# Workshop: Petstagram

## Part I – CBV's

### CBV's

**Refactor the code** in your project, so you use **Class-Based Views** instead of **function-based views**

Try adding some **admin functionality** (admin views)

## Part II – Users

For this part, we need to do some major refactoring. We want to change the way the app can be accessed, and we want to improve the user experience by adding features for user registration, login, profile customization, and restricting content to only certain registrants.

You will be provided with a changed home-page.html file.

Also, you will be given a new template - login-page.html.

### Creating 'accounts' app

Create a new **app** called **'accounts'**, where the **registration**, **login**, **logout,** and **user profiles** will be handled. Refactor the written code so that all the functionality for the user profile is moved to the new app.

### URLs

Add/ refactor the following **urls**:

* **'localhost:8000/accounts/create-profile/'** - create profile page
* **'localhost:8000/accounts/login/'** - login page
* **'localhost:8000/accounts/profile/<int:pk>/'** - profile details page
* **'localhost:8000/accounts/edit-profile/<int:pk>/'** - edit profile page
* **'localhost:8000/accounts/edit-password/<int:pk>/'** - edit profile page

### Modifying Models

* **Pet** model - add a new **field** called **user** that is linked to a **user profile**
* New **Like** model - add a new **field** called user that is linked to a **user profile**
* **Profile** model - find a way to extend the profile to have a **username** and a **password**. They should meet the standard on access control and authentication.

### Modifying old templates

* Add **login/logout/register** links in the navbar and implement the needed **template logic**
* In the **pet\_detail** template, implement the logic, so the **edit** and **delete** buttons are **visible only** in the pet **belongs** to the **user**. If not, display the **'heart'** button (**filled** if the user **has liked** the pet **already**, **not filled** if the user has **not yet liked** this pet). Display the **comment form** if the pet does **not belong** to the user.

### Restrictions

All users can see the Dashboard and the details for each photo.

Only anonymous users:

* Can see the "Home" page button
* Can open **Home** **page**, **Create** **Profile** **page**, and **Login** **page**

Only **logged**-in users can:

* Like the pet photos - one user can give only one like to a photo.
* Add pets to their profiles
* Create pet photos to their profiles
* Edit/ Delete their pet info and photos.
* Not like their own photos (the heart button should not be visible for their photos)

### Home Page

The home page is only visible if the user is not logged in. If a logged-in user tries to open this page, he/she should be redirected to the Dashboard Page.

There are **four buttons** on the **Home Page**:

* **"Home"** button in the navigation bar that links to the **Home** **Page** (same page)
* **"Dashboard"** button in the navigation bar that links to the **Dashboard Page** (old Home Page with Profile)
* **"Login"** button that links to the **Login Page**
* **"Register"** button that links to the **Create Profile Page**

A group of dogs

Description automatically generated with low confidence

**Create Profile Page**

This page is only visible if the user is not logged in. If a logged-in user tries to open this page, he/she should be redirected to the Dashboard Page.

There are **three new fields**:

* **"Username:"**
  + Placeholder **"Enter username"**
  + CSS class **"form-control"**
* **"Password:"**
  + Placeholder **"Enter password"**
  + CSS class **"form-control"**
* **"Confirm Password:"**
  + Placeholder **"Enter password again"**
  + CSS class **"form-control"**

When the button "**Create**" is clicked, the profile is **created**, and it **redirects to the Home page**. Then the user should log in to enter the app.

Graphical user interface, application

Description automatically generated

### Login Page

This page is only visible if the user is not logged in. If a logged-in user tries to open this page, he/she should be redirected to the Dashboard Page.

Template: **login-page.html**

There are **two fields**:

* **"Username:"**
  + Placeholder **"Enter username"**
  + CSS class **"form-control"**
* **"Password:"**
  + Placeholder **"Enter password"**
  + CSS class **"form-control"**

When the button "**Log in**" is clicked, the user is logged in, and it **redirects to the Dashboard page**.

Graphical user interface, application

Description automatically generated

### Photo Details Page

Anonymus users **can't like** the photos. They **can't edit/ delete** the photo:

A picture containing text, screenshot, cat

Description automatically generated

A logged-in user can like other photos. He/ she can only like one photo once:

A picture containing text, screenshot, cat

Description automatically generated

The owner **cannot like his photo**. He can **edit/delete** his pet photo:

A picture containing text, screenshot, cat

Description automatically generated

### Log Out

Logged-in users see an additional **"Log out"** button. When the user clicks on it, he/ she is logged out and redirected to the home page.

### Additional

* Try to add a Password Change page (CSS and HTML of this page is optional)
* Try adding a "Forgot password?" functionality that sends an email to the user to change his/ her password