

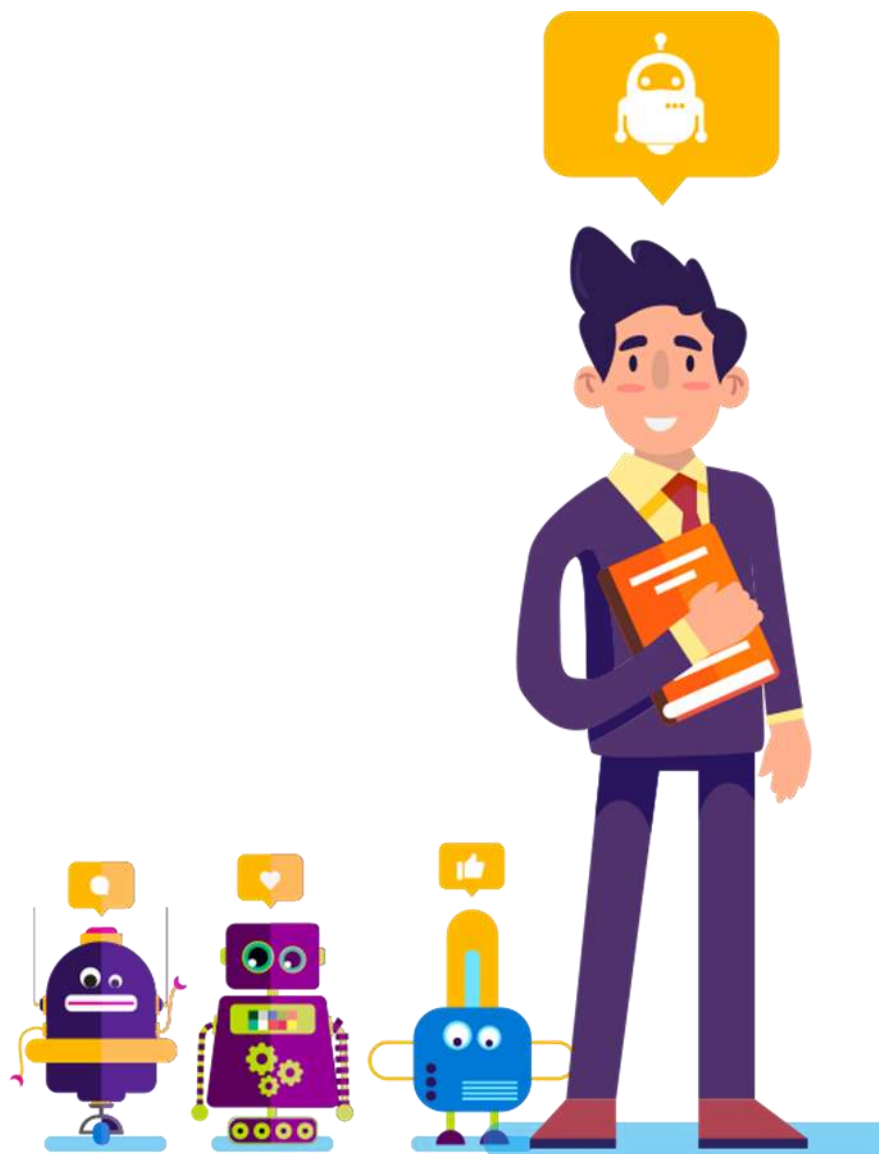
Maratona Bots

Hands on

Versão: 1.0.0

Janeiro de 2018

[Angelo Belchior](#)



Aprenda ainda mais!

Confira o repertório de artigos técnicos da Microsoft e tenha acesso a conteúdos exclusivos produzidos pelo nosso time de especialistas. [Acesse já!](#)

Hands-on Lab Bot Framework

O que é o Bot Framework?

O Bot Framework é um serviço criado pela Microsoft para criação de Bots que poderão ser disponibilizados em vários canais como Skype, Telegram, Messenger, Teams, etc. de forma simples e rápida.

Esse serviço está disponível em <https://dev.botframework.com/> e é possível utilizá-lo usando um SDK ou comunicando através das API's Rest que ele disponibiliza.

O que é o Bot Builder?

O Bot Builder é o SDK totalmente Open Source usado para criação de Bots que serão integrados ao serviço Bot Framework. Esse SDK está disponível em Asp.Net e Node.js.

Com o Bot Builder é possível gerenciar Diálogos, Memória do Bot, entregar conteúdo multimídia ao usuário, criar conversação baseada em Formulários de forma simples e rápida.

Além disso, você pode baixar as extensões do Bot Builder para Visual Studio, facilitando a criação do seu projeto Asp.Net.

Com essa extensão, num simples Arquivo -> Novo -> Projeto -> Bot Application, já é criada toda a estrutura necessária para o desenvolvimento do Bot, sendo que o próprio projeto já é funcional podendo ser testado no Bot Emulator.

Você pode baixar o SDK aqui: <https://github.com/Microsoft/BotBuilder>

Instalando as Extensões do Visual Studio

Instalar extensões no Visual Studio é muito simples.

Você vai precisar baixar três arquivos *.zip:

Bot Application - <http://aka.ms/bf-bc-vstemplate>

Bot Controller - <http://aka.ms/bf-bc-vscontrollertemplate>

Bot Dialog - <http://aka.ms/bf-bc-vsdialogtemplate>

Copie o arquivo Bot Application.zip para a pasta %USERPROFILE%\Documents\Visual Studio 2017\Templates\ProjectTemplates\Visual C#.

Copie os arquivos Bot Controller.zip e Bot Dialog.zip para a pasta %USERPROFILE%\Documents\Visual Studio 2017\Templates\ItemTemplates\Visual C#\

O Bot Application.zip é o template do projeto de Bot. Ele contém toda a estrutura necessária para o projeto.

Já os arquivos Bot Controller.zip e Bot Dialog.zip são templates de para arquivos de Controller e de Dialog.

Para saber mais acesse: https://docs.microsoft.com/pt-br/bot-framework/dotnet/bot-builder-dotnet-quickstart?WT.mc_id=CSE_diplomado_handson

O que é o LUIS

O Bot Builder por si só gerencia apenas o fluxo das conversações e entrega das mensagens ao usuário. Ele não tem nenhuma inteligência, serve apenas como um serviço que recebe uma mensagem de um usuário a partir de um canal e envia uma mensagem como resposta.

O LUIS é um Serviço Cognitivo de Processamento de Linguagem Natural. Ele utiliza Aprendizagem de Máquina para, após treinado, identificar intenções e entidades numa frase. Para um BOT, um serviço como o LUIS é essencial para interagir com o usuário de maneira natural.

O BOT Builder disponibiliza um tipo de diálogo exclusivo para o LUIS, facilitando ainda mais a integração e criação de um bot inteligente.

Para saber mais acesse <https://www.luis.ai/>

O que é o Bot Emulator

O Bot Emulator é um simulador de Chat. É um software Open Source criado para a Microsoft, para se conectar ao Bot Builder, simulando um Chat.

Ele será o seu melhor amigo no desenvolvimento de um Bot.

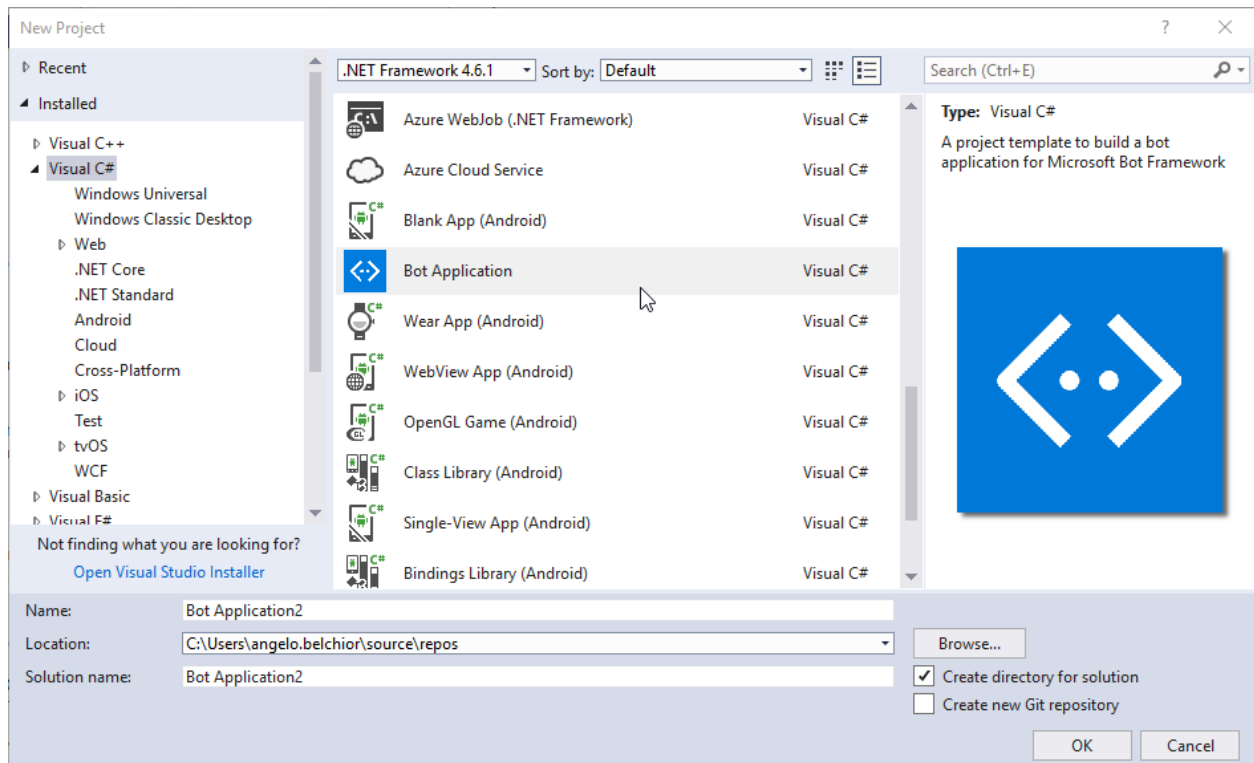
Com ele é possível analisar todas as mensagens enviadas e recebidas, enviar anexos, testar respostas e fluxos de conversação, além de simular entrada e saída de membros de uma conversa.

Você pode ainda enviar "sinais" (ou tipo de mensagens) ao Bot Builder como "Digitando" e "Ping", que são úteis para testes.

Baixe-o aqui: https://docs.microsoft.com/pt-br/bot-framework/bot-service-debug-emulator?WT.mc_id=CSE_diplomado_handson

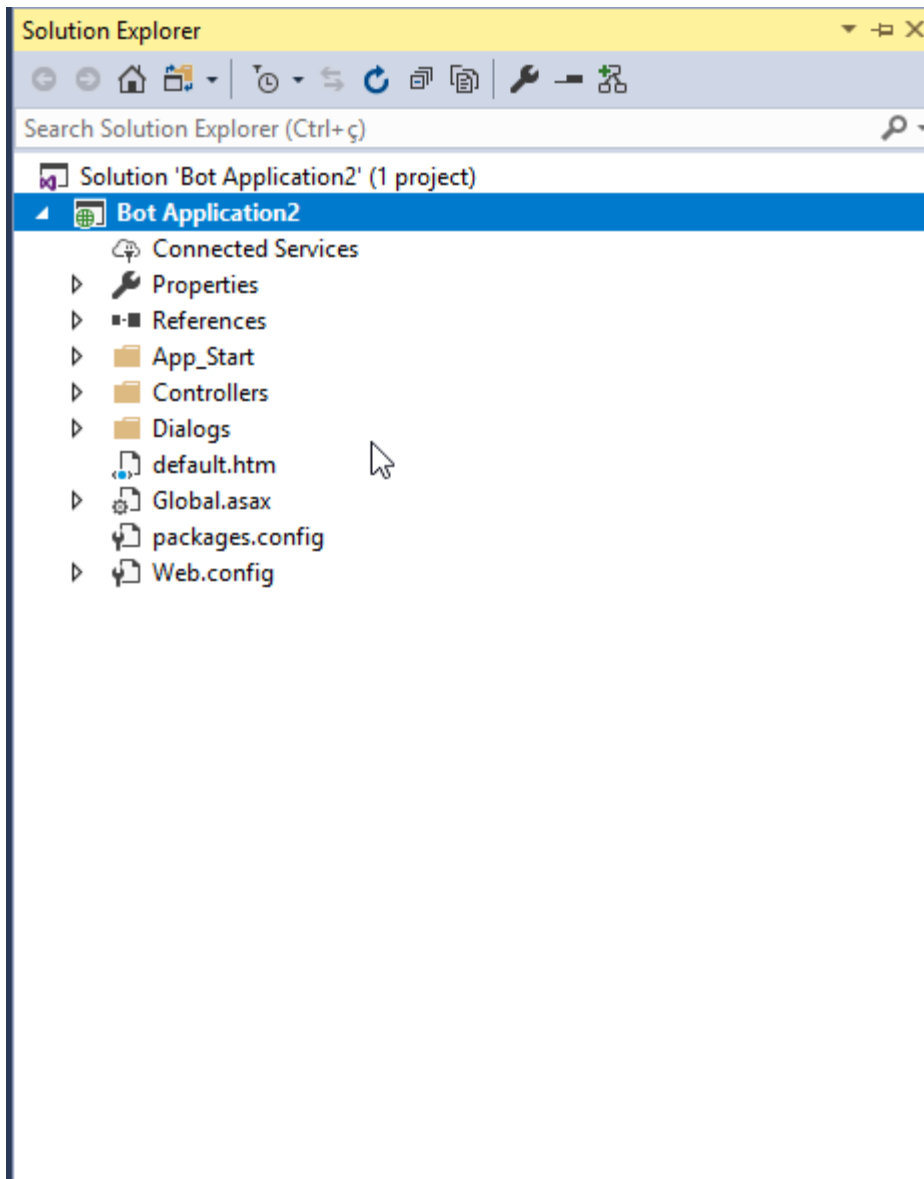
Criando um Projeto de Bot

No Visual Studio vá em Arquivo, Novo, Projeto.
Uma janela se abrirá.



Selecione Bot Application. Caso o template não apareça, é necessário instalá-lo. Vá para o tópico “Instalando as Extensões do Visual Studio” e siga as instruções.

Após a criação do Projeto, a estrutura de arquivos será essa:



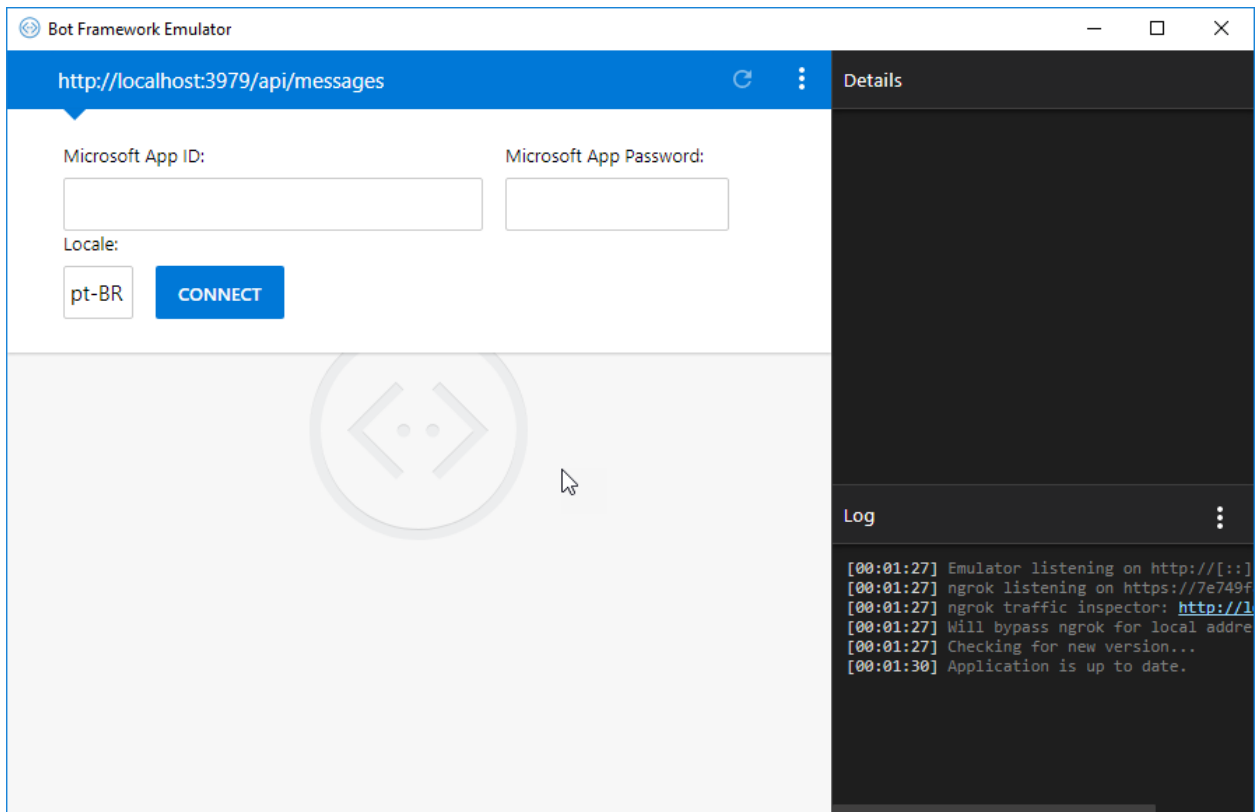
Se você já conhece Asp.Net MVC, com certeza estará em casa. O Bot Builder nada mais é do que um framework que roda em cima de um projeto Asp.Net Web API.

Não existe uma estrutura rígida para pastas e arquivos, mas por padrão, todos os nossos diálogos ficarão na pasta Dialog.

Clique com o botão direito em cima da “Solution” e selecione “Rebuild Solution”. Dessa maneira, as bibliotecas utilizadas pelo projeto serão baixadas.

Aperte F5. Se tudo der certo e nada der errado o projeto será executado e a janela de um navegador será aberta.

Feito isso, abra o Bot Emulator.



Na caixa de texto azul escrita “Enter your endpoint URL” coloque o endereço do site aberto.

Geralmente esse endereço é <http://localhost:3979/api/messages>.

Note que os campos Microsoft App ID e Microsoft App Password deverão ficar em brancos até então.

Vai ter um módulo específico para publicação e teste do seu Bot.

Clique em Connect.

Digite algo bem criativo no campo “Digite sua mensagem”. Não vale escrever “Hello World” e nem “Palmeiras não tem Mundial” ;p

O bot vai dar uma resposta dizendo a quantidade de caracteres do texto.

Seu projeto já está rodando. \o/

Eu considero a janela de Log e Details as partes mais importantes do Bot Emulator:

```
Details

{
  "type": "message",
  "timestamp": "2018-01-31T02:05:17.404Z",
  "localTimestamp": "2018-01-31T00:05:17-02:00",
  "serviceUrl": "http://localhost:45505",
  "channelId": "emulator",
  "from": {
    "id": "jnmnm0ig2e4i93541",
    "name": "Bot"
  },
  "conversation": {
    "id": "62h84n19bb51"
  },
}
```

```
Log

[00:01:27] ngrok listening on https://7e749fa5.ngrok.io
[00:01:27] ngrok traffic inspector: http://localhost:4040
[00:01:27] Will bypass ngrok for local addresses
[00:01:27] Checking for new version...
[00:01:30] Application is up to date.
[00:05:07] -> POST 200 [conversationUpdate]
[00:05:07] -> POST 200 [conversationUpdate]
[00:05:17] Warning: The Bot Framework State API is not recommended for p
[00:05:17] <- GET 200 getConversationData
[00:05:17] <- GET 200 getPrivateConversationData
[00:05:17] <- GET 200 getUserData
[00:05:17] <- POST 200 Reply[message] You sent olá mundo which was 9 cha
[00:05:17] <- POST 200 setConversationData
[00:05:17] <- POST 200 setUserData
[00:05:17] <- POST 200 setPrivateConversationData
[00:05:17] -> POST 200 [message] olá mundo
[00:05:18] Warning: The latest bot SDK version is 3.12.2 but the bot is
```

Essas informações auxiliam e muito o desenvolvimento e o “debug” do Bot. Nesse ponto, temos toda a infraestrutura necessária para desenvolver nosso Bot. Daqui em diante você vai ter contato com serviços cognitivos e fluxo de conversação. Na parte de serviços cognitivos, vai ter um módulo específico para tratar deles, então não se preocupe, você vai poder dar mais inteligência ao seu bot, fazendo com que ele reconhece imagem, texto, voz, etc.

Aprenda ainda mais!

Confira o repertório de artigos técnicos da Microsoft e tenha acesso a conteúdos exclusivos produzidos pelo nosso time de especialistas. [Acesse já!](#)