

// TODO: Team Name

Nugget

<https://github.com/mcculloughsco/CompSci326TermProject>

Note: Github submission is on the 'Master' branch, not the 'dev' branch.

LOGIN CREDENTIALS FOR A TEST USER

Username: TestUser // Password: testing123

The Team:

Arwa Farrag (arwaGypt), Scott McCullough (mcculloughsco), Pinak Kapoor (pinakkapoor), Malachai Purgahn (mpurgahn), Emily Goroza (egoroza), Sangmin Yun (joeyun)

Project Overview:

Our application is a virtual pet site in which users can create a Nugget and interact with other Nuggets through battles. Work hasn't been done yet on the Chat feature, but this is something that we intend to work on in our Team Choice part as we can use APIs to add this functionality. Nothing has really changed in terms of what we hope to accomplish by our final submission. **It's important to note that some of our forms aren't currently working or either not fully functional on the pages My Nugget, Shop, and Battles. These forms are a lot more difficult than we previously anticipated and will require a bit more time to complete. The Shop page will not work completely.**

Design Overview:

The login/logout largely is based around the default Django authentication system. Where things were a little tricky was with our index page and our User model. The index page contains both the login and signup form, so configuring those two forms to work simultaneously was a challenge, but works. Secondly, we had created a User model that contains data that extends and exists separately from Django's User. We made Django's user push data into our User (now named Profile) and everything works fine.

Problems/Successes:

The first hurdle to overcome was having two User models, our User (now called Profile) and Django's user. This altered a lot of how we were accessing the User and retrieving User related data so changing all of that code over took a while. We also ran into a lot of problems with forms. We encountered problems with having multiple forms on one page (especially when it came to combining login and sign up on the same page), editing data in multiple models from a single form, and creating dynamic forms that generate based on dynamic data (i.e. since a user can have a variable number of items in their inventory, generating forms to accommodate the dynamic nature of the inventory). Finally, Thanksgiving break made working collaboratively difficult since we weren't in the same physical location anymore, which made editing common files such as views and forms difficult.

Team Choice:

Our team choice is to implement chat into our application. Right now, it is not implemented at all because we were waiting to use an API to make it easy to add. We aren't quite sure which API we will use, or what the complete functionality of the chat feature will be, but we should be able to get that to work. If we have time, we may add some quality of life features such as password reset via email, and also rounding out the usability of our application.

Individual Write Ups:

Arwa Farrag:

I helped brainstorm with the team about how the forms should look and debate which parts of our project needed a form and which didn't. I worked with Pinak on the Battle views in order to determine how to structure it and what functionality we wanted for it. Then I collaborated with Emily and Malachai on the battle page to figure out how to create the forms and pull the necessary data required to request users for a battle and generate updated stats for the user post battle. I think I contributed to around 18% to the project as a whole.

Scott McCullough:

I worked mainly on getting user authentication working, and combining our user model with Django's user. This was a bit tricky because we have both the signup and login form on the same page, and converting all over of models to accommodate the new User model setup took some time. Beyond that, I helped others where I could in getting forms on other pages to work, and helped with cleaning up some other pages like the logout page, dedicated login page, etc. I think I probably contributed around 20% to the project.

Pinak Kapoor:

I worked alongside Arwa on the Battle page views to determine the proper formatting and establishment of the page. I was unable to accompany Arwa alongside Malachai and Emily due to travel plans; as I had my flight to return onto campus around the times scheduled to collaborate. I also helped generate ideas in how certain forms should be implemented and with some brief troubleshooting amongst the team. I believe I contributed about 12% to this project as I was unable to meet up for our final gathering due to travel plans.

Malachai Purgahn:

Over the break I worked on creating two separate forms for the shop.html page, the forms would allow users to buy/sell items from their inventory. This involved editing the views for the Shop as well so that the form could function correctly. I then assisted Emily and Arwa on the Battle page. Our goal was to have it completed by this submission, but we ran into some difficulties. Overall I would say I contributed 15% to the project.

Emily Goroza:

I mainly worked on getting all of the data integrated into the home.html page for a user that is logged in as well as creating the working forms for the create-a-nugget.html page. I experimented with jQuery so that a user creating their nugget can view the customizations as they select different nugget options, and got the "color" option to change upon selection. There was some difficulty with displaying the form and how to save the data because of some of Django's restrictions, since the create a nugget page consists of 2 separate forms for 2 separate models. I also worked on creating redirects for users logged in/not logged in and taking appropriate redirects based on this condition for each page. I also attempted to work on the Battle page which we had lots of difficulty with due to how our data model is structured and the complexity of the battle setup. (i.e. interaction with other users) We are aiming get this up and running by the final project submission. Overall, I contributed to about 20% of the project.

Sangmin Yun:

I helped with implementing and debugging the user's inventory in views.py, forms.py, and nugget.html page and added a new "Price" integer field under the Items model in models.py (Which is necessary in order to have a 'sell' functionality in the user's inventory forms and give purpose to our coins, buy functionality, and shop page). In forms.py I aided in creating the 'InventoryForm' class which constructs the form for items in a user's inventory, displaying three options of, "feed", "discard", and "sell". Then

in nugget.html, I helped debug and display each item as its own separate form with its name and quantity displayed under the .png picture by iterating through each item in the array of items we have in our database that is associated with the current logged in user and pulling the correct data. I then helped with adding and debugging the nugget function in views.py. I had to polish the current code to pull the correct profile data from the database rather than using hard-code. Lastly, I helped with adding additional code that would correctly initialize and pre-populate our formset for each individual item in the users inventory with a hidden name so that we could keep track of which form/item was being submitted so we could execute the correct functionality that was chosen for that specific item. I believe I contributed about 16% of the work to the project.