Alteration guide to Spa Parts Depot E-Commerce App

V1.1

**Table of Contents**

Cover Page 1

Table of Contents 2

Introduction 3

List of Materials Needed 3

Getting Started 3

Making alterations to category pages:

Main carousel top images 6

Finishing Your Changes, Saving, and Syncing 7

Proper Syntax For Links 9

**Introduction**

This guide is meant to serve as an instruction manual for Spa Parts Depot employees to be able to make link and image changes to designated areas within the app. Although many of the controls are contained within the admin UI for most product and category options/controls, in some cases a change will be required to be made to code for the app. For these cases, instructions will be provide in this manual so as to minimize confusion about how these changes can be made to the code. Each section will contain the step by step instructions on how these alterations can be done and must be followed in their exact order. Failure to do so may results in undesired results. Please refer to the table of contents for page in which your desired process begins. Always refer to the list of materials needed and getting started sections BEFORE beginning work on any changes as failure to include these steps before beginning any changes will results in either your changes getting de-synchronized/lost or overwriting existing code, which can be disastrous. This guide is updated regularly and will contain up to date procedures for completing your changes. Please check your project repo for the most up to date version of this guide before starting with any changes to ensure you are performing the correct process. The version number in the cover page will provide the indication of what version of this guide you are using.

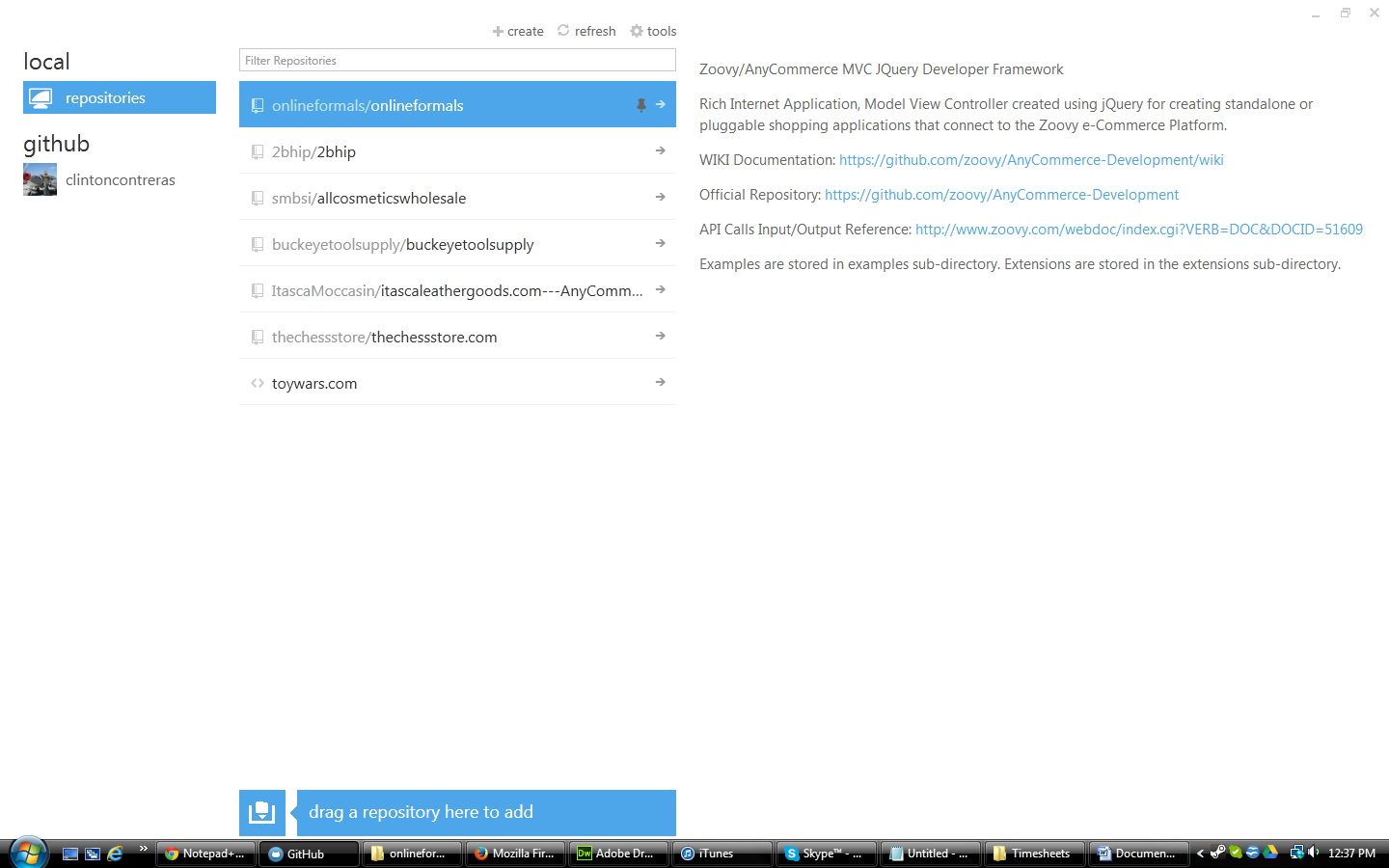
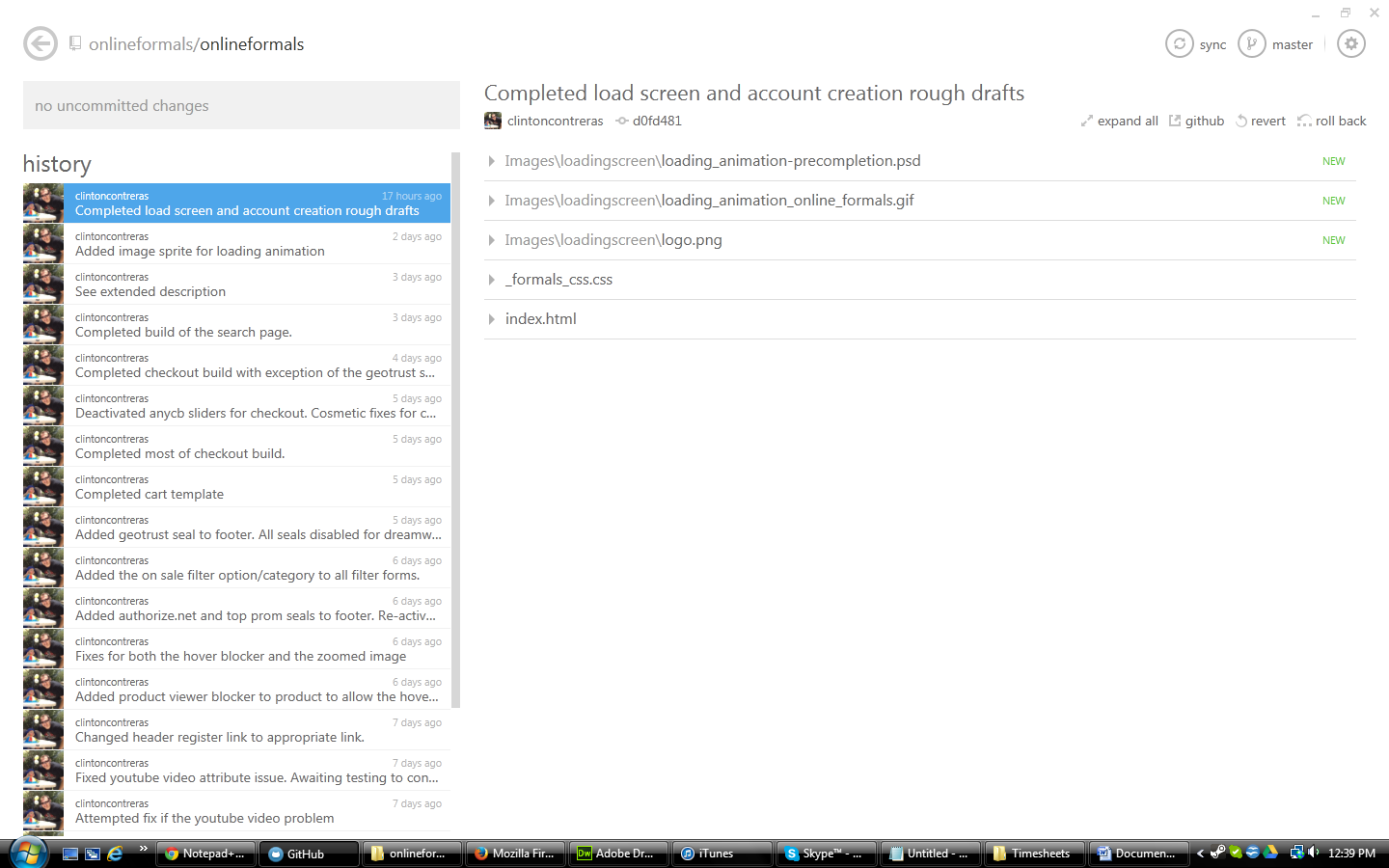
**List of Materials Needed**

To get started you will need the following items and software:

* Github client: can be downloaded from <http://windows.github.com/> for windows or <http://mac.github.com/> for mac users.
* An html/css text editor of your choice. If you do not have one or do not know what this is, I recommend downloading notepad++ as it is lightweight, and very easy to use and understand. You can download this for free at: <http://notepad-plus-plus.org/>
* Access to the Podio Workspace. This will be needed so that you can access the webhooks and sync your work from the github repository to your live site.

**Getting Started**

Once you have the above mentioned items, you are ready to get started. Before you start making any changes FOLLOW THESE STEPS FIRST!

1. Start up the Github client.
2. Your screen should look something like this:  
   
3. Click on the Spa Parts Depot repo. If it does not appear for you, please click your account icon under github, refresh the repo list, find Spa Parts Depot repo and clone it. If Spa Parts Depot is not appearing for you, please contact Zoovy so we can get that setup for you.
4. Your screen should now look something like this:  
   
5. Click the sync button. It should be with the rest of the buttons in the top right hand corner of the client.
6. Once the sync has completed, you should now have the most up to date version of the app.
7. Sign onto podio and make a post in the online formals workspace that you are accessing and changing the app code. This is done so that any dev that might be working in it at that time knows to sync their changes and halt any work they are doing so that you both don’t have de-synchronization issues.
8. Wait a few minutes and sync the repo one more time to make sure any changes the devs might have been working on are included in your most up to date version, and so you don’t unintentionally cause them to lose all the work they were working on.
9. Click the button in the top right hand corner that looks like a gear and click the option: open in explorer.
10. The online formals repo folders/files should now be visible and are ready to receive your changes. Refer to the correct section in the table of contents on how to make changes to your desired section.

**Category Page: Banner Images**

The process for changing category banner images is completely different than the above. To get started, you will need access to the admin UI/media library and you will need to open the file titled \_banners.json . To get started:

* Access the admin UI and go to the media library. Find the folder titled: T (this folder will change before deployment). Upload the banner image to this folder.
* Open up \_banners.json with notepad++. Follow the below instructions for making changes to this file:

WHEN ADDING A NEW CATEGORY BANNER, FOLLOW THESE INSTRUCTIONS TO AVOID BREAKING ANYTHING:

1. Ensure the category item above the one you are creating has a , after its associated closing } bracket.

2. Use the following layout when adding a new category:

"categoryname" : {

"src" : ["T/imagename.png"],

"alt" : ["alternatename"],

"title" : ["titlename"],

“height” : [“300”],

"bannersPerCat" : ["1"]

}

src field is the media library path to the banner image you want loaded in this area.

alt is the alternate name for the banner.

title is the title name for the banner image.

Height is the custom height of the banner. Make this match the actual height of the banner you uploaded to avoid any kind of issues with whitespace being added.

bannersPerCat is amount of banners you are attaching to that category. For now, this number will always be 1.

a. Add either "prodLink" : "productID" if the banner should link to a product or "catLink" : ".navcat" if linking to a category. You must use ONLY one or the other, not both. This is a required field to have so you must pick either one or the other. If you want to link to something other than a category or product page, contact your developer.

b. for the categoryname part, if the category is a subcategory, list the navcat path WITHOUT any .'s . For example: the category you want the banner to appear on is .designers.bg\_haute, you should name it as designersbghaute . Note that all spaces, underscores, periods and slashes have been removed. Forgetting to remove these and the banner will not appear on that category page.

c. All fields MUST have a value or some kind, be it "" or anything you want. failure to do so will cause a crash with this functionality.

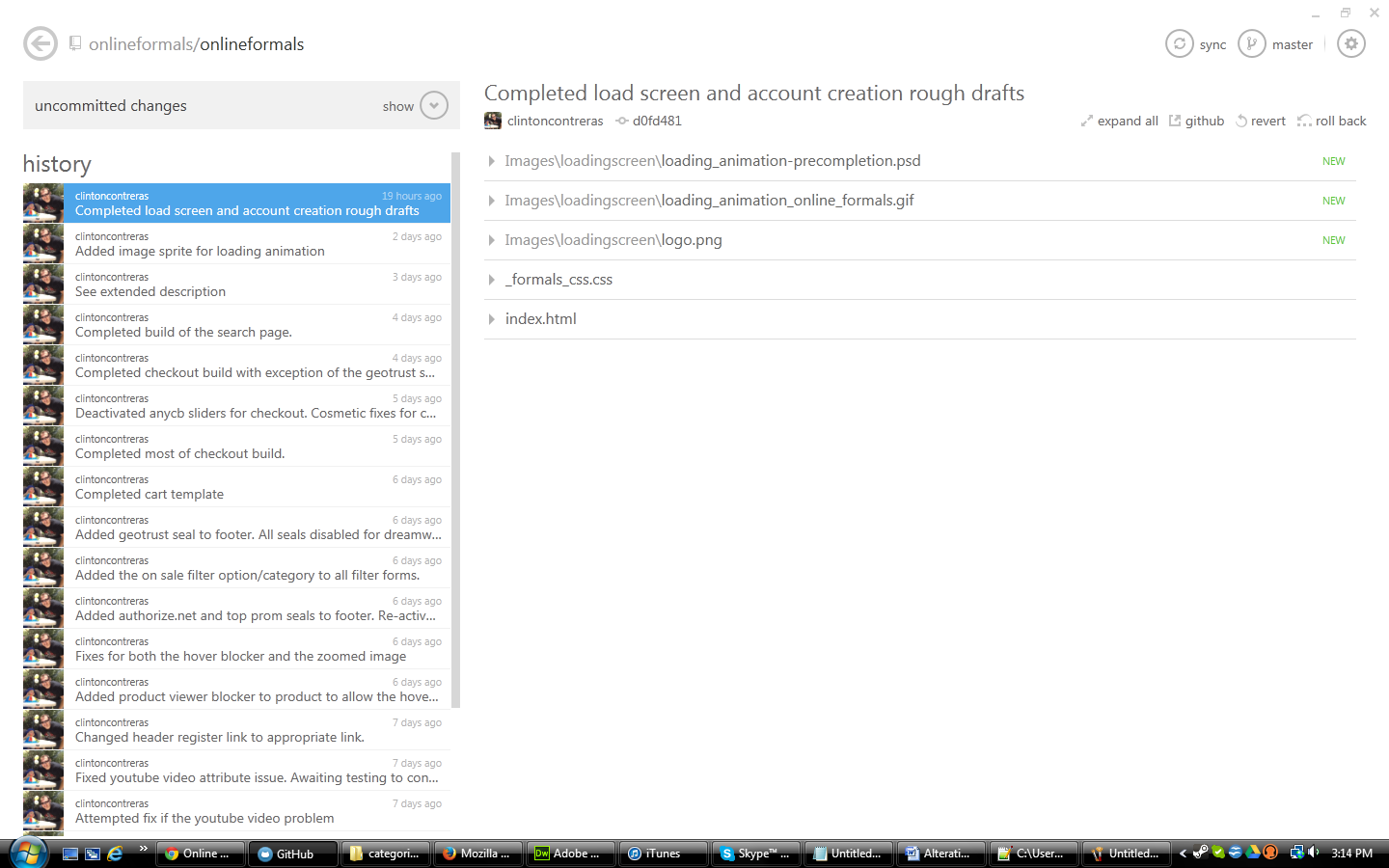
d. SAVE THE FILE.

e. Before you continue on, Go back to the project folder, find index.html, right click, open with firefox. Test EVERY category you added or modified. If you get a nasty looking error or the banner does not appear, there is something wrong with the syntax you used in banners.json. DO NOT SYNC THESE CHANGES YET! Compare your entries against any of the other entries that were there before you got started to figure out what exactly is wrong. The most common mistake is forgetting to add a , or having an extra , you didn’t need. Sadly enough, these can in fact break your entire banner system so please pay special care to place them where they need to be. Once all your banners are loading as they should, save the file again. This should complete your category banner process.

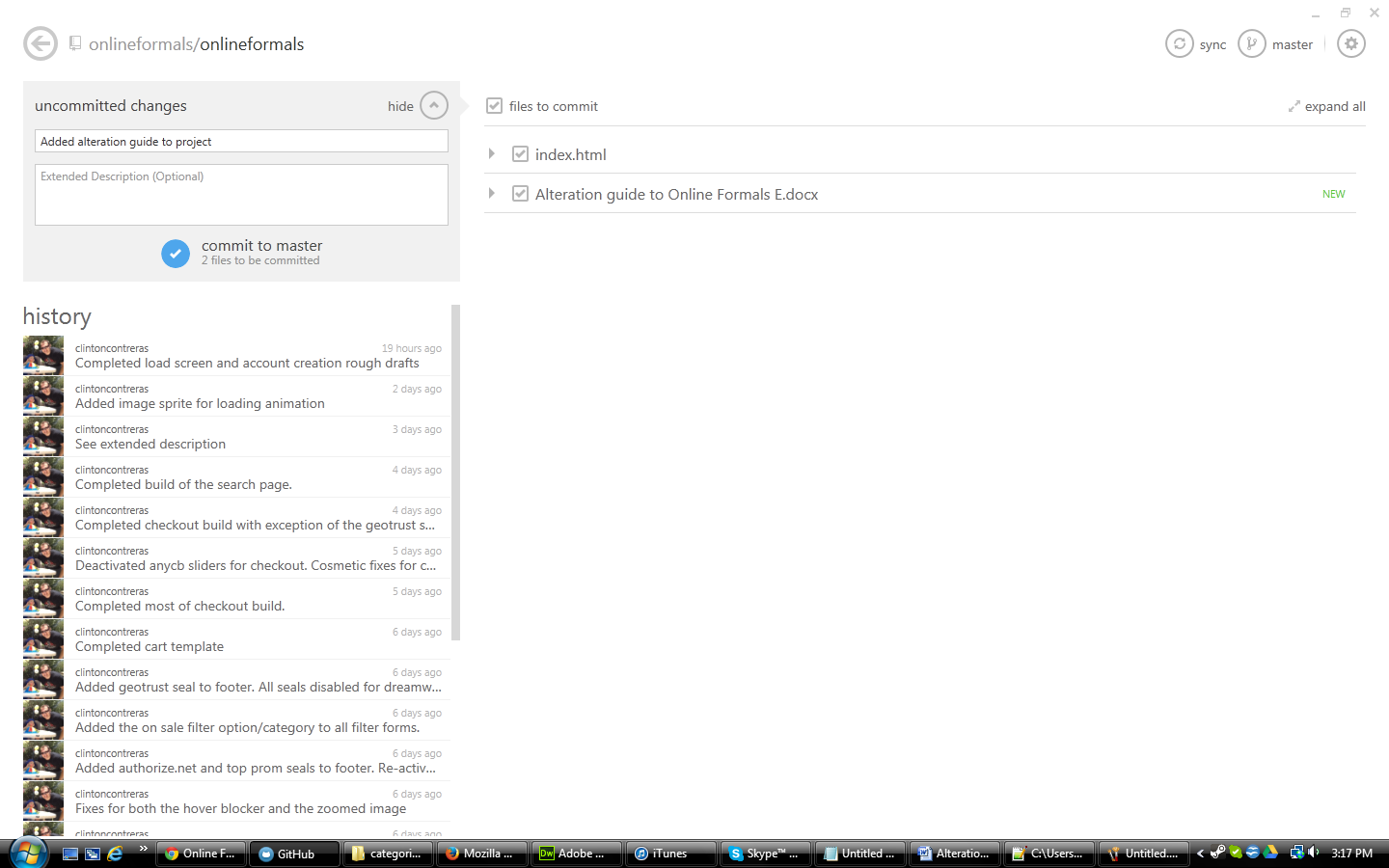
**Finishing Your Changes, Saving, and Syncing**

Once you have made all the changes you want to do, SAVE YOUR WORK! A simple file > save or ctrl + s will do. After you do that, follow these instructions to complete your changes:

1. Return to your Github client. If you closed it down, follow the getting started steps up to step 4. Your screen should now say it has uncommitted changes in the upper left hand part of the page. It should look something like this:



1. Click the show button. Your screen should now display a text entry area and allow you to name your changes, aka, your commit.
2. Enter the description/name of your changes in the commit message area. Once you have finished writing in this description, click the commit to master check button. Your screen should look something like this:



1. Once the commit finishes syncing, your changes have now been added to the master repo.
2. To push your changes to your live site:
   1. Instructions for this step will be added down the line, as the process for doing this has changed and isn’t set in stone just yet. Contact your developer for instructions on how this can be done.

**Proper Syntax For Links**

This section covers the various ways links can be configured to link to the appropriate page. The following is a quick breakdown of how to configure a link to redirect to any kind of page, including category pages:

**display a navigation category page:**

<a href='#top' onClick="return showContent('category',{'navcat':'[.category.safe.name]'});">A Category Name</a>

**cart page:**

<a href='#top' onClick="return showContent('cart',{});">Cart</a>

**product page:**

<a href='#top' onClick="return showContent('product',{'pid':'[pid]'});">A Product Name</a>

**company:**

The company pageType is responsible for loading and displaying content contained in the store 'profile' (a zoovy term). The profile contains policy information for a store, the email address, logo image, and more - the profile code is included in the config.js file. For company links, set the pageType to 'company' and use the list below to set the pageInfo:

<a href='#top' onClick="return showContent('company',{'show':'contact'});">Contact Us</a>

pageType 'company' supports the following standard 'pageInfo.show':

* 'show':'contact' : Contact Us
* 'show':'about' : Company Information
* 'show':'faq' : Frequently Asked Questions
* 'show':'shipping' : Shipping
* 'show':'return' : Return Policy
* 'show':'payment' : Payment Methods
* 'show':'privacy' : Privacy Policy

If you have any further questions regarding how these changes can be made, please contact your project developer to get further instructions.