**Definições**

**Actions:**

Action vai ser a requisição disparada no frontend e executada no backend. Exemplos:

1. Existe um ComboBox (<select />) que deve ser preenchido através de uma consulta no banco de dados. Essa consulta deve ser previamente cadastrada e um ID (de preferencia uma string) deve ser atribuído. Esse ID é a action. Deve existir um endpoint onde eu passe esse ID e o retorno seja o resultado da consulta no formado ID -> Label.
2. Existe um botão no formulário que precisa executar uma função no backend.
3. Salvar uma informação de um grid ou formulário no backend. Ao clicar no botão de salvar o formulário, será enviado uma action para o backend persistir as informações no banco de dados.

**Domínios:**

Domínio vai ser um array de objetos estáticos utilizados para preencher campos de múltiplas escolhas como combobox, radiobutton e checkbox. Lembrando que serão valores estáticos e que não vão ser alterados durante a execução da aplicação. Exemplos:

1. Campos como gênero, sempre serão valores fixos: Feminino ou Masculino.

**Estrutura**

1. **Formulário - Container:**

Assim que o usuário entrar na aplicação, o backend deve me retornar uma lista de formulários liberados para o usuário logado com as seguintes informações:

1. Nome do formulário
2. ID do formulário

Assim que o usuário clicar no formulário, o frontend vai enviar uma requisição solicitando informações do container das seções e quais seções o formulário possui. Na resposta deve vir a seguinte estrutura:

1. Array de seções;
2. Primary Key (Campo que vai disparar a action para buscar as informações do formulário);
3. Action de busca (A ID da action no backend que vai ser responsável por efetuar a busca no banco de dados;
4. Informações gerais que ficam disponíveis para o usuário.

O array de seções deve conter as seguintes informações:

1. O tipo de seção (Se é formulário ou Grid);
2. Array de condições;
3. Array de campos;
4. Se é somente leitura.
5. Caso o tipo de seção for grid, é necessário enviar o ID da action que será executada quando adicionar/remover/editar os valores da grid.

O array de condições deve conter as seguintes informações:

1. A seção que o campo da condição se encontra;
2. O campo para testar a condição;
3. O valor que o campo precisa possuir.

O array de campos está descrito abaixo.

**Campos da seção de formulário:**

Todos os campos devem possuir as seguintes propriedades no JSON:

- Tamanho em colunas (Mínimo 1, Máximo 12)

- Em qual linha se encontra (A soma dos tamanhos dos campos na linha deve ser no máximo 12)

- Posição do campo na linha (Não precisa ser sequencial, desde que seja de ordem crescente)

- O tipo do campo (Combobox, Radiobutton, Checkbox, Input ou TextArea)

- Se está visível ou não

- Se é somente leitura

- Id (Identificador principal do campo, que será enviado ao clicar em salvar)

- Label (Texto informando a descrição do campo)

- Valor padrão do campo

**Campos ComboBox, RadioButton e Checkbox:**

As imagens dos RadioButtons devem ser cadastradas no backend com um ID onde o frontend encontre os respectivos arquivos na pasta de imagens. Os campos serão populados de duas formas:

1. **Domínios:** opções estáticas previamente cadastrados, exemplo:

*Sexo: Feminino / Masculino.*

1. **Actions**: consultas que serão realizadas no banco de dados. No backend deve existir uma action cadastrada com determinado ID. Esse ID (podendo conter parâmetros ou não) vai ser enviado ao backend através de um endpoint e o retorno será as informações necessárias para popular o campo, exemplo:

*Cidades: Opções carregadas de acordo com o UF selecionado. Para carregar será realizada uma requisição para o backend informando o ID: “****carregar\_cidadades****” e o parâmetro UF “****SC****”.*

**Campos de Input:**

1. As máscaras devem ser em forma de caracteres (99/99/9999) ou Regex, essas definições devem vir no JSON.
2. As funções para validar informações como CPF/CNPJ/Email devem ser criadas no frontend com seus respectivos ID’s, assim podem ser reutilizadas em diversos campos dos formulários da aplicação. Esses ID’s devem ser atribuídos no JSON do backend.
3. O tipo de input (texto ou número) deve ser atribuído ao JSON.

Em casos de botão:

1. O botão deve ter um identificador único.
2. Caso a ação do botão for abrir um modal, informar o ID do modal.
3. Caso precise executar uma action no backend, informar a action e um array de parâmetros necessários. O array de parâmetros deve seguir o padrão:
   1. A seção que o campo para parâmetro está;
   2. O ID do campo.
4. A ação do campo pode ser programada via código no frontend.

**Campos da seção de grid:**

Os campos da seção de grid devem vir no mesmo formato dos campos da seção de formulário, incluindo também os campos:

1. Se o campo está visível apenas em modo grid ou no formulário de inserção também.
2. Tamanho em porcentagem da coluna no grid.
3. Se o campo estará disponível para filtragem.
4. Se o campo pode ser ordenado ou não.
5. **Formulário - Modais:**

Assim que o usuário solicitar a abertura de um modal, uma requisição será efetuada ao backend informando o ID do modal, então o processo recomeça no item A.