Activating AspCoreGen 6.0 MVC Pro Plus

Note: You need internet connection to activate AspCoreGen 6.0 MVC. You can install one (1) license of AspCoreGen 6.0 MVC for up to 2 computers you own or use for work. You need one (1) license per developer and the license cannot be shared with other developers.

The first time you open AspCoreGen 6.0 MVC Professional Plus edition you will be presented with an activation form right after the Splash screen as shown in Figure 1. You will not be shown the activation form when using the Express edition.

AspCoreGen 6.0 MVC Pr	ofessio	nal+ Activation	
Serial Number:	1		0
Activation Code:	2		Ø
	3	Get Confirm Code	
Confirm Code:	4		0
	5	Activate Exit	
	Тоа	ctivate please enter the following information in order	
	1 -	inter the Seciel Number	
	1. E 2. E	inter the Activation Code.	
	3. C A	lick the Get Confirm Code button. .n email will be sent to the Email Address you used when you purchased our product.	
	4. E 5. L	nter the Confirm Code you received in your email. astly, click the Activate button.	
	5. L		

Figure 1 Activation Form

Enter the *Serial Number (1)* and *Activation Code (2)* we provided you in the respective text boxes. The *Serial Number* and *Activation Code* contains dashes "-", make sure to include these dashes when entering them. Then click the *Get Code* button to get the *Confirm Code*. See Figure 2.

oreGen 6.0 MVC Pi	ofessional+ Activation				
Corial Number	1 11/05 0040 4500				
Senar Number.	HJSE-9313-1F8G				
Activation Code:	2 91888-59827-1234R				
	3 Get Confirm Code	Ø			
Confirm Code:	4				
commin code.	-				
	5 Activat	•	Exit		
	To activate please ente	the following inform:	ation in order:		
	ro delivate, picase cine	the following informe	alon in order.		
	1. Enter the Serial Num	ber.			
	2. Enter the Activation (ode.			
	An email will be sent	the Email Address	vou used when vou	u purchased our pro	duct.
			,,,		
	4. Enter the Confirm Co	de you received in yo	our email.		

Click the *Get Confirm Code (3)* button next. An error will pop up when you enter an invalid *Serial Number* or *Activation Code*. See Figure 3.



Figure 3 Invalid Serial Number or Activation Code

When this happens, click *OK* to close this message. Re-enter the correct *Serial Number* and *Activation Code* and then click the *Confirm Code* button. The *Confirm Code* will be sent via email to the original purchaser's email address we have on file. When the *Confirm Code* is sent, a pop up message will show as seen in Figure 4. Click the *OK* button.



Figure 4 Confirm Code Sent

Enter the *Confirm Code (4)* you received from your email in the respective box and then click the *Activate* button. See Figure 5.

AspCoreGen 6.0 MVC Pi	rofessional+ Activation	
Serial Number:	1 HJSE-9313-1F8G	0
Activation Code:	2 91888-59827-1234RT	0
	3 Get Confirm Code	
Confirm Code:	4 a0913b32ld33r	0
	5	
	To activate please enter the following information in order:	
	Enter the Seriel Number	
	2. Enter the Activation Code.	
	 Click the Get Confirm Code button. An email will be sent to the Email Address you used when you purchased our product. 	
	Enter the Confirm Code you received in your email. Lastly_click the Activate button	

Figure 5 Enter Confirm Code

If you enter an invalid *Serial Number, Activation Code,* or *Confirm Code* a warning pops up. When this happens, click the *Ok* button (See Figure 6), then re-enter the *Serial Number, Activation Code,* and *Confirm Code* Again.



Figure 6 Invalid Serial Number, Activation Code, or Confirm Code

If you enter a valid *Serial Number, Activation Code* and *Confirm Code*, you will be presented with the main window of the application, see Figure 7. You will only see the Activation Form once. After that, you will always go straight to the main window.

elected Tables	Selected Views	Database Settings	Code Settings	UI Settings	App Settings	
Database C	onnection					
Server:	localhos	t				
Patabase	e Name:					
O User Nar	me:					
Passwor	d:			0	Show Passwo	rd
Generated S	SQL					
😧 🔾 Us	e Linq-to-Entities (Entity Framework Cor	e)			
😮 🔾 Us	se Stored Procedur	es	😧 🔾 Use	Ad Hoc/Dynam	ic SQL	
Stored	Procedure		Generated	SQL Script		
00	No Prefix or Suffix		😮 💿 Au	Itomatic		
0	Prefix:		₽ ○ M	S SQL 2008 _B	elow	
0	Suffix					
	_					
bout Close		Genera	te Code for All T	ables	С	ancel

Figure 7 Main Window

A Simple Interface

To keep AspCoreGen 6.0 MVC simple, there's only one main interface as shown in Figure 7. The main window consists of six (6) tabs.

- 1. **Selected Tables**: AspCoreGen 6.0 MVC generates code from all the tables in your database by default. You can choose to generate from selected tables only from the Code Settings tab, and then select just the tables you need to generate from on this tab.
- 2. **Selected Views**: You can choose to generate from selected views only from the Code Settings tab, and then select just the views you need to generate from on this tab.
- 3. **Database Settings:** This is where you enter the database you want to generate code from and whether you want to generate linq-to-entities (entity framework core), stored procedures, or dynamic SQL. This is probably going to be your most used tab.
- 4. **Code Settings**: You'll find a selection here on where to generate your objects from: all tables, all views, selected tables, or selected views. The language the generated code will be in is C#.

Three projects can be generated and are shown in this tab in 3 separate boxes:

- a. Web Application (ASP.NET Core 6.0 Project).
- b. Business Layer and Data Layer API (Core 6.0 Class Library Project).
- c. Web API (ASP.NET Core 6.0 Web API Project). This is optional.

You can enter the Name of each project and the directory for the Web Application. Directories for the Business Layer and Data Layer API and Web API projects are filled automatically. You can choose to Generate Code Examples, and also a Web API project.

- 5. **UI Settings**: You can customize your own settings for the generated ASP.NET MVC Views here. You can choose themes for the JQuery UI controls. You can also select which MVC Views to generate and the MVC View's filename prefix to use for each MVC View.
- 6. **App Settings**: These are application settings. You can see a list of files that are overwritten every time you generate code for the same web project. You can also see a list of files that are only written once every time you generate code for the same web project. These (overwritten and written-once) files are listed per project (web project, class library, web api project).

You can also reset all settings to its original default from here.

That seems like a lot of features, you're probably asking "where's the One Click feature?"

Since AspCoreGen 6.0 MVC remembers the last settings you used such as, e.g. server, database name, directory, namespace, etc., the next time you open AspCoreGen 6.0 MVC, and you're developing for the same web project, you can just click the Generate... button, that simple.

A Quick Tour

Let's learn how to generate an ASP.NET Core Web Application, a Class Library (our middle tier and data tier), the optional ASP.NET Core Web API project using AspCoreGen 6.0 MVC. **We're going to use** *Microsoft's* **Northwind database for this demo. Please Google and download it.** Or you can use your own database; just follow the steps shown below.

- 1. Open the AspCoreGen 6.0 MVC app.
- 2. Click the "*Generate Code for All Tables*" button at the bottom of the app. Notice an error message box pops up showing the required fields to fill. See Figure 8.



Figure 8 Error Message

- 2. Click the *OK* button to close the pop up message. Go to the *Database Settings* tab and start filling the required fields. Do the same in the *Code Settings* tab. See Figures 9 and 10.
 - a. **Database Connection:** This is the MS SQL Server Database connection settings to the database. All of these items are required and cannot be blank.
 - Server: Server Name. If the MS SQL Server Database is local, you can just enter "localhost".
 - Database Name: The database you're trying to access.
 - User Name: An admin user name login that has access to the database.
 - **Password:** The password associated with the user name.

Note: Checking the *Show Password* in Figure 9 will remember the password you enter here so you don't have to enter it again the next time you open AspCoreGen 6.0 MVC.

- b. Generated SQL: SQL script that will be generated.
 - Use Linq-to-entities (Entity Framework Core): Entity Framework.
 - Use Stored Procedures: Scripts generated straight in your MS SQL database.
 - **No Prefix or Suffix:** Names the generated script with no prefix or suffix. E.g. *Products_SelectAll.*
 - Prefix: Names the generated script with a prefix. For example, when you enter "sp_" on the Prefix text box, the generated stored procs will start with an "sp_". E.g. sp_Products_SelectAll.
 - Suffix: Names the generated script with a suffix. For example, when you enter "_sp" on the Suffix text box, the generated stored procs will end with an "_sp". E.g. Products_SelectAll_sp.
 - Use Ad-Hoc/Dynamic SQL: Scripts generated in your C# code.
 - Automatic: Automatically determines your MS SQL Server version before generating scripts.
 - MS SQL 2008 Below: Generates scripts for MS SQL Server versions 2008 and below. There's a slight difference with the generated script for older versions because some keywords are not available during this time.

alected	d Tables Sel	ected Views	Database Settings	Code Settings	UI Settings	App Settings
Data	abase Conne	ction				
0	Server:	localhos	t			
0	Database Nam	ne: Northwir	nd			
0	User Name:	sa				
Ø	Password:	mypassv	word		0	Show Password
Gen	erated SOI					
Gen						
(😮 🔾 Use Lind	q-to-Entities (I	Entity Framework Core	:)		
(🛛 🔾 Use Sto	red Procedure	es	😧 🔾 Use /	Ad Hoc/Dynam	ic SQL
	Stored Proc	edure		Generated	I SQL Script	
	😧 🔍 No P	refix or Suffix		😮 💿 Au	tomatic	
		x:		8 O MS	5 SQL 2008 _B	elow
	😧 🔿 Suffi	x				
	•					
	•					
	•					
	- Club					
out	Close		Generat	e Code for All Ta	ables	Cancel

Figure 9 Database Settings Tab

- a. **Database Object to Generate From:** Generates code from the Database Name you entered on the *Database Settings* tab.
 - All Tables: Generates code for all the database tables.
 - All Views: Generates code for all the database views. Not available for EF Core.
 - Selected Tables Only: Generates code for selected database tables. Clicking this will open the Selected Tables tab. You can load all the tables from your database here and select (check) the tables you want to generate code from.
 - **Selected Views Only:** Generates code for selected database views. Clicking this will open the *Selected Views* tab. You can load all the views from your database here and select (check) the views you want to generate code from. Not available for EF Core.
- b. **Web Application:** There are 3 projects that can be generated by AspCoreGen 6.0 MVC. The web project is the user interface (front end). It houses the MVC views, javascript, css files, etc.
 - Name: Web application name.
 - **Directory:** Directory where the web app will be generated.
 - Generate Code Examples: Generates code examples placed in the web app.
 - **Dev Server Port:** The port number used when you run your web app while developing locally using Visual Studio. The web app will open up in a browser and at the address bar you'll see something like this: http://localhost:**27229**/index. 27229 here is the dev server port.

- c. **Business Layer and Data Layer API:** A class library project. This houses middle-tier (business object) and data-tier (data layer) classes. These classes can be reusable in your other application. Middle-tier classes are referenced by the web app in CRUD operations by default unless the web API project is generated.
 - Name: Class library application name.
 - **Directory:** Directory where the class library app will be generated.
- d. Web API: A web API project. This project is optional.
 - **Use Web API:** When checked, a web API project is generated and referenced by the front end (web application) in CRUD operations instead of the middle-tier.
 - Name: Web API application name.
 - **Directory:** Directory where the web API app will be generated.

elected Tables	Selected Views	Database Setting	s Code Settings	UI Settings	App Settings
Database Obj	ects to Genera	te From			
😮 🗿 All Table	es				
O All View	s				
Selected	d Tables Only				
Selected	d Views Only				
Web Applicati	on (User Interf	ace)			
Name:	МуАрр				
Directory:	C:\inetpub\ww	wroot\			browse
🕜 🔽 Generat	e Code Examples	ົ່	Dev App URI Ports:	7233	5233
Middle Layer (Name:	(Shared API Li MyAppApi	braries)			
Middle Layer ((Shared API Li MyAppApi	braries)			
Middle Layer (Name: Directory:	(Shared API Li MyAppApi C:\inetpub\www	braries) wroot\MyApp\MyA	ррАрі		
Middle Layer (Name: Directory:	(Shared API Li MyAppApi C:\inetpub\www	braries) wroot\MyApp\MyA	ррАрі		
Middle Layer (Name: Directory: Web API (Wel	(Shared API Li MyAppApi C:\inetpub\www	braries) wroot\MyApp\MyA	ррАрі		
Middle Layer (Name: Directory: Web API (Web Use Web	(Shared API Li MyAppApi C:\inetpub\www b Services) b API MyAppSprcs	braries) wroot\MyApp\MyA	ррАрі		
Middle Layer (Name: Directory: Web API (Web Vse Web Name: Directory	(Shared API Li MyAppApi C:\inetpub\www b Services) b API MyAppSrvcs	braries) wroot\MyApp\MyA	ррАрі		
Middle Layer (Name: Directory: Web API (Wel Use We Name: Directory:	(Shared API Li MyAppApi C:\inetpub\www b Services) b API MyAppSrvcs C:\inetpub\www	braries) wroot\MyApp\MyA wroot\MyApp\MyA	.ppApi AppSrvcs		
Middle Layer (Name: Directory: Web API (Wel Veb Use We Name: Directory:	(Shared API Li MyAppApi C:\inetpub\www b Services) b API MyAppSrvcs C:\inetpub\www	braries) wroot\MyApp\MyA wroot\MyApp\MyA	ppApi AppSrvcs	7111	5111
Middle Layer (P Name: Directory: Web API (Wel Vae Web Name: Directory:	(Shared API Li MyAppApi C:\inetpub\www b Services) b API MyAppSrvcs C:\inetpub\www	braries) wroot\MyApp\MyA wroot\MyApp\MyA	ppApi AppSrvcs Dev App URL Ports:	7111	5111

Figure 10 Code Settings Tab

3. Simply hover over any of the *Question Mark* images if you need information from the respective fields.

4. Click the "*Generate Code for All Tables*" button. AspCoreGen 6.0 MVC will start generating code. See Figure 11.

	Selected Views Database Settings Code Settings UI Settings App Settings
Database O	bjects to Generate From
😮 💿 All Ta	ables
	ews
😮 🕕 Selec	ted Tables Only
O Select	ted Views Only
Web Applica	ation (User Interface)
Name:	МуАрр
Directory	/: C:\inetpub\wwwroot\ browse
😮 🔽 Gene	rate Code Examples Dev App URI Ports: 7233 5233
Directory	C:\inetpub\wwwroot\MyApp\MyAppApi
Web API (W	c: C:\inetpub\www.root\MyApp\MyAppApi
Web API (W	:: C:\inetpub\www.root\MyApp\MyAppApi /eb Services) Veb API
Web API (W ② Use V ③ Name:	:: C:\inetpub\www.root\MyApp\MyAppApi /eb Services) //eb API MyAppSrvcs
Web API (W C Use V R Name: Directory	C:\inetpub\www.root\MyApp\MyAppApi /eb Services) Neb API MyAppSrvcs /: C:\inetpub\www.root\MyApp\MyAppSrvcs
Web API (W ? Use V ? Name: ? Directory	C:\inetpub\wwwroot\MyApp\MyAppApi /eb Services) //eb API MyAppSrvcs /: C:\inetpub\wwwroot\MyApp\MyAppSrvcs /: C:\inetpub\wwwroot\MyApp\MyAppSrvcs /: Dev App URL Ports: 7111 5111
Web API (W P Use V Name: Directory out Close	C: C:\inetpub\www.root\MyApp\MyAppApi Veb Services) Veb API MyAppSrvcs (C:\inetpub\www.root\MyApp\MyAppSrvcs C:\inetpub\www.root\MyApp\MyAppSrvcs Dev App URL Ports: 7111 5111 Generate Code for All Tables Cance

Figure 11 Generating Code

5. Wait for a few seconds. When AspCoreGen 6.0 MVC is done generating objects, a message pops up. See Figure 12.



Figure 12 Done Generating Objects

6. Click the *Close Application* button to close the message box and exit the application.

 To view the generated Web Application simply go to the directory you specified from the *Code Settings* tab of AspCoreGen 6.0 MVC. You will see three folders and a solution file like the one shown in Figure 13.

МуАрр	×	+				
🕀 New -	0 lì			✓		
$\leftrightarrow \rightarrow \checkmark \uparrow$	📒 « Local Disk (C	:) > ir	netpub > wwwroot > MyApp	~ C	Q	Search MyApp
A Home			Name	Date modified		Туре
> 🗋 OneDrive - Perso	nal		📜 МуАрр	2/17/2023 4:52 F	M	File folder
		_	📒 МуАррАрі	2/17/2023 4:52 F	PM	File folder
E Desktop		*	MyAppSrvcs	2/17/2023 4:52 F	PM	File folder
↓ Downloads		*	MyApp.sln	2/17/2023 4:52 F	PM	Visual Studio Solution
Documents		*				

Figure 13 Generated Web Application

8. Double-Click the solution file (*MyApp.sln*). An *Elevated Permissions* dialog may appear. Click *Restart this application under different credentials*. See Figure 14.

Microsoft Visual Studio X
Do you want to restart the application with elevated permissions?
The solution or project you want to open is located in a directory that this application does not have permission to work in. If you grant permission by using the User Account Control feature of your operating system, the application may be able to complete the requested tasks.
Restart this application under different credentials
Saves the current changes and then restarts the application. You will be prompted to change your user account.
→ Ignore
Does not elevate user rights, and continues to run this application under the current user rights.
Cancel

Figure 14 Elevated Permissions Dialog

9. Double-Click the solution file (MyApp.sln) in Visual Studio's recent apps list as seen in Figure 15.



Figure 15 Visual Studio 2022 – Recent Apps

10. The generated Web Application will now be opened/loaded into Visual Studio 2022. See Figure 16.

File Edit View Project Build Debug Test Analyze Tools Extensions Window Help	Р МуАрр — 🗆 🗙
© • ⊙ ∰ • ≝ 🗎 🗿 り • ୯	
Tooltox	Solution Explorer ↓ × Image: Solution Explorer (Ctrl+.) Image: Solution Splorer (Ctrl+.) Image: Solution 'MyApp' (3 of 3 projects) Image: Solution 'MyApp' (3 of 3 projects) Image: Solution 'MyAppApi Image: Solution 'MyAppApi
	Solution Team Exp Git Chan Notificati
CodeLens Error List Output	Properties • 4 × MyApp General • General • UserSecretsId • UserSecretsId
\Box	↑ Add to Source Control 🔺 🚨 📓

Figure 16 Generated Web Application in Visual Studio

- 11. Since we generated the optional Web API project by checking the *Use Web API* under the Code Settings Tab, we need to run both the generated Web Application project and the Web API project. Do the following:
 - a. Visual Studio, right-click the Solution name in the Solution Explorer.
 - b. Choose Set Up Startup Projects.
 - c. Choose *Multiple startup projects* in the Solution Property Pages pop up.

d. Under *Action*, choose *Start* for both the *Web App* and *Web API* projects. Note: If you did not change the default project naming under the Code Settings tab of the application, the Web App should be the 1st one on the list and the Web API is the last one on the list.

e. Click OK.

Solution 'MyApp' Property Pages		? ×	
Configuration: N/A	V Platform: N/A	 Configuration Manager 	
Common Properties Startup Project Project Dependencies Code Analysis Settings Debug Source Files b Configuration Properties	Current selection Single startup project MyApp Multiple startup projects:	v	
	Project MyApp MyAppApi	Action Start V None V	
	MyAppSrvcs	Start *	
		OK Cancel Apply	

Figure 17 Multiple Startup Projects

11. Run the solution by pressing *F5*. Two browsers will run, the web application and the web API. In the web application, you will see a list of all the generated ASP.NET Core Views in the *Home Page*. You can click any link to preview the functionality of each of the generated MVC View. However, we're not going to discuss this in this tutorial. In the web API you will see the swagger index page, it shows the web methods that was generated by AspCoreGen 6.0 MVC. You can test these web methods here. See Figure 18.



Figure 18 Shown Side-by- Side. Web App (left), Web API (right)

12. When you click any of the links on the web application you are redirected to that specific page. Play around to see the functionality of each web page. See Figure 19, this is an example of one of the pages.

U L	https://localho	st:7233/Produ	ct/ListCrudRed	irect					A» Q	20	5
МуАрр											
List of Pro	ducts										
LISC OF THO	ducts										
Add New Pro	duct										
List of Broduct											
Product ID \$	Product Name	Supplier ID	Category ID	Quantity Per Unit	Unit Price	Units In Stock	Units On Order	Reorder Level	Discontinued		-
	1 Chai	1	1	10 boxes x 20 bags	\$18.00	39	0	10		0	*
											-
	2 Chang	1	1	24 - 12 oz bottles	\$19.00	17	40	25			
	2 Chang 3 Aniseed Syrup	1	1	24 - 12 oz bottles 12 - 550 ml bottles	\$19.00 \$10.00	17	40 70	25		0	-
	2 Chang 3 Aniseed Syrup 4 Chef Anton's Cajun S	1	1 2	24 - 12 oz bottles 12 - 550 ml bottles 48 - 6 oz jars	\$19.00 \$10.00 \$22.00	17 13 53	40 70 0	25 25 0			
	 2 Chang 3 Aniseed Syrup 4 Chef Anton's Cajun S 5 Chef Anton's Gumbo 	1 1 2 2	1 2 2 2 2	24 - 12 oz bottles 12 - 550 ml bottles 48 - 6 oz jars 36 boxes	\$19.00 \$10.00 \$22.00 \$21.35	17 13 53 0	40 70 0	25 25 0 0			
	2 Chang 3 Aniseed Syrup 4 Chef Anton's Cajun S 5 Chef Anton's Gumbo 6 Grandma's Boysenbe	1 1 2 2 3	1 2 2 2 2 2 2	24 - 12 oz bottles 12 - 550 ml bottles 48 - 6 oz jars 36 boxes 12 - 8 oz jars	\$19.00 \$10.00 \$22.00 \$21.35 \$25.00	17 13 53 0 120	40 70 0 0	25 25 0 0 25			
	 2 Chang 3 Aniseed Syrup 4 Chef Anton's Cajun S 5 Chef Anton's Cumbo 6 Grandma's Boysenbe 7 Uncle Bob's Organic 1 	1 1 2 2 3 3 3	1 2 2 2 2 2 7 7	24 - 12 oz bottles 12 - 550 ml bottles 48 - 6 oz jars 36 boxes 12 - 8 oz jars 12 - 1 lb pkgs.	\$19.00 \$10.00 \$22.00 \$21.35 \$25.00 \$30.00	17 13 53 0 120 15	40 70 0 0 0	25 25 0 0 25 10			
	 2 Chang 3 Aniseed Syrup 4 Chef Anton's Cajun S 5 Chef Anton's Cumbo 6 Grandma's Boysenbe 7 Uncle Bob's Organic I 8 Northwoods Cranber 	1 1 2 2 3 3 3 3	1 2 2 2 2 7 7 2	24 - 12 oz bottles 12 - 550 ml bottles 48 - 6 oz jars 36 boxes 12 - 8 oz jars 12 - 1 lb pkgs. 12 - 12 oz jars	\$19.00 \$10.00 \$22.00 \$21.35 \$25.00 \$30.00 \$40.00	17 13 53 0 120 15 6	40 70 0 0 0 0	25 25 0 0 25 10			
	 Chang Aniseed Syrup Chef Anton's Cajun S Chef Anton's Cumbo Crandma's Boysenbe Uncle Bob's Organic I Northwoods Cranber Mishi Kobe Niku 	1 2 2 3 3 3 3 4	1 2 2 2 2 2 2 7 7 2 6	24 - 12 oz bottles 12 - 550 ml bottles 48 - 6 oz jars 36 boxes 12 - 8 oz jars 12 - 1 lb pkgs. 12 - 12 oz jars 18 - 500 g pkgs.	\$19.00 \$10.00 \$22.00 \$21.35 \$25.00 \$30.00 \$40.00 \$97.00	17 13 53 0 120 15 6 29	40 70 0 0 0 0 0	25 25 0 25 25 10 0 0			
	2 Chang 3 Aniseed Syrup 4 Chef Anton's Cajun S 5 Chef Anton's Cumbo 6 Crandma's Boysenbe 7 Uncle Bob's Organic I 8 Northwoods Cranber 9 Mishi Kobe Niku 10 Kura	1 2 2 3 3 3 3 4 4 4	1 2 2 2 2 7 7 2 6 8 8	24 - 12 oz bottles 12 - 550 ml bottles 48 - 6 oz jars 36 boxes 12 - 8 oz jars 12 - 11 b pkgs. 12 - 12 oz jars 18 - 500 g pkgs. 12 - 200 ml jars	\$19.00 \$10.00 \$22.00 \$21.35 \$25.00 \$30.00 \$40.00 \$97.00 \$31.00	17 13 53 0 120 15 6 29 31	40 70 0 0 0 0 0 0 0 0	25 25 0 0 25 10 0 0 0 0			

Figure 19 Example Page – Products/ListCrudRedirect

13. From the web application's *Home Page*, Each black bar is the specific Project generated by AspCoreGen 6.0 MVC. Each black bar contains blue bars. Each blue bar on this page is clickable. Each blue bar is a section of the generated project. When clicked, each blue bar toggles which would either show or hide the list beneath them. See Figure 20.

	Generated ASP.N	ET Core Code - 🗙	+								
$\leftarrow \rightarrow$	C	https://localh	ost:7233			A∌	Q	ŵ	£≡	œ	
MyApp Thank You application Microsoft S Controllers	for using AspC Is/projects for I SQL Server. Son 5, Business Laye	oreGen 6.0 MVC ise with Visual St ne of the main ob rs, Data Reposito	Professional+. As udio 2022 or late jects generated : ries, View Model	spCoreGen 6.0 M er. In addition, s are listed below ls, Models, and 1	MVC Profession tored proced r in their resp Web API Con	onal+ lures w ective	gener vere al proje s.	ated 3 so gene cts inclu	rated in ding: V	n your iews,	
1. Web A	pplication Proj	ect - ASP.NET Cor	e Web Applicatio	on (.NET Core)							
VIEWS - View	rs Folder (click to sh	ow list)									
CONTROLLER	tS - Controllers Fold	er (click to show list)									
2. Busine	ss Layer/Data F	Repository - Class	Library Project ((.NET Core)							
BUSINESS LA	YER/MIDDLE TIER -	BusinessLayer Folder (e	lick to show list)								
DATA REPOS	ITORY/DAL/DATA TI	ER - DataRepository Fo	lder (click to show list	Ð							
VIEW MODEL	S - ViewModels Fol	der (click to show list)									
MODELS - M	odels Folder (click to	show list)									
3. Web A	Pl Project - ASI	P.NET Core Web #	pplication (.NET	Core)							
CONTROLLER	IS - Controllers Fold	er (click to show list)									
4. Stored	Procedures										
STORED PRO	CEDURES - In Micro	soft SQL Server (click t	show list)								

Figure 20 Three Generated Projects

14. From the web API's swagger page we can test any of the web methods on the list as seen in Figure 21. The web method shown below gets the *number of records* from the Category table, just click *Try it out*.

🖸 📵 Swagger UI x +			×
← C (♪ https://localhost.7111/swagger/index.html A [®] t₀ t	Ē		
GET /CategoryApi/getrecordcount	^	`	٩
Parameters Try it	out		+
No parameters			fr Fr
Responses			دي م
Code Description L	.inks		+
200 Success	Vo links		
Media type text/plain Controls Accept. header.			
Example Value Schema			
•			Ð
		ļ	ŝ

Figure 21 Testing a Web API Method

15. Click the *Execute* button when it comes up.

Swagger UI × +			
← C	A* to t=	Ē	
GET /CategoryApi/getrecordcount		^	
Parameters	Cano	el	
No parameters			
Execute			

Figure 22 Execute

16. A response code 200 (success) is returned. In the response body it shows **8**, which is the number of records in the Categories table. It also shows the web API *Request URL*, which would be the URL the client needs to call to access this web API method. See Figure 23.

-	Intips://iocamost./iii/swagger/index.ntm		A"	20	£≡	
GET	/CategoryApi/getrecordcount					
Paramete	re			ſ	Cano	el.
T urunioto				L	Carro	
No parame	eters					
	Execute	Clear				
Response	25					
Curl						
_						
curl -X '	GET' \ //localhost:7111/CategoryApi/getrecordcount' \					_
curl -X ' 'https: -H 'acc	GET' \ //localhost:7111/CategoryApi/getrecordcount' \ ept: text/plain'					[
curl -x ' 'https: -H 'acc Request UF	GET' \ //localhost:7111/CategoryApi/getrecordcount' \ sept: text/plain' RL					
curl -X ' 'https: -H 'acc Request UR https://2	GET`\ //localhost:7111/CategoryApi/getrecordcount`\ eept: text/plain` RL localhost:7111/CategoryApi/getrecordcount					
curl -X ' 'https: -H 'acc Request UR https://T	GET' \ //localhost:7111/CategoryApi/getrecordcount' \ ept: text/plain' RL localhost:7111/CategoryApi/getrecordcount bonse					
curl -X ' https: -H 'acc Request UR https://X Server resp Code	GET' \ //localhost:7111/CategoryApi/getrecordcount' \ eept: text/plain' RL localhost:7111/CategoryApi/getrecordcount conse Details					
curl -X 'https: -H 'acc Request UR https:/// Server resp Code 200	GET' \ //localhost:7111/CategoryApi/getrecordcount' \ eept: text/plain' RL localhost:7111/CategoryApi/getrecordcount bonse Details Response body					
curl -X 'https: -H 'acc Request UR https://3 Server resp Code 200	GET' \ //localhost:7111/CategoryApi/getrecordcount' \ eept: text/plain' Cocalhost:7111/CategoryApi/getrecordcount Docalhost:7111/CategoryApi/getrecordcount Docalhost:7111/CategoryApi/getrecordcount Source Details Response body 8					
curl -X 'https: 'https: -H 'acc Request UR https:/// Server resp Code 200	GET' \ //localhost:7111/CategoryApi/getrecordcount' \ eept: text/plain' RL localhost:7111/CategoryApi/getrecordcount ponse Details Response body 8			Ê	Downl	oa
curl -X ' 'https: -H 'acc Request UR https:/// Server resp Code 200	GET' \ //localhost:7111/CategoryApi/getrecordcount' \ rept: text/plain' RL localhost:7111/CategoryApi/getrecordcount conse Details Response body Response headers		(Downl	loa
curl -X ' 'https: -H 'acc Request UF https://1 Server resp Code 200	GET' \ //localhost:7111/CategoryApi/getrecordcount' \ eept: text/plain' RL localhost:7111/CategoryApi/getrecordcount bonse Details Response body 8 Response headers content-type: application/json; charset=utf-8			ġ	Downl	loa
curl -X ' 'https: -H 'acc Request UF https://i Server resp Code	GET' \ //localhost:7111/CategoryApi/getrecordcount' \ exept: text/plain' Clubeline Conse Details Response body Response headers content-type: application/json; charset=utf-8 date: Tue,21 Feb 2023 05:43:50 GMT server: Kestrel				Downl	loa
curl -X ' 'https: -H 'acc Request UR https://j Server resp Code 200	GET' \ //localhost:7111/CategoryApi/getrecordcount' \ rept: text/plain' RL localhost:7111/CategoryApi/getrecordcount conse Details Response body Response body Response headers content-type: application/json; charset=utf-8 date: Tue_21 Feb 2023 05:43:50 GMT server: Kestrel			4 >	Downl	loa
curl -X ' 'https: -H 'acc Request UF https:/// Server resp Code 200 Responses Code	GET' \ //localhost:7111/CategoryApi/getrecordcount' \ rept: text/plain' RL localhost:7111/CategoryApi/getrecordcount conse Details Details Response body 8 Response headers content-type: application/json; charset=utf-8 date: Tue,21 Feb 2023 05:43:50 GMT server: Kestrel Description			Ê	Downl	oa

Figure 23 Web Response

- 16. Close the web application's page and go back to Visual Studio 2022. Expand each one of the projects in the *Solution Explorer*. The generated code's layout are separated in 3 projects. See Figure 24.
 - a. MyApp Web Project. Presentation (UI) Layer.
 - b. MyAppApi Class Library Project. Business Layer and Data Layer (Data Repository), and reusable/shared code.
 - c. MyAppServices Web API Project. Web methods.



Figure 24 Three Projects Generated By AspCoreGen 6.0 MVC Shown in Visual Studio 2022

This tutorial offers a quick look in using AspCoreGen 6.0 MVC. You can read end-to-end tutorials on more subjects on using AspCoreGen 6.0 MVC Professional Plus that came with your purchase. These tutorials are available to customers and are included in a link on your invoice when you purchase AspCoreGen 6.0 MVC Professional.

Note: Some features shown here are not available in the Express Edition.

End of tutorial.