

```
    }
}

Asset aReturn = Asset.Load(szFolderPath);

return aReturn.IsLoaded ? aReturn : null;
}

public static string GetAllRelatedAssetIds(string szAssetId, ref string szAssetIdString, string szSitePath)
{
    string szRet = "";

    if (!string.IsNullOrEmpty(szAssetId) && !string.Equals(szAssetId, "0", StringComparison.OrdinalIgnoreCase) && !int.Equals(szAssetId, 0))
    {
        if (!szAssetId.Contains("|" + szAssetId + "|"))
        {
            szAssetIdString = string.IsNullOrEmpty(szAssetIdString) ? "|" + szAssetId + "|" : szAssetIdString + szAssetId + "|";

            FilterParams fpFilter = new FilterParams();
            fpFilter.Add(AssetPropertyNames.TemplateLabel, Comparison.Equals, "Relationship Config");
            fpFilter.Add("filter_string", Comparison.Equals, "|" + szAssetId + "|");

            List<Asset> lpRelationshipConfig = Asset.Load(szSitePath + "/Relationships Config/").GetFilterList(fpFilter);
            foreach (Asset aData in lpRelationshipConfig)
            {
                szRet = string.Equals(szAssetId, aData["source_id"], StringComparison.OrdinalIgnoreCase) ? aData["destination_id"] : aData["source_id"];
            }
        }
    }

    return szRet;
}

public static IEnumerable<LocaleId> CreateLocaleConfigCache(string szSitePath)
{
    return Asset.Load(szSitePath + "/Locales Config/").GetFileList().Select(aData => new LocaleId(aData));
}

public static string GetLocaleId(Asset asset, string szSitePath)
{
    string szRet = "";

    List<Asset> laLocalConfig = Asset.Load(szSitePath + "/Locales Config/").GetFileList();
    foreach (Asset aData in laLocalConfig)
    {
        string szAssetPath = asset.AssetPath.ToString().ToUpper();
        if (asset.Type.Equals(AssetType.Folder))
            szAssetPath = asset.AssetPath.ToString().ToUpper() + (string.Equals(szAssetPath.Last(), "/") ? "" : "/");

        if (szAssetPath.Contains(aData["folder_root"].ToUpper() + (string.Equals(aData["folder_root"].Last(), "/") ? "" : "/")))
    }
}
```



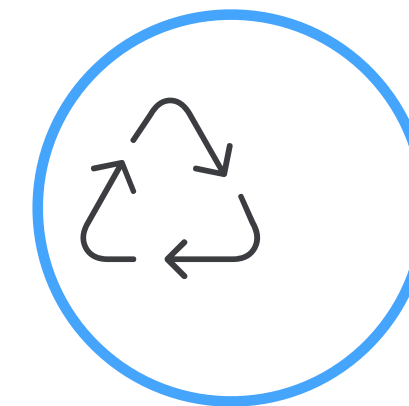
PATH TO SERVICES LIBRARY

Denise Duncan, Director Enablement Services October 29, 2014

Path to Services Library



The Path



Governance



Your Feedback



Resources

Using Template Patterns

Identify use case

Identify common patterns or recognize great idea in the form of customer request

- Add meta data fields to every template in a newly configure CMS to support SEO

```
<meta name="title" content="IndexIndex"/>  
<meta name="description" content="Pharmaceutical CMS is committed to your health"/>  
<meta name="keywords" content=""/>  
<meta name="asset_id" content="11301"/>  
<meta name="template_id" content="11215"/>
```

Using Template Patterns

- Find meta data input and output example from a previous implementation
- Ask developers on the forums if they have an example to share

```
Input.ShowTextBox("HTML Title", "html_title", width: 70, helpMessage: "<br />in brow  
Input.ShowTextBox("Page Meta Title", "meta_title", width: 70);  
Input.ShowTextBox("Page Meta Description", "meta_desc", width: 70, height: 5);  
Input.ShowTextBox("Page Meta Keywords", "meta_keywords", width: 70, height: 5);
```

Identify code

Identify best version of implementation to use and refine

Using Template Patterns

- Create ShowMetaInput in /System/Library/InputHelper
- Create MetaStringBuilder in /System/Library/OutputHelper

Add to Local Library

Code is validated and made into a method, distributed in local CMS library.

```
public static void ShowMetaInput(Boolean gConfig = false)
{
    Input.ShowHeader("Page Meta Information");
    if (gConfig)
    {
        Input.ShowMessage("This is the default meta data and will be included in the configuration file. ");
    }
    else
    {
        Input.ShowMessage("This meta content will be added to the page. ");
    }
    InputHelper.InsertBlank();
    Input.ShowTextBox("HTML Title", "html_title", width: 70, helpMessage: " ");
    Input.ShowTextBox("Page Meta Title", "meta_title", width: 70);
    Input.ShowTextBox("Page Meta Description", "meta_desc", width: 70, helpMessage: " ");
    Input.ShowTextBox("Page Meta Keywords", "meta_keywords", width: 70, helpMessage: " ");
    InputHelper.InsertBlank();
}
```

```
public static String MetaStringBuilder(Asset asset, Asset globalConfigAsset)
{
    StringBuilder metaOutput = new StringBuilder(String.Empty);
    Asset config = Asset.Load("/") + asset.Parent.AssetPath + "/_Config";
    try
    {
        String myPageTitle = "";
        if (config.IsLoaded)
        {
            myPageTitle = config["html_title"];
        }
        myPageTitle = String.IsNullOrEmpty(myPageTitle) ? asset["html_title"] : myPageTitle;
        myPageTitle = String.IsNullOrEmpty(myPageTitle) ? asset["page_title"] : myPageTitle;
        myPageTitle = String.IsNullOrEmpty(myPageTitle) ? asset.Label : myPageTitle;
        if (!String.IsNullOrEmpty(myPageTitle))
        {
            metaOutput.AppendFormat("<title>{0}</title>", myPageTitle);
            metaOutput.AppendLine();
        }
    }
    catch { }
    string metaTag = "";
    try
```

InputHelper.ShowMetaInput();

InputHelper.ShowMetaInput();

InputHelper.ShowMetaInput();

Now the meta data fields can be added all the templates using the new methods available in the local library.

Path to Services Library

Steps to get to the API

Identify use case

Identify common patterns or recognize great idea in the form of customer request

Identify Code

Identify best version of implementation to use and refine

Add to Local Library

Code is validated and made into a best practice, distributed in local CMS library.

Document

Document configuration, user scenario, benefits and effort in a playbook

Add to Services Library

Validated, documented code is integrated in product as a part of the Services API

ServicesInput.ShowMetaInput();

ServicesInput.ShowMetaInput();

ServicesInput.ShowMetaInput();

Now the meta data fields can be added all the templates using the new methods available in the **Services** library.

ServicesInput.ShowMetalInput();

Options – sConfig:true, shows a different help message when using this function on a site config.

Design pattern: Set your defaults on a site config, then use the same fields in output.

Fields are used in ServicesOutput.MetaOutputBuilder();

Fields created: html_title, meta_author, meta_desc, meta_keywords, meta_title

Page Meta Information Collapse

This meta content will be added to the page. If incomplete, the CMS will include the default meta information added to the Site Configuration.

HTML Title

in browser bar, default title if no page title entered. Leave blank to use page's label or title.

Page Meta Author

Page Meta Title

Page Meta Description

Page Meta Keywords

ServicesOutput.MetaOutputBuilder()

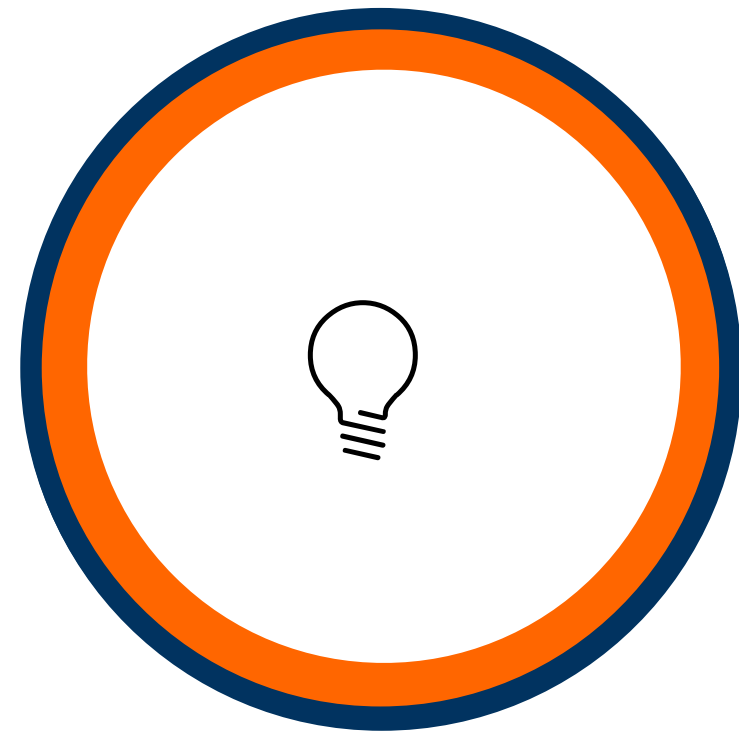
Output method supporting fields generated by ServicesInput.ShowMetaInput, ServicesInput.ShowMetaRobotsInput. Also includes parameters for showing common crownPeak properties (id, template)

Example:

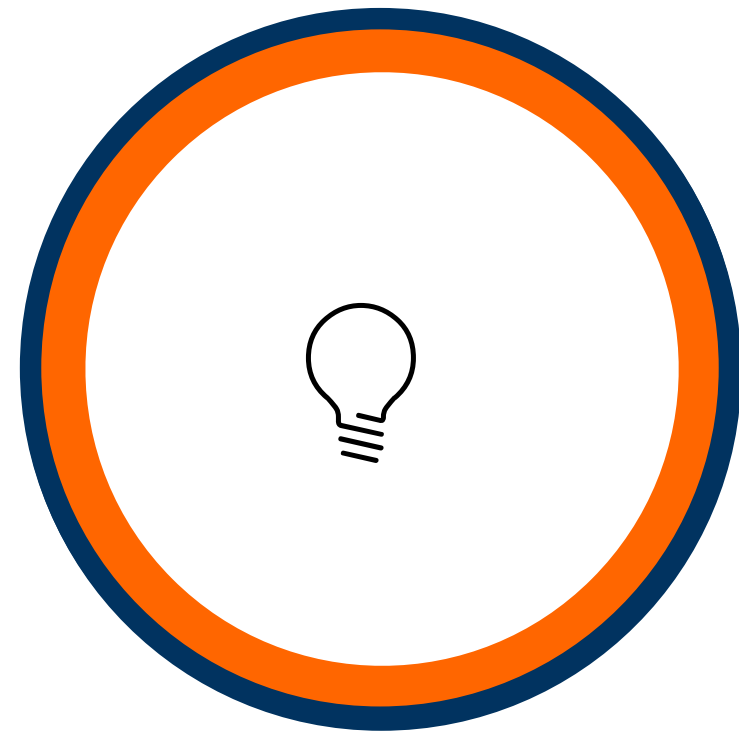
```
Asset configAsset = Asset.Load("/Site/_Site Config/_Site Config");  
Out.WriteLine(ServicesOutput.MetaOutputBuilder(asset, configAsset, true, true,  
false) );
```

Returns / Outputs

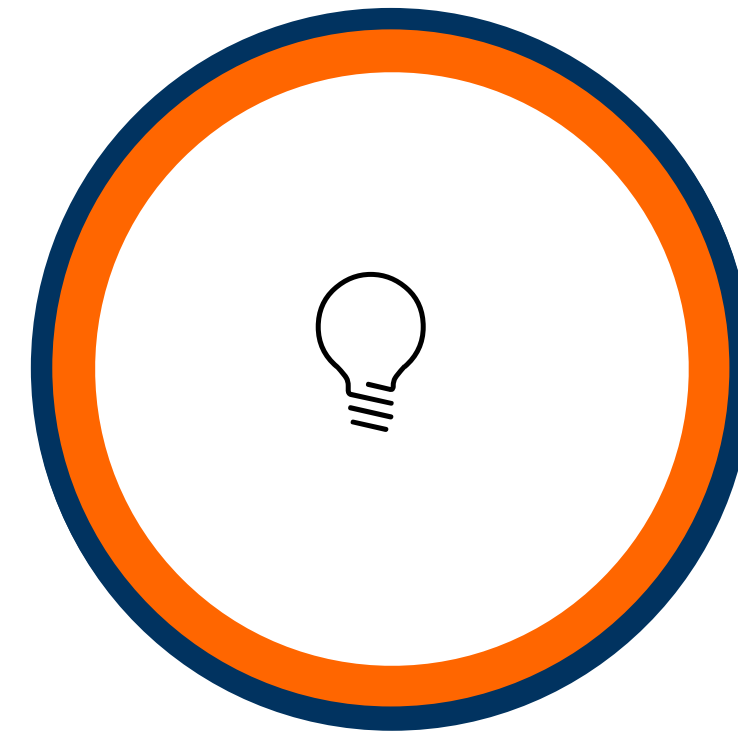
```
<meta name="title" content="page or default from config"/>  
<meta name="description" content="page or default from config"/>  
<meta name="keywords" content="page,or,default,from,config"/>  
<meta name="asset_id" content="6176"/>  
<meta name="template_id" content="6173"/>
```



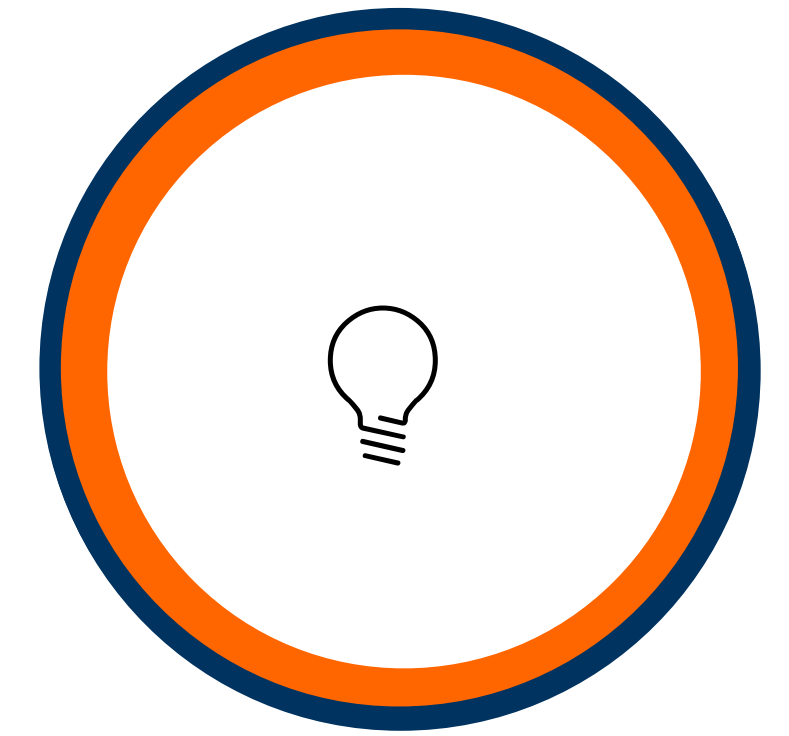
ServiceInput
Define Your Template



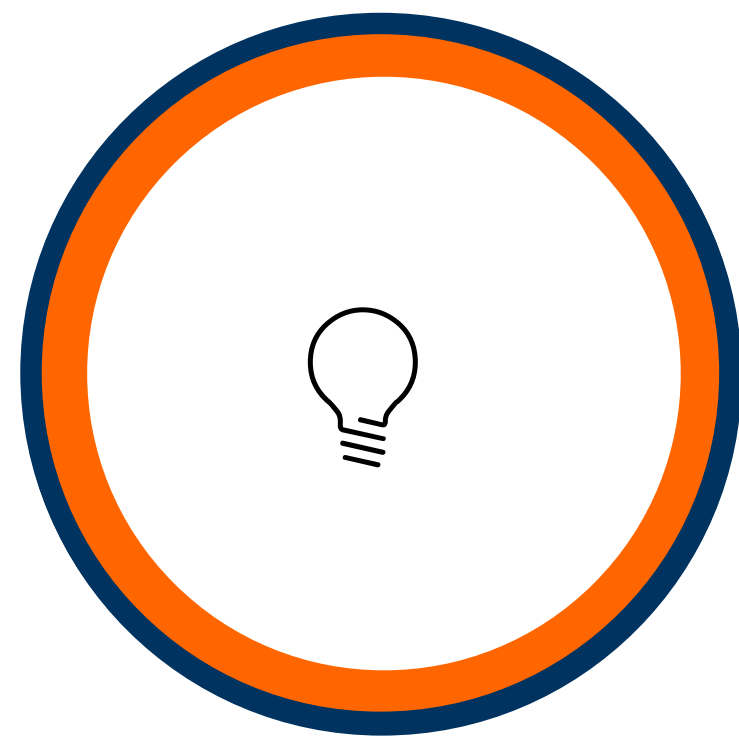
ServicesOutput
Define Your Templates



ServicesUtil
Common Service Utilities



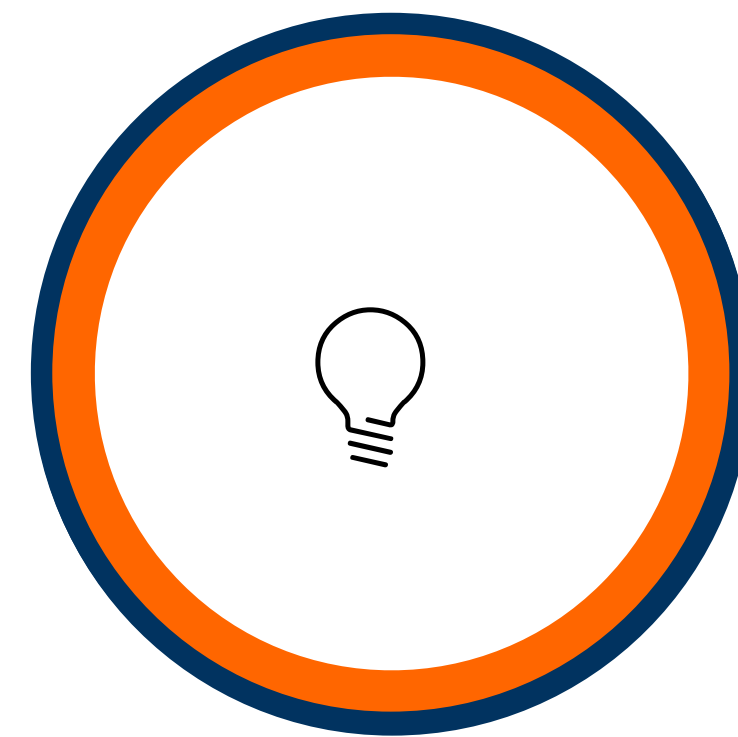
ServicesMinifier
Tools for Minification



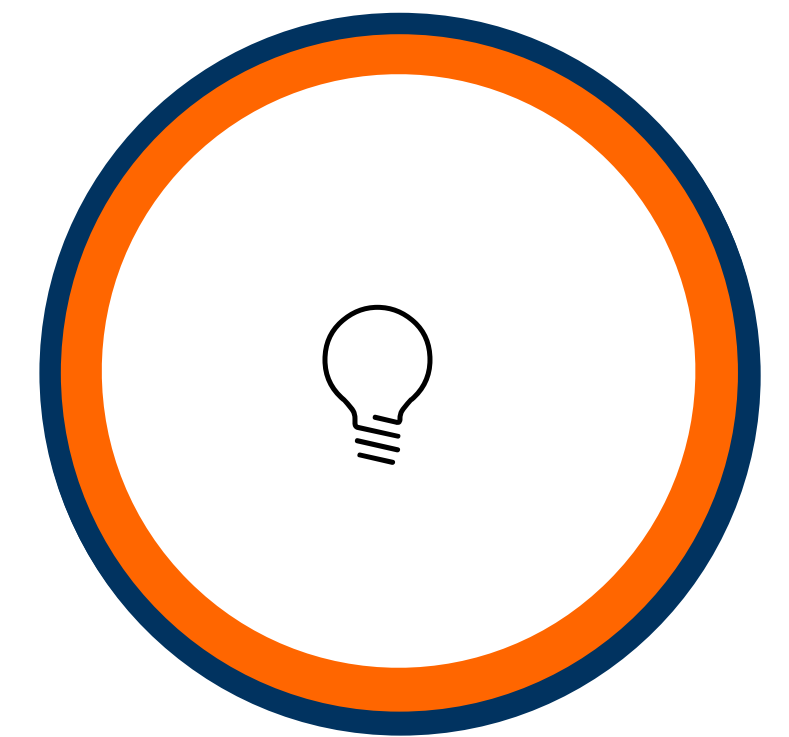
ServicesTMFGeneral
General Functions



ServicesTMFInput
Enable TMF in the Input



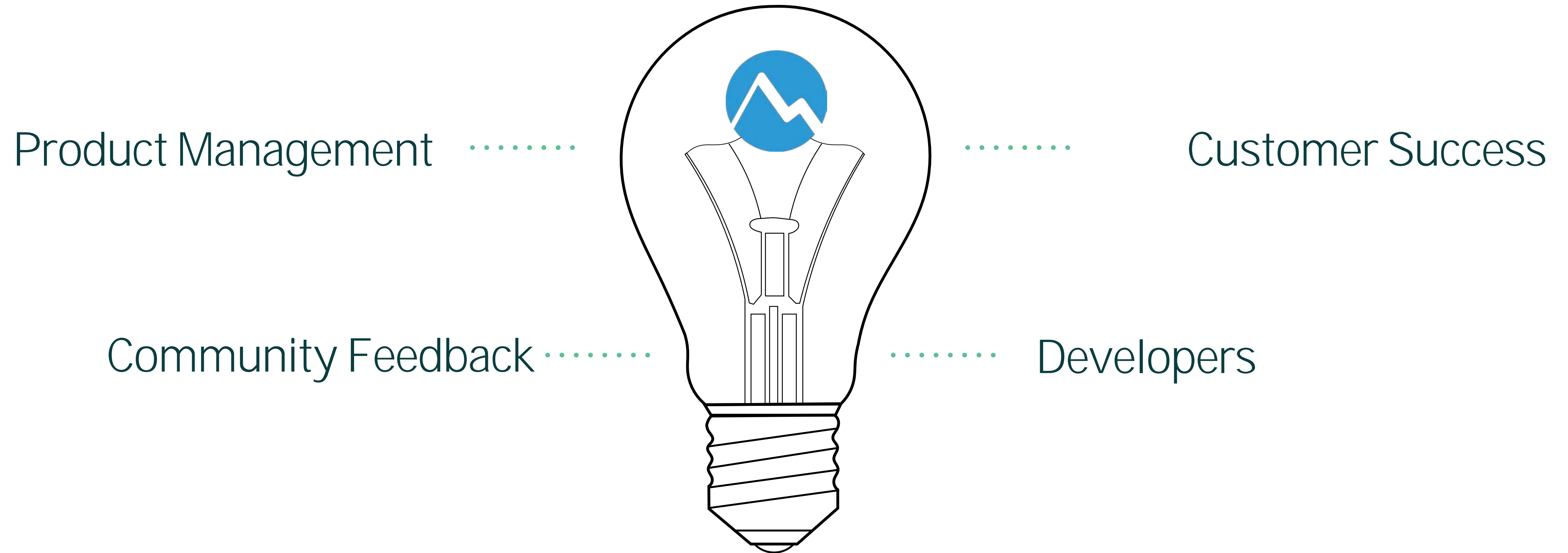
ServicesTMFOutput
Enable TMF in the Output



ServicesTMFPostInput
Function to Execute

We want your feedback!

We are here to hear your ideas and requests.



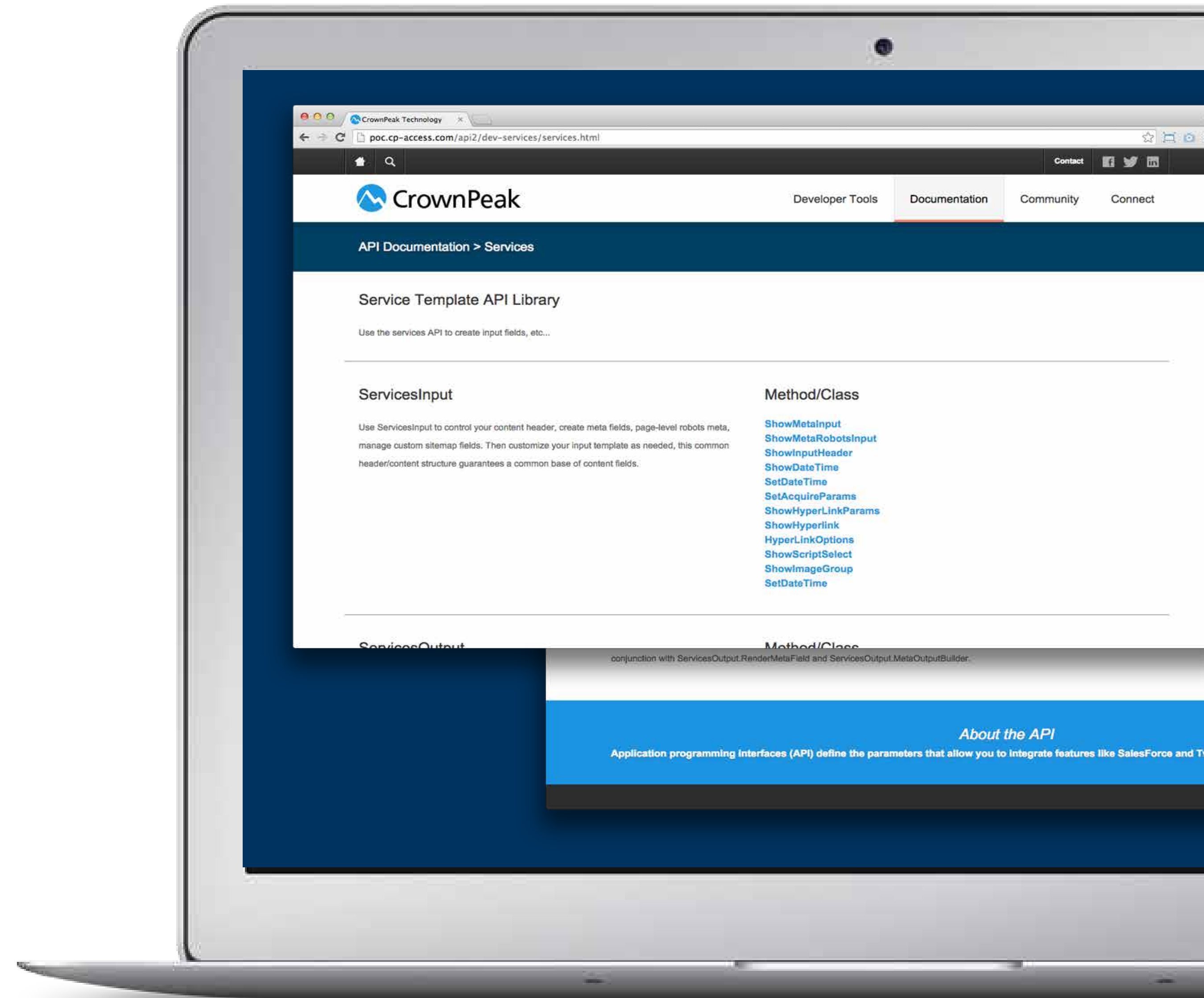


Resources

Service Template Documentation

<http://poc.cp-access.com/api2/Index.html>

- Designed to support the CrownPeak Extensions (i.e.: TMF)
- CMS developers can use this documentation to leverage
- Advanced features of the CrownPeak platform
- Best practices of the CrownPeak Professional Services (PS) team



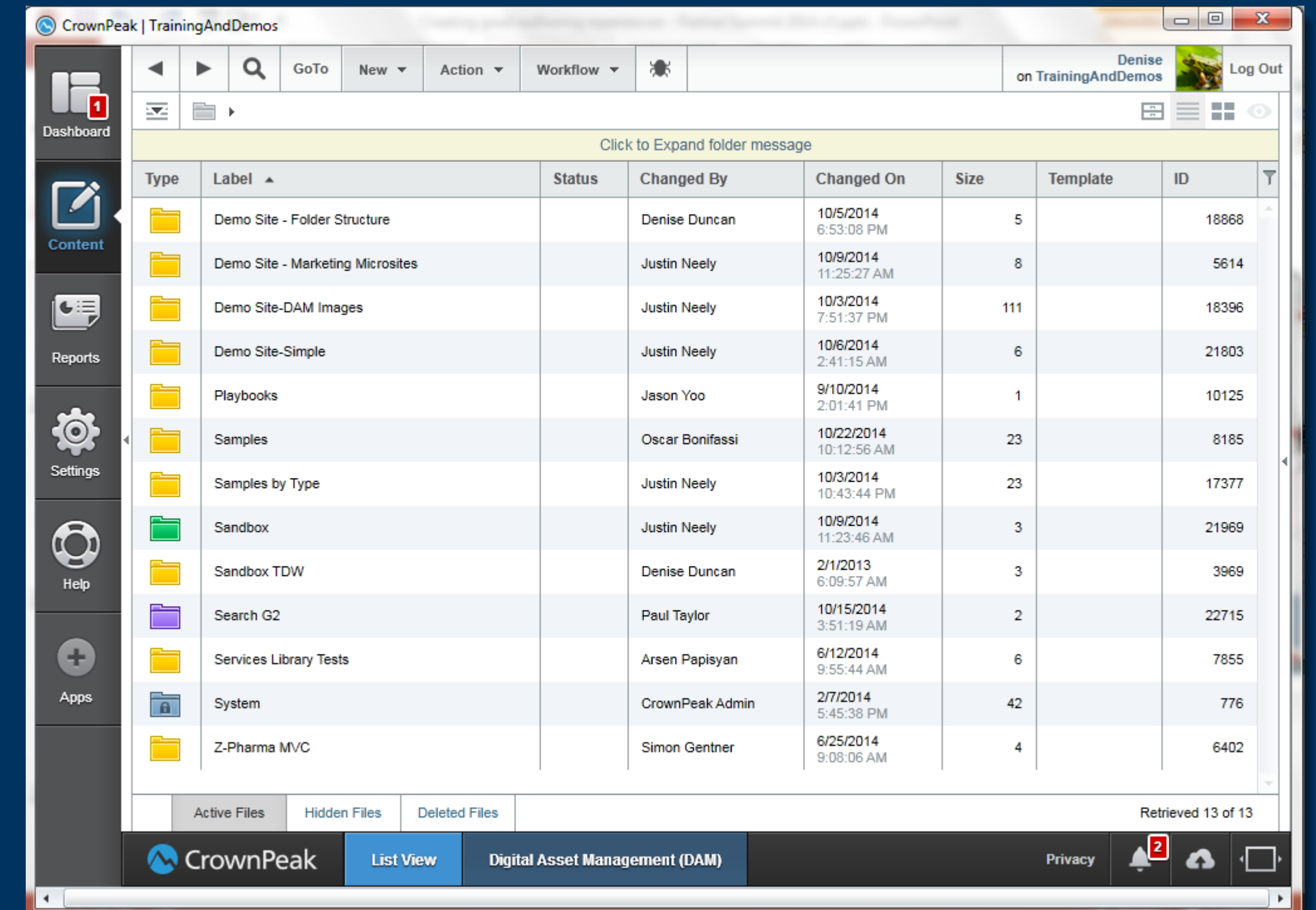
Demos and Documentation

- Code samples are available in the training instance

<https://cms.crownpeak.com/TrainingAndDemos/UI/>

- Reference documents and articles available on Connect

<https://connect.crownpeak.com/products/platform/cms>



The screenshot displays the CrownPeak TrainingAndDemos interface. The main content area shows a table of files and folders. The table has columns for Type, Label, Status, Changed By, Changed On, Size, Template, and ID. The data is as follows:

Type	Label	Status	Changed By	Changed On	Size	Template	ID
Folder	Demo Site - Folder Structure		Denise Duncan	10/5/2014 6:53:08 PM	5		18868
Folder	Demo Site - Marketing Microsites		Justin Neely	10/9/2014 11:25:27 AM	8		5614
Folder	Demo Site-DAM Images		Justin Neely	10/3/2014 7:51:37 PM	111		18396
Folder	Demo Site-Simple		Justin Neely	10/6/2014 2:41:15 AM	6		21803
Folder	Playbooks		Jason Yoo	9/10/2014 2:01:41 PM	1		10125
Folder	Samples		Oscar Bonifassi	10/22/2014 10:12:56 AM	23		8185
Folder	Samples by Type		Justin Neely	10/3/2014 10:43:44 PM	23		17377
Folder	Sandbox		Justin Neely	10/9/2014 11:23:46 AM	3		21969
Folder	Sandbox TDW		Denise Duncan	2/1/2013 6:09:57 AM	3		3969
Folder	Search G2		Paul Taylor	10/15/2014 3:51:19 AM	2		22715
Folder	Services Library Tests		Arsen Papisyan	6/12/2014 9:55:44 AM	6		7855
Folder	System		CrownPeak Admin	2/7/2014 5:45:38 PM	42		776
Folder	Z-Pharma MVC		Simon Gentner	6/25/2014 9:08:06 AM	4		6402

The interface also includes a sidebar with navigation options (Dashboard, Content, Reports, Settings, Help, Apps) and a footer with the CrownPeak logo, 'List View' button, 'Digital Asset Management (DAM)', and 'Privacy' link.

Demos and Documentation

- Playbooks
https://connect.crownpeak.com/developers_and_partners/playbooks
- Best Practices
https://connect.crownpeak.com/developers_and_partners/best_practices



What's Next?

- Service Library Governance

https://connect.crownpeak.com/developers_and_partners/getting_started/documents/developers/serviceslibrarygovernancemodelv3pdf

- Forum open for discussion

https://connect.crownpeak.com/developers_and_partners/forums/authoring_experience



New Powerpoint Template
Keynote



CrownPeak + Web Experience Management

