

PEMROGRAMAN LANJUT MVC Pattern dan REST API (Web Service)

Kamarudin, M.Kom

kamarudin@amikom.ac.id

<http://coding4ever.net/>

Persiapan Mengakses REST API Menggunakan C#

- ✓ Buka project *PerpustakaanAppMVC* (folder sample project)
- ✓ Jalankan project *PerpustakaanAppMVC* (shortcut F5)
- ✓ Menginstall library *RestSharp* menggunakan NuGet Package.

Apa itu Nuget? NuGet adalah paket manager untuk platform pengembangan menggunakan Microsoft .NET.

Dengan menggunakan NuGet akan mempermudah kita menginstall library siap pakai yang dikembangkan oleh Microsoft atau pihak ketiga.

Apa Itu RestSharp?

RestSharp adalah library pihak ketiga yang akan mempermudah kita untuk mengakses REST API (web service).

Dengan menggunakan RestSharp data/informasi yang dihasilkan oleh REST API yang secara default menggunakan format JSON bisa dengan mudah di konversi ke dalam sebuah objek.

Objek JSON

```
{  
    "npm": "15.02.9011",  
    "nama": "ABDUL HARI SUHUD",  
    "alamat": "Jl. Pengadilan Lama No. 253"  
}
```



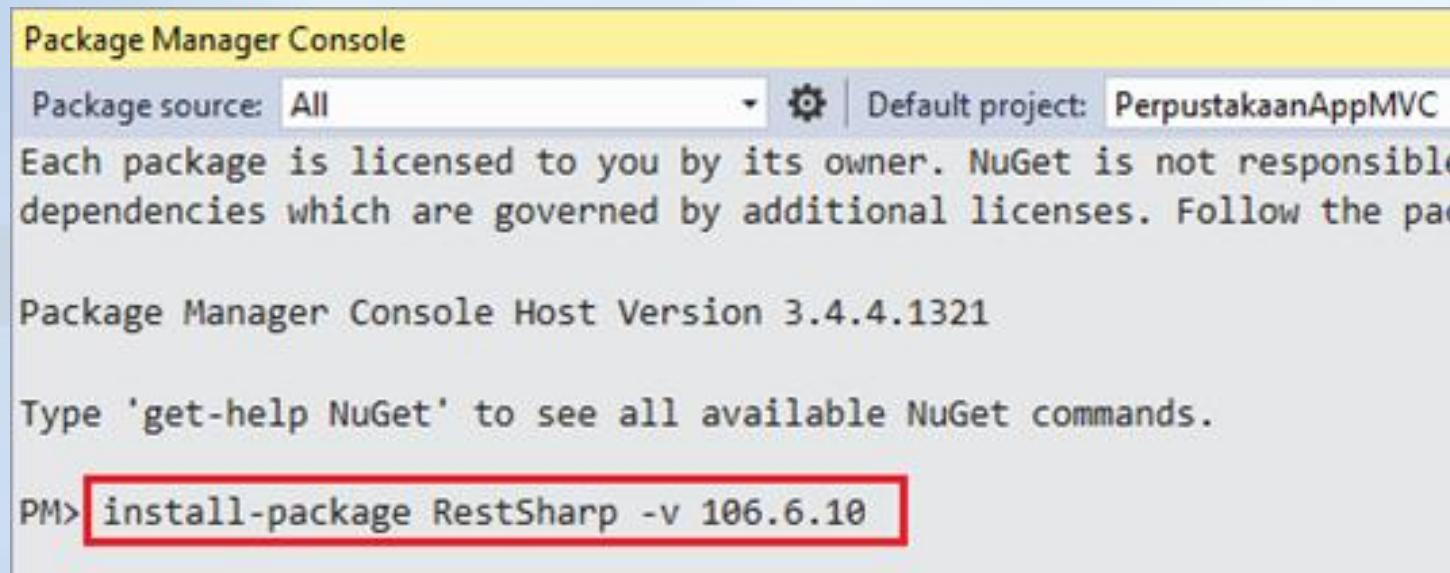
Objek C#

```
// membuat objek mahasiswa  
var mhs = new Mahasiswa();  
mhs.npm = "15.02.9011";  
mhs.nama = "ABDUL HARI SUHUD";  
mhs.alamat = "Jl. Pengadilan Lama No. 253";
```

Latihan 1: Menginstall RestSharp

Langkah-langkah:

1. Klik menu Tools -> NuGet Package Manager -> Package Manager Console.
2. Kemudian ketik perintah *install-package RestSharp -v 106.6.10* pada console NuGet Package Manager.



The screenshot shows the Windows Taskbar at the bottom with icons for Start, Task View, File Explorer, Edge, and others. Above the taskbar is the System tray showing battery level and volume. The main window is the Package Manager Console, which has a yellow header bar with the title 'Package Manager Console'. Below the header, there are settings for 'Package source: All' and 'Default project: PerpustakaanAppMVC'. A message about package licensing is displayed. The console window shows the command 'PM> install-package RestSharp -v 106.6.10' with a red rectangular box highlighting it. Below the command, the output 'Type 'get-help NuGet' to see all available NuGet commands.' is visible.

```
Package Manager Console
Package source: All | Default project: PerpustakaanAppMVC
Each package is licensed to you by its owner. NuGet is not responsible for dependencies which are governed by additional licenses. Follow the package author's terms of use.
PM> install-package RestSharp -v 106.6.10
Type 'get-help NuGet' to see all available NuGet commands.
```

Latihan 1 (Lanjutan): Menginstall RestSharp

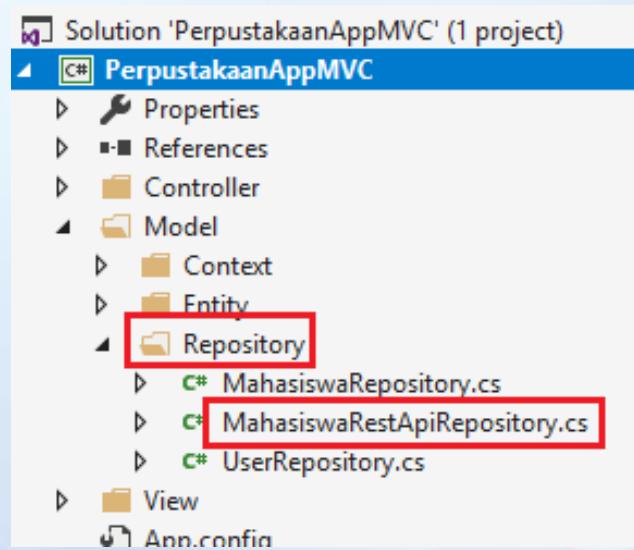
Package Manager Console

Package source: nuget.org Default project: PerpustakaanAppMVC

```
Attempting to resolve dependencies for package 'RestSharp.106.6.10' with
Resolving dependency information took 0 ms
Resolving actions to install package 'RestSharp.106.6.10'
Resolved actions to install package 'RestSharp.106.6.10'
Retrieving package 'RestSharp 106.6.10' from 'nuget.org'.
Adding package 'RestSharp.106.6.10' to folder 'E:\Praktikum\01 Pemrogramma
Added package 'RestSharp.106.6.10' to folder 'E:\Praktikum\01 Pemrogramman
Added package 'RestSharp 106.6.10' to 'packages.config'
Successfully installed 'RestSharp 106.6.10' to PerpustakaanAppMVC
Executing nuget actions took 1.97 sec
Time Elapsed: 00:00:02.2410146
PM> |
```

Latihan 2: Menambahkan Class Repository

Tambahkan class baru dengan nama
MahasiswaRestApiRepository di folder Repository.



Kemudian tambahkan beberapa method untuk operasi CRUDnya.

Latihan 2 (Lanjutan): Menambahkan Class Repository

```
public class MahasiswaRestApiRepository
{
    public int Create(Mahasiswa mhs)
    {
        throw new NotImplementedException();
    }

    public int Update(Mahasiswa mhs)
    {
        throw new NotImplementedException();
    }

    public int Delete(Mahasiswa mhs)
    {
        throw new NotImplementedException();
    }

    public List<Mahasiswa> ReadAll()
    {
        throw new NotImplementedException();
    }

    public List<Mahasiswa> ReadByNama(string nama)
    {
        throw new NotImplementedException();
    }

    public Mahasiswa ReadByNpm(string npm)
    {
        throw new NotImplementedException();
    }
}
```

Latihan 5

Latihan 3

Latihan 4 (tugas)

Latihan 3:

Melengkapi Kode untuk Method ReadAll

Method *ReadAll* digunakan untuk menampilkan semua data mahasiswa. Adapun informasi alamat REST API yang diakses sebagai berikut:

- ✓ Base Url
 - <http://latihan.coding4ever.net:5555/>
- ✓ Endpoint
 - api/mahasiswa
- ✓ HTTP Method
 - GET

Latihan 3 (Lanjutan): Melengkapi Kode untuk Method ReadAll

```
public List<Mahasiswa> ReadAll()
{
    string baseUrl = "http://latihan.coding4ever.net:5555/";
    string endpoint = "api/mahasiswa";

    // membuat objek rest client
    var client = new RestClient(baseUrl);

    // membuat objek request
    var request = new RestRequest(endpoint, Method.GET);

    // kirim request ke server
    var response = client.Execute<List<Mahasiswa>>(request);

    return response.Data;
}
```

Berisi collection dari objek mahasiswa

Latihan 3 (Lanjutan)

Revisi Method ReadAll di Class MahasiswaController

```
public List<Mahasiswa> ReadAll()
{
    // membuat objek collection
    List<Mahasiswa> list = new List<Mahasiswa>();

    // membuat objek context menggunakan blok using
    /*
    using (DbContext context = new DbContext())
    {
        // membuat objek dari class repository
        _repository = new MahasiswaRepository(context);

        // panggil method GetAll yang ada di dalam class repository
        list = _repository.ReadAll();
    }
    */

    // membuat objek dari class MahasiswaRestApiRepository
    var repo = new MahasiswaRestApiRepository();
    list = repo.ReadAll();

    return list;
}
```

Hasil Latihan 3

Setelah menyelesaikan latihan 3, jalankan aplikasi dan pastikan outputnya seperti berikut:

| No. | Npm | Nama | Angkatan |
|-----|------------|--------------------------|----------|
| 1 | 22.11.0013 | ACHMAD NURUL SIDIQ | 2022 |
| 2 | 22.11.0020 | ACHMADDITO | 2022 |
| 3 | 22.11.0032 | ADHITYA DWI NUGROHO | 2022 |
| 4 | 22.11.0047 | AGUS PARYONO | 2022 |
| 5 | 22.11.0021 | ANNAS MA'RUF | 2022 |
| 6 | 22.11.0022 | ARDIAN YUDHI PRIYATMOJO | 2022 |
| 7 | 22.11.0007 | ARIS PAMUNGKAS | 2022 |
| 8 | 22.11.0038 | BAGUS PRATOMO | 2022 |
| 9 | 22.11.0024 | DESI ANGGRAENI WULANDARI | 2022 |
| 10 | 22.11.0023 | DIAN DARU WIDIAHMOKO | 2022 |
| 11 | 22.11.0006 | DIMAS JANUAR AMALUDIN | 2022 |
| 12 | 22.11.0029 | DWI HARIYANTI | 2022 |
| 13 | 22.11.0043 | DYAN AKBAR SUPARDI | 2022 |
| 14 | 22.11.0042 | EKO CAHYADI | 2022 |
| 15 | 22.11.0050 | ERNADI | 2022 |
| 16 | 22.11.0041 | HAIDI HAJAR WIDAGDO | 2022 |
| 17 | 22.11.0016 | HARWIN CHANDRA PRASETYA | 2022 |
| 18 | 22.11.0004 | IMAM SANTOSO | 2022 |
| 19 | 22.11.0036 | INTAN KURNIATUN | 2022 |
| 20 | 22.11.0010 | IRAWAN ADITYA | 2022 |
| 21 | 22.11.0005 | KENDRIYANI SUHENDRIYANI | 2022 |

Latihan 4:

Melengkapi Kode untuk Method ReadByNama

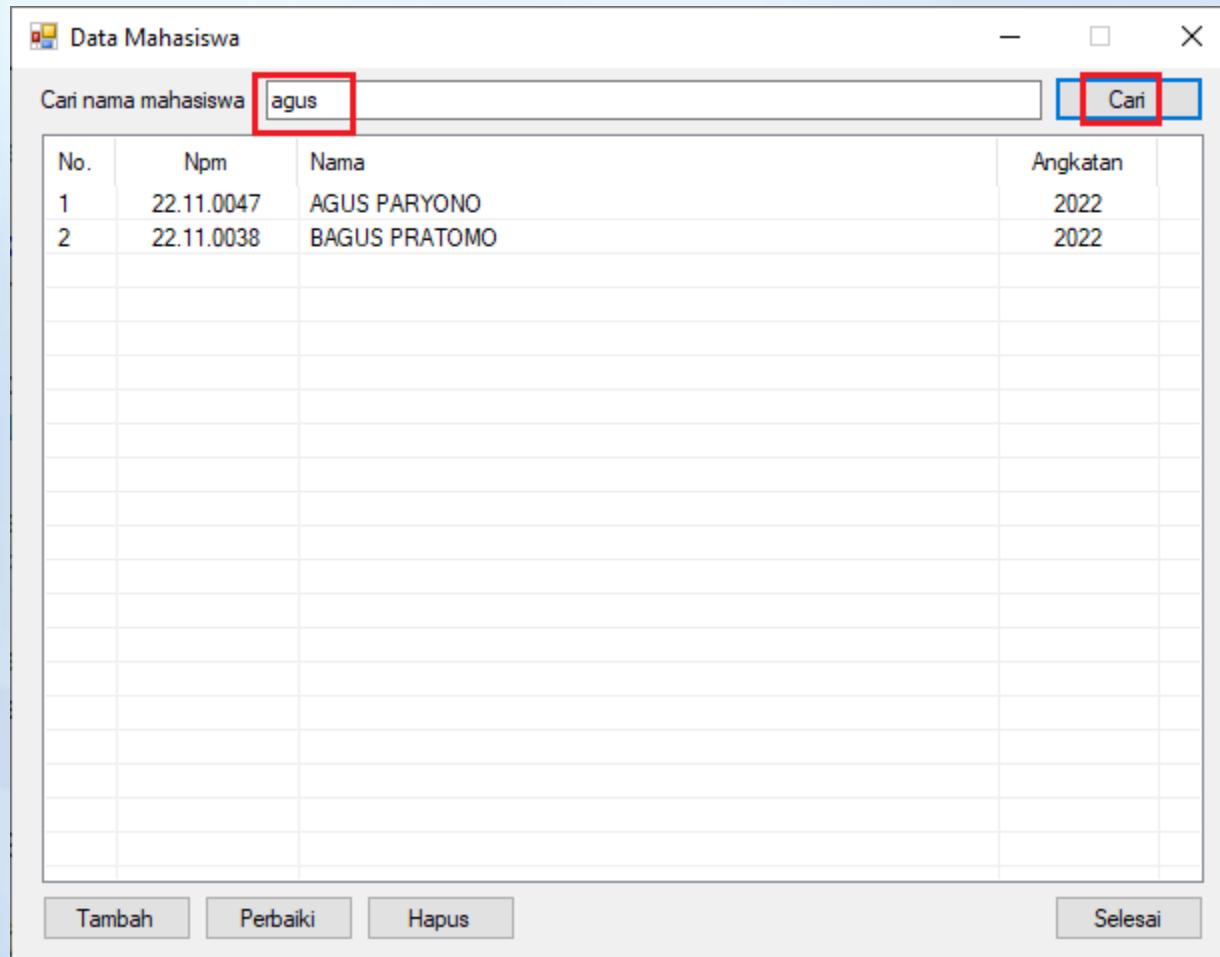
Method *ReadByNama* digunakan untuk melakukan pencarian data mahasiswa berdasarkan nama. Adapun informasi alamat REST API yang diakses sebagai berikut:

- ✓ Base Url
 - <http://latihan.coding4ever.net:5555/>
- ✓ Endpoint
 - api/mahasiswa?**nama**=diisi nama mhs
- ✓ HTTP Method
 - GET

Silahkan dicoba sendiri 😊

Hasil Latihan 4

Setelah menyelesaikan latihan 4, jalankan aplikasi dan pastikan outputnya seperti berikut:



Data Mahasiswa

Cari nama mahasiswa Cari

| No. | Npm | Nama | Angkatan |
|-----|------------|---------------|----------|
| 1 | 22.11.0047 | AGUS PARYONO | 2022 |
| 2 | 22.11.0038 | BAGUS PRATOMO | 2022 |

Tambah Perbaiki Hapus Selesai

Latihan 5: Melengkapi Kode untuk Method Create

Method *Create* digunakan untuk menambahkan data baru. Adapun informasi alamat REST API yang diakses sebagai berikut:

- ✓ Base Url
 - <http://latihan.coding4ever.net:5555/>
- ✓ Endpoint
 - api/mahasiswa
- ✓ HTTP Method
 - POST
- ✓ Body
 - Objek mahasiswa

Latihan 5 (Lanjutan): Melengkapi Kode untuk Method Create

```
public int Create(Mahasiswa mhs)
{
    string baseUrl = "http://latihan.coding4ever.net:5555/";
    var endpoint = "api/mahasiswa";

    // membuat objek rest client
    var client = new RestClient(baseUrl);

    // membuat objek request
    var request = new RestRequest(endpoint, Method.POST);

    // tambahkan objek mahasiswa ke dalam body
    request.AddJsonBody(mhs);

    // request ke server
    var response = client.Execute(request);

    // cek nilai content, jika 1 berhasil selain itu gagal
    var result = response.Content.IndexOf("1") > 0 ? 1 : 0;
}

return result;
```

JSON → { "result": 1 }

Latihan 5 (Lanjutan): Revisi Method Create di Class MahasiswaController

```
// membuat objek context menggunakan blok using
/*
using (DbContext context = new DbContext())
{
    // membuat objek class repository
    _repository = new MahasiswaRepository(context);

    // panggil method Create class repository untuk menambahkan data
    result = _repository.Create(mhs);
}
*/
var repo = new MahasiswaRestApiRepository();
result = repo.Create(mhs);

if (result > 0)
{
    MessageBox.Show("Data mahasiswa berhasil disimpan !", "Informasi",
                    MessageBoxButtons.OK, MessageBoxIcon.Information);
}
else
    MessageBox.Show("Data mahasiswa gagal disimpan !!!", "Peringatan",
                    MessageBoxButtons.OK, MessageBoxIcon.Exclamation);
```

Hasil Latihan 5

Setelah menyelesaikan latihan 5, cobalah dengan menambahkan data baru, kemudian cek hasilnya apakah sudah tersimpan.

The image shows two windows of a Windows application for managing student data. The left window is titled "Tambah Data Mahasiswa" (Add Student Data) and contains three text input fields: "NPM" (22.11.1234), "Nama" (PAIJO), and "Angkatan" (2022). The "Simpan" (Save) button is highlighted with a red box. The right window is titled "Data Mahasiswa" (Student Data) and displays a table with one row of data. The table has columns "No.", "Npm", "Nama", and "Angkatan". The data is: No. 1, Npm 22.11.1234, Nama PAIJO, and Angkatan 2022. A search bar at the top of this window contains the text "paijo", and the "Cari" (Search) button is also highlighted with a red box. A red arrow points from the "Simpan" button in the first window to the "Cari" button in the second window, indicating the flow of data entry and search.

| No. | Npm | Nama | Angkatan |
|-----|------------|-------|----------|
| 1 | 22.11.1234 | PAIJO | 2022 |

Latihan 6: Melengkapi Kode untuk Method Update

Method *Update* digunakan untuk mengupdate data yang sudah ada. Adapun informasi alamat REST API yang diakses sebagai berikut:

- ✓ Base Url
 - <http://latihan.coding4ever.net:5555/>
- ✓ Endpoint
 - api/mahasiswa/**NIM**
- ✓ HTTP Method
 - PUT
- ✓ Body
 - Objek mahasiswa

Latihan 7: Melengkapi Kode untuk Method Delete

Method *Delete* digunakan untuk menghapus data. Adapun informasi alamat REST API yang diakses sebagai berikut:

- ✓ Base Url
 - <http://latihan.coding4ever.net:5555/>
- ✓ Endpoint
 - api/mahasiswa/**NIM**
- ✓ HTTP Method
 - DELETE
- ✓ Body
 - dikosongkan

Selesai ☺