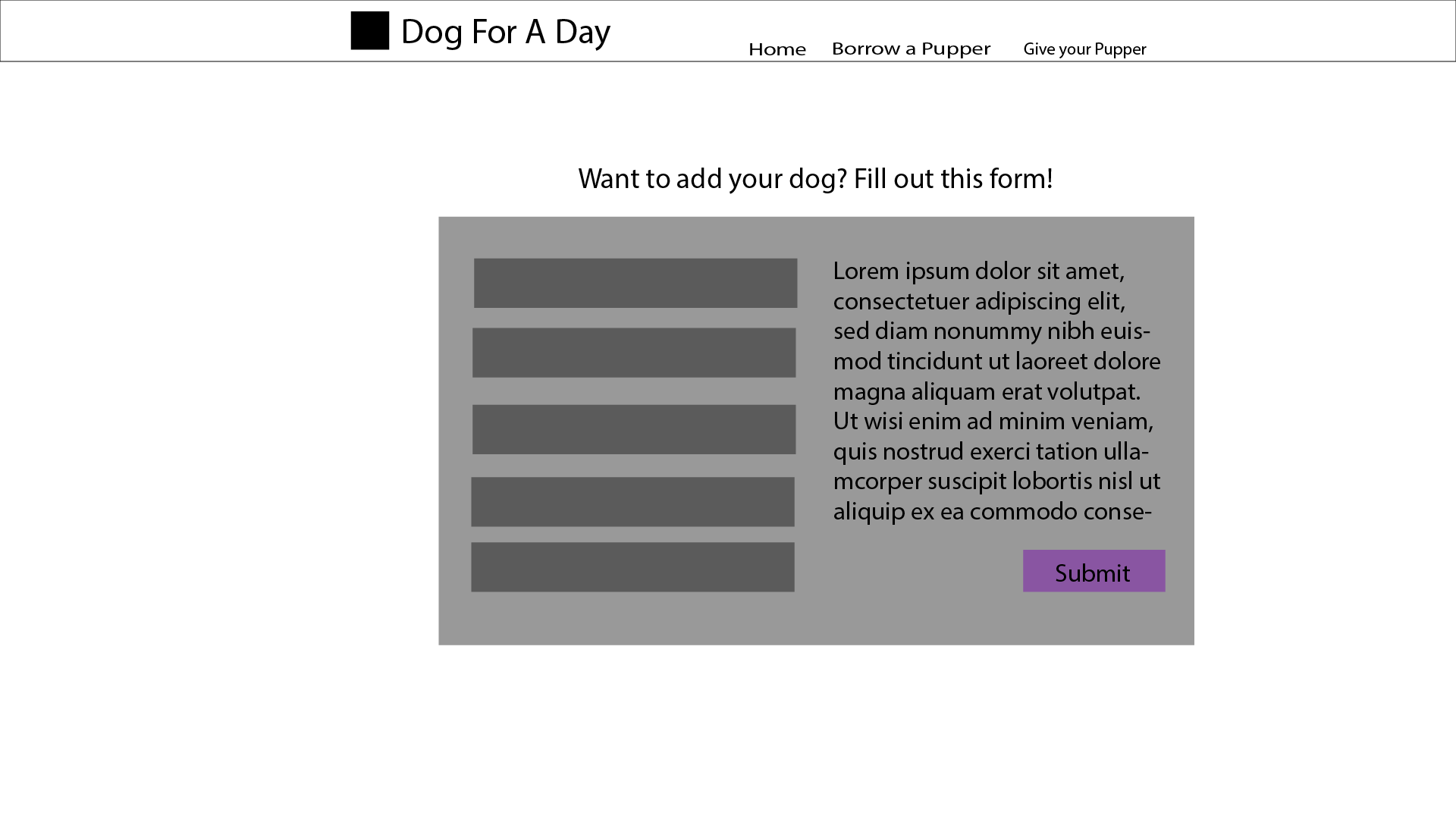
Version 4.0

Task 0: Explain what you are doing/ going to accomplish

Within this version I am going to set up a page for dog owners to add a new dog to the page.

Task 1: Sketch interface design



A nice clean form to be filled out by the user, this should be very basic and easy to fill out.

Task 2: Identify any classes required

Class Dog

Will need the variables, Name, Age, Description, Breed, Available, Gender

Task 3: Identify information to be displayed

This page will display the input forms for the Dog class and all of its variables in the dog\_list

Task 4: Identify user inputs

The user will input all of the information about their dog.

Name. age, gender, breed and friendliness

Task 5: Identify any constants or existing data if required

No constants for this version

Task 6: Identify indexed data structures

Data dictionary called dog\_list is what the new dog classes will be inputted to.

Task 7: Determine what calculations are necessary

No calculations needed in this version

Task 8: Develop a modular structure for your program

Set route to (‘/new-dog-page)

Set view to (‘new-dog-page’)

Set showcase function to:

Set var data to be dictionary of dog\_list

Return data to page

Set route to (‘/new-dog-action)

Set view to (‘new-dog-action’)

Set showcase function to:

Set name to be request.forms.get(“name”)

Repeat this step with each variable in the Dog class

Set new\_dog to be class Dog with the parameters of all of the above defined variables

Append new\_dog to dog\_list

Set route to (‘/new-dog-success/<dog\_id>)

Set view to (‘new-dog-success’)

Set showcase function to:

Dog\_id equal to integer of dog.id

Set found dog to none

For dog in dog\_list

If dog id is equal to dog\_id

Found\_dog = dog

Break

Set data to dictionary of dog

Return data

Task 9: Define the functions identified

Task 10: Address any relevant implications such as usability, functionality, legal/ethical requirements.

Within this version, I will need to create a website interface that is easy to read and simple to understand as many of the users may be older/unfamiliar with technology. I also need to follow the general rules of design when it comes to websites with colours layout etc. Buttons need to be clear and laid out, everything should make sense.

It need to be functional, it should first fufill its purpose and secondly look aesthetically pleasing,

No copyrighted images. No illegal or explicit images etc.

Task 11: Document test cases for testing the program

Task 12: Refine the plan

When adding a new dog, then going into the dogs personal page, I got the error. Dog.friendliness is not applicable. Str type.

This meant I needed to add the new dogs friendliness rating as an integer. It was as simple as adding int() around the code friendliness = int(request.forms.get(“friendliness”)

I also needed to add the code to the new-dog-action page to return the data of the dog so it could show it to the user

New PseudoCode

Set data to a dictionary of new\_dog

Return data to the page

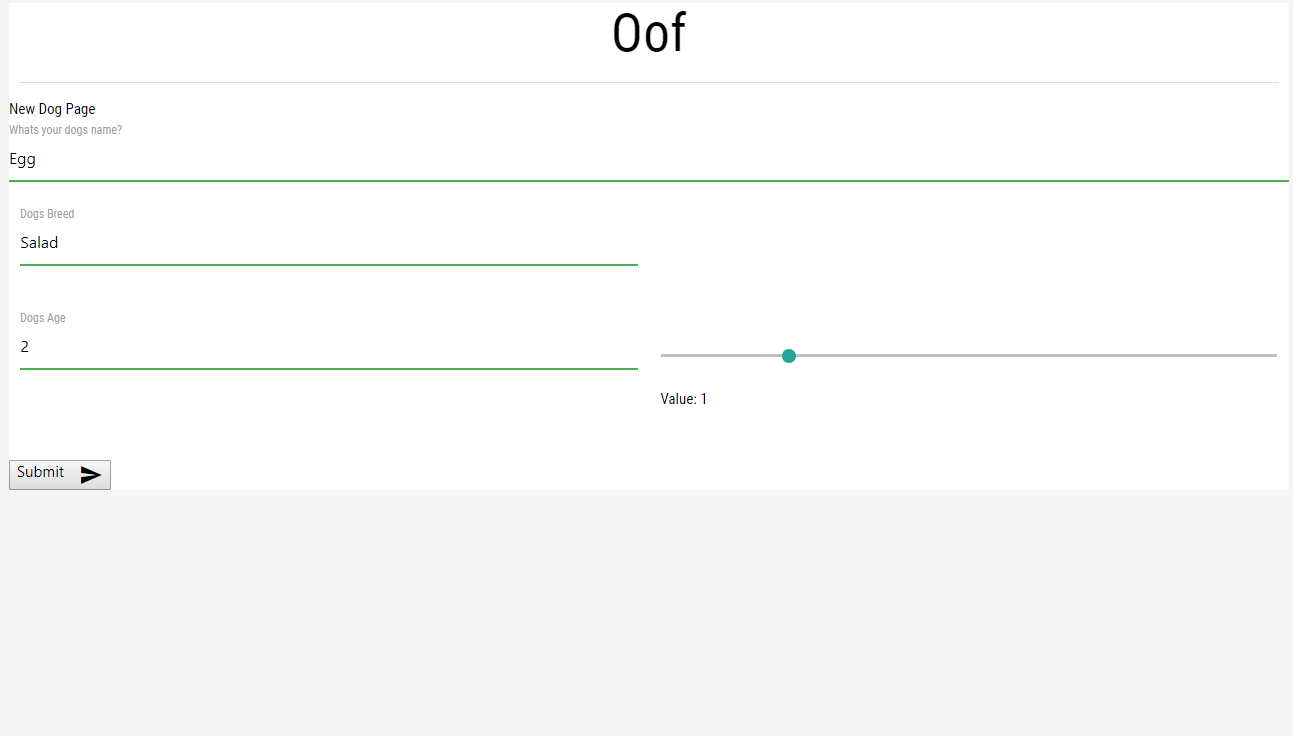
This meant that the new dog could now show up

To add the friendliness slider, I needed Javascript code to show the user the current value of the slider, this I copied from online <https://www.w3schools.com/howto/howto_js_rangeslider.asp>

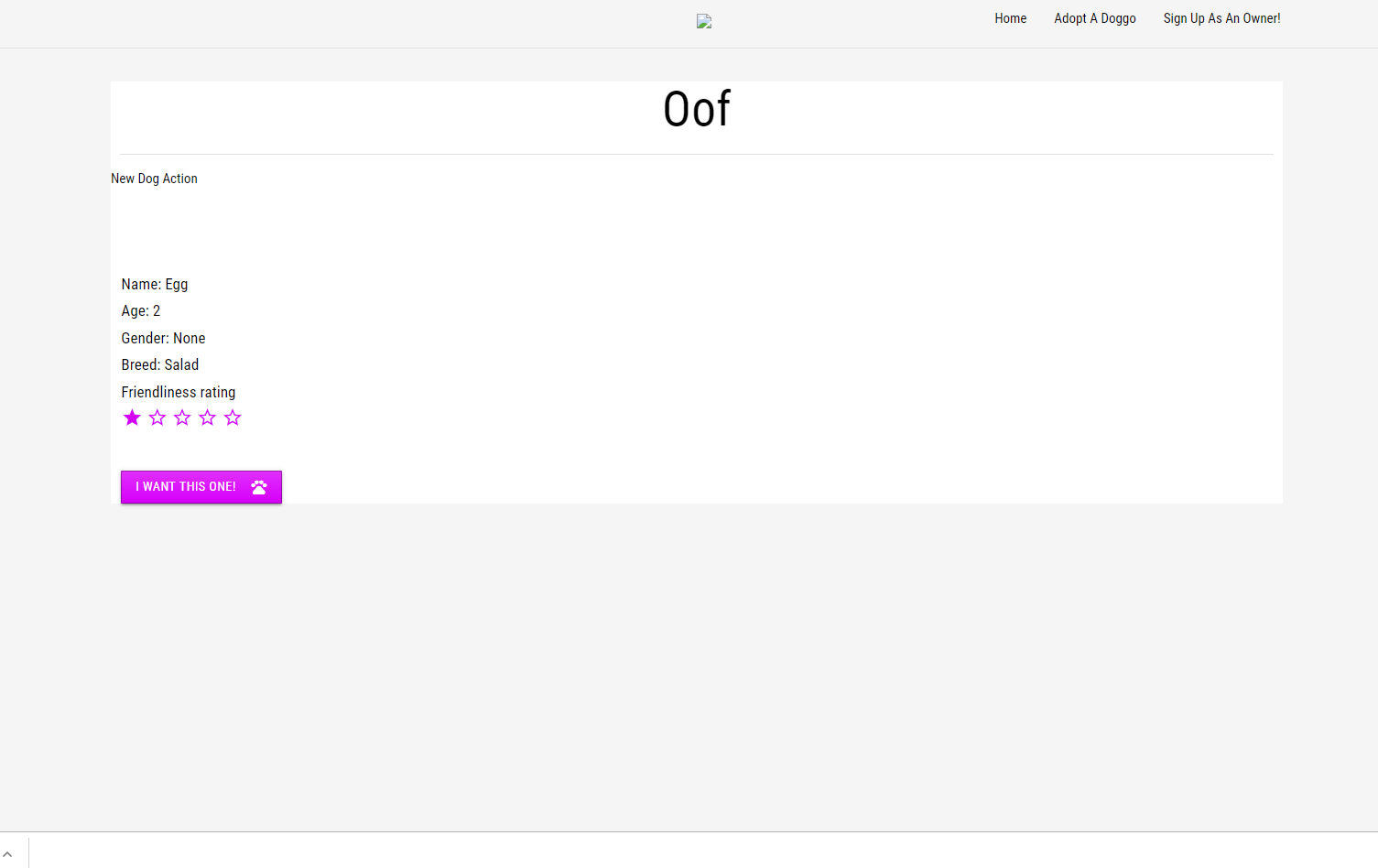
It was an easy fix for a problem.

* I also realised within building the pages, that the new-dog-success page was uneeded, as it was just splitting the page up unnecessarily, So I deleted it.

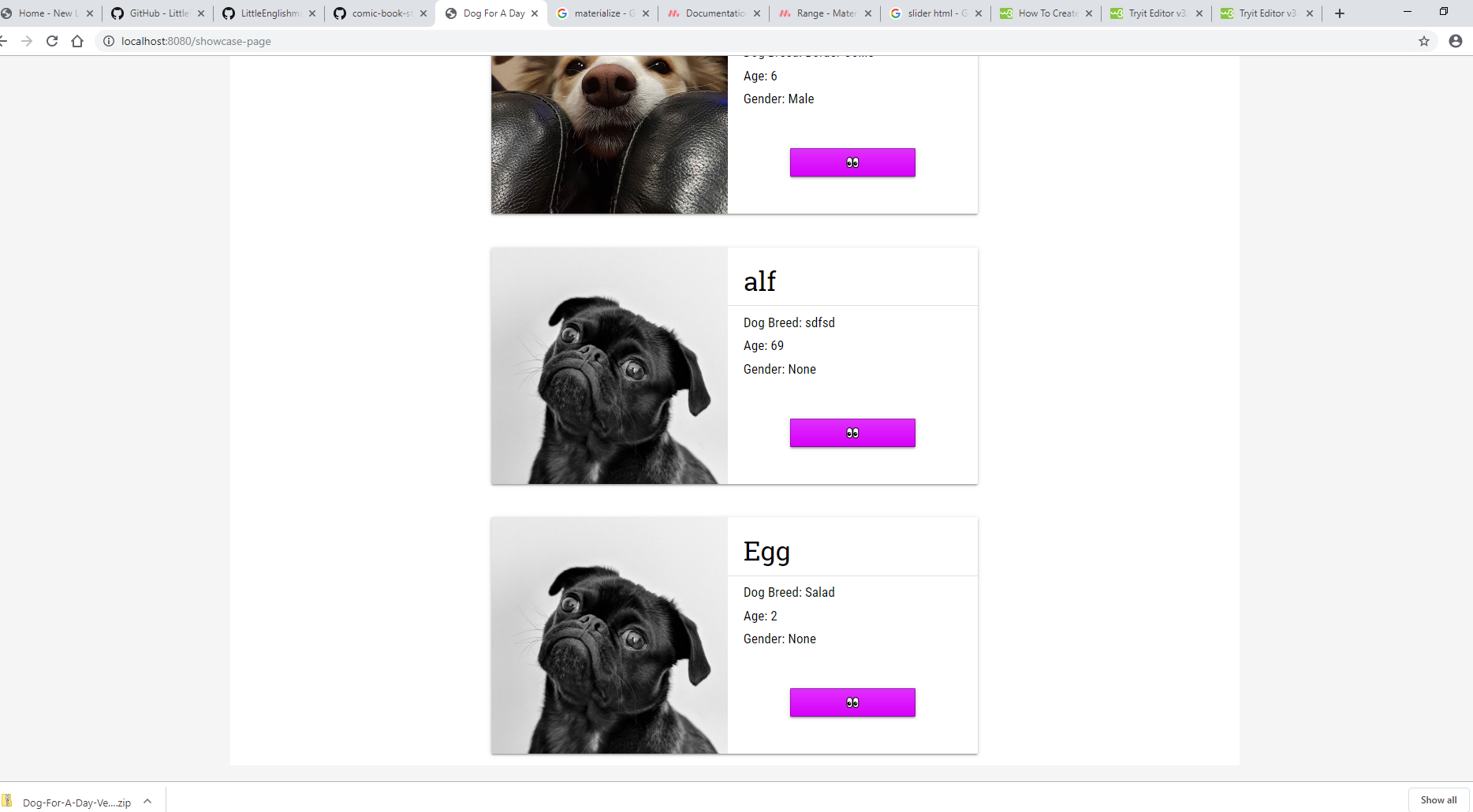
Task 13: Document testing

Can I add a new dog? 

Yes. It works



Does it show up on the dog list? / on the showcase page?



Yes it does.

Does it throw errors with any inputs? No, as I am using materialize, it locks the user out form entering the wrong inputs. Name and Breed can be anything the user desires. (These are custom so numbers are allowed). Age is restricted to being a number, no matter how old they are, 0-infinity

While the friendliness is a slider limited from 0-5, no other inputs are allowed. This stops the user form being able to enter inputs that are outside the desired range so there is no need to test them. Each value works otherwise.

Task 14: Evaluation