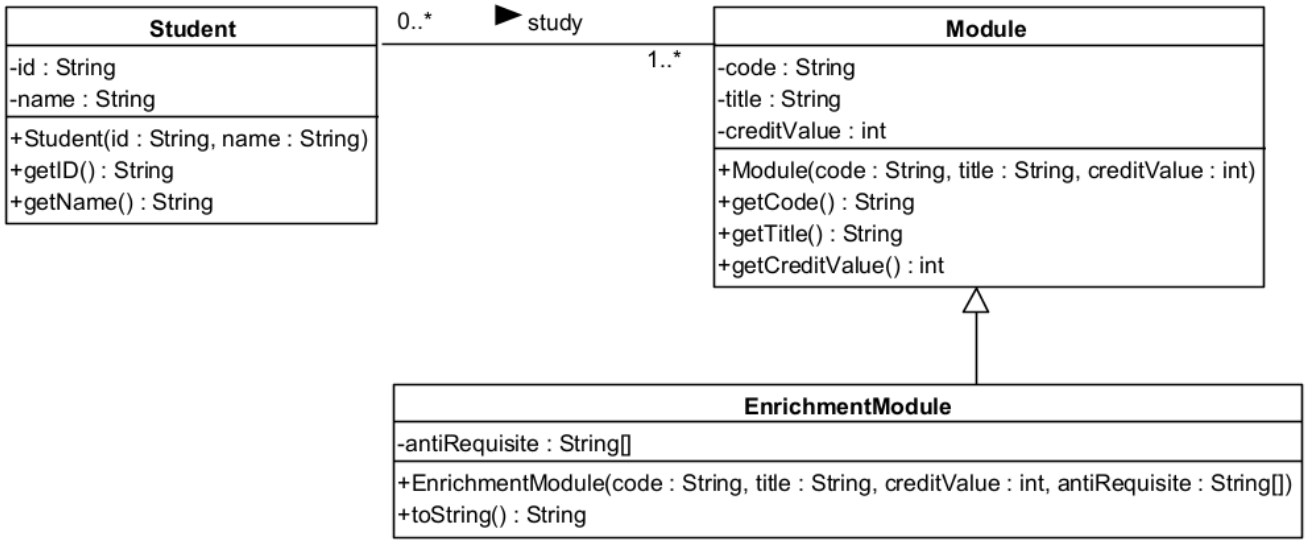
Section A (40 marks) This section contains THREE questions. Answer ALL questions.

A1. The robot company specializes in three types of robots: industrial robots, service robots, and medical robots, each with a distinct model number and a specific control unit. There are three types of control units. The industrial robot is equipped with a power control unit, while the service robot uses a cleaning control unit, and the medical robot operates with a telemedicine control unit. The power control unit offers various overload protection options, the cleaning control unit comes with multiple water tank capacities, and the telemedicine control unit has different levels of patient data encryption. Moreover, the industrial robot's operating arm has various sizes available while the service robot comes with a range of cleaning tools, and the medical robot features different patient monitoring sensors. Different versions of control units are available, and any programmer within the company can maintain them. If any mechanical problem arises, any engineer within the company can perform repairs on the robots.

Draw a class diagram to model the classes and the relationships between classes for the above problem statement. You are required to show the **name** and **multiplicity** of each association in your diagram. You only need to show the attributes in each class based on the given information. You should also structure the classes with inheritance if possible. [13 marks]

A2. Given the following class diagram and the test program **TestModule.java**:



QUESTION A2 CONTINUES ON THE NEXT PAGE

QUESTION A2 CONTINUES FROM THE PREVIOUS PAGE

```
import java.util.*;

public class Test {
  public static void main(String[] agrv) {
    Student[] students = new Student[2];
    Module[] modules = new Module[3];

    students[0] = new Student("220123456", "Lai Tuk Sui");
    students[1] = new Student("220534534", "Ng Tak Kei");
    modules[0] = new Module("ITP4909", "Object Oriented Technology", 13);
    modules[1] = new EnrichmentModule("ITE3905", "Artificial Intelligence
          Fundamentals", 6, new String[]{"ITP4211", "ITP4221", "ITP4886"});
    modules[2] = new EnrichmentModule("ITE3902", "Smartphone Apps
          Fundamentals", 6, new String[]{"ITP4203"});

    students[0].addModule(modules[1]);
    students[0].addModule(modules[2]);
    students[1].addModule(modules[0]);
    students[1].addModule(modules[1]);

    modules[0].addStudent(students[1]);
    modules[1].addStudent(students[0]);
    modules[1].addStudent(students[1]);
    modules[2].addStudent(students[0]);

    Enumeration moduleList = students[0].getModules();
    System.out.println("Enrichment modules studied by " +
          students[0].getName() + ": ");
    while (moduleList.hasMoreElements()) {
      Module module = (Module) moduleList.nextElement();
      if(module instanceof EnrichmentModule)
        System.out.println((EnrichmentModule) module);
    }
  }
}
```

Based on the given class diagram and the test program, implement the classes **Student**, **Module** and **EnrichmentModule** in Java. Your implementation should reflect the requirement in the class diagram and be able to run the **Test.java** program without error.

The following is a sample output when the test program is run.

```
Enrichment modules studied by Lai Tuk Sui:
ITE3905 Artificial Intelligence Fundamentals, credit value: 6, anti-requisite: ITP4211 ITP4221 ITP4886
ITE3902 Smartphone Apps Fundamentals, credit value: 6, anti-requisite: ITP4203
```

[15 marks]

A3. Consider the following procedure for an Internet user using the Internet jobs application system (JAS) to apply a job. It is assumed that the user has registered a user account of JAS.

The JAS displays a login screen and asks the user to login the JAS by entering user ID and password. After login, the JAS displays a textbox for entering keywords, a "SEARCH" button and a "QUIT" button. The user enters keywords. If there is/are matching job(s), the system displays a list of matching jobs. If there is no matching job, JAS displays "No matching job found" and the user can enter keywords to search again. At any time, the user can click the "QUIT" button to end the procedure.

The user selects a job in the list of matching jobs to apply for a job. The JAS displays the requirements of the job and ask for resume, the user clicks the "UPLOAD" button to upload his/her resume or click the "QUIT" button to end the procedure. If the user clicks the "UPLOAD" button, the JAS will ask the user to complete an online job test. The user takes the test. If the user was scored less than the required scores, the procedure is ended. If the user's score was equal to or more than the required scores, the JAS displays an interview date and time, and ask the user to accept the interview. If the user clicks the "ACCEPT" button, the JAS will display a confirmation message for the interview and the procedure is ended. If the user clicks the "REJECT" button, the procedure is ended.

(a) Draw an activity diagram to model the procedure for a user to apply a job in the JAS. You should include the initial node, final node, the name of each action, decision node (if any) in your activity diagram.

[6 marks]

(b) Draw a state machine diagram to model the procedure for a user to apply for a job in the JAS. You should include the initial state, final state, the name of each state, the transition, guard condition (if any), and effects in your state machine diagram.

[6 marks]

******* END OF SECTION A *******

Section B (60 marks)
This section contains THREE questions.

Answer ALL questions.

The problem statement for the questions in Section B.

A healthcare company is developing an online vaccination booking system (VBS). The company has several vaccination centres. A vaccination centre provides one or more vaccines. A clinic nurse works for one vaccination centre and is responsible for helping the patient to get a vaccination. A clinic nurse has a name and an employee ID. There are two kinds of vaccination centre: clinic and medical centre. For a medical centre, vaccination service is only available in certain day(s) of a week. For a clinic, vaccination service is available only in a certain time period of a day.

A public user can register a patient account using VBS. To register an account, the public user enters his/her mobile phone number, HKID, name, date of birth and password. VBS sends a verification code to his/her mobile phone through an external SMS gateway. The public user enters the verification code and VBS sends a confirmation SMS to the public user.

A public user or a patient (registered user) can browse vaccines using VBS. The user clicks "Browse Vaccines" button. The system displays a list of vaccines with names of vaccines and names of manufacturers. The user selects a vaccine in the list. The system displays a list of vaccination centres with addresses and opening hours.

A patient can reserve for taking a vaccine in a vaccination centre using VBS. The patient logs in his/her patient account by entering his/her phone number and password. The system displays the "Browse Vaccines" button. The patient browses vaccines using VBS as described in the previous paragraph. The patient selects a vaccination centre. The system displays available dates. The patient selects a date. The system displays available times. The patient selects a time. The system displays the booking details and asks the patient to confirm. The patient clicks the "Confirm" button. The system displays the booking record with a booking reference number.

A clinic nurse can check a booking using VBS. The clinic nurse enters the patient phone number. The system displays the bookings of the patient.

A clinic nurse can create a vaccination record for a patient. The clinic nurse enters the patient phone number, the date and the name of the vaccine used. The system creates the vaccination record of the patient.

Answer questions B1 to B3 on the next page based on the given problem statement.

B1. Draw a use case diagram for VBS. Show all use cases, actors, communication links between actors and use cases, and <<include>> / <<extend>> relationships between use cases.

[20 marks]

B2. **(a)** Perform a textual analysis on the problem statement to identify candidate classes of VBS. List the candidate classes with the corresponding reasons for the choices of the candidate classes. [8 marks]

(b) Draw an initial class diagram to show the classes of VBS. Show the relationships among classes in your answer. Give appropriate names to associations. You are NOT required to show multiplicities on associations, attributes/methods of classes, or any association class.

[8 marks]

B3. **(a)** Write the flow of events for the scenario in which a patient successfully reserves "seasonal influenza vaccine" at 16:30 on 1 June 2023 in Tsing Yi Clinic with the following information:

- phone number: 77778888
- password: 1234

[10 marks]

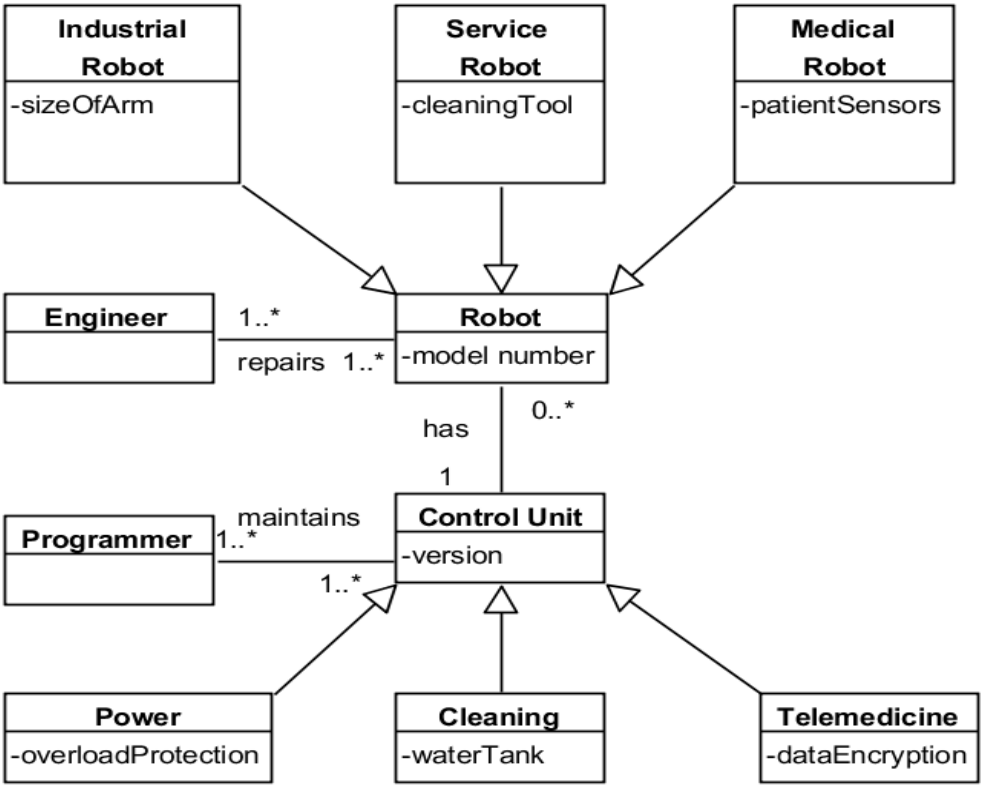
(b) Draw a system level sequence diagram to illustrate the scenario described in B3 (a).

[14 marks]

******* END OF SECTION B *******

******* END OF PAPER *******

Suggested Solution

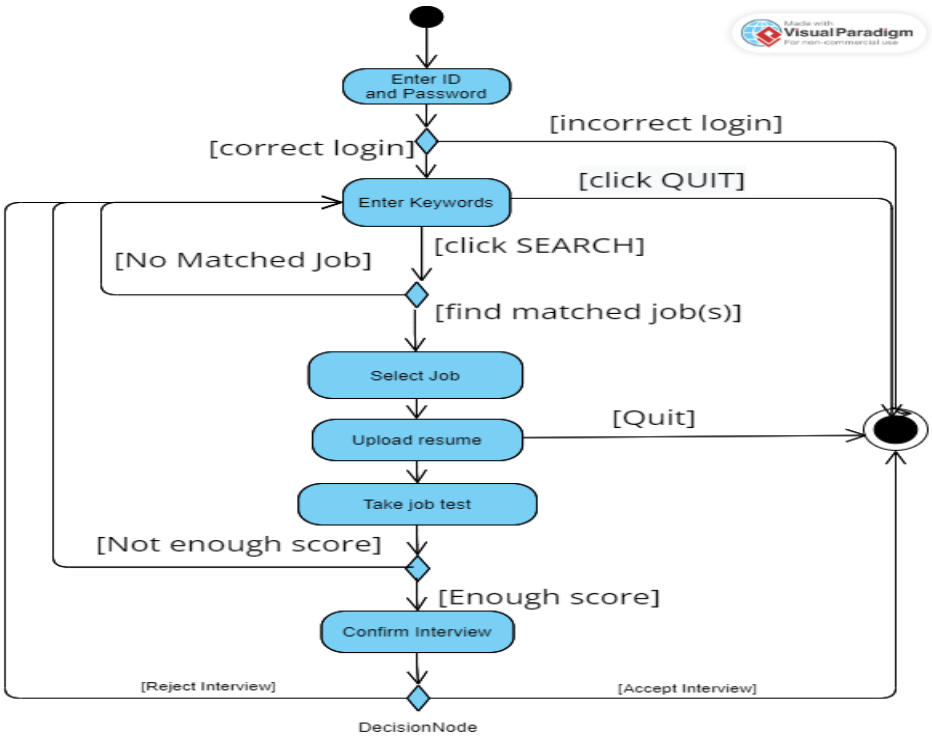
Qn.	Solution
A1	

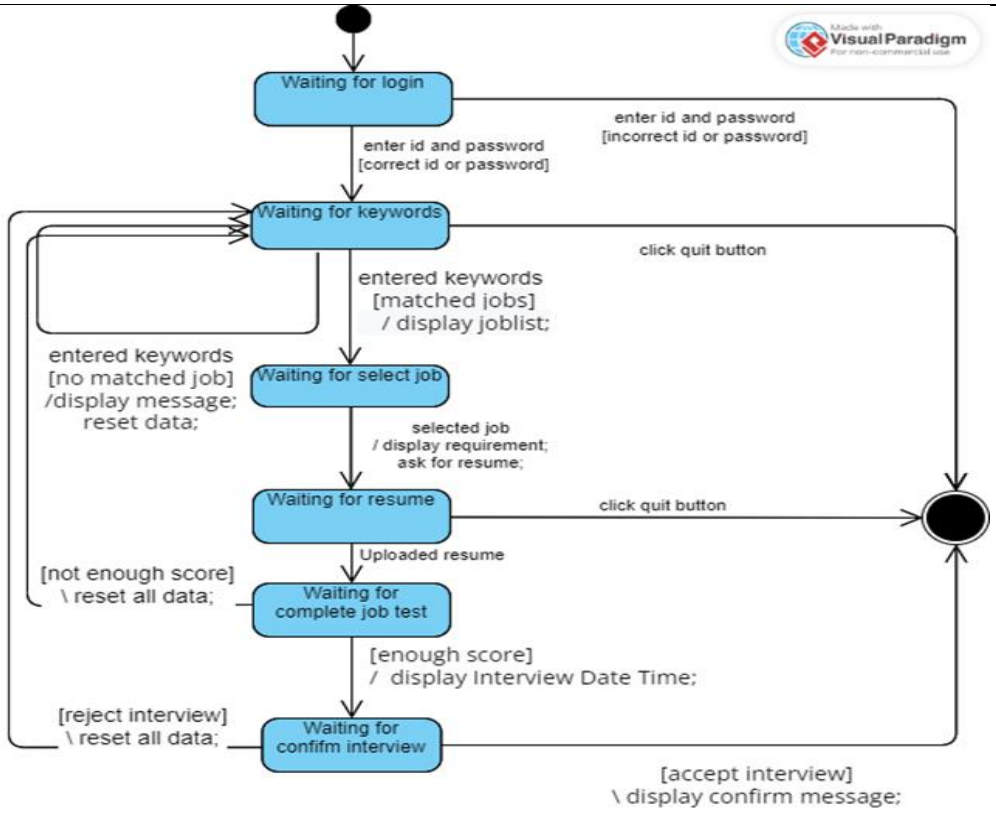
Qn.	Solution

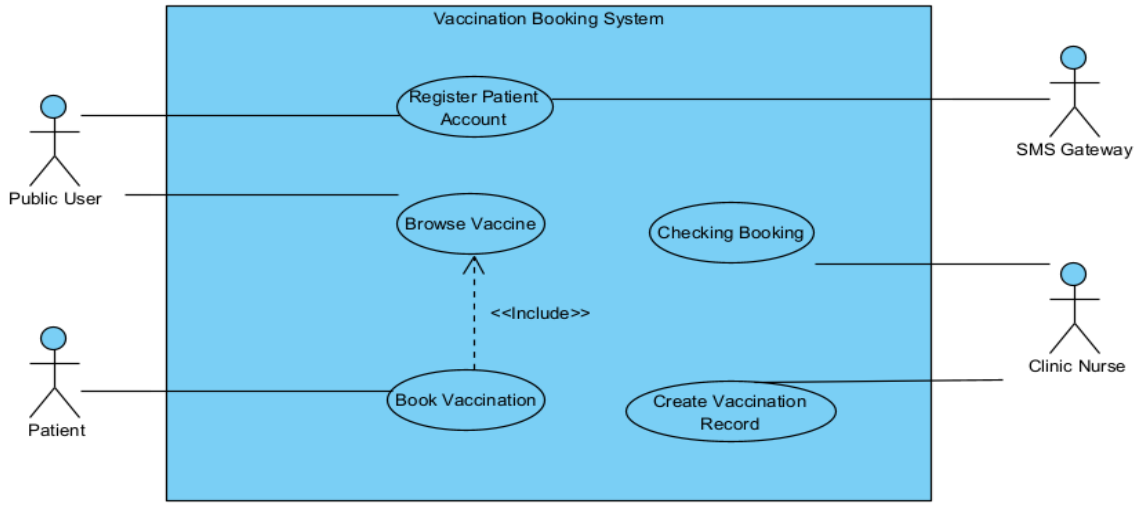
A2	import java.util.*;
	public class Student {
	private String id, name; private Vector _modules;
	
	public Student(String id, String name) {
	this.id = id; this.name = name; _modules = new Vector();
	}
	
	public String getID() { return id; }
	public String getName() { return name; }
	public void addModule(Module module) { _modules.add(module); }
	public Enumeration getModules() { return _modules.elements(); }
	}

	import java.util.*;
	public class Module {
	private String code, title;
	private int creditValue;
	private Vector _students;
	
	public Module(String code, String title, int creditValue) {
	this.code = code;
	this.title = title;
	this.creditValue = creditValue;
	_students = new Vector();
	}
	public String getCode() { return code; }
	public String getTitle() { return title; }
	public int getCreditValue() { return creditValue; }
	public void addStudent(Student student) { _students.add(student); }
	}

	public class EnrichmentModule extends Module{
	private String[] antiRequisite;
	
	public EnrichmentModule
	(String code, String title, int creditValue, String[] antiRequisite) {
	super(code, title, creditValue); this.antiRequisite = antiRequisite;
	}
	
	public String toString() {
	String arModules = "";
	for(String ar: antiRequisite)
	arModules += ar + " ";
	return super.getCode() + " " + super.getTitle() + ", credit value: " +
	super.getCreditValue() + ", anti-requisite: " + arModules;
	}
	}

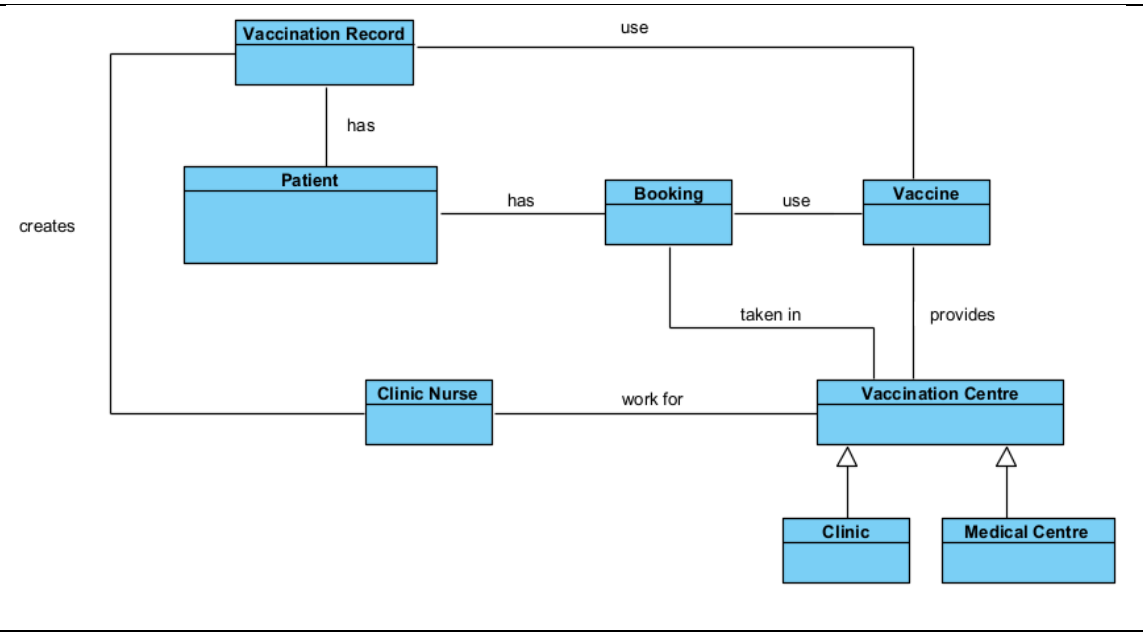
Qn.	Solution
A3(a)	

A3(b)	

Qn.	Solution
B1	

B2 (a)	Patient (role played) Booking (event) Vaccine (tangible thing) Clinic Nurse (role played) Vaccination Centre (tangible thing) Clinic (tangible thing) Medical Centre (tangible thing) Vaccination Record (event)

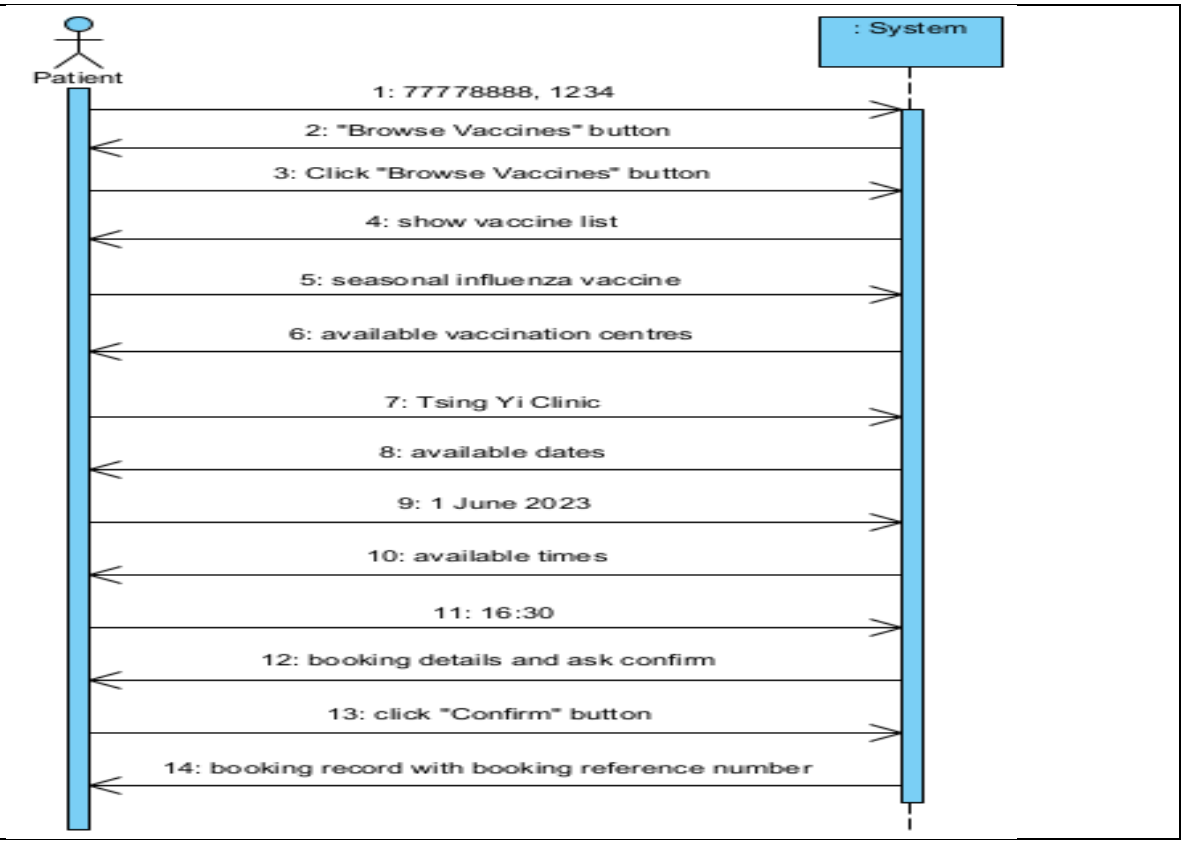
B2 (b)



B3 (a)

Patient	Vaccination Booking System
Enter “77778888”, “1234”	
	Display “Browse Vaccines” button
Click “Browse Vaccines” button	
	Display vaccine list
Select “seasonal influenza vaccine”	
	Display available vaccination centres
Select “Tsing Yi Clinic”	
	Show available dates
Select “1 June 2023”	
	Display available times
Select “16:30”	
	Display booking details and ask confirm
Click "Confirm" button	
	Display booking record with booking reference number

B3 (b)



*****END OF SUGGESTED SOLUTION*****