

# Telegram Bot Framework dan Azure AI Translator

**Junindar, ST, MCPD, MOS, MCT, MVP**

*junindar@gmail.com*

***Lisensi Dokumen:***

*Copyright © 2003 IlmuKomputer.Com*

*Seluruh dokumen di IlmuKomputer.Com dapat digunakan, dimodifikasi dan disebarkan secara bebas untuk tujuan bukan komersial (nonprofit), dengan syarat tidak menghapus atau merubah atribut penulis dan pernyataan copyright yang disertakan dalam setiap dokumen. Tidak diperbolehkan melakukan penulisan ulang, kecuali mendapatkan ijin terlebih dahulu dari IlmuKomputer.Com.*

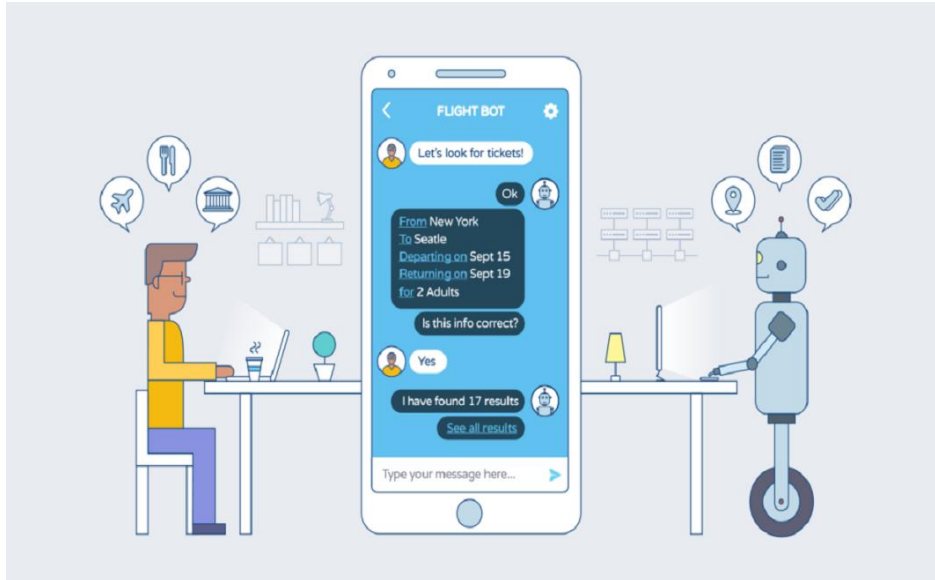
<http://junindar.blogspot.com>

## Abstrak

Chatbot atau chatterbot adalah sebuah layanan obrolan robot/tokoh virtual dengan kecerdasan buatan atau AI (Artificial Intelligent) yang menirukan percakapan manusia melalui pesan suara, obrolan teks ataupun keduanya. Chatbot menggunakan berbagai teknologi yang mutakhir di dalamnya, seperti Artificial Intelligence (AI), Machine Learning, Deep Learning, dan Natural Language Processing (NLP). Machine Learning dapat di aplikasikan sebagai mesin yang mempelajari, menganalisa dan mengenal bermacam-macam Bahasa pada chatbot, sementara Natural Language Processing (NLP) memiliki kemampuan untuk mengerti dan memahami Bahasa manusia lalu memberikan respon yang sesuai dengan Bahasa yang digunakan pengguna chatbot.

## Pendahuluan

Chatbot atau chatterbot adalah sebuah layanan obrolan robot/tokoh virtual dengan kecerdasan buatan atau AI (Artificial Intelligent) yang menirukan percakapan manusia melalui pesan suara, obrolan teks ataupun keduanya.



Pada dasarnya bots bekerja dengan cara melihat kata kunci dalam data yang masuk dan membalasnya dengan kata kunci yang paling cocok, atau pola kata-kata yang paling mirip dari basis data tekstual. Artinya, jika pengguna mengirim suatu permintaan maka bots akan membalasnya dengan respon yang spesifik sesuai dengan kata kunci yang dikirim.

## Telegram Bot Framework

Telegram Bot Framework adalah sebuah framework yang memungkinkan para programmer untuk membuat bot pada Telegram dengan menggunakan platform .NET. Framework ini menyediakan tool dan fitur yang mempermudah pembuatan dan implementasi bot Telegram menggunakan bahasa pemrograman seperti C#.

Dengan menggunakan Telegram Bot Framework, kita dapat mengakses API Telegram, memproses pesan, menjawab permintaan dari pengguna, dan mengelola berbagai fungsi bot, seperti mengirim pesan teks, media, menangani perintah, dan banyak lagi.

Terdapat banyak fasilitas pada framework ini, sehingga kita dapat mengembangkan bot yang powerfull dan fleksibel dengan menyediakan berbagai fungsi yang dapat digunakan untuk membuat bot yang sesuai dengan kebutuhan penggunaannya.

Telegram Bot Framework menggunakan beberapa library atau pustaka tertentu untuk memudahkan interaksi dengan API Telegram dan menyediakan berbagai fitur bagi para programmer. salah satunya adalah Telegram.Bot. Telegram.Bot adalah library utama yang digunakan dalam framework ini. Telegram.Bot merupakan library resmi dari Telegram untuk C#/.NET yang menyediakan akses ke API Telegram. library ini memungkinkan pengembang untuk membuat, mengelola, dan berinteraksi dengan bot Telegram menggunakan berbagai fitur yang disediakan oleh Telegram.

Azure AI Translator adalah layanan terjemahan otomatis yang disediakan oleh Microsoft Azure. Layanan ini menggunakan kecerdasan buatan (AI) untuk menerjemahkan teks dan ucapan dari satu bahasa ke bahasa lain secara otomatis. Ini dapat digunakan untuk menerjemahkan teks pada aplikasi, situs web, atau platform lainnya secara real-time.

Azure AI Translator menggunakan teknologi terbaru dalam pemrosesan bahasa alami dan jaringan saraf untuk meningkatkan akurasi terjemahan antar bahasa. Layanan ini mendukung sejumlah besar bahasa dan dapat diintegrasikan dengan mudah ke dalam berbagai aplikasi dan platform melalui API (Antarmuka Pemrograman Aplikasi) yang disediakan oleh Azure.

Selain menerjemahkan teks, Azure AI Translator juga dapat digunakan untuk menerjemahkan suara, membaca teks dari gambar melalui OCR (Optical Character Recognition), dan menyediakan terjemahan berbasis konteks yang lebih akurat.

Azure Text Translation adalah layanan dalam Microsoft Azure yang menyediakan kemampuan untuk menerjemahkan teks dari satu bahasa ke bahasa lain secara otomatis. Layanan ini memungkinkan pengguna untuk menerjemahkan teks dari dan ke berbagai bahasa menggunakan API (Antarmuka Pemrograman Aplikasi) yang disediakan oleh Azure.

Dengan menggunakan Azure Text Translation, Anda dapat mengintegrasikan kemampuan terjemahan ke dalam aplikasi, situs web, atau platform lainnya. Layanan ini menggunakan teknologi pemrosesan bahasa alami dan kecerdasan buatan untuk memberikan terjemahan yang akurat antar bahasa.

Azure Text Translation mendukung banyak bahasa yang berbeda dan memiliki kemampuan untuk menerjemahkan teks dalam konteks tertentu, meningkatkan akurasi terjemahan untuk pengalaman pengguna yang lebih baik. Ini memungkinkan organisasi untuk memperluas jangkauan internasional dan menyediakan konten multibahasa kepada pengguna mereka dengan lebih mudah.

Untuk menggunakan Azure AI Translator terlebih dahulu kita harus membuat *Translator resource* pada Azure Portal. Azure AI Translator adalah translation service berbasis cloud yang merupakan bagian dari rangkaian layanan Azure AI. Semua permintaan API ke layanan Azure AI memerlukan URL *endpoint* dan *key* untuk mengautentikasi akses. Jika belum memiliki Azure Account, kita dapat membuatnya terlebih dahulu disini <https://azure.microsoft.com/en-us/free/> .

**Create Translator** ...

multi-language support such as website localization, e-commerce, customer support, messaging applications, internal communication, and more.

**Project Details**

Subscription \*

Resource group \*  [Create new](#)

**Instance Details**

Region \*

Name \*

Pricing tier \*

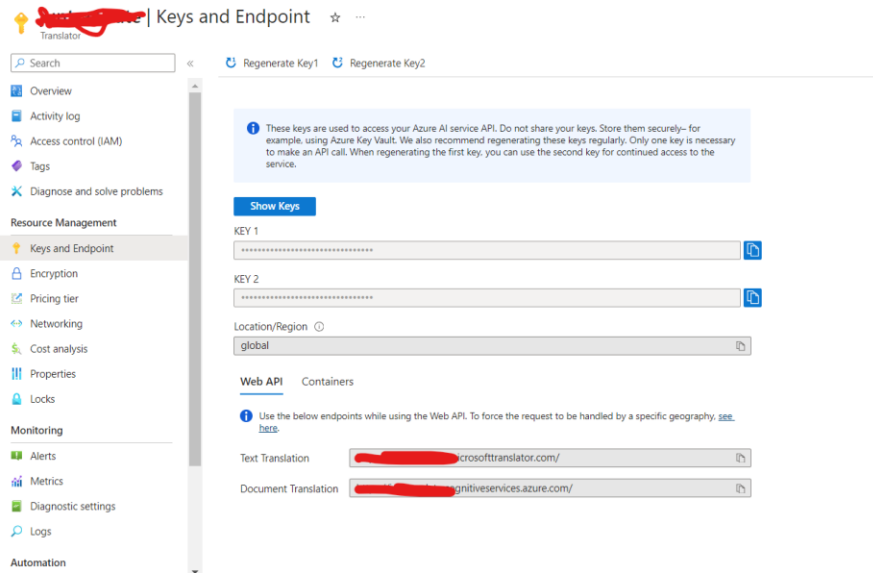
[View full pricing details](#)

[Previous](#) [Next](#) [Review + create](#)

Untuk detail cara membuat Translator resource pada Azure portal dapat dilihat pada url berikut : <https://learn.microsoft.com/en-us/azure/ai-services/translator/create-translator-resource> .

Semua layanan Azure AI API memerlukan URL endpoint dan read only key untuk autentikasi. Authentication key adalah string unik yang diteruskan pada setiap permintaan ke Translation service. Untuk passing key dapat melalui parameter query string atau dengan menentukannya di header permintaan HTTP.

Buka halaman Translator resource yang baru kita buat diatas, lalu pada panel navigasi di sebelah kiri, di bawah **Resource Management**, pilih **Key and Endpoint**. Lalu copy dan paste **Key dan Endpoint** nya pada notepad atau document lainnya.



Sebelum kita memulai dengan coding pada C#, terlebih dahulu kita gunakan tools (postman atau yang lainnya) untuk mencoba menggunakan REST API. Pertama-tama kita akan mencoba untuk mendapatkan bahasa apa saja yang didukung oleh service ini. Dengan menggunakan Url seperti dibawah.

<https://api.cognitive.microsofttranslator.com/languages?api-version=3.0>

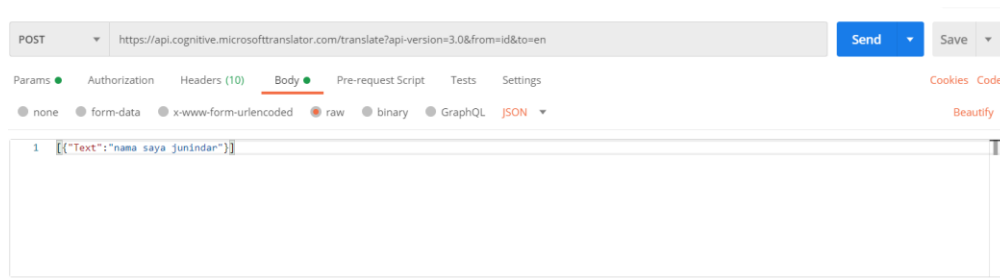
Maka kita akan mendapatkan hasil seperti pada gambar dibawah.

```
1 [{"translation": {
2   "af": {
3     "name": "Afrikaans",
4     "nativeName": "Afrikaans",
5     "dir": "ltr"
6   },
7   "am": {
8     "name": "Amharic",
9     "nativeName": "አማርኛ",
10    "dir": "ltr"
11  },
12  "ar": {
13    "name": "Arabic",
14    "nativeName": "العربية",
15    "dir": "rtl"
16  },
17  "as": {
18    "name": "Assamese",
19    "nativeName": "অসমীয়া",
20    "dir": "ltr"
21  },
22  }
}]
```

Selanjutnya kita akan mencoba menerjemahkan sebuah kalimat dari Bahasa Indonesia ke English. Dengan menggunakan Url seperti dibawah.

<https://api.cognitive.microsofttranslator.com/translate?api-version=3.0&from=id&to=en>

Dari url diatas dapat kita lihat, untuk menerjemahkan text kita akan menggunakan method “translate“, dengan api-version 3.0. Lalu terdapat paramater **from** dan **to**. Dimana **from** adalah kode negara dari bahasa yang akan diterjemahkan. Sedangkan **to**, adalah kode negara dari bahasa yang menjadi target untuk diterjemahkan.

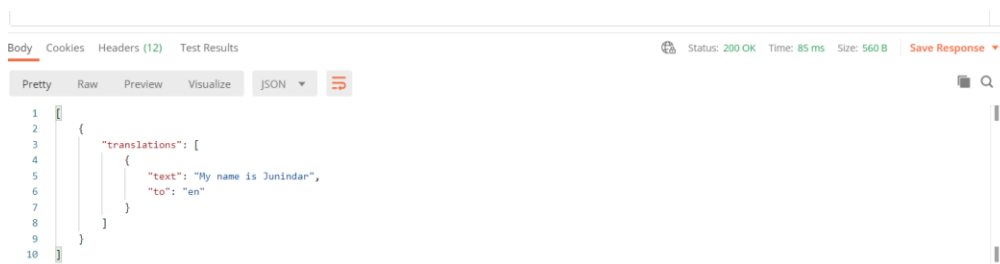


Dan pada Header kita masukkan Key (Ocp-Apim-Subscription-Key) dan Region (Ocp-Apim-Subscription-Region).

Headers 8 hidden

KEY	VALUE
<input checked="" type="checkbox"/> Ocp-Apim-Subscription-Key	[REDACTED]
<input checked="" type="checkbox"/> Ocp-Apim-Subscription-Region	global
Key	Value

Dan hasil yang didapat adalah sebagai berikut.



Setelah selesai dengan latihan diatas, selanjutnya kita akan menggunakan Translator Service ini pada Telegram Bot.. Mari ikuti langkah-langkah dibawah ini.

Note : Pada artikel ini, penulis menggunakan Visual Studio 2022 dan .Net 8.0. Dan pastikan telah mengikuti latihan pada artikel sebelumnya.

Pada Visual Studio buat sebuah “Console App” project dengan menggunakan .Net 8.0. Setelah project berhasil dibuat, tambahkan nuget “TelegranBotBase” dan “Newtonsoft.Json” pada project.

Lalu buat sebuah class dengan nama “ StartForm.cs” seperti dibawah. Class ini akan digunakan sebagai StartForm (sintaks detail pada Program.cs) yang mana pada class ini akan kita buat sebuah function untuk menggunakan Translator Service.

```
public class StartForm : FormBase
{
}

async Task<string> translate(string text)
{
    string key = "apiKey";
    string endpoint = "https://api.cognitive.microsofttranslator.com";
    string location = "global";

    string route = "/translate?api-version=3.0&from=id&to=en";
    string textToTranslate = text;
    object[] body = new object[] { new { Text = textToTranslate } };
    var requestBody = JsonConvert.SerializeObject(body);

    using (var client = new HttpClient())
    using (var request = new HttpRequestMessage())
    {
        request.Method = HttpMethod.Post;
        request.RequestUri = new Uri(endpoint + route);
        request.Content = new StringContent(requestBody, Encoding.UTF8,
            "application/json");
        request.Headers.Add("Ocp-Apim-Subscription-Key", key);
        request.Headers.Add("Ocp-Apim-Subscription-Region", location);
        HttpResponseMessage response = await
            client.SendAsync(request).ConfigureAwait(false);

        var responseBody = await response.Content.ReadAsStringAsync();
        var result = JsonConvert.DeserializeObject<List<Dictionary<string,
            List<Dictionary<string, string>>>>>(responseBody);
        return result[0]["translations"][0]["text"];
    }
}
```

Pada sintaks diatas dapat dilihat kita menggunakan “HttpClient” untuk melakukan request terhadap TranslatorService. HttpClient adalah kelas dalam bahasa pemrograman C# yang terdapat dalam namespace System.Net.Http. Kelas ini menyediakan fungsionalitas untuk membuat dan mengelola permintaan HTTP, serta menerima respons dari server HTTP. HttpClient memungkinkan Anda untuk berinteraksi dengan API web, mengambil data dari server, atau mengirim data ke server dengan menggunakan protokol HTTP. Dan dari response yang didapat, selanjutnya kita gunakan method “DeserializeObject” dari class “Newtonsoft.Json. JsonConvert” untuk mendapat hasil dari terjemahan. Lalu hasilnya akan digunakan sebagai *return* dari function ini.

Selanjutnya ketikkan dua method dibawah.

```
public override async Task Load(MessageResult message)
{
    if (message.MessageText.Trim() == "")
        return;
    if (message.MessageText.ToLower() != "/start")
    {
        return;
    }
    await Device.HideReplyKeyboard(".....");
}
```

```
public override async Task Render(MessageResult message)
{
    try
    {
        if (message.MessageText.ToLower() != "/start")
        {
            var result = await translate(message.MessageText);
            await this.Device.Send(result);
            await this.Device.Send("Ketikkan kalimat atau kata yang akan di terjemahkan");
            return;
        }
        await this.Device.Send("Selamat datang di Ilmu Komputer Bot.  
Ketikkan kalimat atau kata yang akan di terjemahkan");
    }
    catch (Exception ex)
    {
        await this.Device.Send("Terjadi kesalahan sistem, silahkan hubungi admin", null);
    }
}
```

Pada method Render terdapat kondisi, dimana jika MessageText tidak sama dengan “/start“, maka function “translate“ akan dipanggil dan hasilnya akan dikirim melalui Bot. Untuk hasilnya dapat dilihat pada gambar dibawah ini.





Untuk sintaks pada Program.cs dapat dilihat pada project lampiran.

Selain dengan menggunakan cara diatas, kita dapat menggunakan Azure Text Translation SDK. Azure Text Translation SDK adalah sekumpulan pustaka dan alat yang memungkinkan kita dengan mudah mengintegrasikan kemampuan REST API. Dengan Text Translation ke dalam aplikasi. Text Translation SDK tersedia di seluruh platform pemrograman di C#/.NET, Java, JavaScript, dan Python.

Sebelum menggunakan SDK ini, terlebih dahulu kita perlu meng-install SDK ini pada project. Buka Package Manager Console dan copy command dibawah.

```
NuGet\Install-Package Azure.AI.Translation.Text -Version 1.0.0-beta.1
```

Untuk penggunaan SDK pada C# dapat dilihat pada function dibawah ini. Kedua cara ini akan mendapatkan hasil yang sama.

```
async Task<string?> translateWithSdk(string text)
{
    Uri endpoint = new("https://api.cognitive.microsofttranslator.com");
    string apiKey = "";
    string result = "";

    TextTranslationClient client = new TextTranslationClient(new AzureKeyCredential(apiKey),
        endpoint);
    Response<IReadOnlyList<TranslatedTextItem>> response = await client.TranslateAsync("en",
        text, sourceLanguage: "id");
    IReadOnlyList<TranslatedTextItem> translations = response.Value;
    if (translations.Any())
    {
        TranslatedTextItem translation = translations.First();
        result= translation.Translations.FirstOrDefault()?.Text;
    }

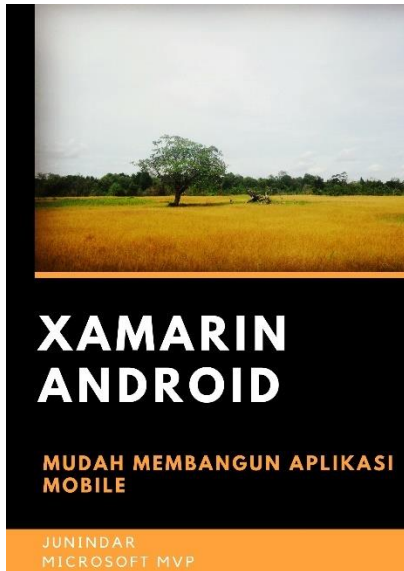
    return result;
}
```

## Penutup

Sedangkan untuk memudahkan dalam memahami isi artikel, maka penulis juga menyertakan dengan full source code project latihan ini, dan dapat di download disini

<https://junindar.blogspot.com/2024/01/telegram-bot-framework-dan-azure-ai.html>

## Referensi



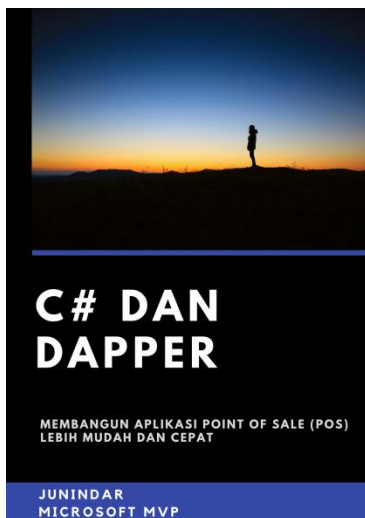
<https://play.google.com/store/books/details?id=G4tFDgAAQBAJ>



<https://play.google.com/store/books/details?id=VSLiDQAAQBAJ>



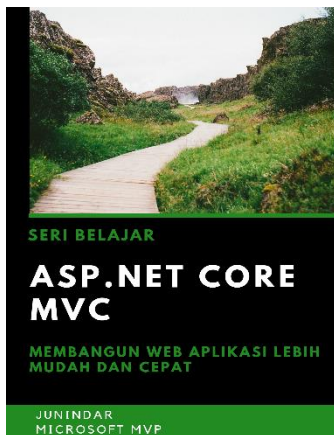
[https://play.google.com/store/books/details/Junindar\\_Xamarin\\_Forms?id=6Wg-DwAAQBAJ](https://play.google.com/store/books/details/Junindar_Xamarin_Forms?id=6Wg-DwAAQBAJ)



[https://play.google.com/store/books/details/Junindar\\_C\\_dan\\_Dapper\\_Membangun\\_Aplikasi\\_POS\\_Point?id=6TErDwAAQBAJ](https://play.google.com/store/books/details/Junindar_C_dan_Dapper_Membangun_Aplikasi_POS_Point?id=6TErDwAAQBAJ)



[https://play.google.com/store/books/details/Junindar\\_ASP\\_NET\\_MVC\\_Membangun\\_Aplikasi\\_Web\\_Lebih?id=XLlyDwAAQBAJ](https://play.google.com/store/books/details/Junindar_ASP_NET_MVC_Membangun_Aplikasi_Web_Lebih?id=XLlyDwAAQBAJ)



[https://play.google.com/store/books/details/Junindar\\_ASP\\_NET\\_CORE\\_MVC?id=xEe5DwAAQBAJ](https://play.google.com/store/books/details/Junindar_ASP_NET_CORE_MVC?id=xEe5DwAAQBAJ)

## Biografi Penulis.



Junindar Lahir di Tanjung Pinang, 21 Juni 1982. Menyelesaikan Program S1 pada jurusan Teknik Inscreenatika di Sekolah Tinggi Sains dan Teknologi Indonesia (ST-INTEN-Bandung). Junindar mendapatkan Award Microsoft MVP VB pertanggal 1 oktober 2009 hingga saat ini. Senang mengutak-atik computer yang berkaitan dengan bahasa pemrograman. Keahlian, sedikit mengerti beberapa bahasa pemrograman seperti : VB.Net, C#, SharePoint, ASP.NET, VBA. Reporting: Crystal Report dan Report Builder. Database: MS Access, MY SQL dan SQL Server. Simulation / Modeling Packages: Visio Enterprise, Rational Rose dan Power Designer. Dan senang bermain gitar, karena untuk bisa menjadi pemain gitar dan seorang programmer sama-sama membutuhkan seni. Pada saat ini bekerja di salah satu Perusahaan Consulting dan Project Management di Malaysia sebagai Senior Consultant. Memiliki beberapa sertifikasi dari Microsoft yaitu Microsoft Certified Professional Developer (MCPD – SharePoint 2010), MOS (Microsoft Office Specialist) dan MCT (Microsoft Certified Trainer) Mempunyai moto hidup: **“Jauh lebih baik menjadi Orang Bodoh yang giat belajar, dari pada orang Pintar yang tidak pernah mengimplementasikan ilmunya”**.