

✓ OPEN EDITORS

✓ TUGAS7

✓ public

✓ src

✓ Components

JS Button.js

JS Footer.js

JS Form.js

JS FormMahasiswa.js

JS Header.js

JS Input.js

JS Label.js

JS Modal.js

JS ProfileMenu.js

JS Sidebar.js

✓ Data

JS user.js

✓ Layouts

JS AdminLayout.js

JS AuthLayout.js

✓ Pages

✓ Admin

JS Dashboard.js

✓ Auth

JS Login.js

✓ Mahasiswa

JS Mahasiswa.js

JS MahasiswaModal.js

JS MahasiswaTable.js

JS MahasiswaDetail.js

✓ Routes

JS ProtectedRoute.js

✓ Utils/Helpers

JS SwalHelper.js

JS ToastHelper.js

JS App.js

# index.css

JS index.js

◆ .gitignore

{ } package-lock.json

## Login

Email

admin@gmail.com

Password

.....

Login

## Admin Panel

Dashboard

Mahasiswa

## Mahasiswa

Login Berhasil! Selamat datang!

### Dashboard Admin

Selamat datang! Berikut ringkasan data sistem saat ini.

Jumlah Mahasiswa

5

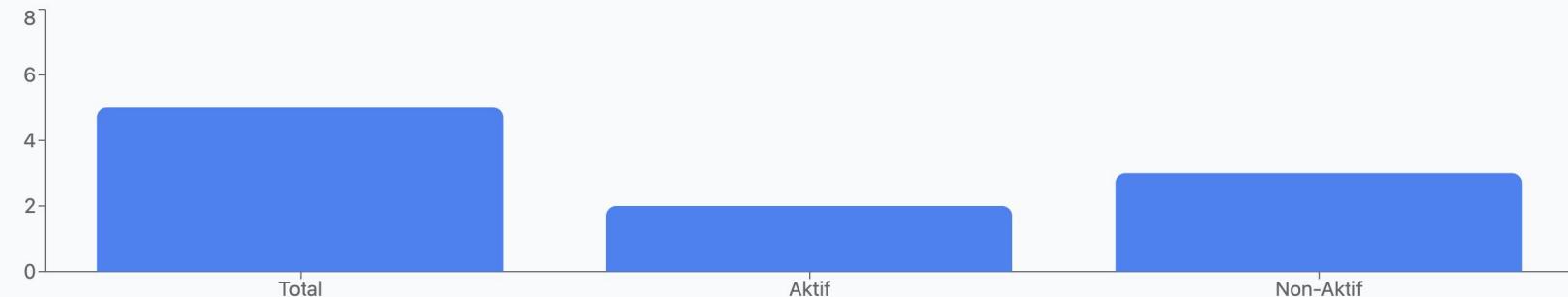
Mahasiswa Aktif

2

Mahasiswa Tidak Aktif

3

#### Grafik Mahasiswa



Mahasiswa

Carbon | Create and share be...

Dashboard

localhost:3000/Login

Apply Activity | Me... Digital Talent Scho... (154) Belajar Java... Home | AWS Back... Telegram Evafadhillahulia at... Apple Developer iOS Academy - Yo... WhatsApp Learn Swift Progra... All Bookmarks

 **Email atau password salah. Coba lagi ya!**

## Login

Email

Password

**Login**

## Dashboard Admin

Selamat datang! Berikut ringkasan data sistem saat ini.

Jumlah Mahasiswa

**5**

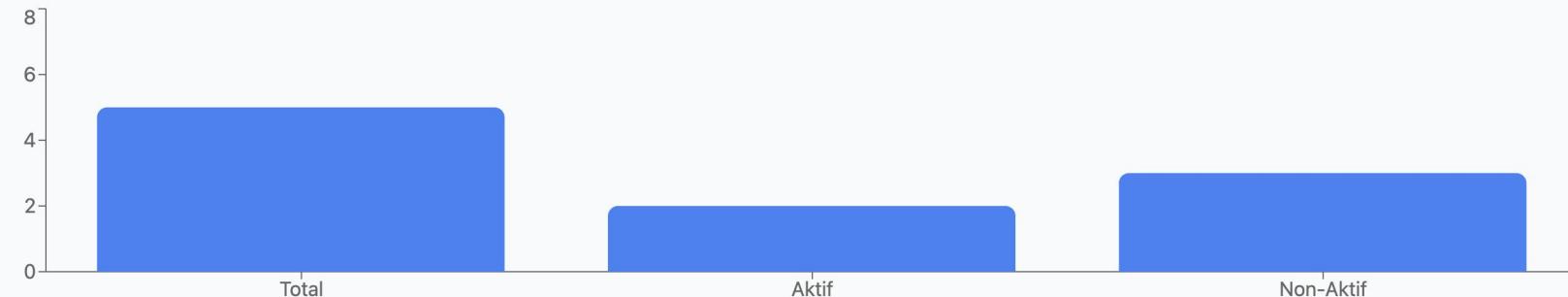
Mahasiswa Aktif

**2**

Mahasiswa Tidak Aktif

**3**

### Grafik Mahasiswa



[Dashboard](#)[Mahasiswa](#)

## Daftar Mahasiswa

[+ Tambah Mahasiswa](#)

NIM	Nama	Status	Aksi
A11.2022.14283	Eva Fadhillah Ulia	Aktif	<a href="#">Detail</a> <a href="#">Edit</a> <a href="#">Hapus</a>
A11.2022.14284	Sofyan Roy Irawan	Aktif	<a href="#">Detail</a> <a href="#">Edit</a> <a href="#">Hapus</a>
A11.2022.14285	Elzian Deva Alzarif	Tidak Aktif	<a href="#">Detail</a> <a href="#">Edit</a> <a href="#">Hapus</a>
A11.2022.14286	Wartini	Tidak Aktif	<a href="#">Detail</a> <a href="#">Edit</a> <a href="#">Hapus</a>
A11.2022.14287	Reza Erwia Wardani	Tidak Aktif	<a href="#">Detail</a> <a href="#">Edit</a> <a href="#">Hapus</a>

Menampilkan 1 - 5 dari 5 mahasiswa //

[Previous](#)[Next](#)

Halaman 1 dari 1

Apply Activity | Me...

Digital Talent Scho...

(154) Belajar Java...

Home | AWS Back...

Telegram

Evafadhillahulia at...

Apple Developer

iOS Academy - Yo...

WhatsApp

Learn Swift Progra...

&gt;&gt;

All Bookmarks

## Admin Panel

Dashboard

Mahasiswa

## Mahasiswa



## Daftar Mahasiswa

Cari nama atau NIM...

+ Tambah Mahasiswa

NIM	Nama	Status	Aksi
A11.2022.14283	Eva Fadhillah Ulia	Aktif	<button>Detail</button> <button>Edit</button> <button>Hapus</button>
A11.2022.14284	Sof		<button>Detail</button> <button>Edit</button> <button>Hapus</button>
A11.2022.14285	Elz		<button>Detail</button> <button>Edit</button> <button>Hapus</button>
A11.2022.14286	Wa		<button>Detail</button> <button>Edit</button> <button>Hapus</button>
A11.2022.14287	Re		<button>Detail</button> <button>Edit</button> <button>Hapus</button>

Menampilkan 1 - 5 dari 5 mahasiswa //

Previous Next

Halaman 1 dari 1

## Tambah Mahasiswa

NIM

A11.2022.14288

Nama

Halda Triyani

Status

Aktif

Batal

Simpan

Mahasiswa

localhost:3000/admin/mahasiswa

Apply Activity | Me... Digital Talent Scho... (154) Belajar Java... Home | AWS Back... Telegram Evafadhillahulia at... Apple Developer iOS Academy - Yo... WhatsApp Learn Swift Progra... All Bookmarks

## Admin Panel

Dashboard Mahasiswa

### Mahasiswa

Data mahasiswa berhasil ditambahkan!

#### Daftar Mahasiswa

NIM	Nama	Status	Aksi
A11.2022.14283	Eva Fadhillah Ulia	Aktif	<button>Detail</button> <button>Edit</button> <button>Hapus</button>
A11.2022.14284	Sofiyan 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 Wardani	Tidak Aktif	<button>Detail</button> <button>Edit</button> <button>Hapus</button>

Cari nama atau NIM... + Tambah Mahasiswa

Menampilkan 1 - 5 dari 6 mahasiswa //

Halaman 1 dari 2

© 2025 Admin Dashboard. All rights reserved.

Mahasiswa

Carbon | Create and share be...

localhost:3000/admin/mahasiswa

Apply Activity | Me... Digital Talent Scho... (154) Belajar Java... Home | AWS Back... Telegram Evafadhillahulia at... Apple Developer iOS Academy - Yo... WhatsApp Learn Swift Progra... All Bookmarks

Admin Panel

Dashboard Mahasiswa

## Mahasiswa

### Daftar Mahasiswa

NIM	Nama	Status	Aksi
A11.2022.14288	Halda Triyani	Aktif	<a href="#">Detail</a> <a href="#">Edit</a> <a href="#">Hapus</a>

Cari nama atau NIM... + Tambah Mahasiswa

Menampilkan 6 - 6 dari 6 mahasiswa //

Previous Next Halaman 2 dari 2

© 2025 Admin Dashboard. All rights reserved.

[Dashboard](#)[Mahasiswa](#)

## Daftar Mahasiswa

[+ Tambah Mahasiswa](#)

NIM	Nama	Status	Aksi
A11.2022.14288	Halda Triyani	Aktif	<a href="#">Detail</a> <a href="#">Edit</a> <a href="#">Hapus</a>

Menampilkan 6 - 6 dari 6 mahasiswa //

[Previous](#)[Next](#)

Halaman 2 dari 2



## Konfirmasi Edit

Edit data mahasiswa "Halda Triyani"?

[Ya, Edit!](#)[Batal](#)

[Dashboard](#)[Mahasiswa](#)

## Daftar Mahasiswa

[+ Tambah Mahasiswa](#)

NIM	Nama	Status	Aksi
-----	------	--------	------

A11.2022.14288

Halda Triyani

Aktif

[Detail](#)[Edit](#)[Hapus](#)

Menampilkan 6 - 6 dari 6 mahasiswa //

[Previous](#)[Next](#)

Halaman 2 dari 2

## Edit Mahasiswa



NIM

A11.2022.14289

Nama

Halda Triyanida

Status

Aktif

[Batal](#)[Simpan Perubahan](#)

[Dashboard](#)[Mahasiswa](#)

## Daftar Mahasiswa

[+ Tambah Mahasiswa](#)

NIM	Nama	Status	Aksi
A11.2022.14288	Halda Triyani	Aktif	<a href="#">Detail</a> <a href="#">Edit</a> <a href="#">Hapus</a>

Menampilkan 6 - 6 dari 6 mahasiswa //

[Previous](#) [Next](#)

Halaman 2 dari 2



## Konfirmasi Update

Yakin ingin mengupdate data mahasiswa NIM  
A11.2022.14288?

[Ya, Update!](#) [Batal](#)

Mahasiswa

Carbon | Create and share be...

localhost:3000/admin/mahasiswa

Apply Activity | Me... Digital Talent Scho... (154) Belajar Java... Home | AWS Back... Telegram Evafadhillahulia at... Apple Developer iOS Academy - Yo... WhatsApp Learn Swift Progra... All Bookmarks

Admin Panel

Dashboard Mahasiswa

## Mahasiswa

✓ Data mahasiswa berhasil diupdate!

### Daftar Mahasiswa

NIM	Nama	Status	Aksi
A11.2022.14289	Halda Triyanida	Aktif	<a href="#">Detail</a> <a href="#">Edit</a> <a href="#">Hapus</a>

Menampilkan 6 - 6 dari 6 mahasiswa //

[Previous](#) [Next](#)

Halaman 2 dari 2

© 2025 Admin Dashboard. All rights reserved.

## Mahasiswa

## Daftar Mahasiswa

Cari nama atau NIM...

+ Tambah Mahasiswa

NIM	Nama	Status	Aksi
A11.2022.14289	Halda Triyanida	Aktif	<button>Detail</button> <button>Edit</button> <button>Hapus</button>

Menampilkan 6 - 6 dari 6 mahasiswa //

Previous

Next

Halaman 2 dari 2

[Dashboard](#)[Mahasiswa](#)

## Daftar Mahasiswa

[+ Tambah Mahasiswa](#)

NIM	Nama	Status	Aksi
A11.2022.14289	Halda Triyanida	Aktif	<a href="#">Detail</a> <a href="#">Edit</a> <a href="#">Hapus</a>

Menampilkan 6 - 6 dari 6 mahasiswa //

[Previous](#)[Next ➔](#)

Halaman 2 dari 2



## Konfirmasi Hapus

Yakin ingin menghapus data mahasiswa "Halda Triyanida" (NIM: A11.2022.14289)?

[Ya, Hapus!](#)[Batal](#)

Mahasiswa

localhost:3000/admin/mahasiswa

Apply Activity | Me... Digital Talent Scho... (154) Belajar Java... Home | AWS Back... Telegram Evafadhillahulia at... Apple Developer iOS Academy - Yo... WhatsApp Learn Swift Progra... All Bookmarks

Admin Panel

Dashboard Mahasiswa

## Mahasiswa

✓ Data mahasiswa berhasil dihapus!

### Daftar Mahasiswa

NIM	Nama	Status	Aksi
Tidak ada data mahasiswa yang sesuai			

Menampilkan 6 - 5 dari 5 mahasiswa //

Cari nama atau NIM... + Tambah Mahasiswa

Previous Next

Halaman 2 dari 1

© 2025 Admin Dashboard. All rights reserved.

Mahasiswa

Carbon | Create and share be... X +

localhost:3000/admin/mahasiswa

Apply Activity | Me... Digital Talent Scho... (154) Belajar Java... Home | AWS Back... Telegram Evafadhillahulia at... Apple Developer iOS Academy - Yo... WhatsApp Learn Swift Progra... All Bookmarks

Admin Panel

Dashboard Mahasiswa

Mahasiswa

Daftar Mahasiswa

Cari nama atau NIM...

Admin admin@gmail.com

Profile Logout

NIM

Nama

Status

Aksi

Aktif

Detail Edit Hapus

Aktif

Detail Edit Hapus

Detail Edit Hapus

Detail Edit Hapus

Detail Edit Hapus

Menampilkan 1 - 5 dari 5 mahasiswa //

Yakin ingin logout?

Ya, logout Batal

Halaman 1 dari 1

© 2025 Admin Dashboard. All rights reserved.

NIM	Nama	Status	Aksi
A11.2022.14283	Eva Fadhillah Ulia	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			<button>Detail</button> <button>Edit</button> <button>Hapus</button>
A11.2022.14286			<button>Detail</button> <button>Edit</button> <button>Hapus</button>
A11.2022.14287			<button>Detail</button> <button>Edit</button> <button>Hapus</button>

!

Yakin ingin logout?

Ya, logout Batal



```
// button.js => src/Components/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;
```



```
// Footer.js => src/Components/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;
```

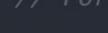


```
// Form.js => src/Components/Form.js

import React from "react";
import Label from "../Components/Label";
import Input from "../Components/Input";
import Button from "../Components/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/FormMahasiswa.js

import React, { useState, useEffect } from "react";
import Label from "../Components/Label";
import Input from "../Components/Input";
import Button from "../Components/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;
```



```
// Header.js => src/Components/Header.js

import React from "react";
import ProfileMenu from "../Components/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;
```



```
// Input.js => src/Components/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.js => src/Components/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;
```



```
// Modal.js => src/Components/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">
      <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>
      {children}
    </div>
  </div>
);
};

export default Modal;
```



```
// ProfileMenu.js => src/Components/ProfileMenu.js

import { useState } from "react";
import { FaUserCircle, FaSignOutAlt, FaUser } from "react-icons/fa";
import { confirmLogout } from "../Utils/Helpers/SwalHelper";

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

  const handleLogout = () => {
    confirmLogout(() => {
      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;
```



```
// Sidebar.js => src/Components/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;
```



```
// user.js => src/Data/user.js

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

export const dummyMahasiswa = [
  { nim: "A11.2022.14283", nama: "Eva Fadhillah Ulia", 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 Wardani", status: false },
];
```



```
// AdminLayout.js => src/Layouts/AdminLayout.js

import React from "react";
import { Outlet } from "react-router-dom";
import Sidebar from "../Components/Sidebar";
import Header from "../Components/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;
```



```
// AuthLayout.js => src/Layouts/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;
```



```
// Dashboard.js => src/Pages/Admin/Dashboard.js

import React, { useEffect, useState } from "react";
// Import komponen dari Recharts untuk membuat grafik batang
import { BarChart, Bar, XAxis, YAxis, Tooltip, ResponsiveContainer } from "recharts";

// Komponen utama Dashboard
const Dashboard = () => {
    // State untuk menyimpan jumlah total mahasiswa
    const [jumlahMahasiswa, setJumlahMahasiswa] = useState(0);
    // State untuk menyimpan jumlah mahasiswa yang aktif
    const [jumlahAktif, setJumlahAktif] = useState(0);

    // useEffect dijalankan setelah komponen dirender pertama kali
    useEffect(() => {
        // Ambil data mahasiswa dari localStorage, jika tidak ada defaultnya array kosong
        const data = JSON.parse(localStorage.getItem("mahasiswa")) || [];
        // Set jumlah total mahasiswa berdasarkan panjang data
        setJumlahMahasiswa(data.length);
        // Hitung jumlah mahasiswa yang aktif (status = true)
        setJumlahAktif(data.filter((mhs) => mhs.status).length);
    }, []);

    // Data untuk grafik batang
    const dataChart = [
        { name: "Total", value: jumlahMahasiswa }, // Jumlah total mahasiswa
        { name: "Aktif", value: jumlahAktif }, // Mahasiswa aktif
        { name: "Non-Aktif", value: jumlahMahasiswa - jumlahAktif } // Mahasiswa non-aktif
    ];

    return (
        // Container utama dengan background putih, padding, dan tampilan melengkung (rounded)
        <div className="bg-white p-6 rounded-xl shadow-md">
            {/* Judul halaman dashboard */}
            <h1 className="text-3xl font-bold text-gray-800 mb-6">Dashboard Admin</h1>
            {/* Deskripsi singkat dashboard */}
            <p className="text-gray-600 mb-8">Selamat datang! Berikut ringkasan data sistem saat ini.</p>
            {/* Bagian statistik */}
            <div className="grid grid-cols-1 md:grid-cols-3 gap-6 mb-8">
                {/* Kartu jumlah mahasiswa total */}
                <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>
                {/* Kartu mahasiswa aktif */}
                <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>
                {/* Kartu mahasiswa tidak aktif */}
                <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>
            {/* Bagian grafik batang */}
            <div className="bg-gray-50 p-4 rounded-lg shadow">
                <h2 className="text-lg font-semibold text-gray-700 mb-4">Grafik Mahasiswa</h2>
                {/* Container responsif agar grafik menyesuaikan ukuran layar */}
                <ResponsiveContainer width="100%" height={250}>
                    {/* Grafik batang dengan data dari dataChart */}
                    <BarChart data={dataChart}>
                        {/* Sumbu X menggunakan properti 'name' */}
                        <XAxis dataKey="name" />
                        {/* Sumbu Y tanpa angka desimal */}
                        <YAxis allowDecimals={false} />
                        {/* Tooltip untuk menampilkan detail saat hover */}
                        <Tooltip />
                        {/* Batang dengan nilai 'value' dan warna biru, ujung atas membulat */}
                        <Bar dataKey="value" fill="#3b82f6" radius={[8, 8, 0, 0]} />
                    </BarChart>
                </ResponsiveContainer>
            </div>
        );
};

export default Dashboard;
```



```
// Login.js => src/Pages/Auth/Login.js

import React from "react";
import { useNavigate } from "react-router-dom";
import { toast } from "react-hot-toast";
import Label from "../../Components/Label";
import Input from "../../Components/Input";
import Button from "../../Components/Button";
import { dummyUser } 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 login dengan dummy user
    if (email === dummyUser.email && password === dummyUser.password) {
      localStorage.setItem("isLoggedIn", "true");

      toast.success(" Login Berhasil! Selamat datang!", {
        duration: 3000,
        position: "top-center",
        style: {
          background: "#4ade80", // hijau
          color: "#1e3a8a", // biru tua
          fontWeight: "bold",
          padding: "16px",
          borderRadius: "12px",
          fontSize: "16px",
        },
        icon: "✅",
      });
      navigate("/admin");
    } else {
      toast.error("❌ Email atau password salah. Coba lagi ya!", {
        duration: 3000,
        position: "top-center",
        style: {
          background: "#fecaca", // merah muda
          color: "#7f1d1d", // merah tua
          fontWeight: "bold",
          padding: "16px",
          borderRadius: "12px",
          fontSize: "16px",
        },
        icon: "⚠️",
      });
    }
  };

  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/Mahasiswa.js
```

```
import React, { useState, useEffect } from "react";
import { useNavigate } from "react-router-dom";
import toast from "react-hot-toast"; // Mengimpor library untuk menampilkan toast notifikasi
import Swal from "sweetalert2"; // Mengimpor SweetAlert2 untuk menampilkan dialog konfirmasi
import Button from "../../Components/Button";
import Modal from "../../Components/Modal";
import FormMahasiswa from "../../Components/FormMahasiswa";
import { dummyMahasiswa } from "../../Data/user"; // Import data dummy
```

```
const Mahasiswa = () => {
  const [isModalOpen, setModalOpen] = useState(false); // State untuk membuka dan menutup modal
  const [mahasiswaList, setMahasiswaList] = useState([]); // State untuk menyimpan daftar mahasiswa
  const [selectedMahasiswa, setSelectedMahasiswa] = useState(null); // State untuk menyimpan mahasiswa yang sedang dipilih untuk edit
  const [searchTerm, setSearchTerm] = useState(""); // State untuk menyimpan kata pencarian
  const [currentPage, setCurrentPage] = useState(1); // State untuk menyimpan halaman yang sedang aktif
  const itemsPerPage = 5; // Menentukan jumlah mahasiswa yang ditampilkan per halaman
```

```
  const navigate = useNavigate();
```

```
  useEffect(() => { // Mengambil data mahasiswa dari localStorage atau menggunakan data dummy jika tidak ada data
    const savedData = localStorage.getItem("mahasiswa");
    if (savedData && JSON.parse(savedData).length > 0) {
      setMahasiswaList(JSON.parse(savedData)); // Mengecek apakah data mahasiswa ada di localStorage
    } else { // Mengambil data mahasiswa dari localStorage
      setMahasiswaList(dummyMahasiswa); // Jika tidak ada, gunakan data dummy
      localStorage.setItem("mahasiswa", JSON.stringify(dummyMahasiswa)); // Menyimpan data dummy ke localStorage
    }
  }, []);
```

```
  useEffect(() => { // Menyimpan daftar mahasiswa ke localStorage setiap kali daftar mahasiswa berubah
    localStorage.setItem("mahasiswa", JSON.stringify(mahasiswaList));
  }, [mahasiswaList]);
```

```
  const storeMahasiswa = (newData) => { // Fungsi untuk menyimpan data mahasiswa baru
    if (!newData.nim || !newData.nama) { // Mengecek apakah NIM dan Nama diisi
      showTopCenterToast("NIM dan Nama harus diisi!"); // Menampilkan pesan error jika NIM atau Nama tidak diisi
      return;
    }

    const exist = mahasiswaList.find((mhs) => mhs.nim === newData.nim); // Mengecek apakah NIM sudah ada di daftar mahasiswa
    if (exist) {
      showTopCenterToast("NIM sudah digunakan!"); // Menampilkan pesan jika NIM sudah ada
      return;
    }

    setMahasiswaList([...mahasiswaList, newData]); // Menambahkan mahasiswa baru ke dalam daftar mahasiswa
    setModalOpen(false); // Menutup modal setelah data berhasil ditambahkan
    showTopCenterToast("Data mahasiswa berhasil ditambahkan!"); // Menampilkan toast notifikasi sukses
  };

```

```
  const showTopCenterToast = (message, type = "success") => { // Fungsi untuk menampilkan toast notifikasi
    toast.custom(
      () => (
        <div
          className="flex items-center gap-3"
          style={{
            background: type === "success" ? "#4ade80" : type === "error" ? "#f87171" : "#fcfd3d",
            color: "#1e3a8a", // biru tua
            fontWeight: "bold",
            padding: "16px",
            borderRadius: "12px",
            fontSize: "16px",
            boxShadow: "0 4px 14px rgba(0,0,0,0.15)",
          }}
        >
          <span>{type === "success" ? "✓" : type === "error" ? "✗" : "⚠"}</span>
          <span>{message}</span> /* Menampilkan pesan toast */
        </div>
      ),
      {
        position: "top-center", // Menampilkan toast di bagian atas tengah layar
        duration: 3000,
      }
    );
  };
};

const updateMahasiswa = async (updatedData) => { // Fungsi untuk memperbarui data mahasiswa
  if (!updatedData.nim || !updatedData.nama) { // Mengecek apakah NIM dan Nama diisi
    showTopCenterToast("NIM dan Nama harus diisi!");
    return;
  }

  const isSameNIM = selectedMahasiswa?.nim === updatedData.nim; // Mengecek apakah NIM yang diedit adalah NIM yang sama
  const exist = mahasiswaList.find(
    (mhs) => mhs.nim === updatedData.nim && !isSameNIM // Mengecek apakah NIM sudah ada di mahasiswa lain
  );
  if (exist) {
    showTopCenterToast("NIM sudah digunakan oleh mahasiswa lain!");
    return;
  }

  const result = await Swal.fire({ // Menampilkan SweetAlert untuk konfirmasi update data
    title: "Konfirmasi Update",
    text: `Yakin ingin mengupdate data mahasiswa NIM ${selectedMahasiswa?.nim}?`,
    icon: "question",
    showCancelButton: true,
    confirmButtonColor: "#3085d6",
    cancelButtonColor: "#d3d",
    confirmButtonText: "Ya, Update!",
    cancelButtonText: "Batal",
  });

  if (result.isConfirmed) { // Jika pengguna mengonfirmasi, update daftar mahasiswa
    setMahasiswaList(
      mahasiswaList.map((mhs) =>
        mhs.nim === selectedMahasiswa.nim ? updatedData : mhs
      )
    );
    setModalOpen(false);
    setSelectedMahasiswa(null); // Menghapus mahasiswa yang dipilih untuk diedit
    showTopCenterToast("Data mahasiswa berhasil diupdate!");
  }
};

const deleteMahasiswa = async (nim) => { // Fungsi untuk menghapus data mahasiswa
  const mhs = mahasiswaList.find((m) => m.nim === nim); // Mencari mahasiswa berdasarkan NIM

  const result = await Swal.fire({ // Menampilkan SweetAlert untuk konfirmasi hapus data
    title: "Konfirmasi Hapus",
    text: `Yakin ingin menghapus data mahasiswa "${mhs?.nama}" (NIM: ${nim})?`,
    icon: "warning",
    showCancelButton: true,
    confirmButtonColor: "#e3342f",
    cancelButtonColor: "#6c757d",
    confirmButtonText: "Ya, Hapus!",
    cancelButtonText: "Batal",
  });

  if (result.isConfirmed) { // Jika pengguna mengonfirmasi, hapus mahasiswa dari daftar
    setMahasiswaList(mahasiswaList.filter((item) => item.nim !== nim));
    showTopCenterToast("Data mahasiswa berhasil dihapus!");
  }
};

const openAddModal = () => { // Fungsi untuk membuka modal tambah mahasiswa
  setSelectedMahasiswa(null);
  setModalOpen(true);
};

const openEditModal = async (mhs) => { // Fungsi untuk membuka modal edit mahasiswa
  const result = await Swal.fire({
    title: "Konfirmasi Edit",
    text: `Edit data mahasiswa "${mhs?.nama}"?`,
    icon: "question",
    showCancelButton: true,
    confirmButtonColor: "#f59e0b",
    cancelButtonColor: "#6c757d",
    confirmButtonText: "Ya, Edit!",
    cancelButtonText: "Batal",
  });

  if (result.isConfirmed) {
    setSelectedMahasiswa(mhs);
    setModalOpen(true);
  }
};

const handleSubmit = (data) => { // Fungsi untuk menangani pengiriman data dari form
  if (selectedMahasiswa) {
    updateMahasiswa(data);
  } else {
    storeMahasiswa(data);
  }
};

const filteredMahasiswa = mahasiswaList.filter((mhs) => // Mencari mahasiswa berdasarkan NIM atau
  mhs.nama.toLowerCase().includes(searchTerm.toLowerCase()) ||
  mhs.nim.toLowerCase().includes(searchTerm.toLowerCase())
);

const indexOfLastItem = currentPage * itemsPerPage; // Menghitung indeks item terakhir
const indexOfFirstItem = indexOfLastItem - itemsPerPage; // Menghitung indeks item pertama
const currentMahasiswa = filteredMahasiswa.slice(indexOfFirstItem, indexOfLastItem); // Mengambil mahasiswa yang ditampilkan pada halaman saat ini
const totalPages = Math.ceil(filteredMahasiswa.length / itemsPerPage);

const handlePrevious = () => { // Fungsi untuk menangani tombol Previous
  if (currentPage > 1) setCurrentPage(currentPage - 1);
};

const handleNext = () => { // Fungsi untuk menangani tombol Next
  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);
          }}
        >
        <button
          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"
          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 ? ( // Jika tidak ada mahasiswa yang sesuai dengan pencarian
            <tr>
              <td colSpan="4" className="text-center py-6 text-gray-500">
                Tidak ada data mahasiswa yang sesuai
              </td>
            </tr>
          ) : (
            currentMahasiswa.map((mhs) => ( // Menampilkan daftar mahasiswa
              <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="bg-green-700 text-green-700 rounded-full px-1 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}`, { state: mhs })}
                  >
                    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>
  <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)} 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;
```



```
// MahasiswaModal.js => src/Pages/Mahasiswa/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/Pages/Mahasiswa/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;
```



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

import React from "react";
import { useParams, useLocation, useNavigate } from "react-router-dom";
import Button from "../Components/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;
```



```
// ProtectedRoute.js => src/Routes/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;
```



```
// SwalHelper.js => src/Utils/Helpers/SwalHelper.js

import Swal from "sweetalert2";

const showConfirm = (title, icon, confirmButtonText, successMessage, onConfirm) => {
  Swal.fire({
    title,
    icon,
    showCancelButton: true,
    confirmButtonText,
    cancelButtonText: "Batal",
  }).then((result) => {
    if (result.isConfirmed) {
      onConfirm();
      Swal.fire({
        icon: "success",
        title: successMessage,
        showConfirmButton: false,
        timer: 1500,
      });
    }
  });
};

export const confirmLogout = (onConfirm) => {
  showConfirm("Yakin ingin logout?", "warning", "Ya, logout", "Logout berhasil", onConfirm);
};

export const confirmDelete = (onConfirm) => {
  showConfirm("Yakin ingin menghapus data ini?", "warning", "Ya, hapus", "Data berhasil dihapus", onConfirm);
};

export const confirmUpdate = (onConfirm) => {
  showConfirm("Yakin ingin memperbarui data ini?", "question", "Ya, perbarui", "Data berhasil diperbarui", onConfirm);
};
```



```
// ToastHelper.js => src/Utils/Helpers/ToastHelper.js

import { toast } from "react-hot-toast";

const toastOption = {
  duration: 3000,
  position: "top-right",
};

export const toastSuccess = (message) => toast.success(message,
  toastOption);
export const toastError = (message) => toast.error(message, toastOption);
export const toastInfo = (message) => toast(message, toastOption);
```



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

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

import Login from "./Pages/Auth/Login";
import Dashboard from "./Pages/Admin/Dashboard";
import Mahasiswa from "./Pages/Mahasiswa/Mahasiswa";
import AuthLayout from "./Layouts/AuthLayout";
import AdminLayout from "./Layouts/AdminLayout";
import ProtectedRoute from "./Routes/ProtectedRoute";
import MahasiswaDetail from "./Pages/MahasiswaDetail";

const App = () => {
  return (
    <>
      <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>

      {/* Fallback route */}
      <Route path="/" element={<Navigate to="/login" />} />
    </Routes>
  </>
);
};

export default App;
```



```
// index.js => src/index.js

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>
);
```