# SharpShooter Reports.Web MVC の基本的な使い方

Last modified on: November 15, 2012

※本ドキュメント内のスクリーンショットは英語表記ですが SharpShooter Reports JP(日本語版)では日本語で表示されます。

# 目次

システムの必要条件
よじめに3
Neb アプリケーションの作成
手順 1. Web プロジェクトの作成 3
手順 2. Web アプリケーションの設定5
手順 3. アセンブリ参照の追加6
手順 4. レポートサービスの追加
手順 5. データソースの作成9
手順 6. サービスにデータを追加する 13
手順 7. レポートスロットの追加 13
手順 8. ウィザードを使ったレポート作成17
手順 9. レポートの設定 21
手順 10. ナビゲーションの追加 23
手順 11. コントローラの追加 24
手順 12. サービスの役割をするコントローラの追加 25
手順 13. スクリプトファイルを追加する 25
手順 14. スタイルの追加 27
手順 15. イメージの追加
手順 16. コントローラの追加
手順 17. ビューの追加
手順 18. ページにスクリプトやスタイルを追加する 33
手順 19. Web ページにレポートを表示する 34
手順 20. 外観設定
手順 21. ページのマークアップ

# システムの必要条件

ASP.NET MVC Web アプリケーションで SharpShooter Reports.WebViewer を使用するには以下が必要です。

- .NET Framework 4
- Visual Studio 2010
- ASP.NET MVC 3

# はじめに

このマニュアルは基本的な使い方を説明し、SharpShooter Reports.Webの使用に最低限必要なスキルを 提供します。SharpShooter Reports.Web を使用した Web アプリケーションの作成方法について順を追っ て説明していきます。サービスの作成および設定方法、レポートの作成方法、Web ビューアを統合す る方法を説明します。

手順1~8はこのアプリケーションのサーバー部分の作成および設定について説明しています。

手順9~14 はクライアントアプリケーションの設定について説明しています。

# Web アプリケーションの作成

#### 手順 1. Web プロジェクトの作成

「SharpShooterWebViewerMVC」という ASP.NET MVC 3 Web アプリケーションの新規プロジェクトを作 成します。Visual Studio のメインメニューから [新規作成\プロジェクト...] を選択します。

File	Edit View Debug Team Data Tools Tes	Ana	lyze Window Help	
	New		Project	Ctrl+Shift+N
	Open	1	Web Site	Shift+Alt+N
	Close	1	Team Project	
ā	Close Solution	1	File	Ctrl+N
	Save Selected Items Ctrl+S		Project From Existing Code	
	Save Selected Items As			

「OK」ボタンをクリックします。

ASP.NET MVC 3 Web アプリケーションを選択し、「名前」フィールドにプロジェクト名を入力し、



#### 「空」のテンプレートを選択します。

Select a template:	Description: An empty ASP.NET MVC 3 project.	*
View engine: Razor  Create a unit test project		Ŧ
Test <u>p</u> roject name: SharpShooterWebViewerMVC.Tests		
Test <u>f</u> ramework: Visual Studio Unit Test	→ Additional <u>I</u> nfo	
·	OK Cance	

# 手順 2. Web アプリケーションの設定

コンテキストメニューを開き、[プロパティ]を選択します。「SharpShooterWebViewer」プロジェクト のプロパティをここで変更できます。



「Web」タブの[Visual Studio 開発サーバーを使用する]を選択し、特定ポートを「5555」に設定します。

PERPETUUM www.pe	erpetuumsoft.com
SharpShooterWebViewerMV	c ×
Application Build	Configuration: N/A    Platform: N/A
Web	Start Action
Package/Publish Web	© Cu <u>r</u> rent Page © <u>S</u> pecific Page
Package/Publish SQL	Start external program
Silverlight Applications	Command <u>l</u> ine arguments
Build Events	Working directory
Resources	© Start <u>U</u> RL
Settings	On't open a page. Wait for a request from an external application.
Reference Paths	Servers
Signing	Apply server settings to all users (store in project file)
Code Analysis	Ose Visual Studio <u>D</u> evelopment Server
	O Auto-assign Port
	Sp <u>e</u> cific port     5555

# 手順3.アセンブリ参照の追加

アセンブリ参照を追加します。ソリューションエクスプローラの「SharpShooterWebViewerMVC」プロ ジェクトのコンテキストメニューから [参照の追加] を選択します。

Soluti	ion Ex	plorer	тųх						
		2 3 1							
	Solut	ion 'SharpShooterWebViewerMVC' (1 project)							
4	🧟 S	harpShooterWebViewerMVC							
		Build	- 1						
		Rebuild							
		Clean							
	6	Build Deployment Package							
	2	Publish							
	3	Package/Publish Settings							
		Run Code Analysis							
		Convert to Web Application							
	0	Check Accessibility							
		Calculate Code Metrics							
		Add	-						
		Add Reference							
		Add Service Reference							
		Add Deployable Dependencies							
		Add Library Package Reference							
	æ,	View Class Diagram							
		Set as StartUp Project							

プロジェクトに次のアセンブリを追加します。

- · PerpetuumSoft.Reporting.WebViewer.Model.dll;
- · PerpetuumSoft.Reporting.WebViewer.Server.dll;
- · PerpetuumSoft.Reporting.WebViewer.Server.Mvc.dll;
- · System.ServiceModel.dll.

VET COM Projects Browse Recent Filtered to: .NET Framework 4							
Component Name Version Runtime Path							
PerpetuumSoft.Reporting.Silverlight.Export	5.3.1.63	v2.0.50727	C:\Program Files (x86)\Perpetuum Software\N				
PerpetuumSoft.Reporting.Silverlight.Server	5.3.1.63	v2.0.50727	C:\Program Files (x86)\Perpetuum Software\N				
PerpetuumSoft.Reporting.Web	5.3.1.63	v2.0.50727	C:\Program Files (x86)\Perpetuum Software\N				
PerpetuumSoft.Reporting.WebViewer.Client	1.0.0.0	v4.0.30319	C:\Program Files (x86)\Perpetuum Software\Ne				
PerpetuumSoft.Reporting.WebViewer.Client.Model	1.0.0.0	v4.0.30319	C:\Program Files (x86)\Perpetuum Software\Ne				
PerpetuumSoft.Reporting.WebViewer.Model	5.3.1.63	v2.0.50727	C:\Program Files (x86)\Perpetuum Software\Ne				
PerpetuumSoft.Reporting.WebViewer.Server	5.3.1.63	v2.0.50727	C:\Program Files (x86)\Perpetuum Software\N				
PerpetuumSoft.Reporting.WebViewer.Server.Mvc	5.3.1.63	v4.0.30319	C:\Program Files (x86)\Perpetuum Software\N				
PerpetuumSoft.Writers.Excel	5.3.1.63	v2.0.50727	C:\Program Files (x86)\Perpetuum Software\N				
PerpetuumSoft.Writers.Flash	5.3.1.63	v2.0.50727	C:\Program Files (x86)\Perpetuum Software\N				
PerpetuumSoft.Writers.Pdf	5.3.1.63	v2.0.50727	C:\Program Files (x86)\Perpetuum Software\N				
PerpetuumSoft.Writers.Svg	5.3.1.63	v2.0.50727	C:\Program Files (x86)\Perpetuum Software\N				
System.Web.Mvc	2.0.0.0	v2.0.50727	C:\Program Files (x86)\Microsoft ASP.NET\ASF 🔻				
< III			Þ				

OK

Cancel

Version	Runtime	Path
4.0.0.0	v4.0.30319	C:\Program Files (x86)\Reference Assemblies\N
4.0.0.0	v4.0.30319	C:\Program Files (x86)\Reference Assemblies\N
4.0.0.0	v4.0.30319	C:\Program Files (x86)\Reference Assemblies\N
4.0.0.0	v4.0.30319	C:\Program Files (x86)\Reference Assemblies\N
4.0.0.0	v4.0.30319	C:\Program Files (x86)\Reference Assemblies\N
4.0.0.0	v4.0.30319	C:\Program Files (x86)\Reference Assemblies\N
4.0.0.0	v4.0.30319	C:\Program Files (x86)\Reference Assemblies\N
4.0.0.0	v4.0.30319	C:\Program Files (x86)\Reference Assemblies\N
4.0.0.0	v4.0.30319	C:\Program Files (x86)\Reference Assemblies\N
4.0.0.0	v4.0.30319	C:\Program Files (x86)\Reference Assemblies\N
4.0.0.0	v4.0.30319	C:\Program Files (x86)\Reference Assemblies\N
4.0.0.0	v4.0.30319	C:\Program Files (x86)\Reference Assemblies\N
4.0.0.0	v4.0.30319	C:\Program Files (x86)\Reference Assemblies\N =
	Version 4.0.0.0.0 4.0.0.0.0 4.0.0.0.0 4.0.0.0 4.0.0.0.0 4.0.0.0.0.0.0	Version         Runtime           4.0.0.0         v4.0.30319           4.0.0.0         v4.0.30319

#### 手順4.レポートサービスの追加

プロジェクトに、サービスロジックを実装するクラスを追加します。「SharpShooterWebViewerMVC」 プロジェクトのコンテキストメニューを開き、[追加\新しい項目...]を選択します。

	Solution Explorer				
	🌄 Solution 'Sha	rpSho	oterWebViewerMVC' (1 proj	ect)	
_	🔺 🛃 SharoSho	oterV	VebViewerMVC		
	Build	ies			
	Rebuild	ces			
	Clean	Lerc			
3	Build Deployment Package	leis			
	Publish				
1	Package/Publish Settings				
	Run Code Analysis	asax			
	Convert to Web Application	ntig			
<b>Q</b>	Check Accessibility				
	Calculate Code Metrics				
	Add •	1	Area		
	Add Reference	1	New Item	Ctrl+Shift+A	
	Add Service Reference		Existing Item	Shift+Alt+A	
	Add Deployable Dependencies	-**	New Folder		
	Add Library Package Reference		Add ASP.NET Folder	•	
2	View Class Diagram	93	Class	Shift+Alt+C	

#### 「Class」を選択し、「名前」フィールドに「ServiceClass.cs」というクラス名を設定します。

Add New Item - SharpShooterWebViewer	MVC	
Installed Templates	Sort by: Default   Search Installed Templat	tes 🔎
▲ Visual C#	📧 Web Form using Master Page 🛛 👔 🖉 Type: Visual C#	
Data	🗈 Web User Control 📝 🖌 An empty class declara	tion
General ⊿ Web	省 Class 🗔 A	
MVC 3	🔽 Master Page 🕺 🖸	
Windows Forms WPF	📄 Nested Master Page 🔤 🔤	
Script#	🗃 HTML Page 🛛 🙀 🛛	
Extensibility Reporting	🔊 Style Sheet 📓 G	
RSS for SL	🗿 JScript File 🔉 S	
Silverlight SVforRS	🗱 AJAX-enabled WCF Service 📴 S	
Workflow 👻	🖹 ASP.NET Handler 🛛 🙀 V	
Online Templates	< <u> </u>	
Name: ServiceClass.cs		
	<u>A</u> dd	Cancel

~

「ServiceClass」に、System.ServiceModel.Activation、System.Data、 PerpetuumSoft.Reporting.WebViewer.Server 名前空間を追加します。それには、using ディレクティブを 使用します。

```
using System.ServiceModel.Activation;
using System.Data;
using PerpetuumSoft.Reporting.WebViewer.Server;
```

この「ServiceClass.cs」クラスを「ReportServiceBase」クラスから継承します。こうして、既存の実装や 標準の動作を変更する機能を取得します。このサービスクラスに「AspNetCompatibilityRequirements」 属性を付けると、このサービスは ASP.NET のコンテキストにアクセスします。このサービスは、他の キャッシュメカニズムが実装されていない場合、ドキュメントデータのキャッシュに ASP.NET のコン テキストが必要になります。

「ServiceClass」クラスに、「InitializeComponent」メソッドとコンストラクタ(このメソッドを呼び出 します)を追加します。

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.ServiceModel.Activation;
using System.Data;
using PerpetuumSoft.Reporting.WebViewer.Server;
namespace SharpShooterWebViewerMVC
{
    [AspNetCompatibilityRequirements(RequirementsMode = AspNetCompatibilityRequirementsMode.Required)
]
    public class ServiceClass : ReportServiceBase
    {
        public ServiceClass()
        {
            InitializeComponent();
        }
        private void InitializeComponent()
        { }
    }
}
```

#### 手順 5. データソースの作成

「ServiceClass」クラスのデザイナを開き、ソリューションエクスプローラの「ServiceClass.cs」ファイルのコンテキストメニューから[デザイナの表示]を選択します。



まず、データソースの構造を作成します。ツールボックスから「DataSet」を追加します(ツールボッ クスの DateSet をダブルクリックします)。



「型指定のないデータセット」を選択します。

Add Dataset	2 ×
Choose a typed or untyped dataset to add to	the designer.
Typed dataset	
Name:	•
Creates an instance of a typed dataset class with a dataset that has a built-in schema. Se	already in your project. Choose this option to work e Help for details on generating typed datasets.
<u>Untyped dataset</u>	
Creates an instance of an untyped dataset cl when you want a dataset with no schema.	ass of type System.Data.DataSet. Choose this option
	OK Cancel

そうすると、デザイナにデータセットのノード(dataSet1)が表示されます。

×

Pro	operties	<b>*</b> ₽	×
da	taSet1 System.Data.I	DataSet	•
•	2↓ 💷 🖌 🖾		
	DataSetName	NewDataSet	*
	EnforceConstraints	True	
	GenerateMember	True	
	Locale	en-US	
	Modifiers	Private	
	Namespace		
	Prefix		11
	Relations	(Collection)	
	RemotingFormat	Xml	
	Tables	(Collection)	
			+

dataSet1 に「Customers」テーブルを追加します。(「追加」ボタンをクリックし、TableName プロパ ティを「Customers」に設定し、Name プロパティを「customers」に設定します。)

その後、列コレクションエディタ(テーブルコレクションエディタのプロパティグリッドの Columns プロパティのボタン)を開きます。

	1000	14 2.24		
	Cus	stomers <u>P</u> roperties:		
		Columns	(Collection)	
		Constraints	(Collection)	
		DisplayExpression		
	1	MinimumCapacity	50	
		Namespace		
	1	Prefix		
		PrimaryKey	DataColumn[]	
		TableName	Customers	
	۵	Design		
		(Name)	customers	
12-22		GenerateMember	True	
$\mathbf{X}$		Modifiers	Private	
	4	Misc		
	X	Cu:	Customers Properties:	Customers Properties:         Data         Columns       (Collection)         Constraints       (Collection)         DisplayExpression       DisplayExpression         MinimumCapacity       50         Namespace       Prefix         PrimaryKey       DataColumn[]         TableName       Customers         Oesign       (Name)         GenerateMember       True         Modifiers       Private

「追加」ボタンをクリックして列を 2 つ追加し、 ColumnName プロパティをそれぞれ「Name」と 「Phone」に設定します。

Members:		Pho	one Properties:		
0 Name		۵	Data		-
1 Phone			AllowDBNull	True	
			AutoIncrement	False	
			AutoIncrementSee	0	
			AutoIncrementStep	1	
			Caption	Phone	
			ColumnName	Phone	
			DataType	System.String	
			DateTimeMode	UnspecifiedLocal	
			DefaultValue	<dbnull></dbnull>	
			Expression		
			MaxLength	-1	
	X		Namespace		
			Prefix		
Add			ReadOnly	False	-
		_			

#### 手順6.サービスにデータを追加する

データの定義が終わったので、次は Customers テーブルにデータを代入します。ソースコードを表示 するには、デザイナ領域を右クリックし、コンテキストメニューから[コードの表示]をクリックします。

🕂 ServiceClass.cs [Design] 🛛 🗙								
□ □ dataSet1								
š	F	View <u>C</u> ode	F7					
	∦ Cu <u>t</u>		Ctrl+X					
		Сору	Ctrl+C					
	Ē.	<u>P</u> aste	Ctrl+V					
	×	<u>D</u> elete	Del					
	2	P <u>r</u> operties						

ServiceClass クラスの OnLoadData メソッドをオーバーライドして、データソースにデータを代入してください。

protected override void OnLoadData(IDictionary<string, object> parameters, string reportName, PerpetuumSoft.Reporting.Components.ReportSlot reportSlot)

```
{
    base.OnLoadData(parameters, reportName, reportSlot);
    DataRow row = customers.NewRow();
    row["Name"] = "Johnson Leslie";
    row["Phone"] = "613-442-7654";
    customers.Rows.Add(row);
    row = customers.NewRow();
    row["Name"] = "Fisher Pete";
    row["Phone"] = "401-609-7623";
    customers.Rows.Add(row);
    row = customers.NewRow();
    row = customers.NewRow();
    row["Name"] = "Brown Kelly";
    row["Phone"] = "803-438-2771";
    customers.Rows.Add(row);
}
```

}

#### 手順7.レポートスロットの追加

では、(ツールボックスの ReportManager をダブルクリックして) ReportManager コンポーネントを 追加します。このコンポーネントはレポート生成を行います。





ServiceClass クラスの ReportManager プロパティを設定します。それには、[プロパティ]ウィンドウの ServiceClass のプロパティを開き、リストから reportManager1 を選択します。

Se	ServiceClass PerpetuumSoft.Reporting.WebViewer.Server.Repc -							
	(Name)	ServiceClass						
	ImageDPI	96						
	Language	(Default)						
	Localizable	False						
	ReportCache							
⊳	ReportManager	reportManager1 🔹						
		(none)						
		reportManager1						

reportManager1を右クリックし、「エディタの起動」を選択して、レポートマネージャのエディタを 立ち上げます。

ServiceClass.cs	也 Ser	viceO	lass.cs [Design]	×
🖭 dataSet1 🛛 🛄 repo	rtMana	Y	View Code	F7
			Run Editor	
		¥	Cut	Ctrl+X
		Þ	Сору	Ctrl+C
		ß	Paste	Ctrl+V
		×	Delete	Del
		2	Properties	

レポートテンプレートを作成する前に、レポートを生成するデータソースを追加します。「データソ ース」タブのデータバインドリストに「Customers」テーブルを追加します。(「追加」ボタンをクリ ックすると表示される「オブジェクト名の編集」ダイアログの「名称」を「Customers」に、「値」コ ンボボックスから「dataSet1.Customers」を選択します。)

ReportM Data bind	anager editor ata sources		
Edit o	bject name		
Value	customers data Set 1.Customers (none) customers dataColumn 1 dataColumn2 dataSet 1 data Set 1.Customers inlineReportSlot 1 reportManager1 ServiceClass		Cancel
Add	Edit Delete	OK	Cancel

「レポート」タブで「追加」ボタンをクリックして、新しいオブジェクト「InlineReportSlot」を追加し ます。

Reports	Data sources		
Ad	d Delete ileReportSlot	Run Designer	
	an Constant	OK	Cancel
I	nlineReportSlot		

ReportName プロパティを「CustomersReport」に設定します。後でレポートマネージャからその名前で ドキュメントを取得します。次に、「デザイナの起動」ボタンを押してレポートデザイナを立ち上げ ます。

<ul> <li>▲ Design (Name) Generate Member Modifiers</li> <li>▲ Misc Document Exception Mode ReportName</li> </ul>	inlineReportSlot2 True Private Document Waming CustomersReport
ReportName	
Designer	
	Image: Second system       Image: Second system         Image: Second

# 手順8.ウィザードを使ったレポート作成

[ファイル(新規]を選択すると、画面に下図のフォームが表示されます。

20		Repor	t Sharp-Sho	ooter Designer
File	Home	Insert	Layout	View
	New		Ctrl+N	
Ē	Open from file		Ctrl+O	Font
H	Save		Ctrl+S	
B	Save to file	Ctrl	+Shift+S	
	Select Languag	e		
	🔛 Settin	ngs 🗙	Exit Design	er

「新規」タブの「標準のレポート」を選択し、「OK」ボタンを押します。

8	PERPETUUM software	
	🕎 Wizards Gallery	
	New Saved	
	Blank Report	
	Standard Report	
	Pivot table Report	

「スタンダードウィザード」ウィンドウが表示されます。

下図のように、ドキュメントのパラメータを設定します。

📴 Standard Wizard		23
Pages	Common	
+ x 🖻 🗟 🗅	Report template pro	operties
Common	🔽 Show title	Customers
	Measure unit	Centimeter
	Style sheet	None
	Script language	CSharp 🔽
	Page properties — Size — Paper kind A4 Page size 0; 0	Orientation       Page numbers         ● Portrait       Position         ● Landscape       Align
	Columns layout Columns count Columns gap	1
	S	ave Preview template Preview document OK Cancel

Load from file

OK

23

Cancel

「追加」 (十) ボタンを使ってデータソースを追加します。

9	PERPETUUM software	petuumsoft.com						
	Standard Wizard Pages Common (Data Section)	(DataSection)		Visible fields Field name	Groups	Caption	Width	Aggregate
	·	Sav	ve Previ	ew template	Preview do	cument	OK	Cancel

8	PERPETUUM software	www.perpetuumsoft.com
---	-----------------------	-----------------------

🕎 Standard Wizard		23
Pages + X           Q Q Common (Data Section)	(DataSection) – Data source Field layout	(none)  
		Image: Caption       Width       Aggregate       Save       Preview template       Preview document       OK

レポートに出力するフィールドを選択します。(Name フィールドと Phone フィールドの両方を移動してください。)

🕎 Standard Wizard 🛛 🕅 🕅								
Pages Pages Common Customers	Customers Data source Customers Field layout List All fields Customers Name Mame	₽ ↓ 1	Visible fields Groups Field name Name Phone	Caption Name Phone	Width 100 100	Aggregate None None		
		¢	Caption Phone Width 100	III Aggregate	None			
	Save	Previe	ew template Preview of	locument	OK	Cancel		

「OK」ボタンを押します。

# 手順 9. レポートの設定

設定したテンプレートが作成されます。



[プロパティ]ウィンドウで header1 の RepeatEveryPage プロパティを True に設定します。



[プロパティ]ウィンドウで detail1 の NewPageAfter プロパティを True に設定します。

Pro	perties			•	Ţ	×		
det	ail1 Detail					•		
	] ⊉↓   📰 ▫²   📼							
⊿	Appearance							
	StyleName	Normal						
	ToolTip							
⊿	Data							
	Tag							
⊿	Design							
	Name	detail 1						
	ShowCaption	True						
⊿	Layout							
$\triangleright$	Location	0; 1.5 cm						
$\triangleright$	Size	21; 0.7 cm						
۵	Navigation							
	Bookmark							
	Hyperlink							
۵	Render							
	CanBreak	False						
	CanGrow	True						
	CanShrink	True						
	MinSpaceAfter	0 cm						
	NewColumnAfter	False						
	NewColumnBefore	False						
	NewPageAfter	True						
	NewPageBetore	False						
	Visible	True						
۵	Script							
	Aggregates	(Collection)						
	GenerateScript							

#### 手順10.ナビゲーションの追加

レポートにナビゲーションを追加します。レポート間の移動にはブックマークツリーを使用します。

[プロパティ]ウィンドウで detail1\_Customers\_Name のバインドを選択します。

Pro	Properties 🗸 🗸 🖡					
detail1_Customers_Name TextBox						
	)⊉↓   📻 📭 🗃					
$\triangleright$	Margins	0; 0; 0; 0 cm				
	RightToLeft	False				
	StringTrimming	Character				

Bookmark プロパティのスクリプトエディタ (… ボタン)を開きます。

۵	Navigation	
	Bookmark	
	1.1 1.1	

スクリプトエディタに "#" + を入力し、「データソース」タブの「Customer」データソースを展開し、 顧客名を取得するスクリプトコードを追加するために「Name」をダブルクリックします。

🕎 Script Editor			23		
1 📮 "#" + GetData("Customers.Name")	Script Editor				
	Bands	Data So	Parameters		
	⊞ Customers ⊕ Special Fields				
Co	de Snippets	ОК	Cancel		

テンプレートを保存してデザイナを閉じます。

# 手順11.コントローラの追加

プロジェクトにコントローラを追加します。"SharpShooterWebViewerMVC" プロジェクトのコンテキス トメニューから[追加\新しい項目...]を選択します。

	Solution Explorer								
	n an								
Solution 'SharpShooterWebViewerMVC' (1 project)									
A Comparison of the second									
	Properties								
		e e e e e e e e e e e e e e e e e e e	AssemblyInfo.cs						
	📄 licenses.licx								
	References								
		Þ 🚞 (	Content						
			Ontrollers						
	Convert to Web Application		1odels -						
<b>\$</b>	Check Accessibility		cripts						
	Add	•	E Controller						
	Exclude From Project		📴 New Item Ctrl+Shift+A						
*	Cut	Ctrl+X	😳 Existing Item Shift+Alt+A						

コントローラ名を「ReportServiceController」にします。



# 手順 12. サービスの役割をするコントローラの追加

コントローラに "PerpetuumSoft.Reporting.WebViewer.Server.Mvc" 名前空間を追加します。

using PerpetuumSoft.Reporting.WebViewer.Server.Mvc;

作成したコントローラを ReportServiceBaseController から継承し、"CreateReportService"メソッドをオー バーライドします。このオーバーライドされたメソッドは "ServiceClass" クラスのインスタンスを返し ます。従って、 "ReportServiceController.cs" ファイルのコードは次のようになります。

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.Mvc;
using PerpetuumSoft.Reporting.WebViewer.Server.Mvc;
namespace SharpShooterWebViewerMVC.Controllers
{
    public class ReportServiceController : ReportServiceBaseController
    {
        protected override PerpetuumSoft.Reporting.WebViewer.Server.ReportServiceBase CreateReportSer
vice()
        {
            return new ServiceClass();
        }
    }
}
```

アプリケーションのサーバー部分の設定は終わりなので、クライアントアプリケーションの設定を行います。

#### 手順13. スクリプトファイルを追加する

プロジェクトに次のファイルを追加します。

```
jquery-1.5.1.js – jQuery プラグイン
```

jquery.treeview.js-ブックマークツリーの動作を実装します

mscorlib.js-スクリプトでの型システムのような機能や(#スクリプトで作成されたクラスの使用時に必要な)基本的なユーティリティのAPIを提供します

**PerpetuumSoft.Reporting.WebViewer.Client.js** – レポートの取得や表示ロジックを実装するクラス を含んでいます

PerpetuumSoft.Reporting.WebViewer.Client.Model.js – データモデルのクラス

ソリューションエクスプローラの「Scripts」フォルダを選択し、コンテキストメニューの[追加->既存の項目…]を選択します。

	Solution Explorer								
		🖶   🏠 👩	1						
		🌄 Solution 'S	harpS	hooterWebViewerMVC	(1 project)				
	A Comparison of the second								
	Properties								
		Ref	erenc	es					
			ntent ntrolle						
			dels						
		b Scr	ipts						
	Convert to Web Application		vs						
0	Check Accessibility		bal.as	ax					
	Add	•	ncel	New Item	Ctrl+Shift+A				
	Exclude From Project			Existing Item	Shift+Alt+A				
¥	Cut	Ctrl+X	Ľ	New Folder					
	Сору	Ctrl+C		Add ASP.NET Folder	•				
ß	Paste	Ctrl+V	₽\$	Class	Shift+Alt+C				
$\times$	Delete	Del	-						
	Rename								
ſ	Open Folder in Windows Explorer								
	Properties	Alt+Enter							

「Web\Scripts」フォルダから次のファイルを追加します。

jquery-1.5.1.js

jquery.treeview.js

mscorlib.js

PerpetuumSoft.Reporting.WebViewer.Client.Model.js

PerpetuumSoft.Reporting.WebViewer.Client.js

	•	<del>•</del> •	Search Scripts	
н	iquery.treeview jquery-1.5.1 mscorlib PerpetuumSoft.Reportin PerpetuumSoft.Reportin	g.Wel g.Wel	bViewer.Client bViewer.Client.Model	 
- I	rv-1.4.1-vsdoc" "iquerv-1.4.1	•	All Files (*.*)	

#### 手順 14. スタイルの追加

プロジェクトにスタイルを追加します。このスタイルはレポートを正しく表示するために必要になる ので、スタイルを持つ次のファイルをプロジェクトに追加する必要があります。

**ReportViewer.css** – このファイルはレポートの表示スタイルを表します

ソリューションエクスプローラの「Content」フォルダを選択し、コンテキストメニューから[追加->既存の項目...]を選択します。



「Web\Content」フォルダから ReportViewer.css ファイルを追加します。

ReportViewer.css ファイルを選択し、「追加」ボタンをクリックします。

🚧 Add Existing Item - SharpShoot	erWebViewer • • • Searc	h Content
Organize   New folder		iii • 🗔 🔞
( Canada )	^ Name	Date modified
	鷆 images	30-Aug-11 00:30
	A ReportViewer	29-Aug-11 18:16
	• •	
File <u>n</u> ame:	ReportViewer	s (*.*) Id 🔽 Cancel

# 手順 15. イメージの追加

イメージファイルを追加します。これらのイメージはブックマークのツリー表示に使用します。プロ ジェクトに、イメージを格納しているフォルダを追加します。ソリューションエクスプローラの 「Content」フォルダを選択し、コンテキストメニューから[追加\新しいフォルダ]を選択します。

	Solution Explorer										
		🖷   🗗 🧧	]   🎾	)							
	Solution 'SharpShooterWebViewerMVC' (1 project)										
	▲ 🛃 SharpShooterWebViewerMVC										
	Properties										
	▷ 🔂 References										
	Convert to Web Application										
8	Check Accessibility		Re	eportViewer.css							
	Add	•		New Item	Ctrl+Shift+A						
	Exclude From Project			Existing Item	Shift+Alt+A						
X	Cut	Ctrl+X		New Folder							
	Сору	Ctrl+C		Add ASP.NET Folder	•						
6	Paste	Ctrl+V	-	Class	Shift+Alt+C						
X	Delete	Del	Veb.	config							
	Rename			-							
Den Folder in Windows Explorer											
	Properties	Alt+Enter									

フォルダ名を「images」と入力します。



イメージを「images」フォルダに追加し、その追加した「images」フォルダを選択して、コンテキスト メニューから[追加\既存の項目...]を選択します。

「Web\Content\images」に移動し、次のイメージファイルを選択します。

bookmarktreenode.png;

treeview-default.gif;

treeview-default-line.gif.

「追加」ボタンをクリックします。



プロジェクトに必要なファイルをすべて追加したら、レポートビューアを入れるページを作成します。

プロジェクトに、リクエストやビューを処理(データを表示)するコントローラを追加します。

#### 手順 16. コントローラの追加

プロジェクトに、コントローラを追加します。ソリューションエクスプローラの「Controllers」フォル ダを選択し、コンテキストメニューから[追加\コントローラ...]を選択します。



**Controller Name** フィールドに、「HomeController」というコントローラ名を設定します。

e, Update, Delete, and Details scenarios

#### 手順 17. ビューの追加

ビューを追加します。ソリューションエクスプローラの「HomeController.cs」ファイルをダブルクリ ックして開き、「Index」メソッドのコンテキストメニュー(メソッドを右クリック)から[ビューの 追加]を選択します。

HomeController.cs X SharpShooterWebViewerMVC.Controllers.HomeController = Index() + ⊡using System; VS10x - Code Map using System.Collections.Generic; using System.Linq; using System.Web; using System.Web.Mvc; inamespace SharpShooterWebViewerMVC.Controllers { □ ···· public · class · HomeController · : · Controller ••••{ <u>⊨</u>....// ·····//·GET:·/Home/ .....public ActionResult Index() ----{ Open in .NET Reflector ......return View(); Add View... •••••} Go To <u>V</u>iew ····} Refactor ۲ ]} Organize Usings Create Unit Tests... 🔍 Insert Snippet... Ctrl+K, X Surround With... Ctrl+K, S

表示されたウィンドウの「追加」ボタンをクリックします。

Add View
View name:
View <u>e</u> ngine:
Razor (CSHTML)
Create a strongly-typed view
Model class:
▼
Sca <u>f</u> fold template:
Empty   Reference script libraries
Create as a partial view
☑ Use a layout or master page:
(Leave empty if it is set in a Razor _viewstart file)
ContentPlace <u>H</u> older ID:
MainContent
<u>A</u> dd Cancel

# 手順 18. ページにスクリプトやスタイルを追加する

「\_Layout.cshtml」ファイルを開きます。

_Layout.cshtml ×	
Client Objects & Events +	(No Events)
html	
⊡ <html></html>	
⊢ <head></head>	
<pre><title>@ViewBag.Title</title></pre>	
<pre><link.href="@url.content("~ content="" site.css")".rel="&lt;/pre"></link.href="@url.content("~></pre>	"stylesheet"·type="text/css"·/>
<pre><script.src="@url.content("~ jquery-1.4.4.mi<="" pre="" scripts=""></script.src="@url.content("~></pre>	<pre>n.js")" type="text/javascript"&gt;</pre>
i <body></body>	
@RenderBody()	

次の文字列を

<script src="@Url.Content("~/Scripts/jquery-1.4.4.min.js")" type="text/javascript"></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></scr

次のように変更します。

<script src="@Url.Content("~/Scripts/jquery-1.5.1.js")" type="text/javascript"></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script>

スタイルを関連付けるコードを追加します。

```
<link href="@Url.Content("~/Content/ReportViewer.css")" rel="stylesheet" type="text/css" />
```

```
「_Layout.cshtml」のマークアップは次のようになります。
```

```
<!DOCTYPE html>
<html>
<head>
    <title>@ViewBag.Title</title>
    <link href="@Url.Content("~/Content/Site.css")" rel="stylesheet" type="text/css" />
    <link href="@Url.Content("~/Content/ReportViewer.css")" rel="stylesheet" type="text/css" />
    <script src="@Url.Content("~/Scripts/jquery-1.5.1.js")" type="text/javascript"></script>
</head>
<body>
    @RenderBody()
```

```
@RenderBody
</body>
</html>
```

```
(ソリューションエクスプローラの「Index.cshtml」ファイルをダブルクリックして)「Index.cshtml」
ファイルを開きます。
```

```
<script src="@Url.Content("~/Scripts/jquery.treeview.js")" type="text/javascript"></script>
<script src="@Url.Content("~/Scripts/mscorlib.js")" type="text/javascript"></script>
<script src="@Url.Content("~/Scripts/PerpetuumSoft.Reporting.WebViewer.Client.Model.js")"
type="text/javascript"></script>
<script src="@Url.Content("~/Scripts/PerpetuumSoft.Reporting.WebViewer.Client.js")"
type="text/javascript"></script>
<script src="@Url.Content("</script>/PerpetuumSoft.Reporting.WebViewer.Client.js")"
type="text/javascript"></script></script src="@Url.Content("</script>/PerpetuumSoft.Reporting.WebViewer.Client.js")"
```

#### 手順 19. Web ページにレポートを表示する

ビューに div 要素を追加します。この要素はレポートを表示します。div 要素に識別子を設定します。 この識別子は javascript コードから div 要素を取得するために必要です。

```
<div id="ReportViewerElement"> </div >
```

ビューに javascript コードを追加する必要があります。このコードはサーバーからドキュメントを読み 込みます。

```
<script type="text/javascript">
  $(document).ready(function () {
    var reportViewer = new PerpetuumSoft.Reporting.WebViewer.Client.ReportViewer("#ReportViewerElement");
    reportViewer.setServiceUrl("http://localhost:5555/ReportServiceController");
    reportViewer.reportName = "CustomersReport";
    reportViewer.renderDocument();
    reportViewer.setThumbnailsControl("#ssr_thumbnailContentPanel");
    reportViewer.setDocumentMapControl("#documentMapView");
    });
</script>
```

このクラスのオブジェクトはこのコードで作成されます。このオブジェクトはドキュメントを読み込んで Web ページに表示します。オブジェクトを作成する時、レポート表示に使われる Web ページの要素を定義してからサービスのアドレスやレポート名を設定します。renderDocument メソッドはサーバーからのドキュメントの読込を初期化します。

```
Index.htm のコードは次のようになります。
```

```
<mark>@{</mark>
                     ViewBag.Title = "Index";
}
<script src="@Url.Content("~/Scripts/jquery.treeview.js")" type="text/javascript"></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></scri
<script src="@Url.Content("~/Scripts/mscorlib.js")" type="text/javascript"></script>
<script src="@Url.Content("~/Scripts/PerpetuumSoft.Reporting.WebViewer.Client.Model.js")"</pre>
type="text/javascript"></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></
<script src="@url.Content("~/Scripts/PerpetuumSoft.Reporting.WebViewer.Client.js")"</pre>
type="text/javascript"></script>
<script type="text/javascript">
                     $(document).ready(function () {
                                           var reportViewer = new PerpetuumSoft.Reporting.WebViewer.Client.ReportViewer("#ReportViewerElement");
                                           reportViewer.setServiceUrl("http://localhost:5555/ReportService");
                                           reportViewer.reportName = "CustomersReport";
                                           reportViewer.renderDocument();
                                           reportViewer.setThumbnailsControl("#ssr_thumbnailContentPanel");
                                           reportViewer.setDocumentMapControl("#documentMapView");
                     });
                     </script>
 <div id="ReportViewerElement">
</div>
```

ウェブブラウザでアプリケーションを実行すると、次のようなスロバーが表示されます。



ドキュメントのデータを読み込む時にドキュメントの表示領域が表示されます。この後、アプリケー ションは表示するページを定義し、表示するページの読込要求を送ります。ページの読込中に(下図 のような)プログレスバーが表示されます。

http://localhost:55		P+OX W W W
🥭 Index 🛛 🗙	and a second a second s	An Arrest
File Edit View Favorites To	iols Help	
File Edit View Favorites To		
		*

ページが読み込まれると、ブラウザに表示されます。



ページに、レポートのナビゲート、エクスポート、印刷を行うメソッドを呼び出す要素を追加します。

#### 手順20.外観設定

次のサンプルについて説明します。このサンプルには、印刷ボタン、エクスポートボタン、ページナ ビゲーションボタン、ページ数と現在のページ番号があります。簡単にナビゲートするためにブック マークツリーとサムネイルを追加しました。

$\sim$		And a state of the	
( http://localhost:5555/			③ ☆ ① × 2 + Q
🥖 Index 🛛 🗙			
<u>File Edit View Favorites Tools H</u> elp			
Print Export to Rtf Export to Pdf	Export to Excel Export to Html E	Export to Xps	
Pages			
			1
		Customers	
	lohnson Leslie	Phone 613-442-7654	
E		010-442-1004	
1			
an and a			
2			
Bookmark			
E Customers			
Johnson Leslie			
- Eisher Pete			
Brown Kelly			
			<b>T</b>
First page Prev page 1 of 3	3 Next page Last page		

このようなサンプルの作成方法について説明します。

```
手順 21.ページのマークアップ
```

```
この(上図のような)ページのマークアップは次のようになります。
```

```
<div style="margin: 10px">
    <input id="printButton" type="button" value="Print" class="ExportButton" />
    <input id="exportToRtfButton" type="button" value="Export to Rtf" class="ExportButton" />
    <input id="exportToPdfButton" type="button" value="Export to Pdf" class="ExportButton" />
    <input id="exportToExcelButton" type="button" value="Export to Excel" class="ExportButton" />
    <input id="exportToHtmlButton" type="button" value="Export to Html" class="ExportButton" />
    <input id="exportToXpsButton" type="button" value="Export to Xps" class="ExportButton" />
</div>
<div style="height: 600px; margin: auto; border: solid 1px black;">
    <div style="height: 600px; background-color: White; float: left; width: 250px;">
        <div style="height: 300px;">
            <div style="height: 30px; background-color: #CCC; padding: 10px 0px 0px 10px;">
                <span>Pages</span>
            </div>
            <div id="ssr_thumbnailContentPanel" style="width:250px; height: 260px;">
            </div>
        </div>
        <div style="height: 300px;">
            <div style="height: 30px; background-color: #CCC; padding: 10px 0px 0px 10px;">
                <span>Bookmark</span>
            </div>
            <div id="documentMapView">
            </div>
        </div>
    </div>
    <div id="ReportViewerElement" style="height: 600px; background-color: Gray; overflow: auto;">
    </div>
</div>
<div style="margin: 10px;">
    <input id="firstPage" type="button" value="First page"/>
    <input id="prevPage" type="button" value="Prev page" />
    <input id="currentPage" type="text" title="Current page" style="width: 60px;" />
    <span>of </span><span id="pageCount">0</span>
    <input id="nextPage" type="button" value="Next page" />
    <input id="lastPage" type="button" value="Last page" />
</div>
```

次のように、"reportViewer" オブジェクトのイベントハンドラを作成します。

documentInfoLoadedEvent – レポートデータの読込時に発生するイベントで、ページサイズ付き のページー覧を取得できます。

currentPageChangedEvent – 現在のページが変更された時に発生するイベントで、現在のページ 番号を取得できます。

errorEvent - エラーが生じると発生するイベントで、エラー情報を取得できます。

サムネイルやブックマークツリーの出力に必要な Web ページの要素を設定します。

reportViewer.setThumbnailsControl("#ssr\_thumbnailContentPanel"); reportViewer.setDocumentMapControl("#documentMapView");

印刷、エクスポート、ページナビゲーションボタンのハンドラを追加します。

javascript コードは次のようになります。

<script type="text/javascript">

# PERPETUUM www.perpetuumsoft.com

```
var reportViewer = null;
   $(document).ready(function () {
        Initialize();
        $("#printButton").click(function () {
            reportViewer.print();
        }):
        $("#exportToRtfButton").click(function () {
            reportViewer.exportToRtf();
        });
        $("#exportToPdfButton").click(function () {
            reportViewer.exportToPdf();
        }):
        $("#exportToExcelButton").click(function () {
            reportViewer.exportToExcel();
        });
        $("#exportToHtmlButton").click(function () {
            reportViewer.exportToHtml();
       });
        $("#exportToXpsButton").click(function () {
            reportViewer.exportToXps();
        });
        $("#firstPage").click(function () {
            reportViewer.firstPage();
        });
        $("#prevPage").click(function () {
            reportViewer.prevPage();
        });
        $("#nextPage").click(function () {
            reportViewer.nextPage();
       });
        $("#lastPage").click(function () {
            reportViewer.lastPage();
        });
   });
   function Initialize() {
        reportViewer = new PerpetuumSoft.Reporting.WebViewer.Client.ReportViewer("#ReportViewerElement");
        reportViewer.setServiceUrl("http://localhost:5555/ReportService");
        reportViewer.reportName = "CustomersReport";
        reportViewer.documentInfoLoadedEvent = function (pages) {
            $("#pageCount").text(pages.length);
        };
        reportViewer.currentPageChangedEvent = function (pageNumber) {
            $("#currentPage").val(pageNumber);
        };
        reportViewer.errorEvent = function (errorModel) {
            switch (errorModel.errorType) {
                case PerpetuumSoft.Reporting.WebViewer.Client.ErrorType.communicationError:
                    alert("communicationError" + errorModel.error._error$1);
                    break;
                case PerpetuumSoft.Reporting.WebViewer.Client.ErrorType.clientError:
                    alert("clientError" + (errorModel.error).message);
                    break;
                case PerpetuumSoft.Reporting.WebViewer.Client.ErrorType.serverError:
                    alert("serverError" + (errorModel.error).message + (errorModel.error).getInformation());
                    break;
                default:
                    alert(errorModel.error.message);
                    break;
            }
        };
        reportViewer.renderDocument();
        reportViewer.setThumbnailsControl("#ssr_thumbnailContentPanel");
        reportViewer.setDocumentMapControl("#documentMapView");
   }
</script>
```

では、Webアプリケーションを実行します。ブラウザに下図のような新しいデザインのページが表示 されます。

| ~  | 10                             | T sales all halls all a second     |               |
|--|--------------------------------|------------------------------------|---------------|
| (Carlo Market Market S555/   |                                |                                    | ③ 公 ① × O + C |
| 🤗 Index 🛛 🗙 🔛  |                                |                                    |               |
| <u>File E</u> dit <u>V</u> iew F <u>a</u> vorites <u>T</u> ools <u>H</u> elp |                                |                                    |               |
| Print Export to Rtf Export to Pdf  | Export to Excel Export to Html | Export to Xps                      |               |
|  |                                |                                    |               |
| Pages  |                                |                                    |               |
| Bookmark<br>Customers<br>Fisher Pete<br>Brown Kelly                          | Name<br>Johnson Leslie         | Customers<br>Phone<br>613-442-7654 | 1             |
| First page Prev page 1 of 3  | Next page Last page            |                                    |               |