

```

7   unless params["filter"].nil?
8     #filter params which have the string "<filtertype>,<post/tagID>,<tag/uniname>"
9     @filter = params["filter"].split(",")
10    @filter_type = @filter[0]
11    @filter_id = @filter[1]
12    @filter_name = @filter[2]
13  else
14    @filter = nil
15  end

```

Listing A controllers/account_moderation_feed.rb

```

11  @user = User.new
12  @user.login(params)
13  @error = nil
14
15  if @user.exist? && @user.valid?
16    @logged_in_user = User.where(username: params.fetch("username","")).first
17
18    session[:logged_in] = 1
19    session[:is_guest] = 0
20    session[:account_type] = @logged_in_user.account_type
21    session[:userID] = @logged_in_user.userID
22    session[:username] = @logged_in_user.username
23    session[:universityID] = @logged_in_user.universityID
24
25    redirect "/"
26  else
27    @error = "The username or password that you entered is incorrect, please try again"
28  end

```

Listing B controllers/login.rb

```

25    if @user[:account_type] == 2
26
27      @user_type_query = "Moderator"
28      @posts = Post.where(universityID: @user[:universityID]).where(is_moderated: 0)
29
30    elsif @user[:account_type] == 3
31
32      @user_type_query = "Administrator"
33      @posts = Post
34
35    else
36      redirect "/"
37    end

```

Listing C controllers/post_moderation_feed.rb

```

29  # Create university records if they don't exist
30  def create_universities
31    File.open("db/universities.txt") do |university_list|
32      university_list.each do |university|
33        DB[:universities].insert(university_name: university.chomp)
34      end
35    end
36  end

```

Listing D db/db.rb

```

3   def load(params)
4     if params.fetch("is_guest", "") == 1
5       self.is_guest = 1
6     else
7       self.is_guest = 0
8       self.userID = params.fetch("userID", "")
9     end
10
11     self.title = params.fetch("title", "").strip
12     self.message = params.fetch("message", "").strip
13     self.date_posted = Time.new.strftime("%Y-%m-%d %H:%M:%S").to_s
14
15     # If user is moderator or admin, post doesn't need to be moderated
16     if params.fetch("account_type").to_i == 2 || params.fetch("account_type").to_i == 3
17       self.is_moderated = 1
18     else
19       self.is_moderated = 0
20     end
21
22     self.image_link = params.fetch("image_link", "").strip
23
24     # If image link is empty, post doesn't have an image
25     if params.fetch("image_link", "").strip.empty?
26       self.is_image = 0
27     else
28       self.is_image = 1
29     end
30
31     self.universityID = params.fetch("universityID", "")
32 end

```

Listing E models/post.rb

```

16  def exist?
17    other_user = User.first(username: username)
18    another_user = User.first(email: username)
19    (!other_user.nil? && other_user.password == password ) ||
20    (!another_user.nil? && another_user.password == password )
21  end

```

Listing F models/user.rb

```

26  def type_of_account
27    # Return name of account type
28    if account_type == 0
29      "Reporter"
30    elsif account_type == 1
31      "Viewer"
32    elsif account_type == 2
33      "Moderator"
34    elsif account_type == 3
35      "Administrator"
36    else
37      # If a wrong number is somehow entered, return nothing
38      ""
39    end
40  end

```

Listing G models/user.rb

```
101 .button {
102   padding: 0.5%;
103   background-color: var(--colour-background);
104   margin: 1%;
105   border-radius: 5px;
106   border: 1px solid;
107   border-color: black;
108   font-family: sans-serif;
109   width: 10%;
110   text-align: center;
111 }
112
113 .button:hover{
114   box-shadow: 0 4px 6px 0 rgba(0, 0, 0, 0.2), 0 4px 20px 0 rgba(0, 0, 0, 0.19);
115   background-color: var(--colour-accent2);
116 }
```

Listing H pubic/styles/style.css

```
50 <#University filter doesn't appear on mod's post moderation feed>
51 <% unless @page_title == "Moderate Posts" && @user[:account_type] == 2 %>
52 <optgroup label="By University">
53   <% University.each do |uni| %>
54     <option value="uni,<%= h uni.universityID %>,<%= h uni.university_name %>"><%= h uni.university_name %></option>
55   <% end %>
56 </optgroup>
57 <% end %>
```

Listing I views/common/feed_filter.erb

```
9 <% post_user = User[post.userID] %>
10 <% university = University[post.universityID] %>
11
12 <article class="feed_post">
13   <p><b>'<%= h post.title %>'</b></p>
14
15   <% if post.is_guest == 1 %>
16     <p><em>By guest user.</em></p>
17   <% else %>
18     <p><em>By <%= h post_user.username %></em></p>
19   <% end %>
20
21   <p><%= h post.message %></p>
22   <p><b><%= h university.university_name %>. Tags:</b></p>
23
24   <% unless PostTag.where(postID: post.postID).nil? %>
25     <% PostTag.where(postID: post.postID).each do |tag| %>
26       <p><%= h Tag[tag.tagID].tag_name %></p>
27     <% end %>
28   <% else %>
29     <p>None </p>
30   <% end %>
```

Listing J views/feed.erb