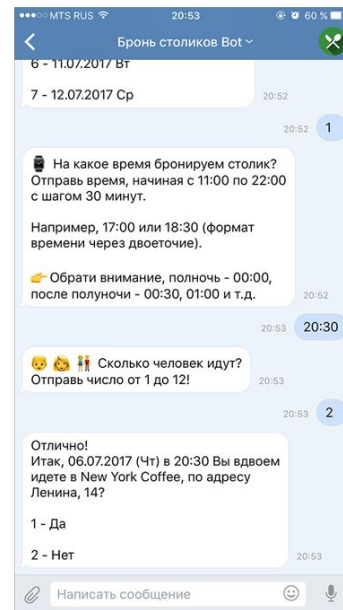
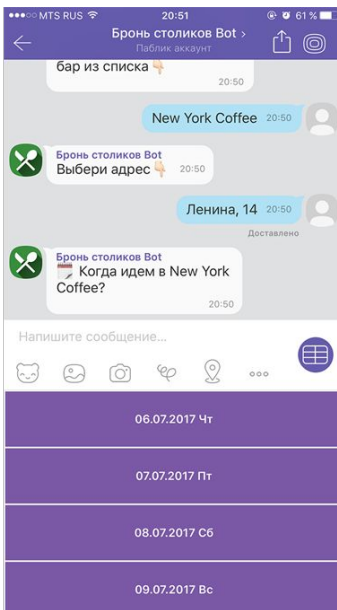


Привет!

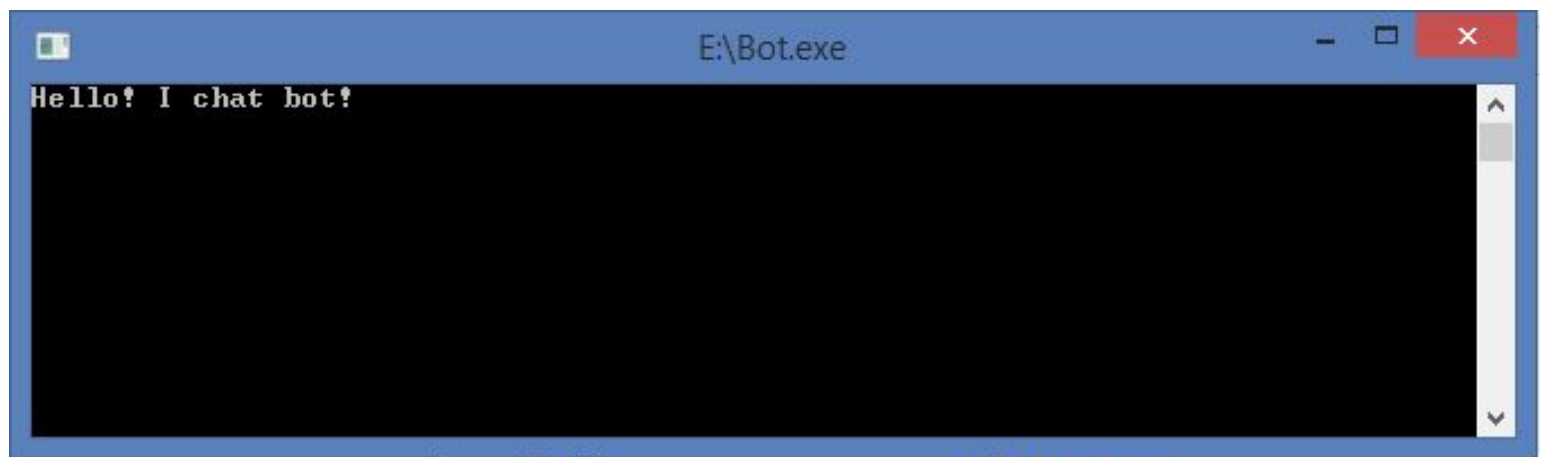
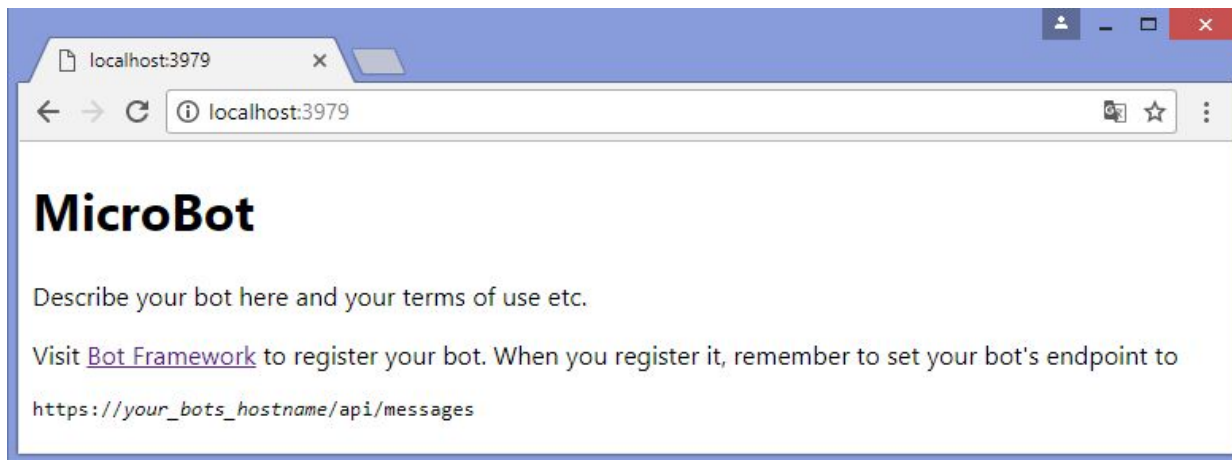
Булат  
Шаехов  
2 курс

# Что такое бот?

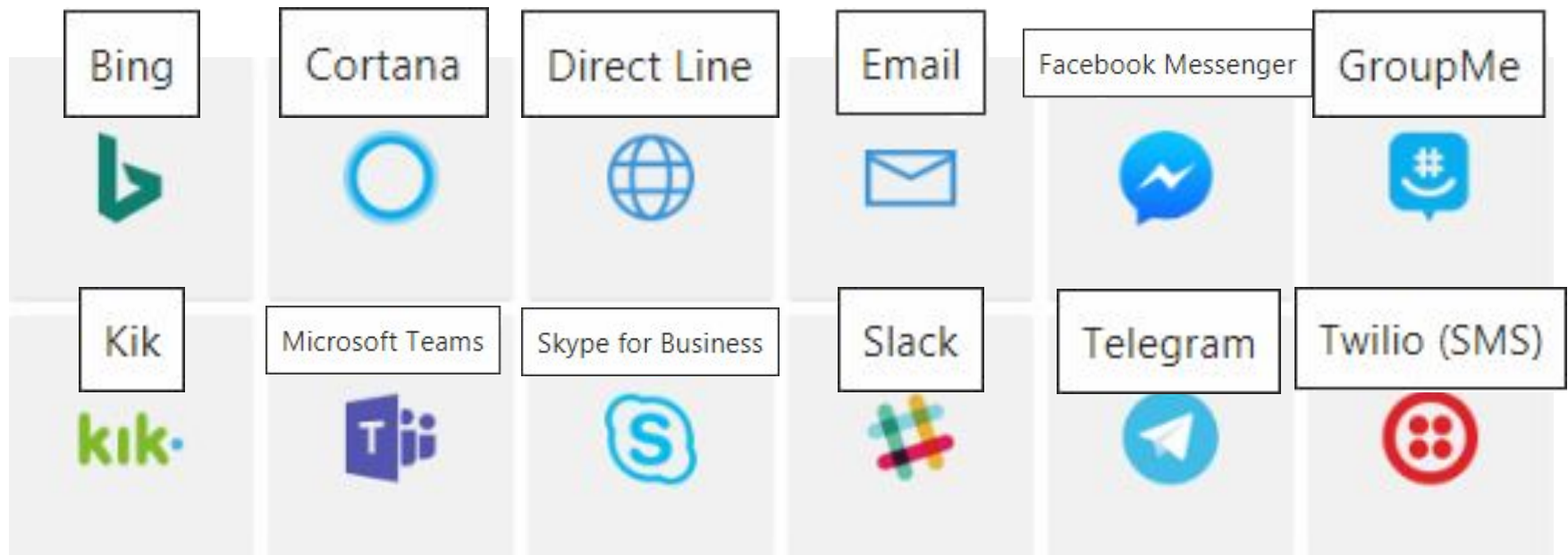
# Что такое чат бот?



# Бот со стороны сервера



# Платформы с ботами



# BotFramework



# Создание локального бота



<https://docs.microsoft.com/ru-ru/bot-framework/dotnet/bot-builder-dotnet-quickstart>

# Создание локального бота

## Prerequisites

Get started by completing the following prerequisite tasks:

1. Install [Visual Studio 2017](#) for Windows.
2. In Visual Studio, [update all extensions](#) to their latest versions.
3. Download the [Bot Application](#), [Bot Controller](#), and [Bot Dialog](#) .zip files. Install the templates by copying the .zip files to your Visual Studio 2017 project templates directory.

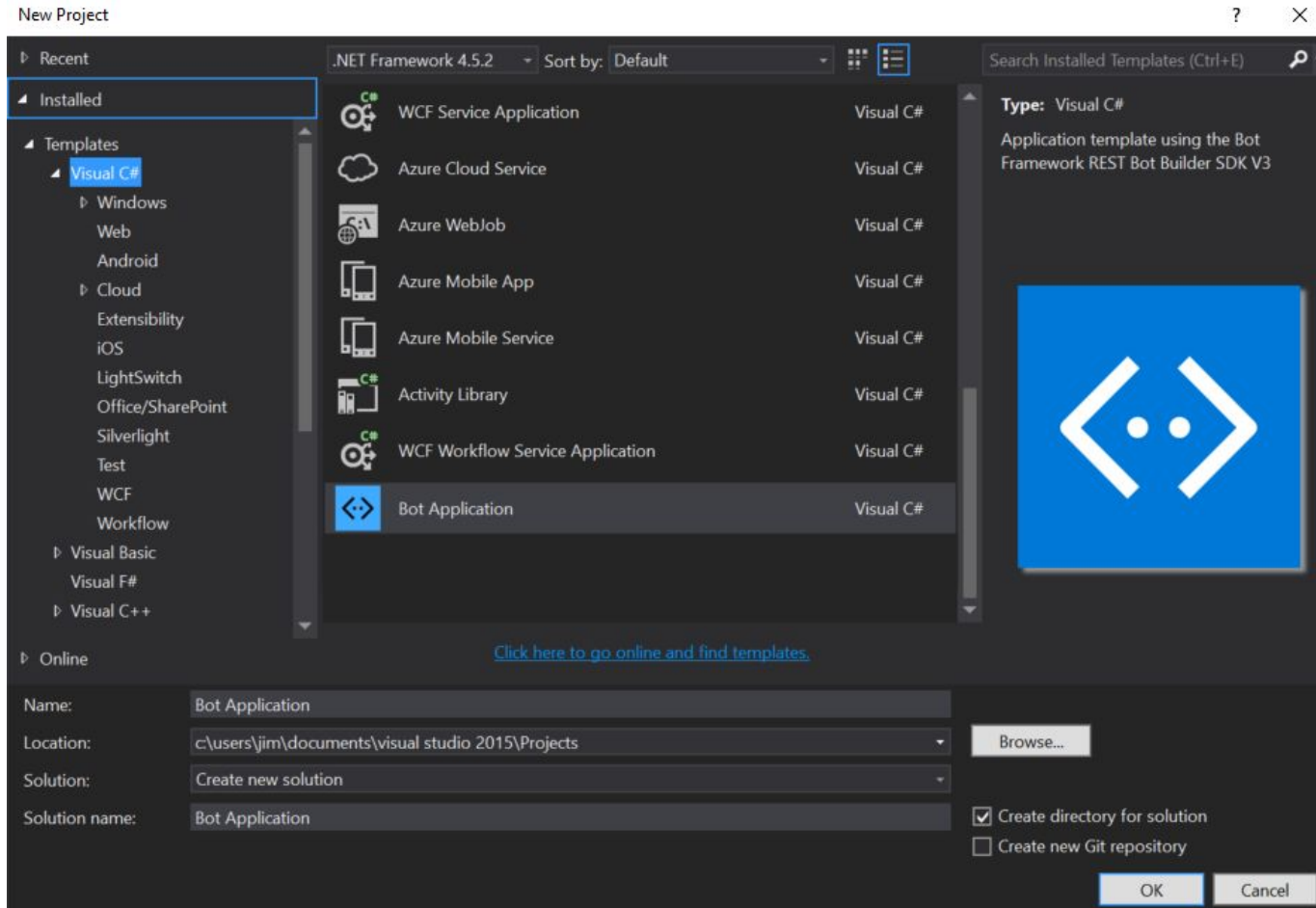
### Tip

The Visual Studio 2017 project templates directory is typically located at

```
%USERPROFILE%\Documents\Visual Studio 2017\Templates\ProjectTemplates\Visual C# \ .
```



# Создание локального бота



# Создание локального бота

The screenshot displays the Microsoft Visual Studio environment for creating a local bot. The main window shows the NuGet package manager interface for the 'NewMicroBot' project. The 'Microsoft.Bot.Builder' package is highlighted, with version 3.11.0 selected. The package description states: 'Microsoft Bot Builder is a powerful framework for constructing bots that can handle both freeform interactions and more guided ones where the possibilities are explicitly shown to the user. It is easy to use and leverages C# to provide a natural way to write bots.' The package is installed, and the version is set to 3.11.0. The 'Параметры' (Parameters) section is visible, and the 'Описание' (Description) section provides details about the framework's capabilities and features.

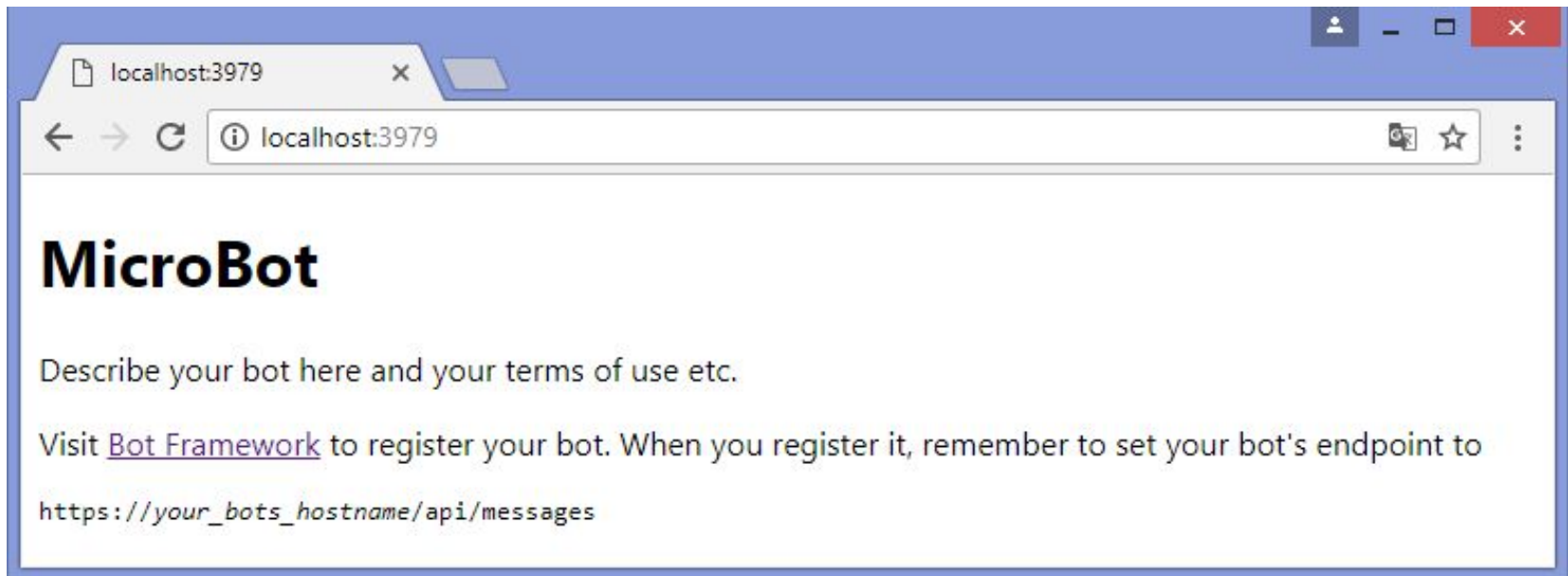
**High Level Features:**

- \* Powerful dialog system with dialogs that are isolated and composable.
- \* Built-in dialogs for simple things like Yes/No strings, numbers, enumerations.
- \* Built-in dialogs that utilize powerful AI frameworks like LUIS <http://luis.ai>.
- \* Bots are stateless which helps them scale.
- \* Form Flow for automatically generating a BC from a C# class for filling in the class and that supports help, navigation, clarification and confirmation.

The interface also shows the project structure on the right, including folders like 'App\_Start', 'WebApi', 'Controllers', 'Message', 'Dialogs', 'RootDialog', 'Global.asax', 'Global.asax', 'packages.config', and 'WebResource'. The code editor on the left shows the 'RootDialog.cs' file with the following code:

```
1 using System;
2 using Microsoft.Bot.Builder;
3 using Microsoft.Bot.Builder.Dialogs;
4 using Microsoft.Bot.Builder.Luis;
5
6 namespace NewMicroBot
7 {
8     [Serializable]
9     public class RootDialog : IDialog
10    {
11    }
12
13    public RootDialog()
14    {
15    }
16
17    public async Task<DialogContext> RunAsync(DialogContext dc)
18    {
19    }
20
21    public async Task<DialogContext> RunAsync(DialogContext dc, CancellationToken cancellationToken)
22    {
23    }
24
25    public async Task<DialogContext> RunAsync(DialogContext dc, CancellationToken cancellationToken, string userMessage)
26    {
27    }
28
29    public async Task<DialogContext> RunAsync(DialogContext dc, CancellationToken cancellationToken, string userMessage, string userLocale)
30    {
31    }
32 }
```

# Тестирование бота

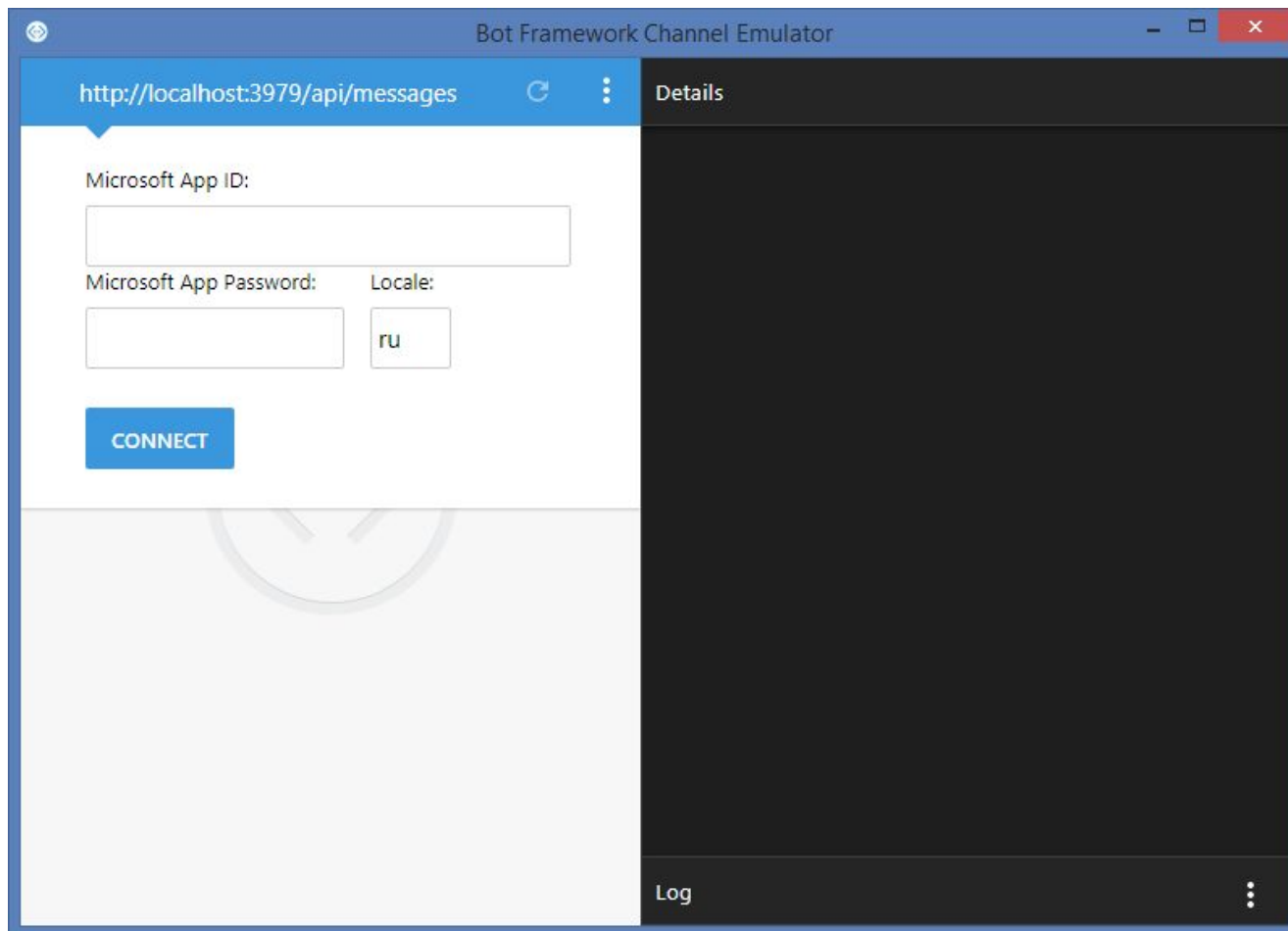


# Тестирование бота (эмулятор канала)

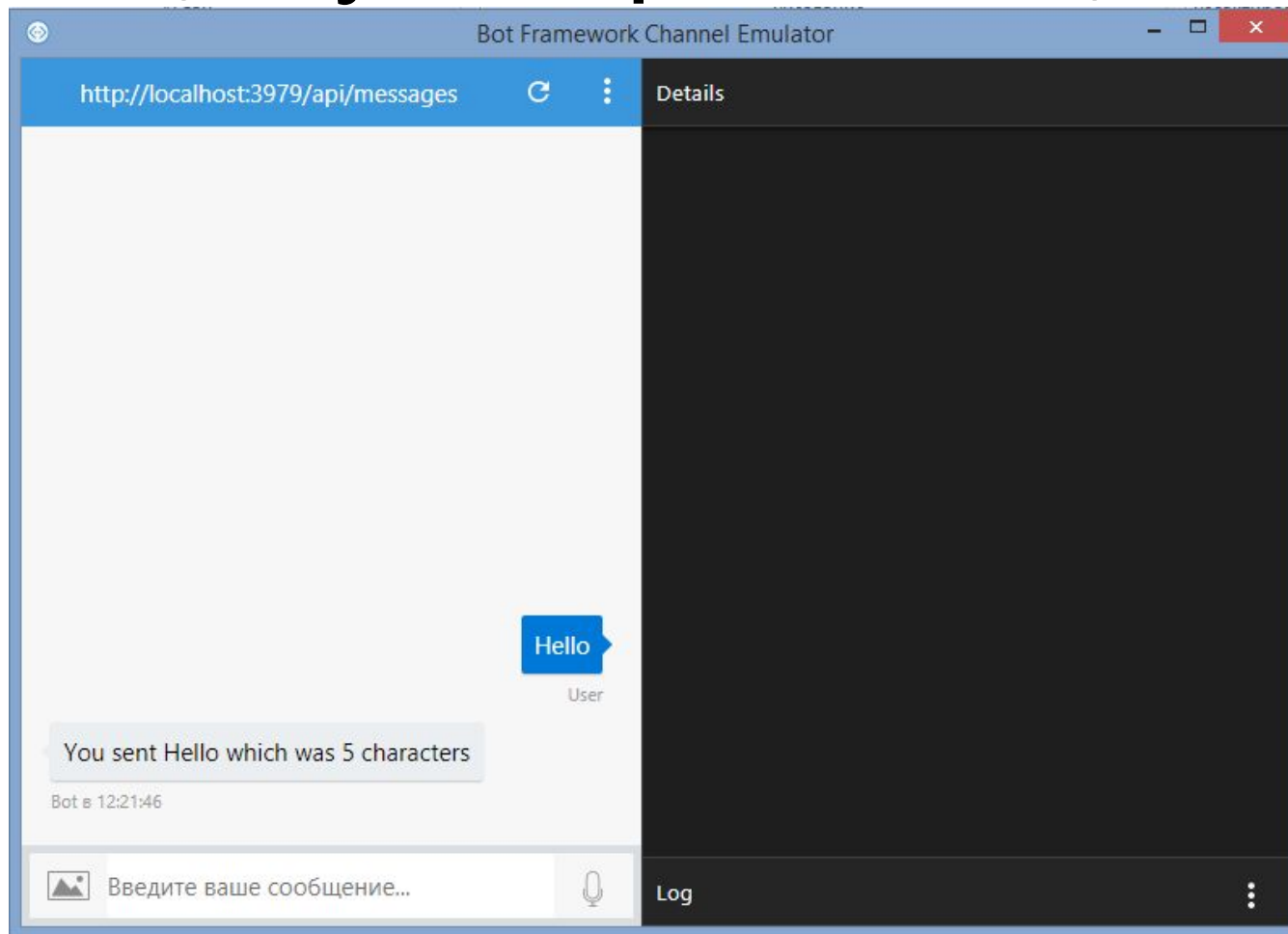


[https://vk.com/away.php?to=https%3A%2F%2Fgithub.com%2FMicrosoft%2FBotFramework-Emulator%2Freleases%2Fdownload%2Fv3.5.32%2Fbotframework-emulator-Setup-3.5.32.exe&cc\\_key=](https://vk.com/away.php?to=https%3A%2F%2Fgithub.com%2FMicrosoft%2FBotFramework-Emulator%2Freleases%2Fdownload%2Fv3.5.32%2Fbotframework-emulator-Setup-3.5.32.exe&cc_key=)

# Тестирование бота (эмулятор канала)



# Тестирование бота (эмулятор канала)



Немного практики...