Sam Wilkins

Quinn Lanik

CSCE 155N

Final Project Paper

5/3/2021

For my GUI coding project, I decided to write a program that outputs the nutritional facts of a custom cattle feed mixture after a user inputs the percentage of each ingredient used in the feed. I am an agricultural engineering major, and I come from a background in production agriculture, so this project was not only interesting to me, but it was also something that could certainly be useful to me in the future. This project also appealed to me because my dad manages a grain elevator and cattle feed business, and I thought this program could be beneficial for him or his customers. I ended up investing a lot of hours into this project, and I’m really pleased with the way it turned out.

I honestly do not remember how I came up with this idea, but I think it all started with wanting to do something with cattle feed. At my dad’s business, they mix their feed by dumping all the ingredients into a mixer with a payloader. It works well, but a lot of businesses have a more precise process that utilizes conveyor belts to transfer the feed ingredients to the mixture. With a payloader, it is easy to accidently dump way more of a certain ingredient into the mixer than intended. I knew I wanted to write a program that would be useful for relatively low-tech feed businesses like my dad’s, and this idea about displaying the nutritional facts came to me. It doesn’t solve the original problem I was talking about, but I knew the program would still be very useful. It is quite easy for a cattle producer to find nutritional facts for one feed ingredient. In my dad’s case, if a rancher can’t look it up online, they could just call my dad and ask him what the protein content of corn is, for example. However, there isn’t a great, time-efficient way to find the nutritional facts of an entire feed mix. I thought it would be relatively simple to create a program that could do this for any customized feed mix, so I did. Both Quinn and Laurel approved this idea before I started.

My first challenge in the design process was figuring out all the information I would need to make this code successful. I already knew I could use weighted sums to find the nutritional facts of an entire feed mix, but I didn’t know what nutritional facts were relevant or what ingredients I wanted to allow the user to select from. Fortunately, I found a website that helped me with all that (<https://www.beefmagazine.com/nutrition/feed-composition-tables-how-use-2017-data-mix-best-feed-your-cattle>). This website explained all the relevant nutritional facts for cattle feed. It also had a table that displayed the nutritional facts for a ton of different feed ingredients. After asking my dad about it and leaning on my own background knowledge, I picked eight common feed ingredients to give the user the option to choose from.

Once I had my plan formulated, the coding finally began. My final code is pretty long, but it’s fairly simple and easy to understand. I was a little bit confused on how callback functions work going into the project. I knew how a callback worked for a push button because there is only one way to interact with a push button. However, I didn’t know what event triggers a callback function for an edit box, and that ended up being really important for my code. Fortunately, it was easy to figure that out. Another major issue I had was figuring out how to create a running total of the percentage of feed that had been entered by the user. Quinn explained that I needed to use a for loop to create the edit boxes and another for loop in the callback function to make that work. Once that was explained, it was not too difficult to understand. Besides those two instances, I found it quite simple to write my code. It took a long time to enter the nutritional facts for each feed ingredient and to write the code for the weighted sum, but both were fairly straightforward. Of course, I made a lot of little mistakes in my code that I had to fix as I went along, but I didn’t have too much trouble with that. Most of the problems were simple to fix on my own.

Completing this project definitely helped me learn a lot more about GUIs. Because we had the freedom to choose our own projects, I was also able to see some ways I could use GUIs in the future. My favorite characteristic of GUIs is that anyone can use them regardless of how much coding experience they have. My dad and his feed customers have absolutely no experience with MATLAB, but the program I wrote is still very useful to them. I think this concept is why I will use GUIs in the future. As an engineer, I could use GUI coding to write a program that could predict how long if will take to complete a certain process on an assembly line. Then, someone from human resources or accounting could use the program just as easily as me. This is very important because the program would be as useful for them as for me. Additionally, there is a possibility I could end up in a career in production agriculture, and I could see GUIs being even more valuable in that field. I could create a GUI to calculate how long a certain farming operation should take. I could also do more GUIs with cattle feed; I could make something that would show the quantity of feed I would need every week based on how many cattle I’m feeding and how big they are. I’m confident that knowing how to create GUIs will be very beneficial for me in the future no matter what my future holds.