# Workshop: Petstagram

## Part II – The Pet Models and Views

### Creating the 'pets' app

For our **pet logic** we will have a **separate app**, so let us **create it** first

### 

Add the new app in the **INSTALLED\_APPS** in the **settings.py** file of the **project**

### Creating the Models

In the **models.py** file create the **Pet** model. Each **pet** should have:

* **type** – some of the following: "cat", "dog", "parrot"; max length = 6
* **name** – max length = 6
* **age** – positive number
* **description** – text field
* **image\_url** – URL field

Now in the **models.py** file create **Like** model. Each like should have:

* **pet** –foreign key to a **Pet**

Let us also **register** our models in the **admin.py** file in the app, so we can see it in the **django admin**

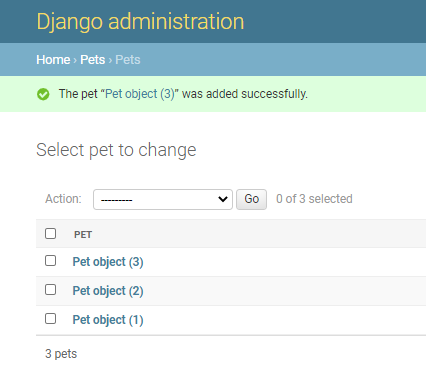
### Make Migrations

The next step is to **make migrations** to the database, since we created some **new models**

Let us also create a **superuser**, so we can **create some pets** using the **django admin**

### Create Pets

**Create** some **pets**, so we can **test our views** that we will create later

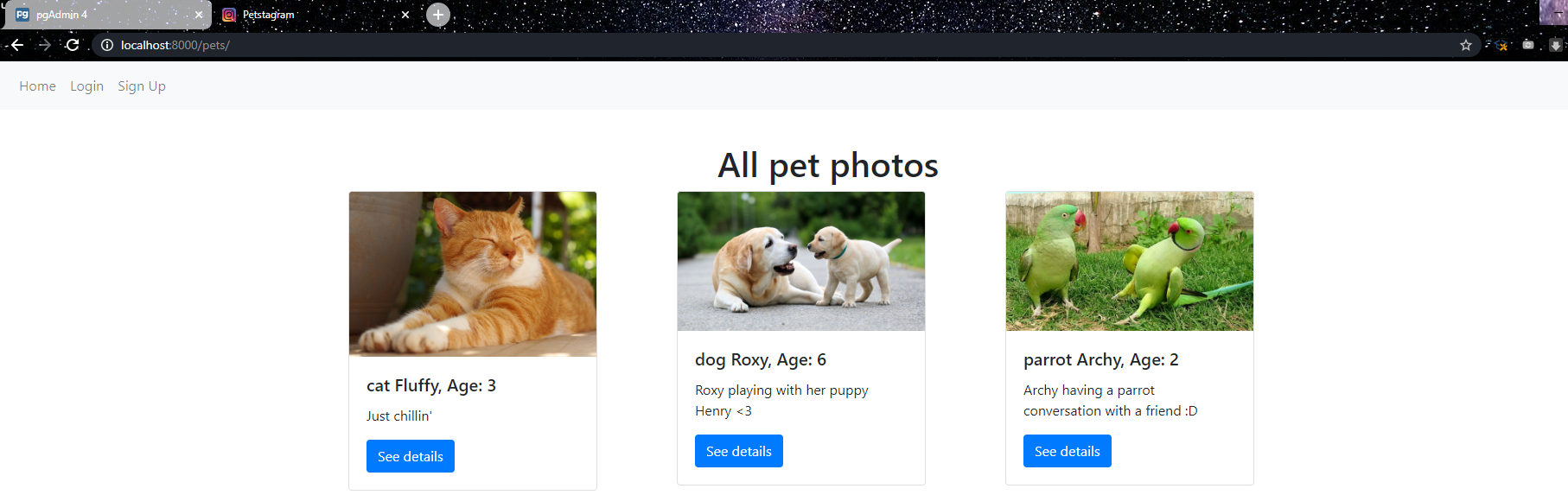


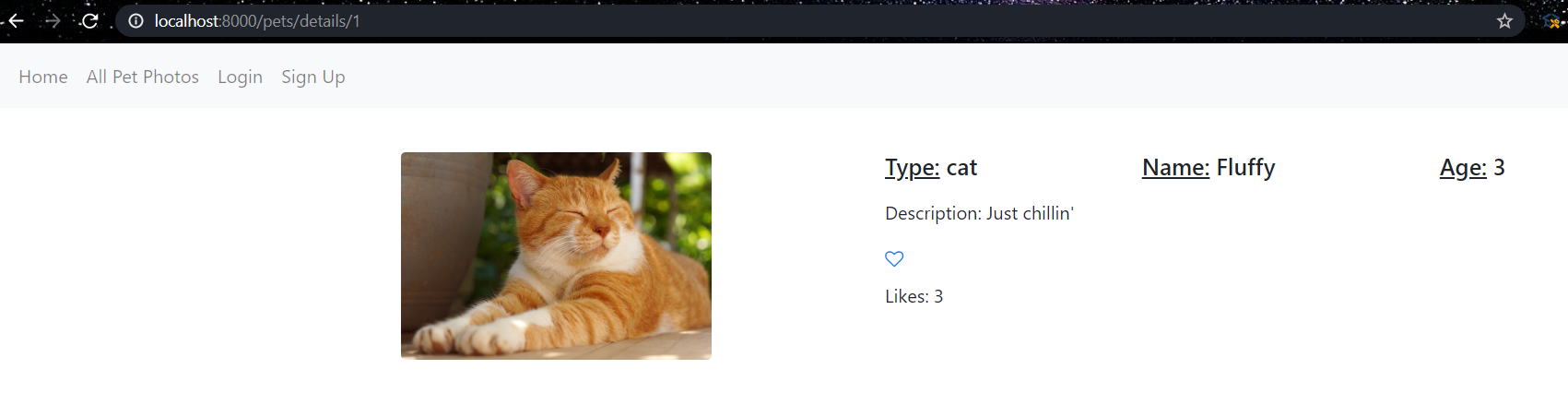
### Creating Some Views

Let us now start **creating** some of our **views**. We will create the **pet\_all** view and the **pet\_detail** view

* **'localhost:8000/pets/'** – list all the pets (using the **pet\_all** view and a template **'pets/pet\_list.html'**)
* **'localhost:8000/pets/details/<int:pk>'** – display the pet with the given **primary key** (using the **pet\_detail** view and a template **'pets/pet\_detail.html'**)
* **'localhost:8000/pets/like/<int:pk>' –** write a view that will create a new like and redirect the user to the details page again

Use the provided **html files** and **add** the needed **logic** in them, so they work as expected





### Add Additional Links

In the navigation edit the **'Home'** link to lead to the **landing page** and add another **link** that leads to **'pet-list'**

