# Cara Menggunakan Synchronizer Library SynX

Pada project Infrastructure, tambahkan nuget project berikut:

* SynX
* SynX.Transport.Ftp
* SynX.FileAdapter.SimpleXml

## Handling Sync Get

Sync get terdiri atas 2 bagian:

1. Scheduler yang akan memeriksa file sync dari suatu tempat (ftp/folder sharing).
2. Sync handler yang akan memproses data hasil sync.

### Scheduler Sync Get

Buat sebuah file pada project Infrastructure pada folder Services/Scheduler dengan nama misalkan GoodReceiptSyncSchedulerService.cs

Class tersebut harus implement dari IBackgroundService, pada method ExecuteService, panggil pengecekan terhadap sync get.

public class GoodReceiptSyncSchedulerService:IBackgroundService {  
 public async Task ExecuteService() {  
 var sync = SyncEngine.CreateInstance(“default”);  
 sync.CheckSyncGet();   
 }  
}

Jika dibutuhkan untuk sync pada spesifik jenis sync tertentu, misal untuk get data sync goodreceipt, maka bisa dengan mengirimkan parameter id sync terhadap method CheckSyncGet.

sync.CheckSyncGet(“default”);

## Sync Handler

Pada file yang sama (GoodReceiptSyncSchedulerService.cs) tambahkan implement interface terhadap SynX.Core.ISync dan implementasikan kedua methodnya.

public class GoodReceiptSyncSchedulerService:IBackgroundService, ISync {  
 ...  
 void OnFileReceived(string syncId, string idNo,  
 Dictionary<string, object> payload, string syncLogId)  
 {  
 ... implementasikan proses received file  
 }  
  
 void OnFileResponseReceived(string syncId, string idNo,  
 Dictionary<string, object> payload, string syncLogId)  
 {  
 ... implementasikan proses received file respon  
 }  
}

Perbedaan dari **OnFileReceived** dan **OnFileResponseReceived** adalah, **OnFileReceived** akan dipanggil jika sync dari IDNo (nomor transaksi) yang dikirim belum pernah ada. Sedangkan jika sudah pernah, maka yang dipanggil adalah method **OnFileResponseReceived**.

## Sync Set

Sync set digunakan untuk mengirimkan file sync ke aplikasi tujuan. File sync yang dikirim bisa merupakan sebuah file request maupun file respon. Gunakan method **SyncService.SendSyncSet()** atau **SyncService.SendSyncSetResponse()**.

Parameter dari kedua method tersebut adalah:

* String syncId : berisi informasi id sync dari konfigurasi applications.config
* String recordId: berisi informasi id record dari data yang akan dikirim (jika ada)
* Dictionary<string, object>: berisi associative dictionary untuk membentuk file xml yang akan dikirim.

Log sync get maupun sync set akan disimpan dalam sebuah table dengan nama SyncLog. Struktur tablenya adalah sebagai berikut:

|  |  |  |
| --- | --- | --- |
| Field | Tipe Data | Keterangan |
| Id | Nvarchar | Id unik (GUID) |
| RecordId | Nvarchar | Record id yang terkait dengan sync |
| IdNo | Nvarchar | IDNo (ID transaksi) dengan aplikasi tujuan/sumber |
| SyncType | Nvarchar | Tipe sync sesuai file konfigurasi |
| FileName | Nvarchar | Nama file sync |
| FileDate | DateTime | Tanggal file di sync |
| IsSyncOut | Bit | True jika sync keluar, false jika sync masuk |
| IsResponseFile | Bit | True jika sync respon, false jika bukan |
| SyncStatus | Nvarchar | Status sync (SENT TO SAP, RECEIVED, EXCEPTION, FAILED) |
| ErrorMessage | Nvarchar | Pesan kesalahan |