Grady Salzman

Shant Haik

Jacob Schilmoeller

Daley Keister

KABSU Inventory Application Developer Documentation

The purpose of this document is to orient a new programmer to the KABSU Inventory Application codebase. The following will detail where important aspects of code are located, diagrams of code structures, and how to properly set up the development environment and manage the database.

Setting up Development Environment

To get the necessary files clone or download the GitHub repository located at :

https://github.com/GradySalzman/KABSU-Inventory-App

In this repository you will find 3 folders. One called “Database Files” which contains files relevant to setting up the database and populating it with sample data. One called “Documents” which contains user documentation, this file, and other relevant documents. Finally, one called “WpfApp” Which contains code relevant to the user interface and interacting with the data in the database. These code files will be described further in later sections.

It is recommended to use Visual Studio 2019 to edit the codebase and test the codebase for your development environment. Visual Studio for Mac is not supported, it is suggested to use the remote.cs.ksu.edu vm to develop on this codebase with a Mac. You can download and learn how to use Visual Studio at https://visualstudio.microsoft.com/. Once the project is open in Visual Studio you can run the project to interact with the user interface and data. A successful start should yield the following screen

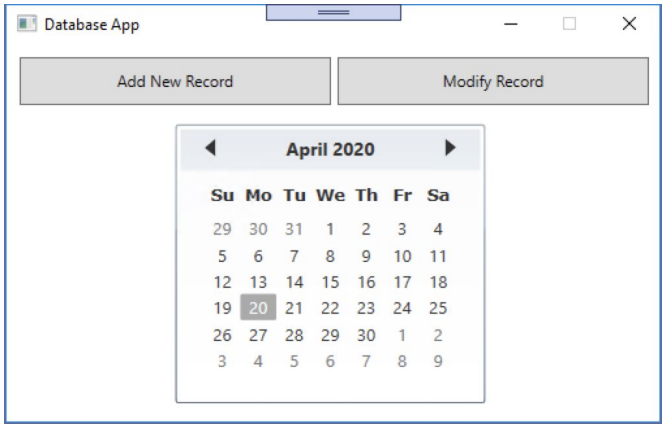

Figure 1. Successfully starting KABSU Inventory application

Database

The mySQL database is currently hosted on the computer science servers. It can be accessed through the link: https://phpmyadmin.cs.ksu.edu/. The username and password can be obtained by contacting Grady Salzman at gradysalzman@gmail.com. This allows you to view the data but it is suggested to use the scripts located in "Database Files" to interact with the data. The tables are populated by converting the excel files to .csvs. The .csvs are then used to populate the tables. "Database Files" contains the following folders:

Database Insertion Files: This folder contains sql procedures to insert data into tables into the database. These procedures are used by the user interface to add new data and update data in the database. For example “InsertAnimal.sql”, will insert/update an animal record in the database.

Database Retrieve Files: These files are sql procedures used by the user interface to load data from the database. For example “RetrieveData.sql” is used by the search feature to search the data and return the result.

Database Sample Data: This folder contains csv files with sample data that was given by KABSU. These sample files are used to initially populate the database with sample data.

Database Setup Files: This folder contains the TableSetup.sql which is used to initially set up the tables in the database.

The database’s structure is shown in the Figure 2

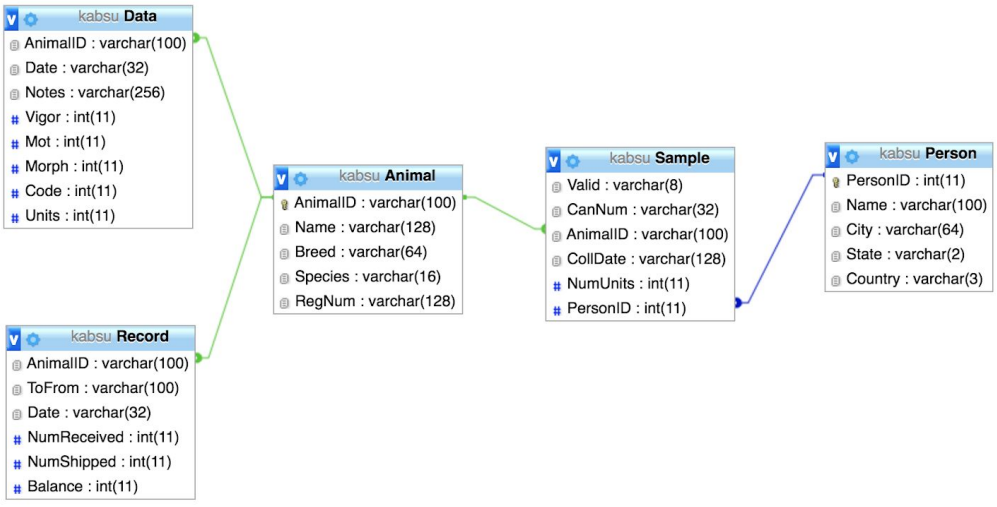

Figure 2: Database diagram

WpfApp

The following class diagram details all classes and how they are associated. Files for these classes are located in the WpfApp/WpfApp folder.

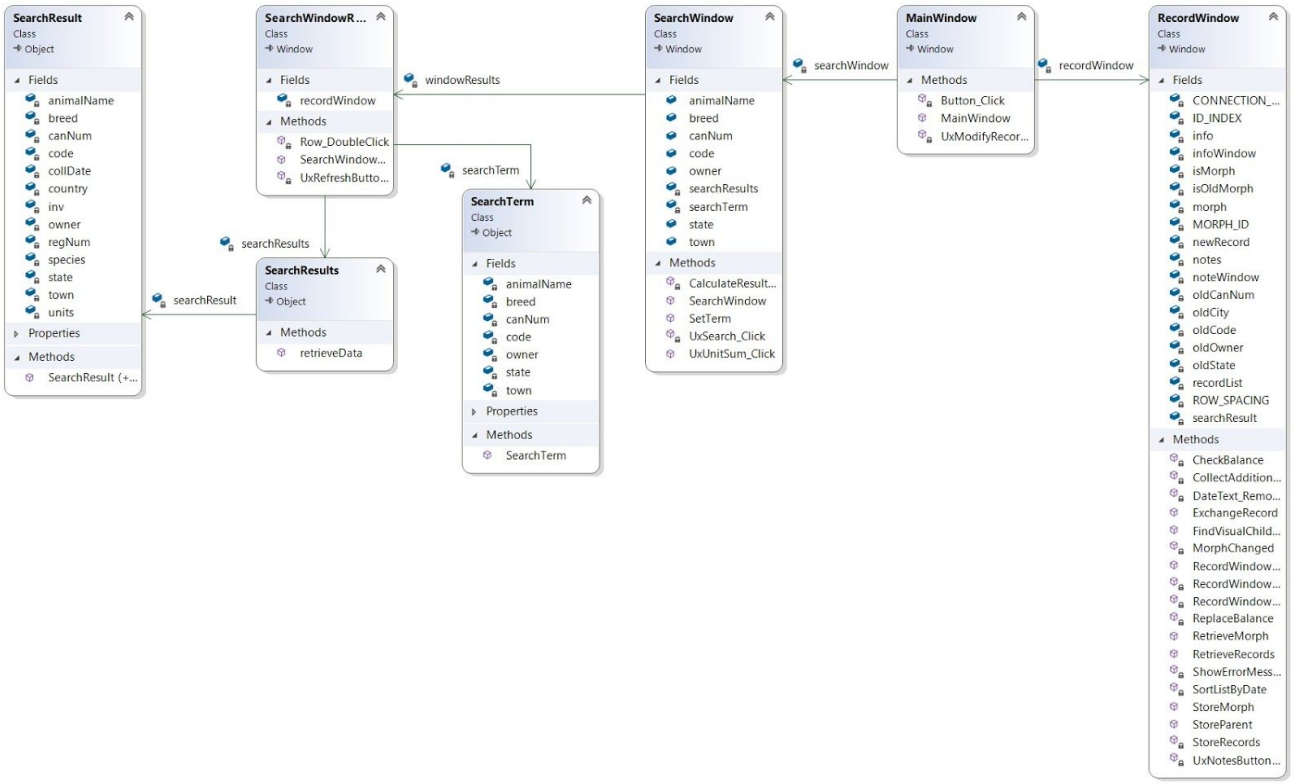

Figure 3. Class Diagram

The following descriptions describe the broad use of each class and is supported by comments inside each file for more detailed information on implementation.

MainWindow: Code for user interface that supports access to RecordWindow and SearchWindow through clicks.

SearchWindow: This code will set the search term value used in the search that is used in SearchWindowResults. It will display the SearchWindowResults on the click of the search button.

SearchWindowResults: This class populates the results in a window. Row_DoubleClick method functions to open a record card containing further information on the sample. This class also contains code to delete a selected row and any connecting parent data.

SearchResults: This class contains the logic to connect and search the database using the SearchTerm object and stores the result in a SearchResult variable.

SearchResult: An object to store results of searching samples and is used to populate the SearchWindowResults.

SearchTerm: An object containing possible search terms and is used for database query in SearchResults.

RecordWindow: This class contains code for populating a record card that is being viewed. It includes the prompt for the additional info screen and will store the information in the database. It includes several functions StoreParent, StoreMorph, etc. That will update information in the database if the user changes the information on the user interface. Retrieve Records will connect to the database to get records matching a specific animal ID.

Future Considerations

The next main goal for this project is to get a copy to the user so they can test and give better feedback on the application. To do this the database needs to be moved from the CS department to the animal science department to ensure confidentiality of information if the user inserts actual client data. Contact information for the animal science IT department is Tim Carson, his office number is 785-532-1191 and email tcarson@ksu.edu. He will be able to assist in the transition. One thing to note is that the connection string when connecting to the database will need to be changed to the new connection string in the following classes: SearchWindow, SeachWindowResults, RecordWindow.