React Hook Form

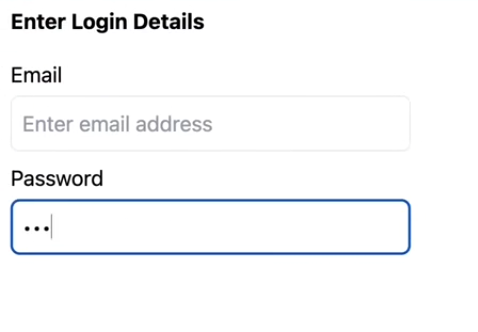
Installation

npm i react-hook-form

এইটা ইন্সটল করার পর দুইটা কম্পোনেন্ট বানাতে হবে-

1. FieldSet
2. Field

FieldSet and Field



* **Step-01**

FieldSet Component

import React from "react";

const FieldSet = ({ label, children }) => {

  return (

    <fieldset className="m-2 border-none p-0">

      {label && <legend className="text-md font-bold mb-2">{label}</legend>}

      <div className="flex flex-col justify-between self-start">{children}</div>

    </fieldset>

  );

};

export default FieldSet;

এইখানে Enter Login Details এবং এর মধ্যে থাকা children হলো Email এবং password দিয়ে থাকা ডিভ টি।

Field Component

import React from "react";

const Field = ({ label, children, htmlFor, error }) => {

  const getChildId = (children) => {

    const child = React.Children.only(children);

    if ("id" in child?.props) {

      return child.props.id;

    }

  };

  const id = htmlFor || getChildId(children);

  return (

    <div className="flex flex-col items-start justify-start mt-2 p-0 w-full mr-2">

      {label && (

        <label className="mb-1" htmlFor={id}>

          {label}

        </label>

      )}

      {children}

      {!!error && <div className="text-red-500">{error.message}</div>}

    </div>

  );

};

export default Field;

Uses of React Hook Form

import { useForm } from "react-hook-form";

const {register,handleSubmit} = useForm()

এইখানে register ব্যবহার করা হয়েছে কারণ input এর যে name আছে এইগুলা অটোমেটিক useForm ষ্টেট হিসেবে হ্যান্ডেল করবে। আমাদের নতুন করে useState ব্যবহার করে এইসব হ্যান্ডেল করতে হবে না।

* **Step-02**

import React from "react";

import React from "react";

import { useForm } from "react-hook-form";

import Field from "./Field";

import FieldSet from "./FieldSet";

const LoginForm = () => {

  const { register, handleSubmit } = useForm();

  const submitForm = (formData) => {

    console.log(formData);

  };

  return (

    <div>

      <form onSubmit={handleSubmit(submitForm)}>

        <FieldSet label={"Enter Login Details"}>

          <Field label={"Email"}>

            <input

              {...register("email")}

              className="p-2 border box-border w-[300px] rounded-md border-gray-200"

              type="email"

              name="email"

              id="email"

              placeholder="Enter Your Email"

            />

          </Field>

          <Field label={"Passowrd"}>

            <input

              {...register("password")}

              className="p-2 border box-border w-[300px] rounded-md border-gray-200"

              type="password"

              name="password"

              id="password"

              placeholder="Enter Your Passowrd"

            />

          </Field>

        </FieldSet>

      </form>

    </div>

  );

};

export default LoginForm;

এইখানে input ফিল্ডে গিয়ে আমরা **{...register("password")}** লিখে দিয়েছি যাতে register state হিসেবে নেয়।

এরপর form এর action এ onSubmit={handleSubmit(submitForm)}> দিয়ে form সাবমিট করে দিলাম

Full Login Form Dynamic With React Hook Form

1. Field.jsx

import React from "react";

const Field = ({ label, children, htmlFor, error }) => {

  // এইখান ফিল্ডের মধ্যে থেকে wrap করে দিলে আমরা লেবেল টা পাব, wrap এর মধ্যে থাকা input form টি children এ পাব  যেইটা আমরা LoginForm এ দেখিয়েছি।

  const id = htmlFor || getChildId(children); // এইখানে htmlFor আসলে আসবে,  না আসলে getChildId ফাংশন থেকে input এর ভেতরে থাকা আইডি পাব।

  return (

    <div className="form-control">

      {label && (

        <label htmlFor={id} className="auth-label">

          {label}

        </label>

      )}

      {children}

      {!!error && (

        <div role="alert" className="text-red-600">

          {error.message}

        </div>

      )}

    </div>

  );

};

const getChildId = (children) => {

  //আইডী পাওয়ার জণ্য

  const child = React.Children.only(children);

  if ("id" in child?.props) {

    return child.props.id;

  }

};

export default Field;

LoginForm.jsx

import React from "react";

import { useForm } from "react-hook-form";

import Field from "../common/Field";

const LoginForm = () => {

  const {

    register,

    handleSubmit,

    formState: { errors },

  } = useForm();

  const submitForm = (formData) => {

    console.log(formData);

  };

  return (

    <form

      class="border-b border-[#3F3F3F] pb-10 lg:pb-[60px]"

      onSubmit={handleSubmit(submitForm)}

    >

      <Field label={"Email"} error={errors.email}>  {/\*এইখানে আমরা Field এর মধ্যে দিয়ে label , error টা পাস করতেছি। কিন্তু এখানে htmlFor নাই এইজন্য এর ভেতরে থাকা children থেকে id আসবে \*/}

        <input                                          {/\*এইখানে আমরা Field এর মধ্যে দিয়ে children টা পাস করতেছি।  \*/}

          {...register("email", { required: "Email is Required" })}

          className={`auth-input ${

            !!errors.email ? "border-red-500" : "border-gray-200"

          }`}

          type="email"

          name="email"

          id="email"

        />

      </Field>

      <Field label={"Password"} error={errors.password}>

        <input

          {...register("password", {

            required: "Password is Required",

            minLength: {

              value: 8,

              message: "Your Password Must be at least 8 characters",

            },

          })}

          className={`auth-input ${

            !!errors.password ? "border-red-500" : "border-gray-200"

          }`}

          type="email"

          name="email"

          id="email"

        />

      </Field>

      <Field>

        <button

          class="auth-input bg-lwsGreen font-bold text-deepDark transition-all hover:opacity-90"

          type="submit"

        >

          Login

        </button>

      </Field>

    </form>

  );

};

export default LoginForm;