



React App

localhost:3000/admin/dashboard

Admin Panel

Dashboard

Mahasiswa

Mahasiswa

Dashboard Admin

Selamat datang! Berikut ringkasan data sistem saat ini.

Jumlah Mahasiswa
5

Mahasiswa Aktif
2

Mahasiswa Tidak Aktif
3

Grafik Mahasiswa

A bar chart titled "Grafik Mahasiswa" showing the distribution of students by status. The Y-axis represents the count from 0 to 8. The X-axis categories are "Total", "Aktif", and "Non-Aktif". The "Total" bar is blue and reaches a value of 5. The "Aktif" bar is blue and reaches a value of 2, with a tooltip indicating "Aktif value : 2". The "Non-Aktif" bar is blue and reaches a value of 3.

© 2025 Admin Dashboard. All rights reserved.

React App

localhost:3000/admin/mahasiswa

Admin Panel

Dashboard

Mahasiswa

Mahasiswa

Cari nama atau NIM...

+ Tambah Mahasiswa

NIM	Nama	Status	Aksi
A11.2022.14283	Eva Fadhillah	Aktif	<button>Detail</button> <button>Edit</button> <button>Hapus</button>
A11.2022.14284	Sofyan Roy Irawan	Aktif	<button>Detail</button> <button>Edit</button> <button>Hapus</button>
A11.2022.14285	Elzian Deva Alzarif	Tidak Aktif	<button>Detail</button> <button>Edit</button> <button>Hapus</button>
A11.2022.14286	Wartini	Tidak Aktif	<button>Detail</button> <button>Edit</button> <button>Hapus</button>
A11.2022.14287	Reza Erwia	Tidak Aktif	<button>Detail</button> <button>Edit</button> <button>Hapus</button>

Menampilkan 1 - 5 dari 5 mahasiswa

Previous

Next

Halaman 1 dari 1

© 2025 Admin Dashboard. All rights reserved.

React App

localhost:3000/admin/mahasiswa

Admin Panel

Mahasiswa

Cari nama atau NIM... + Tambah Mahasiswa

NIM	Nama	Status	Aksi
A11.2022.14283	Eva Fadhillah	Aktif	<button>Detail</button> <button>Edit</button> <button>Hapus</button>
A11.2022.14284	Sofia	Pending	<button>Detail</button> <button>Edit</button> <button>Hapus</button>
A11.2022.14285	Elza	Pending	<button>Detail</button> <button>Edit</button> <button>Hapus</button>
A11.2022.14286	Wulan	Pending	<button>Detail</button> <button>Edit</button> <button>Hapus</button>
A11.2022.14287	Rani	Pending	<button>Detail</button> <button>Edit</button> <button>Hapus</button>

Menampilkan 1 - 5 dari 5 mahasiswa

Halaman 1 dari 1

Tambah Mahasiswa

NIM
A11.2022.142837

Nama
Dwi Yuliyanti

Status
Tidak Aktif

Batal Simpan

© 2025 Admin Dashboard. All rights reserved.

React App

localhost:3000/admin/mahasiswa

Admin Panel

Dashboard

Mahasiswa

Mahasiswa

Cari nama atau NIM...

+ Tambah Mahasiswa

Aksi

NIM	Nama	Status	Aksi
A11.2022.14283	Eva Fadhillah	Aktif	<button>Detail</button> <button>Edit</button> <button>Hapus</button>
A11.2022.14284	Sofyan Roy Irawan	Aktif	<button>Detail</button> <button>Edit</button> <button>Hapus</button>
A11.2022.14285	Elzian Deva Alzarif	Tidak Aktif	<button>Detail</button> <button>Edit</button> <button>Hapus</button>
A11.2022.14286	Wartini	Tidak Aktif	<button>Detail</button> <button>Edit</button> <button>Hapus</button>
A11.2022.14287	Reza Erwia	Tidak Aktif	<button>Detail</button> <button>Edit</button> <button>Hapus</button>

Menampilkan 1 - 5 dari 6 mahasiswa

Previous

Next

Halaman 1 dari 2

© 2025 Admin Dashboard. All rights reserved.



React App

localhost:3000/admin/mahasiswa

Admin Panel

Dashboard

Mahasiswa

Mahasiswa

Daftar Mahasiswa

Cari nama atau NIM...

+ Tambah Mahasiswa

NIM	Nama	Status	Aksi
A11.2022.142837	Dwi Yuliyanti	Tidak Aktif	<button>Detail</button> <button>Edit</button> <button>Hapus</button>

Menampilkan 6 - 6 dari 6 mahasiswa

Previous

Next

Halaman 2 dari 2

© 2025 Admin Dashboard. All rights reserved.

React App

localhost:3000/admin/mahasiswa/A11.2022.142837

Admin Panel

Mahasiswa

Dashboard

Mahasiswa

Detail Mahasiswa

NIM
A11.2022.142837

Nama
Dwi Yuliyanti

Status
Tidak Aktif

← Kembali ke Daftar Mahasiswa

© 2025 Admin Dashboard. All rights reserved.

The screenshot shows a web-based administrative interface for managing students. On the left, a dark blue sidebar titled 'Admin Panel' contains links for 'Dashboard' and 'Mahasiswa'. The main content area is titled 'Mahasiswa' and displays a detailed view of a student record. A large, semi-transparent callout box is centered over the student details. The box has a light gray background and rounded corners. It contains the following information:

- Detail Mahasiswa** (Section title)
- NIM**: A11.2022.142837 (with a blue checkmark icon)
- Nama**: Dwi Yuliyanti (with a person icon)
- Status**: Tidak Aktif (with a red circle and cross icon)

At the bottom of the callout box is a blue button with white text that says "← Kembali ke Daftar Mahasiswa". At the very bottom of the page, there is a small copyright notice: "© 2025 Admin Dashboard. All rights reserved."

React App

localhost:3000/admin/mahasiswa

Admin Panel

Dashboard

Mahasiswa

Mahasiswa

localhost:3000 says
Edit data mahasiswa "Dwi Yuliyanti"?

Cari nama atau NIM... + Tambah Mahasiswa

Daftar Mahasiswa

NIM	Nama	Status	Aksi
A11.2022.142837	Dwi Yuliyanti	Tidak Aktif	Detail Edit Hapus

Menampilkan 6 - 6 dari 6 mahasiswa

Previous Next

Halaman 2 dari 2

© 2025 Admin Dashboard. All rights reserved.

React App

localhost:3000/admin/mahasiswa

Admin Panel

Dashboard

Mahasiswa

Cari nama atau NIM...

+ Tambah Mahasiswa

Daftar Mahasiswa

NIM	Nama	Status	Aksi
A11.2022.142837	Dwi Yuliyanti	Tidak Aktif	<button>Detail</button> <button>Edit</button> <button>Hapus</button>

Menampilkan 6 - 6 dari 6 mahasiswa

Halaman 2 dari 2

Previous

Next

Edit Mahasiswa

X

NIM

A11.2022.142837

Nama

Dwi Yuliyanti Nora

Status

Aktif

Batal

Simpan Perubahan

© 2025 Admin Dashboard. All rights reserved.

React App

localhost:3000/admin/mahasiswa

Admin Panel

Mahasiswa

localhost:3000 says

Yakin ingin mengupdate data mahasiswa NIM A11.2022.142837?

Cancel OK

Cari nama atau NIM... + Tambah Mahasiswa

Daftar Mahasiswa

NIM	Nama	Status	Aksi
A11.2022.142837	Dwi Yuliyanti	Tidak Aktif	<button>Detail</button> <button>Edit</button> <button>Hapus</button>

Menampilkan 6 - 6 dari 6 mahasiswa

Halaman 2 dari 2

Previous Next

Edit Mahasiswa

NIM
A11.2022.142837

Nama
Dwi Yuliyanti Nora

Status
Aktif

Batal Simpan Perubahan

© 2025 Admin Dashboard. All rights reserved.

This screenshot shows a React-based Admin Dashboard for managing students. On the left, a sidebar titled 'Admin Panel' includes links for 'Dashboard' and 'Mahasiswa'. The main area is titled 'Mahasiswa' and contains a table of student records. One record is selected: 'A11.2022.142837' (Dwi Yuliyanti, Tidak Aktif). A modal window titled 'Edit Mahasiswa' is open, allowing changes to the student's NIM, Name, and Status. The status dropdown is set to 'Aktif'. At the bottom of the modal are 'Batal' (Cancel) and 'Simpan Perubahan' (Save Changes) buttons. A confirmation dialog box from 'localhost:3000' asks if the user wants to update the student, with 'Cancel' and 'OK' buttons. The footer of the page includes a copyright notice for 2025.

React App

localhost:3000/admin/mahasiswa

Admin Panel

Dashboard

Mahasiswa

Mahasiswa

localhost:3000 says
Edit data mahasiswa "Dwi Yuliyanti"?

Cari nama atau NIM... + Tambah Mahasiswa

Daftar Mahasiswa

NIM	Nama	Status	Aksi
A11.2022.142837	Dwi Yuliyanti	Tidak Aktif	<button>Detail</button> <button>Edit</button> <button>Hapus</button>

Menampilkan 6 - 6 dari 6 mahasiswa

Previous Next

Halaman 2 dari 2

© 2025 Admin Dashboard. All rights reserved.

React App

localhost:3000/admin/mahasiswa

Admin Panel

Dashboard

Mahasiswa

Mahasiswa

Data mahasiswa berhasil diupdate!

Cari nama atau NIM...

+ Tambah Mahasiswa

Daftar Mahasiswa

NIM	Nama	Status	Aksi
A11.2022.142837	Dwi Yuliyanti Nora	Aktif	<button>Detail</button> <button>Edit</button> <button>Hapus</button>

Menampilkan 6 - 6 dari 6 mahasiswa

Previous

Next

Halaman 2 dari 2

© 2025 Admin Dashboard. All rights reserved.

React App

localhost:3000/admin/mahasiswa

Admin Panel

Dashboard

Mahasiswa

Mahasiswa

localhost:3000 says

Yakin ingin menghapus data mahasiswa "Dwi Yuliyanti Nora" (NIM: A11.2022.142837)?

Cancel OK

Cari nama atau NIM... + Tambah Mahasiswa

Daftar Mahasiswa

NIM	Nama	Status	Aksi
A11.2022.142837	Dwi Yuliyanti Nora	Aktif	<button>Detail</button> <button>Edit</button> <button>Hapus</button>

Menampilkan 6 - 6 dari 6 mahasiswa

Previous Next

Halaman 2 dari 2

© 2025 Admin Dashboard. All rights reserved.

React App

localhost:3000/admin/mahasiswa

Admin Panel

Dashboard

Mahasiswa

Mahasiswa

Data mahasiswa berhasil dihapus!

Cari nama atau NIM...

+ Tambah Mahasiswa

Daftar Mahasiswa

NIM	Nama	Status	Aksi
Tidak ada data mahasiswa yang sesuai			

Menampilkan 6 - 5 dari 5 mahasiswa

Previous

Next

Halaman 2 dari 1

© 2025 Admin Dashboard. All rights reserved.

React App

localhost:3000/admin/mahasiswa#

Admin Panel

Dashboard

Mahasiswa

localhost:3000 says

Logout berhasil!

OK

Daftar Mahasiswa

Cari nama atau NIM...

NIM

Nama

Status

Tidak ada data mahasiswa yang sesuai

Menampilkan 6 - 5 dari 5 mahasiswa

Previous

Next

Admin
admin@gmail.com

Profile

Logout

Halaman 2 dari 1

© 2025 Admin Dashboard. All rights reserved.



```
// Button.js => src/components/atoms/Button.js

import React from "react";
import PropTypes from "prop-types";

const Button = ({ children, type = "button", onClick, className }) => {
  return (
    <button
      type={type}
      className={`${`px-4 py-2 rounded-lg text-white transition duration-300 ease-in-out`}`}
      ${className} onClick={onClick}
    >
      {children}
    </button>
  );
};

Button.propTypes = {
  children: PropTypes.node.isRequired,
  type: PropTypes.oneOf(["button", "submit", "reset"]),
  onClick: PropTypes.func,
  className: PropTypes.string,
};

Button.defaultProps = {
  type: "button",
  onClick: () => {},
  className: "",
};

export default Button;
```



```
// INPUT COMPONENT (Input.js)  ==> src/components/atoms/Input.js

import React from "react";

const Input = ({ type = "text", id, placeholder, value, onChange, required, ...props }) => {
  return (
    <input
      type={type}
      id={id}
      placeholder={placeholder}
      value={value}
      onChange={onChange}
      required={required}
      className="w-full px-4 py-2 mt-1 border rounded-lg focus:outline-none focus:ring focus:ring-blue-300"
      {...props}
    />
  );
};

export default Input;
```



```
// LABEL COMPONENT (Label.js)  ==> src/components/atoms/Label.js

import React from "react";

const Label = ({ htmlFor, children }) => {
  return (
    <label htmlFor={htmlFor} className="block text-sm font-medium text-gray-700"> {children}
      </label>
  );
}

export default Label;
```



```
// FORM COMPONENT (Form.js) ==> src/components/molecules/Form.js

import React from "react";
import Label from "../atoms/Label";
import Input from "../atoms/Input";
import Button from "../atoms/Button";

const Form = ({ onSubmit }) => {
  const handleSubmit = (event) => {
    event.preventDefault();
    if (onSubmit) {
      onSubmit();
    }
  };
  return (
    <form onSubmit={handleSubmit} className="space-y-4">
      <div>
        <Label htmlFor="email">Email</Label>
        <Input type="email" id="email" placeholder="Masukkan email" />
      </div>
      <div>
        <Label htmlFor="password">Password</Label>
        <Input type="password" id="password" placeholder="Masukkan password" />
      </div>
      <Button type="submit" className="w-full bg-blue-600 hover:bg-blue-700">
        Login
      </Button>
    </form>
  );
};

export default Form;
```



```
// FormMahasiswa.js => src/components/molecules/FormMahasiswa.js

import React, { useState, useEffect } from "react";
import Label from "../atoms/Label";
import Input from "../atoms/Input";
import Button from "../atoms/Button";
import PropTypes from "prop-types";

const FormMahasiswa = ({ onSubmit, editData, onClose }) => {
  const [formData, setFormData] = useState({
    nim: "",
    nama: "",
    status: true,
  });

  useEffect(() => {
    if (editData) {
      setFormData({
        nim: editData.nim || "",
        nama: editData.nama || "",
        status: typeof editData.status === "boolean" ? editData.status : true,
      });
    } else {
      resetForm();
    }
  }, [editData]);

  const handleChange = (e) => {
    const { id, value } = e.target;
    setFormData((prev) => ({
      ...prev,
      [id]: id === "status" ? value === "true" : value,
    }));
  };

  const handleSubmit = (e) => {
    e.preventDefault();
    const { nim, nama } = formData;

    if (!nim || !nama) {
      alert("NIM dan Nama harus diisi!");
      return;
    }

    if (nim.length < 4) {
      alert("NIM minimal 4 karakter");
      return;
    }

    onSubmit(formData);
    resetForm();
  };

  const resetForm = () => {
    setFormData({ nim: "", nama: "", status: true });
  };

  const handleCancel = () => {
    resetForm();
    if (onClose) onClose();
  };

  return (
    <form onSubmit={handleSubmit} className="p-4 space-y-4">
      <div>
        <Label htmlFor="nim">NIM</Label>
        <Input
          type="text"
          id="nim"
          value={formData.nim}
          onChange={handleChange}
          required
        />
      </div>
      <div>
        <Label htmlFor="nama">Nama</Label>
        <Input
          type="text"
          id="nama"
          value={formData.nama}
          onChange={handleChange}
          required
        />
      </div>
      <div>
        <Label htmlFor="status">Status</Label>
        <select
          id="status"
          className="w-full border rounded p-2"
          value={formData.status ? "true" : "false"}
          onChange={handleChange}
        >
          <option value="true">Aktif</option>
          <option value="false">Tidak Aktif</option>
        </select>
      </div>
      <div className="flex justify-end space-x-2 mt-4">
        <Button
          type="button"
          className="bg-gray-400 text-white"
          onClick={handleCancel}
        >
          Batal
        </Button>
        <Button type="submit" className="bg-blue-600 hover:bg-blue-700">
          {editData ? "Simpan Perubahan" : "Simpan"}
        </Button>
      </div>
    </form>
  );
};

FormMahasiswa.propTypes = {
  onSubmit: PropTypes.func.isRequired,
  onClose: PropTypes.func,
  editData: PropTypes.shape({
    nim: PropTypes.string,
    nama: PropTypes.string,
    status: PropTypes.bool,
  }),
};

export default FormMahasiswa;
```



```
// MODAL COMPONENT (Modal.js) ==> src/components/molecules/Modal.js

import React from "react";

const Modal = ({ isOpen, onClose, children }) => {
  if (!isOpen) return null;
  return (
    <div className="fixed inset-0 flex items-center justify-center bg-black bg-opacity-50"> <div className="bg-white rounded-lg shadow-lg w-full max-w-md p-4">
      {/* Header Modal */}
      <div className="flex justify-between items-center mb-3">
        <h2 className="text-lg font-semibold">Edit Mahasiswa</h2>
        <button
          className="text-gray-600 hover:text-red-500 text-2xl"
          onClick={onClose}
        >
          &times;
        </button>
      </div>
      {/* Isi Modal */}
      {children}
    </div>
  </div>
);
};

export default Modal;
```



```
// Form component (Form.js) ==> src/components/molecules/Form.js

import { useState } from "react";
import { FaUserCircle, FaSignOutAlt, FaUser } from "react-icons/fa";

const ProfileMenu = () => {
  const [isOpen, setIsOpen] = useState(false);

  const handleLogout = () => {
    alert("Logout berhasil!");
    setIsOpen(false);
    window.location.href = "/Login";
  };

  return (
    <div className="relative inline-block text-left">
      <div>
        <button
          onClick={() => setIsOpen(!isOpen)}
          className="flex items-center gap-2 focus:outline-none"
        >
          <FaUserCircle className="text-3xl text-gray-600 hover:text-gray-800 transition" />
        </button>
      </div>

      {isOpen && (
        <div className="origin-top-right absolute right-0 mt-2 w-56 rounded-lg shadow-lg bg-white ring-1 ring-black ring-opacity-5 animate-fade-in">
          <div className="px-4 py-3 border-b">
            <p className="text-sm font-medium text-gray-900">Admin</p>
            <p className="text-xs text-gray-500">admin@gmail.com</p>
          </div>
          <div className="py-1">
            <a
              href="#"
              className="flex items-center gap-2 px-4 py-2 text-sm text-gray-700 hover:bg-gray-100"
            >
              <FaUser className="text-gray-500" /> Profile
            </a>
            <button
              onClick={handleLogout}
              className="flex w-full items-center gap-2 px-4 py-2 text-sm text-gray-700 hover:bg-gray-100"
            >
              <FaSignOutAlt className="text-gray-500" /> Logout
            </button>
          </div>
        </div>
      )}
    </div>
  );
};

export default ProfileMenu;
```



```
// FOOTER COMPONENT (Footer.js)  ==> src/components/organisms/Footer.js

import React from "react";

const Footer = () => {
  return (
    <footer className="bg-white text-center py-4 shadow-inner">
      <p className="text-sm text-gray-600">© 2025 Admin Dashboard. All rights reserved.
    </p></footer>
  );
};

export default Footer;
```



```
// HEADER COMPONENT (Header.js)  ==> src/components/organisms/Header.js

import React from "react";
import ProfileMenu from "../molecules/ProfileMenu";

const Header = () => {
  return (
    <header className="bg-white shadow-md">
      <div className="flex justify-between items-center px-6 py-4">
        <h1 className="text-2xl font-semibold text-gray-800">Mahasiswa</h1>
        <ProfileMenu />
      </div>
    </header>
  );
};

export default Header;
```



```
// SIDEBAR COMPONENT (Sidebar.js) ==> src/components/organisms/Sidebar.js

import React from "react";
import { NavLink } from "react-router-dom";

const Sidebar = () => {
  return (
    <aside className="w-64 bg-blue-800 text-white p-4">
      <h2 className="text-2xl font-bold mb-8 text-white">Admin Panel</h2>
      <nav className="flex flex-col gap-2">
        <NavLink
          to="/admin/dashboard"
          className={({ isActive }) =>
            isActive
              ? "flex items-center gap-2 bg-blue-600 text-white px-4 py-2 rounded"
              : "flex items-center gap-2 hover:bg-blue-500 hover:text-white px-4 py-2 rounded"
          }
        >
           Dashboard
        </NavLink>

        <NavLink
          to="/admin/mahasiswa"
          className={({ isActive }) =>
            isActive
              ? "flex items-center gap-2 bg-blue-600 text-white px-4 py-2 rounded"
              : "flex items-center gap-2 hover:bg-blue-500 hover:text-white px-4 py-2 rounded"
          }
        >
           Mahasiswa
        </NavLink>
      </nav>
    </aside>
  );
};

export default Sidebar;
```



```
// ADMIN LAYOUT (AdminLayout.js) => src/components/templates/AdminLayout.js

import React from "react";
import { Outlet } from "react-router-dom";
import Sidebar from "../organisms/Sidebar"; // Pindahkan komponen Sidebar ke file terpisah jika
//import Header from "../organisms/Header";

const AdminLayout = () => {
  return (
    <div className="flex h-screen">
      <Sidebar />
      <div className="flex-1 flex flex-col">
        <Header />
        <main className="flex-1 p-6 overflow-y-auto bg-gray-100">
          <Outlet />
        </main>
        <footer className="bg-white text-center py-4 shadow-inner">
          <p className="text-sm text-gray-600">
            © 2025 Admin Dashboard. All rights reserved.
          </p>
        </footer>
      </div>
    </div>
  );
};

export default AdminLayout;
```



```
// AUTH LAYOUT COMPONENT (AuthLayout.js)  ==>
src/components/templates/AuthLayout.js
import React from "react";
import { Outlet } from "react-router-dom";

const AuthLayout = () => {
  return (
    <div className="h-screen bg-gray-100 flex items-center justify-center">
      <div className="w-full max-w-md bg-white rounded-xl shadow-lg p-8 mx-4">
        <Outlet />
      </div>
    </div>
  );
};

export default AuthLayout;
```



```
// PROTECTED ROUTE (Route yang Dilindungi) (ProtectedRoute.js) ===> src/components/ProtectedRoute.js

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

const ProtectedRoute = ({ children }) => {
  const isLoggedIn = localStorage.getItem("isLoggedIn");

  // ✅ Di sinilah <Navigate /> digunakan:
  if (!isLoggedIn) {
    return <Navigate to="/login" replace />;
  }

  return children;
};

export default ProtectedRoute;
```



```
// USER DATA (DUMMY DATA) (user.js) ==> src/data/user.js

const dummyUser = {
    email: "admin@gmail.com",
    password: "admin123",
};

export default dummyUser;
```



```
// Mahasiswa.js => src/pages/Mahasiswa.js

import React, { useState, useEffect } from "react";
import Button from "../components/atoms/Button";
import MahasiswaModal from "../components/organisms/MahasiswaModal";
import MahasiswaTable from "../components/organisms/MahasiswaTable";

const Mahasiswa = () => {
  const [mahasiswa, setMahasiswa] = useState([]);
  const [selectedMahasiswa, setSelectedMahasiswa] = useState(null);
  const [isModalOpen, setModalOpen] = useState(false);

  // Load data dari localStorage saat pertama kali
  useEffect(() => {
    const saved = localStorage.getItem("mahasiswa");
    if (saved) {
      setMahasiswa(JSON.parse(saved));
    } else {
      const dummy = [
        { nim: "A11.2022.14283", nama: "Eva Fadhillah Ulia", status: true },
        { nim: "A11.2022.14284", nama: "Sofian Roy Irawan", status: true },
        { nim: "A11.2022.14285", nama: "Elzian Deva Alzarif", status: false },
        { nim: "A11.2022.14286", nama: "Wartini", status: false },
      ];
      setMahasiswa(dummy);
      localStorage.setItem("mahasiswa", JSON.stringify(dummy));
    }
  }, []);

  // Simpan ke localStorage saat mahasiswa berubah
  useEffect(() => {
    localStorage.setItem("mahasiswa", JSON.stringify(mahasiswa));
  }, [mahasiswa]);

  const storeMahasiswa = (data) => {
    setMahasiswa([...mahasiswa, data]);
  };

  const updateMahasiswa = (updated) => {
    const updatedList = mahasiswa.map((mhs) =>
      mhs.nim === updated.nim ? updated : mhs
    );
    setMahasiswa(updatedList);
  };

  const deleteMahasiswa = (nim) => {
    const confirmDelete = window.confirm(
      `Yakin ingin menghapus mahasiswa dengan NIM ${nim}?`
    );
    if (confirmDelete) {
      setMahasiswa(mahasiswa.filter((m) => m.nim !== nim));
    }
  };

  const openAddModal = () => {
    setSelectedMahasiswa(null);
    setModalOpen(true);
  };

  const openEditModal = (mhs) => {
    setSelectedMahasiswa(mhs);
    setModalOpen(true);
  };

  const handleSubmit = (data) => {
    if (selectedMahasiswa) {
      updateMahasiswa(data);
    } else {
      storeMahasiswa(data);
    }
    setModalOpen(false);
  };

  const handleDelete = (nim) => {
    deleteMahasiswa(nim);
  };

  return (
    <div className="bg-white shadow rounded-lg p-4">
      <div className="flex justify-between items-center mb-4">
        <h2 className="text-lg font-semibold">Daftar Mahasiswa</h2>
        <Button
          className="bg-blue-600 hover:bg-blue-700"
          onClick={openAddModal}
        >
          + Tambah Mahasiswa
        </Button>
      </div>

      <MahasiswaTable
        mahasiswa={mahasiswa}
        openEditModal={openEditModal}
        onDelete={handleDelete}
      />

      <MahasiswaModal
        isOpen={isModalOpen}
        onClose={() => setModalOpen(false)}
        onSubmit={handleSubmit}
        selectedMahasiswa={selectedMahasiswa}
      />
    </div>
  );
}

export default Mahasiswa;
```

```

// MahasiswaModal.js => src/components/organisms/MahasiswaModal.js

import React, { useEffect, useState } from "react";
import PropTypes from "prop-types";

const MahasiswaModal = ({ isOpen, selectedMahasiswa, onSubmit, onClose }) => {
  const [form, setForm] = useState({
    nim: "",
    nama: "",
    status: "true",
  });

  // useEffect: handle isi form berdasarkan selectedMahasiswa
  useEffect(() => {
    if (selectedMahasiswa) {
      setForm({
        nim: selectedMahasiswa.nim || "",
        nama: selectedMahasiswa.nama || "",
        status: selectedMahasiswa.status ? "true" : "false",
      });
    } else {
      setForm({
        nim: "",
        nama: "",
        status: "true",
      });
    }
  }, [selectedMahasiswa]);

  // Handle input
  const handleChange = (e) => {
    setForm({ ...form, [e.target.name]: e.target.value });
  };

  // Form Validasi + Submit
  const handleSubmit = (e) => {
    e.preventDefault();

    // Validasi dasar
    if (!form.nim || !form.nama) {
      alert("NIM dan Nama harus diisi!");
      return;
    }

    if (!/\d+/.test(form.nim)) {
      alert("NIM harus berupa angka!");
      return;
    }

    if (form.nim.length < 8) {
      alert("NIM minimal 8 digit!");
      return;
    }

    onSubmit({
      nim: form.nim,
      nama: form.nama,
      status: form.status === "true",
    });
    onClose();
  };
}

// Jika modal tidak dibuka
if (!isOpen) return null;

return (
  <div className="fixed inset-0 bg-black bg-opacity-50 flex justify-center items-center z-50"> <div className="bg-white p-6 rounded shadow-md w-full max-w-md">
    <h2 className="text-xl font-bold mb-4">Form Mahasiswa</h2>
    <form onSubmit={handleSubmit} className="space-y-4">
      <div>
        <label className="block mb-1 font-medium">NIM</label>
        <input
          type="text"
          name="nim"
          value={form.nim}
          onChange={handleChange}
          className="w-full border px-3 py-2 rounded"
          required
        />
      </div>
      <div>
        <label className="block mb-1 font-medium">Nama</label>
        <input
          type="text"
          name="nama"
          value={form.nama}
          onChange={handleChange}
          className="w-full border px-3 py-2 rounded"
          required
        />
      </div>
      <div>
        <label className="block mb-1 font-medium">Status</label>
        <select
          name="status"
          value={form.status}
          onChange={handleChange}
          className="w-full border px-3 py-2 rounded"
        >
          <option value="true">Aktif</option>
          <option value="false">Tidak Aktif</option>
        </select>
      </div>
      <div className="flex justify-end space-x-2">
        <button
          type="button"
          onClick={onClose}
          className="px-4 py-2 bg-gray-300 rounded hover:bg-gray-400"
        >
          Batal
        </button>
        <button
          type="submit"
          className="px-4 py-2 bg-blue-600 text-white rounded hover:bg-blue-700"
        >
          Simpan
        </button>
      </div>
    </form>
  </div>
)
};

MahasiswaModal.propTypes = {
  isOpen: PropTypes.bool.isRequired,
  selectedMahasiswa: PropTypes.shape({
    nim: PropTypes.string,
    nama: PropTypes.string,
    status: PropTypes.bool,
  }),
  onSubmit: PropTypes.func.isRequired,
  onClose: PropTypes.func.isRequired,
};

export default MahasiswaModal;

```

```
// MahasiswaTable.js => src/components/organisms/MahasiswaTable.js

import React from "react";
import PropTypes from "prop-types";

const MahasiswaTable = ({  
    mahasiswa = [],  
    onDelete,  
    openEditModal,  
    totalPages = 1,  
    currentPage = 1,  
    paginate,  
}) => {  
    const handleDelete = (nim) => {  
        if (window.confirm("Yakin ingin menghapus data ini?")) {  
            onDelete(nim);  
        }  
    };  
  
    return (  
        <div className="overflow-x-auto mt-6">  
            <table className="w-full text-sm text-left text-gray-700 border border-gray-300 rounded-md">  
                <thead className="bg-gray-200 text-gray-800">  
                    <tr>  
                        <th className="px-4 py-2 border">No</th>  
                        <th className="px-4 py-2 border">NIM</th>  
                        <th className="px-4 py-2 border">Nama</th>  
                        <th className="px-4 py-2 border">Status</th>  
                        <th className="px-4 py-2 border text-center">Aksi</th>  
                    </tr>  
                </thead>  
                <tbody>  
                    {mahasiswa.length === 0 ? (  
                        <tr>  
                            <td colSpan="5" className="text-center py-4 text-gray-500">  
                                Tidak ada data mahasiswa  
                            </td>  
                        </tr>  
                    ) : (  
                        mahasiswa.map((mhs, index) => (  
                            <tr key={mhs.nim} className="hover:bg-gray-50">  
                                <td className="px-4 py-2 border">  
                                    {(currentPage - 1) * mahasiswa.length + index + 1}  
                                </td>  
                                <td className="px-4 py-2 border">{mhs.nim}</td>  
                                <td className="px-4 py-2 border">{mhs.nama}</td>  
                                <td className="px-4 py-2 border">  
                                    {mhs.status ? "Aktif" : "Tidak Aktif"}  
                                </td>  
                                <td className="px-4 py-2 border text-center space-x-2">  
                                    <button  
                                        onClick={() => openEditModal(mhs)}  
                                        className="px-3 py-1 bg-yellow-500 text-white rounded hover:bg-yellow-600">  
                                        Edit  
                                    </button>  
                                    <button  
                                        onClick={() => handleDelete(mhs.nim)}  
                                        className="px-3 py-1 bg-red-500 text-white rounded hover:bg-red-600">  
                                        Hapus  
                                    </button>  
                                </td>  
                            </tr>  
                        ))  
                    )}  
                </tbody>  
            </table>  
        </div>  
    );  
  
    MahasiswaTable.propTypes = {  
        mahasiswa: PropTypes.arrayOf(  
            PropTypes.shape({  
                nim: PropTypes.string.isRequired,  
                nama: PropTypes.string.isRequired,  
                status: PropTypes.bool.isRequired,  
            })  
,  
        onDelete: PropTypes.func.isRequired,  
        openEditModal: PropTypes.func.isRequired,  
        totalPages: PropTypes.number,  
        currentPage: PropTypes.number,  
        paginate: PropTypes.func,  
    };  
  
    export default MahasiswaTable;
```



```
// Dashboard.js => src/pages/Dashboard.js

import React, { useEffect, useState } from "react";
import { BarChart, Bar, XAxis, YAxis, Tooltip, ResponsiveContainer } from "recharts";

const Dashboard = () => {
  const [jumlahMahasiswa, setJumlahMahasiswa] = useState(0);
  const [jumlahAktif, setJumlahAktif] = useState(0);

  useEffect(() => {
    const data = JSON.parse(localStorage.getItem("mahasiswa")) || [];
    setJumlahMahasiswa(data.length);
    setJumlahAktif(data.filter((mhs) => mhs.status).length);
  }, []);

  const dataChart = [
    { name: "Total", value: jumlahMahasiswa },
    { name: "Aktif", value: jumlahAktif },
    { name: "Non-Aktif", value: jumlahMahasiswa - jumlahAktif },
  ];

  return (
    <div className="bg-white p-6 rounded-xl shadow-md">
      <h1 className="text-3xl font-bold text-gray-800 mb-6">Dashboard Admin</h1>
      <p className="text-gray-600 mb-8">Selamat datang! Berikut ringkasan data sistem saat ini.</p>
      {/* Statistik */}
      <div className="grid grid-cols-1 md:grid-cols-3 gap-6 mb-8">
        <div className="bg-blue-100 p-5 rounded-lg shadow hover:shadow-lg transition">
          <h2 className="text-sm font-medium text-blue-700">Jumlah Mahasiswa</h2>
          <p className="text-2xl font-bold text-blue-900 mt-1">{jumlahMahasiswa}</p>
        </div>
        <div className="bg-green-100 p-5 rounded-lg shadow hover:shadow-lg transition">
          <h2 className="text-sm font-medium text-green-700">Mahasiswa Aktif</h2>
          <p className="text-2xl font-bold text-green-900 mt-1">{jumlahAktif}</p>
        </div>
        <div className="bg-red-100 p-5 rounded-lg shadow hover:shadow-lg transition">
          <h2 className="text-sm font-medium text-red-700">Mahasiswa Tidak Aktif</h2>
          <p className="text-2xl font-bold text-red-900 mt-1">{jumlahMahasiswa - jumlahAktif}</p>
        </div>
      </div>
      {/* Grafik */}
      <div className="bg-gray-50 p-4 rounded-lg shadow">
        <h2 className="text-lg font-semibold text-gray-700 mb-4">Grafik Mahasiswa</h2>
        <ResponsiveContainer width="100%" height={250}>
          <BarChart data={dataChart}>
            <XAxis dataKey="name" />
            <YAxis allowDecimals={false} />
            <Tooltip />
            <Bar dataKey="value" fill="#3b82f6" radius={[8, 8, 0, 0]} />
          </BarChart>
        </ResponsiveContainer>
      </div>
    );
};

export default Dashboard;
```



```
// LOGIN PAGE (Login.js) => src/pages/Login.js

import React from "react";
import { useNavigate } from "react-router-dom";
import Label from "../components/atoms/Label";
import Input from "../components/atoms/Input";
import Button from "../components/atoms/Button";
import user from "../data/user";

const Login = () => {
  const navigate = useNavigate();

  const handleLogin = (event) => {
    event.preventDefault();
    const email = event.target.email.value;
    const password = event.target.password.value;

    // Cek apakah sesuai dengan dummy user
    if (email === user.email && password === user.password) {
      localStorage.setItem("isLoggedIn", "true"); // LocalStorage untuk menyimpan status login
      alert("Login Berhasil! ✅");
      navigate("/admin");
    } else {
      alert("Email atau password salah ❌");
    }
  };

  return (
    <>
      <h2 className="text-3xl font-semibold text-center text-blue-600 mb-6">
        Login
      </h2>
      <form onSubmit={handleLogin} className="space-y-4">
        <div>
          <Label htmlFor="email">Email</Label>
          <Input type="email" id="email" name="email" placeholder="Masukkan email" required />
        </div>
        <div>
          <Label htmlFor="password">Password</Label>
          <Input type="password" id="password" name="password" placeholder="Masukkan password" required />
        </div>
        <Button type="submit" className="w-full bg-blue-600 hover:bg-blue-700">
          Login
        </Button>
      </form>
    </>
  );
};

export default Login;
```

```
// Mahasiswa.js => src/pages/Mahasiswa.js

import React, { useState, useEffect } from "react";
import { useNavigate } from "react-router-dom";
import toast from "react-hot-toast";
import Button from "../components/atoms/Button";
import Modal from "../components/molecules/Modal";
import FormMahasiswa from "../components/molecules/FormMahasiswa";

const Mahasiswa = () => {
  const [isModalOpen, setModalOpen] = useState(false);
  const [mahasiswaList, setMahasiswaList] = useState([]);
  const [selectedMahasiswa, setSelectedMahasiswa] = useState(null);
  const [searchTerm, setSearchTerm] = useState("");
  const [currentPage, setCurrentPage] = useState(1);
  const itemsPerPage = 5;

  const navigate = useNavigate();

  useEffect(() => {
    const savedData = localStorage.getItem("mahasiswa");
    if (savedData && JSON.parse(savedData).length > 0) {
      setMahasiswaList(JSON.parse(savedData));
    } else {
      const dummyData = [
        { nim: "A11.2022.14283", nama: "Eva Fadhillah", status: true },
        { nim: "A11.2022.14284", nama: "Sofiyah Roy Irawan", status: true },
        { nim: "A11.2022.14285", nama: "Elzian Deva Alzarif", status: false },
        { nim: "A11.2022.14286", nama: "Wartini", status: false },
        { nim: "A11.2022.14287", nama: "Reza Erwia", status: false },
      ];
      setMahasiswaList(dummyData);
      localStorage.setItem("mahasiswa", JSON.stringify(dummyData));
    }
  }, []);

  useEffect(() => {
    localStorage.setItem("mahasiswa", JSON.stringify(mahasiswaList));
  }, [mahasiswaList]);

  const storeMahasiswa = (newData) => {
    if (!newData.nim || !newData.nama) {
      toast.error("NIM dan Nama harus diisi!");
      return;
    }

    const exist = mahasiswaList.find((mhs) => mhs.nim === newData.nim);
    if (exist) {
      toast.error("NIM sudah digunakan!");
      return;
    }

    setMahasiswaList([...mahasiswaList, newData]);
    setModalOpen(false);
    toast.success("Data mahasiswa berhasil ditambahkan!");
  };

  const updateMahasiswa = (updatedData) => {
    if (!updatedData.nim || !updatedData.nama) {
      toast.error("NIM dan Nama harus diisi!");
      return;
    }

    const isSameNIM = selectedMahasiswa?.nim === updatedData.nim;
    const exist = mahasiswaList.find(
      (mhs) => mhs.nim === updatedData.nim && !isSameNIM
    );
    if (exist) {
      toast.error("NIM sudah digunakan oleh mahasiswa lain!");
      return;
    }

    const confirmUpdate = window.confirm(`Yakin ingin mengupdate data mahasiswa NIM ${selectedMahasiswa?.nim}?`);
    if (!confirmUpdate) return;

    setMahasiswaList(
      mahasiswaList.map((mhs) =>
        mhs.nim === selectedMahasiswa.nim ? updatedData : mhs
      )
    );
    setModalOpen(false);
    setSelectedMahasiswa(null);
    toast.success("Data mahasiswa berhasil diupdate!");
  };

  const deleteMahasiswa = (nim) => {
    const mhs = mahasiswaList.find((m) => m.nim === nim);
    const confirmDelete = window.confirm(`Yakin ingin menghapus data mahasiswa "${mhs?.nama}" (NIM: ${nim})?`);
    if (confirmDelete) {
      setMahasiswaList(mahasiswaList.filter((item) => item.nim !== nim));
      toast.success("Data mahasiswa berhasil dihapus!");
    }
  };

  const openAddModal = () => {
    setSelectedMahasiswa(null);
    setModalOpen(true);
  };

  const openEditModal = (mhs) => {
    const confirmEdit = window.confirm(`Edit data mahasiswa "${mhs.nama}"?`);
    if (confirmEdit) {
      setSelectedMahasiswa(mhs);
      setModalOpen(true);
    }
  };

  const handleSubmit = (data) => {
    if (selectedMahasiswa) {
      updateMahasiswa(data);
    } else {
      storeMahasiswa(data);
    }
  };

  const filteredMahasiswa = mahasiswaList.filter((mhs) =>
    mhs.nama.toLowerCase().includes(searchTerm.toLowerCase()) ||
    mhs.nim.toLowerCase().includes(searchTerm.toLowerCase())
);

  const indexOfLastItem = currentPage * itemsPerPage;
  const indexOfFirstItem = indexOfLastItem - itemsPerPage;
  const currentMahasiswa = filteredMahasiswa.slice(indexOfFirstItem, indexOfLastItem);
  const totalPages = Math.ceil(filteredMahasiswa.length / itemsPerPage);

  const handlePrevious = () => {
    if (currentPage > 1) setCurrentPage(currentPage - 1);
  };

  const handleNext = () => {
    if (currentPage < totalPages) setCurrentPage(currentPage + 1);
  };

  return (
    <div className="bg-white shadow rounded-lg p-6">
      <div className="flex flex-col md:flex-row justify-between items-center mb-4 gap-4">
        <h2 className="text-xl font-bold text-gray-800">Daftar Mahasiswa</h2>
        <div className="flex gap-2 w-full md:w-auto">
          <input
            type="text"
            placeholder="Cari nama atau NIM..."
            value={searchTerm}
            onChange={(e) => {
              setSearchTerm(e.target.value);
              setCurrentPage(1);
            }}
            className="w-full md:w-64 border border-gray-300 rounded-lg px-4 py-2 focus:outline-none focus:ring-2 focus:ring-blue-500"
          />
          <Button
            className="bg-blue-600 hover:bg-blue-700"
            onClick={openAddModal}
          >
            + Tambah Mahasiswa
          </Button>
        </div>
      </div>
      <div className="overflow-x-auto">
        <table className="w-full text-sm text-gray-700">
          <thead className="bg-blue-600 text-white">
            <tr>
              <th className="py-3 px-4 text-left">NIM</th>
              <th className="py-3 px-4 text-left">Nama</th>
              <th className="py-3 px-4 text-left">Status</th>
              <th className="py-3 px-4 text-center">Aksi</th>
            </tr>
          </thead>
          <tbody>
            {currentMahasiswa.length === 0 ? (
              <tr>
                <td colSpan="4" className="text-center py-6 text-gray-500">
                  Tidak ada data mahasiswa yang sesuai
                </td>
              </tr>
            ) : (
              currentMahasiswa.map((mhs) => (
                <tr key={mhs.nim} className="even:bg-gray-100 odd:bg-white">
                  <td className="py-2 px-4 font-medium">{mhs.nim}</td>
                  <td className="py-2 px-4">{mhs.nama}</td>
                  <td className="py-2 px-4">
                    <span className="px-3 py-1 rounded-full text-xs font-semibold ${mhs.status ? 'bg-green-100 text-green-700' : 'bg-red-100 text-red-700'}">
                      {mhs.status ? "Aktif" : "Tidak Aktif"}
                    </span>
                  </td>
                  <td className="py-2 px-4 text-center space-x-2">
                    <Button
                      className="bg-blue-500 hover:bg-blue-600"
                      onClick={() => navigate(`/admin/mahasiswa/${mhs.nim}`)}
                    >
                      Detail
                    </Button>
                    <Button
                      className="bg-yellow-500 hover:bg-yellow-600"
                      onClick={() => openEditModal(mhs)}
                    >
                      Edit
                    </Button>
                    <Button
                      className="bg-red-500 hover:bg-red-600"
                      onClick={() => deleteMahasiswa(mhs.nim)}
                    >
                      Hapus
                    </Button>
                  </td>
                </tr>
              ))
            )}
          </tbody>
        </table>
      </div>
      <div className="flex flex-col md:flex-row justify-between items-center mt-6 gap-4">
        <p className="text-sm text-gray-600 text-center md:text-left">
          Menampilkan {indexOfFirstItem + 1} - {Math.min(indexOfLastItem, filteredMahasiswa.length)} dari {filteredMahasiswa.length} mahasiswa
        </p>
        <div className="flex flex-col items-center gap-2">
          <Button
            className={`${px-4 py-2 rounded-full ${
              currentPage === 1
                ? "bg-gray-200 text-gray-400 cursor-not-allowed"
                : "bg-blue-600 text-white hover:bg-blue-700"
            }}`}
            onClick={handlePrevious}
            disabled={currentPage === 1}
          >
            - Previous
          </Button>
          <Button
            className={`${px-4 py-2 rounded-full ${
              currentPage === totalPages || totalPages === 0
                ? "bg-gray-200 text-gray-400 cursor-not-allowed"
                : "bg-blue-600 text-white hover:bg-blue-700"
            }}`}
            onClick={handleNext}
            disabled={currentPage === totalPages || totalPages === 0}
          >
            Next -
          </Button>
        </div>
        <span className="text-sm text-gray-600">
          Halaman {currentPage} dari {totalPages}
        </span>
      </div>
    </div>
  <Modal isOpen={isModalOpen} onClose={() => setModalOpen(false)}>
    <FormMahasiswa
      onSubmit={handleSubmit}
      editData={selectedMahasiswa}
    />
  </Modal>
</div>
);
};

export default Mahasiswa;
```

```
// App.js => src/App.js

import React from "react";
import { Routes, Route, Navigate } from "react-router-dom";
import { Toaster } from "react-hot-toast"; // ✓ import Toaster

import Login from "./pages/Login";
import Dashboard from "./pages/Dashboard";
import Mahasiswa from "./pages/Mahasiswa";
import AuthLayout from "./components/templates/AuthLayout";
import AdminLayout from "./components/templates/AdminLayout";
import ProtectedRoute from "./components/ProtectedRoute";
import MahasiswaDetail from "./pages/MahasiswaDetail";

const App = () => {
  return (
    <>
      {/* ✓ Ini komponen untuk menampilkan toast */}
      <Toaster
        position="top-right"
        toastOptions={{
          duration: 3000,
          style: {
            background: "#fff",
            color: "#333",
            fontWeight: "500",
            borderRadius: "8px",
            boxShadow: "0 4px 12px rgba(0, 0, 0, 0.1)",
          },
        }}
    />

    {/* Routing */}
    <Routes>
      {/* Halaman Login */}
      <Route element={<AuthLayout />}>
        <Route path="/login" element={<Login />} />
      </Route>

      {/* Halaman Admin (Nested Routes) */}
      <Route
        path="/admin"
        element={
          <ProtectedRoute>
            <AdminLayout />
          </ProtectedRoute>
        }
      >
        {/* Nested Admin Routes */}
        <Route index element={<Navigate to="dashboard" />} />
        <Route path="dashboard" element={<Dashboard />} />
        <Route path="mahasiswa" element={<Mahasiswa />} />
        <Route path="mahasiswa/:id" element={<MahasiswaDetail />} />
      </Route>
    </Routes>
  );
};

export default App;
```



```
// MahasiswaDetail.js => src/pages/MahasiswaDetail.js

import React from "react";
import { useParams, useLocation, useNavigate } from "react-router-dom";
import Button from "../components/atoms/Button";
import { User, BadgeCheck, CheckCircle, XCircle } from "lucide-react";

const MahasiswaDetail = () => {
  const { id } = useParams();
  const { state } = useLocation();
  const navigate = useNavigate();

  const data = state;

  return (
    <div className="max-w-xl mx-auto mt-10 p-6 bg-white shadow-xl rounded-2xl border border-gray-200">
      <h1 className="text-3xl font-extrabold text-center text-blue-700 mb-6">
        Detail Mahasiswa
      </h1>

      {data ? (
        <div className="space-y-5 text-gray-800 text-[16px]">
          <div className="flex items-center gap-4">
            <BadgeCheck className="text-blue-600" />
            <div>
              <p className="text-gray-500 text-sm">NIM</p>
              <p className="font-semibold">{data.nim}</p>
            </div>
          </div>
        </div>

        <div className="flex items-center gap-4">
          <User className="text-blue-600" />
          <div>
            <p className="text-gray-500 text-sm">Nama</p>
            <p className="font-semibold">{data.nama}</p>
          </div>
        </div>
      ) : (
        <div className="flex items-center gap-4">
          {data.status ? (
            <CheckCircle className="text-green-600" />
          ) : (
            <XCircle className="text-red-600" />
          )}
          <div>
            <p className="text-gray-500 text-sm">Status</p>
            <span
              className={`inline-block px-3 py-1 text-sm rounded-full font-medium ${data.status
                ? "bg-green-100 text-green-700"
                : "bg-red-100 text-red-700"
              }`}
            >
              {data.status ? "Aktif" : "Tidak Aktif"}
            </span>
          </div>
        </div>
      ) : (
        <div className="text-red-600 font-medium text-center">
          Data mahasiswa tidak tersedia atau tidak dikirim 😞
        </div>
      )}
    </div>
  );
};

export default MahasiswaDetail;
```



```
// INDEX (Index.js) - Entry point of the React application

import React from "react";
import ReactDOM from "react-dom/client";
import { BrowserRouter as Router } from "react-router-dom";
import App from "./App";
import "./index.css"; // Import Tailwind CSS

const root =
ReactDOM.createRoot(document.getElementById("root"));
  <React.StrictMode>
    <Router>
      <App />
    </Router>
  </React.StrictMode>
);
```