



Salesforce.com Ideas Base Theme – Branding Design Guidelines and Best Practices

Salesforce.com Professional Services

August 16, 2009

Table of Contents

Contents

1. Revision History.....	3
2. Overview.....	4
2.1. Scope.....	4
2.2. References.....	4
2.3. Document Flow.....	4
3. About Ideas Base Theme.....	5
3.1. When to use Ideas Base Theme.....	5
3.2. What comes with the Ideas Base Theme (BT).....	5
4. Configuration, Setup and Branding Assets.....	6
4.1. Where to get Ideas Base Theme.....	6
4.2. SF Organization details for Ideas Base Theme.....	6
4.3. Customization Assets.....	6
4.3.1. Best Practices and Recommendations.....	7
4.4. Setting up the Force.com Ideas project.....	8
5. Incorporating Branding Assets.....	9
5.1. Planning	9
5.2. Develop.....	9
5.3. Configure.....	11
5.3.1. Configuring the Customer Portal.....	11
5.3.2. Sites.....	11
5.4. Test.....	11
5.5. Other information.....	12
5.5.1. Browser compatibilities.....	12

1. Revision History

Date	Version	Description	Author
7/16/2009	1.0	Created	Suchin Rengan
7/17/2009	1.1	Updated version	Suchin Rengan

2. Overview

2.1.Scope

This document provides detailed steps for branding an Ideas site using Salesforce.com Ideas Base Theme.

Salesforce.com Ideas Base Theme provides ability for branding and customization for building an Ideas Site.

2.2.References

This document references the following documents:

- Salesforce_ideas_theme.pdf – A guide that provides details on how to install and configure the Ideas Base Theme

2.3.Document Flow

The document has the following flow:

- About Ideas Base Theme
- Configuration, Setup and Branding Assets
- Incorporating the branding assets into the Base Theme

3. About Ideas Base Theme

This section provides some high level information regarding base themes. Specifically, it discusses when to use Base Theme, resources and timeline associated to create a branded Ideas site.

3.1. When to use Ideas Base Theme

- Branding customization needs are high and cannot be met using Customer Portal. For a quick start it is recommended that a customer use Customer Portal for a quick win and then use Base Theme for a full fledged roll out to production
 - It should be noted that you can create a branded site using Customer Portal, however, there are constraints in terms of how much branding and customizations can be done using a customer portal.
- When the out of the box Ideas pages do not meet needs put forth by the customer
- Additional functionality is required that is not already provided by Ideas application
- Integration requirements that need one to customize process using code rather configuration

3.2. What comes with the Ideas Base Theme (BT)

- BT comes with a flexible framework to allow a customer to customize an Ideas Site
- The framework comprises of the following
 - Code framework
 - Object framework
 - Configuration framework
- The code framework includes the following:
 - Visualforce pages for Ideas that allows developers to incorporate appropriate look and feel that supports their branding requirements
 - Visualforce components for Ideas that allows reusable UI components across the Visualforce pages
 - Visualforce controllers for Ideas associated to various Visualforce pages and incorporates logic to render functionality related to user action on these pages
 - Apex Classes that are reusable business logic components called from within the Visualforce controllers
 - Apex Test Classes for test coverage across various Apex classes, including the Visualforce controllers
- The Object framework leverages the SF Ideas and Ideas related objects. Besides the OOTB objects, Ideas Base Theme also includes an Ideas Setup object that has various setup fields for customizing the Ideas Base Theme. The setup fields allow for overriding various VF pages related to Ideas Base Theme
- The configuration framework comes with the following:
 - A Page Layout for entering the configuration related information related to the various setup fields that is part of the Ideas Base Theme
 - VF pages that can be overridden by cloned VF pages

4. Configuration, Setup and Branding Assets

This section discusses how to configure and setup Salesforce.com Ideas Base Theme. Also, this section discusses how to setup Ideas Project in Eclipse IDE

4.1. Where to get Ideas Base Theme

The Ideas Base Theme is developed and maintained by the SF Ideas Product team. To request Ideas Base Theme, contact the SF product team, led by Nori Yoshida (nyoshida@salesforce.com)

- You should receive a link for installing the Ideas Base Theme in your org
- A configuration and setup PDF, also mentioned in the Reference section

4.2. SF Organization details for Ideas Base Theme

A few things to remember:

- It is recommended that the Ideas Base Theme be installed in a Sandbox org rather than a Production org
- If your customization needs require login, then you need to have a Customer Portal license enabled in the org

Once installed, follow the configuration and setup PDF to configure the Ideas Base Theme. This includes overriding the pages etc. You do not need to have all the fields entered. You can enter them as you go about branding the site

You can create a branded Ideas site using the Ideas Base Theme and the following org structures/licences:

- Customer Portal: Allows for a completely branded site
 - Access to this site is a SF generated Portal URL
 - SSO using SAML
 - Users are the Customer Portal users having access to the VF pages and
 - No Vanity URL
- Platform
 - Access to this Ideas application is using SF username/ password
 - Cannot have a branded login page
- Sites and Customer Portal
 - Access can be using custom login pages
 - Vanity URL to allow for a customized URL access

4.3. Customization Assets

Following is a list of customization assets that should be requested from a customer:

- HTML Pages that depicts a branded site
 - These HTML Pages may include
 - Header component that has the Logo and the Ideas site tabs
 - Footer component
 - Left Navigation bar

- Home Page
- Ideas Pages including Popular, Recent, Top All Time and Comments
- My Profile Activity Pages
- Search Results
- If providing a custom login capability include the following
 - Custom Login Page
 - Change password
 - Forgot password
 - Logout page
- Cascading Style Sheets (CSS) that includes all the various branding elements referenced in the HTML
 - It is preferred to have a single CSS file that has all the branding elements referenced in the HTML pages
- Images and Logos
 - Company Logo
 - Ideas Vote Buttons (Promote, Demote)
 - Background styling and images
- Static Content (as needed)
 - Disclaimers
 - Terms and Conditions

Customers can reference the following sites to create these branding assets:

- DowJones Ideas site
- Dell IdeaStorm site
- Employee Ideas site (<http://employeeideas.force.com/>)

Customers can also begin from scratch and conceptualize how they want the site to look like. In such scenarios the likelihood of the timeline associated with branding their site will be more.

4.3.1.Best Practices and Recommendations

Following are some of the best practices and recommendations regarding the assets

- It is recommended that any styling including referencing an image should be encapsulated as a CSS element. Example,

```
#loginbox {
background-image: url(login_table_box.jpg);
background-repeat: repeat-x;
background-color: #262626;
}
```

Due to the Apex framework, it is easy to reference the CSS styling as a value in the **styleClass** attribute

- Have consistency in terms of having the structure for each of the Ideas page. What is specifically requested here is that the HTML structure itself is consistent across the pages such that they only differ in the content that is appropriate to each Ideas Page. This will greatly help with incorporating branding into the Ideas site
- Each HTML should represent enough content that will depict branding for various user scenarios. An example is that for an Ideas page, provide content that depicts branding for an Idea that has not been voted, an Idea that has been demoted, an Idea that has been promoted etc.

4.4. Setting up the Force.com Ideas project

Using the Eclipse IDE, create a Force.com project configured to the Ideas org. Once configured, perform the following steps:

- Refresh the project to fetch all the code related to the Ideas Base Theme
- Verify that the various components are being seen and there are no errors
- You should see the following project folders populated
 - Classes
 - This folder has the VF controllers, Apex Classes and Apex Test classes
 - Pages
 - This folder consists of the VF pages
 - Components
 - This folder comprises of the VF components
 - Static Resources
 - This folder comprises of the static files such as CSS and Images to be referenced in the VF pages and components
 - Triggers
 - This folder comprises of Apex triggers

Once these assets are verified, follow the steps in the next section to incorporate the branding assets into the Ideas Base Theme framework.

5. Incorporating Branding Assets

This section discusses in detail how to incorporate the branding assets into the Ideas Base Theme framework.

5.1.Planning

The first step is to plan for how to go about branding. Specifically, considerations have to be made on what pages have to be branded and customized. As mentioned before, the Ideas Base Theme framework is a set of modularized components designed with the intent of scalability and easy way to incorporate branding.

The core components within the Ideas Base Theme include the following:

- **ideasLayout page:** This VF page provides the overall layout of the ideas site. This is a core VF page that is invoked upon mostly every action by a User using the Ideas application. This page included and references other components such as Header, Left Navigation and
- **ideasSecondary component:** This VF component is responsible for the left navigation
- **ideasHeader component**
- **ideasHome Page**
- **ideasList Page**
- **ideasView Page**
- **ideasProfileActivity**

The best practice is to clone these above pages and components such that you can incorporate the branding accordingly.

As part of planning, one should first focus on the framework components such as the Layout, Header, Footer and Left Navigation that appear on all the ideas pages. Once branding is applied to these components, the branding to the content area of other pages would be performed in an iterative manner.

5.2.Develop

Just to emphasize that it is always a good practice to develop or incorporate branding by cloning the base theme asset. Coding directly into the original base theme asset may introduce complexities when upgrading.

As discussed in the Planning section, focus should lie on branding the framework components first. These include the following:

- **Layout page (ideaLayout.page):** The first step is to clone the ideasLayout page and define the overall layout and structure as defined in the HTML
- **Header component (ideaHeader.component):** The ideas Header component is referenced from the Layout page and encapsulates the header section for each page of the Ideas application. The header section comprises of the following:
 - **Company Logo**
 - **Ideas application tabs**

- **Left Navigation component (ideaSecondary.component):** This VF component is also referenced from within the Layout page and encapsulates the left navigation that appears in each of the ideas application pages
- **Footer component (ideaFooter.component):** This VF component is the footer section on each of the Ideas page and is referenced from within the Layout page

```
<apex:page controller="IdeaLayoutController"
standardStylesheets="false" showHeader="false" cache="{!doCache}"
expires="{!cacheMaxAge}">
    <apex:stylesheet value="{!$Resource.css_reset}" />
    <apex:stylesheet value="{!$Page.ideaCss}" />
    <apex:stylesheet value="{!customCssUrl}" />
    <apex:includeScript value="{!$Resource.js_prototype}" />
    <apex:includeScript value="{!$Resource.js_effects}" />
    <apex:includeScript value="{!$Resource.js_vote}" />
    <apex:includeScript value="{!$Resource.js_base}" />
    <apex:stylesheet value="{!URLFOR($Resource.IM_CSS_Styles,
'styles_alt.css')}" />

    <body>
    <div align="center">
    <div id="container_main">
        <c:IM_ideaHeader />

        <div id="content_area">

            <div id="content_area_main">
                <apex:outputPanel layout="none">
                    <c:IM_ButtonBanner />
                    <apex:pageMessages id="error" />
                    <apex:insert name="content" />
                </apex:outputPanel>

                <br class="clear" />
                <apex:outputPanel layout="none" style="display:none">
                    <apex:outputText style="display:none" value="{!
currentPageUrl}" />
                    <apex:outputText style="display:none" value="{!
currentPageString}" />
                    <apex:outputText style="display:none" value="{!
doCacheThisPage}" />
                    <apex:outputText style="display:none" value="{!
doCache}" />
                    <apex:outputText style="display:none" value="{!
cacheMaxAge}" />
                </apex:outputPanel>

            </div>

            <div id="content_area_left">
                <c:IM_ideaSecondary />
            </div>
            <div class="clear2"></div>

        </div>
    </div>
</div>
```

```
</div>
</div>
</body>

</apex:page>
```

Figure 1: An example of a branded ideas Layout Page

Also, when applying the branding, following steps are suggested:

- Clone the page/ component where branding needs to be applied
- Isolate the HTML snippet from a branded HTML page that needs to be incorporated in that page/ component
- Replace static text with dynamic Apex components (tags)

5.3.Configure

Once the pages and components have been cloned, it is required that you make appropriate configurations such that the Ideas application can be accessed. The configurations may include:

- Configuring the Customer Portal
- Configuring the Sites
- Validating configurations on the Ideas Setup page

5.3.1.Configuring the Customer Portal

You will need to configure Customer Portal if there is a need for a customized login experience. Additionally, you will need to enable Sites if customized login experience and Vanity URL is desired.

5.3.2.Sites

You will need to configure Sites when customers request for a vanity URL that allows for a customized access to their ideas application. Under sites, you can configure the vanity URL as well as Login details.

Following are some of the things you need to keep in mind when working with Sites:

- Only have the VF pages that can be accessible without the need to be logged in to be configured on Sites
 - These include the Login, Exception, Error, Forgot Password pages
- If the Ideas application is an internal only and needs a login, then configure the Sites home page with your custom Login page
 - Once a user is logged in, if you need user to be redirected to a custom home page, then hard code the startURL string in the ideaSiteLoginController class.

5.4.Test

This section provides details on how to test your Ideas application. Following steps need to be taken into consideration:

- Loading appropriate set of users to perform testing. Make sure that the users loaded belong to various profiles such that testing can be done taking into consideration the end user who is logged in

- User access and validation that a profile is able to access appropriate functionality and pages as desired
- Test to validate whether the rendered page matches the HTML templates (HTML assets sent by the customer)

5.5.Other information

5.5.1.Browser compatibilities

There might be incompatibilities with regards to rendering on various browsers, which would require applying a custom style sheet based on the browser being used.