*washingtonalpineclub.org* Administrator Guide

Table of Contents

[Membership Management: 3](#_Toc480191844)

[Roles: 3](#_Toc480191845)

[Subscriptions 3](#_Toc480191846)

[Groups and Capabilities 3](#_Toc480191847)

[Processing Orders: 5](#_Toc480191848)

[Orders that are in Processing status: 5](#_Toc480191849)

[Orders that are in Pending Payment status: 5](#_Toc480191850)

[Manually Entered Orders: 8](#_Toc480191851)

[Creating a Subscription: 8](#_Toc480191852)

[Creating an Order: 9](#_Toc480191853)

[Assigning an Order Made by One Member to Another Member: 12](#_Toc480191854)

[Updating Products – e.g. Basic Climbing Class: 15](#_Toc480191855)

[Each year the products that represent the classes offered need to be updated for the current year: 15](#_Toc480191856)

[Troubleshooting: 16](#_Toc480191857)

[Trouble signing-in, resetting password 16](#_Toc480191858)

[Checkout – Paying with Stripe 16](#_Toc480191859)

[Error Message: 16](#_Toc480191860)

[Action: 16](#_Toc480191861)

[Purchases – no “Add to Cart” button 16](#_Toc480191862)

[Symptom: 16](#_Toc480191863)

[Action: 16](#_Toc480191864)

[Wordpress – Force Upgrade 17](#_Toc480191865)

[Error Condition 17](#_Toc480191866)

[Action: 17](#_Toc480191867)

[Infrastructure: 17](#_Toc480191868)

[Wordpress 17](#_Toc480191869)

[Hosting 17](#_Toc480191870)

[Domains 17](#_Toc480191871)

[Test Sites: 17](#_Toc480191872)

[Themes: 18](#_Toc480191873)

[Active Theme: Responsive (Child) 18](#_Toc480191874)

[A Word about Child Themes 18](#_Toc480191875)

[Plugins: 18](#_Toc480191876)

[Subscription, Group, Checkout Functionality 18](#_Toc480191877)

[Additional Site Functionality 19](#_Toc480191878)

[Utilities 19](#_Toc480191879)

[Custom 19](#_Toc480191880)

[Custom Code: 19](#_Toc480191881)

[Plugins 20](#_Toc480191882)

[Smartwaiver Integration: 20](#_Toc480191883)

[API Key 20](#_Toc480191884)

[Autotagging Waiver 20](#_Toc480191885)

[Wordpress shortcode text 20](#_Toc480191886)

[Retreiving 21](#_Toc480191887)

[Getting list of waiver types 21](#_Toc480191888)

[Getting a specific membership waiver by tag 21](#_Toc480191889)

[Parsing the xml 22](#_Toc480191890)

[Viewing the PDF 22](#_Toc480191891)

[PHP Sample Code 22](#_Toc480191892)

[Custom Shortcode 22](#_Toc480191893)

# Membership Management:

## Roles:

Roles are the basic concept used by Wordpress to control who can do what on a site. Six pre-defined roles exist: Super Admin, Administrator, Editor, Author, Contributor and Subscriber. On the WAC website roles are used to distinguish administrative permissions (whereas *groups* are used to control access to member content – more on groups can be found under Groups and Capabilities, further below):

Roles usage:

* Administrators – a few technical individuals with full administrative control of the site.
* Editors – can manage all content, create/modify/delete pages, change menus, etc. Note that deleting and modifying pages used for payment and membership management could make the site unusable, requiring a restore. Therefore Editors should have a care when working with pages. We’ll have a few Editors.
* Authors – can post content. We can have many of these, such as class co-chairs, membership chairs, etc.
* Subscribers – by default members will be subscribers.
* There’s plenty of info available on how roles work at the [Wordpress Codex](http://codex.wordpress.org/Roles_and_Capabilities).

## Subscriptions

* Memberships are managed as subscriptions using the Woocommerce Subscriptions module. I’ve defined subscription products which address new and renewing members. These subscriptions are variable, allowing the user to choose a 1-5 year term. When a person buys one of these subscriptions they are automatically added to the “Active Members”, and when their subscription expires they are removed from the group (more on this later). I’ve also created a subscription product for annual cabin use.

## Groups and Capabilities

* To control access to content, and to track and report on membership status, the solution uses the Wordpress Groups plugin, and connects this to the Woocommerce products and subscriptions using the Woocommerce Groups plugin.
* Users can be assigned to groups, and groups can be assigned capabilities. In this way the authority to read and update pages can be managed. For example,

1. A membership subscription is sold to an individual,
2. That individual is added to the “Active Members” group,
3. The “Active Members” group has a capability called “Member\_Read\_Post”,
4. Pages with member-only content are defined as requiring the “Member\_Read\_Post” capability, with the result being that this user will now be able to see those pages. Note that this access control and visibility extends to menues that reference restricted pages—i.e. unless you are a member of a group with sufficient capabilities these you won’t even see menus referencing these pages.

* Currently defined groups and capabilities:

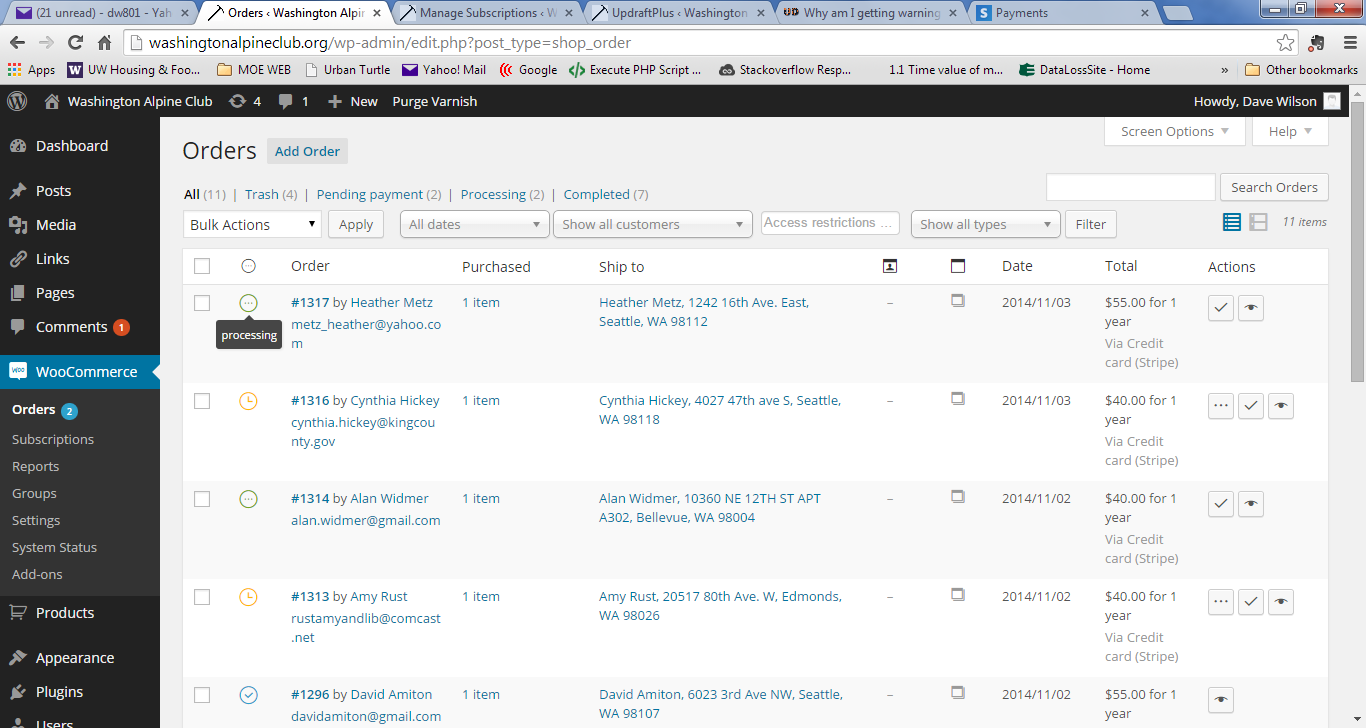
|  |  |  |
| --- | --- | --- |
| **Group** | **Capability** | **Users in Group** |
| Registered | Registered\_Read\_Posts | Everyone with a signin. This includes members (active or inactive) as well as those who have applied for membership. |
| Member Waiver | Waiver\_Read\_Post | Used as an indicator of who has signed a WAC waiver. |
| Annual Cabin User | No capabilities assigned to this group. | When users pay annual cabin dues they are added to this group, and removed when those dues expire. |
| Members | Member\_Read\_Post | All active and inactive members. This is a parent of Active and Inactive Members. |
| Active Members | Member\_Read\_Post | When a user pays their membership dues they are added to this group. When their membership expires they are removed from this group and added to group Inactive Members. |
| Inactive Members | Member\_Read\_Post | This group is for members whose dues are not current. |

* Note that, as currently defined, Active and Inactive members have the same capabilities but that the structure allows for differentiated capabilities should we decide we need that.
* Future: it would be easy to establish groups to control updates and content for classes, students, and instructors – e.g. Basic Class Cochairs, Basic Class Students, Basic Class Instructors.

# Processing Orders:

## Orders that are in Processing status:

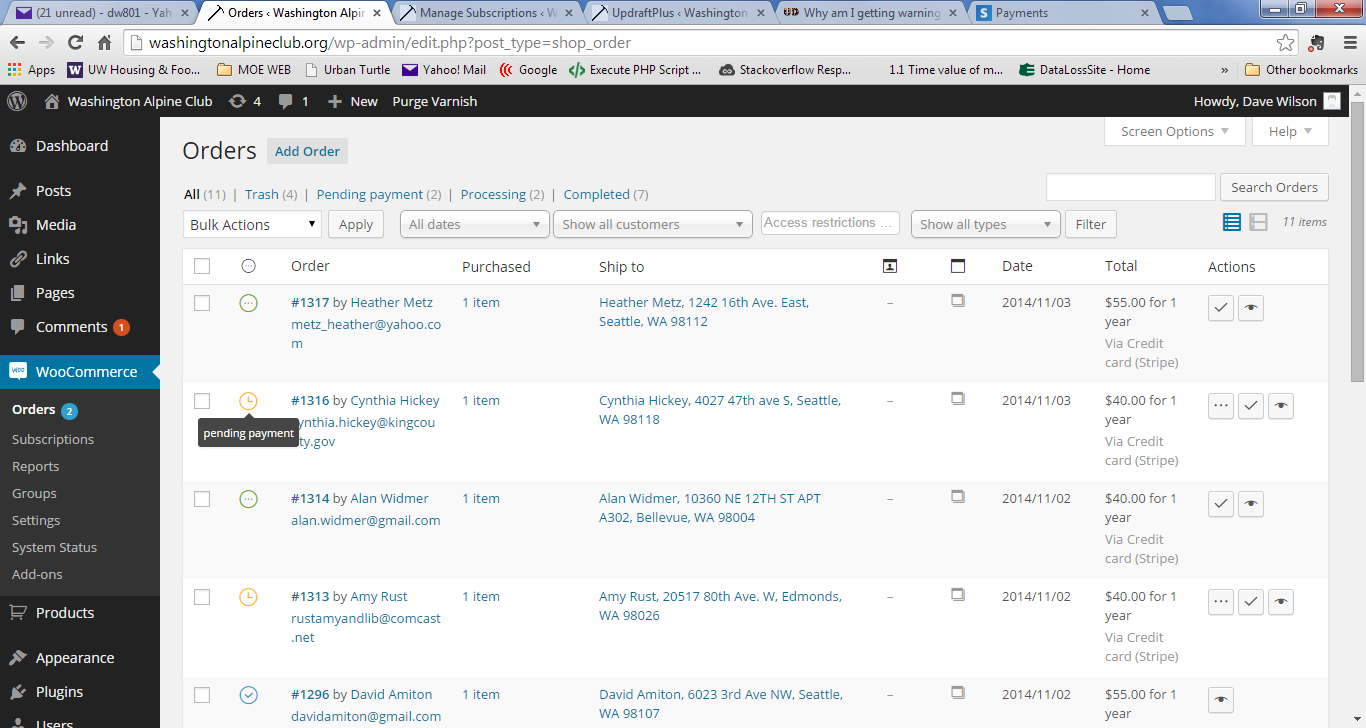
Just click the checkmark to process…



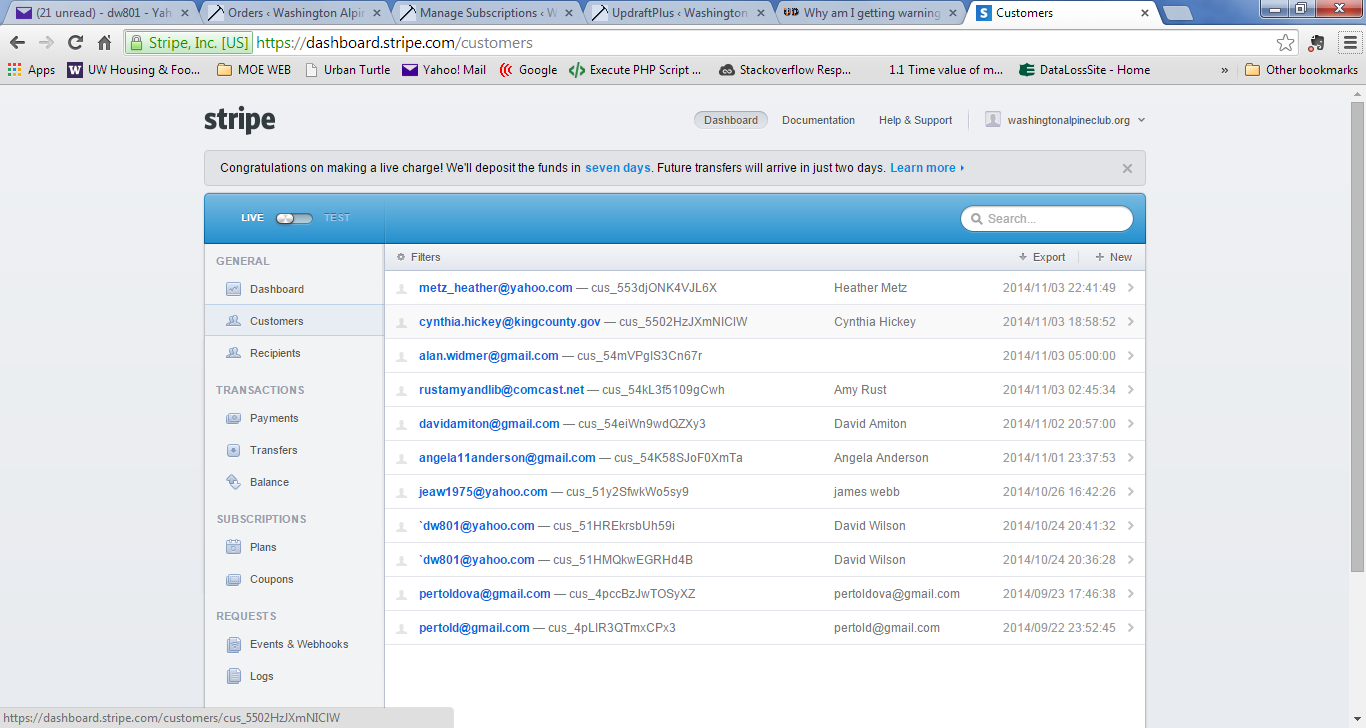
## Orders that are in Pending Payment status:

Occasionally payments don’t go through, so one needs to manually complete the payment by entering it in the Stripe console.

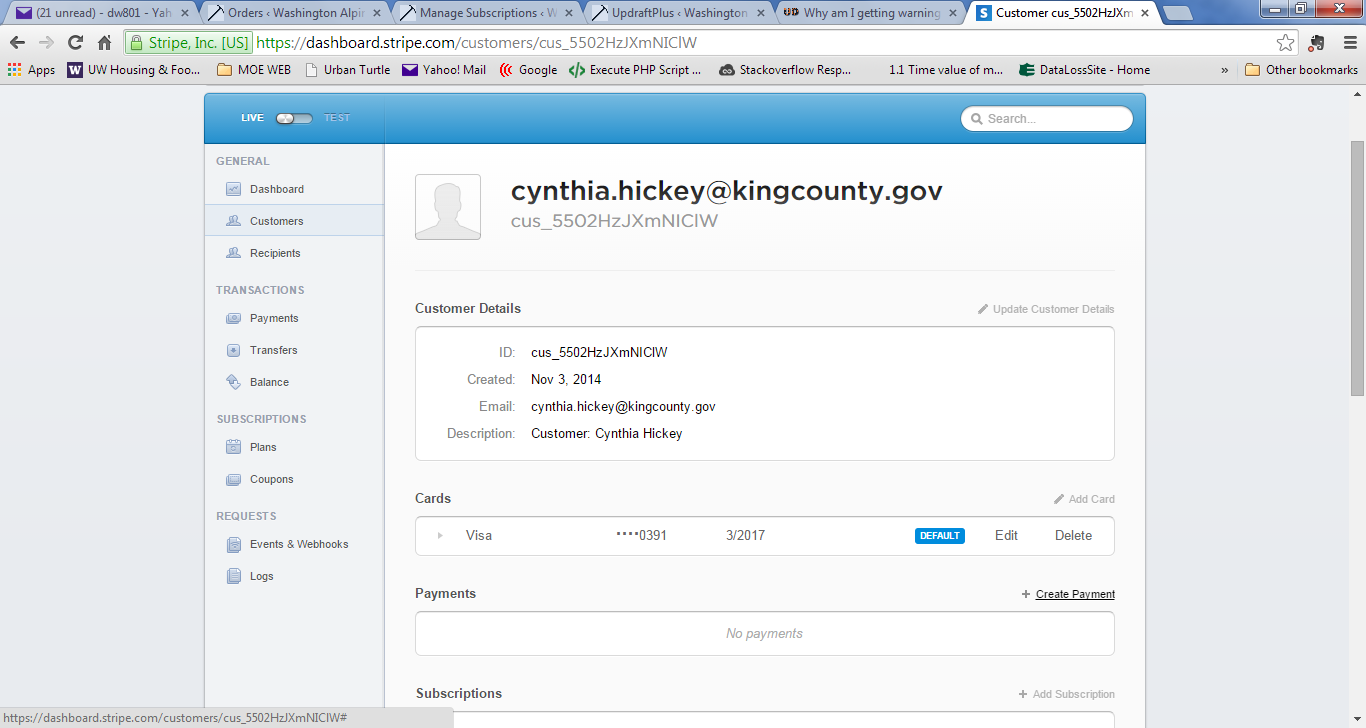
Note the Pending Payment status…



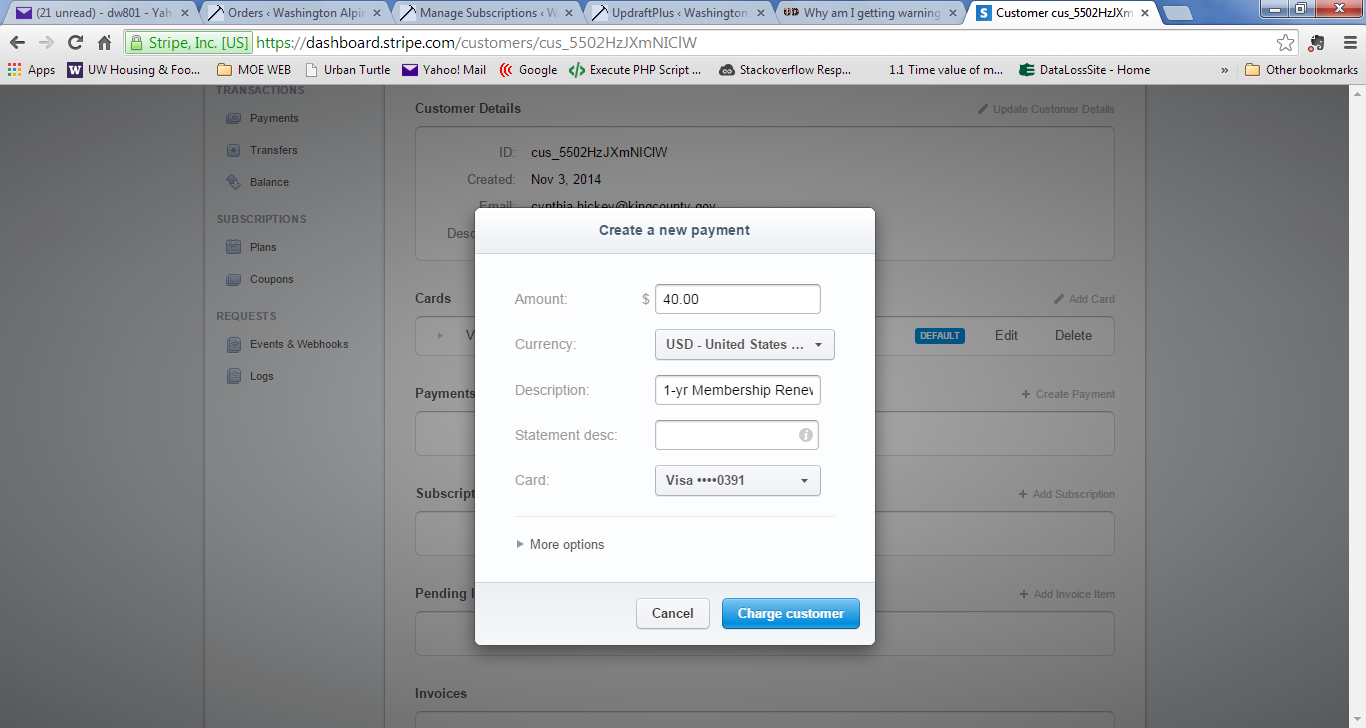
Sign into Stipe, find the customer…

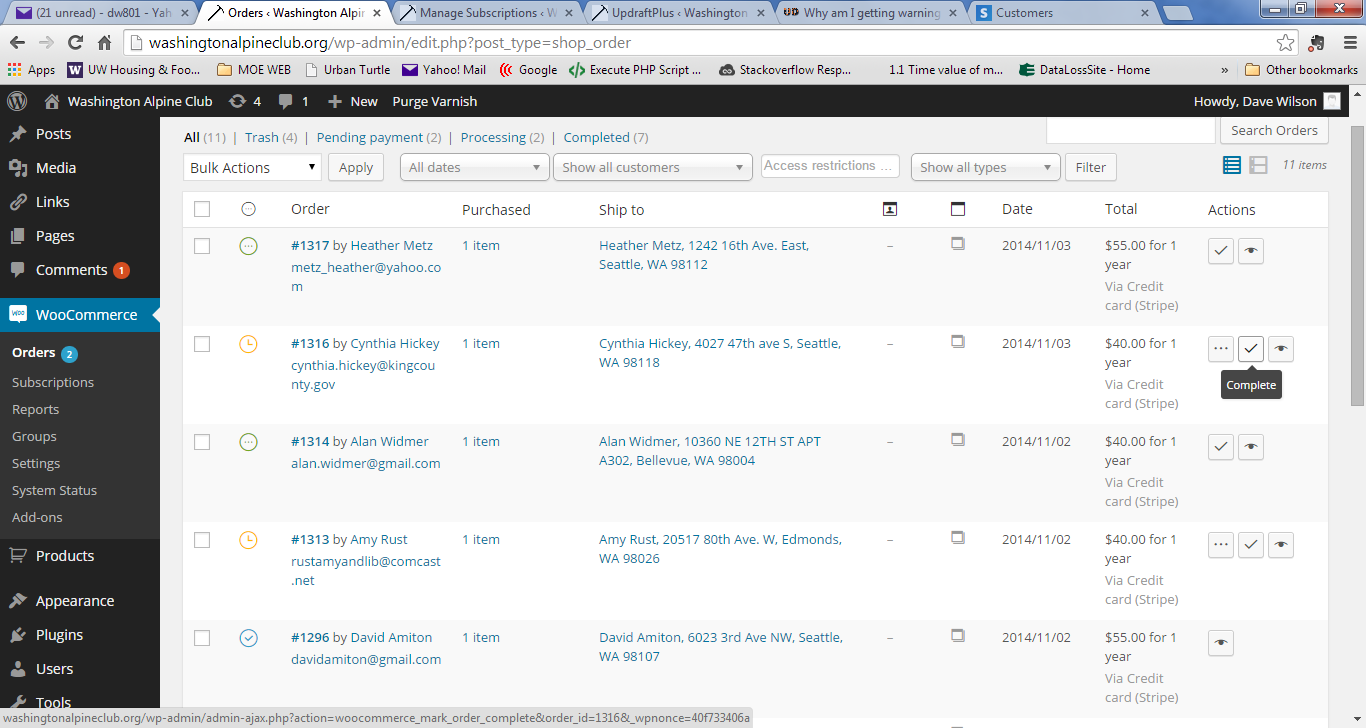


Verify the payment hasn’t gone through – not below that there are no payments processed…



Enter the payment amount and description…



Finallly, with that all done mark the order complete in Woocommerce.

## Manually Entered Orders:

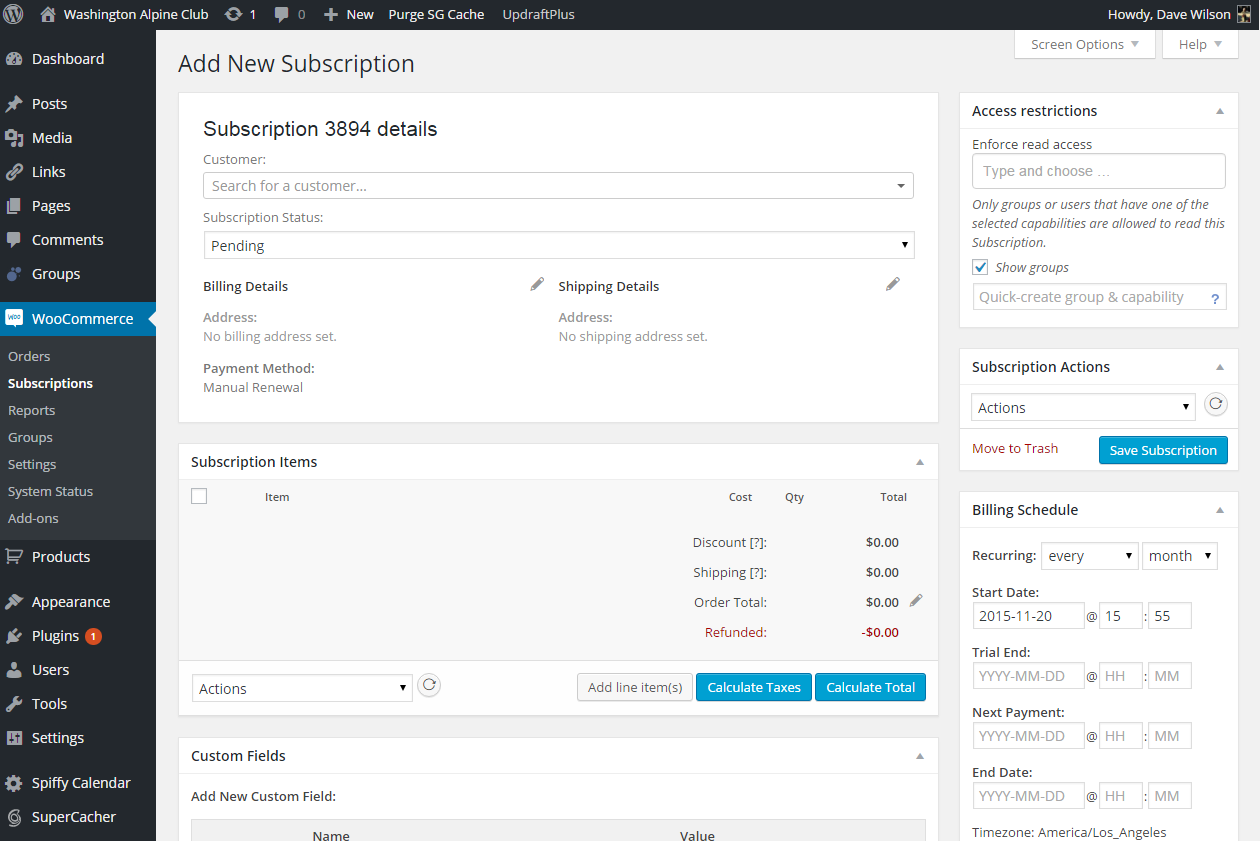
NOTE: **If you are manually entering a membership or annual cabin fee,** these are subscriptions. For these you only need to create a subscription by going Subscriptions, clicking on Add Subscription, and filling out the forms. Make sure to update the billing schedule to yearly and set the correct subscription end date. See the “Create a Subscription” section below.

Also see: <https://docs.woothemes.com/document/subscriptions/add-or-modify-a-subscription/>

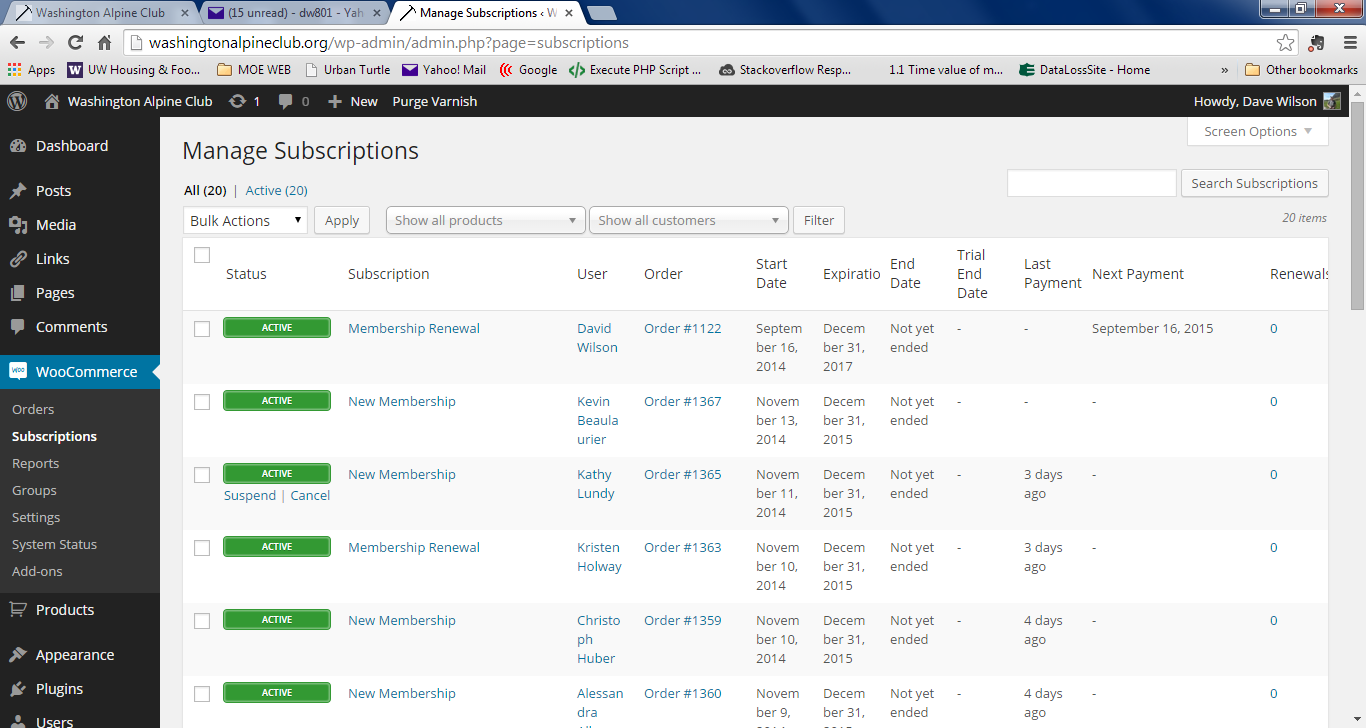
### Creating a Subscription:

In Woocommerce, memberships and annual cabin fees are ***Subscriptions***. If you receive a check or cash for a membership or an annual cabin fee you do not need to create an order, but you must create a new subscription.

1. Go to Woocommerce 🡪 Subscriptions 🡪 Add New Subscription…
2. Search for a customer – in this box just enter some characters of the customer’s name and select the appropriate customer.
3. Under “Subscription Items” click “Add Line Item(s)”, which reveals the “Add Product(s)” button.
4. Click the “Add Product(s)” button.
5. Enter some characters to search for a product.
   1. Entering RRM will return skus for renewing membership.
   2. Entering NRM will return skus for new memberships.
   3. Entering RSM will return skus for renewing senior/out of state memberhip.
   4. Entering NSM will return skus for new senior/out of state memberhip.
   5. Entering ACAB will return skus for annual cabin fees.
6. Select the appropriate product **– VERY IMPORTANT**: it will ALWAYS have a number at the end. In other words, don’t must select RRMEM for renewing membership. This is a product family sku and doesn’t indicate the duration of the subscription. Select RRMEM1, 2, etc instead.
7. Click Save
8. On the right hand side in the Billing Schedule section change the Recurring fields so that:
   1. The second field says “year” and the first field indicates the number of years for the subscription, e.g. “every” for one year, “every 2nd” for 2-year, etc.
9. Enter the appropriate end date and time:
   1. E.g. 12/31/16 (you’ll need to pick the right year associated with the subscription duration.
   2. For time pick 12:00. (not all fields in woocommerce deal with GMT well, so picking 12:00 means the date will stay 12/31/16 rather than 1/1/17.
10. Save the subscription using the “Save Subscription” button on the right.
11. Change the status from Pending to Active and save again.
12. **VERY IMPORTANT** Make sure the Next Payment field is blank. If there is a date in there the user will get a renewal invoice emailed to them on that date.



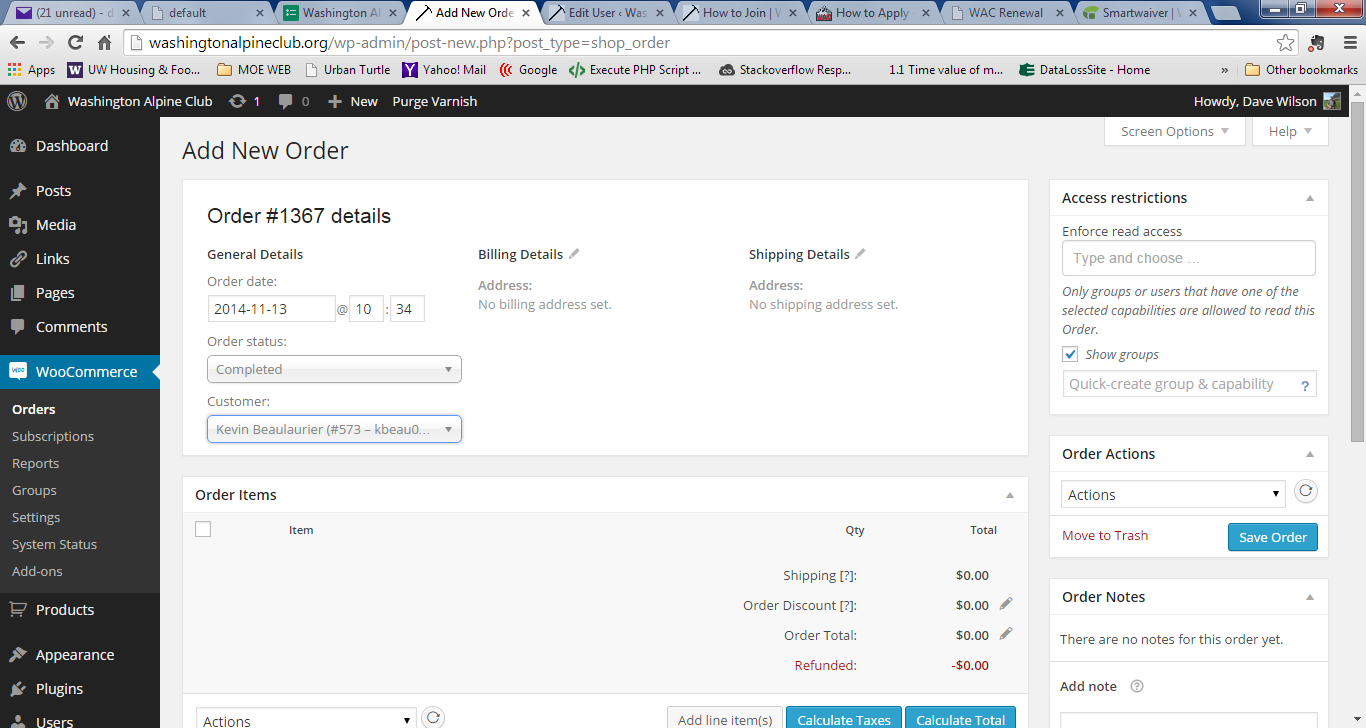
Verify the subscription looks good by going to Woocommerce 🡪 Subscriptions. Notice that the expiration date is correct.



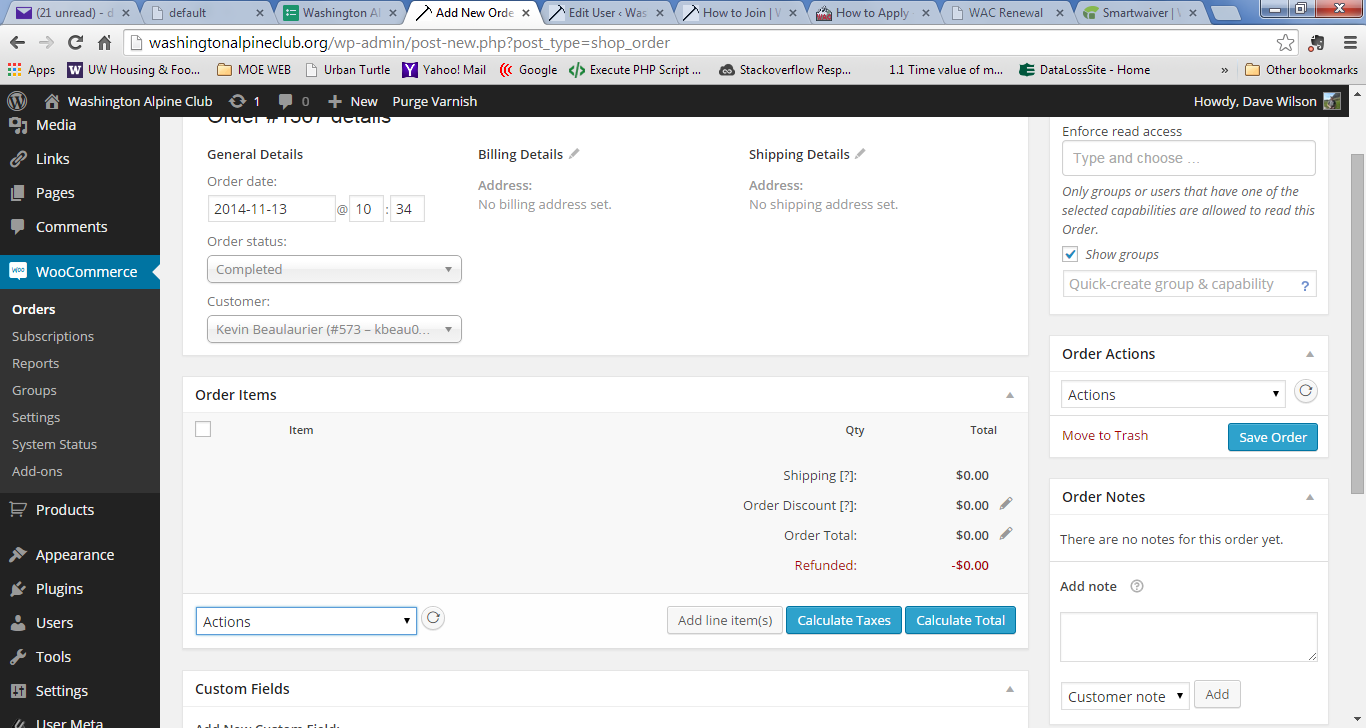
### Creating an Order:

If you receive cash or checks for things other than memberships or annual cabin fees, you will add those into the system as ***Orders***:

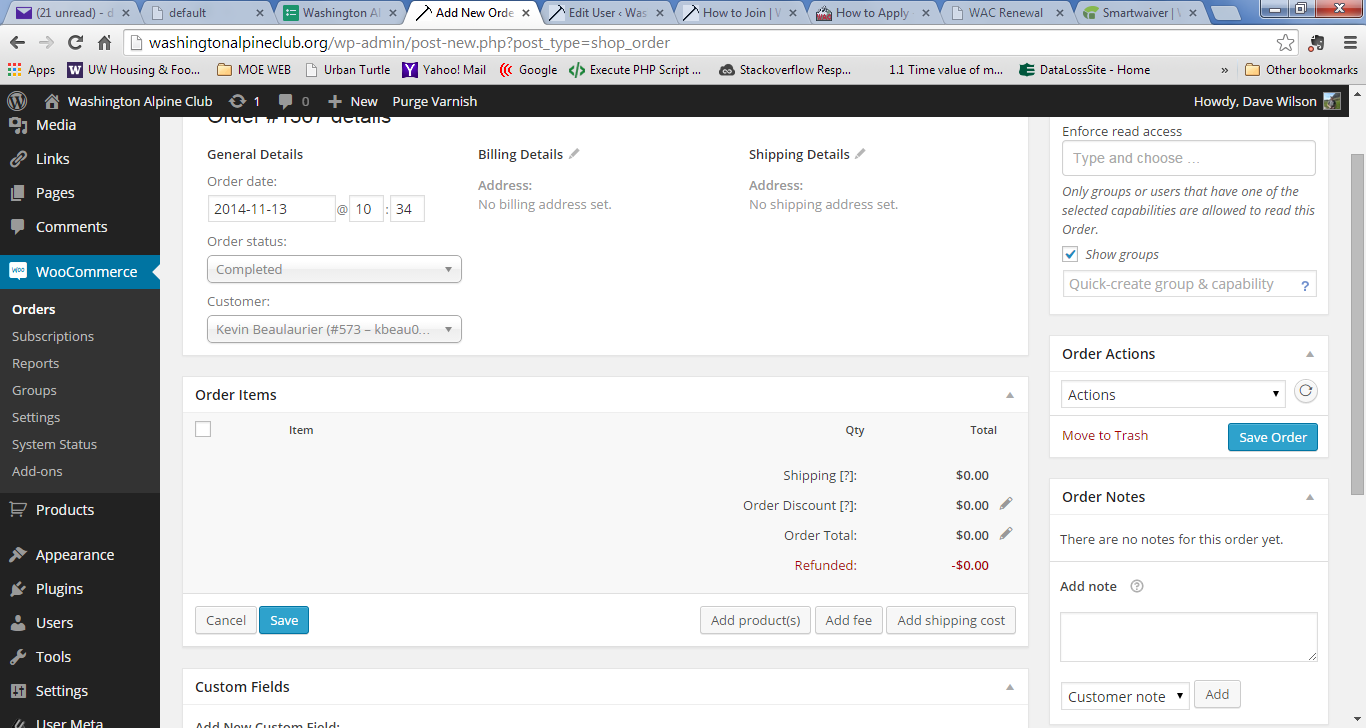
Go to Woocommerce 🡪 Add New Order…



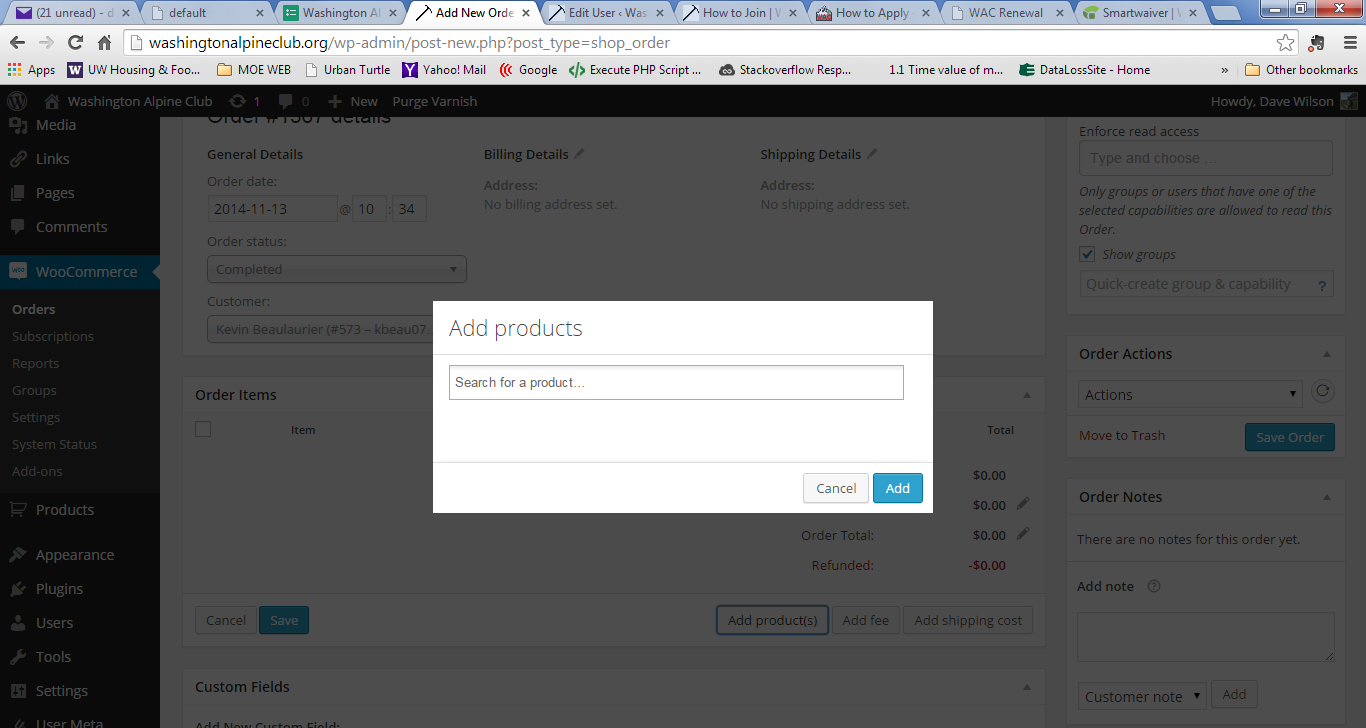
Enter the customer’s name in the customer search box and give it a minute to search. Then select the customer for the order. NOTE: The customer needs to already be created on the site. If the customer hasn’t already been created you will need to manually do that by going through the same process a new member would when they register for the site. Finally, click “add line item”…



Click on “Add products”…

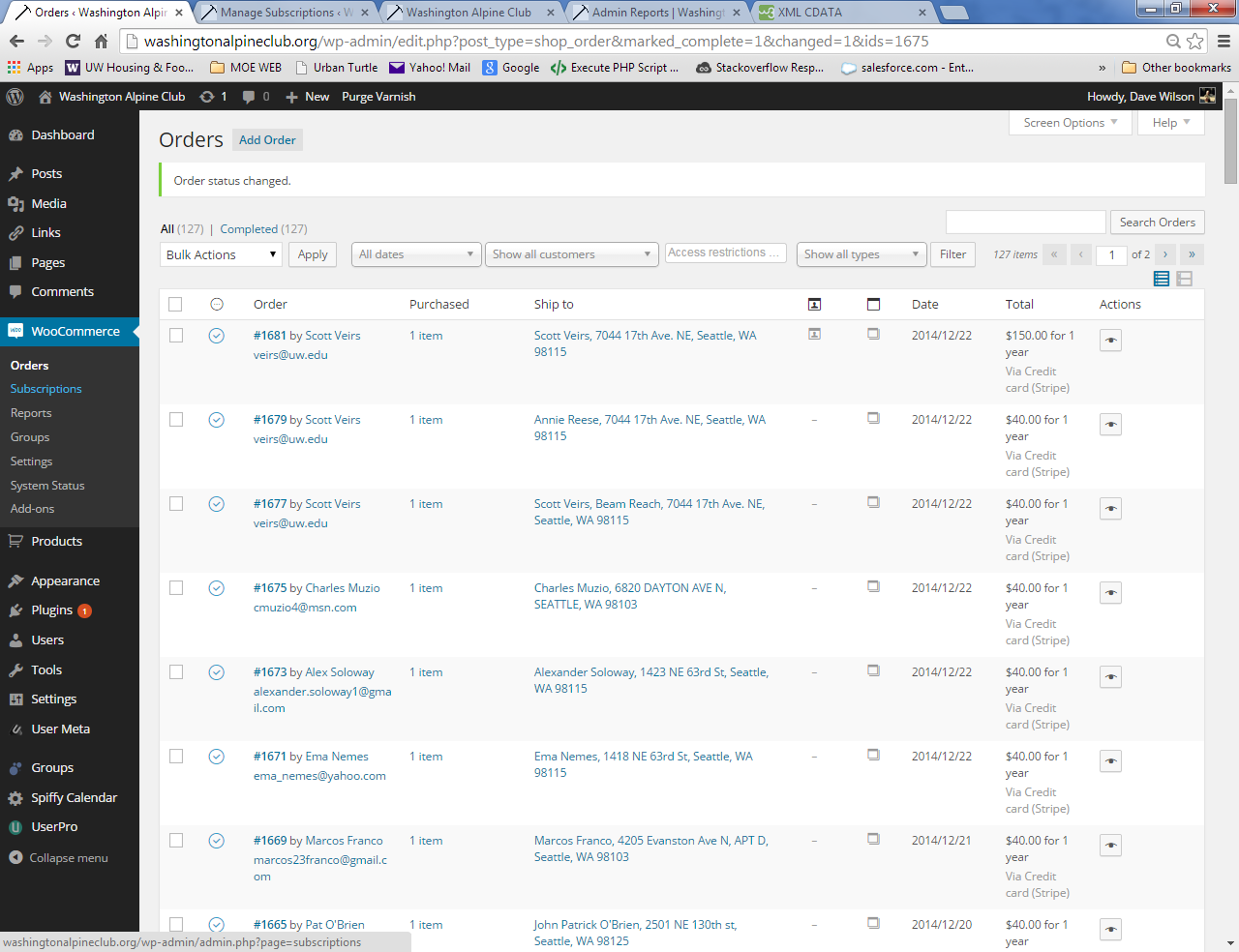


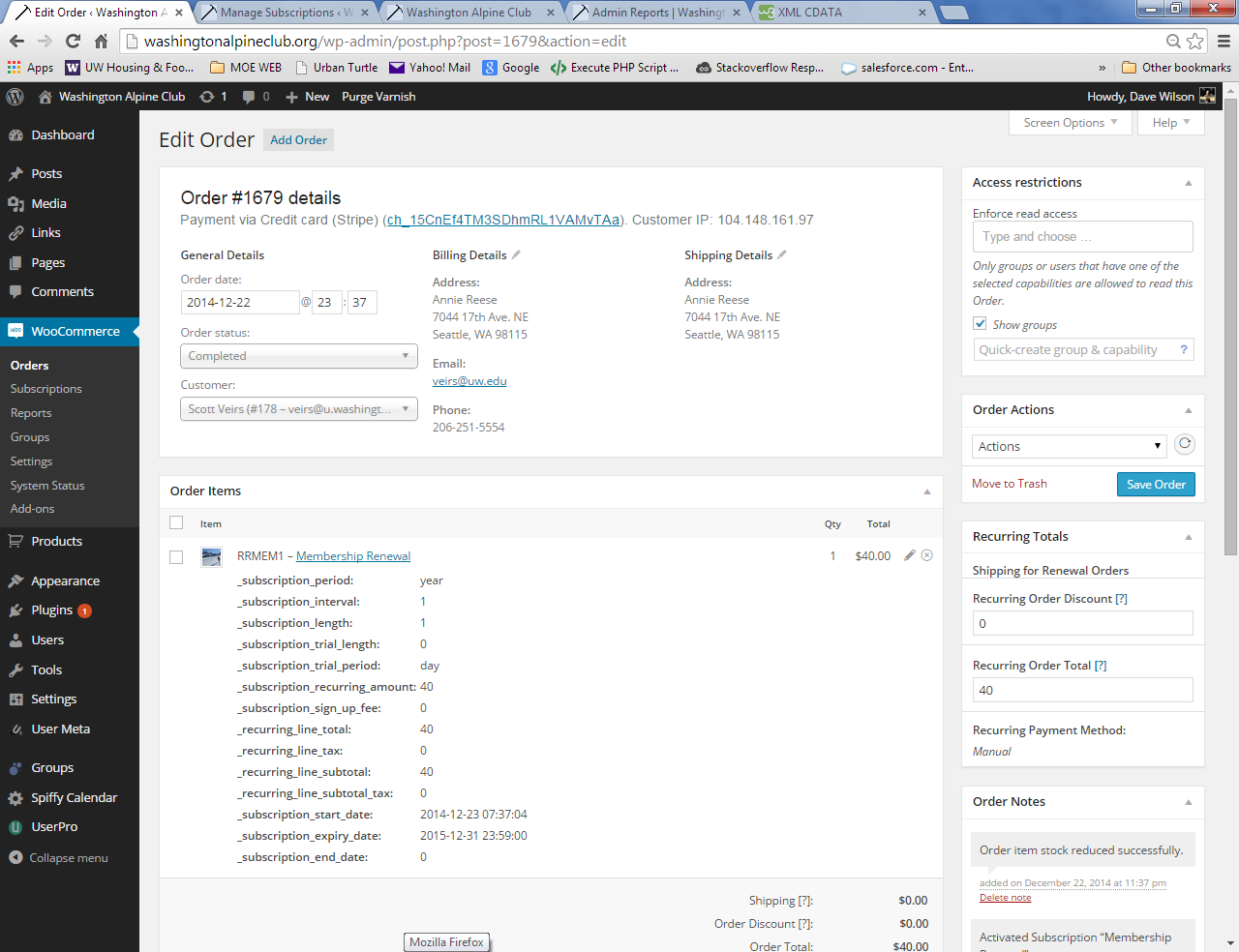
In the add products box enter a bit of the product name, such as “new” for new membership…

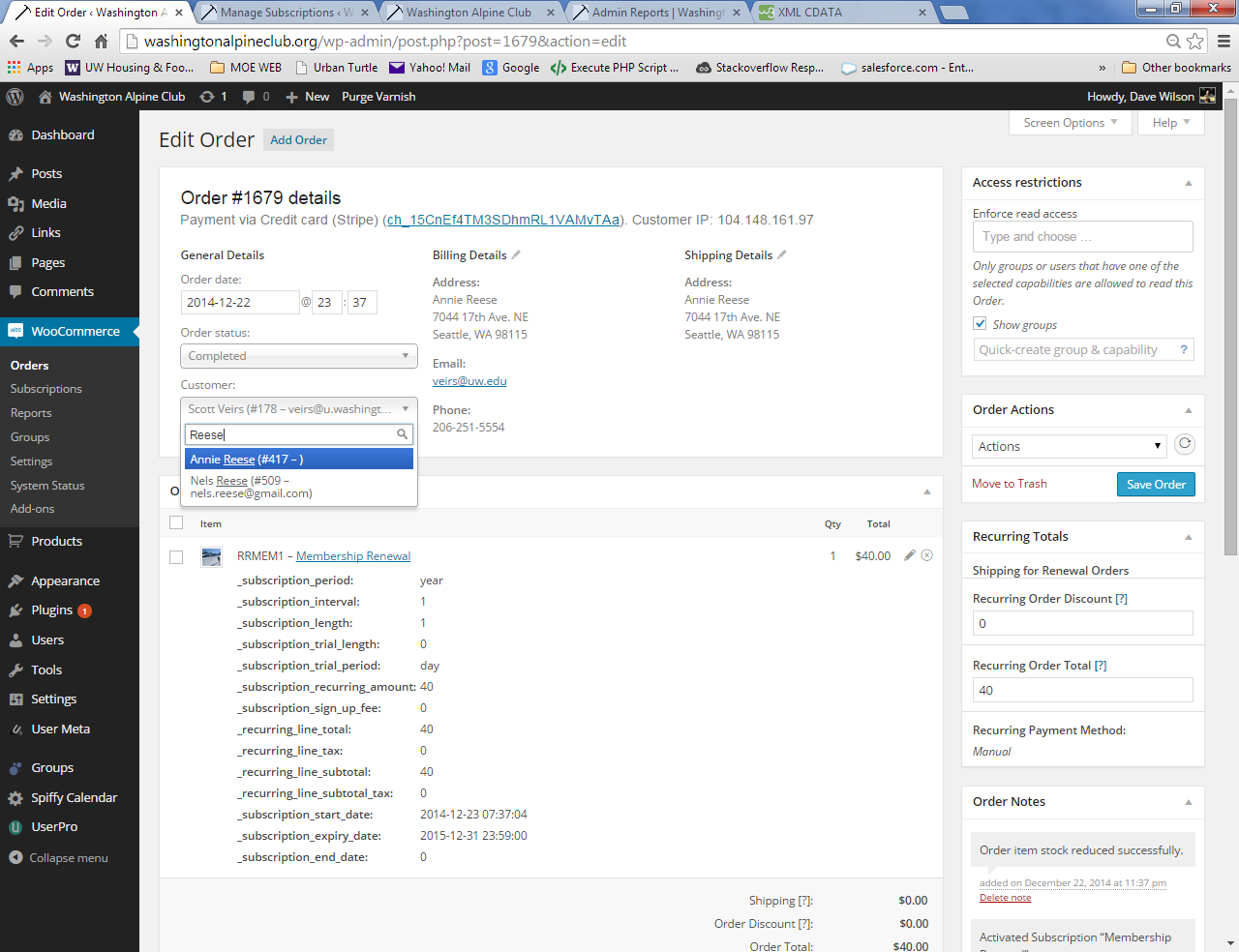


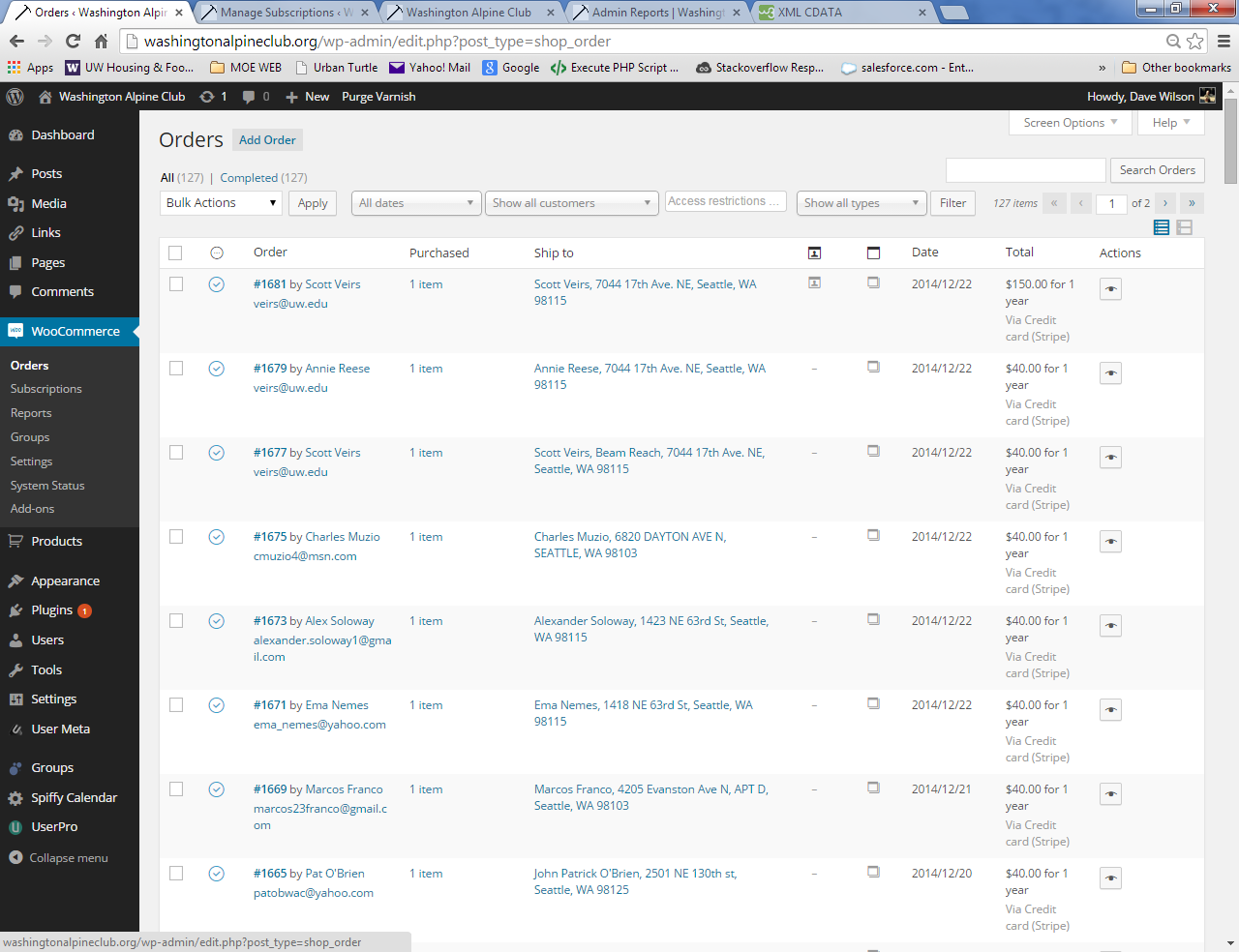
## Assigning an Order Made by One Member to Another Member:

Say person A pays for Person B’s membershp. Here’s how you assign that gift subscription to Person B.

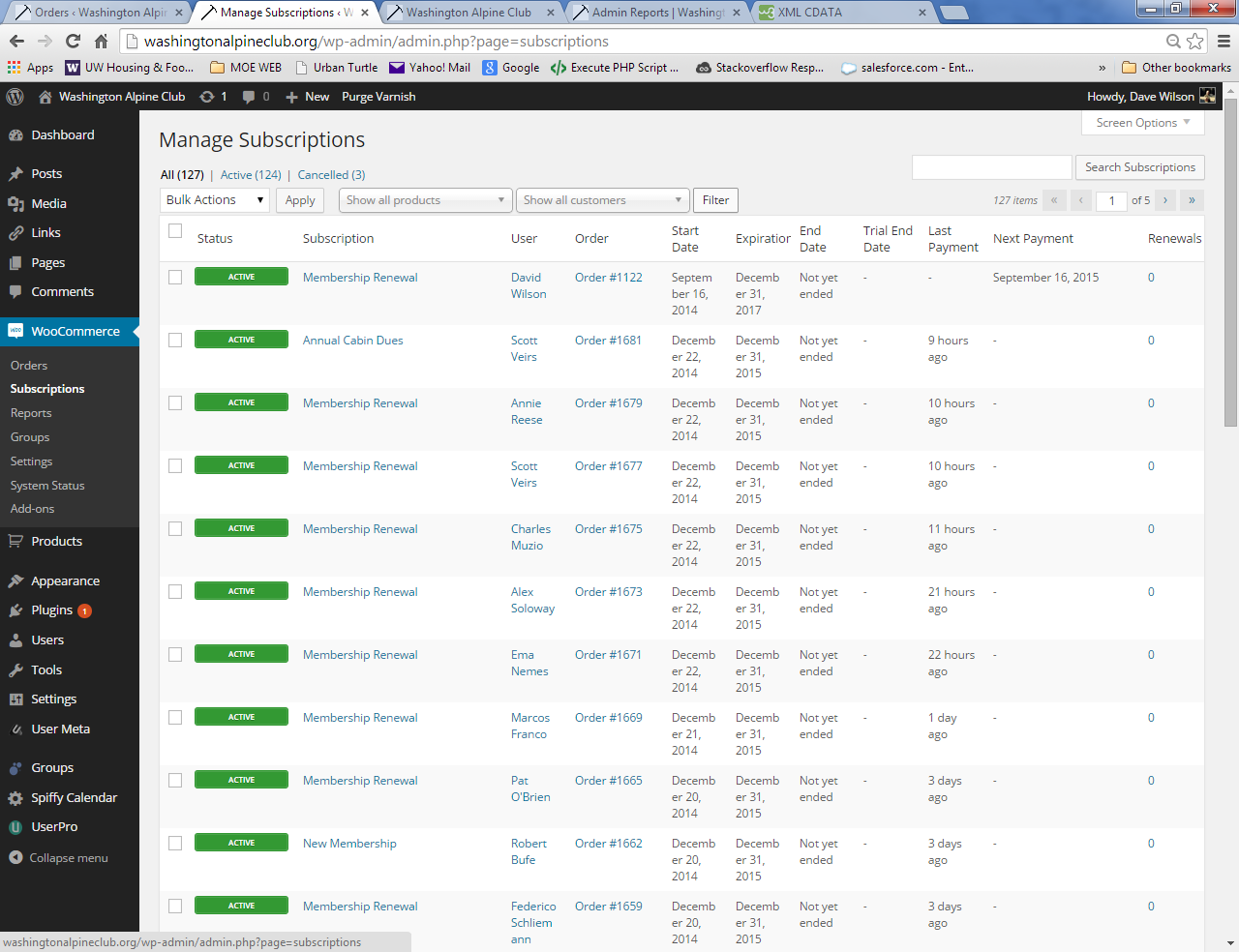
See two renewal orders below by Scott Veirs, one for Scott (order #1677), and one for Annie Reese (order #1679). Click on the order number #1679 to edit the order:

Editing Order #1679: Notice the customer identified under General Details is Scott Veirs. 

Click the customer dropdown and enter Reese to search for Annie. Then click “Save Order”:

Return to the Orders page to verify it all worked. Notice how order #1679 is now for Annie Reese.

You can also check to make sure the subscription is updated by going to the subscriptions page:



# Updating Products – e.g. Basic Climbing Class:

## Each year the products that represent the classes offered need to be updated for the current year:

1. Create a new group which will include the members of the current year’s class: e.g. Groups🡪add Group … add group “Basic Climbing Class 2016”. Model the group capabilities on the previous year’s class, which in this case would be “basic\_class\_student”.
2. Edit the Basic Climbing Class product thus (continuing with the 2016 basic class example):
   1. Change all 2015 text to 2016.
   2. Change SKU from “BCL15” to “BCL15”.
   3. Change Group from “Basic Climbing Class 2015” to “Basic Climbing Class 2016”.
3. Edit the page where the students pay for the class (in this case … /classes/basic-climbing):
   1. Change all 2015 text to 2016.
   2. Make the link to pay viewable by taking “HIDE” out of the active\_member group shortcode. i.e change this…

[groups\_member group="Active Members HIDE"]

to this…

[groups\_member group="Active Members"]

(adding the “HIDE” onto the end of this group filter once the class is closed again hides the link to the payment page for the class.)

# Troubleshooting:

## Trouble signing-in, resetting password

Often the results of javascript not being enabled by the browser, which is required to reset a password.

If a user’s browser does not have JavaScript enabled and they have successfully signed in they should see a red warning message at the top of their “My Membership Status” page, along with a link directing them how to enable JavaScript on their browser. In general, one way to troubleshoot suspected browser configuration issues is to have the user navigate to: [whatsmybrowser.org](http://www.whatsmybrowser.org/). They will be presented with a page of browser settings, at the top of which is a link that they can send to you. When you go to that link you’ll be able to see their browser settings.

## Checkout – Paying with Stripe

### Error Message:

*Thanks Dave*

*I think I was able to submit my waiver and I figured out how to pay the dues but when I submitted my visa info, it rejected the payment and said* ***customer not recognized****. Any suggestions?*

*Also, can I add payment for the telemark lessons?  I could not figure out how to add that to my cart.*

*Thanks*

*Rory*

* Was able to replicate with my own id: Error: "No such customer: cus\_4n7tQYhyBtNUiN; a similar object exists in test mode, but a live mode key was used to make this request."

### Action:

1. Delete all items from Cart. May also work to delete the cart metakey from usermeta.
2. For the User in question, delete metakey “\_stripe\_customer\_id” from the usermeta table.

## Purchases – no “Add to Cart” button

### Symptom:

User goes to buy a membership or other product, and there is no “Add to Cart” button for them to push.

### Action:

The Woocommerce purchase and checkout functionality requires that JavaScript be enabled. If a user’s browser does not have JavaScript enabled they should see a red warning message at the top of their “My Membership Status” page, along with a link directing them how to enable JavaScript on their browser. In general, one way to troubleshoot suspected browser configuration issues is to have the user navigate to: [whatsmybrowser.org](http://www.whatsmybrowser.org/). They will be presented with a page of browser settings, at the top of which is a link that they can send to you. When you go to that link you’ll be able to see their browser settings.

## Wordpress – Force Upgrade

### Error Condition

* Intermittently received error message: “You do not have sufficient permissions to access this page.” Used the Force Upgrade code to resolve.

### Action:

1. Created a "force-upgrade.php" file (see code in dropbox under the Wordpress Troubleshooting folder), uploaded it to the same directory as "wp-config.php" and opened the former in a web browser. See the WAC Wordpress/Wordpress directory for force-upgrade.php

# Infrastructure:

## Wordpress

* The site is based on Wordpress. The look and feel is governed by a wordpress theme. A handful of plugins have been added, along with a little bit of custom code, to accomplish membership management. More information on these topics can be found further on in this document.

## Hosting

* Provided by [Siteground](https://www.siteground.com/).
  + Shared hosting account accommodating production and, as subdomains of production, test sites.

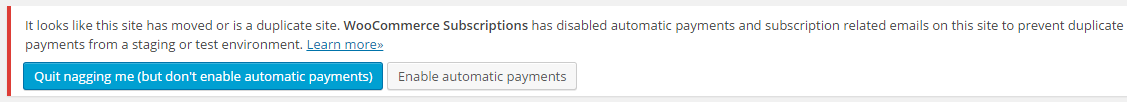
## Domains

|  |  |  |
| --- | --- | --- |
| **Domain** | **Registrar** | **Expiration** |
| washingtonalpine.club | Network Solutions | 1/12/2019 (plan to renew) |
| washingtonalpineclub.org | Network Solutions | 4/11/2021 (plan to renew) |
| washingtonalpineclub.com | Network Solutions | 8/14/2018 (plan to drop) |

# Test Sites:

To create a test site:

1. Create a new install of Wordpress as a subdomain.
2. Install licensed version of UpdraftPlus and provide with appropriate credentials.
3. Link UpdraftPlus to WAC backup dropbox.
4. Select instance of site to restore, as well as the option to rename all references to the new subdomain. Once the restore completes you will need to sign in with the credentials of the site version you just restored.
5. Turn off “force ssl” in Woocommerce 🡪 Settings 🡪 Checkout.
6. Set payment processors to test mode in Woocommerce 🡪 Settings 🡪 Checkout
   1. 🡪 Paypal – Enable Paypal Sandbox
   2. 🡪 Stripe – Enable Test Mode
7. Turn off automatic Updraft backups in Settings 🡪 Updraft 🡪 Settings
8. Change the home page to indicate this is a test site (in case someone stumbles across it thinking it’s the live site). Go to: Appearance 🡪 Theme Options 🡪 Home Page and change the Headline and Subheadline to TEST TEST TEST (note that this is specific to the Responsive Theme).
9. If restoring to a new subdomain, you will need to activate the woocommerce licenses. Our licenses can be used on up to 5 sites. Sign into the woocommerce site (see passwordsafe) to access the license keys, and paste these into the correct fields on the “woothemes helper” page on the admin dashboard menu.
10. Woocommerce Staging notice: you will have a message at the top of admin screens like this:



DO NOT ENABLE AUTOMATIC PAYMENTS UNLESS…you have enabled paypal sandbox and stripe test mode AND you have updated all user email addresses to test email addresses. Otherwise your users will get renewel emails when their test accounts renew. It is OK to click the “quit nagging me” button, but not necessary.

# Themes:

## Active Theme: Responsive (Child)

* The site utilizes a child theme based on the [Responsive Theme by Cyberchimps](http://cyberchimps.com/responsive-theme/). As the website says: *“Responsive is the world’s most popular Responsive WordPress Theme. Responsive Theme is an intuitive foundation with a fluid grid system that automatically adapts your website to mobile devices including smartphones, tablets and desktops so your website looks incredible on any device.”* Creating a child theme that references the parent (i.e. Responsive) enables theme modification without compromising the ability to take theme updates.
* Other premier themes were acquired ([Avada](http://themeforest.net/item/avada-responsive-multipurpose-theme/2833226) and [Enfold](http://themeforest.net/item/enfold-responsive-multipurpose-theme/4519990) and [Canvas](http://www.woothemes.com/products/canvas/)) but in the end I decide to work with Responsive because it’s lightweight and widely used. With a bit of work, however, these or any other themes could potentially be used.

### A Word about Child Themes

* Child themes allow you to modify or add functionality to the parent theme while still allowing you to receive updates of the parent theme. See the [Wordpress Codex on Child Themes](Wordpress%20Codex).
* The child theme’s stylesheet (style.css) is included after the parent theme's and will therefore override those in the parent theme’s stylesheet.
* Unlike style.css, the functions.php of a child theme does not override its counterpart from the parent. Instead, it is loaded in addition to the parent’s functions.php. This allows modification of functions in the parent theme in a manner that won’t disappear when the parent theme is updated. I recommend that you don’t modify functions.php—a WAC plugin was created as a better place to put our custom code. More on this below.

# Plugins:

## Subscription, Group, Checkout Functionality

(For and overview of how to put these parts together see this [article/video by Chris Lema](http://chrislema.com/membership-site-with-woocommerce/).)

1. [Woocommerce](http://www.woothemes.com/woocommerce/) – store and ecommerce functionality. (Free)
   1. Free
2. [Groups](https://wordpress.org/plugins/groups/) – used to establish different groups that have differing capabilities. (Free)
   1. Free
3. [Woocommerce Groups Extension](http://www.woothemes.com/products/groups-woocommerce/) – connects purchases to group membership. (5-site license, $99 for one year support and easy upgrades. Site will still run with expired license.)
4. [Woocommerce Stripe Extension](http://www.woothemes.com/products/stripe/) – adds the Stripe payment gateway. (5-site license, $99 for one year support and easy upgrades. Site will still run with expired license.)
5. [Woocommerce Subscription Extension](http://www.woothemes.com/products/woocommerce-subscriptions/) – enables subscription products (e.g. memberships). (5-site license, $249 for one year support and easy upgrades. Site will still run with expired license.)
6. [Woocommerce Smart Coupon Extension](http://www.woothemes.com/products/smart-coupons/) – currently disabled-future use questionable. (Single site license, $99 for one year support and easy upgrades. Site will still run with expired license.
7. [Woothemes Helper/Updater](http://docs.woothemes.com/document/woothemes-helper/) – manages woocommerce licenses/updates. (Free)

## Additional Site Functionality

1. [Exclude Pages](https://wordpress.org/plugins/exclude-pages/) From Navigation – used so the login page doesn’t have to show up on the menu. (Free)
2. [UserPro](http://userproplugin.com/userpro/) – used for member directory, user profile, registration. Site uses a customized-for-our-needs version of UserPro called UserPro-WAC. (One-time ongoing license $28.)

## Utilities

1. [WPCrontrol](http://wordpress.org/plugins/wp-crontrol/) – for management of the WP Cron system. (Free)
2. [Updraft Plus](https://wordpress.org/plugins/updraftplus/) – for backing-up the site. Backs up site to Dropbox on a schedule, and prior to any software upgrade. Works, but there may be better solutions out there. (Annual License $99. Includes several add-ons for migrating sites and scheduling backups.)
3. [All-in-One-Favicon](https://wordpress.org/plugins/all-in-one-favicon/) – for easily managing favicons. (Free)
4. [Outerbridge Nested Shortcodes](https://wordpress.org/plugins/nested-shortcodes/) – allows use of nested shortcodes. (Free)
5. [Spiffy Calendar](https://wordpress.org/plugins/spiffy-calendar/) – a calendar for upcoming events. (Free)
6. [Groups 404 Redirect](https://wordpress.org/plugins/groups-404-redirect/) – to handle pages that are unavailable because a particular user might not be in the appropriate group to view a page. (Free)
7. [Exclude Pages from Navigation](https://wordpress.org/plugins/exclude-pages/) – to remove a page from primary navigation. (Free)
8. [Author Avatars List](https://wordpress.org/plugins/author-avatars/) – to display a grid of authors. (Free)
9. Other stuff – a number of other tuning and debugging plugins – turned off in production – are also loaded.

## Custom

1. wacsitespecific-plugin.php – used to hold site-specific code so it doesn’t have to be put into the theme’s functions.php file. See Custom Code section.

# Custom Code:

Used these references for creating a [plugin](http://www.wpbeginner.com/beginners-guide/what-why-and-how-tos-of-creating-a-site-specific-wordpress-plugin/) and creating a [widget](http://www.wpbeginner.com/wp-tutorials/how-to-create-a-custom-wordpress-widget/). And [phpbeautifier](http://phpbeautifier.com/beautify.php) for cleaning up the code.

In general, when the code of others’ was modified, it was commented with “wac – daw” so you can search on that to find where the code was changed.

## Plugins

* wacsitespecific-plugin.php - Created a site-specific plugin to hold filters, custom widgets, and so forth so that we don’t need to add these to the theme’s functions.php file.
* userpro-wac – customized version of UserPro.
  + api.php, memberlistfunctions.php – Added, for example, ability to sort member directory by meta value so we could, for example, sort by lastname (search “daw – wac” to find modified code). Also changed index.php to change name of plugin to userpro-wac.
  + Scripts/scripts.min.js – Relaxed password strength requirements. Searched for "password\_not\_strong", and in the related conditional statements changed the following (there were three instances of each):
    - ( element.val().length < 6 ) /\* was < 8
    - ( userpro\_password\_strength\_meter( element ) < 0 ) /\* was < 3
* Woocommerce-subscriptions – Enable the editing of some subscription date fields so that if necessary and admin can fix (or manually adjust) start and expiration dates for a subscription.
  + classes/class-wc-subscriptions-order.php – only file where code was changed.

# Smartwaiver Integration:

## API Key

We can create as many as we want. Here’s one: 87b8c5d6708dee2fb09a34ebc0de4eea-99774

## Autotagging Waiver

This pre-populates a new waiver with information about the WAC Member. The “auto\_tag” value is a key that can be used to retrieve waivers via the api.

### Wordpress shortcode text

https://www.smartwaiver.com/auto/?auto\_waiverid=54090f9a22acc&amp&auto\_tag=[userinfo field='ID' nospan='true'][/userinfo]&auto\_fill\_firstname=[userinfo field='first\_name' nospan='true'][/userinfo]&auto\_fill\_lastname=[userinfo field='last\_name' nospan='true'][/userinfo]&auto\_fill\_phone=[userinfo field='home\_phone' nospan='true'][/userinfo]&auto\_fill\_addresslineone=[userinfo field='address' nospan='true'][/userinfo]&auto\_fill\_city=[userinfo field='city' nospan='true'][/userinfo]&auto\_fill\_state=[userinfo field='state' nospan='true'][/userinfo]&auto\_fill\_zip=[userinfo field='zip' nospan='true'][/userinfo]&auto\_fill\_email=[userinfo field='user\_email' nospan='true'][/userinfo]

#### An example

https://www.smartwaiver.com/auto/?auto\_waiverid=527d7ffc366aa&amp&auto\_tag=1&auto\_fill\_firstname=Dave&auto\_fill\_lastname=Wilson&auto\_fill\_phone206.548.9906&auto\_fill\_addresslineone=3629 Whitman Ave N&auto\_fill\_city=Seattle&auto\_fill\_state=Washington&auto\_fill\_zip=98103&auto\_fill\_email=dw801@yahoo.com

## Retreiving

### Getting list of waiver types

https://www.smartwaiver.com/api/v3/?rest\_request=87b8c5d6708dee2fb09a34ebc0de4eea-99774&resetapi\_listofwaivertypes

### Getting a specific membership waiver by tag

https://www.smartwaiver.com/api/v3/?rest\_request=87b8c5d6708dee2fb09a34ebc0de4eea-99774& rest\_request\_tag=1

#### Returns

<xml>

<participants>

<participant>

<participant\_id>8582482464</participant\_id>

<waiver\_id>54076ec701c73</waiver\_id>

<firstname>Dave</firstname>

<middlename/>

<lastname>Wilson</lastname>

<dob>1961-01-07</dob>

<date\_created\_utc>2014-09-03 19:40:55</date\_created\_utc>

<date\_accepted\_utc/>

<pdf\_url>

NTQwNzZlYzcwMWM3M3x8fDdjY2I4NDk4MDcxMjAyZmZjMzAyOTNkYTcxMjAyNzdm

</pdf\_url>

<tags>

<tag>1</tag>

</tags>

<checkins/>

<customfields/>

<waiver\_title>WAC Membership Waiver</waiver\_title>

<pending\_email\_validation>false</pending\_email\_validation>

<completed\_at\_kiosk>false</completed\_at\_kiosk>

<images/>

<address\_city>Seattle</address\_city>

<address\_country>US</address\_country>

<address\_lineone>3629 Whitman Ave N</address\_lineone>

<address\_state>WA</address\_state>

<address\_zip>98103</address\_zip>

<phone>(206) 518-4101</phone>

<primary\_email>dw801@yahoo.com</primary\_email>

<primary\_email\_confirm>dw801@yahoo.com</primary\_email\_confirm>

<SERVERSIDE\_autofill\_addresslineone>3629 Whitman Ave N</SERVERSIDE\_autofill\_addresslineone>

<SERVERSIDE\_autofill\_city>Seattle</SERVERSIDE\_autofill\_city>

<SERVERSIDE\_autofill\_email>dw801@yahoo.com</SERVERSIDE\_autofill\_email>

<SERVERSIDE\_autofill\_firstname>Dave</SERVERSIDE\_autofill\_firstname>

<SERVERSIDE\_autofill\_lastname>Wilson</SERVERSIDE\_autofill\_lastname>

<SERVERSIDE\_autofill\_state>Washington</SERVERSIDE\_autofill\_state>

<SERVERSIDE\_autofill\_zip>98103</SERVERSIDE\_autofill\_zip>

<how\_many\_auto\_photo\_capture\_photographs\_captured>0</how\_many\_auto\_photo\_capture\_photographs\_captured>

<SERVERSIDE\_csrf\_token>

b15f2fcc9dff35332c42431dde6672421ff239a5b18f5eede8c943a0c4056347

</SERVERSIDE\_csrf\_token>

<web\_browsers\_user\_agent>

Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/36.0.1985.143 Safari/537.36

</web\_browsers\_user\_agent>

<completed\_from\_ip\_address>63.145.199.253</completed\_from\_ip\_address>

<waiver\_type\_guid>527d7ffc366aa</waiver\_type\_guid>

<marketingallowed>not asked</marketingallowed>

</participant>

</participants>

</xml>

### Parsing the xml

<?php  
$xml=simplexml\_load\_file("note.xml");  
print\_r($xml);  
?>

### Viewing the PDF

https://www.smartwaiver.com/api/v3/?rest\_request=87b8c5d6708dee2fb09a34ebc0de4eea-99774&restapi\_viewpdf=NTQwNzZlYzcwMWM3M3x8fDdjY2I4NDk4MDcxMjAyZmZjMzAyOTNkYTcxMjAyNzdm

<https://www.smartwaiver.com/m/vault_temp/54076ec701c736225b321e51648ba76609ce570df69b0bf8a27e19c79313d62a1e605415fc108.pdf>

### PHP Sample Code

#### Github lib

<https://github.com/smartwaivercom/smartwaiverapi/tree/master/v3/php>

#### List Waiver Types

<https://github.com/smartwaivercom/smartwaiverapi/blob/master/v3/php/api_list_of_waivertypes.php>

### Custom Shortcode

Created a new shortcode [checkwaiver group="Member Waiver"] that queries Smartwaiver by Wordpress User\_id to see if a waiver has been created there with that User\_id. If it does, the shortcode then places them into the specified group. On the WAC site, the “Member Waiver” group is used to signify those with current waivers in place.

Note: for this to work the waiver would have to be created using a link from the website, since the userid is passed as a get parameter to Smartwaiver.

The code is in the WAC sitespecific plugin.