Answers to the questions

1. I would add a new endpoint, connect it to some message queuing system and proceed incoming changes in FIFO. So API would be only an entry point, and if needed a queue consumer to deliver address for a new/changed document.

I would strongly avoid processing file in an API, that highly error prone and resource consuming. That would be also a bottleneck of whole system. Instead processing it in a queue is a protection against concurrent changes and flooding the endpoint with huge amount of data.

2. Basically it is following CQRS (Command Query Responsibility Segregation) architectural pattern, which is some kind of extension of letter S from SOLID, it separates data retriving from modifications. I used Windsor.Castle as a container for inversion of control, so most of business logic is using interfaces to proceed. Data layer (XPO ORM) is separated from other parts. DTO's are using AutoMapper to map from domain-specific models to db or business models. Persistance project is resonsible for data storing in DB. Pieces of code are small and strictly related to its purpose (Single responsibility rule).

Running instructions:

(sorry I know its complicated)

1. What you need:

a) Windows OS (7 or higher)

b) IIS (7.5 or higher) (usually added from Server Manager or Programs and Function (Control Panel))

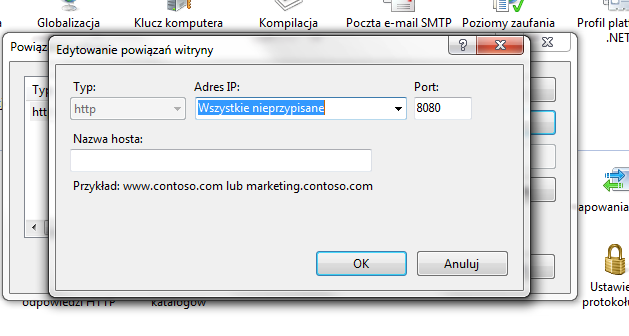
c) MSSQL 2014 (https://www.microsoft.com/en-us/download/details.aspx?id=42299) - please choose -> https://www.microsoft.com/en-us/download/details.aspx?id=42299 to download and install

d) .NET Framework 4.7.2 (https://support.microsoft.com/en-us/help/4054530/microsoft-net-framework-4-7-2-offline-installer-for-windows)

After you install above software, please follow:

2. Import db\_backup.bak to your instance of MSSQL Server

3. Create a new website in IIS and unzip there a content from SpreadsheetAPI.zip, please set binding port to 8080

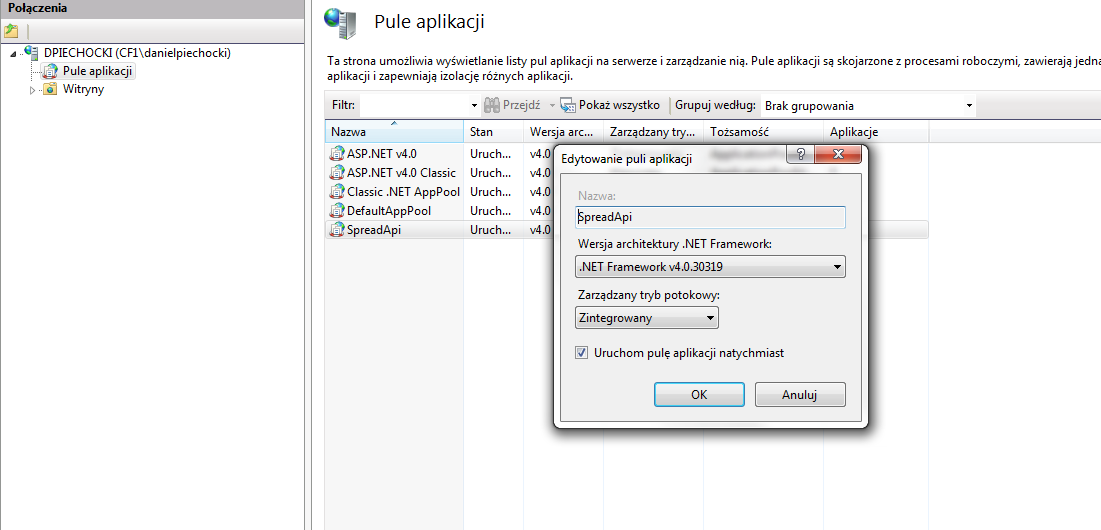


4. Modify the Web.config file on your local machine in IIS directory for newly created website and update connection string to your local MSSQL instance:

a) Here is default setting from my PC: <DatabaseSettings ConnectionString="XpoProvider=MSSqlServer;Data Source=DPIECHOCKI\SQLEXPRESS;Initial Catalog=SpreadsheetAPI;User ID=SpreadUser;Password=someSimplePassword;Application name=Spreadsheets" CreateOption="SchemaOnly" Assembly="Domain, Version=1.0.0.0, Culture=neutral, PublicKeyToken=null" />

b) Here is the same setting with marked ({}) fields to change: <DatabaseSettings ConnectionString="XpoProvider=MSSqlServer;Data Source**={YourMSSQLInstanceName}**;Initial Catalog={DBNameIfYouChangeIt};User ID={YourDBUser};Password={YourDBUserPassword};Application name=Spreadsheets" CreateOption="SchemaOnly" Assembly="Domain, Version=1.0.0.0, Culture=neutral, PublicKeyToken=null" />

5. Please set application pool in IIS for your website so it targets .NET 4 framework and integrated mode:



6. From command-line please run following commands:

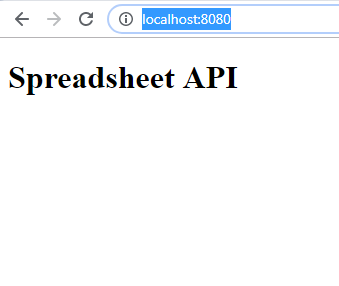
cd c:\Windows\Microsoft.NET\Framework64\v4.0.30319 -> hit enter

aspnet\_regiis -i – hit enter

7. Open your favorite browser and type address:

<http://localhost:8080/>

If you will see this info everything is fine:

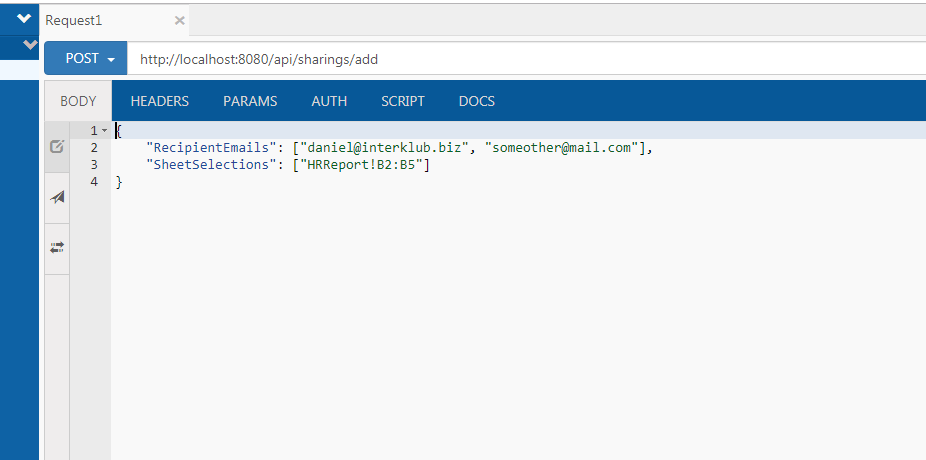


Otherwise something went wrong, something i did not expect.

**I know this procedure is long and error prone, alternatively we can set a call and I will share my screen, so you can see it is working.**

Sample calls (screens from Boomerang [SOAP & REST Client Extension for Chrome]):

Request -> please use POST,



POST http://localhost:8080/api/sharings/add HTTP/1.1

Content-Type: application/json

{

"RecipientEmails": ["daniel@interklub.biz", "someother@mail.com"],

"SheetSelections": ["HRReport!B2:B5"]

}

Response:

HTTP/1.1 200 "OK" 656ms

date: Fri, 17 May 2019 20:09:08 GMT

server: Microsoft-IIS/7.5

x-powered-by: ASP.NET

content-type: application/json; charset=utf-8

access-control-allow-origin: chrome-extension://eipdnjedkpcnlmmdfdkgfpljanehloah

access-control-allow-credentials: true

unique\_id: 2a799782-0131-44fb-803d-c88434e0ac64

content-length: 22

"Operation successful"

**Listing results**

Request:

GET http://localhost:8080/api/sharings/list/all HTTP/1.1

Content-Type: application/json

Response:

HTTP/1.1 200 "OK" 30ms

date: Fri, 17 May 2019 20:13:57 GMT

server: Microsoft-IIS/7.5

unique\_id: 7be1cd68-cea6-4f3c-a4ed-cd74421374e6

x-powered-by: ASP.NET

content-length: 131

content-type: application/json; charset=utf-8

[{"Recipient":"daniel@interklub.biz","Selection":"HRReport!B2:B5"},{"Recipient":"someother@mail.com","Selection":"HRReport!B2:B5"}]