Using WebTextEditor in Microsoft Office SharePoint 2007

This white paper describes the techniques and walkthrough on how to use WebTextEditor Microsoft Office SharePoint Server 2007 as a WebPart.

Prerequisites

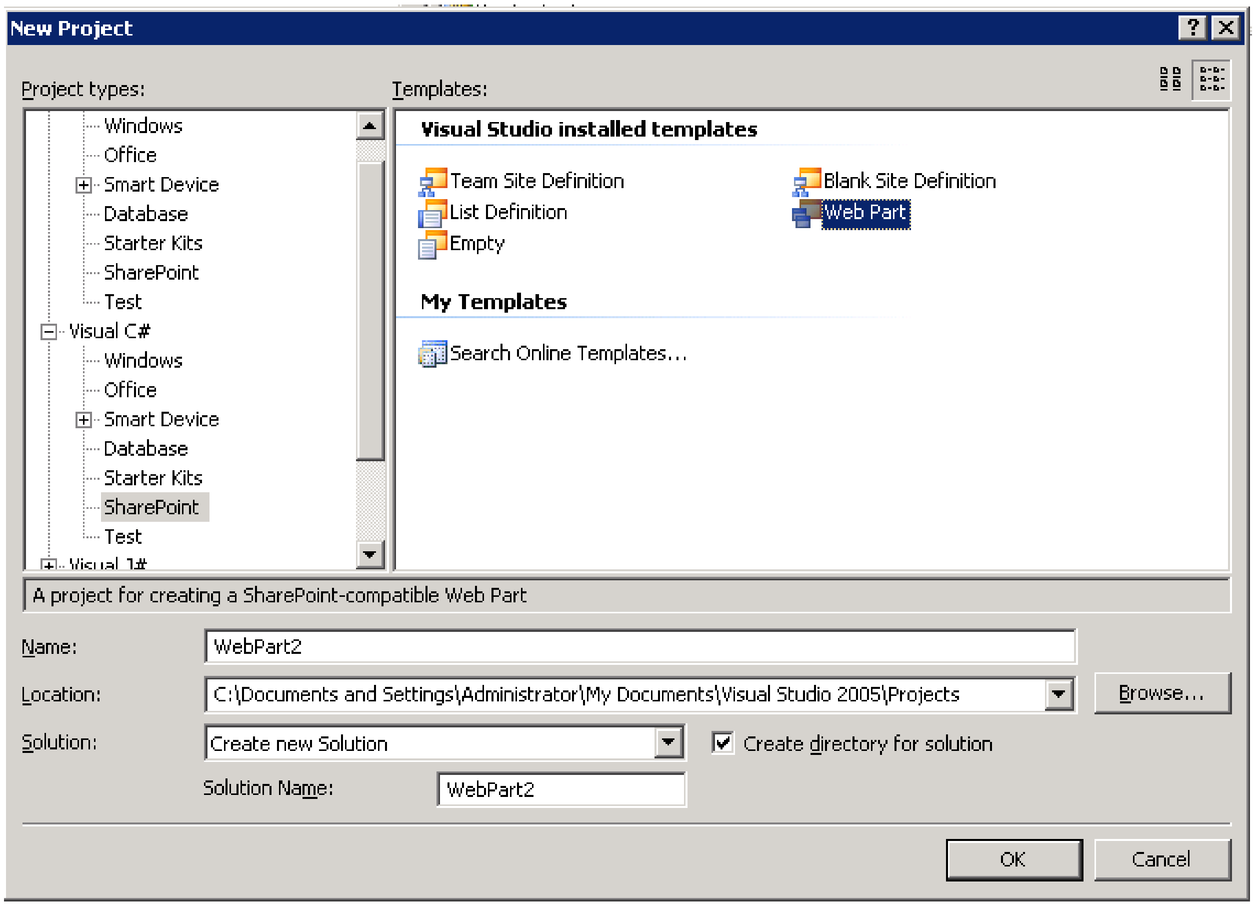
The following are the prerequisites development environments before proceeding further.

- Windows 2003 Server with IIS installed.
- Microsoft Office SharePoint Server 2007.
- Intersoft WebUI Studio 2009 R2.

The virtual PC image of Microsoft Office SharePoint Server 2007 can be obtained here. Click here to download the 30-day fully functional WebUI Studio 2009 R2.

Creating a WebPart

1. Create a blank Share Point WebPart. Simply select New Project > C# > SharePoint > WebPart.



A project for creating a SharePoint-compatible Web Part

Name: WebPart2

Location: C:\Documents and Settings\Administrator\My Documents\Visual Studio 2005\Projects

Solution: Create new Solution

[✓] Create directory for solution

Solution Name: WebPart2

2. Enter a name for your WebPart. This tutorial is using **WebPartWebTextEditor** as the name.

```
using System;
using System.Runtime.InteropServices;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Web.UI.WebControls.WebParts;
using System.Xml.Serialization;

using Microsoft.SharePoint;
using Microsoft.SharePoint.WebControls;
using Microsoft.SharePoint.WebPartPages;

namespace WebPartWebTextEditor
{
    [Guid("d87b47e6-53aa-499a-90f7-9c715f3fd3ef")]
    public class WebPart1 : System.Web.UI.WebControls.WebParts.WebPart
    {
        public WebPart1()
        {
        }
        protected override void CreateChildControls()
        {
            base.CreateChildControls();

            // TODO: add custom rendering code here.
            // Label label = new Label();
            // label.Text = "Hello World";
            // this.Controls.Add(label);
        }
    }
}
```

3. Next it to add reference to all required assemblies. Right click on the References in Solution Explorer and navigate to:
    - %Installation Path\Intersoft Solutions\WebUI Studio for ASP.NET\ WebUI.NET Framework 3.0\Bin. Add **ISNet.dll**, **ISNet.WebUI.dll**, and **ISNet.WebUI.Resources.dll**.
    - %Installation Path\Intersoft Solutions\WebUI Studio for ASP.NET\ WebTextEditor.NET 1.1\Bin. Add **ISNet.WebUI.WebTextEditor.dll** and **ISNet.WebUI.WebTextEditor.Resources.dll**.

4. Create a new class inheriting **EditorPart** class from SharePoint.
   Here's the code:

```
using System;
using System.Collections.Generic;
using System.Text;
using System.Web.UI.WebControls.WebParts;
using ISNet.WebUI.WebTextEditor;
using System.Web.UI.WebControls;
using System.Drawing;

namespace WebPartWebTextEditor
{
    class HtmlEditor : EditorPart
    {
        private WebTextEditor htmlContentText;
        public HtmlEditor()
        {
            this.ID = "HtmlEditor";
        }

        protected override void CreateChildControls()
        {
            htmlContentText = new WebTextEditor();
            htmlContentText.Width = Unit.Pixel(500);
            htmlContentText.Height = Unit.Pixel(700);
            htmlContentText.ImagesDirectory = "http://hpv-win03-
1:82/CommonLibrary/Images/WebTextEditor/";
            htmlContentText.ScriptDirectory = "http://hpv-win03-
1:82/CommonLibrary/WebTextEditor/V1_0_1000/";
            htmlContentText.SharedScriptDirectory = "http://hpv-win03-
1:82/CommonLibrary/Shared/";
            htmlContentText.SharedScriptDirectory = "http://hpv-win03-
1:82/CommonLibrary/Shared/";
            htmlContentText.WebDesktopScriptDirectory = "http://hpv-
win03-1:82/CommonLibrary/WebDesktop/V3_0_7200/";
            this.Controls.Add(htmlContentText);
        }

        public override bool ApplyChanges()
        {
            EnsureChildControls();

            WebPartWebTextEditor part = WebPartToEdit as
WebPartWebTextEditor;
            if (part != null)
                part.DisplayText = htmlContentText.Content;
            else
                return false;

            return true;
        }

        public override void SyncChanges()
        {
            EnsureChildControls();

            WebPartWebTextEditor part = WebPartToEdit as
WebPartWebTextEditor;
            if (part != null)
                htmlContentText.Content = part.DisplayText;
        }
    }
}
```

The CreateChildControls contains the definition of a new WebTextEditor object with few properties set. If you are not using SmartWebResources, you need to manually map the path for the script and resources*. **If you are using SmartWebResources, please follow the step 2 of "Enabling WebTextEditor control in SharePoint Site" below.***

5. Back to the main class (WebPartWebTextEditor) and edit it.
   Here is the code:

```
using System;
using System.Runtime.InteropServices;
using System.Web.UI;
using System.Web.UI.WebControls.WebParts;
using System.Xml.Serialization;

using System.Collections.Generic;
using Microsoft.SharePoint;
using Microsoft.SharePoint.WebControls;
using Microsoft.SharePoint.WebPartPages;

namespace WebPartWebTextEditor
```

```
{
    [Guid("8fa6b1cc-e58c-44ab-b4c1-53cef13340c6")]
    public class WebPartWebTextEditor :
System.Web.UI.WebControls.WebParts.WebPart, IWebEditable
    {
        public WebPartWebTextEditor()
        {
            this.ExportMode = WebPartExportMode.All;
        }

        private string displayText = "Hello World!";
        [WebBrowsable(true), Personalizable(true)]
        public string DisplayText
        {
            get { return displayText; }
            set { displayText = value; }
        }

        protected override void Render(System.Web.UI.HtmlTextWriter
writer)
        {
            writer.Write(displayText);
        }

        #region IWebEditable Members
        EditorPartCollection IWebEditable.CreateEditorParts()
        {
            List<HtmlEditor> editors = new List<HtmlEditor>();
            editors.Add(new HtmlEditor());
            return new EditorPartCollection(editors);
        }
        object IWebEditable.WebBrowsableObject
        {
            get { return this; }
        }
        #endregion
    }
}
```

The ***IWebEditable*** acts as an interface for specifying custom editing control which is later associated to a WebPart control. By declaring the HTMLEditor class as the editor class in the WebPart, user will be able to see and select HTMLEditor in edit mode. The ***DisplayText*** property holds the value for WebPart and must be placed inside the **Render** method.

6. Save and Build the project.

Enabling WebTextEditor control in SharePoint Site

This walkthrough will guide you on how to configure the newly created site in order to integrate WebTextEditor into it.

Three important steps that need to be done:

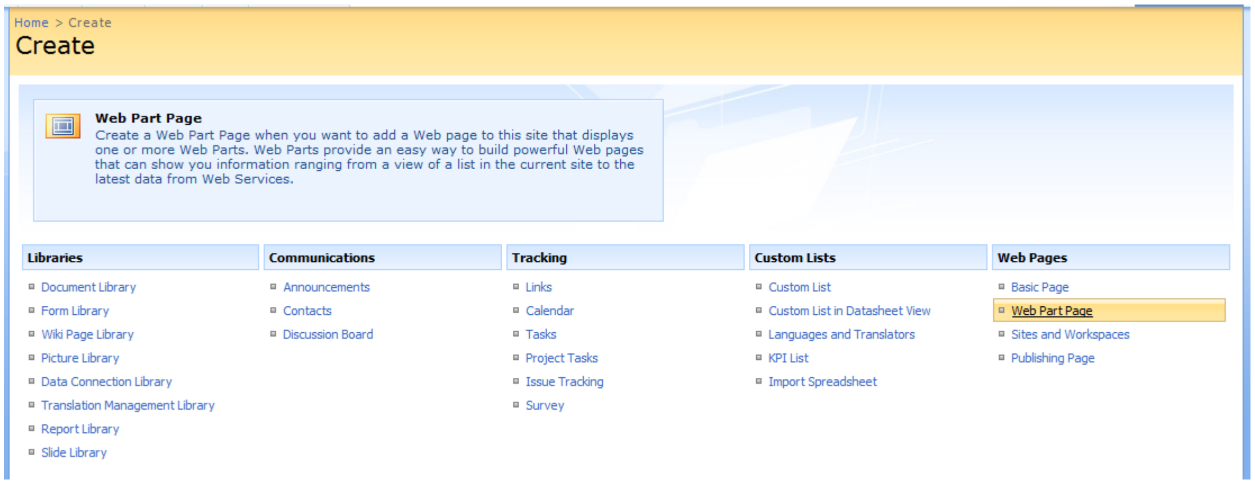
1. Mark your WebPart as Safe Control
    - Launch Windows Explorer and navigate to the folder of your site. E.g, C:\Inetpub\wwwroot\wss\VirtualDirectories\80. **Note:** the number indicates the port number of your designated site.
    - Open the Web.config.
    - Paste the following code before </SafeControls>
      ```
      <SafeControl Assembly="WebPartWebTextEditor, Version=1.0.0.0,
      Culture=neutral, PublicKeyToken=9f4da00116c38ec5"
      Namespace="WebPartWebTextEditor" TypeName="*" Safe="True" />
      ```
      WebPartWebTextEditor is the namespace of the new created dll from the WebPart.
    - Add WebTextEditor RuntimeLicenseKey under the <appSettings>
      ```
      <add key="ISNet.WebUI.WebTextEditor.v1_0_1000.RunTimeLicenseKey"
      value="YOUR-RNTM-LCKY" />
      ```

2. Register SmartWebResources (Optional)
    - Open our Web.config
    - Past this code under the <httpHandlers>
      ```
      <add path="ISRes.axd" verb="GET"
      type="ISNet.WebUI.WebResourceHandler, ISNet.WebUI,
      Version=3.0.5000.1, Culture=neutral,
      PublicKeyToken=b1f2a8511635667a" validate="true"/>
      ```
    - And under the <appSettings>
      ```
      <add key="ISNet.WebUI.ISRes_Registered" value="true"/>
      ```
    - Save your Web.config.

3. Copy WebTextEditor Resources
    - Launch Windows Explorer and go to the bin folder of your site. Eg, C:\inetpub\wwwroot\wss\VirtualDirectories\[YourAppPortNumber]\bin.
    - Copy the following assemblies to the bin folder of your site:
        - ✓ **ISNet.WebUI.WebDesktop.Resources.dll**,
        - ✓ **ISNet.WebUI.WebTextEditor.dll, and,**
        - ✓ **ISNet.WebUI.WebTextEditor.Resources.dll**.
    - Drag and drop your WebPart assembly (for example: WebPartWebTextEditor.dll) to GAC (C:\Windows\assembly).

Create a WebPart page

1. Go to your Microsoft SharePoint page. E.g, http://hpv-win03-1/Pages/Default.aspx
2. Go to View All Site Content > Create.
3. In Web Pages, select WebPart Page



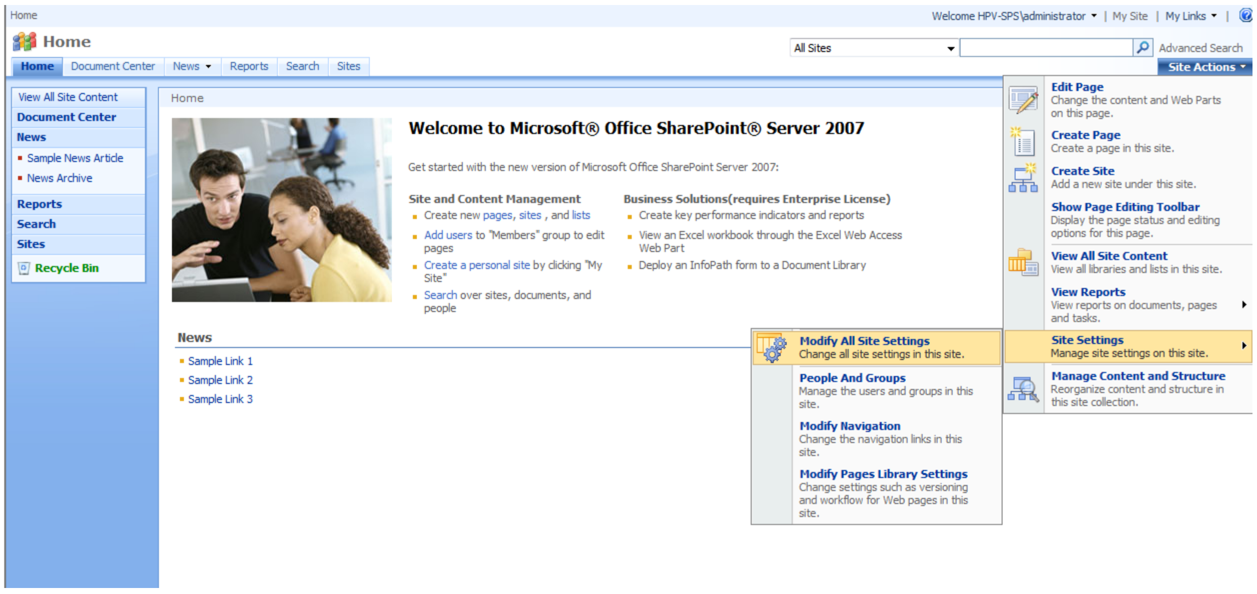
Home > Create

Create

Web Part Page
Create a Web Part Page when you want to add a Web page to this site that displays one or more Web Parts. Web Parts provide an easy way to build powerful Web pages that can show you information ranging from a view of a list in the current site to the latest data from Web Services.

Libraries	Communications	Tracking	Custom Lists	Web Pages
Document Library	Announcements	Links	Custom List	Basic Page
Form Library	Contacts	Calendar	Custom List in Datasheet View	Web Part Page
Wiki Page Library	Discussion Board	Tasks	Languages and Translators	Sites and Workspaces
Picture Library		Project Tasks	KPI List	Publishing Page
Data Connection Library		Issue Tracking	Import Spreadsheet	
Translation Management Library		Survey		
Report Library				
Slide Library				

4. Enter your new WebPart name.
5. Populate it to the SharePoint WebPart Gallery. Go to Site Action > Site Settings > Modify All Settings.



Home | Welcome HPV-SPS\administrator | My Site | My Links

Home

Home | Document Center | News | Reports | Search | Sites

All Sites | Advanced Search | Site Actions

View All Site Content
Document Center
News
- Sample News Article
- News Archive

Reports
Search
Sites
Recycle Bin

Home

Welcome to Microsoft® Office SharePoint® Server 2007

Get started with the new version of Microsoft Office SharePoint Server 2007:

Site and Content Management
- Create new pages, sites , and lists
- Add users to "Members" group to edit pages
- Create a personal site by clicking "My Site"
- Search over sites, documents, and people

Business Solutions(requires Enterprise License)
- Create key performance indicators and reports
- View an Excel workbook through the Excel Web Access Web Part
- Deploy an InfoPath form to a Document Library

News
- Sample Link 1
- Sample Link 2
- Sample Link 3

Edit Page
Change the content and Web Parts on this page.

Create Page
Create a page in this site.

Create Site
Add a new site under this site.

Show Page Editing Toolbar
Display the page status and editing options for this page.

View All Site Content
View all libraries and lists in this site.

View Reports
View reports on documents, pages and tasks.

Site Settings
Manage site settings on this site.

Manage Content and Structure
Reorganize content and structure in this site collection.

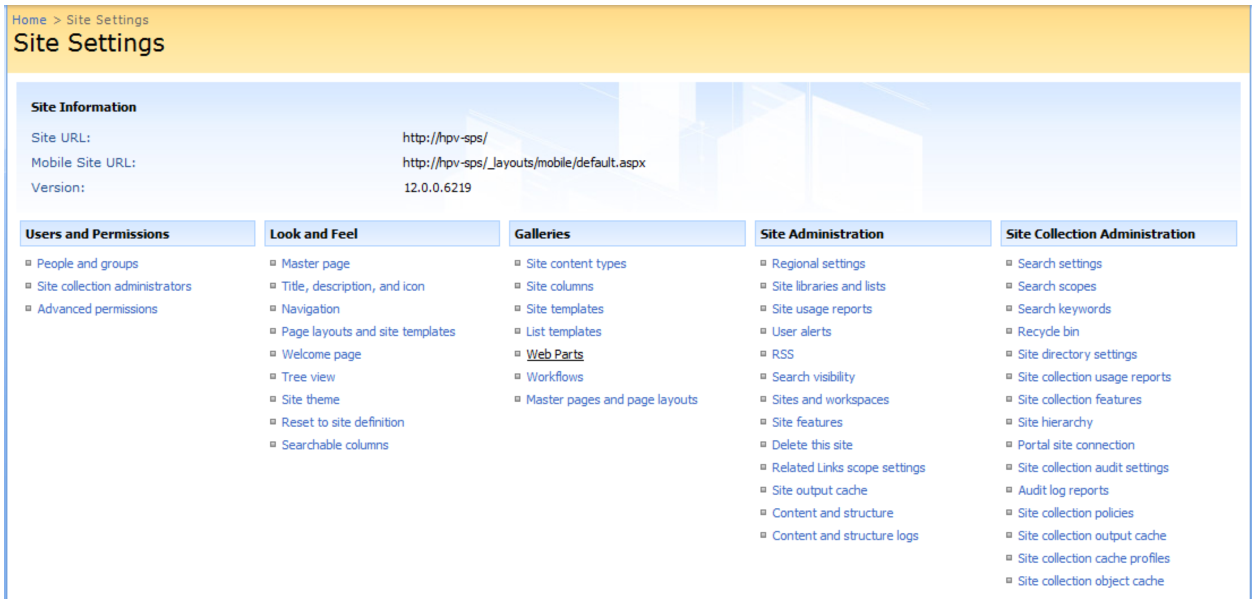
Modify All Site Settings
Change all site settings in this site.

People And Groups
Manage the users and groups in this site.

Modify Navigation
Change the navigation links in this site.

Modify Pages Library Settings
Change settings such as versioning and workflow for Web pages in this site.

Inside the Site Settings, go to Gallery column and choose WebPart



Home > Site Settings

Site Settings

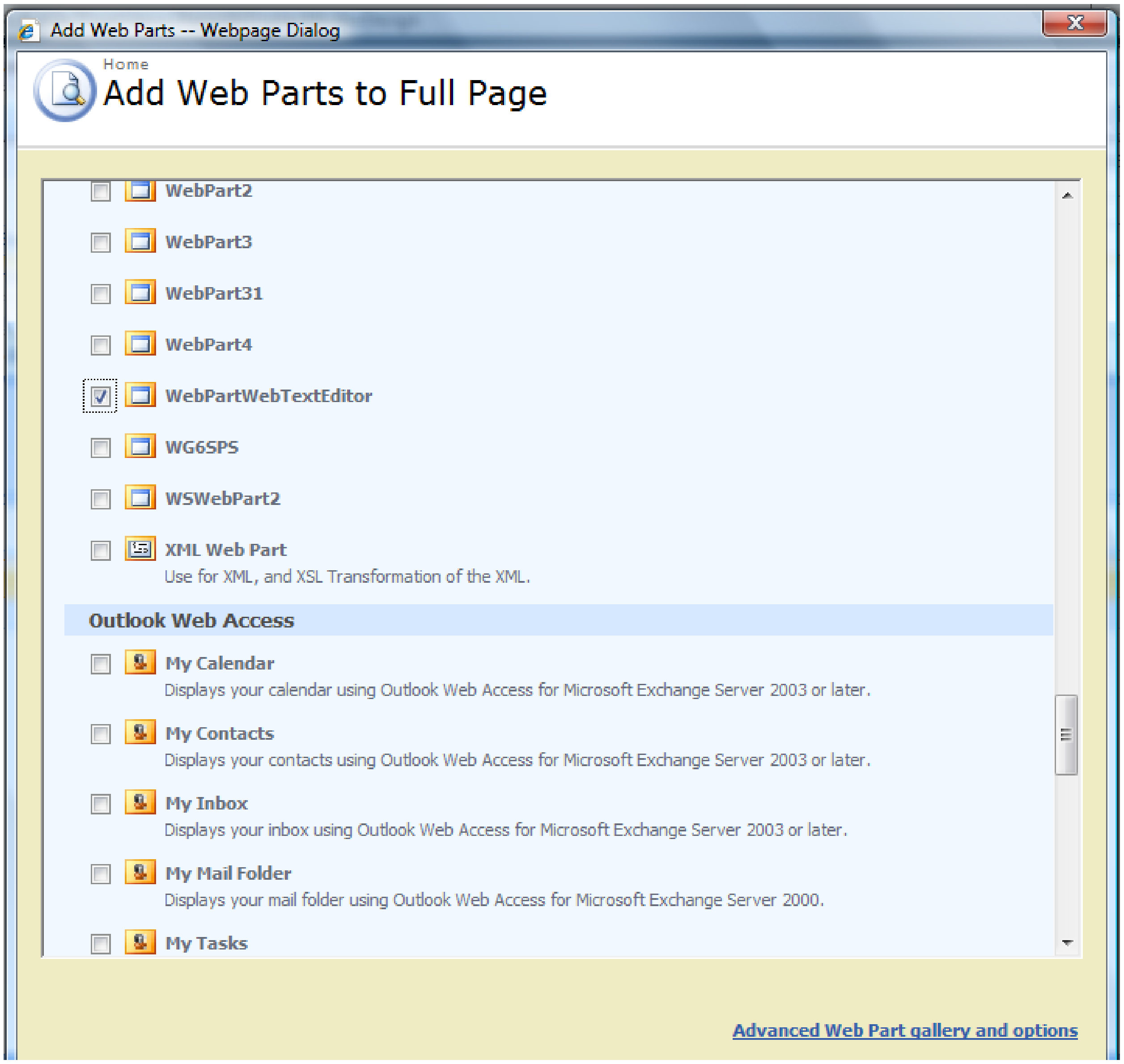
Site Information

Site URL:	http://hpv-sps/
Mobile Site URL:	http://hpv-sps/_layouts/mobile/default.aspx
Version:	12.0.0.6219

Users and Permissions	Look and Feel	Galleries	Site Administration	Site Collection Administration
People and groups	Master page	Site content types	Regional settings	Search settings
Site collection administrators	Title, description, and icon	Site columns	Site libraries and lists	Search scopes
Advanced permissions	Navigation	Site templates	Site usage reports	Search keywords
	Page layouts and site templates	List templates	User alerts	Recycle bin
	Welcome page	Web Parts	RSS	Site directory settings
	Tree view	Workflows	Search visibility	Site collection usage reports
	Site theme	Master pages and page layouts	Sites and workspaces	Site collection features
	Reset to site definition		Site features	Site hierarchy
	Searchable columns		Delete this site	Portal site connection
			Related Links scope settings	Site collection audit settings
			Site output cache	Audit log reports
			Content and structure	Site collection policies
			Content and structure logs	Site collection output cache
				Site collection cache profiles
				Site collection object cache

In WebPart Gallery, click New > Checked the WebPart name (WebPartWebTextEditor.WebPartWebTextEditor) > Populate Gallery.

6. Now, you can add your new WebPart to the SharePoint page.



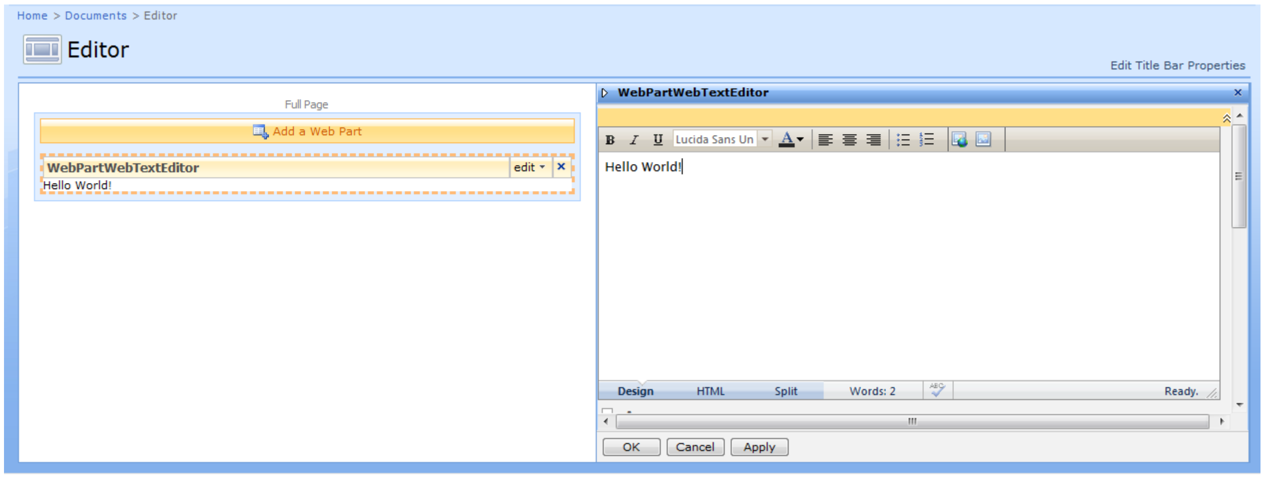
Add Web Parts to Full Page

- [ ] WebPart2
- [ ] WebPart3
- [ ] WebPart31
- [ ] WebPart4
- [✓] WebPartWebTextEditor
- [ ] WG6SPS
- [ ] WSWebPart2
- [ ] XML Web Part
  Use for XML, and XSL Transformation of the XML.

Outlook Web Access

- [ ] My Calendar
  Displays your calendar using Outlook Web Access for Microsoft Exchange Server 2003 or later.
- [ ] My Contacts
  Displays your contacts using Outlook Web Access for Microsoft Exchange Server 2003 or later.
- [ ] My Inbox
  Displays your inbox using Outlook Web Access for Microsoft Exchange Server 2003 or later.
- [ ] My Mail Folder
  Displays your mail folder using Outlook Web Access for Microsoft Exchange Server 2000.
- [ ] My Tasks

7. Try to modify the WebPart Page through the **Modify Shared WebPart** option, you'll see a WebTextEditor instance.


Home > Documents > Editor
Editor
Edit Title Bar Properties
Full Page
Add a Web Part
WebPartWebTextEditor
edit
Hello World!
WebPartWebTextEditor
Lucida Sans Un
Hello World!
Design
HTML
Split
Words: 2
Ready.
OK
Cancel
Apply