

Grady Salzman

Shant Haik

Jacob Schilmoeller

Daley Keister

KABSU Inventory Application Specifications Document

Introduction

The purpose of this document is to instruct developers on design and technology choices made for this project. First, the problem this application is intended to solve will be documented. Then the technology used to solve this problem will be described along with architecture diagrams in order to demonstrate the design of the system. Finally, the primary goals we, as developers, intend to complete in this application.

Problem Statement

The Kansas Artificial Breeding Service Unit (KABSU) is self described as the one-stop shop for all your livestock breeding needs. For the purpose of this application we are only concerned about KABSU's services of storing samples taken from cattle. KABSU keeps large amounts of samples for many different clients. Previously in the past they used paper and excel spreadsheets to keep track of the inventory. This becomes difficult to manage as searching through large amounts of paper documents or excel spreadsheets has

become too time consuming. This current method will also not be able to scale in the future with KABSU's growing inventory of samples.

The main problem the KABSU Inventory Application will solve is to provide better ease of managing and searching this inventory. The second problem this will solve is the current process would become unmaintainable in the future due to the growing records. KABSU expects the amount of samples they will keep track of to grow exponentially in the coming years. This application will provide the means scalability with KABSU's growing inventory of samples.

Workflow Diagram

SEE ATTACHED FILES:

Workflow-Diagram.PNG

Shows how the steps the user would take when using the system

Technology

In order to implement the user interface we will utilize the Windows Forms library. In order to implement the front-end, C# will be used. C# will also be used to communicate from the front end to the database. C# was chosen because the client's use windows as their main operating system. This application will be hosted locally on their machine. This will also allow us to take advantage of windows forms. We believe windows forms will allow us to be able to implement the user interface. C# was chosen because of the group's

initial knowledge of the language. The application will only run on windows machines and will not be a web-based project.

The database is stored and interacted with using MySQL. We have chosen MySQL due to it not costing use compared to other choices. We will use stored procedures in our application to handle queries of data during runtime.

Architecture

SEE ATTACHED FILES:

Class_Diagram.PNG

Shows a detailed class diagram of the class architecture for the application.

Database-Diagram.PNG

Shows a detailed database diagram of how the different tables in the database interact.

Sample Data Used

The primary bulk of data used to populate the database was given from KABSU. CSV files are hosted on github containing sample data for animals (Animal.csv) and related clients (People.csv). For now, this is enough information to implement and test features we are developing. For more sample information Leon Prather should be contacted directly if needed in the future.

Primary Goals

First and foremost, our primary goals are to continue to implement and refine features that help to solve the problems outlined in the Problem Statement. We want to make the user interface more intuitive to navigate. This will evolve refining the look of the user interface with our customer to find the best design. It will specifically involve implementing the ability to have entries that can be selected from a search and have it auto-populate a pop up with more information.

Another primary goal we have is to streamline the installation process so that it is easier for us to get our customers up and running, but also so that in the event of a tragedy, they can also get themselves up and running without our assistance. This will also help possible future developers in their ability to get the application running on their machine and start development.

Finally, we hope to if time allows provide a way to edit the database from multiple machines. This will allow it to fall in line with how the customer may use the system. To do this, a user login system may need to be implemented as well as auto-save and manual saving features.