// TODO: Team Name

Nugget

https://github.com/mcculloughsco/CompSci326TermProject

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

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. Two aspects of our project that have changed slightly comes down to battles and chat features. The issues we foresee with these two aspects of our project is dynamic updating of a given user's view. For example, implementing chat so that when one user posts a message, the receiver sees the message without refreshing the page might be difficult to implement. These are things that might be implemented in the final submission, but might not due to complexity.

Design Overview:

*Our model picture is in the same directory as this pdf.
Our data model is a bit complex, but the major routes are between a User, their Nugget, and aspects of their Nugget. A User is our base model, in which User information is stored. From that, a Nugget is tied to a User, and aspects such as player attributes, inventory items, etc are attached to a user's Nugget. The main URL routes haven't changed. Our base page is /nugget. This is the index page. Since we don't have login information implemented it, selecting LogIn->Login will allow you to access the meat of our content. From here, each item on the top bar has a URL associated with it, and each corresponds to a template. The only view not implemented is the chat feature as we intend to use libraries to ease that implementation.

Problems/Successes:

Our team encountered some problems applying what we completed on the Django tutorial homeworks with our models. Specifically, the relationships between different models and whether or not we were referencing them the right way and creating the right associations in order to get our admin database to connect to the models properly. Another problem we encountered was having to delete then remigrate and resync the database because we continued to update our associations between the models. Another problem we encountered was implementing some of the javascript necessary to complete the views. Our team succeeded in communicating and collaborating. We met outside of the class work time multiple times and assigned tasks to each person. We were also successful at helping each other debug and re-learn the logic every time we changed the models. We can improve on going to office hours at the beginning of the new project release and making sure we can apply the skills we gained from the homework to our project correctly and not waste time implementing it the wrong way.

Individual Write Ups:

Arwa Farrag:

I helped brainstorm with the team on how the layout of the models should be. I helped work on the first draft of the models, specifically the Nugget model. After series of edits to the model page, I worked with Emily on completing the Shop model and then the views file in order to fetch the correct data and parse it so that it returns a list rather than a string. I also completed the problems/successes portion of the write up. I think I contributed to around 15% to the project as a whole.

Scott McCullough:

I helped mainly with organizing the models.py file, helping to gather data in the views.py, and helping out on the html templates when necessary. I also served as the coordinator for the backend, including generating and managing the database/admin panel. Most of what I did involved creating the models, and working out templates for others to follow in implementing other models/views/templates. I think I contributed around 35-40% to the project.

Pinak Kapoor:

Throughout project 2 I helped work on models.py and views.py and did some work on html templates, such as the battle page and with creating dummy data sporadically through pages. I helped create a better understanding of the relationships among models and with the diagram of our models. I also helped debug at certain parts of our code. I believe I contributed about 10% of the overall work of this section of the project.

Malachai Purgahn:

I assisted in creating models and views as well as brainstorming ideas on how to fix a few of our logical errors. I mainly worked on the first implementation of the battles view and began working on the friends list view.I created dummy data for an opponent so that we could implement our battle views/models.
Percentage wise I contributed to ~10% of the total project.

Emily Goroza:

I worked on models.py, views.py, admin.py, and the shop.html and create-a-nugget.html templates. In the models, I worked on creating Foreign Keys and relationships between the models, and figuring out how to extract the data in views.py. In the shop.html, I created implemented the data from Items into the shop and displayed them with a custom JS script. For the create-a-nugget.html, I implemented the data from Nugget Attributes into the create a nugget form. I contributed to around 35%-40% of the project as a whole.

Sangmin Yun:

Over the course of project #2, I assisted with helping create the models.py and views.py files. In the models.py draft file, I implemented the inventory and shop models along with fixing a slight error under the battles model. In the views.py draft file, I helped with creating the home views page so that it would correctly pull live data instead of hardcoding dummy data. Also, went back to fix and edit the battle views page to match up with our updated template and fixing existing errors. I believe I contributed about 13% of the overall work put into this project #2.