

# Pengenalan Telegram Bot Framework Pada C#

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

***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.*

*junindar@gmail.com*

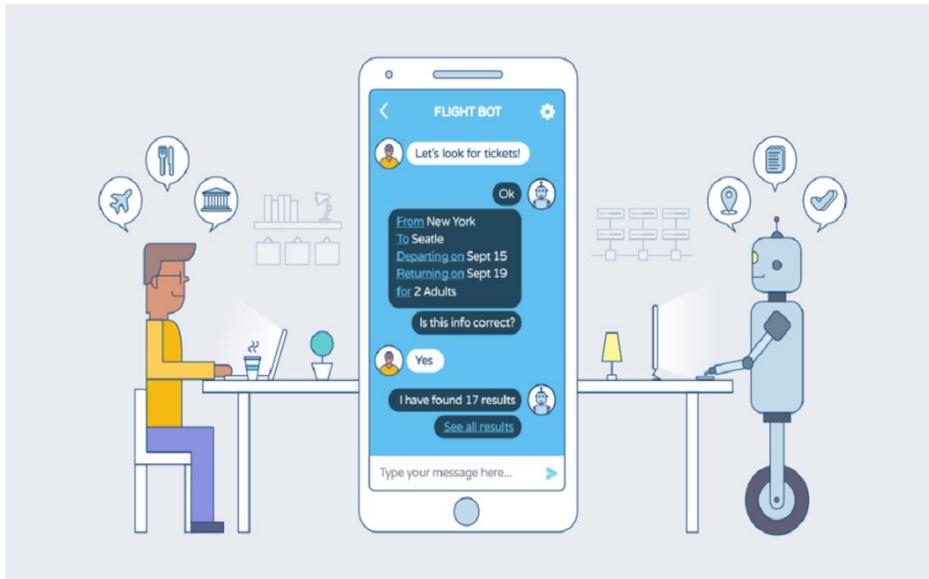
<http://junindar.blogspot.com>

## 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.

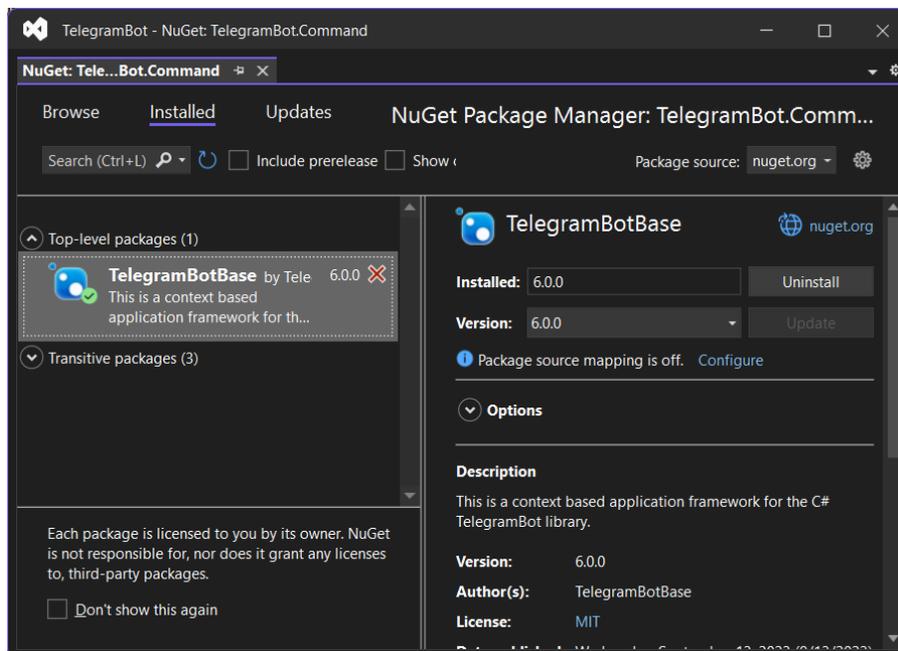
Terdapat banyak fasilitas pada framework ini, sehingga kita dapat mengembangkan bot yang powerfull dan fleksibel dengan menyediakan berbagai fungsi yang dapat digunakan untuk membuat bot yang sesuai dengan kebutuhan penggunaannya.

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.

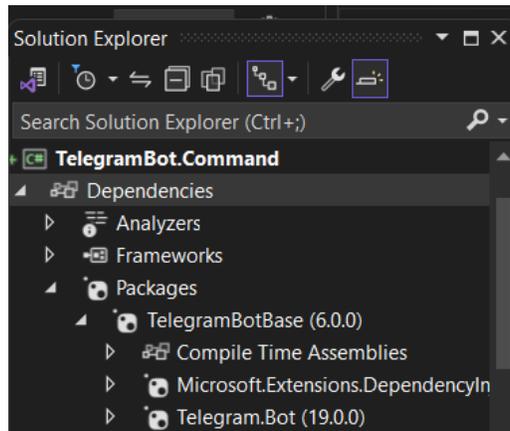
Selanjutnya untuk membuat chatbot dengan menggunakan Telegram.Bot Library dapat mengikuti langkah-langkah dibawah ini.

Note : Pada artikel ini, penulis menggunakan Visual Studio 2022 dan .Net 7.0.

Pada Visual Studio buat sebuah “Console App“ project dengan menggunakan .Net 7.0. Setelah project berhasil dibuat, tambahkan “TelegramBotBase“ nuget pada project.



Pada gambar dibawah, dapat kita lihat TelegramBotBase menggunakan library Telegram.Bot.



Setelah selesai dengan langkah-langkah diatas, maka kita lanjutkan dengan memulai coding.

Buat sebuah class dengan nama "StartForm", seperti pada contoh dibawah ini. Terdapat beberapa tipe form pada TelegramBotFramework.

```
public class StartForm : FormBase
{
    public override async Task Load(MessageResult message)
    {
        await this.Device.Send("Hi Selamat Datang.");
    }
}
```

Disini kita akan mengirimkan pesan setelah loading pada form.

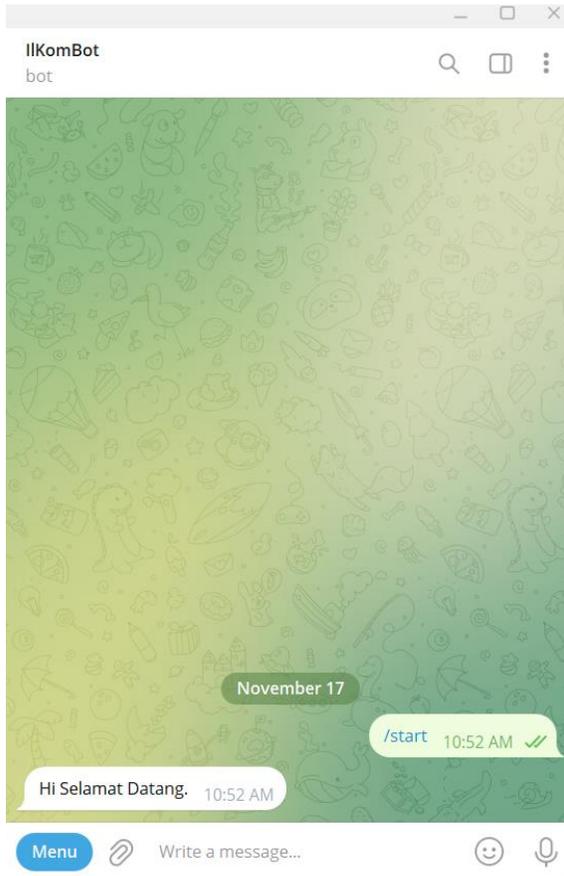
Lalu pada Program.cs ketikkan sintaks dibawah ini.

```
var bb = BotBaseBuilder
    .Create()
    .QuickStart<StartForm>("[ApiKey]")
    .Build();

bb.Start();
Console.WriteLine("Telegram Bot started...");
Console.WriteLine("Press q to quit application.");
Console.ReadLine();
bb.Stop();
```

Pada class BotBase mengelola banyak fungsi yang dapat kita gunakan. Seperti bot command, action event dan lainnya. StartForm adalah form pertama yang mana setiap pengguna akan diarahkan ke form tersebut, seperti halaman awal pada sebuah aplikasi. Form tersebut harus berupa subkelas FormBase yang dapat kita temukan di namespace TelegramBotBase.Base

Jalankan program dan pastikan mendapatkan hasil seperti dibawah ini.



Latihan selanjutnya kita akan membuat, Bot Command pada TelegramBot. Bot Command adalah serangkaian perintah yang dapat digunakan oleh pengguna untuk berinteraksi dengan bot. Dalam konteks pengembangan bot Telegram, Bot Command digunakan untuk memberi tahu Telegram tentang perintah yang dapat digunakan pengguna untuk berkomunikasi dengan bot tersebut.

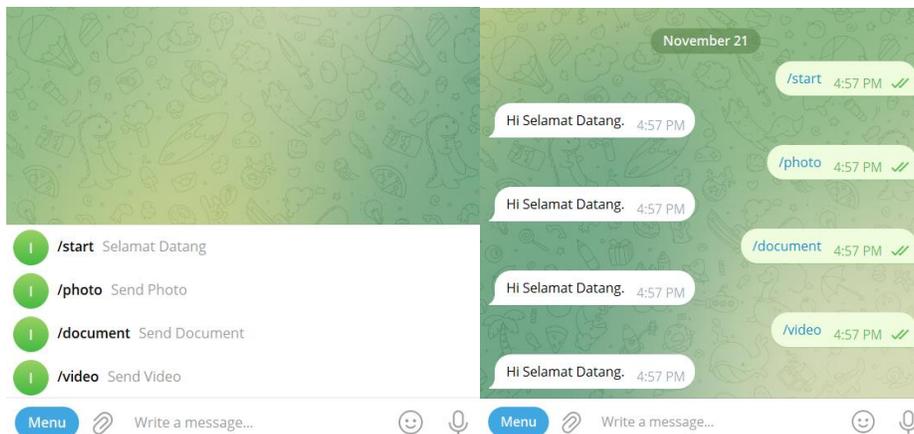
Misalnya, jika kita membuat bot untuk layanan cuaca, Anda dapat menetapkan perintah seperti /cuaca atau /suhu yang akan memicu bot untuk memberikan informasi cuaca terkini. Dengan menetapkan perintah-perintah ini, pengguna dapat dengan mudah berinteraksi dengan bot dengan cara yang terstruktur dan terorganisir.

Bot Command memungkinkan pengguna untuk lebih memahami cara menggunakan bot dengan menyediakan daftar perintah yang valid. Ini membuat pengguna lebih mudah untuk berkomunikasi dengan bot dan mengakses fungsionalitas yang disediakan oleh bot tersebut.

```
var bb = BotBaseBuilder
    .Create()
    .WithAPIKey("[ApiKey]")
    .DefaultMessageLoop()
    .WithStartForm<StartForm>()
    .NoProxy()
    .CustomCommands(a =>
    {
        a.Start("Selamat Datang");
        a.Add("photo", "Send Photo");
        a.Add("document", "Send Document");
        a.Add("video", "Send Video");
    })
    .NoSerialization()
    .UseEnglish()
    .Build();

bb.UploadBotCommands().Wait();
bb.Start();
Console.WriteLine("Telegram Bot started...");
Console.WriteLine("Press q to quit application.");
Console.ReadLine();
bb.Stop();
```

Pada sintaks diatas, dapat dilihat terdapat 4 buah command dan masih menggunakan StartForm menjadi form pertamanya. Jika project ini kita jalankan maka pada Bot kita akan terdapat “Menu“ yang memiliki 4 buah command dan jika kita panggil command-command tersebut, maka balasan dari setiap command itu akan sama. Seperti pada gambar dibawah ini.



Selanjutnya kita modifikasi sintaks yang ada pada StartForm, ikuti langkah-langkah dibawah ini. Pada “Load“ ganti sintaks menjadi seperti dibawah.

```
if (message.MessageText.Trim() == "")
    return;
if (message.MessageText.ToLower() != "/start")
{
    return;
}
await Device.HideReplyKeyboard(".....");
```

Pada method ini akan dicek nilai dari MessageText. Jika nilainya sama dengan “/start“, maka bot akan membalas dengan menggunakan method “HideReplyKeyboard“, dimana method ini digunakan untuk membalas pesan dan nantinya akan dihapus kembali.

Selanjutnya kita tambahkan sintaks dibawah ini (masih pada class StartForm).

```
public override async Task Render(MessageResult message)
{
    try
    {
        if (message.MessageText.ToLower() != "/start")
        {
            return;
        }
        await this.Device.Send("Selamat datang di Ilmu Komputer Bot.
        Silahkan pilih command/menu untuk mendapatkan informasi");
    }
    catch (Exception ex)
    {
        await this.Device.Send("Terjadi kesalahan sistem, silahkan hubungi admin", null);
    }
}
```

Sama seperti “Load“, pada method ini juga akan mengecek nilai dari MessageText, Jika nilainya sama dengan “/start“, maka bot akan mengirimkan pesan selamat datang.

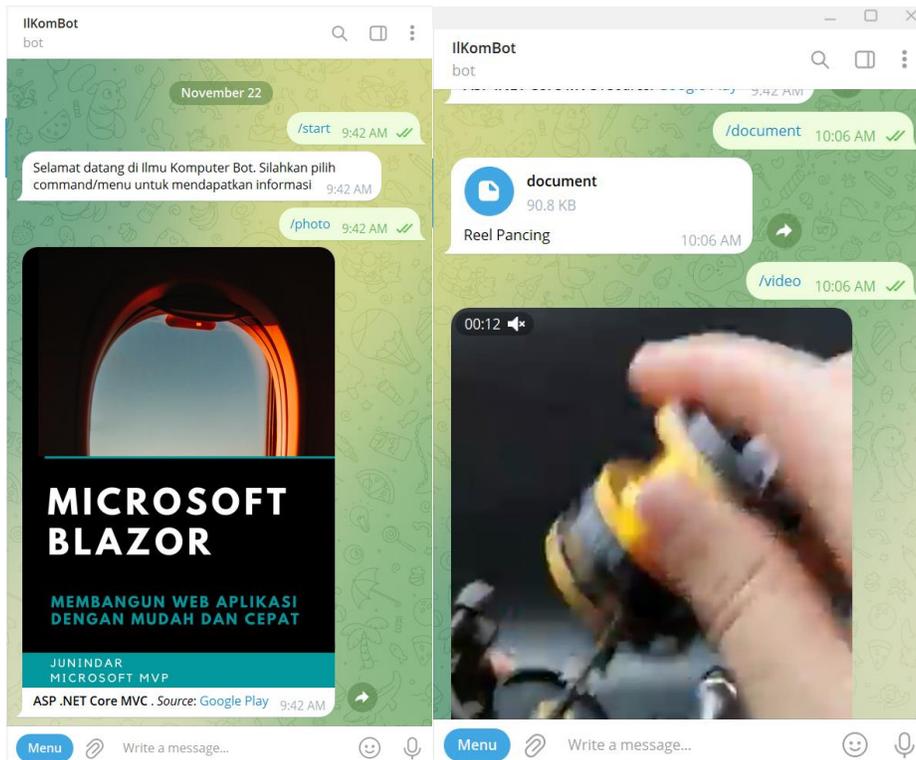
Lalu buka Program.cs, disini kita akan membuat perintah pada setiap command seperti pada sintaks dibawah. Dengan menggunakan BotCommandEvent, kita akan mendapatkan nilai dari command yang dipilih oleh pengguna untuk kemudian kita gunakan pada switch case.

```
bb.BotCommand += async (s, en) =>
{
    FileStream fStream = null;
    switch (en.Command)
    {
        case "/start":
            var start = new StartForm();
            await en.Device.ActiveForm.NavigateTo(start);
            break;
        case "/photo":
            // Detail pada lampiran project
            break;
        case "/document":
            // Detail pada lampiran project
            break;
        case "/video":
            // Detail pada lampiran project
            break;
    }
};
```

Jika command yang dipilih adalah start, maka StartForm akan aktif. Lalu akan mendampirkan pesan selamat datang.

Untuk detail-detail perintah pada command lainnya dapat dilihat pada project lampiran.

Jalankan program dan pastikan mendapatkan hasil seperti dibawah.

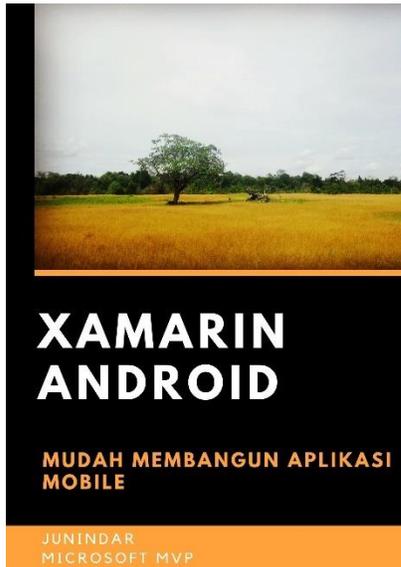


## **Penutup**

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/2023/12/pengenalan-telegram-bot-framework-pada-c.html>

## Referensi



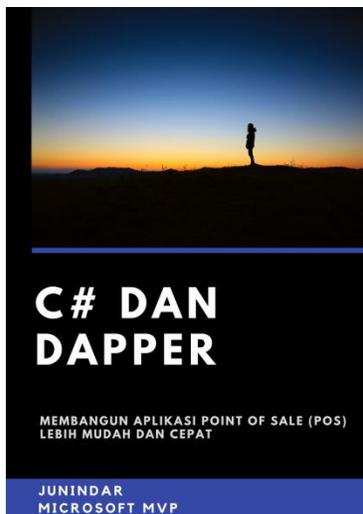
<https://play.google.com/store/books/details?id=G4tFDgAAQBAJ>



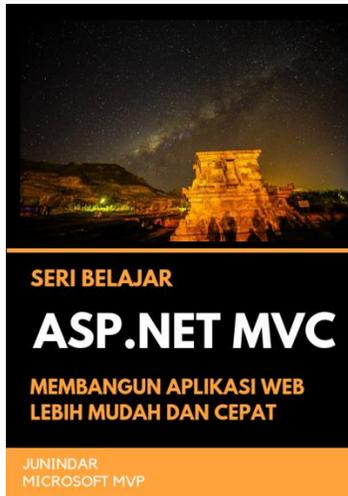
<https://play.google.com/store/books/details?id=VSLiDQAAQBAJ>



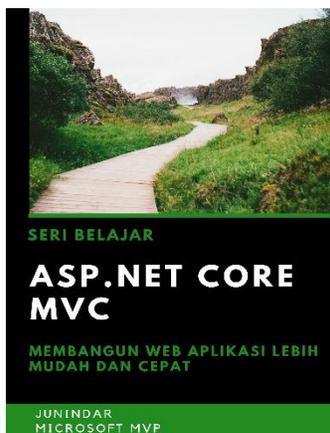
[https://play.google.com/store/books/details/Junindar\\_Xamarin\\_Forms?id=6Wg-DwAAQBAJ](https://play.google.com/store/books/details/Junindar_Xamarin_Forms?id=6Wg-DwAAQBAJ)



[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”**.