



Social Connected 1.1 for Sitecore CMS 6.5

# Administrator's and Developer's Guide

*A Quick Start Guide and Configuration Reference for Administrators and Developers*

## Table of Contents

Chapter 1	Introduction.....	4
1.1	Features .....	5
1.2	Architectural Requirements and Limitations .....	6
Chapter 2	Social Connector.....	7
2.1	Configuration.....	8
2.1.1	Usage .....	8
2.1.2	Configuration Actions .....	8
2.2	Configuring Access to Social Networks .....	10
2.2.1	Creating and Configuring a Facebook Web Application .....	10
2.2.2	Creating and Configuring a Twitter Web Application .....	12
2.2.3	Creating and Configuring a LinkedIn Web Application .....	15
2.2.4	Creating and Configuring a Google+ Web Application .....	17
2.3	Configuring Access to the Visitor's Profile Information .....	22
2.3.1	Web Application Access Request for Facebook, LinkedIn and Google+ .....	22
2.3.2	Twitter Web Application Access Request .....	23
2.4	Placing Social Connector Controls on a Web Page.....	25
2.4.1	Selecting a Data Source for the Controls.....	25
2.5	Mapping the Fields in the Visitor Profile.....	26
2.6	Synchronizing User Information .....	27
Chapter 3	Posting Updates to Social Networks on Goal Triggering.....	28
3.1	Configuration .....	29
3.1.1	Usage .....	29
3.1.2	Configuration Actions .....	29
3.2	Creating the Update Text.....	30
3.3	Asking the Visitor if they Want Updates Posted Automatically .....	31
3.4	Placing Goal Context Information in the Message Body .....	32
Chapter 4	Posting Content Updates to Social Networks .....	33
4.1	Configuration.....	34
4.1.1	Usage .....	34
4.1.2	Configuration Actions .....	34
	URL Generating .....	34
4.2	Configuring Access to Social Networks .....	35
4.2.1	Facebook.....	35
4.2.2	Twitter.....	35
4.3	Creating an Account Item.....	36
4.4	Creating an Update Text.....	40
Chapter 5	Analytics Features.....	41
5.1	Putting Like and Tweet Buttons on the Web Page .....	42
5.2	Tracking Like and Tweet Buttons Usage .....	43
5.3	The Personalization Rule for the Social Connected Module .....	44
Chapter 6	Developer's Notes .....	45
6.1	The Social Connected Module Pipelines .....	46
6.1.1	CreateSocialMessage Pipeline .....	46
6.1.2	ReadSocialMessage Pipeline .....	46
6.1.3	PublishMessage Pipeline .....	46
6.1.4	BuildMessage Pipeline.....	46
6.1.5	MatchUser Pipeline .....	47
Chapter 7	Tips and Tricks.....	48
7.1	How to Switch Language in the Sitecore Desktop .....	49
7.2	How to Use the Error Handling Mechanism .....	51

7.2.1	Error Handling in Sitecore Desktop.....	51
7.2.2	Error Handling on the Website.....	51

# Chapter 1

## Introduction

This document is designed for Sitecore administrators and developers. It contains information about how to set up, configure, and tune the module.

The document contains the following chapters:

- **Chapter 1 — Introduction**  
An introduction to the Sitecore Social Connected module.
- **Chapter 2 – Social Connector**  
Instructions for administrators about configuring and using the Social Connector.
- **Chapter 3 – Posting Updates to Social Networks on Goal Triggering**  
Instructions for administrators how to configure Goal Messaging.
- **Chapter 4 – Posting Content Updates to Social Networks**  
Instructions for administrators how to configure Content Messaging.
- **Chapter 5 – Analytics Features**  
A description of the analytics features in the module.
- **Chapter 6 – Developer’s Notes**  
Information about module pipelines and processors.
- **Chapter 7 – Tips and Tricks**  
Tips and tricks for administrators.

## 1.1 Features

The Sitecore Social Connected module provides the following features:

### **Social Connector**

The Social Connector lets website visitors log in to the website using the credentials from their social network accounts. The website then receives information about the visitor from their social network profile. You can also use the Social Connector to personalize the website.

### **Posting Updates to Social Networks on Goal Triggering**

This feature works with the Social Connector and posts updates to the visitor's wall in the social network every time they trigger a goal on the website.

### **Posting Content Updates to Social Networks**

This feature posts updates about Sitecore content automatically to the walls of the selected social network accounts when you publish Sitecore items. You can also post updates manually without publishing Sitecore items.

### **Analytics**

You can place **Like** and **Tweet** buttons on a web page as Sitecore controls and then use Sitecore analytics to track the activity of website visitors.

In this document the phrase "post update" means to send a message to a social network and put it on the Facebook wall or the Twitter timeline.

## 1.2 Architectural Requirements and Limitations

The Sitecore Social Connected module cooperates with third party products and services. You should therefore pay special attention to the requirements and limitations related to your Sitecore solution.

### Internet Access to Social Services

In a distributed Sitecore solution that is built on one or more environments, such as a Content Management server and one or more Content Delivery servers, make sure the servers have access to the social network services over the Internet.

### Configuring Security

The Social Connector creates a new Sitecore user the first time a website visitor uses their social network credentials to log in to the website. If the Sitecore solution is built on one or more environments, such as a Content Management server and one or more Content Delivery servers, make sure you configure security according to the Sitecore recommendations. Security is an inseparable part of the Core database that must be shared between all the Content Management and Content Delivery servers if you want to keep security accounts synchronized. If the Core database cannot be shared between the servers, use database replication instead.

For more information about configuring security in multi-environment solutions, see the *Sitecore Scaling Guide*, the section *Isolating Instances within the CM and CD Environments* and the section *How to: Address Security Caching Clearing*.

## Chapter 2

# Social Connector

Social Connector is a feature of the Sitecore Social Connected module that lets website visitors log in to the website with their social network credentials. Sitecore does not get access to the visitor's credentials; it uses the API to communicate with the social network.

When a website visitor logs in with their social network credentials, the Sitecore Social Connected module receives all the information from the visitor's profile that the visitor has allowed the social network to share. This information is saved in the user profile, which Sitecore creates automatically in the *extranet* domain.

On the one hand, Social Connector lets website visitors log in to the website with their social network credentials and thereby avoid the registration process. On the other hand, Social Connector lets the website collect more information about a website visitor from their social network profile.

This chapter contains the following sections:

- Configuration
- Configuring Access to Social Networks
- Configuring Access to the Visitor's Profile Information
- Placing Social Connector Controls on a Web Page
- Mapping the Fields in the Visitor Profile
- Synchronizing User Information

## 2.1 Configuration

Before we explain how to configure Social Connector, we will demonstrate how to log in to the website with social network credentials.

### 2.1.1 Usage

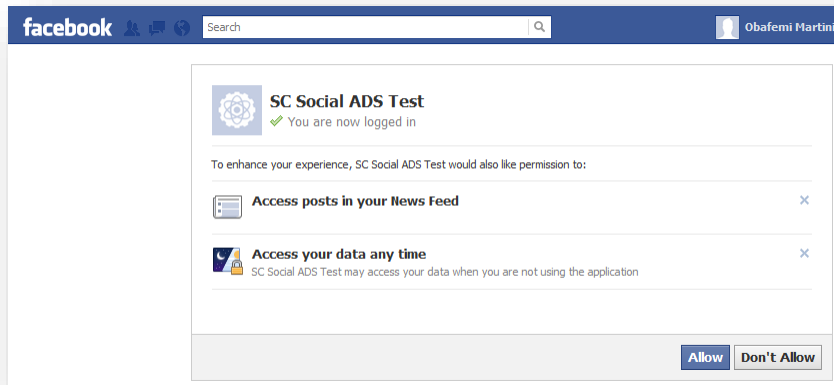
In this example, a website visitor uses their Facebook credentials to log in to the website.

To connect with Facebook credentials:

1. A website visitor navigates to the website:
2. To log in to the website with their social network credentials, the visitor clicks the social network icon. The module redirects them to the social network login page.



3. On the login page, the visitor enters their credentials. The social network asks the visitor to grant the web application of the website access to the visitor's profile information.



4. If the visitor grants access to their profile information, the system redirects them back to the website being already logged in. Sitecore creates a new user profile or updates information for the existing user profile.
5. If the visitor does not grant access to their profile information, the system redirects them back to the website. The visitor still has to log in to the website.

### 2.1.2 Configuration Actions

To configure Social Connector:

1. Create a web application for each social network and a corresponding Sitecore item. This web application will concatenate the module with the social network. For more information about configuring a web application, see the section *Configuring Access to Social Networks*.



2. Configure an access request for the information that the web application will receive from the website visitor's profile on the social network. For more information about configuring an access request, see the section *Configuring Access to the Visitor's Profile Information*.
3. Configure and place log in controls for the social networks on the web page. For more information about how to place log in controls, see the section, *Placing Social Connector Controls on a Web Page*.

After you have performed the obligatory configuration actions, you can:

- Arrange a Sitecore scheduled task to periodically update the user profile with information from the social network. For more information about how to schedule a Sitecore task, see the section *Synchronizing User Information*.

## 2.2 Configuring Access to Social Networks

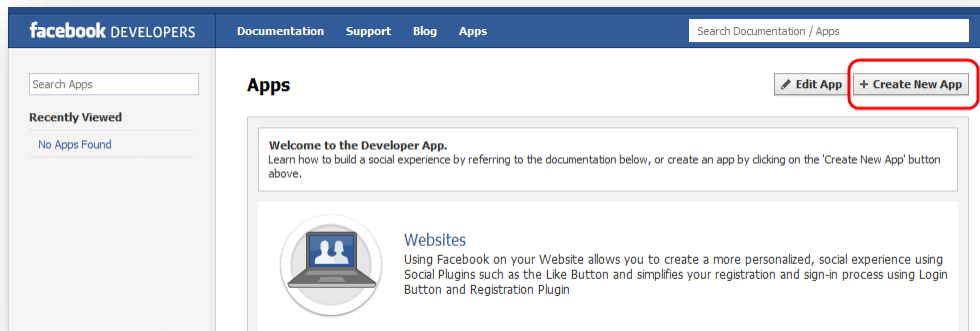
When interacting with social networks, Sitecore Social Connector uses the API of the social networks or the OAuth protocol for Google+. You must create a web application for every social network that you want the module to communicate with. You must also create a Sitecore item for each web application.

In a multisite scenario, you should create a web application for each website.

### 2.2.1 Creating and Configuring a Facebook Web Application

To create and configure a web application for Facebook:

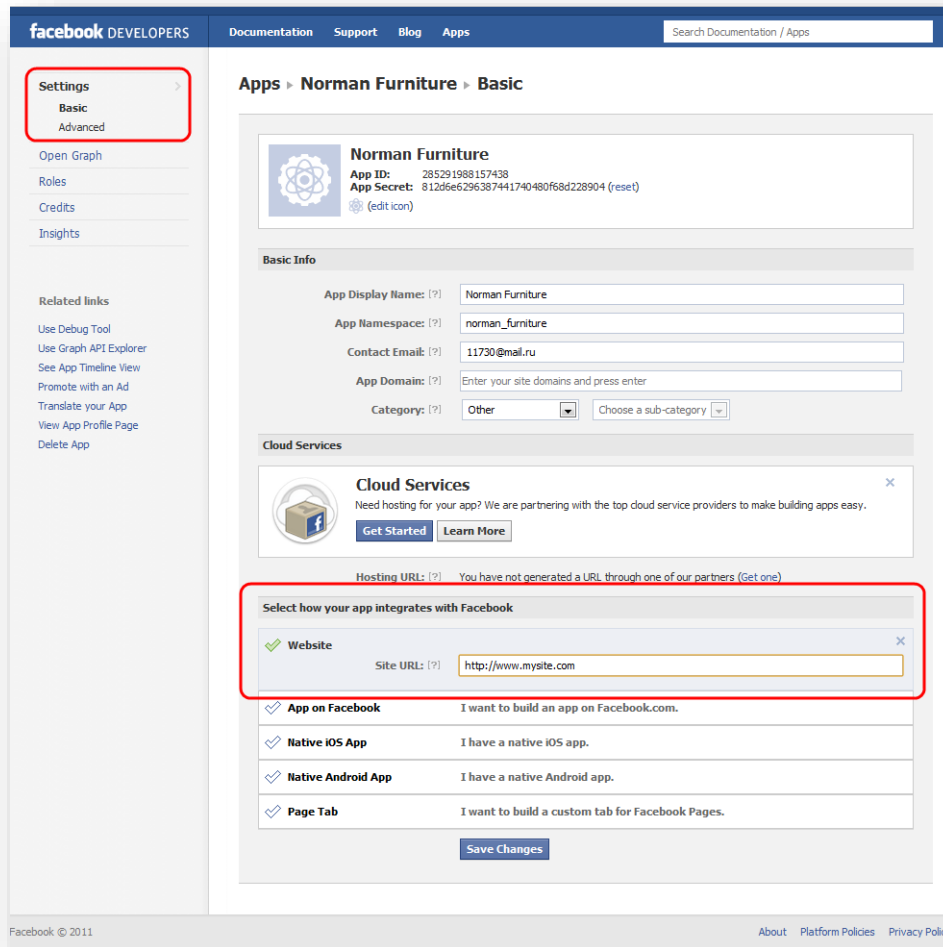
1. Go to <https://developers.facebook.com/apps> and create a new web application.



For more information about how to create a new web application, go to <https://developers.facebook.com/docs/guides/canvas/>

2. In the web application, on the **Basic** settings page, fill in all the required fields.

- In the **Select how your app integrates with Facebook** section, select **Website** and in the **Site URL** field, enter the URL of the web page where you want to place the login controls.



- In Sitecore, in the **Content Editor**, navigate to the `sitecore/system/social/applications/default` folder.
- In this folder create an item based on the `sitecore/templates/system/social/application` template.

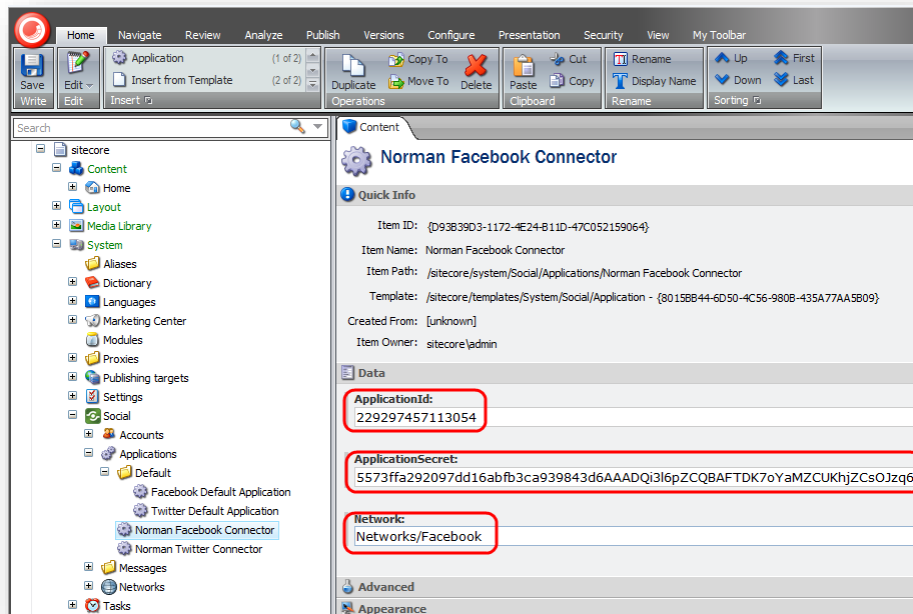
**Important**

The `Default` folder must contain only one item for the corresponding social network. If you are going to use several web applications for one social network, create the new items in the `sitecore/system/social/applications` folder.

- In the new item, enter the appropriate information:

Field	Value
ApplicationId	The <b>App ID</b> value of the web application
ApplicationSecret	The <b>App Secret</b> value of the web application

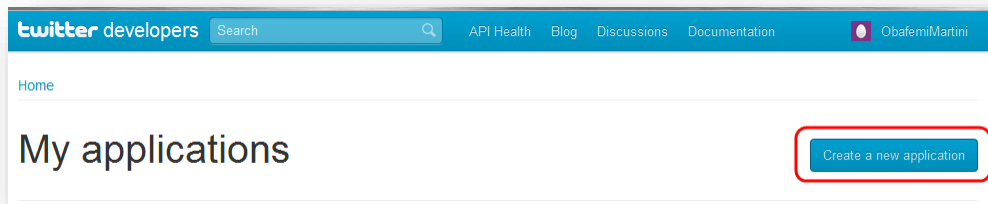
Field	Value
Network	Facebook



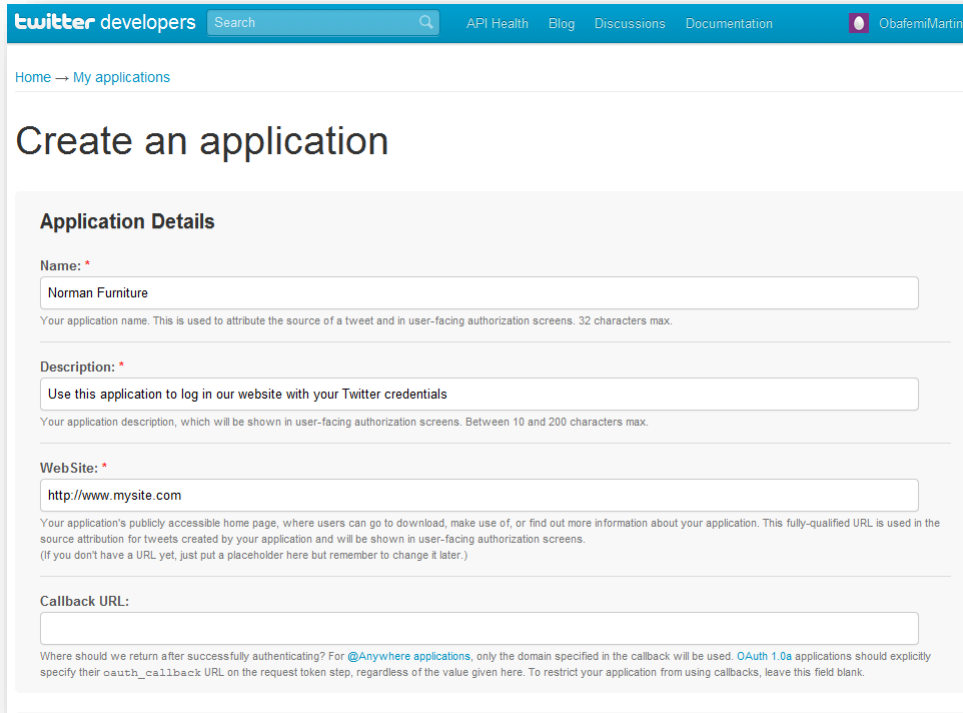
## 2.2.2 Creating and Configuring a Twitter Web Application

To create and configure a web application for Twitter:

1. Go to <https://dev.twitter.com/apps> and create a new web application.



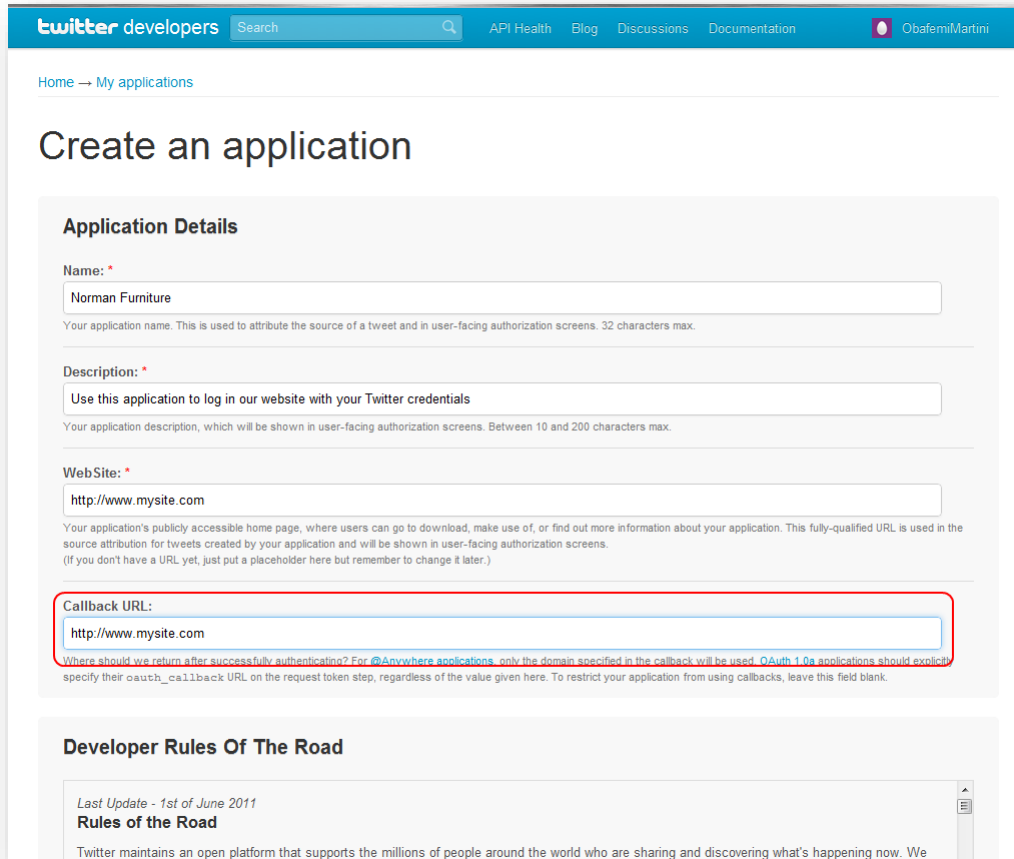
2. In the web application, in the **Application details** section, fill in all the required fields.



The screenshot shows the 'twitter developers' interface. The page title is 'Create an application'. The 'Application Details' section contains the following fields:

- Name:** \* Norman Furniture  
Your application name. This is used to attribute the source of a tweet and in user-facing authorization screens. 32 characters max.
- Description:** \* Use this application to log in our website with your Twitter credentials  
Your application description, which will be shown in user-facing authorization screens. Between 10 and 200 characters max.
- Web Site:** \* http://www.mysite.com  
Your application's publicly accessible home page, where users can go to download, make use of, or find out more information about your application. This fully-qualified URL is used in the source attribution for tweets created by your application and will be shown in user-facing authorization screens. (If you don't have a URL yet, just put a placeholder here but remember to change it later.)
- Callback URL:**  
Where should we return after successfully authenticating? For @Anywhere applications, only the domain specified in the callback will be used. OAuth 1.0a applications should explicitly specify their oauth\_callback URL on the request token step, regardless of the value given here. To restrict your application from using callbacks, leave this field blank.

- In the **Callback URL** field, enter the URL of the webpage where you want to place the login controls.

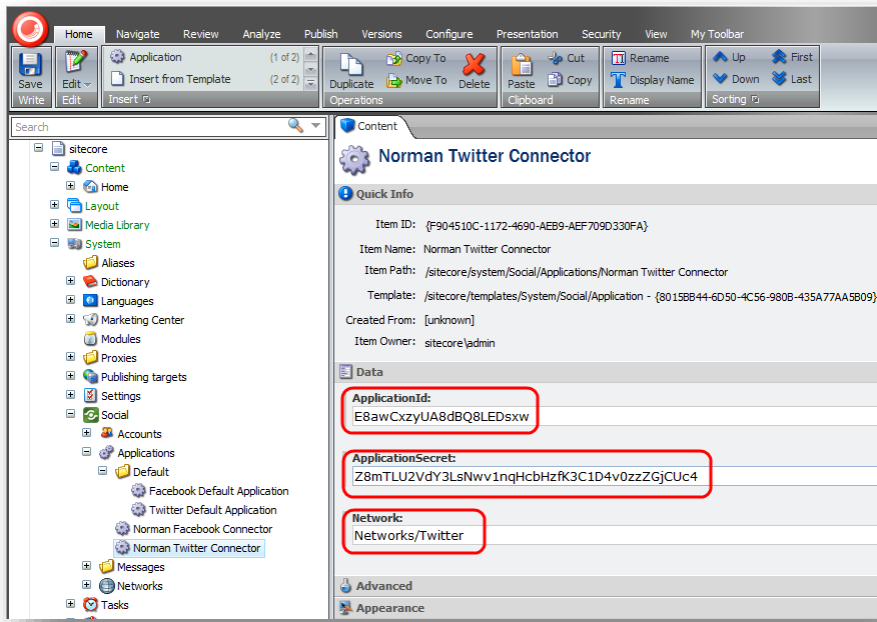


- In Sitecore, in the **Content Editor**, navigate to the `sitecore/system/social/applications/default` folder.
- In this folder create an item based on the `sitecore/templates/system/social/application` template.

**Important**  
 The **Default** folder must contain only one item for the corresponding social network. If you are going to use several web applications for one social network, create the new items in the `sitecore/system/social/applications` folder.

- In the new item, enter the appropriate information:

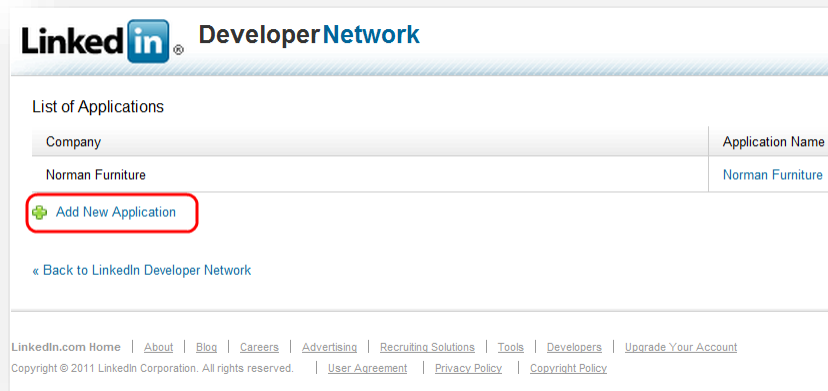
Field	Value
ApplicationId	The <b>Consumer Key</b> value of the web application
ApplicationSecret	The <b>Consumer Secret</b> value of the web application
Network	Twitter



### 2.2.3 Creating and Configuring a LinkedIn Web Application

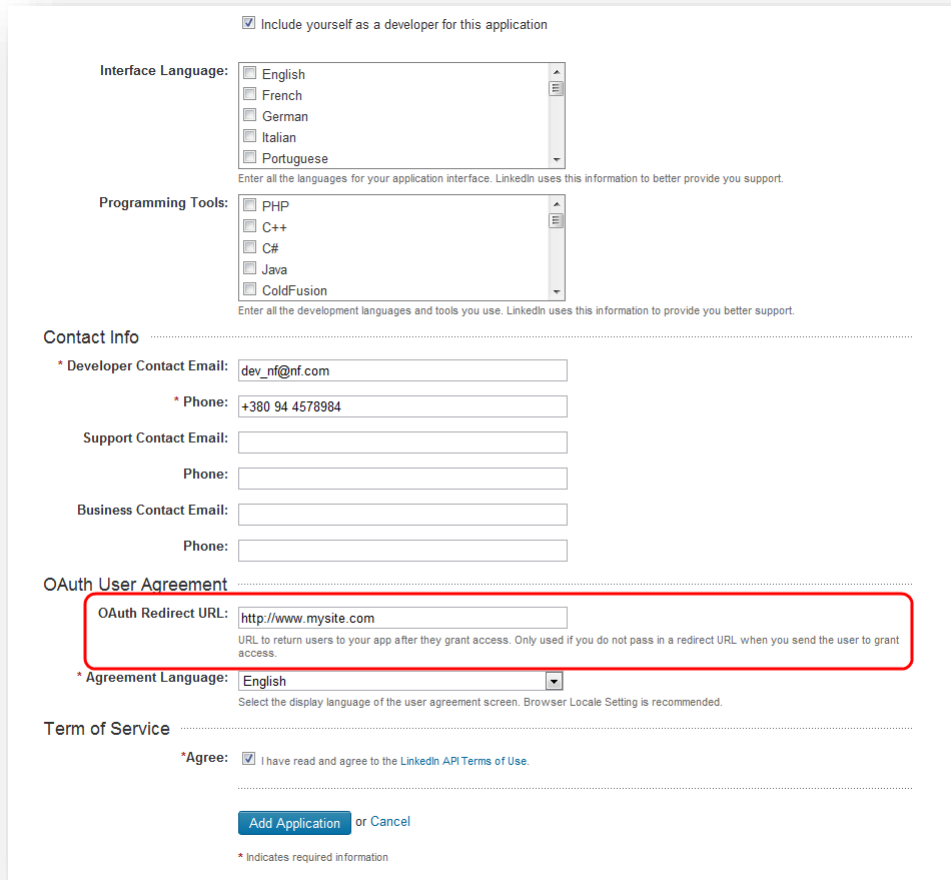
To create and configure a web application for LinkedIn:

1. Go to <https://www.linkedin.com/secure/developer> and create a new web application.



2. In the web application, fill in all the required fields.

- In the **OAuth User Agreement** section, in the **OAuth Redirect URL** field, enter the URL of the web page where you want to place the login controls.



Include yourself as a developer for this application

Interface Language:  English  
 French  
 German  
 Italian  
 Portuguese

Programming Tools:  PHP  
 C++  
 C#  
 Java  
 ColdFusion

Contact Info

\* Developer Contact Email:   
 \* Phone:   
 Support Contact Email:   
 Phone:   
 Business Contact Email:   
 Phone:

OAuth User Agreement

OAuth Redirect URL:   
URL to return users to your app after they grant access. Only used if you do not pass in a redirect URL when you send the user to grant access.

\* Agreement Language:

Term of Service

\* Agree:  I have read and agree to the [LinkedIn API Terms of Use](#)

or

\* Indicates required information

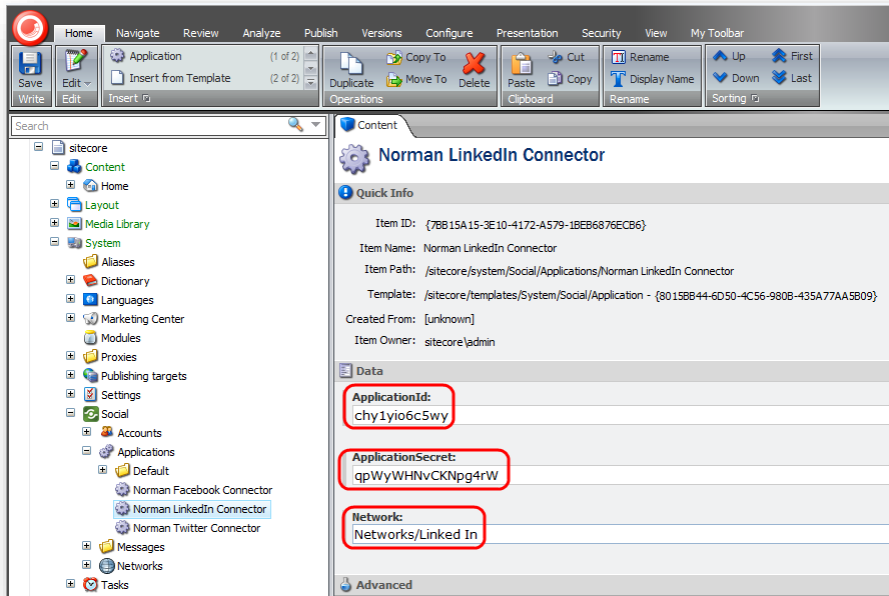
- In Sitecore, in the **Content Editor**, navigate to the `sitecore/system/social/applications/default` folder.
- In this folder create an item based on the `sitecore/templates/system/social/application` template.

**Important**  
 The `Default` folder must contain only one item for the corresponding social network. If you are going to use several web applications for one social network, create the new items in the `sitecore/system/social/applications` folder.

- In the new item, enter the appropriate information:

Field	Value
ApplicationId	The <b>API Key</b> value of the web application
ApplicationSecret	The <b>Secret Key</b> value of the web application
Network	LinkedIn

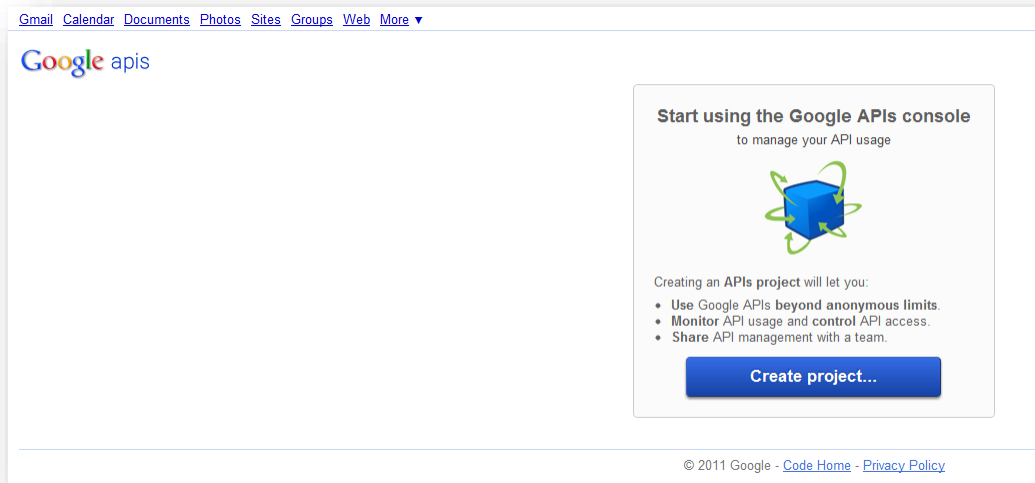




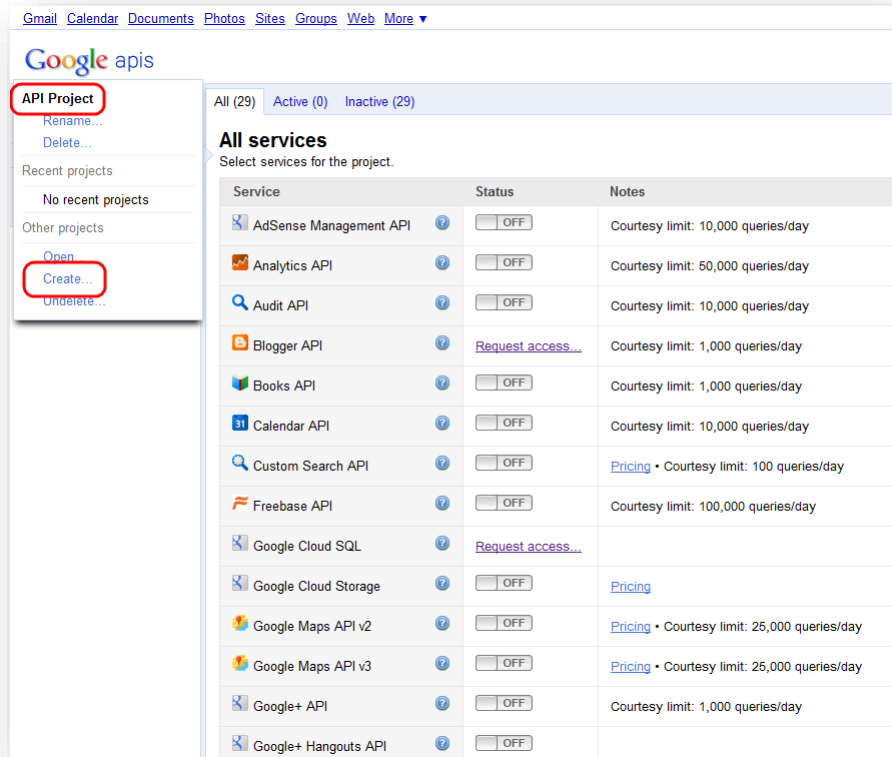
## 2.2.4 Creating and Configuring a Google+ Web Application

To create and configure a web application for Google+:

1. Go to <https://code.google.com/apis/console> and create a new project:



- On the **All Services** page, expand **API Project** menu and click **Create**.

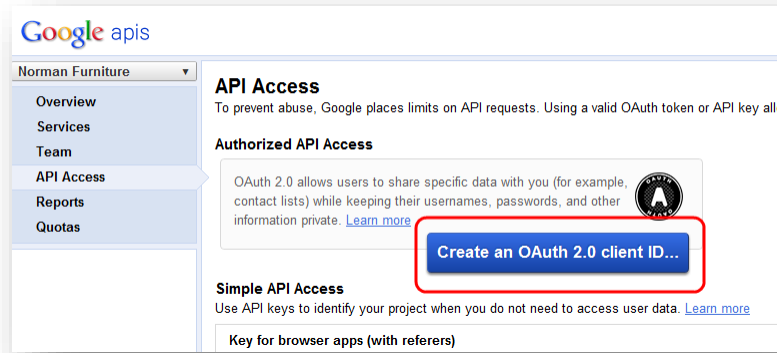


- In the **Create Project** dialog box, enter the name of the project and then click **Create Project**.
- On the **All Services** page, set **Google+ API** to **On**.



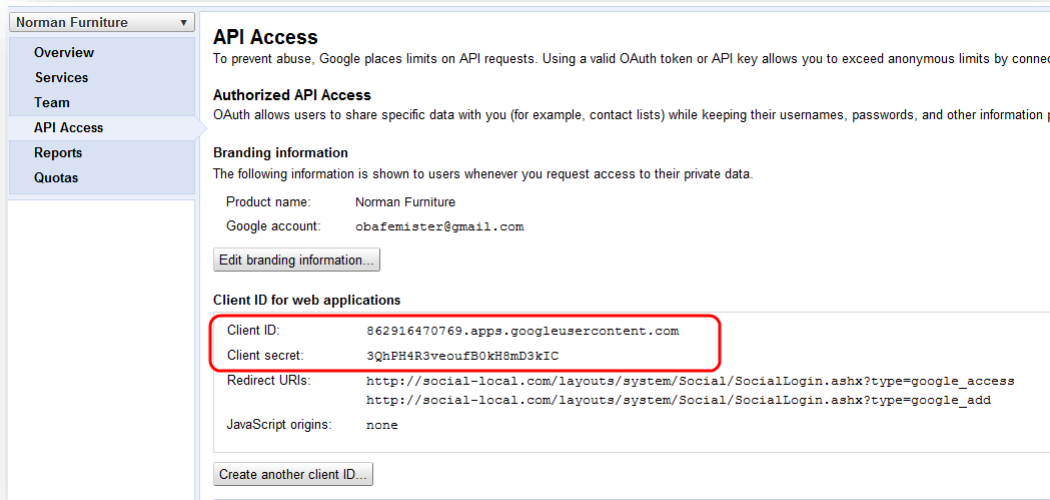
- In the left-hand menu, click **API Access**.

- On the **API Access** page, click **Create an OAuth 2.0 client ID**.



- In the **Create Client ID** dialog box, on the **Branding Information** page, enter the name in the **Product Name** field and click **Next**.
- In the **Create Client ID** dialog box, on the **Client ID Settings** page, select **Web Application** type. Skip the website field.
- In the **Create Client ID** dialog box, on the **Client ID Settings** page, click **Create Client ID**.
- On the **API Access** page, in the **Client ID** for web application section, click **Edit Settings**.
- In the **Edit Client Settings** dialog box, enter the following URIs in the **Authorized Redirect URIs** field:
  - `http://hostname/layouts/system/Social/SocialLogin.ashx?type=google_access`
  - `http:// hostname /layouts/system/Social/SocialLogin.ashx?type=google_add`
  - `http:// hostname /layouts/system/Social/SocialLogin.ashx?type=access`
 where `hostname` is the name of the host that is running the website.
- In the **Edit Client Settings** dialog box, click **Update**.

13. On the **API Access** page, save the values of the **Client ID** and **Client secret** fields.



**Norman Furniture**

- Overview
- Services
- Team
- API Access**
- Reports
- Quotas

### API Access

To prevent abuse, Google places limits on API requests. Using a valid OAuth token or API key allows you to exceed anonymous limits by connecting to the API.

#### Authorized API Access

OAuth allows users to share specific data with you (for example, contact lists) while keeping their usernames, passwords, and other information private.

#### Branding information

The following information is shown to users whenever you request access to their private data.

Product name: Norman Furniture  
Google account: obafemister@gmail.com

[Edit branding information...](#)

#### Client ID for web applications

Client ID: 862916470769.apps.googleusercontent.com  
Client secret: 3QhFH4R3veoufB0kH8mD3kIC

Redirect URIs: [http://social-local.com/layouts/system/Social/SocialLogin.ashx?type=google\\_access](http://social-local.com/layouts/system/Social/SocialLogin.ashx?type=google_access)  
[http://social-local.com/layouts/system/Social/SocialLogin.ashx?type=google\\_add](http://social-local.com/layouts/system/Social/SocialLogin.ashx?type=google_add)

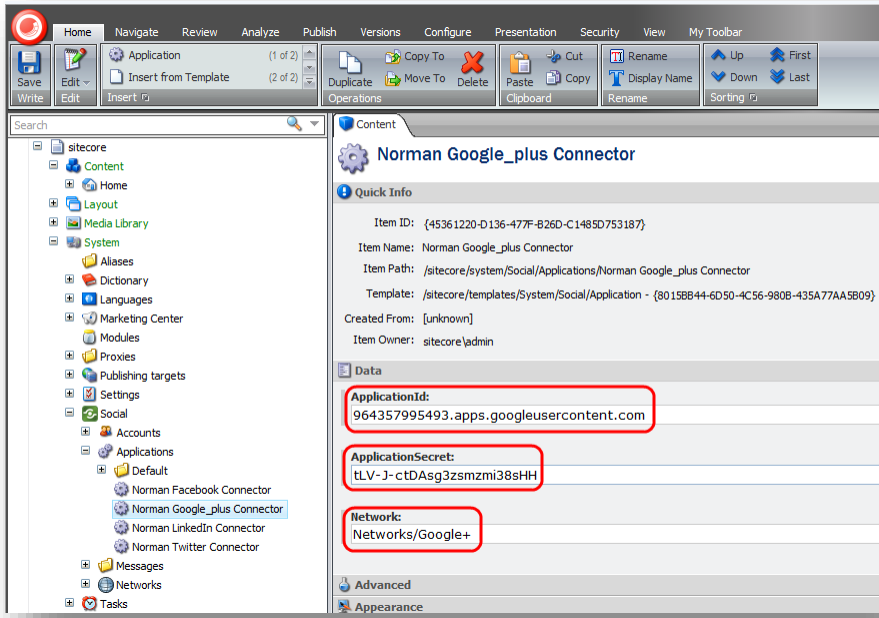
JavaScript origins: none

[Create another client ID...](#)

14. In Sitecore, in the **Content Editor**, navigate to the `sitecore/system/social/applications/default` folder.
15. In this folder create an item based on the `sitecore/templates/system/social/application` template.

#### Important

The `Default` folder must contain only one item for the corresponding social network. If you are going to use several web applications for one social network, create the new items in the `sitecore/system/social/applications` folder.

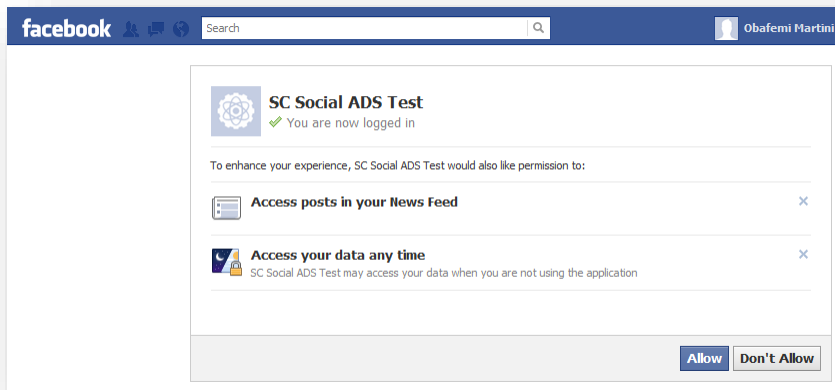


16. In the new item, enter the appropriate information:

Item Field	Value
ApplicationId	The <b>Client ID</b> value of the web application
ApplicationSecret	The <b>Client secret</b> value of the web application
Network	Google+

## 2.3 Configuring Access to the Visitor's Profile Information

When a website visitor logs in to the website with their social network credentials, the module redirects them to the login page of the social network. On that page, the visitor enters their credentials and the web application asks the visitor to grant it access to their profile information.



You can configure the exact information that the web application will ask for.

### 2.3.1 Web Application Access Request for Facebook, LinkedIn and Google+

For Facebook, LinkedIn, and Google+, you configure the access request in the `Sitecore.social.profilemapping.config` file.

To configure the fields that the web application can access:

1. Open the `[website_root]\app_config\include\sitecore.social.profilemapping.config` file.
2. In the configuration section for the social network, in the configuration section for the field that you want the web application to use, set the `field enabled` parameter to `true`.
3. If the field is private, you must set its `permission` parameter according to the official social network documentation.

For example, to get access to the **Groups**' field on Facebook, the field section in the `Sitecore.social.profilemapping.config` file should look like this:

```
<!-- 'Groups' field. -->
<!-- The Groups that the user belongs to. -->
<field enabled="true" originalKey="" sitecoreKey="fb_groups"
permission="user_groups" access="/me/groups" />
```

#### Note

At the moment, the Google+ API doesn't allow you to access some user profile fields such as **Formatted name**, **Family name**, **Given name**, and so on. This is why those fields are disabled in the `Sitecore.Social.ProfileMapping.config` file by default.

For more information about user profile fields, see the official documentation of the social network:

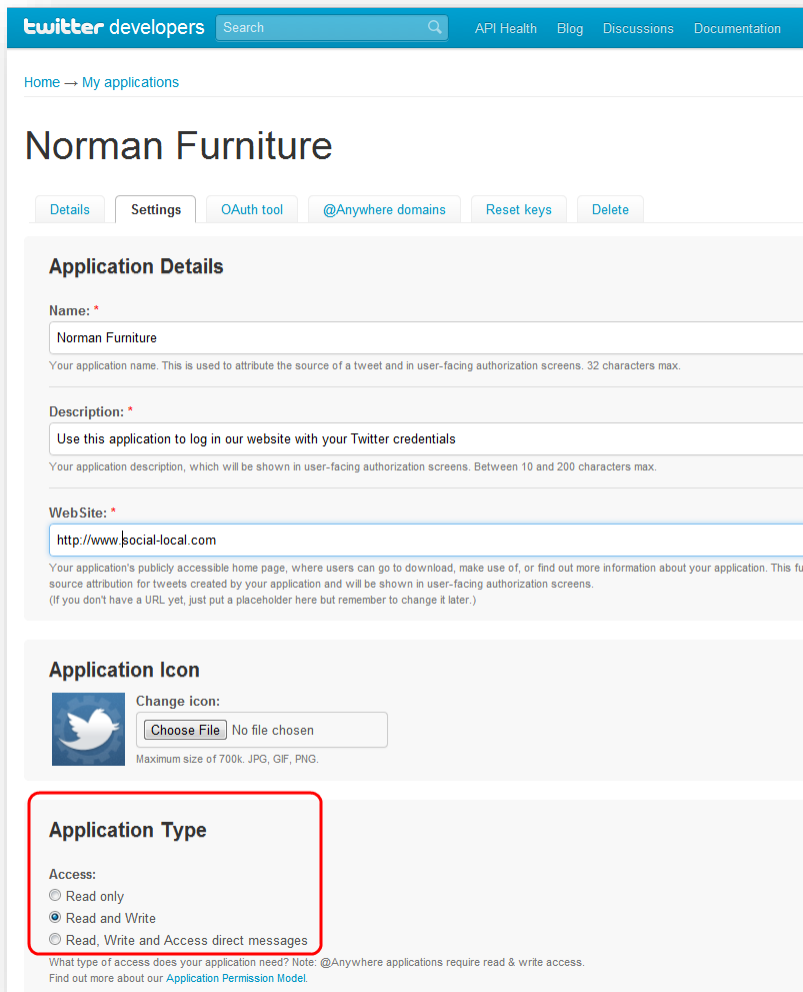
- For Facebook documentation, go to <http://developers.facebook.com/docs/reference/api/user/>.
- For LinkedIn documentation, go to <https://developer.linkedin.com/documents/profile-fields>.
- For new Google+ API, go to <https://developers.google.com/+/api/latest/people>.

### 2.3.2 Twitter Web Application Access Request

For Twitter, you can configure access request on the web application page.

To configure the fields that the web application can access:

1. Go to <https://dev.twitter.com/apps> and then go to the web application.
2. On the **Settings** tab, in the **Application Type** section, select the **Read and Write** or **Read, Write and Access direct messages** check box.



The screenshot shows the Twitter Developer application settings page for an application named "Norman Furniture". The page is titled "Norman Furniture" and has tabs for "Details", "Settings", "OAuth tool", "@Anywhere domains", "Reset keys", and "Delete". The "Settings" tab is active.

**Application Details**

**Name:** \*  
  
Your application name. This is used to attribute the source of a tweet and in user-facing authorization screens. 32 characters max.

**Description:** \*  
  
Your application description, which will be shown in user-facing authorization screens. Between 10 and 200 characters max.

**WebSite:** \*  
  
Your application's publicly accessible home page, where users can go to download, make use of, or find out more information about your application. This fully source attribution for tweets created by your application and will be shown in user-facing authorization screens. (If you don't have a URL yet, just put a placeholder here but remember to change it later.)

**Application Icon**

Change icon:  
 No file chosen  
Maximum size of 700k. JPG, GIF, PNG.

**Application Type**

**Access:**

Read only  
 Read and Write  
 Read, Write and Access direct messages

What type of access does your application need? Note: @Anywhere applications require read & write access. Find out more about our [Application Permission Model](#).

3. Click **Update these twitter application settings** to save the changes.

For more information about the user profile fields on Twitter, see the official Twitter documentation <https://dev.twitter.com/docs/api/1/get/users/show>.



## 2.4 Placing Social Connector Controls on a Web Page

You can configure and place log in controls for the social networks on the website. You can find the controls in the `sitecore/layout/sublayouts/social/connector` folder. You can fully customize these controls (change images, texts, and so on).

### 2.4.1 Selecting a Data Source for the Controls

If you have several web applications that you can use to log in, you must select one of them as a data source for the social connector controls. If you do not select a data source the social connector control uses the data source that is stored in the `sitecore/system/social/applications/default` folder.

#### Note

If you have more than one domain, you can only use the default folder web applications for one of the domains. For the other domains, you must specify the data sources explicitly.

When you select a web application for the data source of the control, make sure that the web application's settings contain the URL of the current Sitecore solution:

Social Network Web Application	The Field Containing Website URL
Facebook	Site URL
Twitter	Callback URL
LinkedIn	OAuth Redirect URL
Google+	Redirect URIs

## 2.5 Mapping the Fields in the Visitor Profile

When a website visitor logs in to the website with their social network credentials, the module receives their visitor profile information from the social network. This information is saved in the corresponding user profile on the website. The information is stored in the *Core* database, in the `aspnet_profile` table. By default, the module saves the information from the social network in the fields with the corresponding prefixes. For example, `fb` is a prefix for Facebook fields. You can change the field mappings in the `sitecore.social.profilemapping.config` file.

To change the field mappings between a social network visitor profile and the user profile on the website:

1. Open the `[website_root]\app_config\include\sitecore.social.profilemapping.config` file.
2. In the configuration section for the social network, set the following fields according to the official social network API documentation:

Field	Description
<code>originalKey</code>	All social networks use this field.
<code>Permission</code>	Facebook uses this field.
<code>Access</code>	Facebook uses this field.
<code>xmlPath</code>	LinkedIn uses this field.

3. In the configuration section for the social network, set the **sitecoreKey** field to the name of the Sitecore database field.

### Important

We recommend that you do not change the **sitecoreKey** field values.

For more information about the user profile fields, see the official documentation of the social network:

- For Facebook documentation, go to <http://developers.facebook.com/docs/reference/api/user/>.
- For LinkedIn documentation, go to <https://developer.linkedin.com/documents/profile-fields>.
- For new Google+ API, go to <https://developers.google.com/+/api/latest/people>.
- For Twitter documentation, go to <https://dev.twitter.com/docs/api/1/get/users/show>.

For more information about the **originalKey** field for Twitter, see the properties of the **TwitterUser** class in the **Twitterizer 2** library.

For more information about the **originalKey** field for Google+, see the **WellKnownAttribute** class in the **DotNetOpenAuth** library.

## 2.6 Synchronizing User Information

When a website visitor logs in to the website, Sitecore Connector receives and saves all the user profile information from the social network that the visitor has allowed the network to share.

The module receives user profile information in two stages:

- When the website visitor enters their social network credentials, the module receives basic data from the social network: ID, email, application key, application secret, access token, and access token secret. The module uses this information to identify the visitor.
- When the visitor is identified, the module receives all the information from the user profile that the visitor has allowed to be shared. As a result this information comes later than basic data.

To regularly update user information with the user profile data from the social network, you can use a Sitecore scheduled task. This allows you to synchronize the user information with the social network regardless of the user's activity on your website.

By default the scheduled task for synchronizing data between the users' profiles on the social networks and their profile information on your website is created but not run.

To run the scheduled task:

1. In Sitecore, in the **Content Editor**, select the `Sitecore/system/tasks/schedules/social data synchronization` item.
2. In the **Schedule** field, set the schedule.

For more information about using the **Schedule** field, see the [Scheduled Tasks](#) article on the SDN.

Alternatively, you can set and run the scheduled task in the `sitecore.social.config` file:

1. In the `[website_root]\app_config\include\sitecore.social.config` file, in the `scheduling` section, uncomment the following string:

```
<agent type="Sitecore.Social.Connector.ScheduledTaskPerformer" method="SynchronizeData"
interval="23:59:59">
</agent>
```

2. In the `interval` parameter, specify how regularly the task should be run.

When the module synchronizes information, it follows these rules:

- If the field in the social network user profile has been updated, Sitecore updates this field in the user profile.
- If the field in the social network user profile has become blank, Sitecore clears this field in the user profile.

## Chapter 3

# Posting Updates to Social Networks on Goal

## Triggering

The Sitecore Social Connected module can automatically post updates to Facebook and Twitter when a website visitor triggers a goal on the website.

This feature requires that you have the Digital Marketing Suite 2.0 installed and that Social Connector is configured on the website.

For Facebook, you can post updates both to the visitor's wall and Facebook pages. For Twitter you can post tweets to the visitor timeline.

This chapter contains the following sections:

- Configuration
- Creating the Update Text
- Asking the Visitor if they Want Updates Posted Automatically
- Placing Goal Context Information in the Message Body

## 3.1 Configuration

The Social Connected module can post updates to social networks when a website visitor triggers a goal on the website. To enable this feature, you must configure the Social Connector because this feature requires that the website visitor logs in to the website with their social network credentials.

You can use this feature to post updates to Facebook and Twitter.

### 3.1.1 Usage

A visitor logs in to the website with their social network credentials. When the visitor triggers a goal the module posts an update to the corresponding social network. You can configure the goal and the text used in the update.

**Example:**

A website visitor logs in to the website with their Facebook credentials. The visitor uploads an image *Friday Party* to the website. The *uploading an image* goal is triggered and the Social Connected module posts an update to the visitor's wall on Facebook saying that the visitor uploaded the image called *Friday Party*.

### 3.1.2 Configuration Actions

To configure the posting of updates when a goal is achieved:

1. Configure Social Connector.

For more information about Social Connector configuration, see the section *Configuration*

2. Create a goal and assign a page event to it.

For more information about configuring goals and page events, see the document [Marketing Operations Cookbook](#), chapter *Goals, Campaigns, and Events*.

3. Create the update text that is sent to the social network.

For more information about editing update texts, see the section *Creating the Update Text*.

After you have configured this, you can:

- Create a UI control that asks website visitors for permission to post updates to their social networks when they trigger goals.

For more information about creating UI control, see the section *Asking the Visitor if they Want Updates Posted Automatically*.

- Create personalized updates using goal context parameters.

For more information about the usage of goal context parameters, see the section *Placing Goal Context Information in the Message Body*.

**Note**

The analytics features used by the Sitecore Social Connected module, such as goals, page events, and so on require Sitecore DMS 2.0.0 or later.

## 3.2 Creating the Update Text

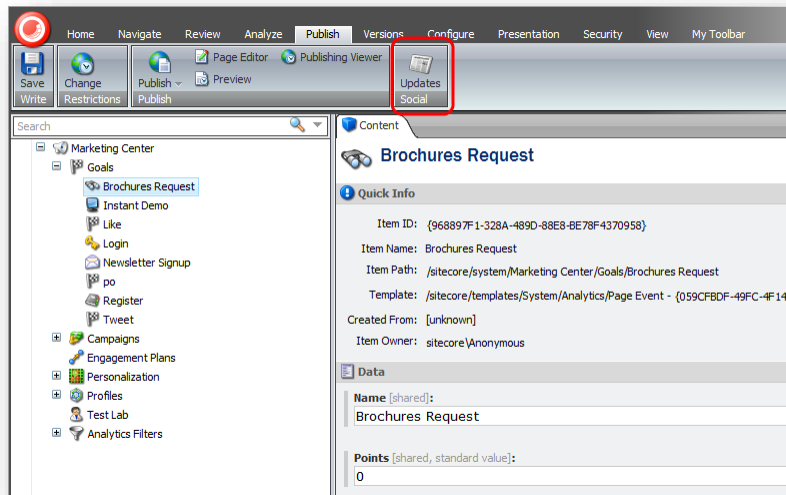
When you have configured Social Connector and assigned a goal to an item, you can edit the update text.

### Note

Twitter does not allow you to post updates with the same text more than once.

To edit the update text:

1. In Sitecore, in the **Marketing Center**, select the goal that you want to bind an update to. Usually, the goals are stored in the folder `sitecore/system/marketing center/goals`.
2. In the **Marketing Center**, on the **Publish** tab, click **Updates**. The **Updates** tab appears.



3. On the **Updates** tab, click **New Facebook Update** or **New Twitter Update**.
4. In the **Facebook message** or **Twitter message** dialog box, enter the text of update in the **Message** field. The rest of the fields are optional.

### Note

Facebook limits the update text to 5000 symbols and Twitter limits the text to 140 symbols.

5. In the **Facebook message** or the **Twitter message** dialog box, click **OK** to close the dialog box.

### Important

You must publish the messages that you have created for the goals before they can be sent to the social networks. You can find the messages folder in the `Sitecore/system/social` folder.

To put a URL in the middle of the update text, in the **Twitter message** dialog box, use the `$link` token in the **Text** field and enter the URL in the **Link** field. This token does not work for Facebook messages in the current version of the module.

### 3.3 Asking the Visitor if they Want Updates Posted Automatically

When you have configured Sitecore to post social network updates when a visitor achieves a goal on the website, this functionality applies to every visitor.

If you want to ask website visitors whether or not they want messages to be posted on their social network walls, you must disable the sending of messages by default and create a UI control to ask the visitor what they prefer:

1. In the `[website_root]\app_config\include\sitecore.social.config` file, set the value of the `Social.AllowPublishByDefault` setting to `false`.
2. Use the Sitecore API to create a UI control that asks the visitor whether or not they want to allow Sitecore Social Connected to send messages to their social network walls.

Use the following methods of the

`Sitecore.Social.Security.Managers.PublishPreferenceManager` class:

- o `GetPreference`
- o `SetPreference`.

For example, the following code snippet asks the current visitor to decide whether or not to publish a specific message. The message is specified in the message id:

```
var manager = new Sitecore.Social.Security.Managers.PublishPreferenceManager();
var status = manager.GetPreference(id, Sitecore.Context.User);
var newstatus = status == PublishPreference.Allowed ? PublishPreference.Denied :
PublishPreference.Allowed;
manager.SetPreference(id, Sitecore.Context.User, newstatus);
```

### 3.4 Placing Goal Context Information in the Message Body

You can use the Sitecore API to place some of the context information about how or when a goal is achieved in a personalized message that is posted on a social network.

For example, when a website visitor uploads an image to the website and triggers the *uploading image* goal, you can transfer the name of the image as a goal parameter and then add this information as a token to the message body. Use the `GoalUtil` class and the `RegisterEventParameters` method to transfer goal parameters.

For example, the following code snippet passes a username parameter to the social publishing engine:

```
var parameters = new Dictionary<string, string>();
parameters.Add("username", Sitecore.Context.Use.Name);
GoalUtil.RegisterEventParameters("Login", parameters);
```



## Chapter 4

# Posting Content Updates to Social Networks

When you publish Sitecore items on the website, the Sitecore Social Connected module can post updates to one or more accounts in Facebook and Twitter. You can also post a social network update without publishing items.

This chapter contains the following sections:

- Configuration
- Configuring Access to Social Networks
- Creating an Account Item
- Creating an Update Text

## 4.1 Configuration

The Sitecore Social Connected module can post updates to social networks when you publish Sitecore items. You bind the update text to the content item and the module posts this update automatically when you publish the content item. You can also post these updates manually without publishing the content.

This feature works with Facebook and Twitter. You can post updates to a visitor's wall in Facebook and to a Facebook page. You can post updates (tweets) to the timeline of a Twitter user who used the web application to create a Twitter account.

### 4.1.1 Usage

#### Example.

Your company produces and sells smartphones. Every time the company produces a new smartphone, a new web page is created on the website describing its features. To attract more attention to the new model you can automatically post updates to the website's wall on Facebook or to the corresponding timeline on Twitter publicizing the fact that you have unveiled a new product. This update can contain the link to the new webpage.

### 4.1.2 Configuration Actions

To configure the posting of content updates:

1. Create and configure a web application in the social network and create a Sitecore item for the web application. The web application will concatenate the module with the social network.

For more information about how to create and configure a web application, see the section *Configuring Access to Social Networks*.

2. Create a Sitecore item for the social network account.

For more information about how to create an account item, see the section *Creating an Account Item*.

3. Create an update and post it immediately to the social network or post it later when you publish the Sitecore item.

For more information about creating and publishing messages, see the section *Creating an Update Text*.

### URL Generating

In a single site solution, where the Content Management and Content Delivery servers are on different domains, in the `social.config` file, in the `Social.LinkDomain` attribute, specify the host name.

In a multisite solution, on the Content Management server, in the `web.config` file, the module uses the path of the content item that the message is associated with, to look for a matching website in the `<sites>` section and its `startItem` attribute. The module then uses the `hostName` attribute of that website to generate the URLs.

## 4.2 Configuring Access to Social Networks

When interacting with social networks, the module uses the API of the social networks. You must create a web application for every social network that you want the module to communicate with and create a Sitecore item for the social network account.

### 4.2.1 Facebook

To configure the posting of content updates to Facebook:

1. Create a new web application or use an existing one and then create a Sitecore item for the web application.

For more information about creating a web application, see the section *Creating and Configuring a Facebook Web Application*.

2. In Sitecore, in the Content Editor, in the `/sitecore/system/Social/Accounts/Facebook/` folder, create a new account item using the wizard. For more information about how to use the wizard, see the section *Creating an Account Item*.

### 4.2.2 Twitter

To configure posting content updates to Twitter:

1. Create a new web application or use the existing one and create a Sitecore item for the web application. For more information about how to create a web application, see the section *Creating and Configuring a Twitter Web Application*.

2. Set the required permissions on the web application.

For more information about the required permissions, see the section *Twitter Web Application*.

3. In Sitecore, in the Content Editor, in the `/sitecore/system/Social/Accounts/Twitter/` folder, use the wizard to create a new account item.

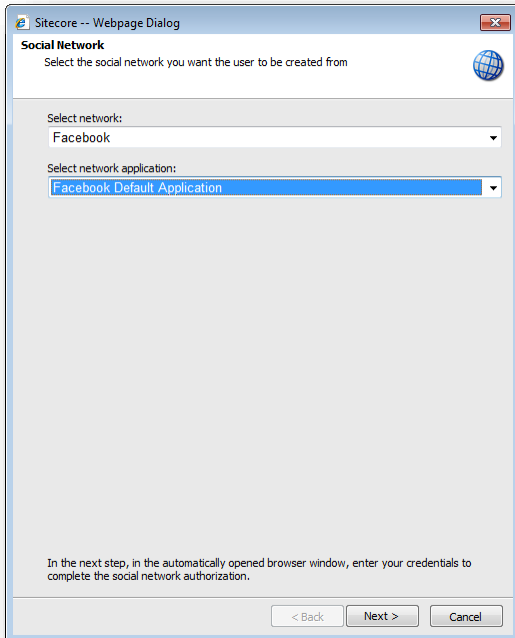
For more information about the wizard, see the section *Creating an Account Item*.

### 4.3 Creating an Account Item

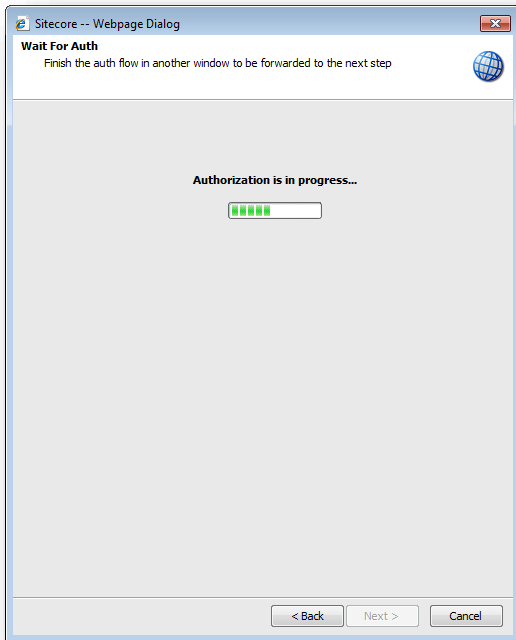
To create an account item, in the Content Editor, select the folder `sitecore/system/social/accounts`. In the Content Editor, on the **Social** tab, click **Add Account** to start the **Add Network Account** wizard..

The Add Network Account wizard guides you through the process of creating a network account.

In the **Select Network** window, select the social network that you want to create an account for.



If you have created more than one web application for this social network, in the **Choose Web Application** field, select the web application that the account should use to cooperate with the social network.



As part of this step, the wizard opens a new browser window, in which you must enter the credentials of the social network user account. The module will post updates to the wall of this user or its pages.



To set the user groups which can see activity from the web application on Facebook, click **Public** and select one of the user groups.

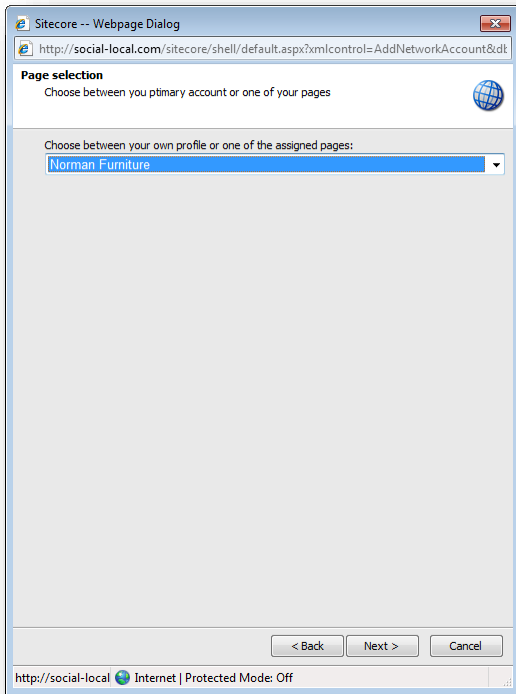
Click **Go to App**.

Review the permissions that the web application asks for and click **Allow**.

**Note**

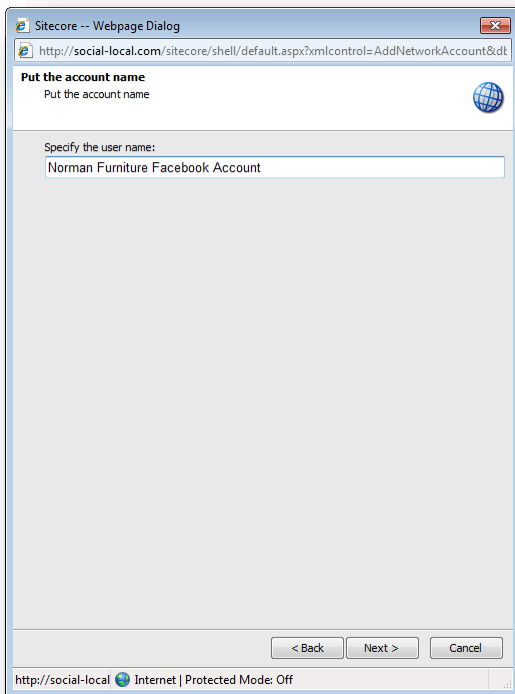
To create an account, you must allow all the required permissions.

In the **Page Selection** window, you specify which profile or page the module should update.



The module can post updates to the user's wall or to any assigned Facebook pages. Select where you want to post updates to in the drop-down list.

In the **Account Name** window, enter the name for this new network account.



## 4.4 Creating an Update Text

When you have configured the web application and created a Sitecore item for the social network account, you can create an update text.

To create and post an update text:

1. In Sitecore, in the **Content Editor**, select the item that you want to bind an update to.
2. In the **Content Editor**, on the **Publish** tab, click **Updates**. The **Updates** tab appears.
3. On the **Updates** tab, click **New Facebook Update** or **New Twitter Update**.
4. In the **Facebook message** or **Twitter message** dialog box:
  - 1) In the **Message** field, enter the text of update.
  - 2) Select the **Publish with item** check box, to post the update automatically when you publish the Sitecore item.
  - 3) In the **Accounts** field, select the social network accounts that you want to post the update to.
  - 4) Fill in the rest of the optional fields and click **OK** to close the dialog box.
5. Click **Publish** to post an update immediately.

To put the URL in the middle of the update text, in the **Twitter message** dialog box, use the `$link` token in the **Text** field and enter the URL in the **Link** field. This token does not work for Facebook messages in the current version of the module.

### Note

You can only post the same update to a social network once.



## Chapter 5

# Analytics Features

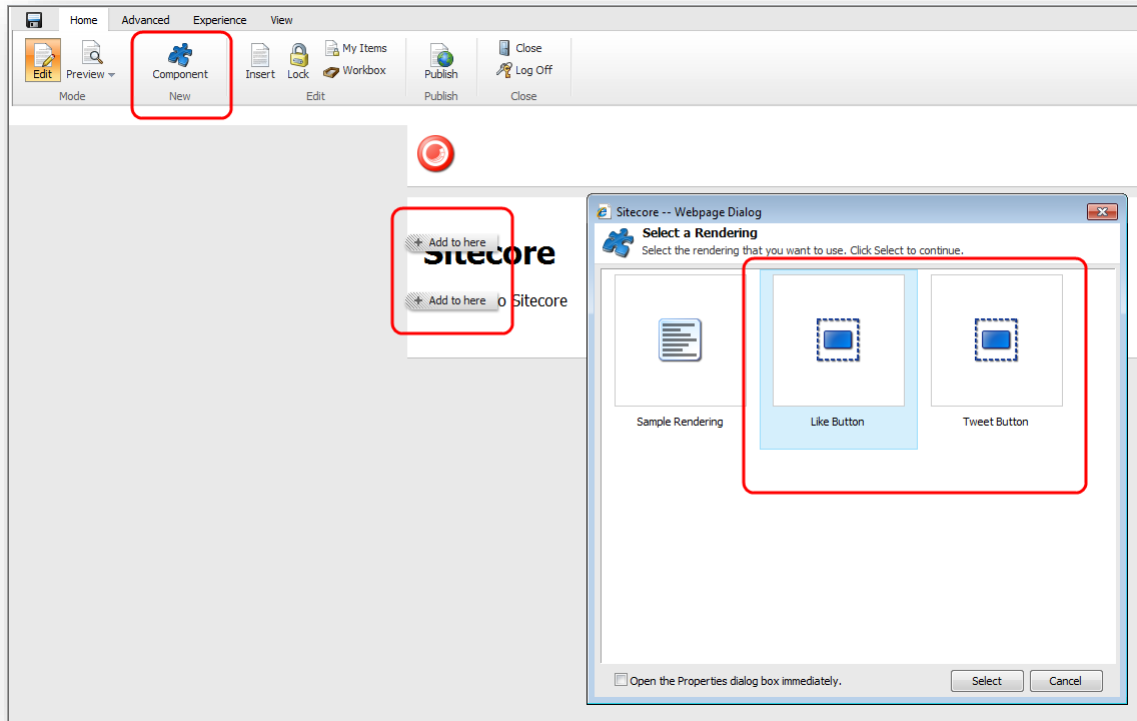
You can use the Sitecore Social Connected module and the Sitecore Digital Marketing Suite to personalize your website. The Social Connected module provides the Like and Tweet buttons as standard Sitecore controls that track visitors' activity .

This chapter contains the following sections:

- Putting Like and Tweet Buttons on the Web Page
- Tracking Like and Tweet Buttons Usage
- The Personalization Rule for the Social Connected Module

## 5.1 Putting Like and Tweet Buttons on the Web Page

You can put the Like (for Facebook) and Tweet (for Twitter) buttons on the website.



Sitecore provides them as standard controls. You can find the controls in the `sitecore/layout/sublayouts/social/sharing` folder.

To make these controls available for users to put on a webpage, you must enable the controls on the appropriate placeholder:

1. In the Content Editor, navigate to `sitecore/layout/placeholder settings` folder and select the placeholder item where the controls will be located
2. In the **Allowed Controls** field add the `Like Button` and the `Tweet Button` controls to the list of the selected controls.

For more information about the *Tweet* button, see the official Twitter documentation:

<https://dev.twitter.com/docs/tweet-button>

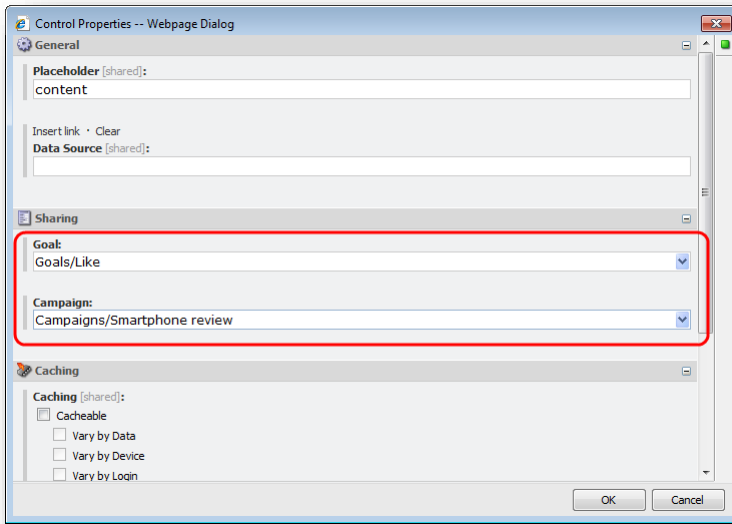
For more information about the *Like* button, see the official Facebook documentation:

<http://developers.facebook.com/docs/reference/plugins/like/>

## 5.2 Tracking Like and Tweet Buttons Usage

After you place the *Like* and *Tweet* buttons on the website, you can assign campaigns and goals to these buttons. Every time a website visitor clicks the button a specific goal or campaign is triggered and registered in Sitecore Analytics.

The Sitecore Social Connected module provides *Like* and *Tweet* goals for the *Like* and *Tweet* buttons. You can find these goals in the `sitecore/system/marketing center/goals` folder. Make sure to configure the *Like* and *Tweet* goals before using them.



For more information about goals and campaigns, see the manual *Marketing Operation Cookbook* on the SDN.

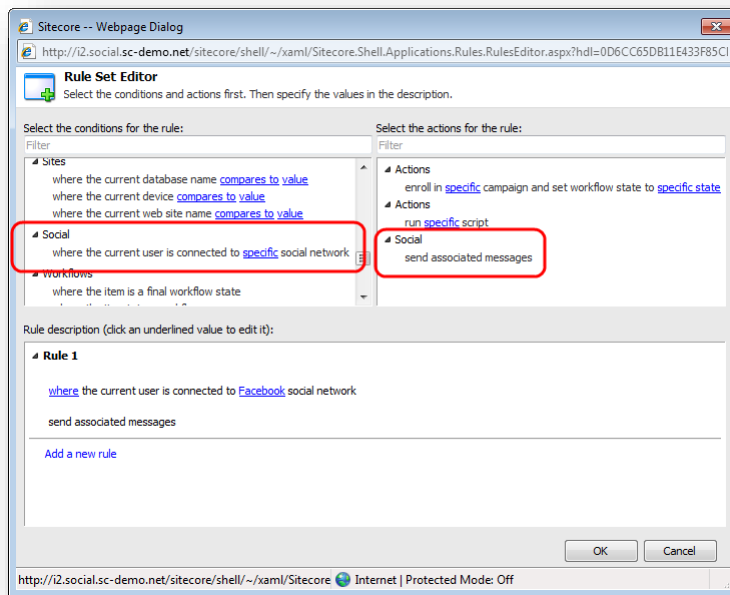
### 5.3 The Personalization Rule for the Social Connected Module

The Social Connected module adds one personalization condition and action to the Sitecore Rule Set Editor.

You can use this personalization condition and action with the Social Connected module to personalize the website.

The Sitecore Social Connected module adds the following condition and action

- *where the current user is connected to specific social network.*
- *send associated messages.*



For more information about personalization rules, see the manual *Marketing Operation Cookbook* on the SDN.

## Chapter 6

### Developer's Notes

The Sitecore Social Connected module uses 5 pipelines. This chapter describes the goals of the pipelines and their processors.

The chapter contains the following sections:

- The Social Connected Module Pipelines

## 6.1 The Social Connected Module Pipelines

The Sitecore Social Connected module uses 5 pipelines.

### 6.1.1 CreateSocialMessage Pipeline

The `CreateSocialMessage` pipeline is used when the module creates an update for a social network.

#### GetRoot Processor

The `GetRoot` processor creates a path in the content tree for the update item.

#### CreateItem Processor

The `CreateItem` processor creates two items under the update item: message and source items.

#### SaveData Processor

The `SaveData` processor saves update data from its fields.

### 6.1.2 ReadSocialMessage Pipeline

The `ReadSocialMessage` pipeline reads all the updates for the current item from the content tree and returns the updates collection.

#### GetMessagesRoot Processor

The `GetMessageRoot` processor reads the update item path for the current Sitecore item.

#### ReadMessages Processor

The `ReadMessages` processor reads all the updates for the current Sitecore item.

### 6.1.3 PublishMessage Pipeline

The `PublishMessage` pipeline posts the update to the social network.

#### ResolvePublishProvider Processor

The `ResolvePublishProvider` processor selects a publishing provider to post the update.

#### PublishMessage Processor

The `PublishMessage` processor posts the update to the social network.

### 6.1.4 BuildMessage Pipeline

The `BuildMessage` pipeline builds the update before it is posted to the social network. It returns the update object that is ready to be posted.

#### ResolveRenderer Processor

The `ResolveRenderer` processor returns the update builder for the source (goal or content).

#### BuildMessage Processor

The `BuildMessage` processor builds the update.

## ReplaceTokens Processor

The `ReplaceTokens` processor replaces tokens with their values.

### 6.1.5 MatchUser Pipeline

The `MatchUser` pipeline uses the data provided by the user to search for an existing user or creates a new user if no existing user is found.

#### PrepareUserData Processor

The `PrepareUserData` processor creates data for the user: username and email. This is based on the user basic data.

#### FindByEmail Processor

The `FindByEmail` processor uses the email information provided by the user to search for an existing user.

#### FindByDomain Processor

The `FindByDomain` processor searches for an existing user by domain.

#### FindByNetworkCredentials Processor

The `FindByNetworkCredentials` processor uses the network credentials (ID) to search for an existing user.

#### CreateUser Processor

The `CreateUser` processor creates a new user.

## Chapter 7

### Tips and Tricks

This chapter contains some tips and tricks for developers and administrators.

This chapter contains the following sections:

- How to Switch Language in the Sitecore Desktop
- How to Use the Error Handling Mechanism

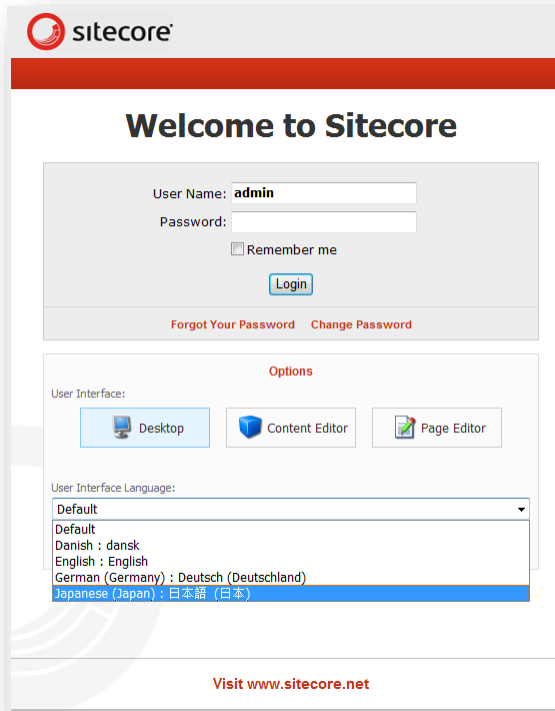


## 7.1 How to Switch Language in the Sitecore Desktop

The Sitecore Social Connected module provides several languages for the Sitecore Desktop UI.

To switch the language used in the Sitecore Desktop:

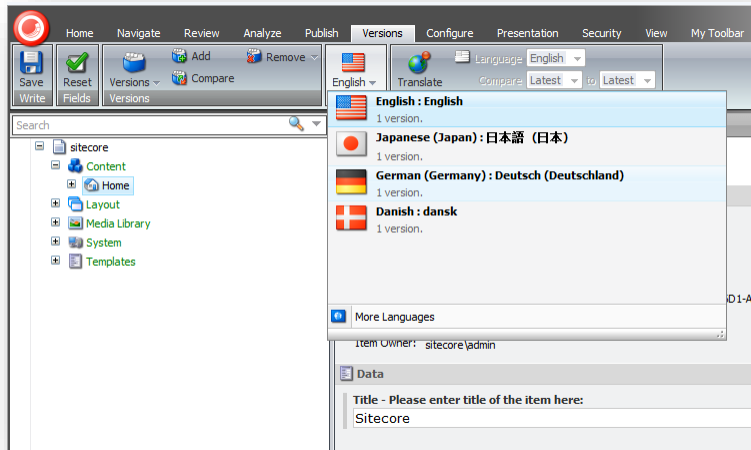
1. Go to Sitecore Desktop login page.
2. In the **User Interface Language** field, select the language you want to use.



The screenshot shows the Sitecore Desktop login page. At the top, there is a Sitecore logo and the text "Welcome to Sitecore". Below this is a login form with fields for "User Name" (containing "admin") and "Password", a "Remember me" checkbox, and a "Login" button. There are also links for "Forgot Your Password" and "Change Password". Below the login form is an "Options" section with three buttons: "Desktop", "Content Editor", and "Page Editor". Underneath is a "User Interface Language" dropdown menu. The dropdown is open, showing a list of languages: "Default", "Danish : dansk", "English : English", "German (Germany) : Deutsch (Deutschland)", and "Japanese (Japan) : 日本語 (日本)". The "Japanese (Japan) : 日本語 (日本)" option is currently selected and highlighted in blue. At the bottom of the page, there is a link to "Visit www.sitecore.net".

3. Log in to the Sitecore Desktop

4. In the **Content Editor**, on the **Versions** tab, in the **Language** group, select the language you want to use.



## 7.2 How to Use the Error Handling Mechanism

The Sitecore Social Connected module contains an error handling mechanism that you can use to display error messages both on the website and in the Sitecore Desktop. This mechanism lets you personalize the look and style of the errors that are displayed in your solution.

The error handling mechanism deals with the module features that directly cooperate with social networks. An error is typically comprised of a simple message and a detailed message. You can find the detailed message in the module log files under the folder `[installation_root]\data\logs`. The simple message is displayed in the UI.

You can use the language packages provided with the module to localize the simple error messages.

### 7.2.1 Error Handling in Sitecore Desktop

The features in the social module display error messages in different ways.

#### Errors when Posting Content Updates to Social Networks

If any errors occur when the module posts content updates to social networks, all the errors are registered as detailed messages in the log files.

#### Errors when Posting Updates to Social Networks on Goal Triggering

If any errors occur when the module posts updates when a visitor achieves a goal, the detailed error messages are registered in the log files, and the simple error messages are displayed as a tooltips in the UI.

#### Errors when Using Create Account Item Wizard

If any errors occur when you are using the Create Account Item wizard, the detailed messages are registered in the log files, and the simple error messages are displayed in the wizard UI.

### 7.2.2 Error Handling on the Website

Any errors that are caused by visitors on the website when they use the Social Connector log in buttons are also registered. The module registers detailed error messages in the log files.

To display the error messages to website visitors in the style used by your website, a developer should follow this procedure:

1. Create a control that looks for authentication errors in the query string and gets the GUID of the error.
2. Get the simple and detailed error message using exception manager and method `Sitecore.Social.Exceptions.ExceptionManager.Get(errorKey)`
3. Display this information to the website visitors in the same style as the website design.