**OOP2 lab work 2015**

Your task is to design the GUI for an animal shelter. How you design your GUI is up to you but it must fulfil the requirements. You will initially just concentrate on creating a functional attractive GUI. Remember to include in your GUI options for saving to a file and to a database and also restoring the system. This assignment is worth 30% of the total 50% continuous assessment.

**Note** :-

**When creating a system like this it is a good idea to create files for any information needed in the system. It is a bonus to have a maintenance section to allow you to add and delete any lists in the dropdown menus or required elsewhere in the system.**

**Main Application:** This will maintain a List of Animals.

**Section A**

**Step 1:** Get the classes written without any GUI and test them. 10%

**Step 2:** Save and load (from a file) any items needed by the various

Dropdown menus etc at runtime. 5%

**Step 3**:

* 1. **Implement** the **GUI** part leaving out for the moment the Save option.
  2. In your **design** remember to include all sections, including **reports** and **maintenance**.
  3. Get the GUI working first and add the functionality later.
  4. You need to be able to move from screen to screen and back home again in a logical

fashion.

* 1. Add the general functionality.
  2. Test it and ensure that functionally everything is working.
  3. Remember to include exception handling.

35%

**Step 4:** Add the save and load options. Save all data to a file and load all from this

file at start-up.

10%

**Step 5**: Generate the reports as requested. 20%

**Make a copy of your system. Include comments in your code where necessary.**

**Hand in Section A for marking on or before 8th April 2016 @ 18:00 hours.**

**Section B**

**Step 7:** Using the copy version of your system **edit it (Arraylist and files are no longer needed)** so that instead of storing the information in a file you store it in a **database**. Remember to normalise your data. Any **DBMS** can be used.

**Week 12** 20%

**Hand in Section B for marking on or before 8th May 2015 @ 18:00 hours.**

*A demonstration of the program in the lab session is required by each individual after each specific step or at the end of each section.*

**DBMS:** Database Management System

**Lost Animal Section**

Functionality: You must be able to

**TASK A**

* **Add** an animal to the system giving each lost animal a unique ID, their name, description, the date lost and the location. A photo of the animal may also be given. Details of the owner of the lost animal.
* **Remove** an animal from the system if found. Owners details are kept on file for sponsorship events.
* **Display** all animals details. Allow owner’s name and phone number to be displayed when required.
* **Quit** the system without saving.

**TASK B**

* **Reports:** 
  + Display all animals (either dog/cat) lost in a certain location.
  + Display all cats lost in a certain location on a specific date.
* **Save** all the information to a file or files and then **Quit**
* **Restore**  all the information

**TASK C**

* **Save** all the information to database and then **Quit**
* **Restore**  all the information

**Found Animal Section**

Functionality: You must be able to

**TASK A**

* **Add** an animal to the system giving each found animal a unique ID, a name (if they have a tag), description, the date found and the location.
* **Remove** an animal from the system if owner comes forward. Get Owners details and keep on file for sponsorship events.
* **Display** all animals details
* **Quit** the system without saving.

**TASK B**

* **Reports:** 
  + *Display all animals (either dog/cat) found in a certain location.*
  + *Display all animals found between certain dates organised by gender.*
  + *Display all animals (either dog/cat) found in a certain location between certain dates.*
* **Save** all the information to a file or files and then **Quit**
* **Restore**  all the information

**TASK C**

* **Save** all the information to database and then **Quit**
* **Restore**  all the information

**Adoption Animal Section**

Functionality: You must be able to

**TASK A**

* **Transfer** an animal to this section if not claimed after a period of a month.
* **Name** the dog if no tag present.
* **Add** a photo of the dog.
* **Create** a list of people who register an interest in adopting. **Get** their details and keep on file for sponsorship events.
* **Allocate** the animal to a family once they have meet the animal and agree to go ahead.
* **Ensure** the animal is neutered, sprayed and chipped include dates etc.
* **Register** when the animal is available to go home as they may need extra help to get ready.
* **Remove** an animal from the system only when they leave the rescue centre.
* **Display** all animals details
* **Quit** the system without saving.

**TASK B**

* **Reports:** 
  + *Display the details of all animals ready for adoption organised by name. (1 report)*
  + *Display the details of all Cats/Dogs ready for adoption by age. (2 reports)*
  + *Display all puppies up for adoption who are in training. (1 reports)*
* **Save** all the information to a file or files and then **Quit**
* **Restore**  all the information

**TASK C**

* **Save** all the information to database and then **Quit**
* **Restore**  all the information

**General Reports:**

*Display the details of all sponsors in the system. (1 report)*

*Display the details of all Cats and Dogs in the system organised by age. (1 reports)*

Remember to include all sets/gets in each class

Adoption Lost Found

AnimalShelter

.

AnimalList

List: Animal

AnimalList()

add(Animal)

remove(Animal)

etc……..

printList()

Animal

ID: int

age: int

aType: String (Dog/Cat)

colour: String

gender: boolean

description: String

name: String

picture:jpeg

breed: String

animalCat: Category

Animal(int, String, boolean, String)

(include all sets/gets)

toString(): String

print()

Category

date: Date

contact: Person

Category(Date)

Lost

location: String

Lost(Date,String)

allAnimals(location)

Found

location: String

Found(Date,String)

Adoption

neutered: boolean

chipped:boolean

vaccinated: boolean

status: String {inTraining, ready}

reserved: boolean

Adoption(Date)

…..etc.

print()

Person

name:String

address: String

phone: String

email: String

animalID: int

Person() …….. etc

toString() : String

print()