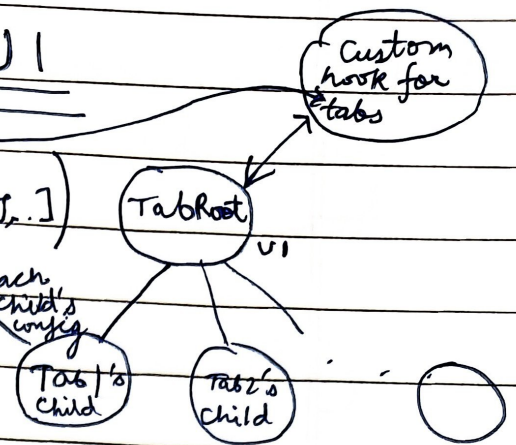


Tab-based FORM-UI

1) Config-driven UI (tabs = [x, y, ...])

```
{  
  name: 'Profile',  
  Component: Profile,  
  validate: () => { ... },  
  validateWithoutSettingState: ...  
}
```

error



- Also Custom Hook will have a state for
- data ~~of~~ all tabs for their fields
 - error of all tabs's fields

2) We have a relative global State in Custom hook (or TabRoot) for all child states.

3) In TabRoot we have a state:

- activeTab for representing current tab & a derived state which has current Component from tabs that we display
- const ActiveComponent = tabs[activeTab].component;
in JSX → <ActiveComponent props />

- 4) Based on events that happen in TabRoot & it's childs we manipulate the states in TabRoot & Custom hook.
- Also basically we might do some things like setting a state directly or might put a condition.
 - We pass states & setters as props to child or we can use useContext or Redux Toolkit.