

SharpShooter Reports.Web

MVC アプリケーションでの Web ビューアの使用

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.....	3
はじめに.....	3
手順 1. 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.....	4
手順 3. サービスロジックの実装.....	6
手順 4. コントローラの追加	6
手順 5. サービスの役割をするコントローラの設定.....	7
手順 6. プロジェクトに JavaScript ファイルを追加する	7
手順 7. スタイルの追加.....	9
手順 8. デフォルトのスキン ファイルを追加する	10
手順 9. プロジェクトに「ビュー」を追加する	13
手順 10. ビューにスクリプトやスタイルを追加する	15
手順 11. ビューに Report Viewer コンポーネントを追加する.....	16
手順 12. アプリケーションの実行	19



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。

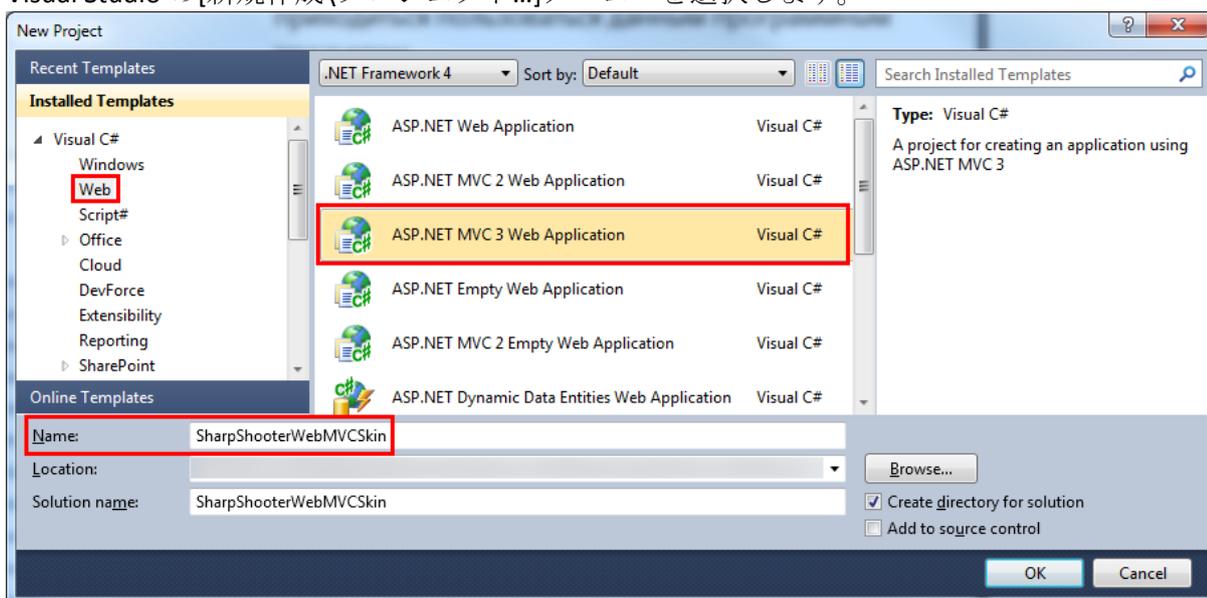
このマニュアルどおりに行えば、Web アプリケーション/ウェブサイトでのサービスの設定や定義済みのスキンを使用したレポートビューアの付加が 15 分ほどで行えるようになります。

では、SharpShooter Reports.Web を使った Web アプリケーションの作成手順について説明します。サービスの作成および設定、レポートの作成、Web ビューアの統合について説明していきます。

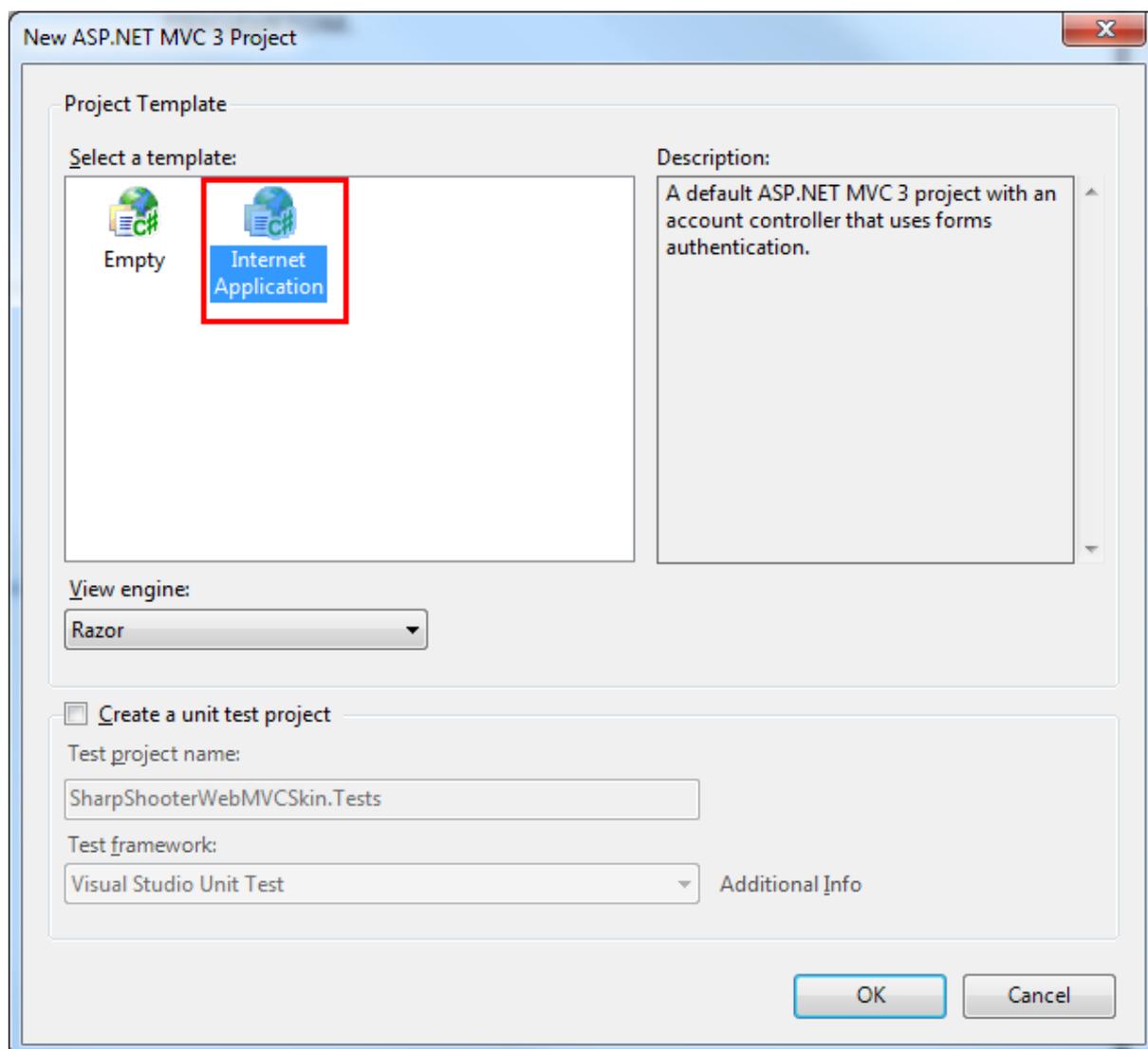
手順 1. プロジェクトの作成

新規プロジェクト「ASP.NET MVC 3 Web アプリケーション」を作成し、プロジェクト名を「SharpShooterWebMVCSkin」にします。

Visual Studio の[新規作成\プロジェクト...]メニューを選択します。

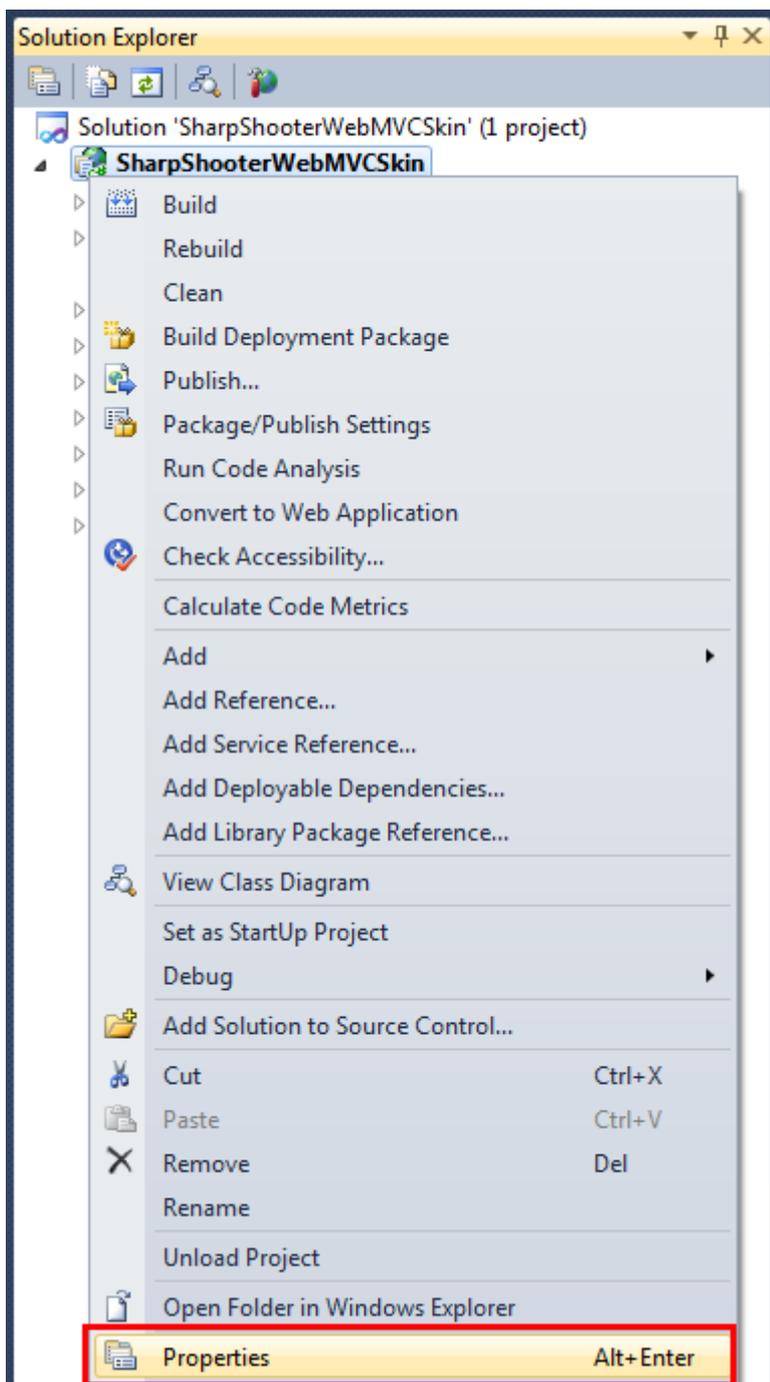


「Internet Application」テンプレートを選択し、「OK」ボタンを押します。

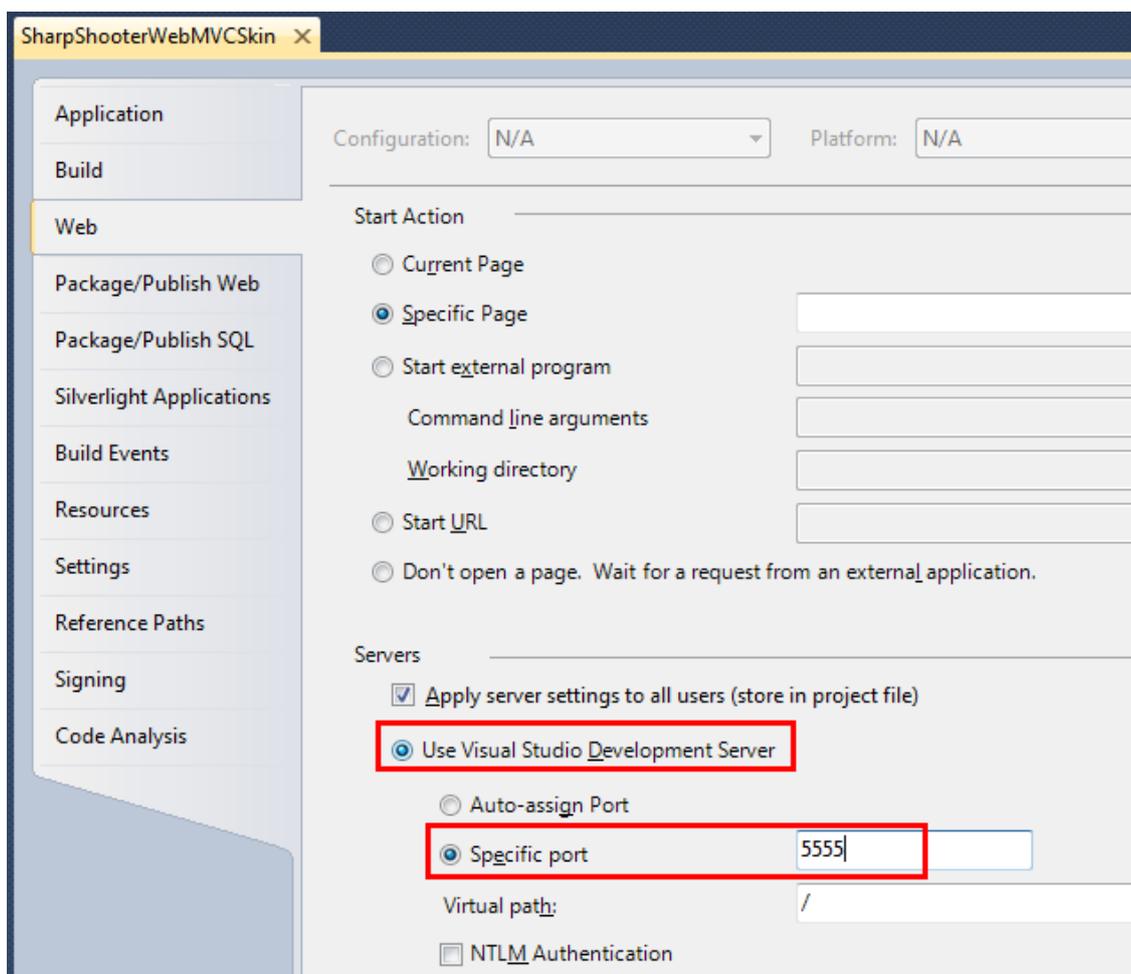


手順 2. Web アプリケーションの設定

「SharpShooterWebMVCSkin」プロジェクトのプロパティを変更します。コンテキストメニューから [プロパティ...] を選択します。



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

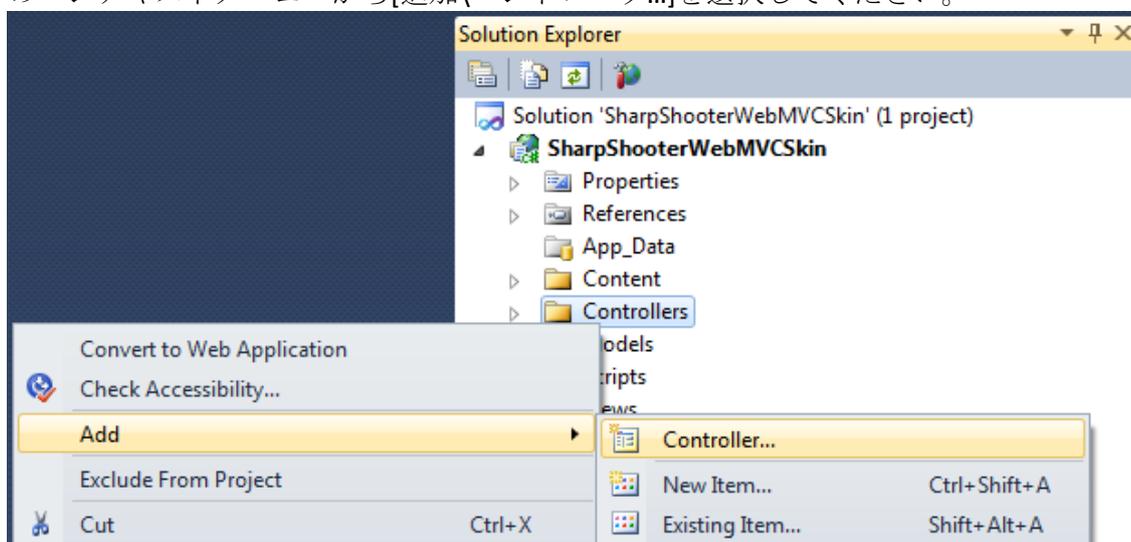


手順 3. サービスロジックの実装

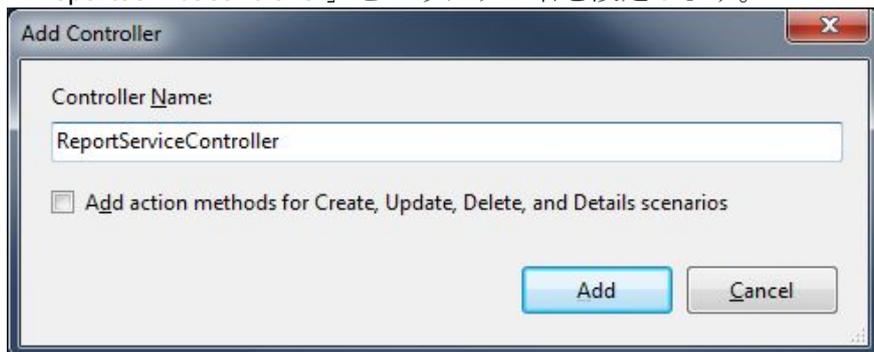
プロジェクトに、Report Engine のサービスを実装するクラスを追加します。このクラスの追加方法の詳細は、[Reporting Engine の設定](#)をご覧ください。

手順 4. 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」フォルダのコンテキストメニューから[追加\コントローラ...]を選択してください。



「ReportServiceController」というクラス名を設定します。



手順 5. サービスの役割をするコントローラの設定

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るはずです。

```
using System;  
using System.Collections.Generic;  
using System.Linq;  
using System.Web;  
using System.Web.Mvc;
```

```
using PerpetuumSoft.Reporting.WebViewer.Server.Mvc;
```

```
namespace SharpShooterWebMVCSkin.Controllers  
{  
    public class ReportServiceController : ReportServiceBaseController  
    {  
        protected override PerpetuumSoft.Reporting.WebViewer.Server.ReportServiceBase CreateReportService()  
        {  
            return new ServiceClass();  
        }  
    }  
}
```

この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。

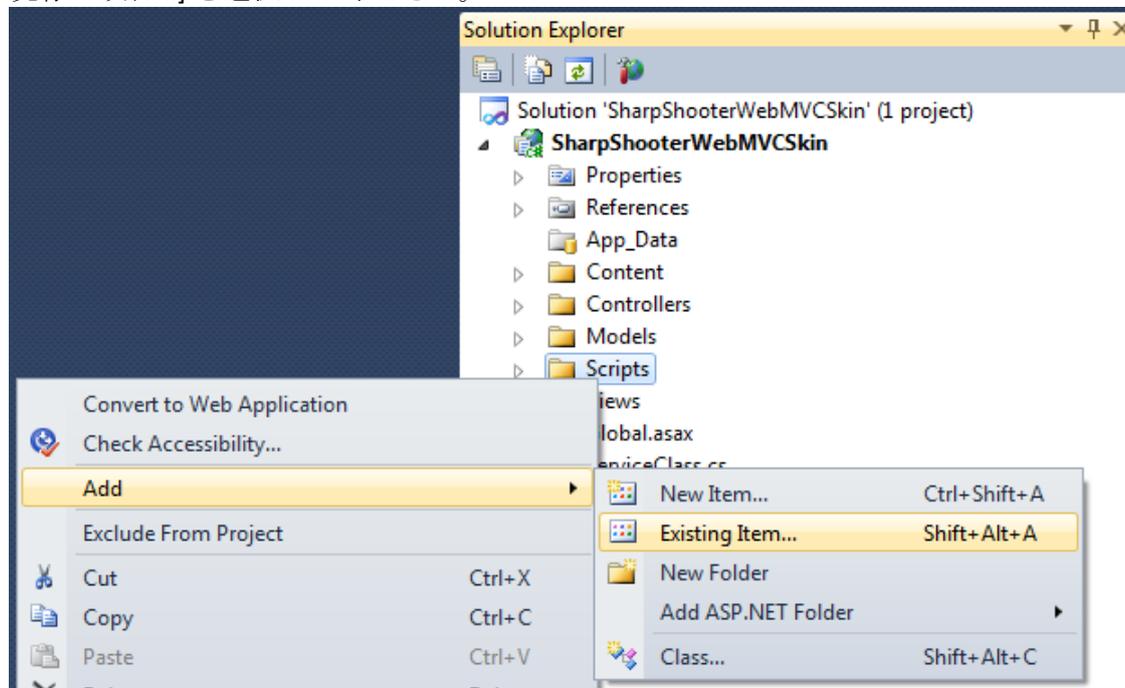
手順 6. プロジェクトに JavaScript ファイルを追加する

以下のファイルをプロジェクトに追加する必要があります。

- jquery-1.5.1.js – jQuery のプラグイン
- jquery-ui-1.8.11.js – UI 用の jQuery のプラグイン
- jquery.treeview.js – ブックマークのツリーの機能を実装します
- mscorlib.js – スクリプトに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

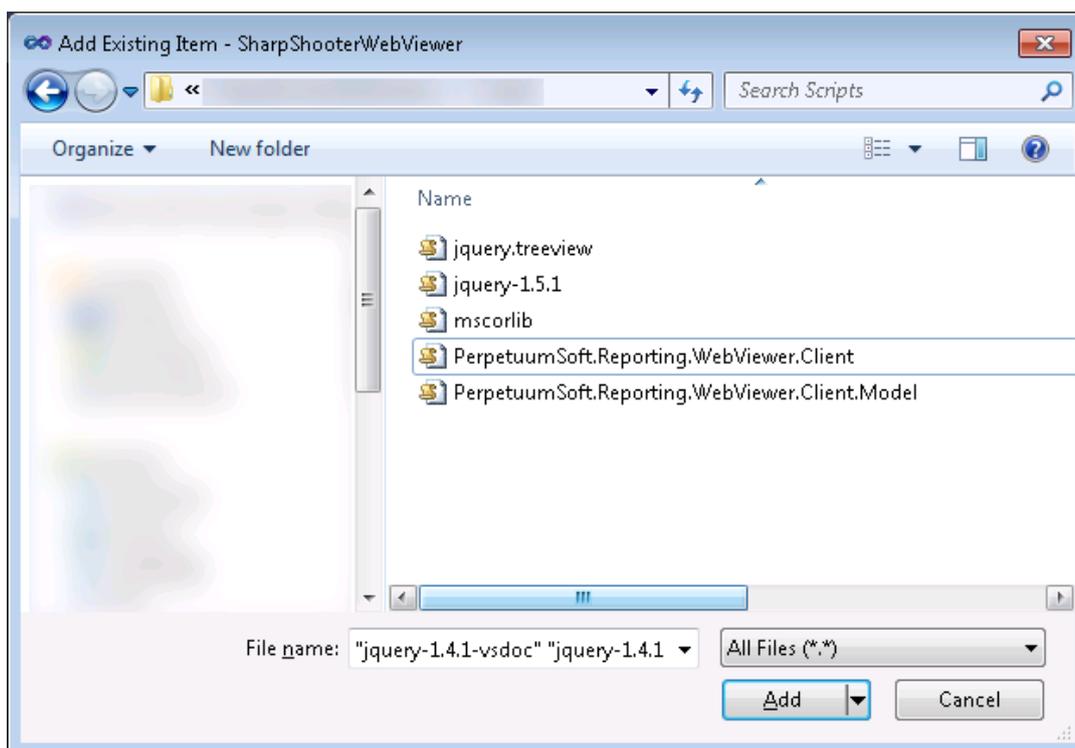
- PerpetuumSoft.Reporting.WebViewer.Client.js – レポートを受け取って表示するためのロジックを実装します
- PerpetuumSoft.Reporting.WebViewer.Client.Model.js – データモデルのクラス
- WebViewer.Style.js – WebViewer コンポーネントのスタイルやスキンを制御するメソッド

ソリューションエクスプローラから「Scripts」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...] を選択してください。



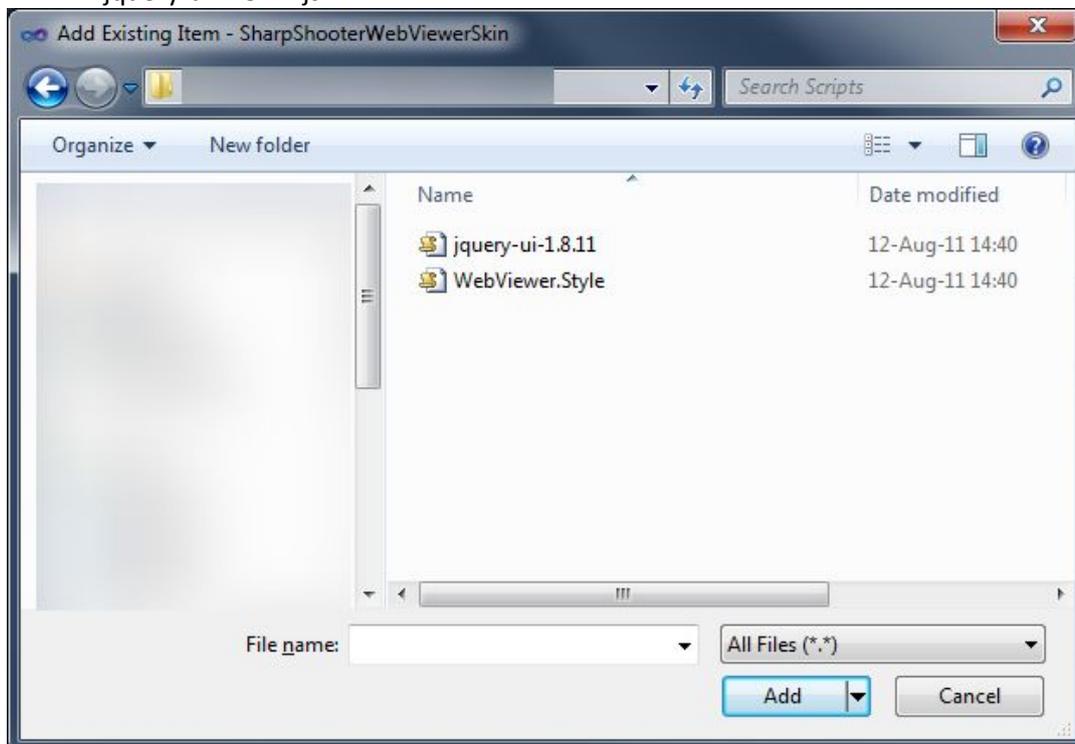
SharpShooter Reports.Web のインストールディレクトリの「Web\Scripts」フォルダから、次のファイルを追加します。

- jquery-1.5.1.js;
- jquery.treeview.js;
- mscorlib.js;
- PerpetuumSoft.Reporting.WebViewer.Client.Model.js;
- PerpetuumSoft.Reporting.WebViewer.Client.js.



SharpShooter Reports.Web のインストールディレクトリの「Web\Skins\Default\Scripts」フォルダから、次のファイルを追加します。

- WebViewer.Style.js;
- jquery-ui-1.8.11.js.

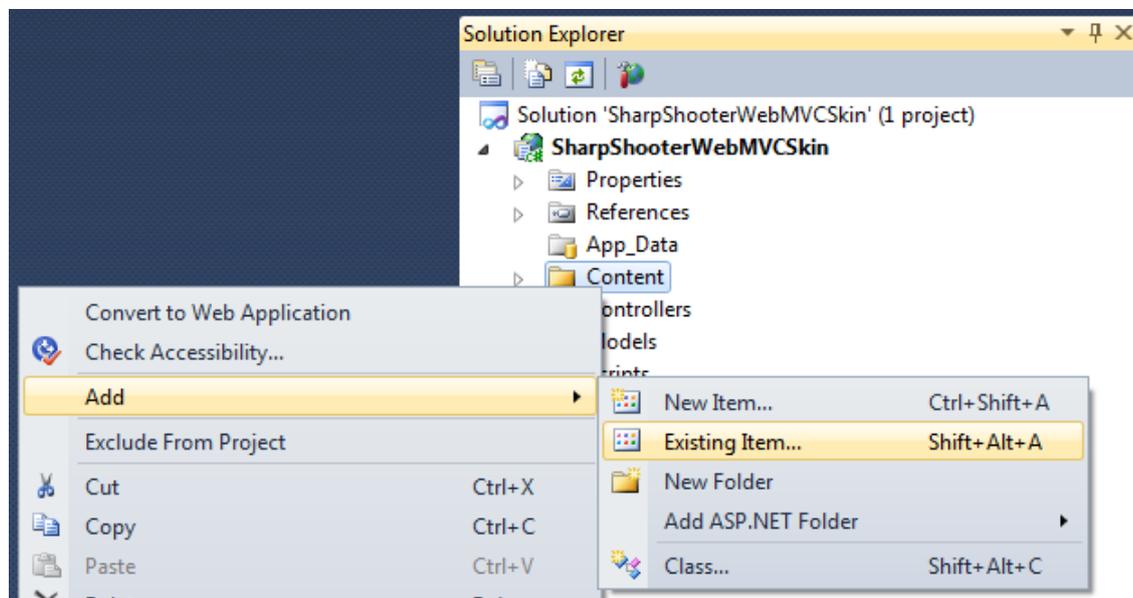


手順 7. スタイルの追加

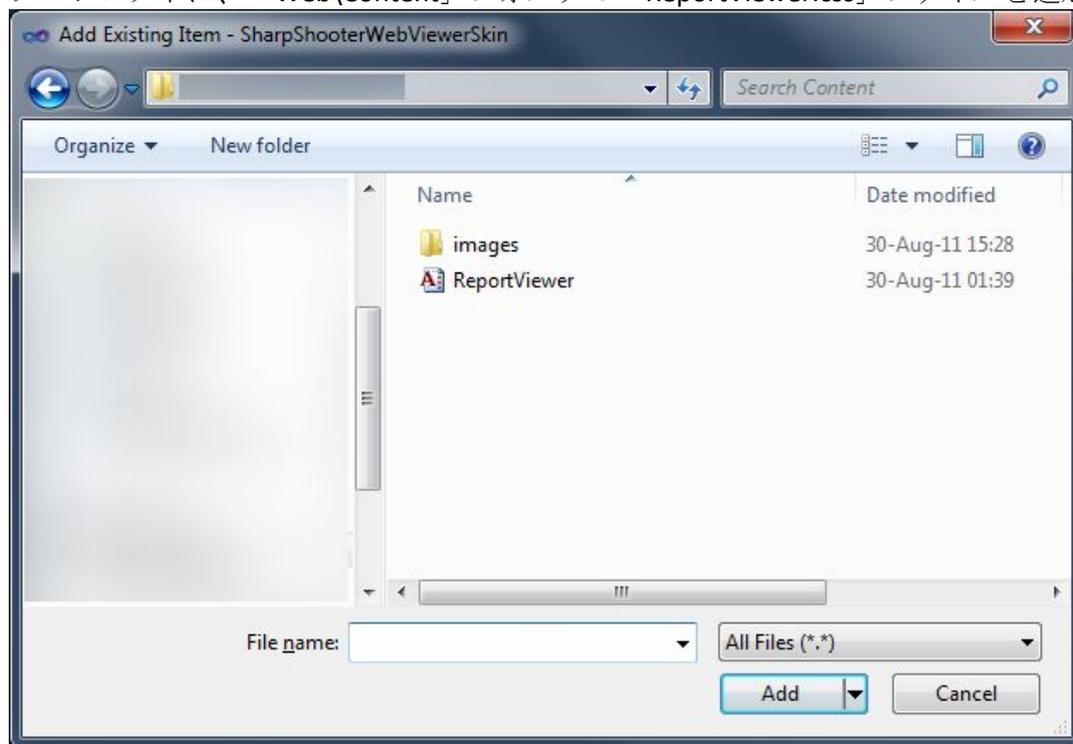
プロジェクトにスタイルを追加します。これらのスタイルは、レポートを正しく表示するために必要になるので、スタイルを持つ次のファイルをプロジェクトに追加する必要があります。

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...]を選択してください。



プロジェクトに、「Web\Content」フォルダの「ReportViewer.css」ファイルを追加します。

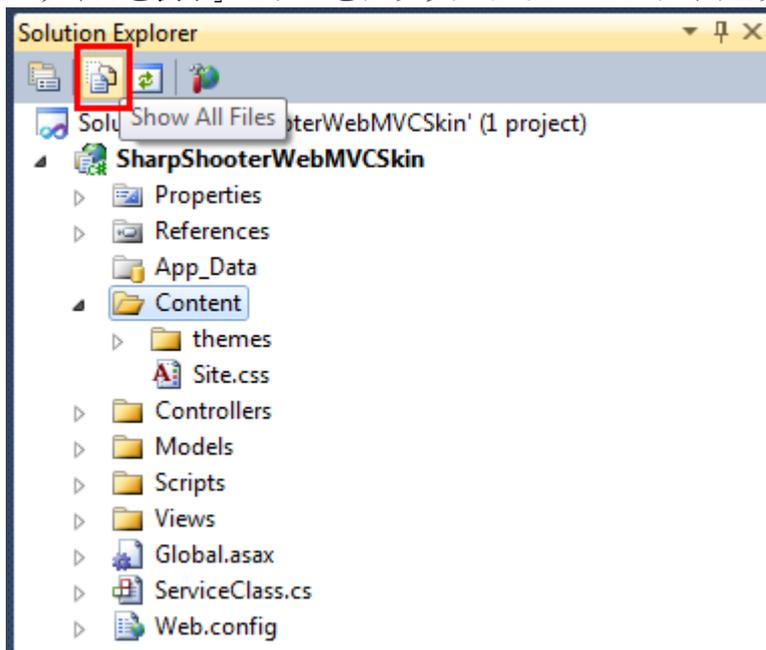


手順 8. デフォルトのスキンファイルを追加する

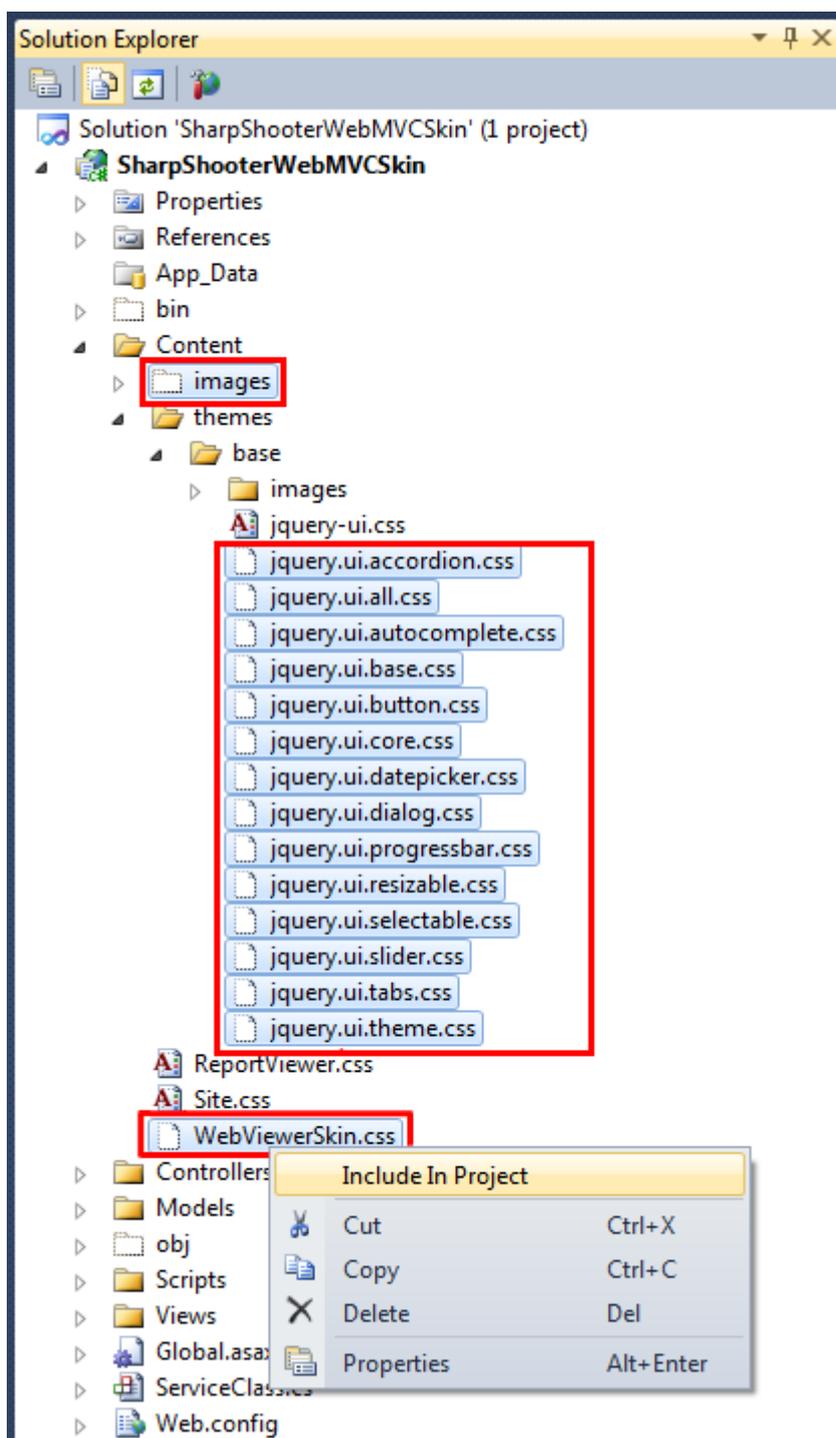
スタイルやイメージ付きのファイルをいくつか追加します。これらはコンポーネントのユーザーインターフェイスのセットアップに使用します。インストールディレクトリの

「Web\Skins\Default\Content\」フォルダの中身をプロジェクトの「Styles」フォルダにコピーします。

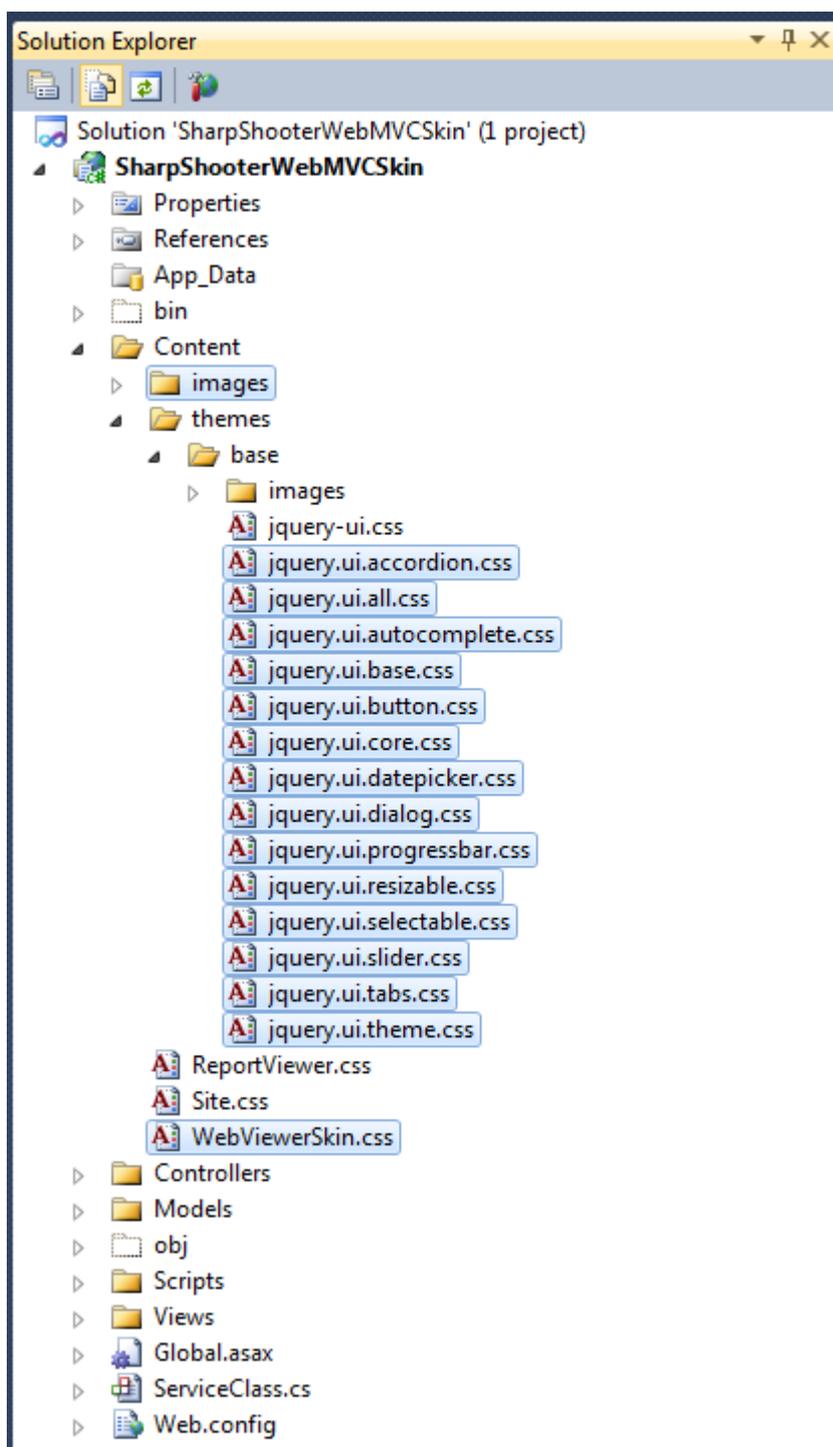
ソリューションエクスプローラの「SharpShooterWebViewSkin」プロジェクトを選択し、「すべてのファイルを表示」ボタンをクリックしてプロジェクトにある全ファイルを表示します。



「Content」フォルダのプロジェクトに追加されていないファイルをすべて選択し、コンテキストメニューから[プロジェクトに含める]をクリックします。



そうすると、プロジェクトにファイルや入れ子のファイルが追加されます。

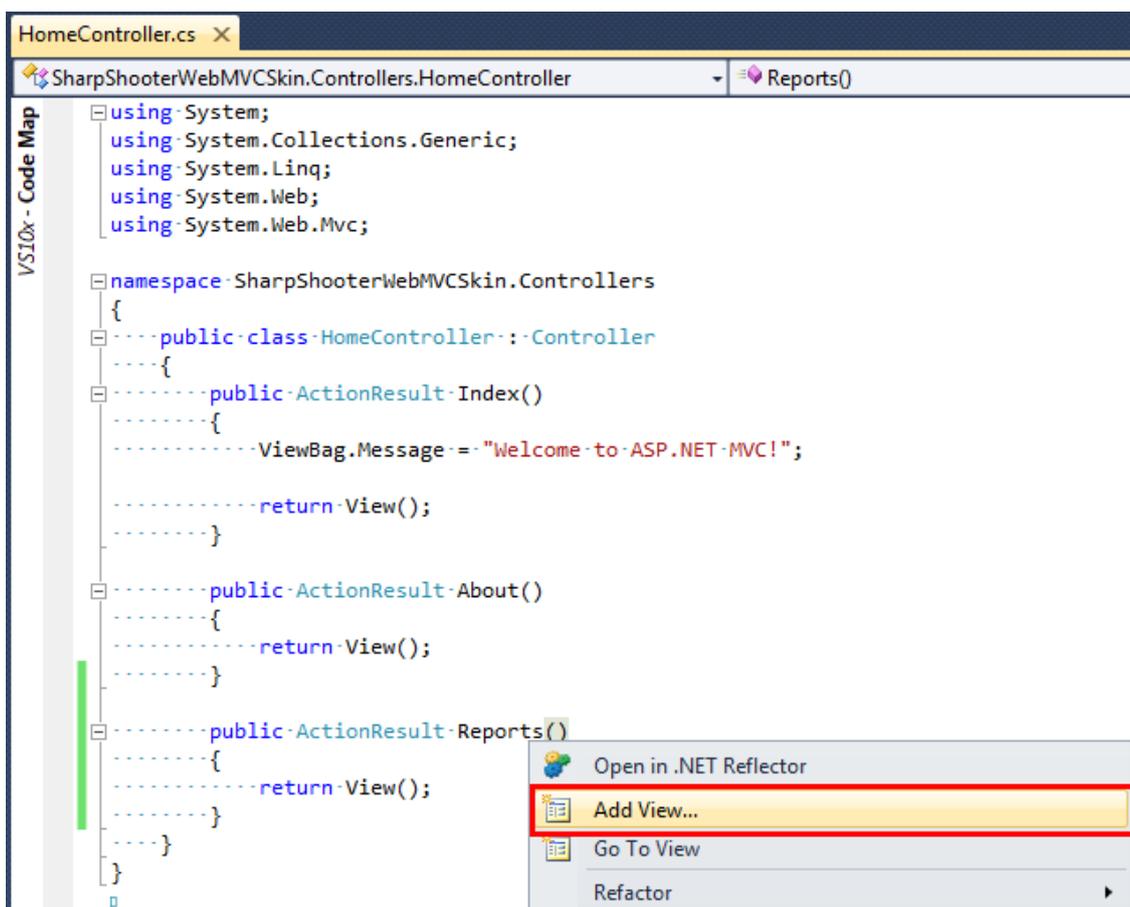


手順 9. プロジェクトに「ビュー」を追加する

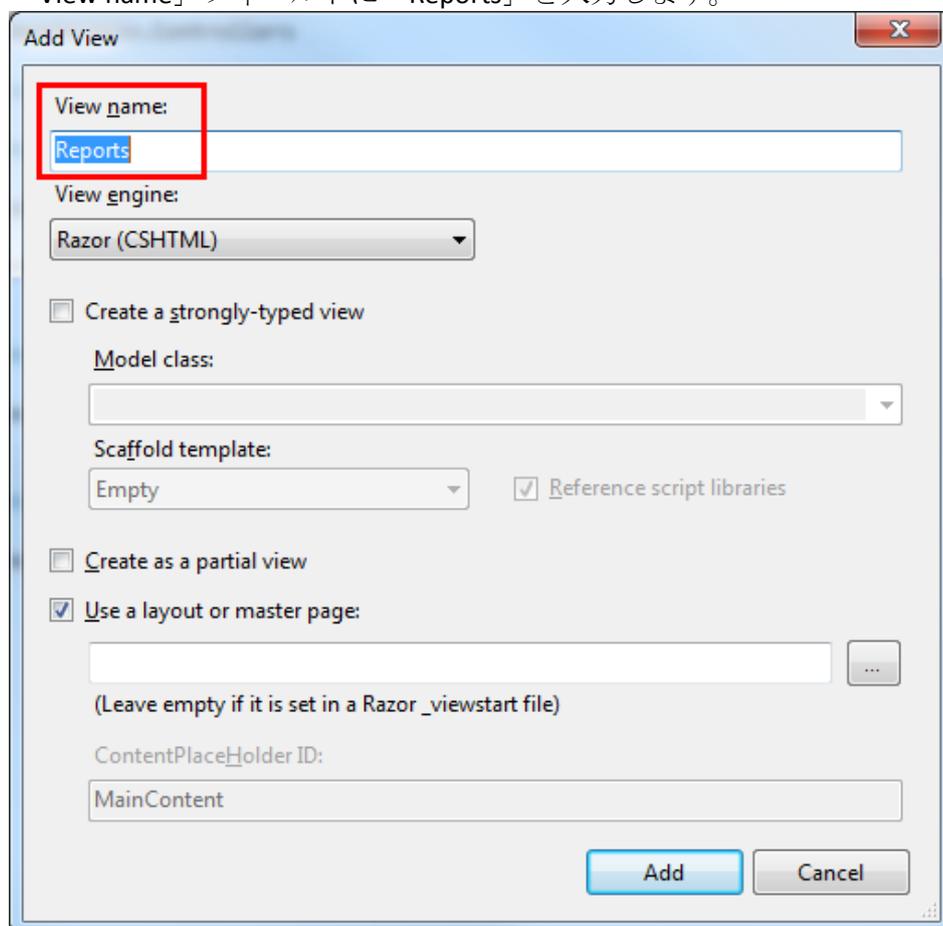
「HomeController」コントローラに「Reports」というアクションを追加します。

```
public ActionResult Reports()  
{  
    return View();  
}
```

ビューを追加します。「Reports」アクションを右クリックし、コンテキストメニューから [ビューの追加...] を選択します。

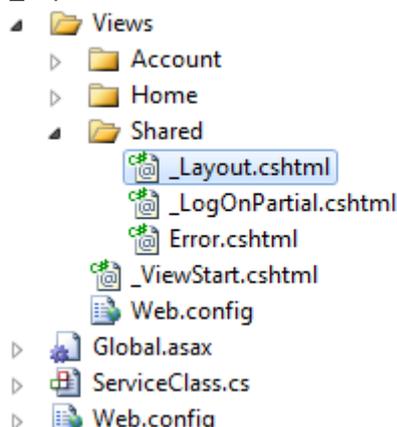


「View name」フィールドに「Reports」と入力します。



手順 10. ビュー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。



次の文字列を

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

以下のように変更します。

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

スタイルを参照するコードを追加します。

```

<link href="@Url.Content("~/Content/themes/base/jquery.ui.all.css")" rel="stylesheet" type="text/css" />
<link href="@Url.Content("~/Content/ReportViewer.css")" rel="stylesheet" type="text/css" />
<link href="@Url.Content("~/Content/WebViewSkin.css")" rel="stylesheet" type="text/css" />

```

「_Layout.cshtml」の page の menu に、「Reports.cshtml」ページのリンクを追加します。

```
<li>@Html.ActionLink("Reports", "Reports", "Home")</li>
```

「_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/themes/base/jquery.ui.all.css")" rel="stylesheet" type="text/css" />

```



```

<link href="@Url.Content("~/Content/ReportViewer.css")" rel="stylesheet" type="text/css" />
<link href="@Url.Content("~/Content/WebViewerSkin.css")" rel="stylesheet" type="text/css" />
<script src="@Url.Content("~/Scripts/jquery-1.5.1.js")" type="text/javascript"></script>
</head>
<body>
  <div class="page">
    <div id="header">
      <div id="title">
        <h1>My MVC Application</h1>
      </div>
      <div id="logindisplay">
        @Html.Partial("_LogOnPartial")
      </div>
      <div id="menucontainer">
        <ul id="menu">
          <li>@Html.ActionLink("Home", "Index", "Home")</li>
          <li>@Html.ActionLink("About", "About", "Home")</li>
          <li>@Html.ActionLink("Reports", "Reports", "Home")</li>
        </ul>
      </div>
    </div>
    <div id="main">
      @RenderBody()
      <div id="footer">
      </div>
    </div>
  </div>
</body>
</html>

```

ソリューションエクスプローラの「Reports.cshtml」ファイルをダブルクリックして「Reports.cshtml」ファイルを開き、次のコードを追加して、スクリプトを「Reports.cshtml」ファイルに関連付けます。

```

<script src="@Url.Content("~/Scripts/jquery-1.5.1.js")" type="text/javascript"></script>
<script src="@Url.Content("~/Scripts/jquery-ui-1.8.11.js")" type="text/javascript"></script>
<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("~/Scripts/WebViewer.Style.js")" type="text/javascript"></script>

```

手順 11. ビューに Report Viewer コンポーネントを追加する

次のコードスニペットを追加します。これは、コンポーネントのデフォルトのビューとユーザーインターフェイスを定義しています。

```

<div class="ssr_reportViewerControl">
  <!--Toolbar panel, contains print and exports button-->
  <div id="ssr_toolBarPanel" class="ssr_toolBar">
    <div class="ssr_toolButtonContainer">
      <div id="ssr_printButton" class="ssr_toolButton ssr_printButtonDisable" title="Print">
      </div>
    </div>
    <div class="ssr_toolButtonContainer">
      <div id="ssr_rtfExportButton" class="ssr_toolButton ssr_rtfButtonDisable" title="Export to Rtf">
      </div>
    </div>
    <div class="ssr_toolButtonContainer">
      <div id="ssr_pdfExportButton" class="ssr_toolButton ssr_pdfButtonDisable" title="Export to Pdf">
      </div>
    </div>
  </div>
</div>

```



```

    <div id="ssr_excelExportButton" class="ssr_toolButton ssr_excelButtonDisable" title="Export to
Excel">
    </div>
</div>
<div class="ssr_toolButtonContainer">
    <div id="ssr_htmlExportButton" class="ssr_toolButton ssr_htmlButtonDisable" title="Export to
Html">
    </div>
</div>
<div class="ssr_toolButtonContainer">
    <div id="ssr_xpsExportButton" class="ssr_toolButton ssr_xpsButtonDisable" title="Export to Xps">
    </div>
</div>
</div>
<div class="ssr_centralDiv">
    <!--Contains thumbnails and documentMap elements-->
    <div class="ssr_navigationArea" id="ssr_navigationAreaElement">
        <div class="ssr_navigationIconPanel" id="ssr_navigationIconPanel">
            <div class="ssr_thumbnailButtonContainer" id="ssr_thumbnailButtonContainer">
                <!--shows or hides thumbnail element-->
                <div title="Thumbnails" id="ssr_thumbnailButton" class="ssr_thumbnailButton
ssr_thumbnailButtonDisable">
                </div>
            </div>
            <div class="ssr_documentMapButtonContainer" id="ssr_documentMapButtonContainer">
                <!--shows or hides document map element-->
                <div title="Document Map" id="ssr_documentMapButton" class="ssr_documentMapButton
ssr_documentMapButtonDisable">
                </div>
            </div>
        </div>
        <div id="ssr_thumbnailsPanel" class="ssr_thumbnailsPanel">
            <div id="ssr_thumbnailHeadPanel" class="ssr_thumbnailHeadPanel">
                <span class="ssr_thumbnailHeadPanelTitle">Thumbnails</span>
                <div title="Close" id="ssr_closeThumbnailPanelButton"
class="ssr_closeThumbnailPanelButton">
                </div>
            </div>
            <div id="ssr_wrapThumbnailContentPanel" class="ssr_wrapThumbnailContentPanel">
                <!--element which contains a list of thumbnail-->
                <div id="ssr_thumbnailContentPanel" class="ssr_thumbnailContentPanel">
                </div>
            </div>
        </div>
        <div id="ssr_documentMapPanel" class="ssr_documentMapPanel">
            <div id="ssr_documentMapHeadPanel" class="ssr_documentMapHeadPanel">
                <span class="ssr_documentMapHeadPanelTitle">Document Map</span>
                <div title="Close" id="ssr_closeDocumentMapPanelButton"
class="ssr_closeThumbnailPanelButton">
                </div>
            </div>
            <div id="ssr_wrapDocumentMapContentPanel" class="ssr_wrapDocumentMapContentPanel">
                <!--element which contains a document map-->
                <div id="ssr_documentMapContentPanel" class="ssr_documentMapContentPanel">
                </div>
            </div>
        </div>
    </div>
    <!--reports will be shows in this element-->
    <div id="ssr_reportContent" class="ssr_reportArea">
    </div>
</div>
<!--Statusbar panel-->
<div id="ssr_statusBarPanel" class="ssr_statusPanel">
    <div id="ssr_paginationPanel" class="ssr_paginationPanel">
        <div class="ssr_statusButtonContainer ssr_statusButtonContainerBack">

```



```

        <div id="ssr_firstPageButton" class="ssr_statusButton ssr_statusButtonBack
ssr_firstPageDisable">
            </div>
        </div>
        <div class="ssr_statusButtonContainer ssr_statusButtonContainerBack">
            <div id="ssr_previousPageButton" class="ssr_statusButton ssr_statusButtonBack
ssr_previousPageDisable">
                </div>
            </div>
            <div class="ssr_statusCurrentPageContainer">
                <!--displays the current page number-->
                <input id="ssr_currentPageNumberBox" class="ssr_currentPageNumber" title="Current page" />
                <span class="ssr_pageCountOf">of</span><span> </span><span id="ssr_pageCountBox"
                    class="ssr_pageCount">0</span>
            </div>
            <div class="ssr_statusButtonContainer ssr_statusButtonContainerForward">
                <div id="ssr_nextPageButton" class="ssr_statusButton ssr_statusButtonForward
ssr_nextPageDisable">
                    </div>
                </div>
            <div class="ssr_statusButtonContainer ssr_statusButtonContainerForward">
                <div id="ssr_lastPageButton" class="ssr_statusButton ssr_statusButtonForward
ssr_lastPageDisable">
                    </div>
                </div>
            </div>
            <!--Shows or hides error dialog-->
            <div id="ssr_notificationPanel" class="ssr_notificationPanel">
                <div id="ssr_errorButton" class="ssr_errorButton ssr_errorButtonDisable">
                    </div>
            </div>
        </div>
    </div>
    <div id="ssr_errorDialog" title="Error information" style="display: none;">
        <div id="ssr_errorMessage" class="ssr_errorMessageContent">
            </div>
        </div>
    </div>

```

ページに次の javascript コードを追加します。これは、ドキュメントの読込を制御します。

```

<script type="text/javascript">
    var reportViewer = null;
    $(document).ready(function () {
        Initialize();
        InitializeReportViewer();
    });
    function InitializeReportViewer() {
        reportViewer = new PerpetuumSoft.Reporting.WebViewer.Client.ReportViewer('#ssr_reportContent');
        reportViewer.setServiceUrl("http://localhost:5555/ReportService");
        reportViewer.reportName = "CustomersReport";
        reportViewer.setDocumentMapControl("#ssr_documentMapContentPanel");
        reportViewer.setThumbnailsControl("#ssr_thumbnailContentPanel");
        reportViewer.documentInfoLoadedEvent = function (pages) {
            EnableToolbarState();
            if (pages.length > 1) {
                pageCount = pages.length;
                EnableNextLastPage();
                DisablePreviousFirstPage();
            }
            else {
                DisableNextLastPage();
                DisablePreviousFirstPage();
            }
        };
        $("#ssr_pageCountBox").text(pages.length.toString());
        reportViewer.loadThumbnails();
    };

```



```
reportViewer.errorEvent = function (errorModel) {
    switch (errorModel.errorType) {
        case PerpetuumSoft.Reporting.WebViewer.Client.ErrorType.communicationError:
            $("#ssr_errorMessage").text(errorModel.error._error$1);
            break;
        case PerpetuumSoft.Reporting.WebViewer.Client.ErrorType.clientError:
            $("#ssr_errorMessage").text("clientError" + (errorModel.error).message);
            break;
        case PerpetuumSoft.Reporting.WebViewer.Client.ErrorType.serverError:
            $("#ssr_errorMessage").text("serverError" + (errorModel.error).message +
(errorModel.error).getInformation());
            break;
        default:
            $("#ssr_errorMessage").text(errorModel.error.message);
            break;
    }
    EnableErrorButton();
    ShowErrorDialog();
};
reportViewer.currentPageChangedEvent = function (pageNumber) {
    if (reportViewer.isFirstPage()) {
        DisablePreviousFirstPage();
    }
    else {
        EnablePreviousFirstPage();
    }
    if (reportViewer.isLastPage()) {
        DisableNextLastPage();
    }
    else {
        EnableNextLastPage();
    }
    $("#ssr_currentPageNumberBox").val(pageNumber.toString());
};

reportViewer.renderDocument();
}
</script>
```

コンポーネントがページに追加されたので、テストしてみます。

手順 12. アプリケーションの実行

Visual Studio で F5 をクリックしてアプリケーションを実行し、「Reports」リンクをクリックすると、サンプルのレポートページが表示されます。

