LAB221Assignment

Type:	**Long Assignment**
Code:	**J2.L.P0002**
LOC:	**400**
Slot(s):	**N/A**

Title

Pet Healthcare Management

Program Specifications

In this assignment, you are required to build pet healthcare management application, in the form of a desktop application.

The program has basic functions: add - edit – delete registration information of a pet owner.

You are required to use

- The basic components to design interfaces
- The tabbed pane to organize registration management on the one screen.
- File is used to store information.

Program organization must clearly separate functions according **to MVC model.**

USING RMI WRITE INFORMATION TO FILE

Create an application using java swing to create frames like description below

- **RegistrationServer.java**

```
public interface RegistrationInterface extends Remote{
    boolean createRegistration(RegistrationDTO dto);
    RegistrationDTO findByRegistrationID(String id);
    ArrayList<RegistrationDTO> findAllRegistrations();
    boolean removeRegistration(String id);
    boolean updateRegistration(RegistrationDTO dto);
}
```

- **File name: RegistrationData.txt**

Registration information of a pet owner is shown following table

Field Name	Type
registrationID	String
fullName	String
age	Integer
gender	Boolean
email	String
phone	String
address	String
numberOfPet	Integer
symptoms	String

Features:

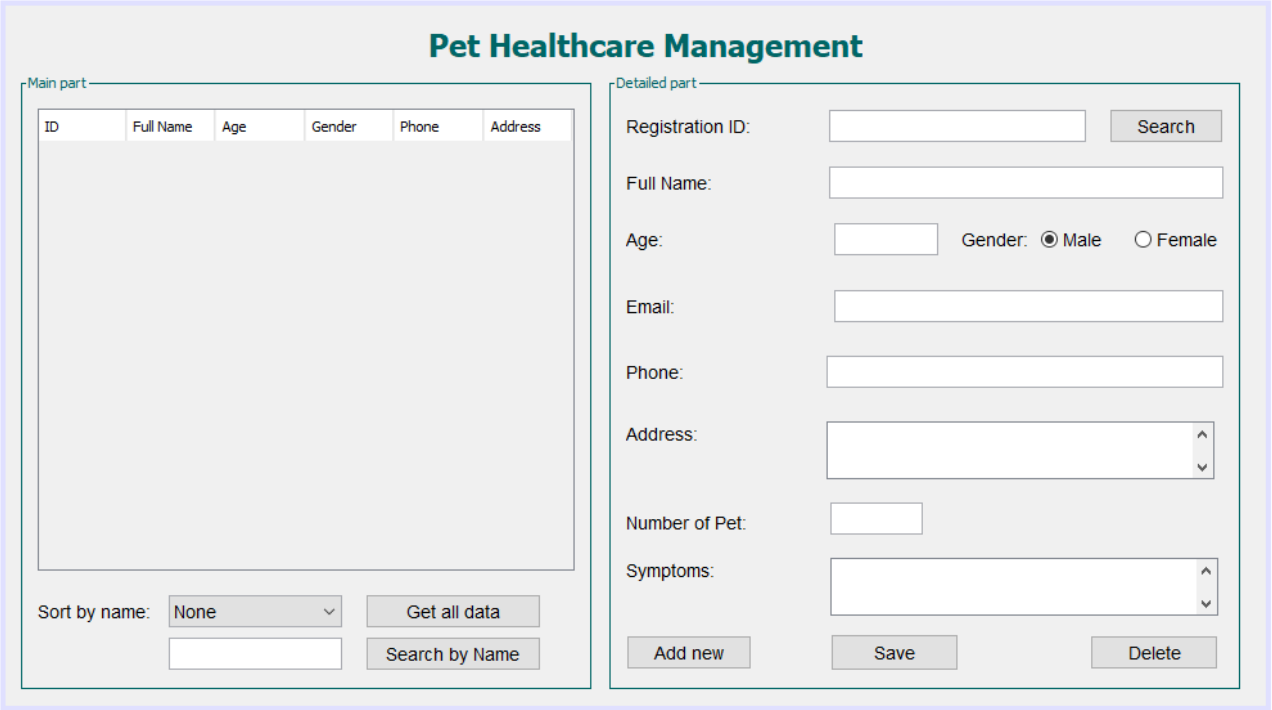
This system contains the following functions:

- **Verify constrain of following data type fields – 50 LOC**
  - RegistrationID: max length is 10, not null, not contains special characters
  - FullName: max length is 50

- Gender: famale or male, radio button is recommended
- Email: max length is 30, contain only one "@" character, not contain other special characters
- Phone: max length is 15, contain numeric characters only (0-9)
- Number of pet: must be > 0

- **Function 1: Create new registration - 50 LOC**
  - The user presses the Add new button to clear the information at detailed part, and system prepares for new data entry.
  - The user inputs new registration information. Then, user clicks the Save button.
  - The program checks the validity of data
    - If data is not valid then display an error message
    - Otherwise system will insert new registration into the FILE.
  - The registration table must be refreshed after new data has been successfully inserted.

- **Function 2: Show all registrations – 100 LOC**
  - The screen is divided into 2 parts: main information and detailed information.
    - Main part: this part lists all available registrations with their information (RegistrationID, Full name, age, gender, phone, address)
    - Detailed part:
      - When user clicks a row on the table or the Search button is clicked, system calls findByRegistrationID method (50 LOC), if the application finds a match RegistrationID.
      - The details of the respective registration are displayed following information as RegistrationID (disable), full name, age, email, phone, address, number of pet and symptoms.
      - Four buttons Search, Add new, Save and Delete button are put in detailed part. (see following sample picture)



- **Function 3: Update registration – 50 LOC**
  - The user clicks on the registration that she wants to modify on the registration table.
  - The details of the respective registration are displayed.
  - The user changes the information of the registration (not allow modify the RegistrationID). Then, the user clicks the Save button.
  - The program checks the validity of data
    - If data is not valid then display an error message
    - Otherwise system will update registration information.
  - The registration table must be refreshed after data has been successfully updated.
- **Function 4: Remove registration– 50 LOC**
  - The user clicks on the registration that she wants to delete on the registration table.

    Then users click the Delete button.
  - The program must display a message to confirm the deletion.
    - If the user confirms, system will delete the selected registration.
  - The registration table must be refreshed after data has been successfully deleted.
- **Function 5: Search by registration name – 50 LOC**
  - The user enters a string name that she wants to find on the registration table. Then, user clicks the Search by name button.
  - The registration table will display all data row that

- Their registration name contains the string name was entered above
- Otherwise, system will display a message to notify that can not found any the registration name.

- The registration table must be refreshed after data has been successfully find.

- **Function 6: Sort by registration name – 50 LOC**
  - The registration table displays the data in unsorted status. The user clicks on a combo box (sort by name) that she wants to sort the data on the registration table.
  - When the user choices the ascending option on the combo box, the system calls sortAscendingByRegistrationName method.
    - The data will be sorted ascending and displayed on the registration table.
  - When the user choices the descending option on the combo box, the system calls sortDescendingByRegistrationName method.
    - The data will be sorted descending and displayed on the registration table.
  - The registration table must be refreshed after data has been successfully sort.

❖ **The above specifications are only basic information; you must perform a requirements analysis step and build the application according to real requirements.**

❖ **The lecturer will explain the requirement only once on the first slot of the assignment.**