

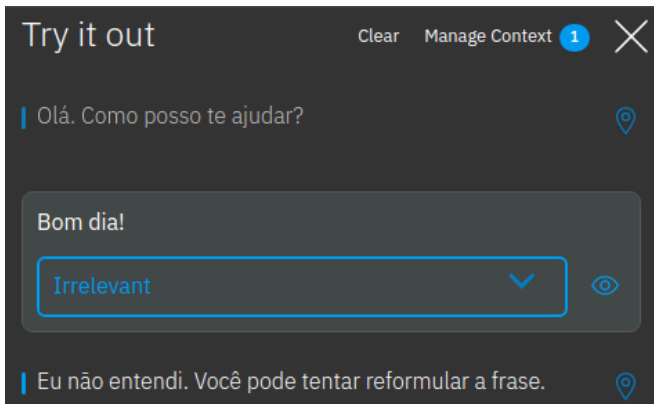
Aula 7

# Chatbots - Parte II

# Introdução

- Nesta aula iremos retomar o *chatbot* da aula passada
- O nosso *chatbot* já conseguia responder razoavelmente algumas perguntas
- Entretanto, se algo saísse do passo a passo programado, o nosso *chatbot* se perdia
- Além disso, as conversas estavam variando mas dentro de um conjunto fixo de possibilidades

# Introdução



# Introdução

- Nessa aula iremos ver como:
  - Tornar o nosso *chatbot* mais robusto as intenções do usuário
  - Permitir a variação de falas do *chatbot* de acordo com algumas variáveis
  - Analisar detalhes do funcionamento da classificação das intenções de forma a nos permitir fazer alguns ajustes
- Também veremos como importar e exportar dados do *Watson Assistant* para nos permitir fazer *backups* e possibilitar o trabalho em equipe

Introdução  
Tornando o Chatbot mais Amigável: Saudações e Despedidas  
Analisando as Confianças das Intenções  
Cumprimentando de Acordo com o Horário e Retreinando  
Importações, Exportações e Analytics  
Material Complementar

Tornando o Chatbot mais Amigável  
Utilizando o Objeto Input  
Melhorando as Saudações  
Assuntos Não Abordados

# Tornando o *Chatbot* mais Amigável

- Primeiramente, precisaremos criar uma intenção de saudação

The screenshot shows the IBM Watson Assistant interface for configuring a new intent. The intent name is '#saudacoes'. The description is 'Intenção para descobrir quando o usuário faz uma saudação'. The user examples section contains a table with 8 examples.

User examples (8) ↑	Added 1j
<input type="checkbox"/> Boa noite!	2 minutes ago
<input type="checkbox"/> Boa tarde!	2 minutes ago
<input type="checkbox"/> Bom dia!	2 minutes ago
<input type="checkbox"/> Oi	2 minutes ago
<input type="checkbox"/> Oi, tudo bem?	2 minutes ago
<input type="checkbox"/> Oi! Tudo certo?	2 minutes ago
<input type="checkbox"/> Olá	2 minutes ago

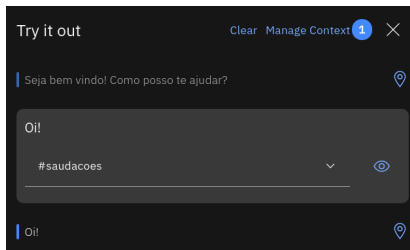
# Tornando o *Chatbot* mais Amigável

- Agora vamos adicionar um nó no diálogo para a nossa intenção de saudação

The screenshot displays the Microsoft Bot Framework Designer interface. On the left, a vertical flowchart shows the chatbot's logic. The 'Saudações' node, which responds with '#saudacoes', is highlighted with a blue border. Above the flowchart are buttons for 'Add node', 'Add child node', and 'Add folder'. On the right, the configuration panel for the 'Saudações' node is shown. It includes a 'Customize' button, a note about node names for disambiguation, and a 'Settings' link. Below this, the 'If assistant recognizes' section shows the intent '#saudacoes' with a plus sign to add more. The 'Assistant responds' section shows a 'Text' response type with the message 'Oi!' and a text area for 'Enter response variation'.

# Tornando o *Chatbot* mais Amigável

Iteração com o chatbot após a inserção do nó para tratar saudações



## Tornando o *Chatbot* mais Amigável

- Porém, idenpendente da frase, “Oi”, ou “Bom dia”, nosso chatbot só responde com “Oi”
- Seria mais educado/apropriado responder com “Bom dia” se o usuário digitou “Bom dia”, e com “Oi” se o usuário digitou “Oi”
- Para diferencias as respostas, poderíamos criar um entidade, avaliar o conteúdo dessa entidade e dar as respostas de acordo com o conteúdo (como foi feito no prazo de entrega para uma determinada cidade)
- Entretanto, iremos utilizar uma **alternativa** que permite analisar, em termos de **programação, a entrada do usuário**



## Tornando o *Chatbot* mais Amigável

- Para isso, vamos utilizar o `input.text`, o qual retorna a variável digitada pelo usuário
- As entradas do usuário ficam armazenadas no objeto `input`
- No caso, o `text` é um campo (referente ao texto digitado pelo usuário) do objeto `input`
- Assim como no Python, o `==` irá verificar se o campo é idêntico ao argumento posterior na comparação

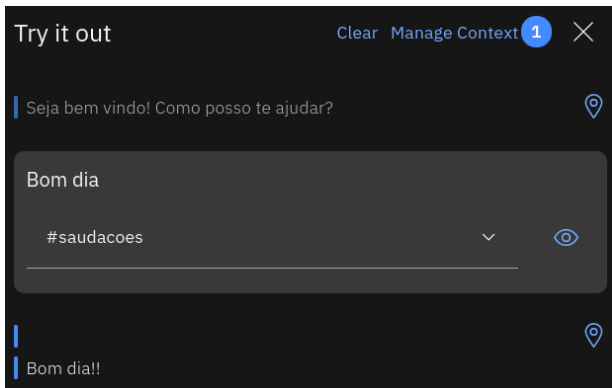
# Tornando o *Chatbot* mais Amigável

Inserindo um diálogo para reconhecer que o usuário digitou literalmente 'Bom dia'

The screenshot displays the Microsoft Bot Framework Designer interface. On the left, a flowchart shows the chatbot's logic: a start node leads to a 'Bem-vindo welcome' node, which then leads to a 'Saudações #saudacoes' node. From the 'Saudações' node, a 'Jump to Bom dia (Evaluate condition)' node is added. This node contains the condition 'Bom dia input.text == 'Bom dia''. Below this, a new node is being added, which also contains the condition 'Bom dia input.text == 'Bom dia'' and is set to 'Return allowed'.

On the right, the configuration for the 'Bom dia' node is shown. The node name is 'Bom dia'. Below the name, a description states: 'Node name will be shown to customers for disambiguation so use something descriptive.' There is a 'Settings' link. The 'If assistant recognizes' section shows the condition 'input.text == 'Bom dia''. The 'Assistant responds' section shows a 'Text' response type with the text 'Bom dia!!' and a 'Enter response variation' field.

# Tornando o *Chatbot* mais Amigável



## Utilizando o Objeto Input

- O *Watson Assistant* utiliza um conceito chamado *SpEL* (*Spring Expression Language*) para acessar ou manipular dados
- Assim como objetos para armazenar/manipular *strings* ou dados, o SpEL possui uma série de métodos para realizar comparações (<, >, ==, antes, depois, contido, etc.)
- Com isso, podemos utilizar o SpEL no lugar de entidades em determinadas situações

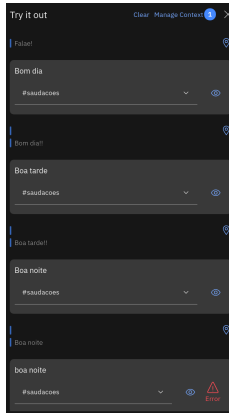
## Utilizando o Objeto Input

- Podemos criar mais nós para comparação das demais saudações
- Porém, vale ressaltar que a comparação com `==` é requer que o usuário digite uma *string* exata com a que foi informada

The screenshot displays a chatbot builder interface. On the left, a flowchart shows a sequence of nodes: 'Bem-vindo / welcome', 'Saudações / #saudacoes', a 'Jump to Bom dia (Evaluate condition)' node, 'Bom dia / input.text == 'Bom dia'', 'Boa tarde / input.text == 'Boa tarde'', and 'Boa noite / input.text == 'Boa noite''. The 'Boa noite' node is highlighted with a blue border. On the right, the configuration panel for the 'Boa noite' node is shown. It includes a search bar with 'Boa noite', a 'Node name will be shown to customers for disambiguation so use something descriptive' note, and a 'Settings' link. The 'If assistant recognizes' section contains the condition 'input.text == 'Boa noite''. The 'Assistant responds' section shows a 'Text' response type with the value 'Boa noite' and an 'Enter response variation' field. A note states 'Response variations are set to sequential. Set to random | multiline' with a 'Learn more' link. At the bottom, there is an 'Add response type +' button.

# Utilizando o Objeto Input

Testando os diálogos de saudações. Perceba que há uma falha no último diálogo



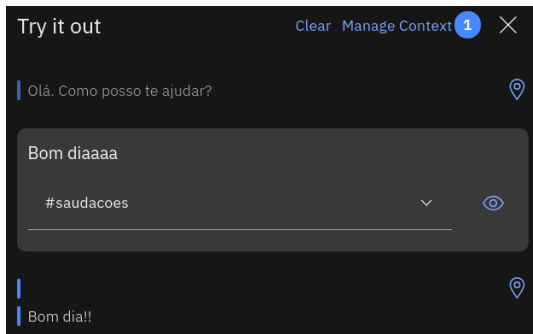
## Utilizando o Objeto Input

- Podemos tornar essa comparação mais flexível utilizando o método `contains`  $\Rightarrow$  `input.text.contains()`

The screenshot displays the Botpress chatbot builder interface. On the left, a flowchart shows three nodes: 'Bem-vindo welcome', 'Saudações #saudacoes', and a 'Jump to Bom dia (Evaluate condition)' node. The 'Jump to Bom dia' node is highlighted, leading to the configuration panel on the right.

The configuration panel for the 'Bom dia' node includes a 'Node name' field with the value 'Bom dia' and a 'Settings' link. Below this, the 'If assistant recognizes' section contains a condition: `input.text.contains('Bom dia')`. The 'Assistant responds' section shows a 'Text' response type with the value 'Bom dia!!' and an 'Enter response variation' field.

## Utilizando o Objeto Input



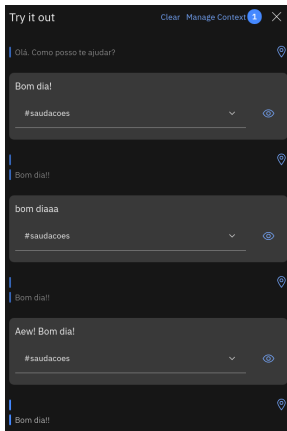


## Utilizando o Objeto Input

- Melhorou, porém, ainda temos que escrever na **mesma caixa** em que foram feitas as comparações e não podem haver qualquer tipo de **variação da escrita**
- Podemos, além de utilizar comandos para converter o texto para caixa alta ou caixa baixa, utilizar o método `matches`, o qual aceita expressões regulares

The screenshot displays a chatbot builder interface. On the left, a flowchart shows a sequence of nodes: 'Bem-vindo welcome', 'Saudações #saudacoes', and a 'Jump to Bom dia' node. The 'Bom dia' node is highlighted, showing its configuration. The configuration panel on the right has a title 'Bom dia' and a 'Node name' field. Below this, it says 'If assistant recognizes'. The condition is set to 'input.text.toLowerCase().matches("(.\*")bom dia.(.\*)")'. Under 'Assistant responds', there is a 'Text' dropdown and a response box containing 'Bom dia!'. At the bottom, there is a field for 'Enter response variation'.

# Utilizando o Objeto Input



# Utilizando o Objeto Input

- Por fim, para melhorar amigabilidade do nosso *chatbot*, vamos trabalhar com as despedidas também
- Para isso, vamos criar intenções e nós para tratar a despedida

IBM Watson Assistant Lite Upgrade

← | #Despedida

Intent name

# Despedida

Name your intent to match a customer's question or goal

Description (optional)

Reconhece que o usuário está se despedindo e apresenta uma mensagem apropriada.

User example

Type a user example here

Add unique examples of what the user might say. (Pro tip: Add at least 5 unique examples to help Watson understand)

Add example

<input type="checkbox"/> User examples (5) ↑	Added To
<input type="checkbox"/> Abração	a few seconds ago
<input type="checkbox"/> Ade	a few seconds ago
<input type="checkbox"/> Ade mais	a few seconds ago
<input type="checkbox"/> Bye bye	a few seconds ago
<input type="checkbox"/> Falew	a few seconds ago
<input type="checkbox"/> Tchau	a few seconds ago

# Utilizando o Objeto Input

The screenshot displays the Microsoft Bot Framework Designer interface. On the left, a flowchart shows several nodes: 'Saudações', 'Dúvida Reserva de Produto no Carrinho de Compra', 'Dúvida Sobre Prazo de Entrega com Cidade', 'Dúvida Sobre Prazo de Entrega', 'Comprar Celular 2', 'Comprar Celular', and 'Despedida'. The 'Despedida' node is selected and highlighted with a blue border. On the right, the configuration panel for the 'Despedida' node is shown. It includes a 'Node name' field with the value 'Despedida', a 'Customize' button, and a 'Settings' link. Below this, the 'If assistant recognizes' section shows a list of recognized intents, including '#Despedida'. The 'Assistant responds' section shows a 'Text' response type with a sample response 'Foi um prazer atendê-lo! Até a próxima!'. Below the response, there is a section for 'Response variations' set to 'sequential', and a 'Learn more' link. At the bottom, the 'Then assistant should' section is visible, with a note about choosing whether the assistant should continue or wait for the customer to respond.

Add node Add child node Add folder

Despedida

Node name will be shown to customers for disambiguation so use something descriptive. Settings

Customize

If assistant recognizes

#Despedida

Assistant responds

Text

Foi um prazer atendê-lo! Até a próxima!

Enter response variation

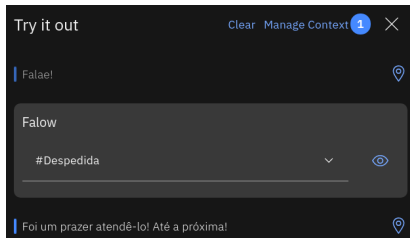
Response variations are set to **sequential**. Set to **random** | **multiline**  
[Learn more](#)

Add response type +

Then assistant should

Choose whether you want your Assistant to continue, or wait for the customer to respond.

## Utilizando o Objeto Input



## Melhorando as Saudações

- Para melhorar as saudações, vamos agora separar os cumprimentos “oi”, “olá”, etc., das saudações “bom dia”, “boa tarde” e “boa noite”
- Vamos criar uma nova intenção #cumprimentar

The screenshot shows the IBM Watson Assistant interface for configuring a new intent named "#cumprimentar". The interface includes a header with "IBM Watson Assistant Lite" and an "Upgrade" button. Below the header, there's a breadcrumb navigation showing a back arrow and "#cumprimentar". The main configuration area has several sections: "Intent name" with a text box containing "#cumprimentar", "Name your intent to match a customer's question or goal", "Description (optional)" with a text box containing "Add a description to this intent", and "User example" with a text box containing "Take a user example here" and a note "Add unique examples of what the user might say. (Pro tip: Add at least 5 unique examples to help Watson understand)". Below these sections is a table of user examples.

<input type="checkbox"/> User examples (0/5)	Added 11
<input type="checkbox"/> Baure?	a few seconds ago
<input type="checkbox"/> Belaxe?	a few seconds ago
<input type="checkbox"/> Blet?	a few seconds ago
<input type="checkbox"/> Como vai você?	a few seconds ago
<input type="checkbox"/> Como você está?	a few seconds ago
<input type="checkbox"/> Oi, tudo bem?	a few seconds ago
<input type="checkbox"/> Oi	a few seconds ago
<input type="checkbox"/> Oi, tudo bem?	a few seconds ago

**OBSERVAÇÃO:** alguns conflitos irão aparecer ⇒ tirar os conflitos da intenção de saudações

## Melhorando as Saudações

- Vamos agora inserir um nó para capturar a intenção de cumprimento no diálogo

The screenshot displays the Microsoft Bot Framework Designer interface. On the left, a tree view shows the bot's structure with nodes for 'Bem-vindo welcome', 'Cumprimentos #cumprimentar' (highlighted with a blue border), 'Saudações #saudacoes', and 'Dúvida Reserva de Produto no Carrinho de Compra #duvida\_carrinho\_compra'. The main workspace shows the configuration for the 'Cumprimentos' node. The 'Node name' is 'Cumprimentos', with a note that it will be shown to customers for disambiguation. Below this, the 'If assistant recognizes' section contains a list with the intent '#cumprimentar'. The 'Assistant responds' section is currently empty, showing a 'Text' dropdown and icons for adding or removing responses.

## Melhorando as Saudações

- Vamos agora continuar a conversa com o usuário e tratar a parte dos cumprimentos
- Para isso, vamos inserir um nó de cumprimentos no diálogo
- Depois vamos simular o seguinte diálogo:

Situação #1

Usuário: Oi, tudo bem?

Chatbot: tudo, e você?

Usuário: Vou bem.

Chatbot: Que ótimo.

Situação #2

Usuário: Oi, tudo bem?

Chatbot: Tudo, e você?

Usuário: Tudo péssimo.

Chatbot: Fique tranquilo, vai melhorar.



## Melhorando as Saudações

- No caso da primeira situação de diálogo temos uma intenção positiva
- No caso do segundo diálogo temos uma intenção negativa
- Portanto, temos que reconhecer os sentimentos (por exemplo, criando uma entidade) para responder apropriadamente

IBM Watson Assistant Lite Upgrade

← | @sentimento

Entity name  
@sentimento

Name your entity to match the category of values that it will detect.

Value:  Synonyms  +

Fuzzy matching <sup>①</sup> ☒ On

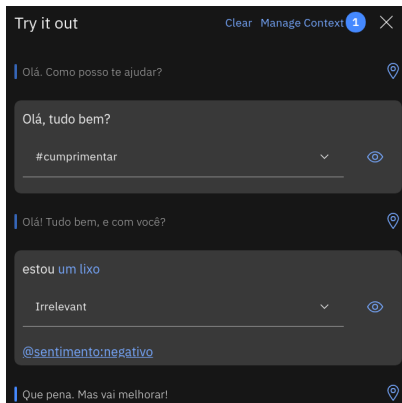
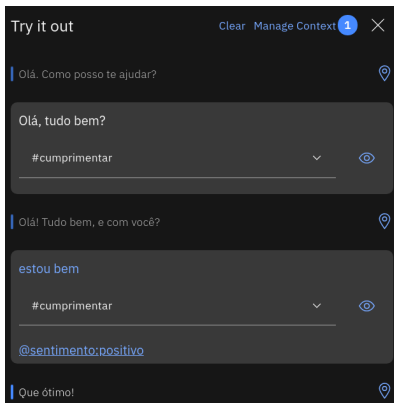
<input type="checkbox"/> Values (2) ↑	Type	
<input type="checkbox"/> negativo	Synonyms	tudo péssimo, nada bem, um lixo, bem mal
<input type="checkbox"/> positivo	Synonyms	vou bem, estou bem, uma maravilha, tudo certo, na paz, suave

## Melhorando as Saudações

- Vamos agora adicionar nós filho nas saudações para o sentimento positivo e sentimento negativo

The screenshot displays the Microsoft Bot Framework Designer interface. On the left, a tree view shows the bot's structure with nodes for 'Bem-vindo welcome', 'Cumprimentos #cumprimentar', 'Sentimento Positivo @sentimento:positivo', and 'Sentimento Negativo @sentimento:negativo'. The 'Sentimento Negativo' node is selected and highlighted in blue. On the right, the configuration panel for this node is shown. It includes a 'Node name' field set to 'Sentimento Negativo', a 'Customize' button, and a 'Settings' link. Below this, the 'If assistant recognizes' section shows the condition '@sentimento:negativo'. The 'Assistant responds' section shows a 'Text' response type with the message 'Que pena. Mas vai melhorar!'. At the bottom, a note states 'Response variations are set to sequential. Set to random | multiline'.

# Melhorando as Saudações



## Melhorando as Saudações

- Vamos também tratar situações onde o usuário diz apenas “oi” ou “olá”, e “oi” com “tudo bem”, etc.

The screenshot displays the Microsoft Bot Framework Designer interface. On the left, a flowchart shows the bot's logic: it starts with a 'Bem-vindo' (Welcome) node, followed by a 'Cumprimentos' (Greetings) node. The 'Cumprimentos' node has a condition 'input.text.toLowerCase().matches(oi[1,2]+) || input.text.toLowerCase().matches(ola[1,2]+)'. If this condition is met, it jumps to a 'Só oi' (Only oi) node. The 'Só oi' node has a response 'Oi! Tudo bem?' and a condition 'input.text.toLowerCase().matches(oi[1,2]+) || input.text.toLowerCase().matches(ola[1,2]+)'. If this condition is met, it jumps to a 'Sentimento Positivo' (Positive Sentiment) node. The 'Sentimento Positivo' node has a response '@sentimento:positivo' and a condition 'input.text.toLowerCase().matches(oi[1,2]+) || input.text.toLowerCase().matches(ola[1,2]+)'. If this condition is met, it returns allowed.

On the right, the configuration for the 'Só oi' node is shown. The node name is 'Só oi'. The condition is 'input.text.toLowerCase().matches(oi[1,2]+) || input.text.toLowerCase().matches(ola[1,2]+)'. The assistant responds with 'Oi! Tudo bem?'. The response variations are set to sequential.

# Melhorando as Saudações

The screenshot displays the Microsoft Bot Framework Designer interface, divided into a flowchart editor on the left and a configuration panel on the right.

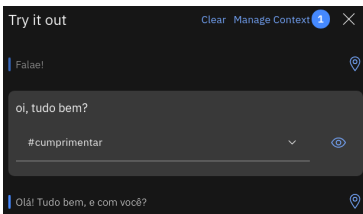
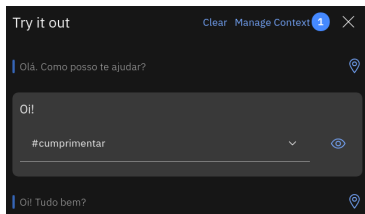
**Flowchart Editor (Left):**

- Nodes:**
  - Bem-vindo welcome**: 1 Responses / 0 Context Set / Does not return.
  - Cumprimentos #cumprimentar**: 1 Responses / 0 Context Set / Jump to / Does not return.
  - Jump to Só oi (Evaluate responses)**: A connector node.
  - Só oi**: `input.text.toLowerCase().matches(/oi/)?` || `input.text.toLowerCase().matches(/oi/)?`. 1 Responses / 0 Context Set / Jump to / Return allowed.
  - Jump to Sentimento Positivo (Wait for user input, then evaluate condition)**: A connector node.
  - Oi com tudo bem #cumprimentar**: 1 Responses / 0 Context Set / Jump to / Return allowed.
  - Jump to Sentimento Positivo (Wait for user input, then evaluate condition)**: A connector node.
  - Sentimento Positivo @sentimento:positivo**: 1 Responses / 0 Context Set / Return allowed.

**Configuration Panel (Right):**

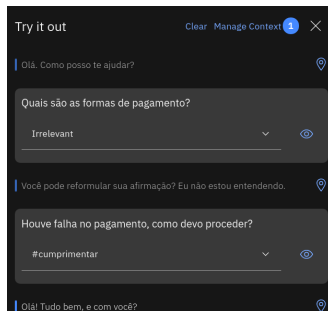
- Node Name:** Oi com tudo bem. (Note: Node name will be shown to customers for disambiguation so use something descriptive. [Settings](#))
- If assistant recognizes:** #cumprimentar.
- Assistant responds:**
  - Text:**
    - Ola! Tudo bem, e com você?
    - Oi! Tudo ótimo, e com você?
    - Enter response variation
  - Response variations:** set to **sequential**. Set to **random** | **multiline**. ([Learn more](#))
- Then assistant should:** Choose whether you want your Assistant to continue, or wait for the customer to respond.
  - Jump to:** Sentimento Positivo (Wait for user input, then evaluate condition).

## Melhorando as Saudações



## Assuntos Não Abordados

- Assuntos ainda não abordados podem ser disparados por outros nós ou ainda disparar o nó “Em outros casos”
- Uma solução mais elegante é criar um nó para conter intenções ainda não abordadas no chat, ou seja, um nó “fora de escopo”



# Assuntos Não Abordados

IBM Watson Assistant Lite Upgrade

← | #fora\_de\_escopo

Intent name

# fora\_de\_escopo

Name your intent to match a customer's question or goal

Description (optional)

Utilizado para capturar intenções não programadas

User example

Type a user example here

Add unique examples of what the user might say. (Pro tip: Add at least 5 unique examples to help Watson understand)

Add example

☐ User examples (6) ↑

Added ↓

☐ Como devo proceder mediante falha de pagamento?

a minute ago

☐ Como faço para acompanhar meu pedido?

a minute ago

☐ Houve falha no meu pagamento, o que fazer?

a minute ago

☐ Onde posso fazer o acompanhamento do meu pedido?

a few seconds ago

☐ Quais métodos de pagamentos vocês aceitam?

a few seconds ago

☐ Qual é o tempo de restituição do pagamento?

a few seconds ago



# Assuntos Não Abordados

Add node
Add child node
Add folder

Dúvida Reserva de Produto no Carrinho de Compra  
#duvida\_carrinho\_compra  
1 Responses / 0 Context Set / Does not return

Dúvida Sobre Prazo de Entrega com Cidade  
#duvida\_prazo\_entrega\_produto && @cidade  
0 Responses / 0 Context Set / Jump to / Does not return

Dúvida Sobre Prazo de Entrega  
#duvida\_prazo\_entrega\_produto  
1 Responses / 0 Context Set / Does not return

Comprar Celular 2  
#comprar\_produto  
1 Responses / 4 Context Set / 4 Slots / Does not return

Comprar Celular  
#comprar\_produto  
1 Responses / 0 Context Set / Does not return

Despedida  
#Despedida  
1 Responses / 0 Context Set / Does not return

Fora de Escopo  
#fora\_de\_escopo  
1 Responses / 0 Context Set / Does not return

Fora de Escopo

Node name will be shown to customers for disambiguation so use something descriptive.
Settings

If assistant recognizes

#fora\_de\_escopo
+

Assistant responds

Text
^ v

Por enquanto eu não sou capaz de responder esse assunto. Por favor, entre em contato com a central de atendimento (ZZZZ ZZZZ ZZZZ) ou pelo email ZZZZZZZZZZZZ@ZZZZZZ.ZZZ.ZZ

Enter response variation

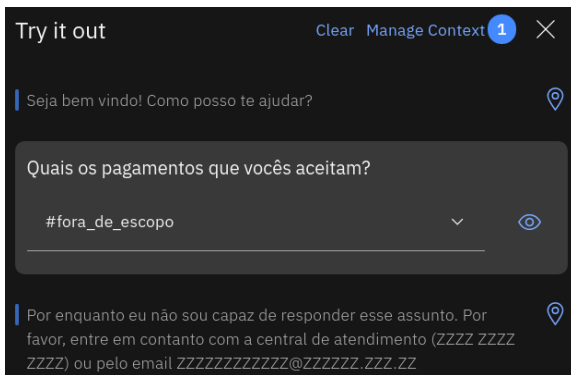
Response variations are set to **sequential**. Set to **random** | **multiline**  
Learn more

Add response type +

Then assistant should

Choose whether you want your Assistant to continue, or wait for the customer to respond.

## Assuntos Não Abordados



## Analisando as Confianças das Intenções

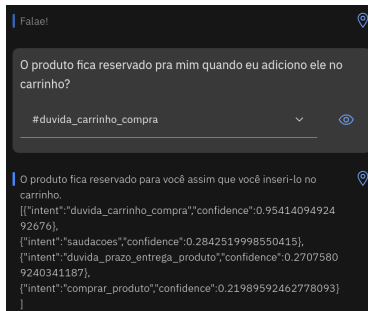
- Vamos agora ver como analisar a confiança de classificação das intenções do Watson Assistant
- Isso poder ser útil para:
  - Fazer algum tipo de análise
  - Estabelecer limites de confiança
  - Ver o que acontece quando o nosso *chatbot* reconhece uma intenção errada

## Analizando as Confianças das Intenções

- Uma das formas de analisar a confiança das intenções é utilizando a diretiva `<? intents ?>` na resposta do chatbot na resposta do *chatbot*

The screenshot displays a chatbot configuration interface. On the left, a sidebar shows a list of nodes: 'Bem-vindo welcome', 'Saudações #saudacoes', 'Dúvida Reserva de Produto no Carrinho de Compra #duvida\_carrinho\_compra', and 'Dúvida Sobre Prazo de Entrega com Cidade #duvida\_prazo\_entrega\_produto && @cidade'. The main panel shows the configuration for the selected node 'Dúvida Reserva de Produto no Carrinho de Compra'. It includes a title bar with 'Customize' and 'Settings' options. Below the title, there is a section 'If assistant recognizes' with a dropdown menu showing '#duvida\_carrinho\_compra'. The 'Assistant responds' section shows a text response: 'O produto fica reservado para você assim que você inseri-lo no carrinho. <? intents ?>'. A second response is also visible: 'A reserva de produto é automaticamente feita assim que o produto for inserido no carrinho.'

# Analisando as Confianças das Intenções



**OBSERVAÇÃO:** a resposta é uma lista de dicionários, JSONs ou objetos

## Analizando Melhor as Intenções

- Vamos considerar a análise de algumas intenções

<p>quero comprar um produto</p> <p>#comprar-produto</p> <p>Legal! Vou te ajudar! Irei fazer algumas perguntas para selecionar o melhor produto para você, ok?</p> <pre>[{"intent": "comprar-produto", "confidence": 0.9060810565948487}, {"intent": "adicionar-produto", "confidence": 0.25613167881965637}, {"intent": "memes-jacquin", "confidence": 0.25350502133369446}, {"intent": "saudacoes", "confidence": 0.2500673830509186}, {"intent": "entrega-de-produto", "confidence": 0.24697065353393555}, {"intent": "cumprimentar", "confidence": 0.24484030902385712}, {"intent": "despedida", "confidence": 0.2368112772703171}, {"intent": "agradecimento", "confidence": 0.20814867690205574}]</pre>	<p>mim quer comprar alguma coisa</p> <p>#comprar-produto</p> <p>Legal! Vou te ajudar! Irei fazer algumas perguntas para selecionar o melhor produto para você, ok?</p> <pre>[{"intent": "comprar-produto", "confidence": 0.8095228672027588}, {"intent": "cumprimentar", "confidence": 0.3229365110397339}, {"intent": "despedida", "confidence": 0.3075103044509888}, {"intent": "saudacoes", "confidence": 0.26557372212409974}, {"intent": "adicionar-produto", "confidence": 0.2487254709005356}, {"intent": "memes-jacquin", "confidence": 0.24314154386520387}, {"intent": "entrega-de-produto", "confidence": 0.2391105443239212}, {"intent": "agradecimento", "confidence": 0.22372073531150818}]</pre>	<p>eu compra celular</p> <p>#comprar-produto</p> <p>Legal! Vou te ajudar! Irei fazer algumas perguntas para selecionar o melhor produto para você, ok?</p> <pre>[{"intent": "comprar-produto", "confidence": 0.5294435977935792}, {"intent": "cumprimentar", "confidence": 0.3277174949645996}, {"intent": "despedida", "confidence": 0.32356187105178835}, {"intent": "entrega-de-produto", "confidence": 0.29732635617256165}, {"intent": "saudacoes", "confidence": 0.2698468923568726}, {"intent": "adicionar-produto", "confidence": 0.2623544454574585}, {"intent": "agradecimento", "confidence": 0.25341902375221254}, {"intent": "memes-jacquin", "confidence": 0.22845303416252138}]</pre>
--	---	--

- Perceba que quanto mais próximo dos exemplos fornecidos, maior é a confiança de determinada intenção

## Analisando Melhor as Intenções

- Podemos retornar o primeiro elemento da lista, que é a intenção com maior confiança, analisar o atributo `confidence` e compará-lo com um limiar definido pelo usuário após testes
- Assim, podemos evitar o disparo de um nó caso achamos que a confiança seja suficiente

# Analizando Melhor as Intenções

Save new version

Comprar Celular 2

Node name will be shown to customers for disambiguation so use something descriptive. [Settings](#)

[Customize](#) [X](#)

If assistant recognizes

#comprar\_produto [🔗](#) +

Then check for [Manage handlers](#)

	Check for	Save it as	If not present, ask	Type
1	@sim-nao:sir	\$sim_nao_aju	(S) Legal! Vou	Required <a href="#">⚙️</a> <a href="#">🔗</a>

[Save new version](#)

Comprar Celular 2

Node name will be shown to customers for disambiguation so use something descriptive. [Settings](#)

[Customize](#) [X](#)

If assistant recognizes

#comprar\_produto [🔗](#) and [▼](#) intent[0].confidence > 0.7 [🔗](#) +

Try it out [Clear](#) [Manage Context](#) [1](#) [X](#)

| False!

eu compra celular

#comprar\_produto [▼](#) [👁](#)

(S) Legal! Vou te ajudar, ok?

```
[{"intent":"comprar_produto","confidence":0.6367680549621582}, {"intent":"cumprimentar","confidence":0.31960536241531373}, {"intent":"despedida","confidence":0.3122413992881775}, {"intent":"fora_de_escopo","confidence":0.27338633537292484}, {"intent":"saudacoes","confidence":0.26534914374351504}, {"intent":"duvida_carrinho_compra","confidence":0.2647180199623108}, {"intent":"duvida_prazo_entrega_produto","confidence":0.25165818929672246}]
```

Try it out [Clear](#) [Manage Context](#) [1](#) [X](#)

| Olá. Como posso te ajudar?

eu compra celular

#comprar\_produto [▼](#) [👁](#)

| Eu não entendi o sentido.



# Cumprimentando de Acordo com o Horário

- Vamos melhorar a parte de bem vindo do *chatbot* dando “Bom dia”, “Boa tarde” ou “Boa noite” de acordo com o horário.
- Para isso, vamos utilizar o conceito de múltiplas respostas condicionadas a condições

Customize "Bem-vindo"

Customize route | Signature

Slots

Enable this to gather the information your bot needs to respond to a user within a single route.

☐ Prompt for everything  
Enable this to ask for multiple pieces of information in a single prompt, so your user can provide them all at once and not be prompted for them one at a time.

Webhooks

Enable this setting to send a POST request from the dialog node to the webhook URL. The URL and headers are defined in the Webhooks settings of the Options tab. After you enable this setting, the Multiple conditional responses setting is enabled automatically to support adding a response to show when the request is successful and another response to show if the request fails. Learn more

Webhook URL missing Specify the request URL for the web app you want to call out.

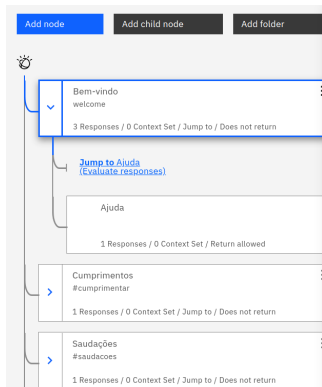
Multiple conditioned responses

Enable multiple responses so that your bot can provide different responses to the same input, based on other conditions.

## Cumprimentando de Acordo com o Horário

- Para saber que horas são agora (horário do usuário), pode-se utilizar o método `now()`
- Pode-se também fazer uso de outros métodos do objeto retornado por `now()`, como `before()` e `after()`
- Com isso podemos apagar o nó de saudações e criar um nó filho em Bem Vindo contento as perguntas de ajuda

# Cumprimentando de Acordo com o Horário



Bem-vindo

Customize

Node name will be shown to customers for disambiguation so use something descriptive.

Settings

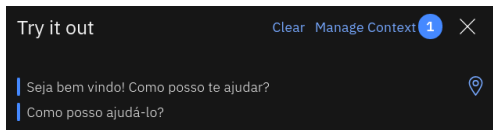
If assistant recognizes

welcome +

Assistant responds

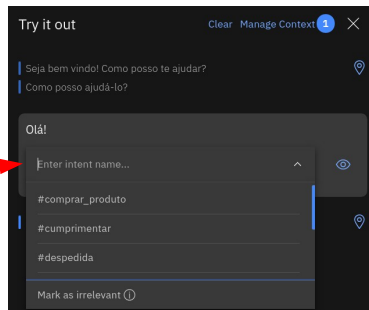
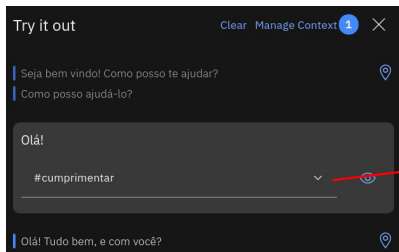
	If assistant recognizes	Respond with		
1	<code>now().before("12:00:00")</code>	Bom dia!		
2	<code>now().before("18:00:00")</code>	Boa tarde!		
3	<code>now().before("04:00:00")</code>	Boa noite!		

# Cumprimentando de Acordo com o Horário



## Retreinando pela Interface do Chat

- Podemos retreinar uma intenção utilizando diretamente a caixa de combinações do *chatbot*

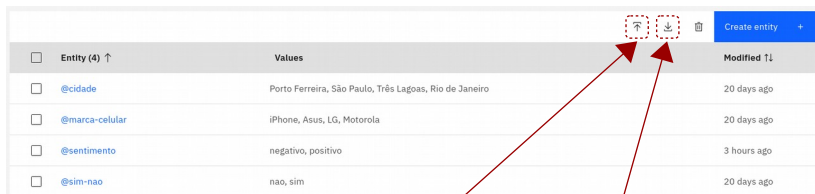


# Importações e Exportações

- 1 Podemos importar e exportar entidades, intenções e *skills* do Watson Assintent
- 2 Isso pode ser útil para:
  - Fazer *backups*
  - Trabalhar em equipe

# Importações e Exportações

## ● Importando e exportando entidades



The screenshot shows a table of entities with columns for selection, name, values, and modification time. Above the table are icons for import, export, and delete, along with a 'Create entity' button. Two red arrows point from the labels 'Importar' and 'Exportar' below to the respective icons in the table header.

<input type="checkbox"/>	Entity (4) ↑	Values	Modified ↑↓
<input type="checkbox"/>	@cidade	Porto Ferreira, São Paulo, Três Lagoas, Rio de Janeiro	20 days ago
<input type="checkbox"/>	@marca-celular	iPhone, Asus, LG, Motorola	20 days ago
<input type="checkbox"/>	@sentimento	negativo, positivo	3 hours ago
<input type="checkbox"/>	@sim-nao	nao, sim	20 days ago

Importar

Exportar

## Importações e Exportações

- As entidades são exportadas em formato CSV (porém, não estruturado)

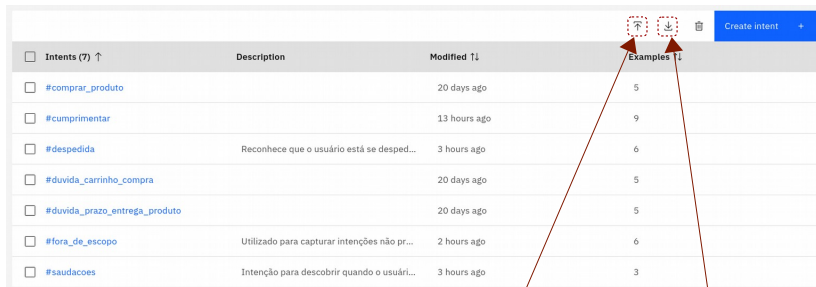
```
cidade,Porto Ferreira,PF,Capital da Cerâmica  
cidade,São Paulo,SP,Sampa,Terra da Garoa  
cidade,Três Lagoas,TL,Trela,Inferno  
cidade,Rio de Janeiro,RJ,Rio,Terra da milica  
marca-celular,Asus  
marca-celular,iPhone,Apple,iOS  
marca-celular,LG  
marca-celular,Motorola,Moto,Moto G,Moto Edge,Motobosta  
sentimento,negativo,tudo péssimo,nada bem,um lixo,bem mal,tô na merda  
sentimento,positivo,vou bem,estou bem,uma maravilha,tudo certo,na paz,suave,tudo ótimo  
sim-nao,sim,ok,blz,beleza,bora,vamos lá  
sim-nao,nao,não,desisto,nem
```

- Portanto, você pode especificar suas entidades em um editor de textos qualquer
- Permite que diferentes pessoas criem diferentes entidades



# Importações e Exportações

- O mesmo vale para intenções



<input type="checkbox"/> Intents (7) ↑	Description	Modified ↑↓	Examples ↑↓
<input type="checkbox"/> #comprar_produto		20 days ago	5
<input type="checkbox"/> #cumprimentar		13 hours ago	9
<input type="checkbox"/> #despedida	Reconhece que o usuário está se desped...	3 hours ago	6
<input type="checkbox"/> #duvida_carrinho_compra		20 days ago	5
<input type="checkbox"/> #duvida_prazo_entrega_produto		20 days ago	5
<input type="checkbox"/> #fora_de_escopo	Utilizado para capturar intenções não pr...	2 hours ago	6
<input type="checkbox"/> #saudacoes	Intenção para descobrir quando o usuári...	3 hours ago	3

Importar

Exportar

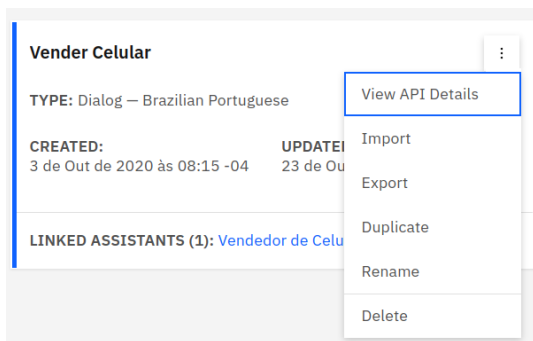
# Importações e Exportações

- As intenções também são exportadas em formato CSV (porém, estruturado)

	Exemplo	Intenção
0	Gostaria de comprar um celular.	comprar_produto
1	Quero comprar um telefone celular.	comprar_produto
2	Quero adquirir um smartphone.	comprar_produto
3	Gostaria de levar um celular.	comprar_produto
4	Preciso de um celular novo.	comprar_produto
5	Olá!	cumprimentar
6	Como vai você?	cumprimentar
7	Como você está?	cumprimentar
8	Oi!	cumprimentar
9	Olá, tudo bem?	cumprimentar
10	Oi, tudo bem?	cumprimentar
11	Baum?	cumprimentar
12	Beleza?	cumprimentar
13	Blz?	cumprimentar
14	Abs	despedida
15	Abraços	despedida
16	Tchau	despedida
17	Até mais!	despedida
18	Bye bye	despedida
19	Falow	despedida

# Importações e Exportações

- Por fim, também podemos exportar as *skills*



# Importações e Exportações

- Porém, diferente das demais, a skill é exportada em formato JSON

```

102 {
103   "intents": [
104     {
105       "intent": "comprar_produto",
106       "examples": [
107         {
108           "text": "Gostaria de comprar um celular."
109         },
110         {
111           "text": "Gostaria de levar um celular."
112         },
113         {
114           "text": "Preciso de um celular novo."
115         },
116         {
117           "text": "Quero adquirir um smartphone."
118         },
119         {
120           "text": "Quero comprar um telefone celular."
121         }
122       ],
123       "description": ""
124     },
125     {
126       "intent": "cumprimentar",
127       "examples": [
128         {
129           "text": "Bom?"
130         },
131         {
132           "text": "Beleza?"
133         },
134         {
135           "text": "Ble?"
136         },
137         {
138           "text": "Como vai você?"
139         },
140         {
141           "text": "Como você está?"
142         },
143         {
144           "text": "Oi!"
145         },
146         {
147           "text": "Oi, tudo bem?"
148         }
149       ]
150     }
151   ]
152 }

```

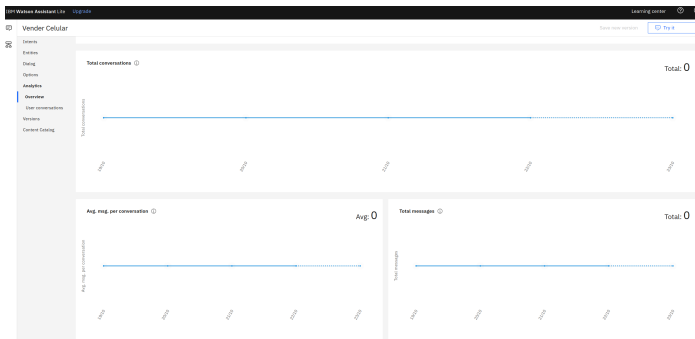
```

153 "entities": [
154   {
155     "entity": "cidade",
156     "values": [
157       {
158         "type": "synonyms",
159         "value": "Porto Ferreira",
160         "synonyms": [
161           "Capital da Cerâmica",
162           "PF"
163         ]
164       },
165       {
166         "type": "synonyms",
167         "value": "Rio de Janeiro",
168         "synonyms": [
169           "Rio",
170           "RJ",
171           "Terra da milica"
172         ]
173       },
174       {
175         "type": "synonyms",
176         "value": "São Paulo",
177         "synonyms": [
178           "Sampa",
179           "SP",
180           "Terra da Garoa"
181         ]
182       },
183       {
184         "type": "synonyms",
185         "value": "Três Lagoas",
186         "synonyms": [
187           "Inferno",
188           "TL",
189           "Trela"
190         ]
191       }
192     ],
193     "fuzzy_match": true
194   }
195 ]

```

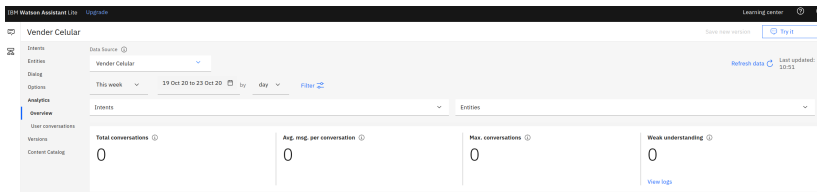
# Analytics

- O *Watson Assistant* fornece um *Analytics* para que possamos analisar estatísticas gerais sobre as conversas com o *chatbot*, bem como analisar as conversas que os usuários estão tendo com o *chat*



# Analytics

- Pode-se filtrar as conversas dos usuários por intenções ou entidades
- Além disso, o *analytics* já provê ferramentas para edições e retreinamentos



## Material Complementar

- Explorando funções avançadas do Watson Assistant:  
linguagem SpEL

<https://medium.com/ibmdeveloperbr/explorando-fun%C3%A7%C3%B5es-avan%C3%A7adas-do-watson-assistant-linguagem-spel-7f030890127c>

- Expressões para acessar objetos

<https://cloud.ibm.com/docs/services/assistant?topic=assistant-expression-language#accessing-and-evaluating-objects>

- sys-date do Watson Assistant

<https://cloud.ibm.com/docs/services/assistant/dialog-methods.html#methods-to-process-values>

## Material Complementar

- Spring Expression Language (SpEL)

`https:`

`//docs.spring.io/spring/docs/current/spring-framework-reference/core.html#expressions`

- IBM Watson II: Desenvolvendo uma API com Watson Assistant + Node JS

`https://medium.com/cwi-software/`

`ibm-watson-ii-desenvolvendo-uma-api-com-watson-assistant-node-js-5ea6212f293`

- Criando um Chatbot com IBM Watson e Node.js — parte 1

`https://medium.com/trainingcenter/`

`criando-um-chatbot-com-ibm-watson-e-node-js-parte-1-ef9bf8f95282`



## Material Complementar

- Watson Assistant with Python-Flask (ChatBots Translation IBM options)

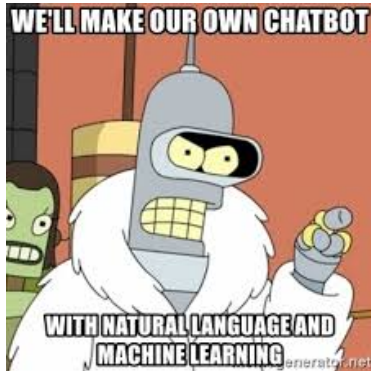
<https://medium.com/@novagenio/>

watson-assistant-with-python-flask-chatbots-translation-ibm-options-fd4885ee4bb8

- watson-developer-cloud/python-sdk

<https://github.com/watson-developer-cloud/python-sdk>

## Imagem do dia



# Tópicos em Inteligência Artificial

<http://ava.ufms.br/>

Rafael Geraldeli Rossi  
[rafael.g.rossi@ufms.br](mailto:rafael.g.rossi@ufms.br)