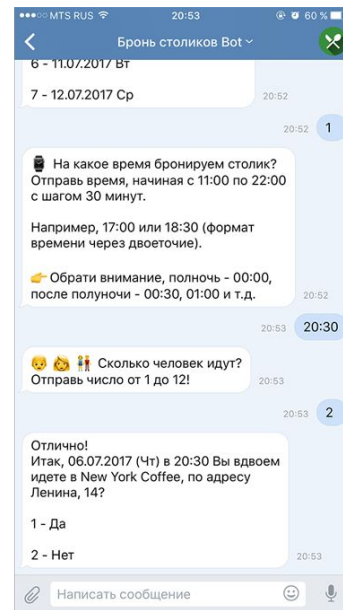
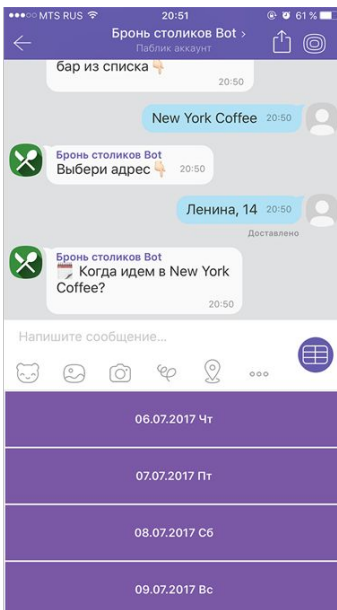


Привет!

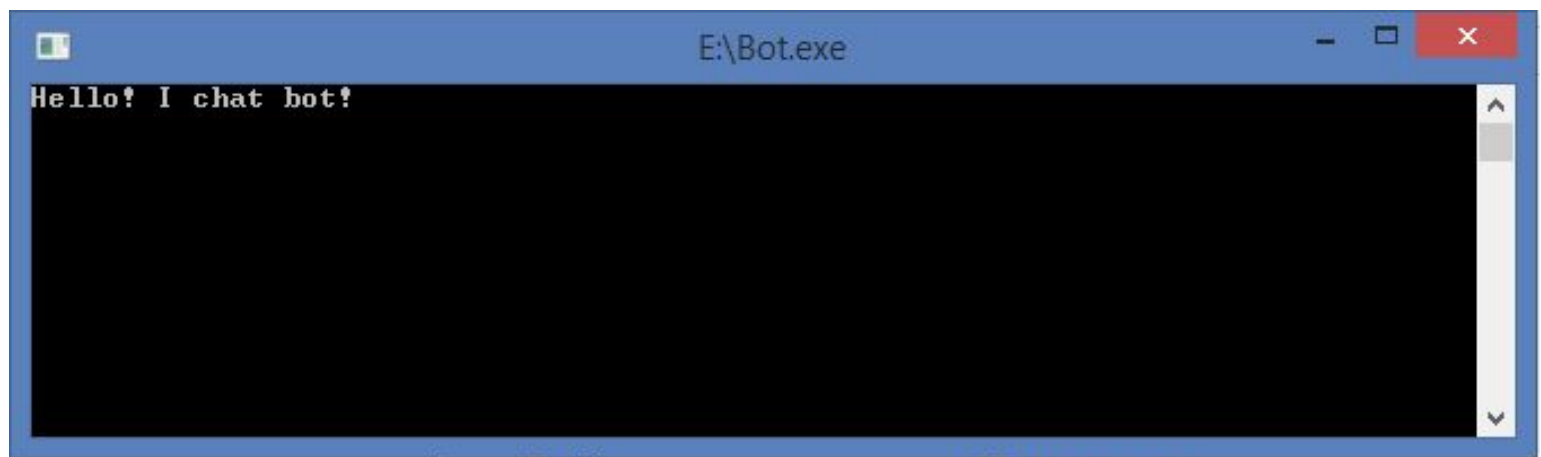
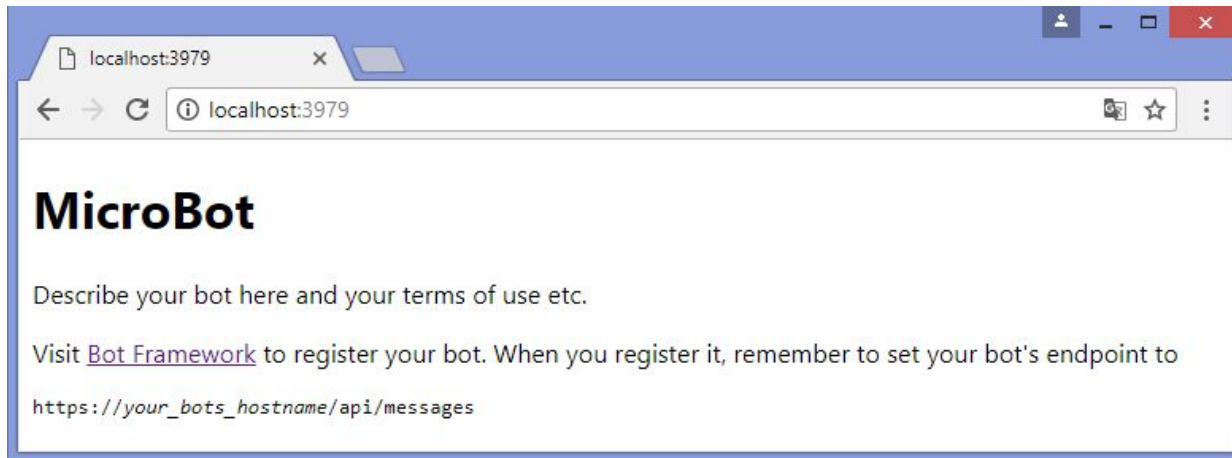
Булат
Шаехов
2 курс

Что такое бот?

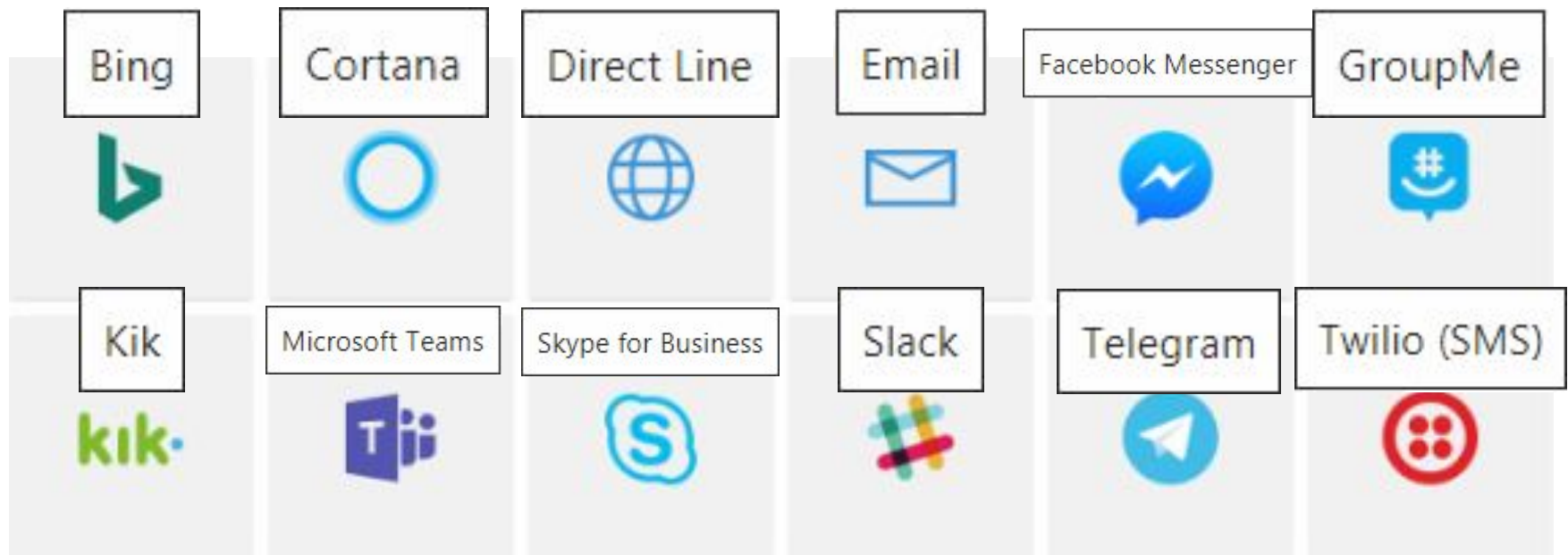
Что такое чат бот?



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:

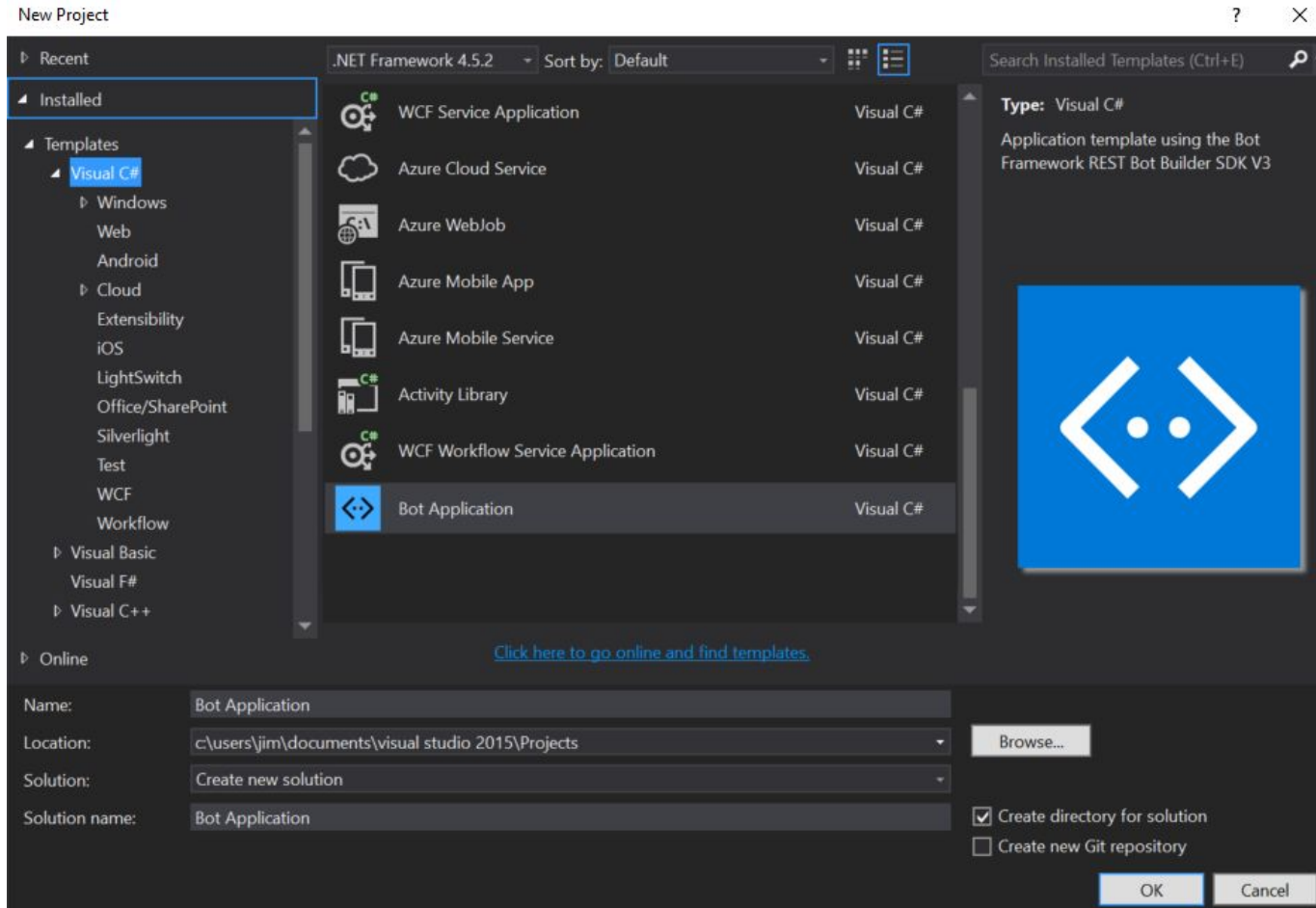
1. Install [Visual Studio 2017](#) for Windows.
2. In Visual Studio, [update all extensions](#) to their latest versions.
3. Download the [Bot Application](#), [Bot Controller](#), and [Bot Dialog](#) .zip files. Install the templates by copying the .zip files to your Visual Studio 2017 project templates 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 is titled "NewMicroBot - Microsoft Visual Studio". The menu bar includes options like "Файл", "Правка", "Вид", "Проект", "Сборка", "Отладка", "Команда", "Сервис", "Архитектура", "Тест", "Анализ", "Окно", and "Справка".

The code editor on the left shows the `RootDialog.cs` file with the following content:

```
1 using Microsoft.Bot.Builder.Dialogs;
2 using Microsoft.Bot.Builder.Luis;
3 using Microsoft.Bot.Builder.TemplateManager;
4 using Microsoft.Bot.Connector;
5
6 namespace NewMicroBot
7 {
8     [Serializable]
9     public class RootDialog : Dialog
10     {
11         public RootDialog(IDialogContext context)
12         {
13             // Add your own initialization here
14         }
15
16         public async Task ContinueAsync(IDialogContext context)
17         {
18             // Add your own logic here
19         }
20
21         public async Task StartAsync(IDialogContext context)
22         {
23             context.Wait(MessageReceived);
24         }
25
26         private async Task MessageReceivedAsync(IDialogContext context, IAwaitable<DialogResult> awaitable)
27         {
28             // Add your own logic here
29         }
30     }
31 }
```

The NuGet package manager window is open, showing the "Диспетчер пакетов NuGet: NewMicroBot" interface. The "Обновления" (Updates) tab is active, displaying a list of installed packages:

- Microsoft.Bot.Builder** автор: Microsoft v3.11.0 (Selected)
- Net.WebApi** автор: Microsoft v5.2.3
- Net.WebApi.Client** автор: Microsoft v5.2.3
- Net.WebApi.Core** автор: Microsoft v5.2.3
- Net.WebApi.WebHost** автор: Microsoft v5.2.3

The description for **Microsoft.Bot.Builder** v3.11.0 is shown:

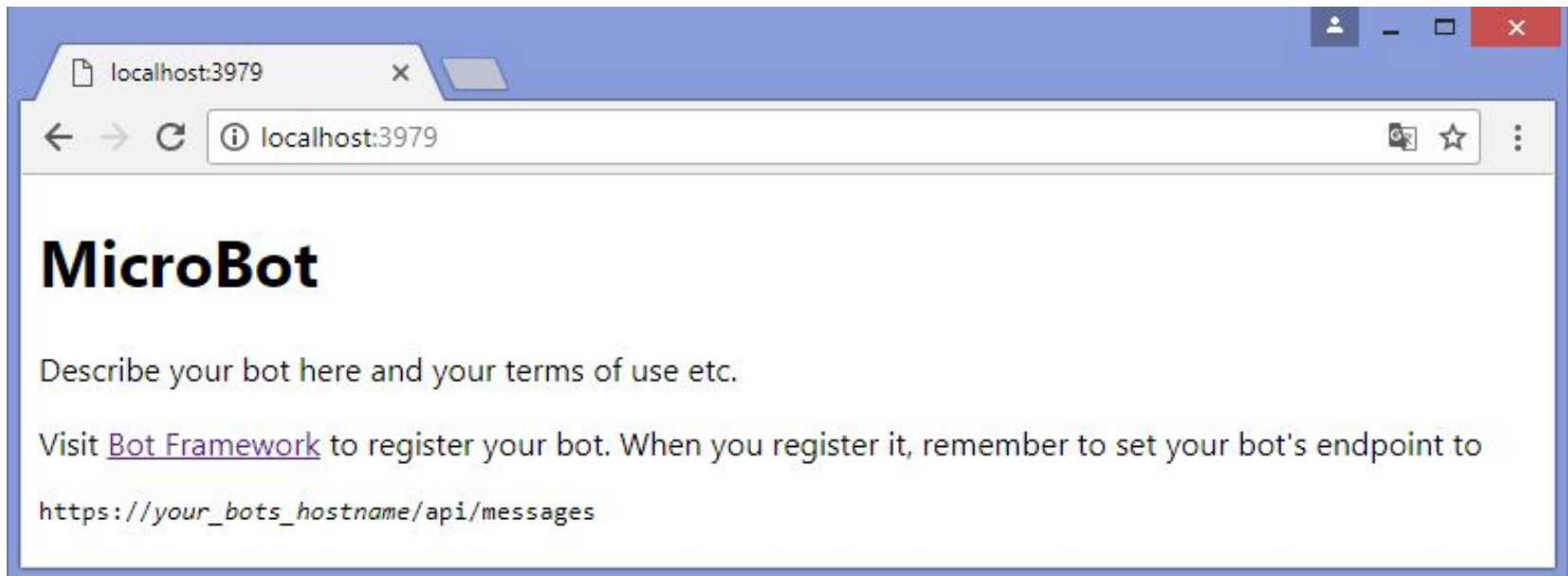
Microsoft.Bot.Builder is a framework for easily building stateless bots that can handle both freeform interactions and more guided ones where the possibilities are explicitly shown to the user. It is easy to use and leverages C# to provide a natural way to write bots.

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 `Bot` from a C# class for filling in the class and that supports help, navigation, clarification and confirmation.

The status bar at the bottom indicates "Готово" (Ready) and includes a button to "Добавить в систему управления версиями" (Add to version control system).

Тестирование бота

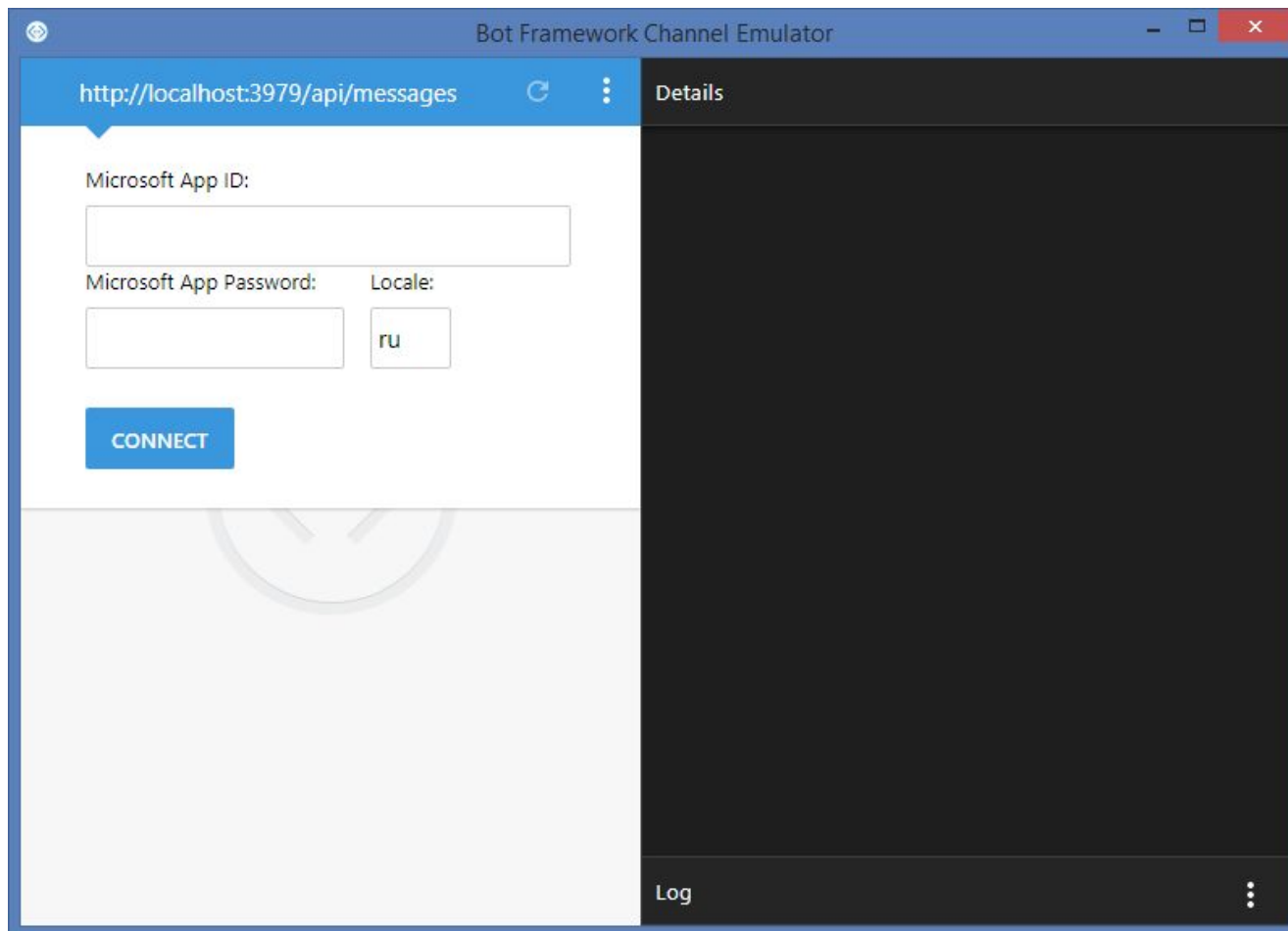


Тестирование бота (эмулятор канала)

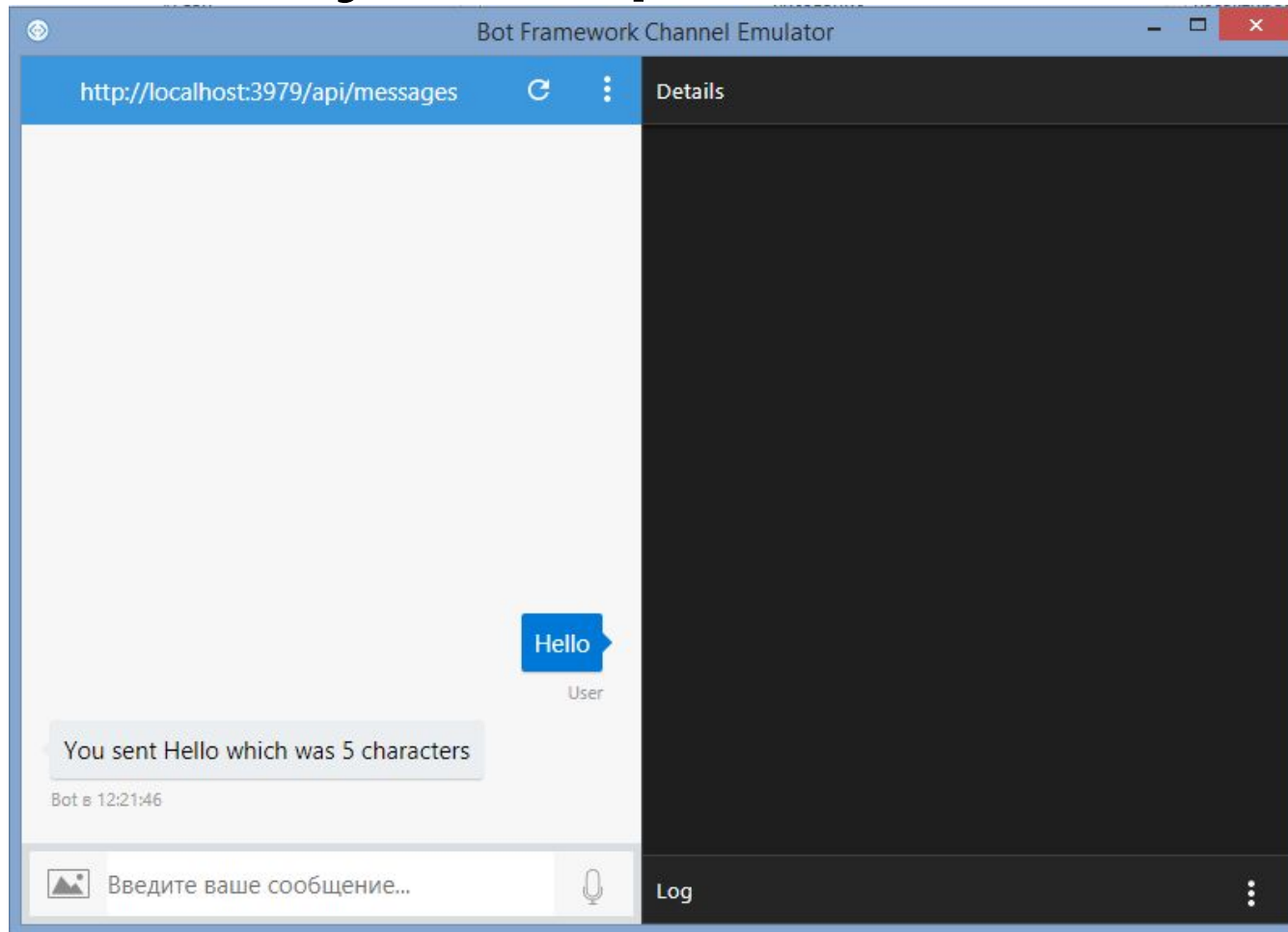


https://vk.com/away.php?to=https%3A%2F%2Fgithub.com%2FMicrosoft%2FBotFramework-Emulator%2Freleases%2Fdownload%2Fv3.5.32%2Fbotframework-emulator-Setup-3.5.32.exe&cc_key=

Тестирование бота (эмулятор канала)



Тестирование бота (эмулятор канала)



Немного практики...