Multi-Tenant Settings

Maxim Noah Khailo

February 20, 2017

1 Problem

Currently many settings specific to a merchant such as their Stripe , Mailchimp, and shipstation keys are stored as environment variables to processes running in marathon. As we go multi-tenant this is clearly not acceptable.

2 Outline

This document provides use cases for implementing mult-tenant settings. Each use case is briefly written describing the Actor, pre-conditions, post-conditions, the main path and alternate paths.

This document does not describe how the system should look but describes what it should be capable of doing.

3 Use Cases

3.1 Onboarding

New Tenants will go through an onboarding flow allowing then to configure the baseline system and 3rd party integrations

3.2 Setting Modification

Tenants should be able to go back through all integrations in ashes and modify existing settings. We also need general Fox commerce related settings available.

3.3 User Settings

Table 1: Use Case "Selecting 3rd Party Integrations"

Action	**Selecting 3rd Party Integrations**
Actor	Tenant Admin
Precondition	Tenant has create an organization and has a master tenant admin account. Tenant Admin is logged in.
Postcondition	System will know which 3RD party integrations they want configured
Main Path	Admin selects from a list of available 3rd party integrations
Alternate Path	No integrations are selected. User goes strait to Admin interface.

Table 2: Use Case "Configuring Settings For an Integration"

Action	**Configuring Settings For an Integration**
Actor	Tenant Admin
Precondition	Admin has selected the integrations they want to configure.
Postcondition	The current integration in process is configured
Main Path	1. The available settings for the particular integration are displayed. 2. The appropriate GUI widget is shown for the setting type. 3. The user fills out all required fields. 4. The user presses the "Test" button. 5. Only with a good test can they continue to next integration.
Alternate Path	User selects "Skip". Integration is marked as inactive but available to edit in the future

Table 3: Use Case "Editing Existing Settings"

Action	**Editing Existing Settings**
Actor	Tenant Admin
Precondition	Tenant is logged into Ashes and has permissions to edit settings
Postcondition	Settings are modified and available immediately
Main Path	1. Admin can go to settings tab and see settings arranged in some grouping. 2. The groups they see are only those they are allowed to edit. 3. They click on a group and can see each individual settings. 4. Individual settings are displayed same way as during onboarding. 5. If the settings are an integration, they press the "Test" button. 6. Only if "Test"" succeeds can they save the changes. 7. If the settings are an integration, they can disable it. 8. If the setting group is not an integration, Then saving button is enabled.
Alternate Path	They don't want to change any settings, they can leave the screen

Table 4: Use Case "Adding an Integration"

Action	**Adding an Integration**
Actor	Tenant Admin
Precondition	Tenant is logged into Ashes and has permissions to add integrations
Postcondition	Integration is added if successfully goes through flow
Main Path	1. Admin can go to settings tab and see settings arranged in some grouping. 2. They click on button to add an integration. 3. They are presented with a list of integrations not already added. 4. They can then go through the "Configuring Settings For an Integration" use case.
Alternate Path	They don't want to add an integration, they can leave the screen

Table 5: Use Case "Adding a Settings Group"

Action	**Adding a Settings Group**
Actor	Tenant Admin
Precondition	Tenant has login credentials and has permissions to add a settings group
Postcondition	Settings group is added and is available for other admins to edit.
Main Path	1. Admin can add a settings group at the tenant level using an API 2. They can define setting keys and their types. Types influence how the settings are represented in the UI. 3. They can do CRUD operations on the group via the API.
Alternate Path	The setting group already exists, they can instead use CRUD API to modify.

Table 6: Use Case "User Edits Their Settings"

Action	**User Edits Their Settings**
Actor	Admin
Precondition	Admin is logged in and has permission to see their own settings
Postcondition	Settings are modified
Main Path	1. Admin can go to settings tab and see settings arranged in some grouping. 2. The first settings groups are settings specific to them. 3. They can modify any settings in their own groups. 4. In addition to their settings, if they have permission they can modify tenant and integration settings.
Alternate Path	They don't want to change any settings, they can leave the screen