**Why?**

[Inversion of Control](https://kentcdodds.com/blog/inversion-of-control) is really cool and [Compound Components](https://kentcdodds.com/blog/inversion-of-control#compound-components) do provide really flexible API.

Yet this library is opinionated, provides bindings for Formik, with necessary paddings etc. You can easily opt-out continue to use Chakra UI if you need custom components. But this library will support %90 of the use-cases when building forms.

**Getting started**

yarn add @chakra-ui/react @emotion/react @emotion/styled framer-motion formik formik-chakra-ui

Or

npm i @chakra-ui/react @emotion/react @emotion/styled framer-motion formik formik-chakra-ui

Than setup providers.

import \* as React from 'react';

// 1. import `ChakraProvider` component

import { ChakraProvider } from '@chakra-ui/react';

function App() {

// 2. Use at the root of your app

return (

<ChakraProvider>

<App />

</ChakraProvider>

);

}

**Documentation**

All documentation can be found [here](https://629c67f66924e51a45fb2029--gregarious-marshmallow-00a41c.netlify.app/) <https://629c67f66924e51a45fb2029--gregarious-marshmallow-00a41c.netlify.app/>

**Example**

[Codesandbox example](https://codesandbox.io/s/formik-chakra-ui-27yzm?file=/src/Form.tsx)

https://codesandbox.io/s/formik-chakra-ui-27yzm?file=/src/Form.tsx

**Roadmap**

**Components**

*  Button
*  Checkbox
*  Input
*  Number Input
*  Pin Input
*  Radio (<RadioGroup/>)
  +  Rendering util function/component for <Radio/>
*  Select
*  Slider
*  Switch
*  Textarea

**Utility Components**

*  Submit button
*  Reset button
*  Form progress