Self-Assessment

Ted Tippets

CS-499

SNHU

# Showcase and strengths

Completing the course work for this program has helped me to build a fully functional software. The software I developed could have most of the code reused to create new similar software’s. This course work allowed me to show case the skills I have gained since the beginning of this course. From the beginning when I wrote everything statically and in one java class. Now I use more of a dynamic approach and multiple classes to make the writing of the software easier to maintain and build upon. For example, in the beginning of my classes I would have written this line this way

String myDriver = "com.microsoft.sqlserver.jdbc.SQLServerDriver";

String url = "jdbc:sqlserver://PC-Home-001:1433;databaseName=Zoo;user=admin;password=@dmin";

I would have used this in multiple places which would have been bad practice. Now I will put the drivers in their own class and call them like this.

//calls the Driver class

Driver Driver = new Driver();

//inputs the driver variable into animal class

String myDriver = (Driver.myDriver);

//inputs the url variable into the animal class

String url = (Driver.url);

Class.forName(myDriver);

By doing this the way I do now if I want to change the database name, location, or port I can do that in one place. I do not have to worry about the software not working right because I must find that one, or two places I might have missed or wrote incorrectly. The same is true for the SQL queriers also. I created a class just for the queries that way they are easy to change and only must be changed in one place. One item I did for security it implements a hash algorithm that way if someone got into the SQL database it would not be stored in plain text. I think if I was in a team environment, I could make this software bigger and better. If I had a team to collaborate with, I could work on tying add an LDAP to this software and could send emails, or messages when there is an issue that needs addressed. The stakeholders for this should see increased productivity, and less time from the ease of the software. Since there is less attention that needs to be paid to the text files that are storing the information. The software has more information so the can start to look for trends. If the animal cage is always dirty by week three then they could set up a preventive measure so that this is not an issue because it is known that it will need to be cleaned. If the trends show that the food is always low or out before the next batch is ordered then it would show that the food needs to be ordered sooner, or more ordered and the same time. Data structures help with the storing and manipulating of the data. The algorithms will help solve problems. Overtime more data structures, and algorithms could easily be added to the software. The software can be added to and changed to fit any needs in the future also. The security in my software is the encryption of the passwords. If the SQL database is ever compromised the passwords would not be easily broken into.

# Summarize

My artifact fit all the categories into one artifact. I had to develop the software and all the functionality for the database from the software. The software will not allow for data to be input into the SQL data base if some items are blank based on other criteria. For example, if the cleanliness is marked as fail in the jcombo box and the comments box is blank the program will throw an error and tell the user to fill in the comment box with the problem. The program will also not allow any input if there are any blanks in the required boxes. The artifact starts out with the logon page. Depending on the username is where the user will be taken. If the user is admin, they will be taken to a page to be able to add usernames, and passwords. All other acceptable names will be taken to a page where they can pick either the animal, or the habitat database. The algorithm covers putting in passwords and validating them to get into the software. The SQL database stores all the information for the database, and the usernames, and passwords.

I think by adding the hash algorithm and using the sql database with them I think this showcased what I have learned throughout my time learning computer science from SNHU.

The software starts by opening a logon screen. Based on the user will be where the software goes next. The software compares the username and password hash to the data in the username SQL database I created. If the username is admin with the correct password the software will go to the admin page to allow for the admin to input new usernames, and passwords. The logon page has a hash algorithm written in it to hash the username to verify to compare it to the hash in the SQL database. When the admin inputs a user and password the software uses a hash algorithm to hash the password going into the SQL database. The SQL database only allows unique names. All other users will go to the start page to select the animal, or the habitat database. In the animal, and habitat database they function mostly the same. All the boxes except for the comment are required. The comment box becomes required if the health concern is checked yes, and/or the feeding schedule is marked none. In the habitat if the food supply is marked low, and/or the cleanliness is marked fail then the comments becomes required to input into the database. The user is given an error, or a success message so they know if it worked. They can also use the show DB button to verify that it was inputted. The software will also clear all the boxes after the data is inputted into the SQL database so that the user does not have to erase it before they type again. This also keeps the user from creating multiple records of the same item. The jcombo boxes are also reset back to -1 to be blank so the next input is the actual input needed. The jtext panes also allow for the tab button to go to the next box. This makes it convenient for the user, so they are not having to use the mouse continually. The user can type an animal/habitat in the animal/habitat box and click show animal/habitat to see all the data for that animal/habitat. The user can place an animal/or habitat in the box and click the alarms and it will show the most current entry and if there is a concern for that animal/habitat. There is also a show DB to see the most current 1000 items in the database.