LAB221Assignment

Type:	**Long Assignment**
Code:	**J2.L.P0001**
LOC:	**400**
Slot(s):	**N/A**

Title

Patient management

Background

N/A

Program Specifications

In this assignment, you are required to build patient management application, in the form of a desktop application. The program has basic functions: add, edit, delete patient information. You are required to use the basic components to design interfaces, use the tabbed pane to organize patient management on the one screen. Array List is used to store information.

Program organization must clearly separate functions according to MVC model.

Patient information is shown following table.

Field Name	Type
patientID	String
patientName	String
birthday	Date
gender	Boolean
occupation	String
workplace	String
phone	String
address	String
healthInsuranceNumber	String
symptom	String

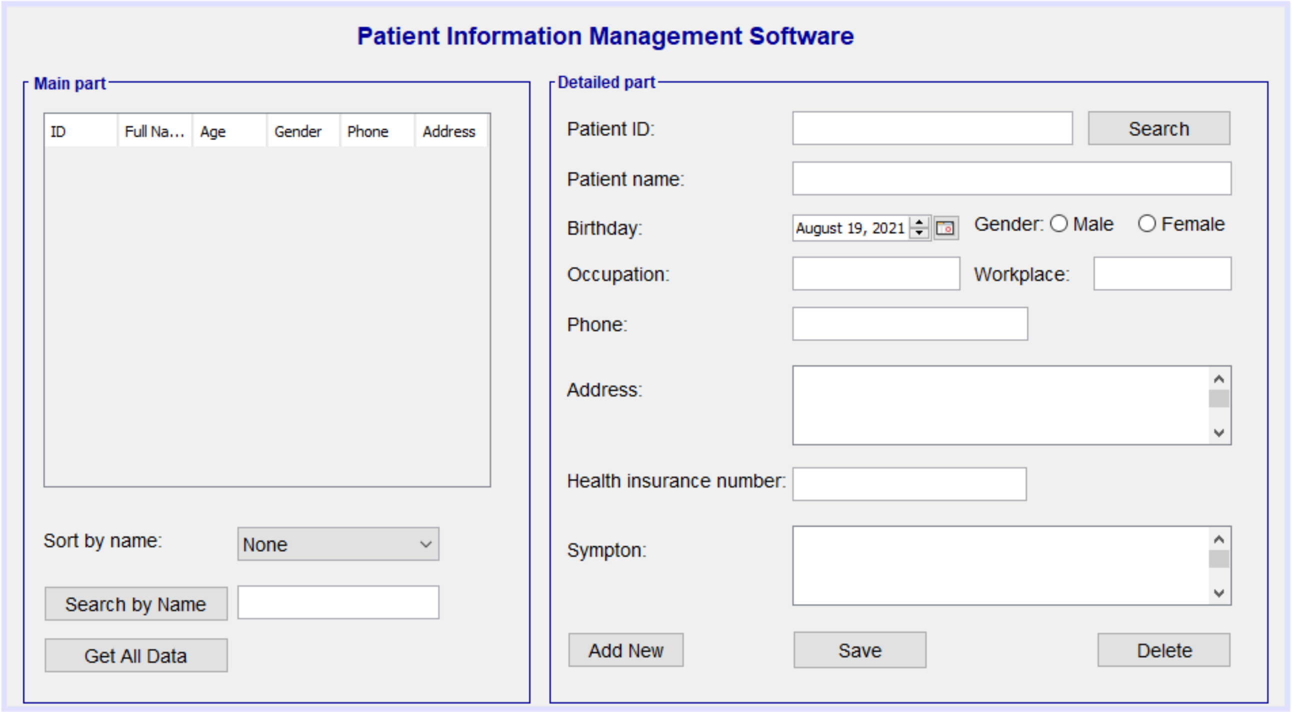
Features:

This system contains the following functions:

- **Verify constrain of following data type fields – 50 LOC**
  - Patient ID: max length is 10, not contains special characters (PA001)
  - Patient name: max length is 50
  - Birthday: real date
  - Gender: male or female
  - Phone: real phone number, max length is 15, contain numeric characters only (0-9).
  - HealthInsuranceNumber: health insurance card number.
- **Function 1: Create patient – 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 patient 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 patient into the Array List.
  - The patient table must be refreshed after new data has been successfully inserted.

- **Function 2: Show all patients – 100 LOC**
  - The screen is divided into 2 parts: main information and detailed information.
  - Main part: this part lists all available patients with their information (Patient ID, Patient name, birthday, gender, phone, address, occupation and workplace) in the system. (50 LOC)
  - Detailed part: when you click a row on the table or the Find by ID button is clicked, system calls findByPatientID method (50 LOC), if the application finds a match PatientID. The details of the respective patient are displayed following information as PatientID (disable), patient name, birthday, gender, phone, address, occupation and workplace.
  - Four buttons Find by ID, Add new, Save and Remove button are put in detailed part.



Sort by name: None

Birthday: August 19, 2021

- **Function 3: Update patient – 50 LOC**
  - The user clicks on the patient that she wants to modify on the patient table.
  - The details of the respective patient are displayed.
  - The user changes the information of the patient (not allow modify the Patient ID). 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 patient information.
  - The patient table must be refreshed after data has been successfully updated.

- **Function 4: Remove patient – 50 LOC**

- The user clicks on the patient that she wants to delete on the patient table.

  Then users click the Remove button.

- The program must display a message to confirm the deletion. If the user confirms, system will delete the selected patient.

- The patient table must be refreshed after data has been successfully deleted.

- **Function 5: Search by patient name – 50 LOC**
  - The user enters a string name that she wants to find on the patient table.

    Then user clicks the Search by name button.

  - The patient table will display all data row that their patient name contains the string name was entered above. Otherwise, system will display a message to notify that can not found any the patient name.

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

- **Function 6: Sort by patient name – 50 LOC**
  - The patient 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 patient table.

  - When the user choices the ascending option on the combo box, the system calls sortAscendingByPatient Name method. The data will be sorted ascending and displayed on the patient table.

  - When the user choices the descending option on the combo box, the system calls sortDescendingByPatient Name method. The data will be sorted descending and displayed on the patient table.

  - The patient 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.