# Office AMS: OneDrive for Business BRANDING CUSTOMIZATION

|  |  |
| --- | --- |
| Summary: | Applies to: |
| This sample demonstrates how to perform taxonomy related operations. | * Office 365 Multi-Tenant (MT) * Office 365 Dedicated (D) * SharePoint 2013 on-premises |
| Solution: | Contoso.Core.OneDriveCustomizer, version 1.0 |
| Author: | Vesa Juvonen, Bert Jansen, Frank Marasco (Microsoft) |
| //\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  // THIS CODE IS PROVIDED \*AS IS\* WITHOUT WARRANTY OF  // ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING ANY  // IMPLIED WARRANTIES OF FITNESS FOR A PARTICULAR  // PURPOSE, MERCHANTABILITY, OR NON-INFRINGEMENT.  //\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* | |

## Version log

Change log between different releases.

|  |  |
| --- | --- |
| Version | Notes |
| 1.0 | Initial |

# General comments

This sample demonstrates how to access an end-user's OneDrive for Business site and apply custom branding to it automatically. The sample accesses the OneDrive for Business site using the social .NET client-side model (CSOM), which provides read access to user profile properties as well as access to the Site object for the OneDrive for Business site.

Branding is applied by uploading a custom them to the Site object by using file upload mechanisms in the CSOM and then applying the theme to the site.

You may also visit Vesa "[vesku](http://blogs.msdn.com/b/vesku/archive/2013/11/25/office365-apply-automatically-custom-branding-to-personal-site-skydrive-pro.aspx)" Juvonen blog for addition information about the techniques in his video recording.

**Note:**

In general, Microsoft recommends branding OneDrive for Business sites using themes, and avoiding custom master pags. If you customize master pages, you'll have to ensure that any new changes on the out-of-the-box master pages are reflected on custom master pages as well. On top of the themes, you can also inject custom CSS to the site to modify layout slightly without needing to change the master page.

Branding is applied from within app part in the solution. You can place the app part anywhere in the tenant since it operates across the site collections as needed. When the app part is accessing the OneDrive for Business site, the end-user will only see a GIF animation that indicates when app part is accessing the personal OneDrive for Business site. These locations are typical for this customizer:

* Intranet front page - When users arrive to Intranet, branding in OneDrive is checked and applied if necessary
* Public side of the personal - for example on the news feed page

# SCENARIo: Branding the ONEDRIVE SIte

This scenario demonstrates how to access the user profile, create the personal sites, and modify the site. We are using the Orange theme that is already available. Themes are not visible in the UI of OneDrive for Business sites, but you can still use them as well as deploy custom a custom theme. See the AMS sample CustomCSS and DeployCustomThemeWeb for additional information.

// Get user profile

ProfileLoader loader = Microsoft.SharePoint.Client.UserProfiles.ProfileLoader.GetProfileLoader(clientContext);

UserProfile profile = loader.GetUserProfile();

Microsoft.SharePoint.Client.Site personalSite = profile.PersonalSite;

clientContext.Load(personalSite);

clientContext.ExecuteQuery();

// Let's check if the site already exists The following code uses a timer job-based approach to schedule the creation of a OneDrive for Business site if it has not yet been created for a particular user.

if (personalSite.ServerObjectIsNull.Value)

{

profile.CreatePersonalSiteEnque(true);

clientContext.ExecuteQuery();

}

else

{

Web rootWeb = personalSite.RootWeb;

clientContext.Load(rootWeb);

clientContext.ExecuteQuery();

// Setting the custom theme to host web yes its orange

SetThemeBasedOnName(clientContext, rootWeb, "Orange");

}

# SharePOINT ONLINE SETUP

The first step to create the application principal. The app principal is an actual principal in SharePoint 2013 for the app that can be granted permissions. To register the app principal, we will use the “\_layouts/AppRegNew.aspx”.

Now we need to grant permissions to the app principal. You will have to navigate to another page in SharePoint which is the “\_layouts/AppInv.aspx”. This is where you will grant the application Tenant permissions, so that our Site Provisioning application may create site collections.

<AppPermissionRequests AllowAppOnlyPolicy="true">

<AppPermissionRequest Scope="http://sharepoint/content/tenant" Right="FullControl" />

</AppPermissionRequests>

# Dependencies

* Microsoft.SharePoint.Client.dll
* Microsoft.SharePoint.Client.Runtime.dll
* Microsoft.SharePoint.Client.UserProfiles