"use client";

import React, { useState } from "react";

import { Input } from "@/components/ui/input";

import { Button } from "@/components/ui/button";

import Image from "next/image";

import Link from "next/link";

import { LoaderIcon } from "lucide-react";

import GlobalApi from "@/app/\_utils/GlobalApi";

import { Label } from "@radix-ui/react-label";

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

import ReCAPTCHA from "react-google-recaptcha";

import { useRouter } from "next/navigation";

import { useAuth } from "@/app/AuthContext";

// Reusable Input Component for NPA PGRI

const NpaPgriInput = ({ value, onChange }) => (

<div className="mb-6">

<Label htmlFor="npaPgri" className="block text-sm">

NPA PGRI

</Label>

<Input

id="npaPgri"

placeholder="123456"

value={value}

onChange={onChange}

className="mt-2 p-3 border border-gray-300 rounded-md focus:outline-none focus:ring-2 focus:ring-blue-500 w-full"

aria-label="NPA PGRI"

/>

</div>

);

// Reusable Input Component for Password (used in Admin tab)

const PasswordInput = ({ value, onChange }) => (

<div className="mb-6">

<Label htmlFor="password" className="block text-sm">

Password

</Label>

<Input

id="password"

placeholder="\*\*\*\*\*\*\*\*"

value={value}

onChange={onChange}

type="password"

className="mt-2 p-3 border border-gray-300 rounded-md focus:outline-none focus:ring-2 focus:ring-blue-500 w-full"

aria-label="Password"

/>

</div>

);

function SignIn() {

const [npaPgri, setNpaPgri] = useState("");

const [tanggalLahir, setTanggalLahir] = useState("");

const [password, setPassword] = useState("");

const [loader, setLoader] = useState(false);

const [error, setError] = useState("");

const [isVerified, setIsVerified] = useState(false);

const [activeTab, setActiveTab] = useState("login");

const router = useRouter();

const { setToken, setUserId } = useAuth();

// Common login logic for both tabs

const handleSignIn = async (loginData, loginMethod) => {

setLoader(true);

setError("");

try {

if (!isVerified) throw new Error("Harap verifikasi reCAPTCHA.");

const response = await loginMethod(loginData);

setToken(response.token);

sessionStorage.setItem("userId", response.id);

sessionStorage.setItem("unitKerja", response.unitKerja);

sessionStorage.setItem("nama", response.namaLengkap);

sessionStorage.setItem("role", response.role);

toast.success(`Selamat Datang ${response.namaLengkap || sessionStorage.getItem("nama")}`);

setTimeout(() => {

router.push("/home");

}, 2000);

} catch (err) {

toast.error(err.message || "Terjadi kesalahan saat login");

} finally {

setLoader(false);

}

};

const onSignIn = async () => {

if (npaPgri.length < 6 || tanggalLahir.length !== 8) {

return toast.error("NPA PGRI harus 6 digit dan Tanggal Lahir harus 8 digit.");

}

const loginData = { npaPgri, tanggalLahir };

handleSignIn(loginData, GlobalApi.login);

};

const onSignInAdmin = async () => {

if (!npaPgri.trim() || !password.trim()) {

return toast.error("NPA PGRI dan Password tidak boleh kosong.");

}

const loginData = { npaPgri: npaPgri.trim(), password: password.trim() };

handleSignIn(loginData, GlobalApi.loginAdmin);

};

const onChange = (value) => {

setIsVerified(!!value);

};

return (

<div className="flex items-baseline justify-center my-9">

<Toaster

toastOptions={{

style: { fontSize: "1.25rem", padding: "16px" },

success: { style: { background: "white", color: "black" } },

error: { style: { background: "#f44336", color: "#fff" } },

}}

/>

<div className="flex flex-col items-center justify-center p-4 sm:p-6 md:p-10 bg-gray-100 border border-gray-200 rounded-lg shadow-md w-full sm:w-[80%] md:w-[60%] lg:w-[40%] mx-auto">

<Image src="/sanduka.png" width={150} height={150} alt="logo" />

<h2 className="font-bold text-2xl sm:text-3xl mt-4">Masuk ke Akun</h2>

<h2 className="text-gray-500 mt-2 text-center">

Masukkan NPA PGRI dan {activeTab === "login" ? "Tanggal Lahir" : "Password"} Anda untuk Masuk

</h2>

{/\* Tab Navigation \*/}

<div className="flex space-x-2 sm:space-x-4 mt-4">

<button

className={`rounded-md p-2 ${activeTab === "login" ? "bg-teal-500 text-white" : "bg-white text-teal-500"}`}

onClick={() => setActiveTab("login")}

>

Anggota

</button>

<button

className={`rounded-md p-2 ${activeTab === "password" ? "bg-teal-500 text-white" : "bg-white text-teal-500"}`}

onClick={() => setActiveTab("password")}

>

Admin & Super Admin

</button>

</div>

<div className="w-full max-w-md mt-6">

{error && <p className="text-red-500 mb-4 text-center">{error}</p>}

{/\* Form sesuai tab yang aktif \*/}

{activeTab === "login" && (

<div>

<NpaPgriInput value={npaPgri} onChange={(e) => setNpaPgri(e.target.value)} />

<div className="mb-6">

<Label htmlFor="tanggalLahir" className="block text-sm">Tanggal Lahir</Label>

<Input

id="tanggalLahir"

placeholder="21082024"

value={tanggalLahir}

onChange={(e) => setTanggalLahir(e.target.value)}

className="mt-2 p-3 border border-gray-300 rounded-md focus:outline-none focus:ring-2 focus:ring-blue-500 w-full"

aria-label="Tanggal Lahir"

/>

</div>

</div>

)}

{activeTab === "password" && (

<div>

<NpaPgriInput value={npaPgri} onChange={(e) => setNpaPgri(e.target.value)} />

<PasswordInput value={password} onChange={(e) => setPassword(e.target.value)} />

</div>

)}

<ReCAPTCHA sitekey="6Lfcxy4qAAAAACy6hmLpVgTejZFZG3xGjn0xOVmd" onChange={onChange} />

<Button

onClick={activeTab === "login" ? onSignIn : onSignInAdmin}

disabled={loader || !isVerified || !npaPgri || (activeTab === "login" && !tanggalLahir) || (activeTab === "password" && !password)}

className="w-full mt-4"

>

{loader ? <LoaderIcon className="animate-spin mr-2" /> : "Masuk"}

</Button>

<p className="mt-4 text-sm text-center text-gray-600">

Belum punya akun?

<Link href={"/create-account/syarat-ketentuan"} className="text-blue-500 ml-1 hover:underline">

Klik di sini untuk membuat akun baru

</Link>

</p>

</div>

</div>

</div>

);

}

export default SignIn;