

Programação Aplicada Ao Direito



U F *m* G



MARCELO CANDIOTTO
Sociedade de Advogados **MCSA**



ASSOCIAÇÃO
BRASILEIRA DE
**LAWTECHS &
LEGALTECHS**



Conteúdo

Dia 06 - Introdução ao Docassemble - Entrevistas

- Entrevista Docassemble
- Blocos da entrevista
- Lógica da entrevista
- Tipos de campos e modificadores
- Exercício Prático: Projeto de Automação do DAJ



O que é uma entrevista no Docassemble?

- é um arquivo YAML, que o Docassemble lê e, com base no que encontra, faz perguntas a um usuário.
- a ordem que as perguntas aparecem na tela nem sempre é a mesma que está no YAML.
- o Docassemble armazena as respostas do usuário em "variáveis".
- os valores dessas variáveis podem ser incorporados no texto das perguntas ou no texto dos documentos.

Blocos da entrevista

Tipos de Blocos:

1. **Bloco inicial** - bloco especial para gravar metadados que afetam toda a entrevista, mas que não aparece para usuário.
2. **Bloco de perguntas** - apresenta uma tela para o usuário com uma pergunta ou uma instrução e os campos para o usuário preencher. Neste bloco podemos ter:
 - Questions e Subquestions (Perguntas e Instruções)
 - Fields (Campos)
 - Modificadores de Campos

Blocos da entrevista

3. Bloco de Impressão - apresenta a última tela da entrevista com o link para download do documento e envio por e-mail.

4. Bloco de Código - são blocos de código em Python. São usados para diversas finalidades, como: definir variáveis, armazenar dados, modificar a ordem da entrevista, dentre outros.

5. Bloco de Features - é um bloco de recursos especiais e opcionais da entrevista, que visam à melhoria da experiência do usuário.

Lógica da Entrevista

Etapas de apresentação dos blocos:

1. Ele encontra a primeira pergunta obrigatória (**mandatory: True**) e tenta fazer a pergunta.
2. Se a pergunta contiver uma variável X não definida, ele procura a pergunta que define a variável X.
3. Assim, ele entra no bloco que contém a variável X e faz a pergunta.
4. Depois, volta à etapa 1. Desta vez, ele é capaz de fazer a pergunta obrigatória.

Lógica da Entrevista

mandatory: True

- Em regra, para o Docassemble mostrar o bloco de perguntas é necessário incluir o comando “mandatory: True”.
- esse pedaço de código é responsável por informar ao Docassemble que a aquele bloco deve ser obrigatoriamente lido pelo sistema.

Criando entrevistas

Exemplo de uma entrevista:

```
metadata:  
| title: Entrevista  
| ---  
question: Qual o seu animal favorito?  
fields:  
| - Animal: animal_favorito  
| ---  
mandatory: True  
question: Que coincidência!  
subquestion: |  
| Meu animal favorito também é ${ animal_favorito } !  
buttons:  
| - Exit: exit  
| ---
```

Bloco inicial

Metadados:

- Um bloco de metadados contém informações sobre a entrevista, como o título, nome do autor.
- Todas as informações incluídas nesse bloco, serão aplicadas para o restante dos blocos de entrevista.

```
---  
metadata:  
  title: |  
    Advice on Divorce  
  short title: |  
    Divorce  
  description: |  
    A divorce advice interview  
  authors:  
    - name: John Doe  
      organization: Example, Inc.  
  revision_date: 2015-09-28  
---
```

Bloco inicial

- Se um **title** for definido, ele será exibido na barra de navegação no aplicativo da web.
- Se um **short title** for fornecido, ele será exibido no lugar do título quando o tamanho da tela for pequeno.
- Ainda posso fornecer uma descrição (**description**), indicar quem são os autores (**authors**) e ainda colocar a última data que o documento foi revisto (**revision_date**).

Bloco de perguntas

Perguntas (questions):

- No Docassemble, todas as telas que um usuário vê são geradas por um bloco de perguntas.
- Para definir a pergunta você deve usar a tag `question`.

```
metadata:  
  title: Bloco de perguntas  
  ---  
question: |  
  Qual o número do CPF do outorgante?  
fields:  
  - Número: cpf_outorgante  
  ---  
mandatory: True  
question: |  
  O número do CPF do outorgante é ${ cpf_outorgante }.  
  ---
```

Bloco de perguntas

Subquestões (subquestions):

- A subquestão é opcional e adiciona texto abaixo da pergunta.
- É normalmente usado para explicar a pergunta com mais detalhes, se tal explicação for necessária.
- Usar tag `subquestion`.

```
metadata:  
  title: Bloco de perguntas com subseção  
  ---  
question: |  
  Qual o número do CPF do outorgante?  
subquestion: |  
  O CPF deverá ser digitado no formato 000.000.000-00  
fields:  
  - Número: cpf_autor  
  ---  
mandatory: True  
question: |  
  O número do seu CPF é ${ cpf_autor }.  
  ---
```

Modificadores de campos

Hint:

- Você pode orientar os usuários sobre como eles devem preencher um campo de texto, mostrando o texto acinzentado em uma caixa de texto que desaparece quando o usuário começa a digitar as informações.

```
---  
mandatory: True  
question: |  
  Hint:  
fields: |  
  - "CPF:": cpf  
    required: True  
    hint: "Ex: 000.000.000-00"  
---
```

Modificadores de campos

Help:

- Você pode fornecer ajuda contextual ao usuário em relação ao significado de um campo. O rótulo do campo ficará verde para indicar que pode ser clicado e o valor da ajuda aparecerá na tela quando o usuário clicar no texto verde.

```
---  
mandatory: True  
question: |  
  Hint:  
fields:|  
  - "CPF:": cpf  
    required: True  
    hint: "Ex: 000.000.000-00"  
---
```

Bloco de perguntas

Campos (fields):

- Permite agrupar informações em uma mesma pergunta. Ex.: campos de endereço.
- Eles podem ser obrigatórios ou optativos.

```
metadata:
  title: Bloco de perguntas com subseção
---
question: |
  Qual o endereço do outorgante?
fields:
  - "Logradouro": outorgante_logradouro
    required: True
  - "Número": outorgante_numero
    required: True
  - "Complemento": outorgante_complemento
    required: False
  - "Bairro": outorgante_bairro
    required: True
  - "Cidade": outorgante_cidade
    required: True
  - "Estado": outorgante_estado
    required: True
  - "CEP": outorgante_cep
    required: True
---
mandatory: True
question: |
  0 endereço do outorgante é ${ outorgante_logradouro }, ${
    outorgante_numero }, ${ outorgante_complemento }, Bairro ${
    outorgante_bairro }, ${ outorgante_cidade }, ${ outorgante_estado },
    ${ outorgante_cep }.
```

Definindo variáveis

Variáveis (variable):

- Os nomes de variáveis podem conter qualquer sequência de letras maiúsculas ou minúsculas, dígitos e underscore.
- Exceto:
 - O primeiro caractere não pode ser um dígito;
 - Nenhum espaço é permitido; e
 - Nenhuma pontuação é permitida, exceto o underscore (_).

Definindo variáveis

A seguir nome de variáveis válidas:

- fried_fish1
- NyanCat
- nyancat
- __f645456DG_greij_43
- USER_PHONE_NUMBER

Nome de variáveis inválidas:

- 8th_plaintiff
- Nyan-Cat
- fried.fish1
- applicant_résumé
- user's_phone_number
- favorite animal
- beneficiary#1

Tipos de campos

Tipos de Campos (Inputs):

- Dentro de um bloco de pergunta, existem muitos valores possíveis de tipos de dados que você pode usar.
- O tipo de campo escolhido afeta o que o usuário vê e como é armazenada a entrada em uma variável.
- Atributo **datatype**

```
metadata:  
  title: Tipos de Inputs  
---  
mandatory: True  
question: |  
  Tipos de Campos:  
fields:  
  - "Texto Curto:": outorgante_logradouro  
    datatype: text  
    required: False  
  - "Texto Longo:": outorgante_logradouro  
    datatype: area  
    required: False  
  - "Passwords:": outorgante_numero  
    datatype: password  
    required: False  
  - "Dates:": outorgante_complemento  
    datatype: date  
    required: False  
  - "Times:": outorgante_bairro  
    datatype: time  
    required: False  
  - "Combined dates and times:": outorgante_cidade  
    datatype: datetime  
    required: False  
  - "E-mail addresses:": outorgante_estado  
    datatype: email  
    required: False
```

Modificadores de campos

Modificadores de campos:

- afeta como os campos serão apresentados para usuário.
- são atributos de cada campo, por exemplo, se ele é obrigatório ou não.

```
metadata:  
  title: Modificadores de Campos  
  ---  
mandatory: True  
question: |  
  Required: True ou False  
subquestion: |  
  - True: campo será de preenchimento obrigatório.  
  - False: campo será de preenchimento facultativo.  
fields:  
  - "CEP:": address_postal_code  
    required: True  
  - "Logradouro:": address  
    required: True  
  - "Número:": address_street_number  
    required: True  
  - "Complemento:": address_complement  
    required: False  
  - "Bairro:": address_neighborhood  
    required: True  
  - "Cidade:": address_city  
    required: True  
  - "Estado:": address_state  
    required: True  
  ---  
mandatory: True  
question: |  
  Datatype: usado para definir o tipo dos campos.  
fields:  
  - "Nome do Cliente:": cliente  
    datatype: text
```

Modificadores de campos

Required:

- afeta se o campo será opcional ou obrigatório. Se um campo for obrigatório, ele será marcado com um asterisco vermelho
- por padrão, os campos têm o seu preenchimento obrigatório.

```
metadata:  
  title: Modificadores de Campos  
  ---  
mandatory: True  
question: |  
  Required:  
fields:  
  - "CEP:": address_postal_code  
    required: True  
  - "Logradouro:": address  
    required: True  
  - "Número:": address_street_number  
    required: True  
  - "Complemento:": address_complement  
    required: False  
  - "Bairro:": address_neighborhood  
    required: True  
  - "Cidade:": address_city  
    required: True  
  - "Estado:": address_state  
    required: True  
  ---
```

Modificadores de campos

Default:

- Você pode fornecer um valor padrão para determinado campo.

```
question: |  
    What are your favorite things to eat?  
subquestion: |  
    Please be specific.  
fields:  
  - Vegetable: target_variable  
    default: eggplant  
  - Fruit: other_target_variable  
    default: |  
        ${ greatest_fruit }  
---  
code: |  
    greatest_fruit = "apples"
```

Modificadores de campos

Show if:

- o campo secundário será exibido dependendo do valor do campo principal.
- essa estrutura tem duas chaves: **variable** e **is**.
 - “variable”: se refere ao nome da variável do campo principal
 - “is”: refere-se ao valor atribuído no principal (é a condição que fará que esse campo seja mostrado).

```
question: |  
  What kind of car do you drive?  
fields:  
  - Make: car_make  
    choices:  
      - Honda  
      - Toyota  
      - Mazda  
      - Other  
  - Other make: car_make  
show if:  
  variable: car_make  
  is: Other
```

Bloco de impressão

Bloco de impressão:

- name
- language
- filename
- docx template file
- valid formats
 - PDF
 - docx

```
---
section: Conclusão
progress: 100
mandatory: True
question: |
    Aqui está o seu documento.
attachment:
    name: Ação Monitoria
    language: pt
    filename: monitoria
    docx template file: com-mcandiotto-monitoria.docx
    valid formats:
        - pdf
        - docx
---
```

Bloco de código python

Armazenar dados:

variable name:

data:

- é usado para guardar itens de uma lista.
- Exemplo: você pode criar a lista de UF's uma única vez e chamá-la na entrevista todas vez que for necessário.

```
--  
variable name: state_list  
data:  
- AC: AC  
- AL: AL  
- AM: AM  
- AP: AP  
- BA: BA  
- CE: CE  
- DF: DF  
- ES: ES  
- GO: GO  
- MA: MA  
- MT: MT  
- MS: MS  
- MG: MG  
- PA: PA  
- PB: PB  
- PR: PR  
- PE: PE  
- PI: PI  
- RJ: RJ  
- RN: RN  
- RS: RS  
- RO: RO  
- RR: RR  
- SC: SC  
- SP: SP  
- SE: SE  
- TO: TO  
--
```

Bloco de features

Exemplos de recursos especiais:

- Barra de progresso
- Menu
- Botão voltar
- Javascript

```
metadata:  
  title: Ação Monitória  
---  
features:  
  navigation: horizontal  
  progress bar: True  
  question back button: True  
  javascript:  
    - jquery-3.min.js  
    - jquery.js  
---  
sections:  
  - Comarca  
  - Autor  
  - Réu  
  - Contrato  
  - Assinatura  
  - Conclusão  
---
```

Atividades

Atividades

1. Automação das peças do DAJ.





Breaking the Law

Copyright 2019 Roberto Novaes (www.robertonovaes.com.br)

Copyright 2019 SiLEX Sistemas (www.silexsistemas.com.br)

Atribuição-NãoComercial-CompartilhaIgual 4.0 Internacional



Realização:



U F *m* G

Apoio:

MARCELO CANDIOTTO **MCSA**
Sociedade de Advogados



ASSOCIAÇÃO
BRASILEIRA DE
**LAWTECHS &
LEGALTECHS**