

Pengenalan Telegram.Bot Pada C#

Junindar, ST, MCPD, MOS, MCT, MVP

junindar@gmail.com

<http://junindar.blogspot.com>

Lisensi Dokumen:

Copyright © 2003 IlmuKomputer.Com

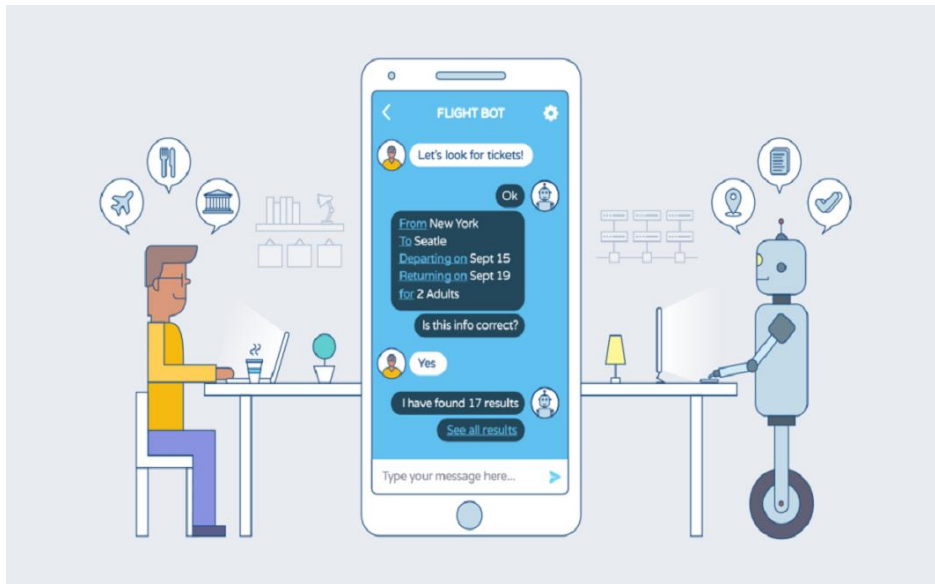
Seluruh dokumen di IlmuKomputer.Com dapat digunakan, dimodifikasi dan disebarkan secara bebas untuk tujuan bukan komersial (nonprofit), dengan syarat tidak menghapus atau merubah atribut penulis dan pernyataan copyright yang disertakan dalam setiap dokumen. Tidak diperbolehkan melakukan penulisan ulang,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

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 Lenguage Processing (NLP). Machine Learning dapat di aplikasikan sebagai mesin yang mempelajari, menganalisa dan mengenal bermacam-macam Bahasa pada chatbot, sementara Natural Lenguage 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



Dilansir dari BT, Telegram adalah aplikasi pesan instan yang dapat digunakan secara gratis dengan menggunakan Wi-Fi atau kuota internet.

Pada dasarnya aplikasi ini memang tidak berbeda jauh dengan WhatsApp atau Facebook Messenger. Meski dua aplikasi itu populer, saat ini banyak sekali yang menggunakan Telegram.

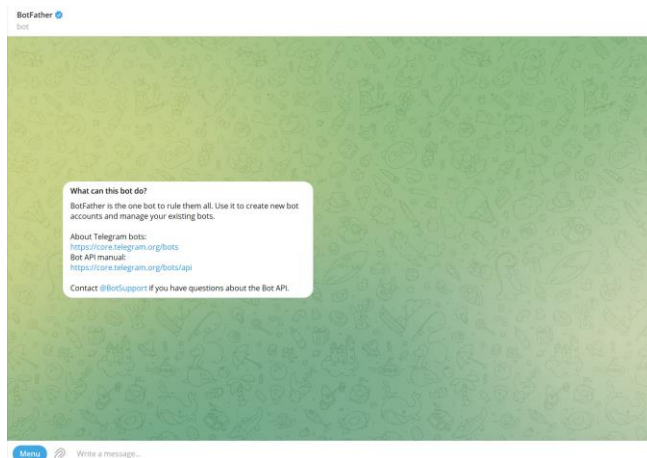
Dikutip dari Telegram, aplikasi ini telah mencapai 400 juta pengguna bulanan aktif. Setidaknya, setiap hari ada 1,5 juta pengguna baru yang mendaftarkan diri ke dalam aplikasi Telegram.

Orang-orang banyak yang langsung beralih menggunakan Telegram karena bisa dibilang aplikasi ini menyajikan berbagai fitur seperti penyimpanan file, cloud, folder, dan lain-lain.

Telegram sendiri didirikan oleh orang Rusia yang bernama Pavel Valerievich Durov pada tanggal 23 Oktober 2013. Ia juga mendirikan salah satu jejaring sosial terbesar di Rusia yang bernama Vkontakte.

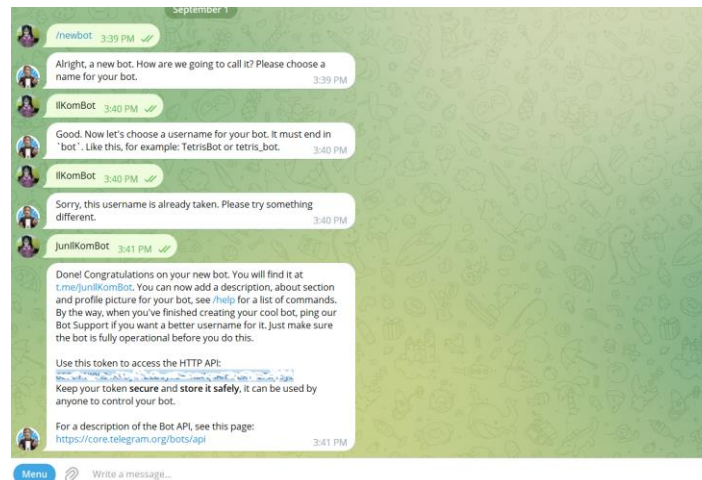
Sebelum kita memulai coding, terlebih dahulu kita buat Bot pada telegram. Untuk membuat Bot baru pada Telegram kita gunakan BotFather. Untuk membuat Bot BotFather cukup ikuti langkah-langkah berikut ini:

1. Buka aplikasi Telegram yang ada di HP ataupun di laptop atau komputer.
2. Pada bagian kolom pencarian masukkan kata BotFather.
3. Klik akun BotFather yang terverifikasi atau ber icon centang biru di samping namanya.



4. Ketika room chat bot terbuka, klik menu Start untuk mengetahui menu bot yang ada.

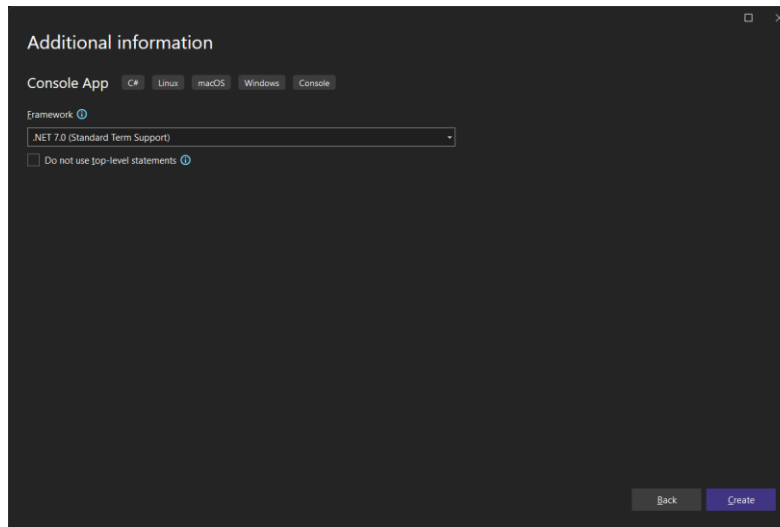
5. Klik perintah atau newbot untuk membuat bot baru.
6. Masukkan nama bot yang akan dibuat. Minimal 5 karakter dan maksimal 32 karakter.
7. Masukkan username bot yang harus diakhiri dengan –bot. Misalnya ingin menggunakan username music maka penulisannya menjadi musicbot.
8. Setelah itu akan menerima token API HTTP berwarna biru dan diketik dengan font berbeda.
9. Selesai, pastikan simpan token tersebut untuk mengendalikan bot yang telah dibuat.



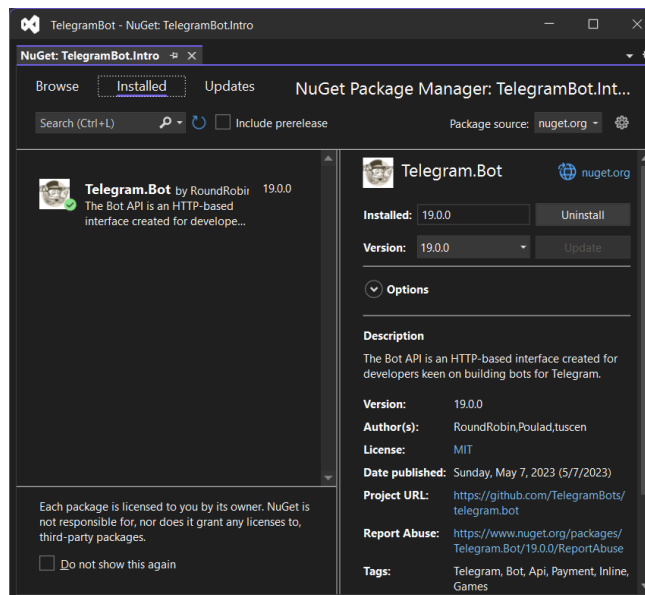
Selanjutnya untuk membuat chatbot dengan menggunakan Telegram.Bot Library dapat mengikuti langkah-langkah dibawah ini.

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 seperti dibawah.



Setelah project berhasil dibuat, tambahkan “Telegran.Bot” nuget pada project.



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

```
var botClient = new TelegramBotClient("xxxxxxxxxx");  
var me = await botClient.GetMeAsync();  
Console.WriteLine($"Hi {me.Id} - BotName: {me.FirstName}.");
```

Ketikkan sintak diatas pada program.cs, pastikan ganti nilai xxxxxxxx dengan token yang didapat dari pembuatan Bot dengan BotFather diatas. Sintaks ini digunakan untuk mengambil informasi Bot berdasarkan token aksesnya dengan memanggil metode GetMe dari Bot API. Lalu jalankan program untuk melihat hasilnya.



Dari gambar diatas kita akan mendapatkan Id dan Nama dari Bot yang telah kita buat sebelumnya.

Lalu kita lanjutkan dengan membuat sintaks untuk berinteraksi dengan Bot melalui Telegram. Tambahkan dua buah method seperti dibawah.

```
async Task HandleUpdateAsync(ITelegramBotClient botClient, Update update, CancellationTokentoken)  
{  
    if (update.Message is not { } message)  
        return;  
    if (message.Text is not { } messageText)  
        return;  
    var chatId = message.Chat.Id;  
    Console.WriteLine($"Received a '{messageText}' message in chat {chatId}.");  
    Message sentMessage = await botClient.SendTextMessageAsync(  
        chatId: chatId,  
        text: "You said:\n" + messageText,  
        cancellationTokentoken: cancellationTokentoken);  
}  
  
Task HandlePollingErrorAsync(ITelegramBotClient botClient, Exception exception, CancellationTokentoken cancellationTokentoken)  
{  
    var ErrorMessage = exception switch  
    {  
        ApiRequestException apiRequestException  
            => $"Telegram API  
            Error:\n[{apiRequestException.ErrorCode}]\n{apiRequestException.Message}",  
        _ => exception.ToString()  
    };  
    Console.WriteLine(ErrorMessage);  
    return Task.CompletedTask;  
}
```

“HandleUpdateAsync“ digunakan untuk mendapatkan pesan yang dikirim dari telegram sekaligus digunakan untuk mengirim atau membalas pesan berdasarkan

“chatId“. Sedangkan “HandlePollingErrorAsync“ dipanggil jika terjadi kesalahan saat melakukan update atau pembaruan. Dan terakhir tambahkan sintaks dibawah ini.

```
using CancellationTokensource cts = new();

ReceiverOptions receiverOptions = new()
{
    AllowedUpdates = Array.Empty<UpdateType>()
};

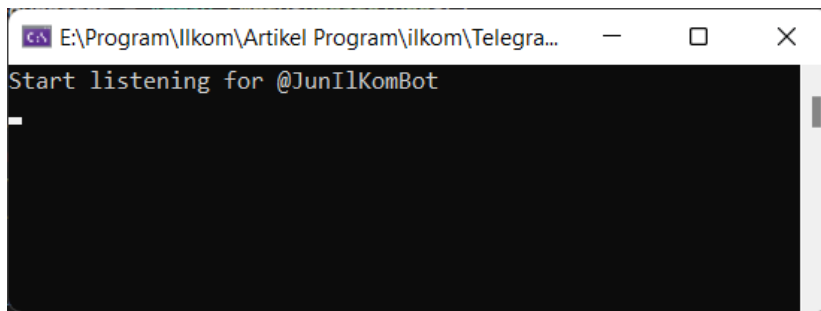
botClient.StartReceiving(
    updateHandler: HandleUpdateAsync,
    pollingErrorHandler: HandlePollingErrorAsync,
    receiverOptions: receiverOptions,
    cancellationToken: cts.Token
);

var me = await botClient.GetMeAsync();

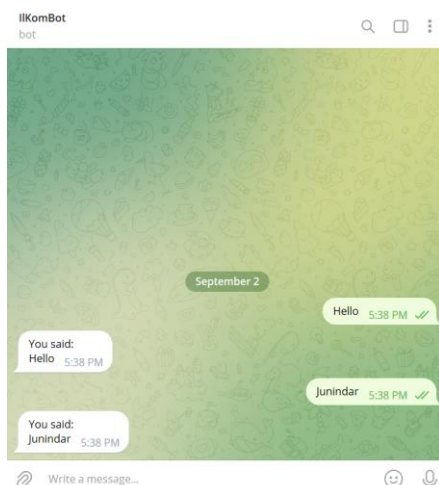
Console.WriteLine($"Start listening for @{me.Username}");
Console.ReadLine();

cts.Cancel();
```

Jalankan program untuk melihat hasilnya.



Lalu buka telegram dan cari Bot dengan nama “IlkomBot“. Setelah itu coba untuk mengirim pesan ke Bot tersebut.



Dengan memanggil klien bot, StartReceiving(...) akan melakukan update atau pembaruan menggunakan metode GetUpdates dari server Telegram. Agar aplikasi tetap berjalan kita gunakan sintaks Console.ReadLine().

Saat pengguna mengirim pesan, method HandleUpdateAsync(...) dipanggil dengan menggunakan objek Update yang diteruskan sebagai argumen. Selanjutnya akan diperiksa tipe dari pesan (Message.Type), yang akan kita gunakan hanya pesan teks. Dan terakhir, bot akan mengirim pesan teks kembali berdasarkan chatId. Jika berhasil seperti pada gambar diatas, maka kita telah sukses membuat simple Chat Bot pada Telegram.

Pada latihan selanjutnya, kita akan membuat percakapan dengan Bot. Dimana Bot akan menanyakan beberapa pertanyaan kepada kita. Ikuti langkah-langkah dibawah ini.

- Buat sebuah class dengan nama "UserProfile" seperti berikut.

```
public class UserProfile
{
    public long Id { get; set; }
    public string Nama { get; set; }
    public string Alamat { get; set; }
}
```

Dan buat variable List dari class UserProfile

```
List<UserProfile> userProfiles = new List<UserProfile>();
```

- Lalu pada method "HandleUpdateAsync" ganti sintaksnya seperti berikut.

```
async Task HandleUpdateAsync(ITelegramBotClient botClient, Update update,
CancellationTokens cancellationTokens)
{
    if (update.Message is not { } message)
        return;

    if (message.Text is not { } messageText)
        return;

    var chatId = message.Chat.Id;

    Console.WriteLine($"Received a '{messageText}' message in chat {chatId}.");

    var userProfile = userProfiles.FirstOrDefault(c => c.Id == chatId);
    string strText = "";
    if (userProfile == null)
    {
        strText = "Masukkan nama :";
        userProfile = new UserProfile();
        userProfile.Id = chatId;
        userProfiles.Add(userProfile);
    }
}
```

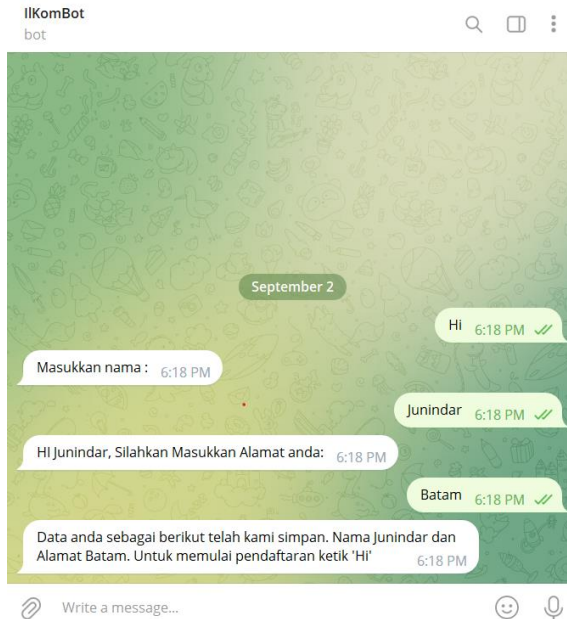


```
else
{
    if (string.IsNullOrEmpty(userProfile.Nama))
    {
        strText = $"HI {messageText}, Silahkan Masukkan Alamat anda:";
        userProfile.Nama = messageText;
    }
    else
    {
        userProfile.Alat = messageText;
        strText =
        $"Data anda sebagai berikut telah kami simpan. Nama
        {userProfile.Nama} dan Alamat {userProfile.Alat}.
        Untuk memulai pendaftaran ketik 'Hi'";
        userProfiles.Remove(userProfile);
    }
}

Message sentMessage = await botClient.SendTextMessageAsync(
chatId: chatId,
text: strText,
cancellationTokens: cancellationTokens);
}
```

Pada method ini akan dilakukan pengecekan pada variable “userProfiles” berdasarkan chatId. Jika tidak ditemukan data pada “userProfiles” maka bot akan mengirimkan pesan “Masukkan Nama :” dilanjutkan dengan membuat object baru dari class UserProfile dan ditambahkan kedalam “userProfiles”.

Lalu jika data ditemukan, maka akan dilakukan pengecekan apakah property Nama pada object tersebut berisi data atau tidak. Jika tidak berisi data maka bot akan mengirimkan pesan “Hi xxxx, Silahkan Masukan Alamat Anda”. Dan jika berisi data maka bot akan mengirimkan pesan Nama dan Alamat yang telah dimasukkan sebelumnya.



Jalankan program dan coba lakukan pengetesan sesuai dengan logic yang telah dibuat.

Pastikan mendapatkan hasil seperti pada gambar diatas.

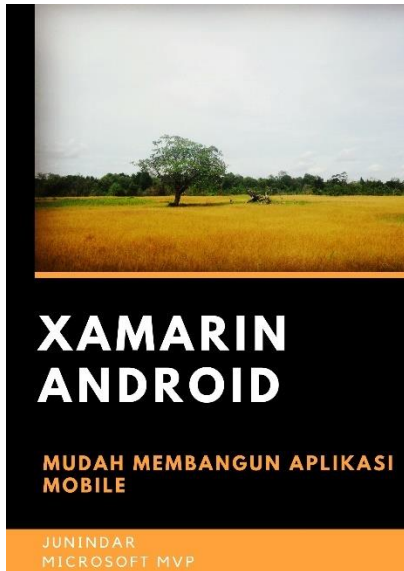
Setelah selesai dengan ketiga latihan diatas, maka kita telah berhasil membuat aplikasi chatbot sederhana dengan menggunakan TelegramBot Library.

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/2023/09/pengenalan-telegrambot-pada-c.html>

Referensi



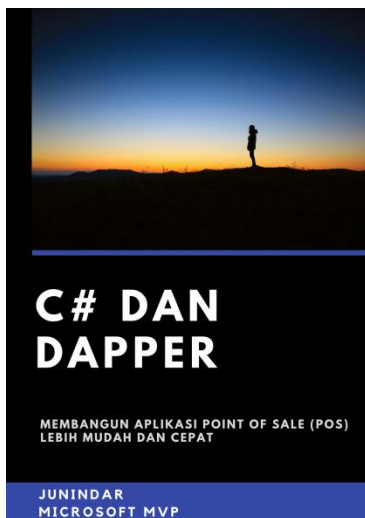
<https://play.google.com/store/books/details?id=G4tFDgAAQBAJ>



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



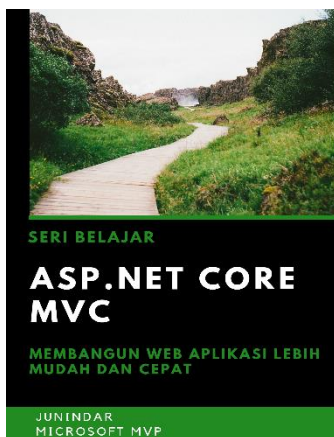
https://play.google.com/store/books/details/Junindar_Xamarin_Forms?id=6Wg-DwAAQBAJ



https://play.google.com/store/books/details/Junindar_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”**.