Status Summary (15 points):
Project Title: Encyclopedia of Life

Team Members: Akhil Tadiparthi, Dylan Kriegman, Srujith Yeturu

Work Done:

Akhil:

- Finished the Project 6 Update document
- Contributed to writing the database interface with the web application (DatabaseWrapper/Facade Pattern)
- Tried out experimental Django implementation in Python

Dylan:

- Wrote AnimalFactory class code as well as other main classes such as AnimalPage
- Setup React implementation with multiple components
- Wrote HTML and Javascript for listing species on the home page and being routed to their corresponding page as well as foundations of species-specific pages
- Tried out experimental Django implementation in Python

Srujith:

- Worked on Project 6 Update document
- Assisted with the planning and development of code.
- Helped with database interface such as the Encyclopedia Navbar class and Facade Pattern

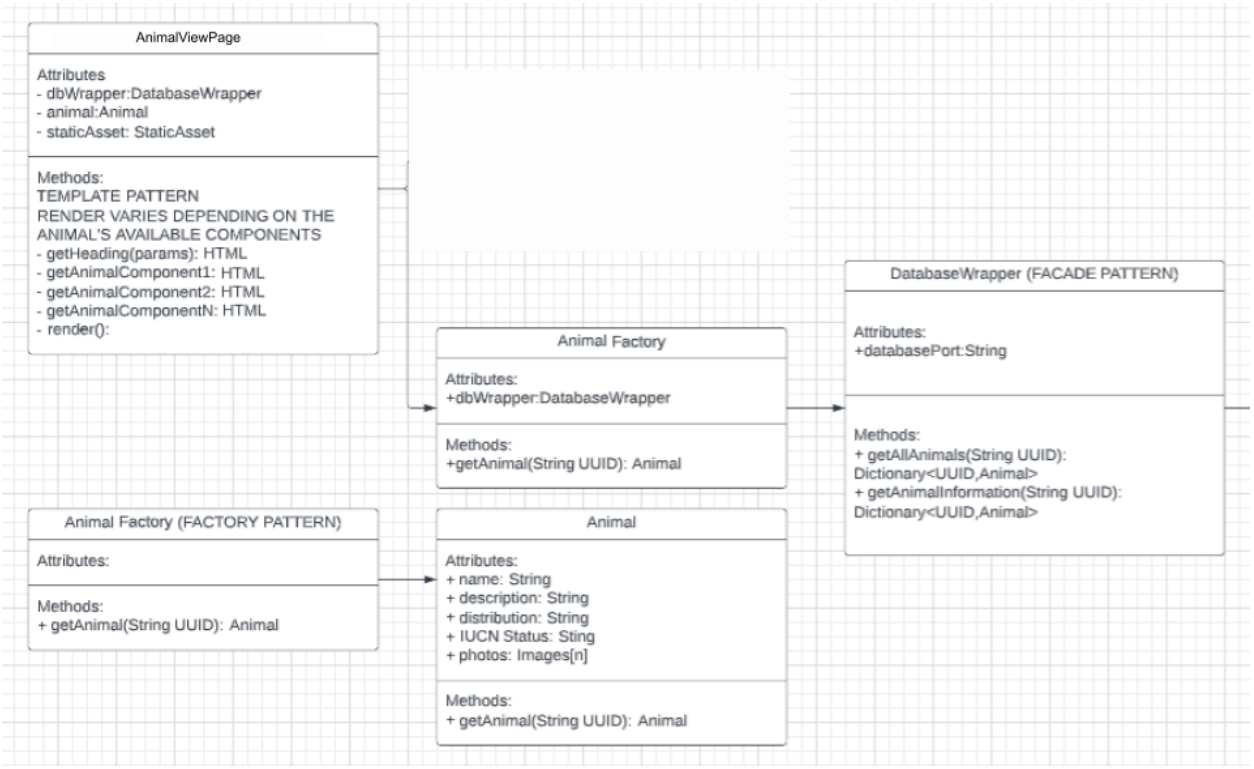
Changes or Issues Encountered:

We had to make some changes from Project 5 with the technologies we will be using. Originally, we were supposed to use a Docker container for front-end and another container for handling server requests, and a Django Framework. Now, we won't be using Docker containers anymore and will be using the React framework with Javascript to implement our design patterns(design patterns using class implementations in js). Other than these changes the rest of the project/the layout will be the same as Project 5. We will still be using MySQL for our database implementation and how it will interact with the other classes and design patterns. After looking into the project, we fine tuned the tables we will be using in our MySQL database.

Patterns:

So far, we have implemented the Facade pattern as the DatabaseWrapper class which simplifies interacting with the backend database. We also have our Template pattern done through the AnimalViewPage class, in which specific steps executed in the render method vary depending on the information available in the animal. Lastly, we implemented the Factory pattern in the AnimalFactory class, in which an Animal object is generated by querying the database using the DatabaseWrapper class. However, we still have a little bit more work with our singleton pattern and the EncylopediaNavbar class. We have to change it a little to fit more of a theme for our web application and make sure it is aesthetically pleasing. The page needs more color and images. We will update the Singleton and EncyclopediaNavbar by the next project due date.

Class Diagram (10 points):



Plan for Next Iteration (10 points):

Currently, we don't have the singleton pattern implemented with full functionality yet, which is essential in the next iteration. We have to edit the EncylopediaNavbar class which contains the single instantiated EncylopediaNavbar object which has an unchanging snippet of HTML for the Navbar. In addition, we have to clean up the CSS formatting to make sure that the web application is more visually pleasing. We need to make the page more aesthetic by color and images. The UI of the page will be updated as well. Lastly, we also have to add more animal data to the database and fill it with animals so that we can dynamically create all the web pages with the animals. The database needs some more finetuning as well. We also need to update the Home Page to host the list of animals which when clicked on will dynamically create the webpage for that specific animal. We also need to extend our implementation of the template pattern to allow for different rendering of animal pages depending on the given properties in the database.