```jsx
import React, { useContext, useState } from "react";
import { Link, useNavigate } from "react-router-dom";
import { AppContext } from "../context/AppContext";
import { loginUser } from "../proxies/backend_api";
import { emailRegex } from "../utils/helper";

const Login = () => {
  const { setShowAlert, setUser } = useContext(AppContext);

  const navigate = useNavigate();

  const [inputs, setInputs] = useState({
    email: "",
    password: "",
  });

  const [error, setErrors] = useState({
    email: "",
    password: "",
  });

  const handleChange = ({ target: { name, value } }) => {
    setErrors((prev) => {
      return { ...prev, [name]: "" };
    });
    setInputs((prev) => ({ ...prev, [name]: value }));
  };

  const checkInputErrors = () => {
    let status = true;
    if (inputs.email.trim() === "" || !emailRegex.test(inputs.email.trim())) {
      setErrors((prev) => {
        return { ...prev, email: "Enter a valid email" };
      });
      status = false;
    }

    if (inputs.password.trim() === "") {
      setErrors((prev) => {
        return { ...prev, password: "Enter a valid password" };
      });
      status = false;
    }

    if (inputs.password.trim().length < 6) {
      setErrors((prev) => {
        return { ...prev, password: "Minimum 6 characters" };
      });
      status = false;
    }
    return status;
  };

  const handleLogin = async () => {
    if (checkInputErrors()) {
```

```
    const data = await loginUser(inputs);
    if (data.error) {
      setShowAlert({ type: "error", message: data.error, duration: 3000 });
      return;
    }
    setUser(data);
    setShowAlert({
      type: "success",
      message: `Welcome back ${data.name}`,
      duration: 3000,
    });
    localStorage.setItem("user", JSON.stringify(data));
    navigate("/dashboard");
  }
};

return (
  <div className="flex flex-col justify-center items-center gap-10 mt-5">
    <div>
      <button className="bg-base-300 rounded-box flex flex-row justify-evenly items-center gap-10
px-10 py-5 w-fit mx-auto">
        <span>Sign in with Github</span>
        <img src={`github-dark.png`} alt="github" width="14%" />
      </button>
      <div className="divider max-w-xs">or</div>
      <form
      onSubmit={(e) => e.preventDefault()}
      className="card bg-base-300 rounded-box flex flex-col justify-center items-center gap-5 px-10
py-5 w-fit mx-auto"
      >
        <div>
          <input
            value={inputs.email}
            type="text"
            name="email"
            placeholder="email"
            className="input input-bordered input-primary w-full"
            onChange={handleChange}
          />
          {error.email !== "" && (
            <p className="text-sm text-red-500 mt-1 font-medium">
              {error.email}
            </p>
          )}
        </div>
        <div>
          <input
            value={inputs.password}
            type="password"
            name="password"
            placeholder="password"
            className="input input-bordered input-primary w-full"
            onChange={handleChange}
          />
          {error.password !== "" && (
```

```jsx
            <p className="text-sm text-red-500 mt-1 font-medium">
              {error.password}
            </p>
          )}
        </div>
        <div className="text-center">
          <button
            type="submit"
            onClick={handleLogin}
            className="btn btn-sm btn-primary mb-4"
          >
            Login
          </button>
          <p>
            Don't have an account?{" "}
            <Link className="text-blue-400" to="/signup">
              Sign up
            </Link>
          </p>
        </div>
      </form>
    </div>
  </div>
  );
};

export default Login;
```