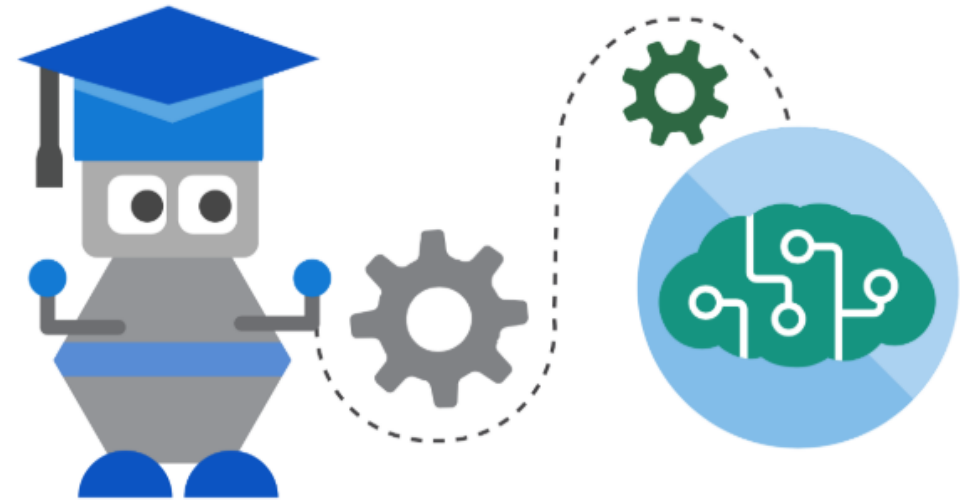


Xamarin



Bot Framework



Apresentação

Nouara Cândida Xavier

Eng. De Controle e Automação

Desenvolvedora de software na True Information Technology

E-mail: nouaracandida@gmail.com

LinkedIn: www.linkedin.com/in/nouaracandidaxavier/

GitHub: github.com/NouaraCandida

Instagram: @nouaradeveloper



Agenda

- ✓ Xamarin – Definições
- ✓ Bot Framework – Definições
- ✓ Ciclo de implementação: Inserindo um bot em seu aplicativo Xamarin
- ✓ Dicas



Definições

O que é Xamarin?

- Xamarin.Forms, Xamarin.iOS ou Xamarin.Android;
- Utiliza features do C# e do .NET no desenvolvimento destes aplicativos, como async/await e lambdas por exemplo.
- Conheça os S.O!



Definições

Definições

O que é Bot?

- Ferramenta usada para implementar o conceito de conversação como plataforma de acordo com o serviço cognitivo implementado assim produzindo diálogos com o usuário.



Definições

Definições



O que é Bot Framework?

Língua

Um serviço baseado em aprendizado de máquina para criar experiências de linguagem natural. Crie rapidamente modelos personalizados prontos para a empresa que melhoram continuamente.



Criador QnA

Publique uma pergunta simples e responda ao bot com base em URLs de FAQ existentes, documentos estruturados e manuais de produtos.



Discurso

Ouçã comandos, identifique oradores individuais, traduza e responda a clientes em linguagem natural com uma voz de marca.



Visão

Reconhecer rostos, moderar conteúdo e indexar imagens e vídeos com serviços de visão computacional.



Procurar

Pesquise em domínios definidos ou na Web para dados de clientes, status de inventário, resultados de pesquisa, registros de transações e muito mais.



Por que?

- Mercado mobile: US\$139 bilhões até 2021;
- A Caixa fez um bot teste para um feirão de imóveis, em que o assistente respondia dúvidas e facilitava algumas cotações. Em apenas um mês, o bot teve 1 milhão de interação e concluiu 35 mil cotações.
- O Banco do Brasil tem vários bots e alguns deles são lançados internamente antes de ir para o público final. Um deles é a Norminha, uma assistente criada para tirar dúvidas de funcionários sobre normas internas do banco e, através de 29 temas, reduziu essa demanda interna em 30% desde jan/2017.
- O Banco Original tem um bot no seu aplicativo que, além de tirar dúvidas, permite algumas transações. De mai ~ jun/2018, o bot teve uma retenção de 63% das interações através de árvores de navegação.



Vamos Implementar?





Bot – Pré Requisito

- Visual Studio [2017](#)
- Bot Framework SDK v4 template for [C#](#)
- Bot Framework [Emulator](#)
- Conhecimento em [ASP.Net Core](#) e programação assíncrona





Bot Builder V4 SDK Template for Visual Studio

Microsoft | 43,063 installs | 📄 84,945 downloads | ★★☆☆☆ (20) | Free

Templates for Microsoft Bot Framework v4. Will let you quickly set up a conversational AI bot using core AI capabilities.

Download

[Overview](#) [Q & A](#) [Rating & Review](#)

Microsoft Bot Builder V4 Templates

The Microsoft Bot Builder V4 Templates are available for .NET, and will let you quickly set up a conversational AI bot using core AI capabilities. They are available as a [VSIX](#) package.












About

Microsoft Bot Builder V4 Templates will help you build new conversational AI bots using the [Microsoft Bot](#)

Ca
Te
Ta
Bo
bc
bc



▼ Assets 11

 BotFramework-Emulator-4.3.3-linux-i386.ApplImage	69.5 MB
 BotFramework-Emulator-4.3.3-linux-x86_64.ApplImage	66.5 MB
 BotFramework-Emulator-4.3.3-mac.dmg	58.3 MB
 BotFramework-Emulator-4.3.3-mac.zip	67.3 MB
 BotFramework-Emulator-4.3.3-windows-setup.exe	40.1 MB
 latest-linux-ia32.yml	279 Bytes
 latest-linux.yml	283 Bytes
 latest-mac.yml	255 Bytes
 latest.yml	346 Bytes
 Source code (zip)	
 Source code (tar.gz)	

Recent

Installed

Visual C#

Get Started

Windows Universal

Windows Desktop

Web

.NET Core

.NET Standard

Android

Apple TV

Apple Watch

Cloud

AzureBotService

Cross-Platform

iOS Extensions

iPhone & iPad

Test

WCF

Not finding what you are looking for?

[Open Visual Studio Installer](#)

Sort by: Default



CoreBot

Visual C#



EchoBot

Visual C#



EmptyBot

Visual C#

Search (Ctrl+E)

**Type:** Visual C#

Core Bot template for Microsoft Bot Builder V4 SDK.

Name:

CoreBot1

Location:

C:\Users\Nouara\Documents\Visual Studio 2017\Projects\HelloTDC2019

Browse...

OK

Cancel

BOT EXPLORER



Welcome

Live Chat (developme... x

ENDPOINT +

development

SERVICES ↑↓ +

Restart conversation | v

Save transcript

http://localhost:3978/api/messages

INSPE

Click o

Add an Endpoint for your bot



Endpoint url*

https://www.helloBotFloripa2019-02.webazure.net/api/mess.

Name*

production

Application Id

Optional

Application Password

Optional. For Microsoft Apps

Azure for US Government [Learn more.](#)

Azure Bot Service configuration



conversationUpdate e

Cancel

Save

Pick a publish target



App Service



App Service Linux



Azure Virtual Machines



IIS, FTP, etc



Folder

Azure App Service

Fully managed, and highly scalable cloud environment

- Create New
- Select Existing

[Advanced...](#)


Import Profile...

Publish

Cancel

Create App Service

Host your web and mobile applications, REST APIs, and more in Azure

 Microsoft account
nouara_rv@hotmail.com

App Name

HelloBot2019-0220190422110521

Subscription

Avaliação Gratuita

Resource Group


HelloTDC201920190422012821ResourceGroup (centralus) [New...](#)

Hosting Plan


HelloTDC201920190422012821Plan (Central US, S1) [New...](#)


Application Insights

None

 Using Application Insights is recommended for performance management and website analytics

Explore additional Azure services

 Create a SQL Database

 Create a storage account

Clicking the Create button will create the following Azure resources


App Service - HelloBot2019-0220190422110521

Export...

Create

Cancel

management and website analytics

 Deploying: Step 1...

```
Web App was published successfully https://
===== Build: 1 succeeded, 0 failed, 0
===== Publish: 1 succeeded, 0 failed,
```


- Página inicial
- Painel
- Todos os serviços
- FAVORITOS
- Todos os recursos
- Grupos de recursos
- Serviços de Aplicativos
- Aplicativo de Função
- Bancos de dados SQL
- Azure Cosmos DB
- Máquinas virtuais
- Balancedores de carga
- Contas de armazename...
- Redes virtuais
- Active Directory do Azure

- Grupo de recursos
- Pesquisar (Ctrl+/)
- Visão geral
 - Log de atividade
 - IAM (Controle de acesso)
 - Marcações
 - Eventos
- Configurações
- Início rápido
 - Implantações
 - Políticas
 - Propriedades
 - Bloqueios
 - Exportar modelo
- Gerenciamento de Custos





[+](#) Adicionar
 [≡](#) Editar colunas
 [🗑️](#) Excluir grupo de recursos
 [🔄](#) Atualizar
 [➔](#) Mover
 [🔖](#) Atribuir marcações

Assinatura ([alterar](#)) Implantações
3 Êxito
 Avaliação [Gratuita](#)




ID da Assinatura
fa17094c-7886-4690-b026-84a2d7cc7aa6

Marcações ([alterar](#))
[Clique aqui para adicionar marcações](#)







4 itens Mostrar os tipos ocultos ⓘ

<input type="checkbox"/>	NOME ↑↓	TIPO ↑↓	LOCALIZAÇÃO ↑↓
<input type="checkbox"/>	 HelloBot2019-0220190422110521	Serviço de Aplicativo	Central US
<input type="checkbox"/>	 HelloTDC2019	Registro de Canais de Bot	global
<input type="checkbox"/>	 HelloTDC201920190422012821	Serviço de Aplicativo	Central US
<input type="checkbox"/>	 HelloTDC201920190422012821Plan	Plano do Serviço de Aplicativo	Central US

<< Página inicial > Nova > Bot Channels Registration

 Criar um recurso Página inicial Painel Todos os serviços

★ FAVORITOS

 Todos os recursos Grupos de recursos Serviços de Aplicativos Aplicativo de Função Bancos de dados SQL Azure Cosmos DB

Bot Channels Registration

Microsoft



Bot Channels Registration

Microsoft

[Criar](#)[Salvar para mais tarde](#)

Seu próprio Bot hospedado onde você quiser, registrado no Serviço de Bot do Azure. Crie, conecte, implante e gerencie os Bots para interagirem com os usuários em qualquer lugar, direto do aplicativo ou site para a Cortana, o Skype, o Messenger e muitos outros serviços.

Links Úteis

[Documentação](#)[Visão Geral da Solução](#)[Detalhes de Preço](#)

- + Criar um recurso
- 🏠 Página inicial
- 📊 Painel
- ☰ Todos os serviços
- ★ FAVORITOS
- 📊 Todos os recursos
- 📁 Grupos de recursos
- 🌐 Serviços de Aplicativos
- ⚡ Aplicativo de Função
- 🗄️ Bancos de dados SQL
- 🌌 Azure Cosmos DB
- 💻 Máquinas virtuais
- 🛡️ Balanceadores de carga
- 📁 Contas de armazename...
- 🔗 Redes virtuais
- 📁 Active Directory do Azure
- 📊 Monitor

Registro de Canais de Bot

Serviço de Bot

* Nome do bot ⓘ

* Assinatura

* Grupo de recursos

[Criar novo](#)

* Localização

Tipo de preço ([Exibir detalhes de preço completos](#))

Ponto de extremidade de mensagens

Application Insights ⓘ

Ativado

Desativado

[Criar](#)[Opções de automação](#)

- Log de atividade
- IAM (Controle de acesso)
- Marcações

Gerenciamento de bot

- Testar no Web Chat
- Análises
- Canais
- Configurações**
- Speech priming
- Preços do Serviço de Bot

Suporte + solução de problemas

Configuração

Ponto de extremidade de mensagens

`https://hellotdc201920190422012821.azurewebsites.net/api/messages`

* Microsoft App ID [\(Gerenciar\)](#) ⓘ

[Redacted]

Análises

Chave de Instrumentação do Application Insights ⓘ

Chave de instrumentação (chave do Azure Application Insights)

Chave de API do Application Insights ⓘ

Chave de API (chave de API do Application Insights gerada pelo usuário)

ID do Aplicativo do Application Insights ⓘ

ID do Aplicativo (ID do Aplicativo do Application Insights)

Configurações de Conexão OAuth

Não há configurações definidas

Propriedades

Nome

HelloTDC2019

ID do Aplicativo

[Redacted]

Segredos do Aplicativo

[Gerar Nova Senha](#) [Gerar Novo Par de Chaves](#) [Carregar Chave Pública](#)

Tipo	Senha/Chave Pública	Criado	
Senha	sjl*****	Apr 22, 2019 1:35:25 AM	Excluir
Senha	oZW*****	Apr 22, 2019 1:33:47 AM	Excluir

Plataforma

```
{
  "name": "HelloTDC2019",
  "description": "",
  "services": [
    {
      "type": "endpoint",
      "endpoint": "https://hellotdc201920190422012821.azurewebsites.net/api/messages",
      "name": "production",
      "id": "HelloTDC2019",
      "appId": "[REDACTED]",
      "appPassword": "[REDACTED]"
    },
    {
      "type": "endpoint",
      "appId": "",
      "appPassword": "",
      "endpoint": "http://localhost:3978/api/messages",
      "id": "1",
      "name": "development"
    }
  ],
  "padlock": "",
  "version": "2.0",
  "path": "C:\\Users\\Nouara\\Documents\\Visual Studio 2017\\Projects\\HelloTDC2019\\HelloTDC2019\\HelloTDC2019.bot",
  "overrides": null
}
```

**PUBLISH
NOVAMENTE!**

Vamos Implementar?







**Como usar no
seu App?**



- Visão geral
- Log de atividade
- IAM (Controle de acesso)
- Marcações
- Gerenciamento de bot
- Testar no Web Chat
- Análises
- Canais**
- Configurações
- Speech priming
- Preços do Serviço de Bot
- Suporte + solução de problemas

Conectar-se a canais

Nome	Integridade	Publicado	
 Direct Line	Corrida	--	Editar 
 Web Chat	Corrida	--	Editar 

[Obter códigos de inserção de bot](#)

Adicionar um canal de destaque

Direct Line







Direct Line

As conversas de linha direta são explicitamente abertas por clientes e podem ser executadas desde que o bot e o cliente participem e tenham credenciais válidas



- Visão geral
- Log de atividade
- IAM (Controle de acesso)
- Marcações
- Gerenciamento de bot
- Testar no Web Chat
- Análises
- Canais**
- Configurações
- Speech priming
- Preços do Serviço de Bot
- Suporte + solução de problemas

Conectar-se a canais

Nome	Integridade	Publicado	
 Direct Line	Corrida	--	Editar 
 Web Chat	Corrida	--	Editar 

[Obter códigos de inserção de bot](#)

Adicionar um canal de destaque

Xamarin



Xamarin

Baixar no NuGet a biblioteca :

Microsoft.Bot.Connector.DirectLine

Microsoft.Rest.ClientRuntime



Xamarin

MVVM

- **Model**
- **View –Model**
- **View**



Xamarin

```
public string botId = "HelloTDC2019";  
public string directLineSecret = "e215IuMHP"  
public DirectLineClient directLineClient;  
public Conversation MainConversation;  
public ChannelAccount Account;
```

- Implementar uma classe BotConnection.cs : Onde haverá a definição de conexão com o bot;
- botId: Determina qual bot está se comunicando com a aplicação
- directLine: Chave secreta fornecida no portal da azure
- directLineClient: Utilizado para instanciar a linha direta. Todas as operações utilizadas pelo bot são feitas aqui (iniciar conversa, envias e receber mensagens...)
- Instância da conversa entre Xamarin e Bot
- Identidade pra o Xamarin, identifica qual conta está se comunicando com o bot



Xamarin

```
public ChatBotViewModel()  
{  
    List = new ObservableCollection<Message>();  
    connection = new BotConnection("ME");  
    var messageTask = connection.GetMessageAsync(List);  
    SendCommand = new Command(Send);  
}
```

```
public BotConnection(string accountName)  
{  
    // Obtain a token using Direct Line Secret  
    var tokenResponse = new DirectLineClient(directLineSecret).  
        Tokens.GenerateTokenForNewConversation();  
  
    // Use token to create conversation  
    directLineClient = new DirectLineClient(tokenResponse.Token);  
    MainConversation = directLineClient.Conversations.StartConversation();  
    Account = new ChannelAccount { Id = accountName, Name = accountName };  
}
```



Xamarin

```
// the message and other activity types
if (turnContext.Activity.Type == ActivityTypes.Message)
{
    // Get the conversation state from the turn context.
    var state = await _accessors.CounterState.GetAsync(turnCon
    // Bump the turn count for this conversation.
    state.TurnCount++;
    if (!state.SaidHello)
    {
```

No bot!

Enviando Mensagem de TEXTO (ActivityTypes.Message) ao Bot

```
public void SendMessage(string message)
{
    Activity activity = new Activity
    {
        From = Account,
        Text = message,
        Type = ActivityTypes.Message
    };
    directLineClient.Conversations.PostActivity
        (MainConversation.ConversationId, activity);
}
```



Xamarin

Recebendo Mensagem! GetMessageAsync

```
public async Task GetMessageAsync(ObservableCollection<Message> collection)
{
    string watermark = null;
    while (true)
    {
        var activitySet = await directLineClient.Conversations.GetActivitiesAsync
            (MainConversation.ConversationId, watermark);
        watermark = activitySet?.Watermark;
        foreach (var activity in activitySet.Activities)
        {
            if (activity.From.Id == botId)
            {
                collection.Add( new Message
                {
                    Text = activity.Text,
                    Image = ImageSource.FromFile("Chatbot.png"),
                    IsTextIn = true,
                    MessageDateTime = DateTime.Now
                });
            }
        }
    }
}
```

- Parâmetro é um ObservableCollection<Message>;
- Com isto a listView será atualizada automaticamente;
- WATERMARK marca a última Activity que o app recebeu, com isto receberemos apenas a última mensagem que o bot envio, em um determinado ciclo de tempo. Então se o id da mensagem é “inédito” a mensagem irá para listview criada.



Obrigada!

Obrigada!

Nouara Cândida Xavier

Eng. De Controle e Automação - UFPel

Desenvolvedora de software at True Information
Technology

E-mail: nouaracandida@gmail.com

LinkedIn: www.linkedin.com/in/nouaracandidaxavier/



THE DEVELOPER'S
CONFERENCE