

          <DialogContentText>

            New account successfully created.

          </DialogContentText>

        </DialogContent>

        <DialogActions>

          <Link to="/signin">

            <Button

  color="primary"

              autoFocus

              variant="contained"

              onClick={handleClose}

            >

              Sign In

            </Button>

          </Link>

        </DialogActions>

      </Dialog>

    </div>

  );

}

**Signin component code - slide 60 – 69**

import React, { useState } from "react";

import Card from "@mui/material/Card";

import CardActions from "@mui/material/CardActions";

import CardContent from "@mui/material/CardContent";

import Button from "@mui/material/Button";

import TextField from "@mui/material/TextField";

import Typography from "@mui/material/Typography";

import Icon from "@mui/material/Icon";

import auth from "./auth-helper.js";

import { Navigate, useLocation } from "react-router-dom";

import { signin } from "./api-auth.js";

export default function Signin() {

  const location = useLocation();

  const [values, setValues] = useState({

    email: "",

    password: "",

    error: "",

    redirectToReferrer: false,

  });

  const clickSubmit = () => {

    const user = {

      email: values.email || undefined,

      password: values.password || undefined,

    };

    signin(user).then((data) => {

      if (data.error) {

        setValues({ ...values, error: data.error });

      } else {

        auth.authenticate(data, () => {

          setValues({ ...values, error: "", redirectToReferrer: true });

        });

      }

    });

  };

  const handleChange = (name) => (event) => {

    setValues({ ...values, [name]: event.target.value });

  };

  const { from } = location.state || {

    from: { pathname: "/" },

  };

const { redirectToReferrer } = values;

  if (redirectToReferrer) {

    return <Navigate to={from} />;

  }

  return (

    <Card

      sx={{

        maxWidth: 600,

        margin: "auto",

        textAlign: "center",

        mt: 5,

        pb: 2,

      }}

    >

      <CardContent>

        <Typography variant="h6" sx={{ mt: 2, color: "text.primary" }}>

          Sign In

        </Typography>

        <TextField

          id="email"

          type="email"

          label="Email"

          sx={{ mx: 1, width: 300 }}

  value={values.email}

          onChange={handleChange("email")}

          margin="normal"

        />

        <br />

        <TextField

          id="password"

          type="password"

          label="Password"

          sx={{ mx: 1, width: 300 }}

          value={values.password}

          onChange={handleChange("password")}

          margin="normal"

        />

        <br />

        {values.error && (

          <Typography component="p" color="error" sx={{ mt: 1 }}>

            <Icon color="error" sx={{ verticalAlign: "middle", mr: 0.5 }}>

              error

            </Icon>

            {values.error}

          </Typography>

        )}

      </CardContent>

      <CardActions>

        <Button

          color="primary"

          variant="contained"

          onClick={clickSubmit}

          sx={{ margin: "auto", mb: 2 }}

        >

          Submit

        </Button>

      </CardActions>

    </Card>

  );

}

**Profile component - slide 70 - 81**

import React, { useState, useEffect } from "react";

import {

  Paper,

  List,

  ListItem,

  ListItemAvatar,

  ListItemSecondaryAction,

  ListItemText,

  Avatar,

  IconButton,

  Typography,

  Divider,

} from "@mui/material";

import EditIcon from "@mui/icons-material/Edit";

import PersonIcon from "@mui/icons-material/Person";

import DeleteUser from "./DeleteUser";

import auth from "../lib/auth-helper.js";

import { read } from "./api-user.js";

import { useLocation, Navigate, Link, useParams } from "react-router-dom";

export default function Profile() {

  const location = useLocation();

  const [user, setUser] = useState({});

  const [redirectToSignin, setRedirectToSignin] = useState(false);

  const jwt = auth.isAuthenticated();

  const { userId } = useParams();

  useEffect(() => {

    const abortController = new AbortController();

    const signal = abortController.signal;

    read({ userId }, { t: jwt.token }, signal).then((data) => {

      if (data && data.error) {

        setRedirectToSignin(true);

      } else {

        setUser(data);

      }

    });

    return () => abortController.abort();

  }, [userId]);

  if (redirectToSignin) {

    return (

      <Navigate to="/signin" state={{ from: location.pathname }} replace />

    );

  }

  return (

    <Paper

      elevation={4}

      sx={{

        maxWidth: 600,

        mx: "auto",

        mt: 5,

        p: 3,

      }}

    >

      <Typography variant="h6" sx={{ mt: 3, mb: 2, color: "text.primary" }}>

        Profile

      </Typography>

      <List dense>

        <ListItem>

          <ListItemAvatar>

            <Avatar>

              <PersonIcon />

            </Avatar>

          </ListItemAvatar>

          <ListItemText primary={user.name} secondary={user.email} />

          {auth.isAuthenticated().user &&

            auth.isAuthenticated().user.\_id === user.\_id && (

              <ListItemSecondaryAction>

                <Link to={`/user/edit/${user.\_id}`}>

                  <IconButton aria-label="Edit" color="primary">

                    <EditIcon />

                  </IconButton>

                </Link>

                <DeleteUser userId={user.\_id} />

              </ListItemSecondaryAction>

            )}

        </ListItem>

        <Divider />

        <ListItem>

          <ListItemText

            primary={

              user.created

                ? `Joined: ${new Date(user.created).toDateString()}`

                : "Loading..."

            }

          />

        </ListItem>

      </List>

    </Paper>

  );

}

**EditProfile component – slide 83 – 96**

import React, { useState, useEffect } from "react";

import {

  Card,

  CardActions,

  CardContent,

  Button,

  TextField,

  Typography,

  Icon,

} from "@mui/material";

import auth from "../lib/auth-helper.js";

import { read, update } from "./api-user.js";

import { Navigate, useParams } from "react-router-dom";

export default function EditProfile() {

  const { userId } = useParams();

  const [values, setValues] = useState({

    name: "",

    password: "",

    email: "",

    open: false,

    error: "",

    NavigateToProfile: false,

  });

  const jwt = auth.isAuthenticated();

  useEffect(() => {

    const abortController = new AbortController();

    const signal = abortController.signal;

    read({ userId }, { t: jwt.token }, signal).then((data) => {

      if (data?.error) {

        setValues((prev) => ({ ...prev, error: data.error }));

      } else {

        setValues((prev) => ({ ...prev, name: data.name, email: data.email }));

      }

    });

    return () => abortController.abort();

  }, [userId]);

  const clickSubmit = () => {

    const user = {

      name: values.name || undefined,

      email: values.email || undefined,

      password: values.password || undefined,

    };

    update({ userId }, { t: jwt.token }, user).then((data) => {

      if (data?.error) {

        setValues((prev) => ({ ...prev, error: data.error }));

      } else {

        setValues((prev) => ({

          ...prev,

          userId: data.\_id,

          NavigateToProfile: true,

        }));

      }

    });

  };

  const handleChange = (name) => (event) => {

    setValues({ ...values, [name]: event.target.value });

  };

  if (values.NavigateToProfile) {

    return <Navigate to={`/user/${values.userId}`} />;

  }

  return (

    <Card

      sx={{

        maxWidth: 600,

        mx: "auto",

        mt: 5,

        textAlign: "center",

        pb: 2,

      }}

    >

      <CardContent>

        <Typography variant="h6" sx={{ mt: 2, mb: 2, color: "text.primary" }}>

          Edit Profile

        </Typography>

        <TextField

          id="name"

          label="Name"

          value={values.name}

          onChange={handleChange("name")}

          margin="normal"

          sx={{ mx: 1, width: 300 }}

        />

        <br />

        <TextField

          id="email"

          type="email"

          label="Email"

          value={values.email}

          onChange={handleChange("email")}

          margin="normal"

          sx={{ mx: 1, width: 300 }}

        />

        <br />

        <TextField

          id="password"

          type="password"

          label="Password"

          value={values.password}

          onChange={handleChange("password")}

          margin="normal"

          sx={{ mx: 1, width: 300 }}

        />

        <br />

  {values.error && (

          <Typography component="p" color="error" sx={{ mt: 1 }}>

            <Icon color="error" sx={{ verticalAlign: "middle", mr: 1 }}>

              error

            </Icon>

            {values.error}

          </Typography>

        )}

      </CardContent>

      <CardActions sx={{ justifyContent: "center" }}>

        <Button

          color="primary"

          variant="contained"

          onClick={clickSubmit}

          sx={{ mb: 2 }}

        >

          Submit

        </Button>

      </CardActions>

    </Card>

  );

}

**DeleteUser component – slide 97 – 104**

import React, { useState } from "react";

import PropTypes from "prop-types";

import IconButton from "@mui/material/IconButton";

import Button from "@mui/material/Button";

import DeleteIcon from "@mui/icons-material/Delete";

import Dialog from "@mui/material/Dialog";

import DialogActions from "@mui/material/DialogActions";

import DialogContent from "@mui/material/DialogContent";

import DialogContentText from "@mui/material/DialogContentText";

import DialogTitle from "@mui/material/DialogTitle";

import auth from "../lib/auth-helper.js";

import { remove } from "./api-user.js";

import { Navigate } from "react-router-dom";

export default function DeleteUser({ userId }) {

  const [open, setOpen] = useState(false);

  const [redirect, setRedirect] = useState(false);

  const jwt = auth.isAuthenticated();

  const clickButton = () => {

    setOpen(true);

  };

  const deleteAccount = () => {

    remove({ userId }, { t: jwt.token }).then((data) => {

      if (data?.error) {

        console.error(data.error);

      } else {

        auth.clearJWT(() => console.log("deleted"));

        setRedirect(true);

      }

    });

  };

  const handleRequestClose = () => {

    setOpen(false);

  };

  if (redirect) {

    return <Navigate to="/" />;

  }

  return (

    <>

  <IconButton

        aria-label="Delete account"

        onClick={clickButton}

        color="error"

      >

        <DeleteIcon />

      </IconButton>

      <Dialog open={open} onClose={handleRequestClose}>

        <DialogTitle>Delete Account</DialogTitle>

        <DialogContent>

          <DialogContentText>

            Are you sure you want to delete your account? This action is

            irreversible.

          </DialogContentText>

        </DialogContent>

        <DialogActions>

          <Button onClick={handleRequestClose} color="primary">

            Cancel

          </Button>

          <Button

            onClick={deleteAccount}

            color="error"

            variant="contained"

            autoFocus

          >

            Confirm

          </Button>

        </DialogActions>

      </Dialog>

    </>

  );

}

DeleteUser.propTypes = {

  userId: PropTypes.string.isRequired,

};

Menu component – slide 105 –

import React from "react";

import AppBar from "@mui/material/AppBar";

import Toolbar from "@mui/material/Toolbar";

import Typography from "@mui/material/Typography";

import IconButton from "@mui/material/IconButton";

import HomeIcon from "@mui/icons-material/Home";

import Button from "@mui/material/Button";

import auth from "../lib/auth-helper";

import { Link, useNavigate, useLocation } from "react-router-dom";

const isActive = (location, path) =>

  location.pathname === path ? "#ff4081" : "#ffffff";

export default function Menu() {

  const navigate = useNavigate();

  const location = useLocation();

  return (

    <AppBar position="static">

      <Toolbar sx={{ display: "flex", gap: 2, alignItems: "center" }}>

        <Typography variant="h6" sx={{ flexGrow: 1 }}>

          MERN Skeleton

        </Typography>

  <Link to="/">

          <IconButton aria-label="Home" sx={{ color: isActive(location, "/") }}>

            <HomeIcon />

          </IconButton>

        </Link>

        <Link to="/users">

          <Button sx={{ color: isActive(location, "/users") }}>Users</Button>

        </Link>

        {!auth.isAuthenticated() && (

          <>

            <Link to="/signup">

              <Button sx={{ color: isActive(location, "/signup") }}>

                Sign up

              </Button>

            </Link>

            <Link to="/signin">

              <Button sx={{ color: isActive(location, "/signin") }}>

                Sign In

              </Button>

            </Link>

          </>

        )}

{auth.isAuthenticated() && (

          <>

            <Link to={`/user/${auth.isAuthenticated().user.\_id}`}>

              <Button

                sx={{

                  color: isActive(

                    location,

                    `/user/${auth.isAuthenticated().user.\_id}`

                  ),

                }}

              >

                My Profile

              </Button>

            </Link>

            <Button

              sx={{ color: "#ffffff" }}

              onClick={() => {

                auth.clearJWT(() => navigate("/"));

              }}

            >

              Sign out

            </Button>

          </>

        )}

      </Toolbar>

    </AppBar>

  );

}

**BACKEND CONNECTION – slide 121**

vite.config.js

import { defineConfig } from "vite";

import react from "@vitejs/plugin-react";

const { PORT = 3000} = process.env;

export default defineConfig({

 plugins: [react()],

server:{

    proxy:{

        '/api':{

            target:`http://localhost:${PORT}`,

            changeOrigin: true,

        },

        '/auth': {

            target:`http://localhost:${PORT}`,

            changeOrigin: true,

        },

    },

},

 build: {

 manifest: true,

 rollupOptions: {

 input: "./src/main.jsx",

 },

 },

});