Georgia, Reece, and Alejandro

Database Design

May 10, 2025

Reflection

Our group worked well together. The three of us worked together to come up with the veterinarian management system and decided the tables, variables, and requirements for the system. We met in person to decided who was going to do what work after our initial planning phase. While together Alejandro and Georgia worked on the ddl and getting it working in the database. Concurrently Reece worked on annotating the requirements for our database. After this Georgia did all of the coding in python and html to get the database to show up and be interactable on the front end. However, Alejandro and Reece helped by working on fixing methods when they weren't working, spotting errors when needed, and providing insight to how we wanted our frontend to look. As stated, most of the work was done while we were together in person. However, when we were not in person, we used texting and discord when we needed to exchange files or documents. Additionally, also used github to share the code we had created seeing as we would need to use it to turn in our project anyways.

During the project we did use AI, but only to help us code. The reason we decided to do this is because it gave us a starting point for our front end and integrating both ends. As per our AI prompt report you can see that we used AI to create an entire file system for our Vet system, however most of that code did not make it to the final version. This was due to the code not serving our purposes, our goal was to see what code produced which features. We then used these files as code examples to complete the files and methods needed for our desired front end. The AI was good at producing code examples that we could use and learn from. Based on our experience we'd say there was a lot of human intervention needed for it to serve our purposes. However, we could have worked with the AI to adjust the code until it properly served our purposes and our requirements. This too would have required human intervention at some point, not including the human describing exactly what we needed and what about the code needed to be fixed/adjusted.

The biggest issue we had while integrating the front and backend was the datetime type, we could not figure out how to get it to transfer from python to sql so we ended up switching a variable type from datetime to text to make it easier. We also changed a few attributes in our tables, we noticed there were things that just slipped our minds. Examples are pet date of birth and clinic name to the pet and clinic table, so those were added while we were coding the front end. The hardest part of integrating the database into a software product for us was the repetition of code and having the tables show up on the front end. The repetition was difficult because the same code could be used for all the tables just slightly tweaked. We noticed a lot of errors slipped through when we tried copying and pasting the code and then making the changes. Getting the

tables to show up was also a little tricky for the same reason, there were a lot of errors from pasting and if one variable was off it wouldn't work which happened a lot. A final difficulty for us in turning the database into a software product was how it would be presented in a way that would be most useful. We tried to make the front easy enough to understand that it can be helpful for both inputting new information and accessing existing information with out experience in the web app. Additionally coming up with an idea that was useful, within our capabilities, and timeframe was difficult.