

Project.

A **custom taxonomy** called "**User Tags**".

Custom Taxonomy Plugin in WordPress®

by
Arvind Arya

Subscriber (8)					
<div>Apply Change role to... Change new york Filter</div>					
Username	Email ↕	Role	Login as User	User Tags	Total posts
ben wyatt	ben@wyatt.com	Subscriber	Login	Dreams, new york, Police	\$100
id minded	broad@minde	Subscriber	Login	Doctor, new	\$100

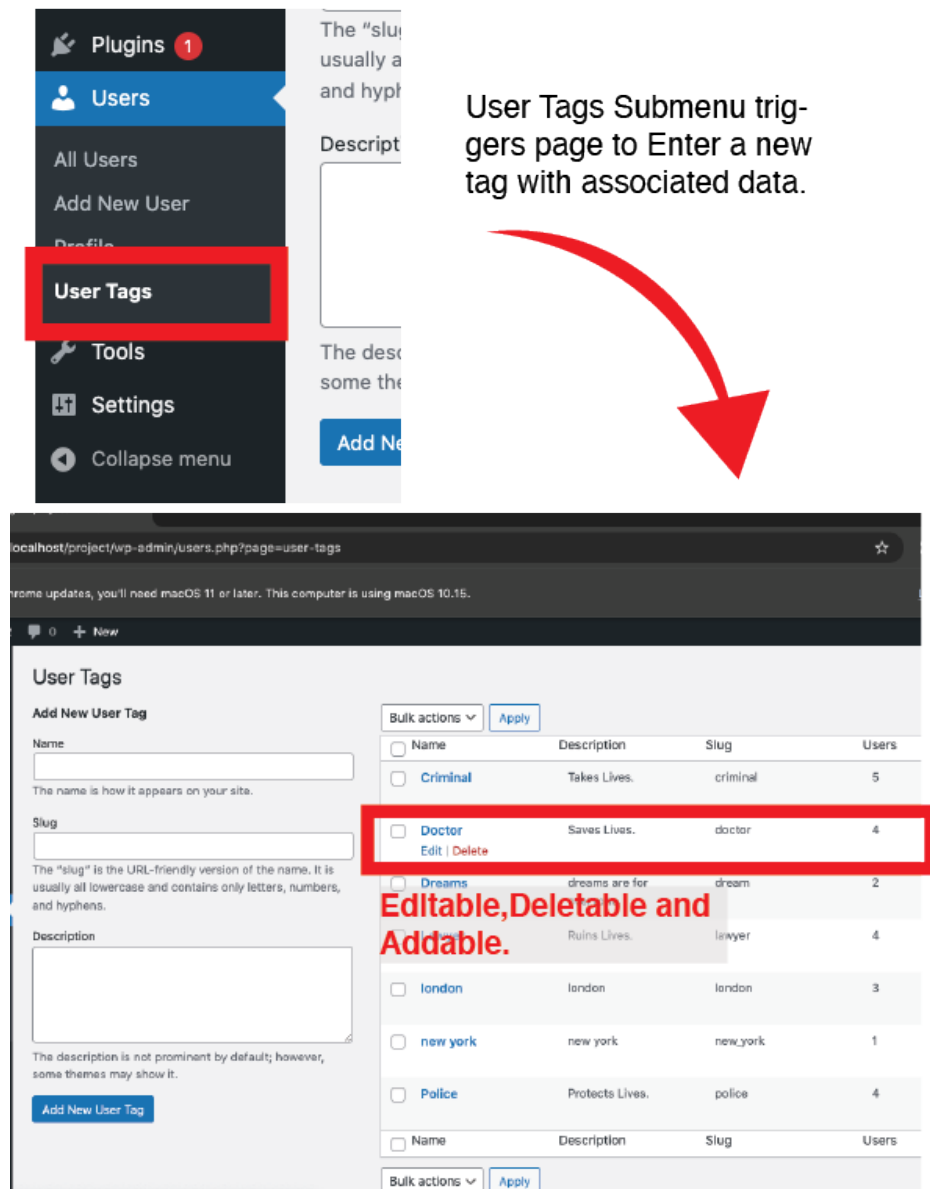


Figure 1

Under the Users Menu in the Wordpress admin panel created a Sub Menu called User Tags, which can add Name Slug and Description for the tag. This is the Entry to the User Tags.

Add New User
Create a brand new user and add them to this site.

Username (required):

Email (required):

First Name:

Last Name:

Website:

Password: Generate password Very weak

Confirm Password: ☐ Confirm use of weak password

Send User Notification: ☒ Send the new user an email about their account.

Role: Subscriber

Additional User Information

User Tags

Criminal Doctor Dreams ...

Selected Tags: Doctor X new york X

Select existing user tags for this user:

Total Spent:

Optional: Total amount spent by the user.

Last Order:

Optional: Last order identifier (e.g., #123).

Count:

Optional: Number of orders placed by the user.

User Tags are added by tapping the tags and can be cancelled and some extra information is taken in during Adding a User.

Figure 2

This is how we Add New Users So that we get User Tags associated with Users.

Dashboard

Users Add New User

All (8) | Administrator (1) | Subscriber (7)

Bulk actions: ▼ Apply Change role to... Change Filter by user tags... Filter

<input type="checkbox"/>	Username	Name	Email	Role	Login as User	User tags	Total Spent	Last Order

Figure 3

A Dropdown setup to Filter on the basis of User Tags.

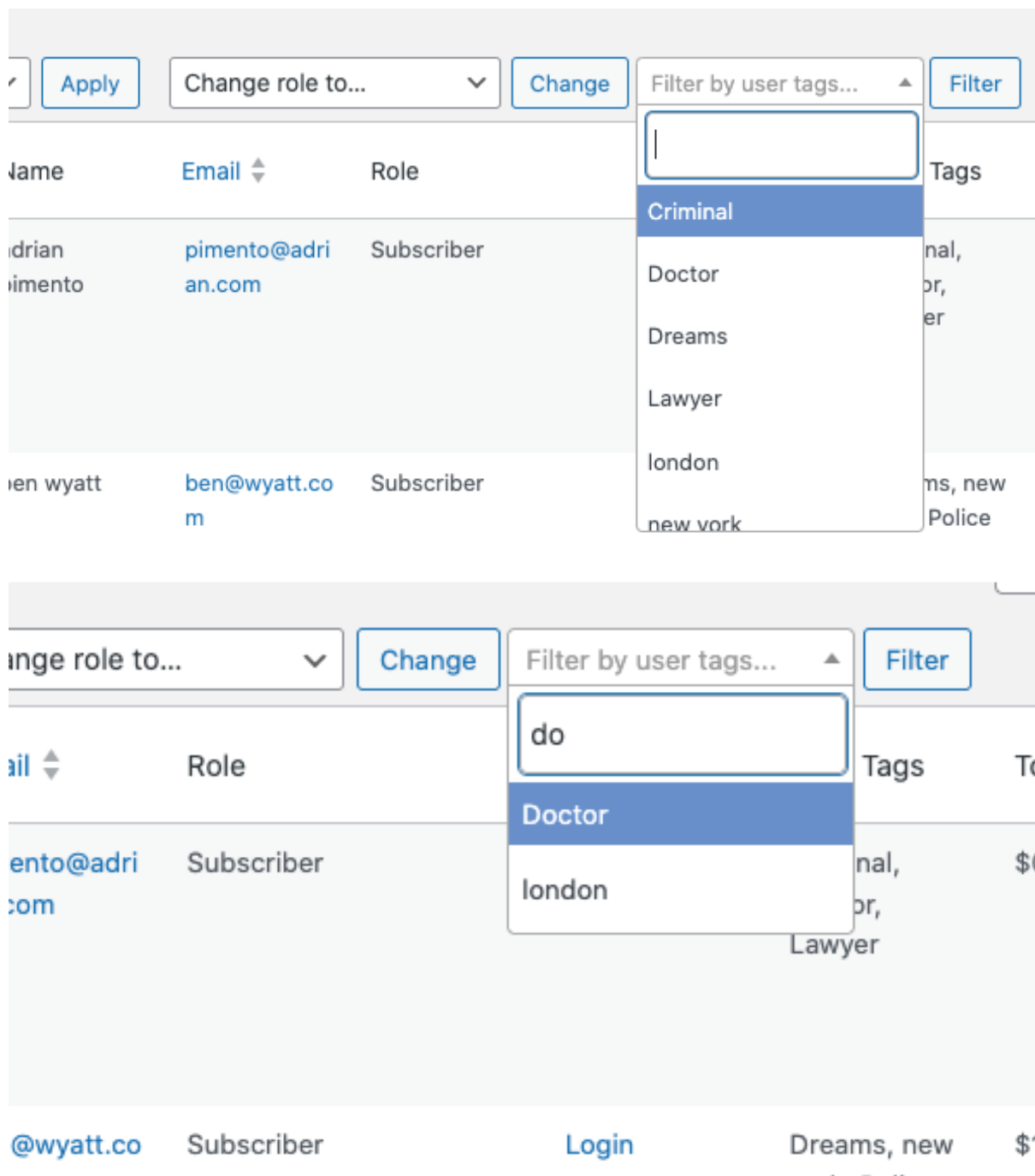


Figure 4

Implemented a **searchable, dynamic dropdown** for selecting User Tags.

Used **Select2** for enhanced UI and AJAX-based data loading.

The dropdown **fetches and displays available user tags dynamically**.

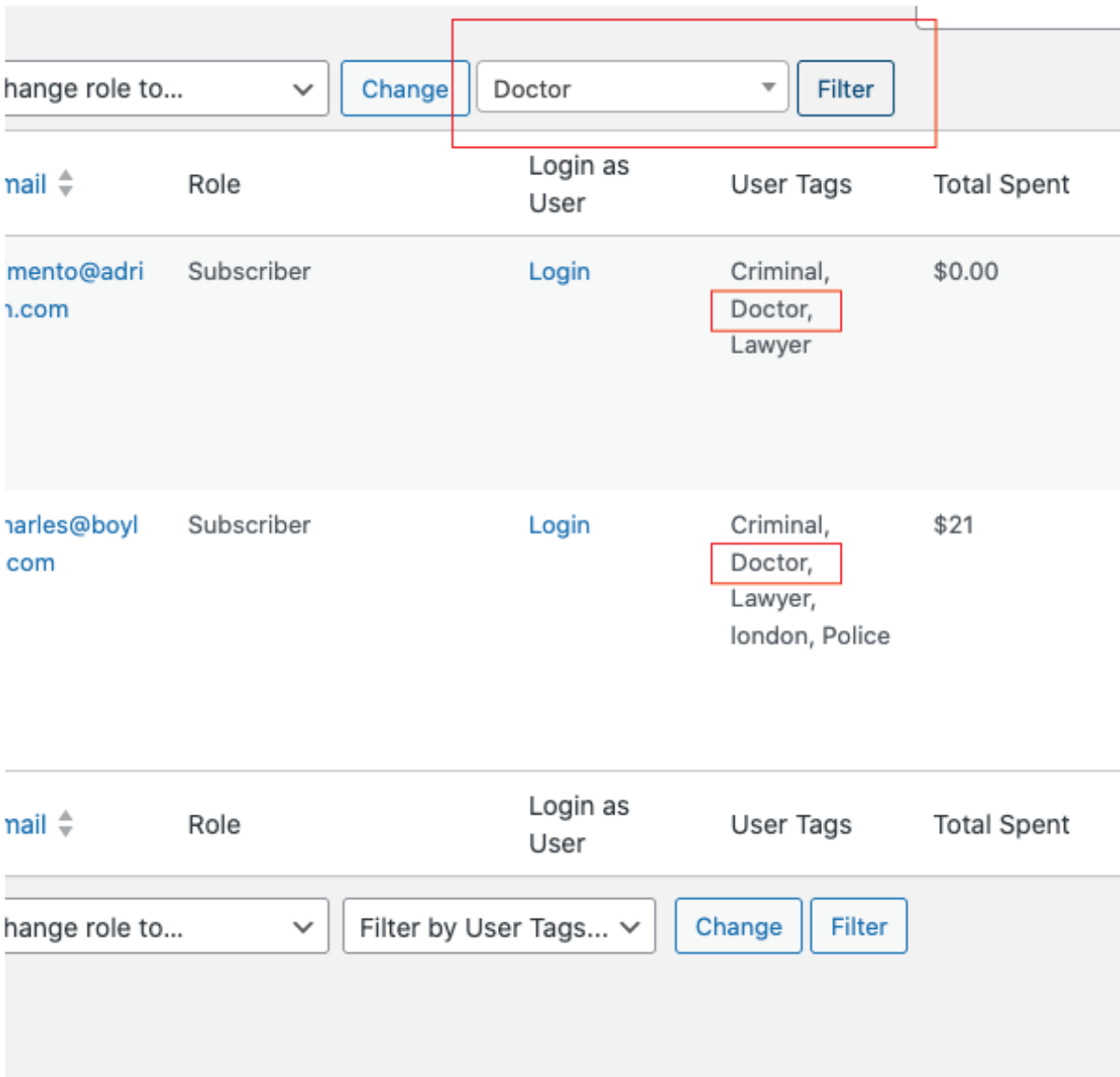


Figure 5

On selecting a User Tag and submitting the filter, the **Users screen** would reload and display only users with the selected tag.

The Program Approach.

The Challenge.

"By default, WordPress **custom taxonomies** can only be assigned to post types. However, in this challenge, you must **override this limitation** and figure out a way to categorize users using the WordPress' **custom taxonomy system**."

Non Technical.

The Challenge was to create a custom taxonomy like the Wordpress framework has built-in for the Posts.

My approach was first to get User Tags these would be used to filter the Users in the User table so that we can search the table in a more meaningful way. So I made a form to get Tags first, that's right tags not User Tags, these tags will be later assigned to users while adding them(Shown in Figure 2). These users will be now Shown in the User table that Wordpress maintains in User Section of the Dashboard of the Wordpress website, which is accessible to a person if the person has permission to access it. I also put in some additional information along with these Users to populate the User table and make it look more real world.

Then I added a Dropdown that contains all the user tags that are being created and with the help of select2(a library or a helpful function used in Wordpress) to make the Dropdown(Shown in Figure 3) dynamic that is Searchable in case you have a lot of tags it saves time.

Now the Final task was to make a selection of the 'User Tag' when you press Filter and get all the Users that have the chosen User Tag associated with them(Shown in Figure 4). That was made using by querying the database.

Technical.

Making custom taxonomy for User Tags was an easy task but it was a little tricky with the select2. The goal was to create a filtering based on User Tags, that is what I have made. I tried to integrate select2 to a custom made dropdown that worked on a taxonomy called User Tags but it won't work, it was giving no results at all and it was difficult to work with then I came up with a workaround which I kept to show that I can come up with clever workarounds when things are not working like they are supposed to.

Let me Explain one by one, also I have attached the code of course if you want to see.

First let us deep dive into our PHP.

```
register_user_tags_taxonomy()
```

This is our first function which registers User Tags, which registers the user_tag taxonomy now this function, tells Wordpress, Hey! I am searchable.

```
function add_login_as_user_column( $columns ) {  
function hide_posts_column_in_users( $columns ) {
```

These two functions adds and subtract columns to be displayed in the user table.

```
function populate_custom_user_columns( $value, $column_name, $user_id ) {
```

This table takes the users created and their attribute stored in a Database and populates them in an aesthetic, clean way in the UI.

```
function user_tags_admin_menu() {
```

This function makes the Sub Menu called "User Tags". This is accessible in the Users pane in Wordpress Dashboard.

```
function my_render_user_tags_page() {
```

This function simply has the HTML and styling for the page of "User Tags" also related function names and id defined, this is very complicated but gets the job done.

```
function my_handle_add_user_tag() {
```

This function handles the submitting of the the form from "User Tags" and hereby creates new tags for further use.

```
function my_add_user_custom_fields_new_user() {
```

This function presents the form to add new user it takes the name of the users along with tags which are presented to them like buttons to choose from and also they can deselect it.

```
function my_save_user_custom_fields_new_user( $user_id ) {
```

This function saves the form and creates another User Meta Data to save along with Tags called Meta Tags these are the same tags just saved in wp_usermeta table in the Database of word press, why I did that would come later. Is saving these Meta tags Redundant? Not really, if we are talking a project like this it won't make a dent, it would if it were twitter but it is a simple tagging plugin and has a lot more meta data than this.

```
function filter_users_by_tags_dropdown( $which ) {
```

This is where the Dropdown is added, now this is where I create the Dropdown, this is where the trick is see this dropdown is populated with meta tags and is not that aesthetically pleasing and for the life of me, I couldn't get the placement right for this one, it always sat between change dropdown and change button tried a lot to move it, but it didn't, it was very but it was useful it could filter the table on

metadata it was the default one that came with the wordpress framework. This is invisible cause I have made it look so. This is the trigger that ultimately sends out the URL that filters the Users Table.

```
function filter_users_by_meta_tags_query( $query ) {
```

Here the dropdown is querying the Database on Meta Tags added earlier.

```
function add_blank_user_tags_filter_html( $which ) {
```

This function adds the second dropdown it is the Dynamic Dropdown which is searchable and works with select2 to make it happen. It is populated by the taxonomy "User Tags"

```
function enqueue_select2_assets_for_users( $hook ) {
```

This is the standard setup of Select2 to work with the Wordpress php, this is how I make the dropdown dynamic.

```
function search_user_tags_callback() {
```

This function makes the Dropdown Searchable.

Now Let us dive into JavaScript.

JavaScript is very crucial to all this functioning.

```
var $select2 = $('#filter-by-user-tag').select2({
```

This sets up the dynamic Dropdown.

```
function getUrlParameter(name) {
```

This function gets the URL so that I can set the Dynamic Dropdown to the filtered value so it looks like it triggered the reload, which it didn't.

```
$('#input[name="filter_action2"]').on('click', function(e) {
```

This is where all the action happens, When I press Filter on the Dynamic Searchable Dropdown, it sends the value to the first dropdown and it selects the said value and then I programmatically trigger the reload and then reload the dynamic dropdown again.