

# Bazaarvoice Local implementation guide



## Disclaimer

Copyright © 2013 Bazaarvoice. All rights reserved.

The information in this document:

- Is confidential and intended for Bazaarvoice clients. No part of this document may be shared with anyone outside your company or your company-partners without prior approval from Bazaarvoice.
- Is provided “as is” without warranty of any kind either expressed or implied, including, but not limited to the implied warranties of merchantability, fitness for a particular purpose, or non-infringement.
- Is periodically updated to be in sync with the improvements and/or changes in our product offers.
- May be changed without notice. Some examples depicted herein are provided for illustration only and are not guarantees of a specific result. You bear the risk of using this document.

Bazaarvoice is not liable for any technical or typographical errors that might be inadvertently present in this document.

Bazaarvoice might have patents and/or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents, or legal rights to any other intellectual property in any Bazaarvoice service or product.

All Bazaarvoice brand and product names are trademarks or registered trademarks of Bazaarvoice in the United States and may be protected as trademarks or registered trademarks in other countries. All other product, service, or company names mentioned here are claimed as trademarks and trade names by their respective companies.

Contact us:

3900 North Capital of Texas Highway

Suite 300, Austin, Texas 78746

Toll-Free: (866) 522-9227 | Phone: (512) 551-6000 | Fax: (512) 551-6001

[www.bazaarvoice.com](http://www.bazaarvoice.com)

---

# Contents

<b>Overview</b> .....	<b>4</b>
<b>Creating and customizing feeds</b> .....	<b>5</b>
Local product feeds.....	5
Categories in the product feed.....	5
Products in the product feed.....	7
Local transaction feeds.....	9
<b>Establish dashboard access</b> .....	<b>11</b>
<b>Setting up review alerts</b> .....	<b>14</b>
<b>Configuring Local settings</b> .....	<b>15</b>
Create themes for local providers.....	15
Customize email templates.....	17
<b>Including local providers</b> .....	<b>20</b>
<b>Appendix: Single sign-on UAS code examples</b> .....	<b>22</b>

# Overview

Bazaarvoice Local helps your company gather and promote reviews of local service affiliates. Unlike third-party service provider listings, with Local you can control the user-generated content related to your service provider reviews and the environment in which the reviews are presented. This gives you instant access to authentic customer feedback, which you can use to determine which local providers are most effective.

This guide will walk you through the processes of implementing the Local product for your company and its local providers. The Local implementation processes use the same configuration tools as Bazaarvoice Conversations. Other Conversations guides can also be useful and are referenced throughout this guide. Any differences in Local implementation tasks from Conversations are noted in this guide.

You can find the most up-to-date guides in the Conversations Resources section of the Bazaarvoice Conversations website at <http://conversations.bazaarvoice.com>.

# Creating and customizing feeds

You will need to send two types of feeds to Bazaarvoice to get Local up and running: product feeds and transaction feeds. Your local providers will not need to provide feeds.

## Local product feeds

You must provide Bazaarvoice with an XML-based product feed that defines and describes both your local providers and the way you want to organize them for reporting purposes.

A "product" in the feed refers to a reviewable object. For Local, this means your local providers.

The product feed is divided into two main sections: categories and products, in that order.

- Categories define your local providers within the context of your organization. Categories allow you to generate reports that target specific regions or other areas of your company.
- Products define the reviewable aspects of your local providers.

---

### Note

You must follow additional rules for your product feed that are not enforced by the Bazaarvoice standard product feed XML schema. These rules allow Local to capture your providers and reporting structure accurately. For example, duplicate ExternalIDs will cause errors. You will need to manually review your feed to ensure these rules are being followed. You should also run your feed through an XML validator, which will catch any poorly formed XML code.

---

## Categories in the product feed

Categories define your providers in relationship to the structure of your organization. You can generate reports by category, so consider setting up categories to match your company's usual reporting structure. You might want to be able to view reports by region or by specialty, among other options.

For example, a lending company might organize its company by region (East Coast, West Coast), by city, and then by specific local providers.

The hierarchical structure would look like the following example:

Lending Company

- LC East Coast
  - Boston
    - LC Somerville
    - LC Back Bay
  - New York City

- Hartford
- West Coast
  - San Francisco
  - San Diego
  - Seattle

The following code shows how the structure of the East Coast local providers in the above example would be established in the categories section of a product feed:

```
<Categories>
  <Category>
    <ExternalId>ec</ExternalId>
    <Name>East Coast</Name>
    <Attributes>
      <Attribute id="BV_SUBJECT_TYPE"><Value>BV_LOCAL</Value></Attribute>
    </Attributes>
  </Category>
  <Category>
    <ExternalId>bos</ExternalId>
    <ParentExternalId>ec</ParentExternalId>
    <Name>Boston</Name>
    <Attributes>
      <Attribute id="BV_SUBJECT_TYPE"><Value>BV_LOCAL</Value></Attribute>
    </Attributes>
  </Category>
  <Category>
    <ExternalId>bos-som</ExternalId>
    <ParentExternalId>bos</ParentExternalId>
    <Name>LC Somerville</Name>
    <CategoryPageUrl>http://lenco.com/somerville</CategoryPageUrl>
    <Attributes>
      <Attribute id="BV_SUBJECT_TYPE"><Value>BV_LOCAL</Value></Attribute>
      <Attribute id="BV_SUBJECT_TYPE"><Value>BV_LOCAL_NODE</Value></Attribute>
    </Attributes>
    <ImageUrl>http://www.lenco.com/images/lcsomerville.jpg</ImageUrl>
  </Category>
  <Category>
    <ExternalId>bos-bbay</ExternalId>
    <ParentExternalId>bos</ParentExternalId>
    <Name>LC Back Bay</Name>
    <CategoryPageUrl>http://lenco.com/backbay</CategoryPageUrl>
    <Attributes>
      <Attribute id="BV_SUBJECT_TYPE"><Value>BV_LOCAL</Value></Attribute>
      <Attribute id="BV_SUBJECT_TYPE"><Value>BV_LOCAL_NODE</Value></Attribute>
    </Attributes>
    <ImageUrl>http://www.lenco.com/images/lcbbay.jpg</ImageUrl>
  </Category>
</Categories>
```

The categories section contains the following tags:

- **Category** -- A single level of the hierarchy. Each level must be represented with an entry surrounded by the **Category** tags. Higher-level parent categories must be listed before subcategories -- in the above example, "East Coast" must be listed first, followed by "Boston" and then "LC Somerville".
- **ExternalID** -- The unique identifier for each category listing. You cannot duplicate an ExternalID in the categories section.
- **ParentExternalID** -- The ExternalID for the parent category for a list. In the above example, "East Coast" is the parent category for "Boston".
- **Name** -- The name of the category that you want to display onscreen and in reports.
- **CategoryPageUrl** -- The web page for a local provider. Use this field only for categories that are also local providers.

- 
- Attributes -- Tags that identify the categories as being associated with Local.
    - In each category listing, you must include the following attribute:
      - `<Attribute id="BV_SUBJECT_TYPE"><Value>BV_LOCAL</Value></Attribute>`
    - For the local provider ("LC Somerville" in the above example), you must also include the following attributes:
      - `<Attribute id="BV_SUBJECT_TYPE"><Value>BV_LOCAL_NODE</Value></Attribute>`
  - imageUrl -- A logo, photo or other image to associate with a local provider. **Note:** The image cannot be larger than 150 x 100 pixels. Local providers will be able to override this specification via the Local dashboard.

---

**Note**

Use the BV\_LOCAL\_NODE attribute only for the most specific categories in the hierarchy. No categories can be nested below these.

---

## Products in the product feed

The products section of your product feed defines reviewable objects, and should include the name and URL for each local provider.

The following company structure is the basis for the example product code.

Lending Company

- LC East Coast
  - Boston
    - LC Somerville
    - LC Back Bay

The following example code shows how the Somerville local providers would be established in the product feed:

```
<Products>
  <Product>
    <ExternalId>LCB1234</ExternalId>
    <CategoryExternalId>bos-som</CategoryExternalId>
    <Name>LC Somerville</Name>
    <Description>LC Somerville offers premiere banking and lending services.</Description>
    <ProductPageUrl>http://lenco.com/somerville</ProductPageUrl>
    <ImageUrl>http://www.lenco.com/images/bos_som_branch.jpg</ImageUrl>
    <Attributes>
      <Attribute id="BV_SUBJECT_TYPE"><Value>BV_LOCAL</Value></Attribute>
      <Attribute id="BV_SUBJECT_TYPE"><Value>BV_LOCAL_NODE</Value></Attribute>
    </Attributes>
  </Product>
  <Product>
    <ExternalId>LCB1296</ExternalId>
    <CategoryExternalId>bos-bbay</CategoryExternalId>
    <Name>LC Back Bay</Name>
    <Description>LC Back Bay offers premiere banking and lending services.</Description>
    <ProductPageUrl>http://lenco.com/backbay</ProductPageUrl>
    <ImageUrl>http://www.lenco.com/images/bos_som_branch.jpg</ImageUrl>
    <Attributes>
      <Attribute id="BV_SUBJECT_TYPE"><Value>BV_LOCAL</Value></Attribute>
      <Attribute id="BV_SUBJECT_TYPE"><Value>BV_LOCAL_NODE</Value></Attribute>
    </Attributes>
  </Product>
</Products>
```

The products section of the feed contains the following tags:

- **Product** -- A reviewable item, such as a local provider or branch.
- **ExternalID** -- The unique identifier for each product. You cannot duplicate an ExternalID in the categories section.
- **CategoryExternalID** -- The ExternalID for the most granular category that contains the reviewable item. For branches, this might be the branch itself.
- **Name** -- The name of the reviewable item that you want to display onscreen and in reports.
- **Attributes** -- Tags that identify the categories as being associated with Local. In each category listing, you must include the following attributes:
  - `<Attribute id="BV_SUBJECT_TYPE"><Value>BV_LOCAL</Value></Attribute>`
  - `<Attribute id="BV_SUBJECT_TYPE"><Value>BV_LOCAL_NODE</Value></Attribute>`
- **ProductPageUrl** -- The web page that will include a link to the Write a Review form. In many cases, this is your company's locator page for local providers, or specific pages on your website for local providers.

### Special case: Individuals or services in the product structure

In rare cases, you might want customers to be able to review not only local providers, but individuals working for those providers or services at the providers. For example, you might want an insurance company's agents to be reviewed individually.

Using the previous example, in addition to being able to review LC Somerville, Lending Company would like customers to have the option of reviewing its customer service managers, to match the following structure:

Lending Company

- LC East Coast
  - Boston
    - LC Somerville
      - Peter Joshua

- Regina Cruikshank

You would include both the local provider and all reviewable individuals in the product feed as separate products, as shown in the following example code:

```
<Products>
  <Product>
    <ExternalId>LCB1234</ExternalId>
    <CategoryExternalId>bos-som</CategoryExternalId>
    <Name>LC Somerville</Name>
    <Description>LC Somerville offers premiere banking and lending services.</Description>
    <ProductPageUrl>http://lenco.com/somerville</ProductPageUrl>
    <ImageUrl>http://www.lenco.com/images/bos_som_branch.jpg</ImageUrl>
    <Attributes>
      <Attribute id="BV_SUBJECT_TYPE"><Value>BV_LOCAL</Value></Attribute>
      <Attribute id="BV_SUBJECT_TYPE"><Value>BV_LOCAL_NODE</Value></Attribute>
    </Attributes>
  </Product>
  <Product>
    <ExternalId>B23579</ExternalId>
    <CategoryExternalId>bos-som</CategoryExternalId>
    <Name>Peter Joshua</Name>
    <Description>Peter Joshua is a senior lending advisor at LC Somerville.</Description>
    <ProductPageUrl>http://lenco.com/somerville/pjoshua</ProductPageUrl>
    <ImageUrl>http://www.lenco.com/images/staff/pjoshua.jpg</ImageUrl>
    <Attributes>
      <Attribute id="BV_SUBJECT_TYPE"><Value>BV_LOCAL</Value></Attribute>
    </Attributes>
  </Product>
  <Product>
    <ExternalId>B23581</ExternalId>
    <CategoryExternalId>bos-som</CategoryExternalId>
    <Name>Regina Cruikshank</Name>
    <Description>Regina Cruikshank is a lending manager at LC Somerville.</Description>
    <ProductPageUrl>http://lenco.com/somerville/rcruikshank</ProductPageUrl>
    <ImageUrl>http://www.lenco.com/images/staff/rcruik.jpg</ImageUrl>
    <Attributes>
      <Attribute id="BV_SUBJECT_TYPE"><Value>BV_LOCAL</Value></Attribute>
    </Attributes>
  </Product>
</Products>
```

### Note

If you want to associate an image with a specific service or individual, use the **ImageUrl** tag, as shown in the above example. Images cannot be larger than 150 x 100 pixels.

## Local transaction feeds

You must send Bazaarvoice transaction feeds to trigger post-interaction emails (PIEs), which encourage customers to write and submit reviews. The transaction feeds for Local follow the same structure as transaction feeds submitted for PIEs in Bazaarvoice Conversations.

Upload the feeds to your account under **/ppe/inbox/bvlocal**. Bazaarvoice will transform the feeds to insert name and image values for the appropriate local service providers -- information the providers have submitted through the Local dashboard. At this point, all images are hosted by Bazaarvoice.

The PIE feed import will process the transformed feed and send email messages to the providers' customers to request reviews.

You can also enable follow-up PIE, which sends a second post-interaction email a specified number of days later than the first message. You can customize follow-up PIE settings from the Email Setup page in the Technical Setup

section of the configuration hub. Refer to the *Post-interaction email* guide for more information about configuring PIE settings.

The following example shows the general structure of a simplified XML interaction feed that exhibits all possible elements:

```
<?xml version="1.0" encoding="UTF-8"?>
<Feed xmlns="http://www.bazaarvoice.com/xs/PRR/PostPurchaseFeed/5.6">
  <Interaction>
    <TransactionDate>...</TransactionDate>
    <EmailAddress>...</EmailAddress>
    <UserName>...</UserName>
    <UserID>...</UserID>
    <Locale>...</Locale>
    <Products>
      <Product>
        <ExternalId>...</ExternalId>
        <Name>...</Name>
        <ImageUrl>...</ImageUrl>
        <Price>...</Price>
      </Product>
    </Products>
  </Interaction>
</Feed>
```

For more information about creating transaction feeds, refer to the Bazaarvoice *Data feeds guide*.

# Establish dashboard access

Bazaarvoice Local includes a single sign-on integration feature that your local providers will use to access the Local dashboard. This keeps them from having to create and use an additional login account. Local providers can begin using the dashboard as soon as you've set up single sign-on and provided your product feed to Bazaarvoice.

## Accessing the Local dashboard

Local providers can access the dashboard via a "jump page" on your web portal. The page explains Local to new users, stressing its value to them. It also includes a button that users click to access the Local dashboard.

Bazaarvoice will provide your company with HTML code for the jump page. Bazaarvoice recommends you modify it only to match your branding and basic site design -- for example, font and color changes.

Alternatively, Bazaarvoice can provide you with a "jump ad" to fit on your portal home page, which includes a brief explanation of the benefits and the button to access the dashboard.

---

### Note

When determining how you will integrate the jump page into your web portal, ensure it is easy for local providers to find after they've logged in to your site.

---

## Integrating single sign-on

When a local provider who is logged in successfully to your portal clicks a link to the Local dashboard, the single sign-on URL causes a pop-up window of the Bazaarvoice-hosted dashboard to appear, without providers realizing they're on a new website.

The single sign-on URL incorporates encryption values from a secret key shared between your company and Bazaarvoice. The provider's login for your company is matched up with account information for the Local dashboard, so providers can access their specific dashboard accounts.

Your technical team will need to integrate the code Bazaarvoice provides with your website to enable single sign-on for your providers.

## Single sign-on URL format

The single sign-on URL takes the following format:

```
https://bvlocal-dashboard.portal.bazaarvoice.com/home/index/?uas=<UAS value>&cn=<brand name>
```

Note the attributes in this URL:

- The URL must use HTTPS, not HTTP.
- The `uas` query parameter is a user authentication string -- an encoded or signed version of the Bazaarvoice authentication and identification of a user.
- The `cn` query parameter indicates your company's username in the Bazaarvoice client portal and configuration hub.

## Building the UAS string

The starting value of the UAS starts with several components indicating the properties of the incoming user. The string requires all the following components to form a strong signing algorithm.

- `un` – user name; an alphanumeric value that uniquely identifies a user, such as the name you use to log in to an account
- `dn` – display name; a display-friendly version of the user name, such as "Brenda Smith". Note that `un` and `dn` can be the same value.
- `em` – email address; the email address to use to contact the user
- `na` – nodes authorized; the ID of a category in your company's product feed that identifies the provider business accessible by the user. The Local dashboard only allows users to select businesses that are either indicated by the category ID itself, or providers under the category represented by the ID.

**Note:** The category ID must match the category ID you provide in the product feed. Refer to the "Categories in the product feed" section of this guide for details about submitting category IDs.

- `dt` – date; an ISO 8601 date/time format of the current time. Including the date in the string helps form a signing algorithm that is difficult to crack.

The values for each component are first URL/percent encoded, then chained together in a continuous string in the same manner as HTML forms are encoded. The string is then signed and encoded using the secret shared key.

For example, the following table associates sample values with each component.

Component	Value	URL encoded
<code>un</code>	<code>bsmith</code>	<code>bsmith</code>
<code>dn</code>	<code>Brenda Smith</code>	<code>Brenda+Smith</code> <b>OR</b> <code>Brenda%20Smith</code>
<code>em</code>	<code>brenda.smith@yourcompany.com</code>	<code>brenda.smith%40yourcompany.com</code> <b>OR</b> <code>brenda%20smith%40yourcompany%20com</code>
<code>na</code>	<code>sample-catid</code>	<code>sample-catid</code> <b>OR</b> <code>sample%2Dcatid</code>
<code>dt</code>	<code>2013-09-01T08:00:00.00Z</code>	<code>2013-09-01T08%3A00%3A00.00Z</code> <b>OR</b> <code>2013%2D09%2D01T08%3A00%3A00%2E00Z</code>

These values would form the following chained string (UAS):

```
un=bsmith&dn=Brenda%20Smith&em=brenda.smith%40yourcompany.com&na=sample-catid&dt=2013-09-01T08%3A00%3A00.00Z
```

## Encoding the string

Bazaarvoice will assign your company an exclusive secret key that your company will share with Bazaarvoice. The key is typically 32 characters long.

The following process uses the key to encode the UAS:

1. Generate the HMAC- MD5 hash of (the shared secret key + the UAS created above).

2. Hex encode the resulting string.
3. Hex encode the UAS.
4. Append the results of step 3 to the end of step 1.

The encoding process is also represented by the following formula:

```
hex_encode(HMAC-MAC(secret_key+UAS))+hex_encode(UAS)
```

Refer to the Appendix of this guide for UAS code examples in the following programming languages: JavaScript, C#, ASP, PHP, and Python.

### **Dashboard access for your company**

Your website administrators should have login access to the Local dashboard. Use the single sign-on encryption process detailed above to create login accounts for them.

---

#### **Note**

Set up your company's admin accounts with the highest level category ID in your structure, so they can view and access all local providers. If you want to create logins for regional or district managers, set up their accounts with the category ID that corresponds to their territories.

---

# Setting up review alerts

You will also need to configure the default review alerts for Local, so your local providers can receive messages about the Local program and the reviews on their services. You must provide Bazaarvoice with a spreadsheet of the local providers you want to include in your Local program, along with their email addresses and the default setting for their review alerts.

The spreadsheet must be in CSV format, which you can create from most spreadsheet applications, like Microsoft Excel or Open Office Calc. It should include the following columns of data for each local provider:

- **nodeName**: The category external ID for the local provider.
- **name**: The name of the local provider, as you want it to display in email messages.
- **email**: The email address to use for sending messages to the local provider
- **rating**: The number representing the rating threshold for the review alert. For example, enter "3" if you want the provider to receive alerts for reviews that are three stars or lower.
- **summaryReport**: A TRUE or FALSE value indicating whether you want to send summary reports to the local provider. TRUE indicates "yes."

The following example demonstrates what a local provider spreadsheet might look like.

nodeName	name	email	rating	summaryReport
bos-som	LC Somerville	bos-som@lendco.com	5	TRUE
bos-bbay	LC Back Bay	bos-bbay@lendco.com	5	TRUE

## Note

If you have already set up an alert for a local provider with the same email address as one in your spreadsheet, the alert configuration will be updated with the spreadsheet information.

# Configuring Local settings

You will need to set up and configure your Local dashboard, which local service providers can use to set up their review options.

powered by **bazaarvoice**

---

Home
Receive notifications
Put reviews on your site
Read & respond
UAS utility
Hub Admin
Encryption
Switch user ▾

---

## Welcome

Learn how it works

Reviews

Average rating

★★★★★ 4.1 | 25 reviews

Source: **BVLOCALTEST**

- [Read and respond to reviews](#)
- [Put reviews on your site](#)

Notifications

✔ Email alerts

- [Manage notifications](#)

Your profile

A-1 Certified Service Inc



- [Edit your profile](#)
- [See where it is used](#)

Your Bazaarvoice implementation team will work with you to integrate the dashboard with your existing website. You will need to select and design the options you want to offer local providers.

### Note

The configuration tools for Local are integrated with the Conversations configuration hub. You must implement the configuration hub before you can implement Local.

## Create themes for local providers

You can give your local providers multiple types of design options for the reviews they display on their websites. These options will be available in a dropdown menu on the Local dashboard, where providers can preview and select the style they prefer.

You'll need to log into the Conversations configuration hub to create the design options. For more information about the configuration hub, refer to the documents and videos on the Conversations Resources page of the Bazaarvoice Conversations website at <http://conversations.bazaarvoice.com>

Using the configuration hub, you can create a separate implementation for each theme you want to offer your local providers. An implementation is an instance of your Bazaarvoice ratings and reviews configuration that includes everything from the features you want to enable to the colors and fonts you want to use. You'll need to create a separate implementation for each style you want to offer your local providers. You'll also create implementations for review styles you want to include on your company's site.

You can create an implementation from the Manage Versions page of the configuration hub. This page appears immediately after you log into the hub. The page displays an icon for each implementation, including those used in staging and production. Click an icon to open that implementation and customize its design.

After you access the implementation in the configuration hub, use the Local Configuration page to name and activate the implementation as a theme available to your providers. From the left navigation column in the configuration hub, locate and click the Local Configuration page in the Technical Setup section.

The following controls are available on the Local Configuration page:

- **Show as Local Provider display style** -- Select **Enabled** to include the implementation in the list of themes on the Local dashboard, which providers can select to use on their websites. In addition, the control causes the full review to display in a pop-up lightbox, as shown in the following example.



- **Style Name** -- Enter the name of the theme to display in the list on the Local dashboard. If you don't specify a name, the list will display the name of the implementation.

You must also enable the **Read-Only Display** on the Ratings and Reviews Configuration page of the configuration hub, in the Review Display section. This ensures that the review display will not include links to pages where users can write a review. In addition, this control disables any domain restrictions, so local providers can add the display code to their websites. Local will not work correctly if this control is not enabled.

### Customize your themes

Use the Content Display page in the User Experience section of the configuration hub to set your review design preferences, such as color and font. You might want to involve your graphic design team in these tasks.

Refer to the *Content display style guide* and the *Ratings and reviews design recommendations* for helpful details about customizing review layout and design.

## Customize email templates

Use the Email Templates page in the Technical Setup section of the configuration hub to select and customize the templates you want to use for post-interaction email messages that customers receive, inviting them to leave reviews on your local providers.

## Email Templates mark complete ✓

Set up the content and appearance of notifications and post-interaction emails sent to your customers.

Changes to the email logo and post-interaction email template are effective only when you install this implementation to your **Main Site** (default) deployment zone.

### EMAIL LOGO

Select the logo image that you want to appear in emails. This logo will be used for all notifications and PIE emails sent. Click **Choose File** to upload an image (.png, .gif, or .jpg), and then click **Submit**.

The image cannot be more than 300 pixels wide.

Upload image Select Global to edit

No file selected.

Logo Link URL Select Global to edit

The following controls are available on the Local Configuration page:

From the Email Templates page, you can:

- **Upload the logo** you want to appear on email messages from the Local dashboard, in the welcome email to local providers, on the dashboard itself, and in the review display. The logo also appears on any standard email messages you send via the configuration hub, such as post-interaction email.

#### Note

Bazaarvoice recommends you upload a logo graphic that is 225 pixels wide and 50 pixels high, for optimal viewing in email messages and on the dashboard.

- **Enter the web address** you want to associate with that image, such as your company's home page.
- **Customize each template** by editing the subject line, message title and body text, and footer information. Do not edit or remove auto-populated fields in the text.

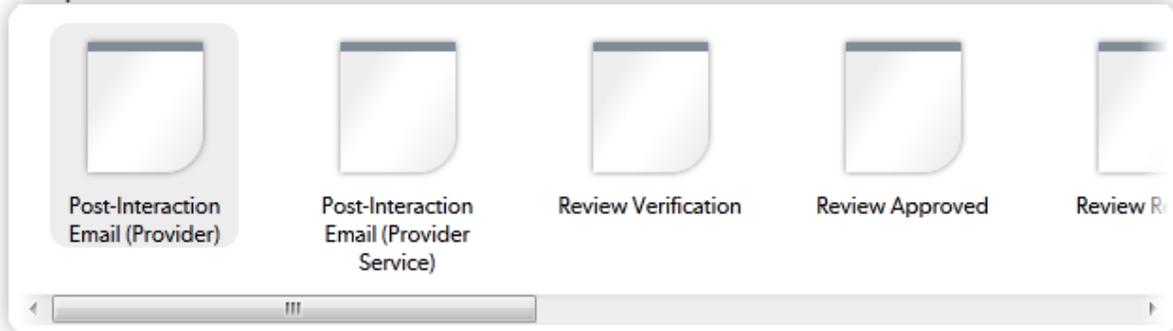
From the Email Logo section of the page, you can upload an image of your company's logo. To promote the authenticity of the review program, it's important to ensure you add your logo. It's used in local provider displays and on the Local dashboard, among other places. The image cannot be wider than 300 pixels.

From the Email Templates Editor section of the page, you can select and customize two email templates for Local:

- **Post-Interaction Email (Provider)** -- This template customizes email messages to customers asking them to review local service providers.

- **Post-Interaction Email (Provider Service)** -- This template customizes email messages to customers asking them to review specific services or individuals at a local provider -- for example, agents in an insurance office. Use this template only if you are requesting reviews of individuals or services associated with local providers.

## Templates



### Post-Interaction Email (Provider)

Preview 

Subject (40 characters maximum)

Insert... 

How was your experience with {{Provider name}}?

Refer to the *Working with email templates* document for more information about customizing email messages.

# Including local providers

It's critical to get your local providers interested and involved in Bazaarvoice Local. You'll want to communicate with them about the ways in which the program will benefit them, and let them know how the process works.

Some effective communication methods regarding Local include announcement emails, videos or webinars, online newsletters, local provider sales meetings or trainings, conveying details through territory or regional managers, the website or portal you use for providers, and mailed or printed materials. Bazaarvoice will provide webinars and documents you can use to help communicate information about Local. Contact Bazaarvoice for guidance and details about these materials.

## Local provider implementation

After you've created your themes, provided Bazaarvoice with an initial product feed, set up single sign-on, and configured review alerts, your company is ready to start requesting and gathering reviews. Local providers will receive alerts whenever reviews about them are published.

To prepare local providers for the review alerts, Bazaarvoice will send local providers a kickoff email message that includes details about the Local program and links to webinars and other information, as shown in the following example.

## Welcome! You've been automatically enrolled

Engage with Customers - Protect your Reputation - Start responding to customer reviews

- 1** We'll notify you when a customer leaves a review


- 2** You can respond to reviews by email or in your console (Responses publish on Basco.com)



Review your current notification settings  
If it all looks good no further action required

 Notifications: 5 Star Reviews and below

Edit

The kickoff email messages include links to the Local dashboard so local providers can customize their name and logo, set notification preferences, view and respond to approved reviews, and find the code to display reviews on their websites.

Providers don't need to use the Local dashboard to respond to reviews. When a provider receives a mobile-friendly email notification about a review, the provider can simply reply to the message with a response. That response then is put in the moderation queue and published when approved.

Your local providers can display reviews on their websites using a simple widget -- code available on the Local dashboard. They can use the dashboard to select one of the design themes you created in the configuration hub.

Refer to the *Getting Started with Local* guide for details about implementation for local providers.

# Appendix: Single sign-on UAS code examples

The following examples demonstrate the code used to encode the UAS in various programming languages. In these examples, the input (`uasStr`) is already built via the method in the “Building the UAS string” section of this guide.

## JavaScript

```
import javax.crypto.Mac;
import javax.crypto.spec.SecretKeySpec;
import org.apache.commons.codec.binary.Hex;

String sharedKey, stringToSign;
SecretKeySpec keySpec = new SecretKeySpec(sharedKey.getBytes(Charset.forName("UTF-8")), "HmacMD5");
Mac mac = Mac.getInstance("HmacMD5");
mac.init(keySpec);
String signature = new String(Hex.encodeHex(mac.doFinal((sharedKey + stringToSign).getBytes("UTF-8"))));
String hexString = new String(Hex.encodeHex(stringToSign.getBytes("UTF-8")));
return signature + hexString;
```

## C#

```
public static string SignAndEncode(string stringToSign, string sharedKey)
{
    MD5CryptoServiceProvider md5 = new MD5CryptoServiceProvider();
    UTF8Encoding utf8 = new UTF8Encoding();
    string signature = BytesToHexString(md5.ComputeHash(utf8.GetBytes(sharedKey + stringToSign)));

    string hexString = BytesToHexString(utf8.GetBytes(stringToSign));
    return signature + hexString;
}

private static string BytesToHexString(byte[] input)
{
    StringBuilder sb = new StringBuilder(64);
    for (int i = 0; i < input.Length; i++)
    {
        sb.Append(String.Format("{0:X2}", input[i]));
    }
    return sb.ToString();
}
```

**ASP**

```

<SCRIPT Language="JavaScript" runat="Server">
/*
 * A JavaScript implementation of the RSA Data Security, Inc. MD5 Message
 * Digest Algorithm, as defined in RFC 1321.
 * Copyright (C) Paul Johnston 1999 - 2000.
 * Updated by Greg Holt 2000 - 2001.
 * Adapted for use in ASP by Ian Brennan, Cyrona Software, 2001
 * (Any modifications covered by LGPL)
 * See http://pajhome.org.uk/site/legal.html for details.
 */

/*
 * Convert a 32-bit number to a hex string with ls-byte first
 */
var hex_chr = "0123456789abcdef";
function pjEnc_rhex(num)
{
  str = "";
  for(j = 0; j <= 3; j++)
    str += hex_chr.charAt((num >> (j * 8 + 4)) & 0x0F) +
          hex_chr.charAt((num >> (j * 8)) & 0x0F);
  return str;
}

/*
 * Convert a string to a sequence of 16-word blocks, stored as an array.
 * Append padding bits and the length, as described in the MD5 standard.
 */
function pjEnc_str2blks_MD5(str)
{
  nblk = ((str.length + 8) >> 6) + 1;
  blks = new Array(nblk * 16);
  for(i = 0; i < nblk * 16; i++) blks[i] = 0;
  for(i = 0; i < str.length; i++)
    blks[i >> 2] |= str.charCodeAt(i) << ((i % 4) * 8);
  blks[i >> 2] |= 0x80 << ((i % 4) * 8);
  blks[nblk * 16 - 2] = str.length * 8;
  return blks;
}

/*
 * Add integers, wrapping at 2^32. This uses 16-bit operations internally
 * to work around bugs in some JS interpreters.
 */
function PJEnc_add(x, y)
{
  var lsw = (x & 0xFFFF) + (y & 0xFFFF);
  var msw = (x >> 16) + (y >> 16) + (lsw >> 16);
  return (msw << 16) | (lsw & 0xFFFF);
}

/*
 * Bitwise rotate a 32-bit number to the left
 */
function PJEnc_rol(num, cnt)
{
  return (num << cnt) | (num >>> (32 - cnt));
}

```

```

/*
 * These functions implement the basic operation for each round of the
 * algorithm.
 */
function PJEnc_cmn(q, a, b, x, s, t)
{
  return PJEnc_add(PJEnc_rol(PJEnc_add(PJEnc_add(a, q), PJEnc_add(x, t)), s), b);
}
function ff(a, b, c, d, x, s, t)
{
  return PJEnc_cmn((b & c) | ((~b) & d), a, b, x, s, t);
}
function gg(a, b, c, d, x, s, t)
{
  return PJEnc_cmn((b & d) | (c & (~d)), a, b, x, s, t);
}
function hh(a, b, c, d, x, s, t)
{
  return PJEnc_cmn(b ^ c ^ d, a, b, x, s, t);
}
function ii(a, b, c, d, x, s, t)
{
  return PJEnc_cmn(c ^ (b | (~d)), a, b, x, s, t);
}

/*
 * Take a string and return the hex representation of its MD5.
 */
function PJEnc_calcMD5(str)
{
  x = pjEnc_str2blks_MD5(str);
  a = 1732584193;
  b = -271733879;
  c = -1732584194;
  d = 271733878;

  for(i = 0; i < x.length; i += 16)
  {
    olda = a;
    oldb = b;
    oldc = c;
    oldd = d;

    a = ff(a, b, c, d, x[i+ 0], 7, -680876936);
    d = ff(d, a, b, c, x[i+ 1], 12, -389564586);
    c = ff(c, d, a, b, x[i+ 2], 17, 606105819);
    b = ff(b, c, d, a, x[i+ 3], 22, -1044525330);
    a = ff(a, b, c, d, x[i+ 4], 7, -176418897);
    d = ff(d, a, b, c, x[i+ 5], 12, 1200080426);
    c = ff(c, d, a, b, x[i+ 6], 17, -1473231341);
    b = ff(b, c, d, a, x[i+ 7], 22, -45705983);
    a = ff(a, b, c, d, x[i+ 8], 7, 1770035416);
    d = ff(d, a, b, c, x[i+ 9], 12, -1958414417);
    c = ff(c, d, a, b, x[i+10], 17, -42063);
    b = ff(b, c, d, a, x[i+11], 22, -1990404162);
    a = ff(a, b, c, d, x[i+12], 7, 1804603682);
    d = ff(d, a, b, c, x[i+13], 12, -40341101);
    c = ff(c, d, a, b, x[i+14], 17, -1502002290);
    b = ff(b, c, d, a, x[i+15], 22, 1236535329);

    a = gg(a, b, c, d, x[i+ 1], 5, -165796510);
    d = gg(d, a, b, c, x[i+ 6], 9, -1069501632);
    c = gg(c, d, a, b, x[i+11], 14, 643717713);
    b = gg(b, c, d, a, x[i+ 0], 20, -373897302);
    a = gg(a, b, c, d, x[i+ 5], 5, -701558691);
    d = gg(d, a, b, c, x[i+10], 9, 38016083);
    c = gg(c, d, a, b, x[i+15], 14, -660478335);
    b = gg(b, c, d, a, x[i+ 4], 20, -405537848);
    a = gg(a, b, c, d, x[i+ 9], 5, 568446438);
    d = gg(d, a, b, c, x[i+14], 9, -1019803690);
    c = gg(c, d, a, b, x[i+ 3], 14, -187363961);
    b = gg(b, c, d, a, x[i+ 8], 20, 1163531501);
    a = gg(a, b, c, d, x[i+13], 5, -1444681467);
    d = gg(d, a, b, c, x[i+ 2], 9, -51403784);
    c = gg(c, d, a, b, x[i+ 7], 14, 1735328473);
    b = gg(b, c, d, a, x[i+12], 20, -1926607734);

    a = hh(a, b, c, d, x[i+ 5], 4, -378558);
  }
}

```

```

d = hh(d, a, b, c, x[i+ 8], 11, -2022574463);
c = hh(c, d, a, b, x[i+11], 16, 1839030562);
b = hh(b, c, d, a, x[i+14], 23, -35309556);
a = hh(a, b, c, d, x[i+ 1], 4, -1530992060);
d = hh(d, a, b, c, x[i+ 4], 11, 1272893353);
c = hh(c, d, a, b, x[i+ 7], 16, -155497632);
b = hh(b, c, d, a, x[i+10], 23, -1094730640);
a = hh(a, b, c, d, x[i+13], 4, 681279174);
d = hh(d, a, b, c, x[i+ 0], 11, -358537222);
c = hh(c, d, a, b, x[i+ 3], 16, -722521979);
b = hh(b, c, d, a, x[i+ 6], 23, 76029189);
a = hh(a, b, c, d, x[i+ 9], 4, -640364487);
d = hh(d, a, b, c, x[i+12], 11, -421815835);
c = hh(c, d, a, b, x[i+15], 16, 530742520);
b = hh(b, c, d, a, x[i+ 2], 23, -995338651);

a = ii(a, b, c, d, x[i+ 0], 6, -198630844);
d = ii(d, a, b, c, x[i+ 7], 10, 1126891415);
c = ii(c, d, a, b, x[i+14], 15, -1416354905);
b = ii(b, c, d, a, x[i+ 5], 21, -57434055);
a = ii(a, b, c, d, x[i+12], 6, 1700485571);
d = ii(d, a, b, c, x[i+ 3], 10, -1894986606);
c = ii(c, d, a, b, x[i+10], 15, -1051523);
b = ii(b, c, d, a, x[i+ 1], 21, -2054922799);
a = ii(a, b, c, d, x[i+ 8], 6, 1873313359);
d = ii(d, a, b, c, x[i+15], 10, -30611744);
c = ii(c, d, a, b, x[i+ 6], 15, -1560198380);
b = ii(b, c, d, a, x[i+13], 21, 1309151649);
a = ii(a, b, c, d, x[i+ 4], 6, -145523070);
d = ii(d, a, b, c, x[i+11], 10, -1120210379);
c = ii(c, d, a, b, x[i+ 2], 15, 718787259);
b = ii(b, c, d, a, x[i+ 9], 21, -343485551);

a = PJEnc_add(a, olda);
b = PJEnc_add(b, oldb);
c = PJEnc_add(c, oldc);
d = PJEnc_add(d, oldd);
}
return pjEnc_rhex(a) + pjEnc_rhex(b) + pjEnc_rhex(c) + pjEnc_rhex(d);
}

function url_encode (value) {
    return encodeURIComponent (value);
}
</SCRIPT>

<%
'Copyright 2001-2004 The Apache Software Foundation.
'Copyright 2006 Bazaarvoice, Inc.
'
'Licensed under the Apache License, Version 2.0 (the "License");
'you may not use this file except in compliance with the License.
'You may obtain a copy of the License at
'
'    http://www.apache.org/licenses/LICENSE-2.0
'
'Unless required by applicable law or agreed to in writing, software
'distributed under the License is distributed on an "AS IS" BASIS,
'WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
'implied. See the License for the specific language governing
'permissions and limitations under the License.

' Returns a String of hex characters that represents the encoded version of
' the specified user ID and uses the default values for other user attributes.
Function encodeUserID(userID, sharedKey)
    If IsNull(userID) Then
        encodeUserID = Null
    Else
        encodeUserID = signAndEncode(createUserToken(userID), sharedKey)
    End If
End Function

' Constructs a UserToken with a specified user ID. The UserToken must be
' signed and encoded with the signAndEncode method before it is usable.
Function createUserToken(userID)
    If IsNull(userID) Then
        createUserToken = Null
    Else

```

```

    Dim userToken, today
    userToken = ""
    today = Date()
    userToken = userTokenAdd(userToken, "date", Year(today) & "-" & Month(today) & "-" &
Day(today))
    userToken = userTokenAdd(userToken, "userid", userID)
    createUserToken = userToken
    End If
End Function

' Sets the user's nickname in their Bazaarvoice profile. This nickname
' is required if the Bazaarvoice application is configured to
' accept profile nicknames in the user authentication string instead
' of collecting nicknames in the submission form, and ignored otherwise.
Function userTokenAddProfileName(userToken, userProfileName)
    userTokenAddProfileName = userTokenAdd(userToken, "username", userProfileName)
End Function

' Sets the default value of the user's e-mail address in the
' Bazaarvoice submission form. This e-mail address will be ignored
' unless the Bazaarvoice application is configured to collect e-mail
' addresses in the content submission form.
Function userTokenAddEmailAddress(userToken, userEmailAddress)
    userTokenAddEmailAddress = userTokenAdd(userToken, "emailaddress", userEmailAddress)
End Function

' Sets or clears the "Staff" badge for the user in their Bazaarvoice
' profile. If this is not specified, the user's "Staff" badge is
' left unchanged and can be set via the Bazaarvoice Workbench UI.
' This value will be ignored unless the Bazaarvoice application is
' configured to accept badge values in the user authentication string.
Function userTokenAddStaffAffiliation(userToken, userIsStaff)
    If userIsStaff Then
        userTokenAddStaffAffiliation = userTokenAdd(userToken, "affiliation", "staff")
    Else
        userTokenAddStaffAffiliation = userTokenAdd(userToken, "affiliation", "none")
    End If
End Function

' Sets the number of days in the future that the signed user
' authentication string expires. After that date, the signed string
' will be considered invalid.
'
' By default, user authentication strings expire after one day.
' This method can be used to create authentication strings that last
' longer, which can be useful when creating pre-authenticated URLs in
' e-mails, for example. The maximum lifespan of a user authentication
' string is 365 days unless the Bazaarvoice application is explicitly
' configured otherwise.
Function userTokenAddExpiresAfterNDays(userToken, numberOfDays)
    ' signed user information must be valid for at least 1 day
    If numberOfDays < 1 Then
        numberOfDays = 1
    End If
    userTokenAddExpiresAfterNDays = userTokenAdd(userToken, "maxage", CStr(numberOfDays))
End Function

' Adds a URL-style name/value pair to the user authentication string.
Function userTokenAdd(userToken, key, value)
    If NOT(IsNull(userToken)) AND NOT(IsNull(value)) AND Len(value) > 0 Then
        If Len (userToken) > 0 Then
            userToken = userToken & "&"
        End If
        userToken = userToken & key & "=" & URLEncode (value)
    End If
    userTokenAdd = userToken
End Function

Function URLEncode (value)
    URLEncode = url_encode (value)
End Function

' Returns a signed, hex-encoded string
Function signAndEncode(string, sharedKey)

```

```

    signature = PJEnc_calcMD5(sharedKey + string)
    hexString = encodeHex(string)
    signAndEncode = signature + hexString
End Function

Function encodeHex(string)
    dim hexString, i
    hexString = ""
    for i = 1 to Len(string)
        hexString = hexString + Hex(Asc(Mid(string, i, 1)))
    next
    encodeHex = hexString
End function

'Uncomment it to test functionality
'sharedKey = "ABCDE"
'res = encodeUserID("ID12345", sharedKey)
'Response.Write (res)
'userToken = createUserToken("ID12345")
'userToken = userTokenAddProfileName (userToken , "MyName")
'userToken = userTokenAddEmailAddress(userToken , "testuser@bazaarvoice.com")
'userToken = userTokenAddStaffAffiliation(userToken , true)
'userToken = userTokenAddExpiresAfterNDays(userToken , 31)
'longRes = signAndEncode (userToken , sharedKey)
'Response.Write (longRes)

%>

```

**PHP**

```

hash_hmac('md5', $sharedKey.$stringToSign, $sharedKey).bin2hex($stringToSign);

```

**Python**

```

import hmac
import binascii
uas=hmac.new(sharedKey,sharedKey+stringToSign).hexdigest()+binascii.hexlify(stringToSign)

```



3900 North Capital of Texas Highway

Suite 300, Austin, Texas 78746

Toll-Free: (866) 522-9227 | Phone: (512) 551-6000 | Fax: (512) 551-6001

[www.bazaarvoice.com](http://www.bazaarvoice.com)