National University of Singapore

School of Computing

IS2103 – Enterprise Systems Server-side Design and Development

Assignment 2: Group Project

Total Marks: 35 marks

BACKGROUND

NUS has allocated a greater amount of money for the development of the NUS Health center (NUSHC). Their main objective is to introduce an enterprise solution for enabling students and staff to make appointments. In their future services, they are planning to use an innovative appointment booking mechanism to shorten students' and staffs' waiting time. They are also expected to optimize administrative and clinical works to improve productivity and maximize the bottom line of clinic financial reports.

In addition, NUS is partnering with Automated Machine Services Pte Ltd (AMS) to leverage a large network of AMS machines to provide better reach to their students and staff. AMS has over 400 automated machines islandwide for its customers to perform transactions such as bill payment, financial services, and now clinic services.

Your team has recently been recruited by NUS to design and develop an enterprise-scale Clinical Appointment Registration System (CARS) for NUSHC's future clinic services. It should provide a comprehensive suite of applications and web services for the clinic to operate with minimal maintenance efforts.

The appointment and registration workflows start with students and staff (hereafter we address them as patients) at the center of NUSHC clinic services. Patients must register for an account with their identity number (e.g., NRIC or Passport Number) and a password. They can make an appointment with a clinic doctor via different means such as telephone, walk-in, self-service kiosk, or AMS machine. The appointment will be confirmed in a real-time mechanism, which utilizes a smart algorithm for appointment scheduling at the clinic. Furthermore, patients, with or without an appointment, can visit the clinic and get into a consultation with a simplified registration workflow. A queue number will be provided for seeing a doctor during their visit.

The clinic staff is responsible for counter operations, appointment operations, and administration operations. They can use the admin terminal located at the clinic counter to perform their daily operations.

HIGH-LEVEL SYSTEM ARCHITECTURE

All software elements constituting the CARS are to be developed in Java using the Java Platform, Enterprise Edition (Java EE). In particular, Enterprise JavaBeans (EJB) and Java Persistence API (JPA) technologies are to be used in conjunction with a suitable Relational Database Management System (RDBMS) such as MySQL. Only the Command-line Interface (CLI) client applications are required. The CARS will consist of:

i) A core backend to be developed with a component-based architecture
ii) A Clinic Admin Terminal to support the clinic appointment and registration workflow
iii) A Self-Service Kiosk Terminal for patients to do self-registration and appointment booking
iv) Web services for interfacing with external vendors
v) An AMS Client for testing purposes of the web services

The high-level architecture of the CARS is depicted in a block diagram as shown in Figure 1 below. Clinic Admin Terminal is a client application used by clinic staff and the Self-Service Kiosk client is used by patients.

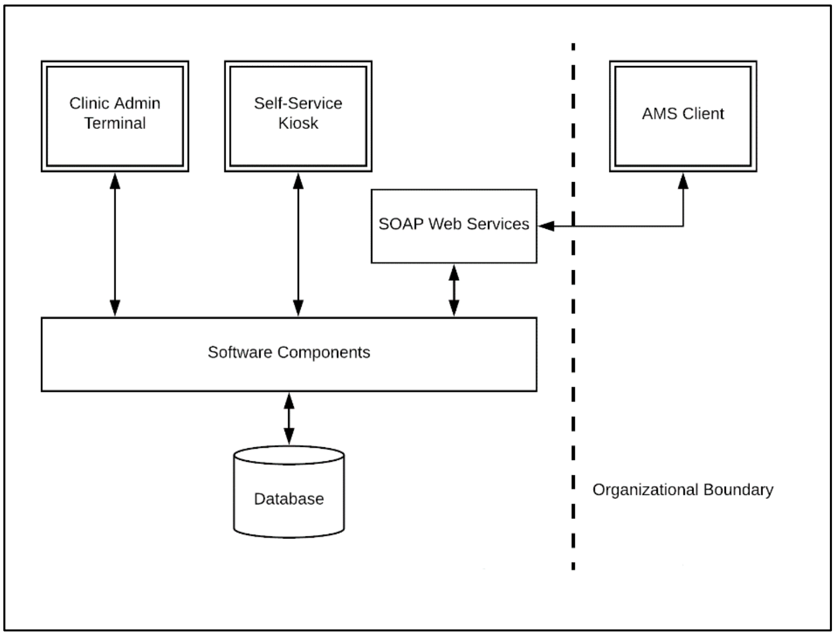

Figure 1 – High-level architecture of the Clinic Appointment Registration System (CARS)

PROJECT TASKS

Your team has been tasked to implement a pilot version of the **Clinic Appointment Registration System (“CARS”)**.

There are several points to note for your development:

1. You are **NOT** required to draw a UML class diagram.
2. You are **NOT** required to draw any formal software engineering diagram.
3. The date format is yyyy-MM-dd (i.e. 2019-03-01) and the time format is HH:mm (i.e. 14:30).
4. The clinic is open from Monday to Wednesday, from 08:30 to 18:00, Thursday from 08:30 to 17:00 and Friday 08:30 to 17:30. Also, the clinic is closed for lunch from 12:30 -13:30. All doctors are scheduled to work during the operating hours, in different consultation rooms.
5. A consultation has a fixed length of 30 minutes. So, the last consultation is 30 minutes before closing.
6. Appointments must be booked in advance by at least two days. There is no modification of appointment once it is confirmed, and it is suggested for patients/staff to cancel the previous appointment and make a new appointment if needed.
7. Walk-in consultation can only be registered for the next three hours. For example, if a patient walks into the clinic at 13:35, there will only be 6 available slots for a consultation: 14:00, 14:30, 15:00, 15:30, 16:00, and 16:30.
8. A queue number will be given to the patient for his/her consultation. It is an incremental number; which will be reset on a daily basis. For example, if patient A registers a consultation with doctor X at the beginning of the day, the queue number 1 will be given to patient A. If patient B registers a consultation with Doctor Y after patient A, the queue number 2 will be given to patient B. The queue numbers will be displayed on top of different rooms for consultations.
9. The doctors may apply leaves for certain dates and admin staff must enter their leave dates in the application. However, doctors must apply for a leave a one-week advance and can be applied only 1 day per week. In case a doctor is allocated for an appointment, he or she may not be offered the leave.
10. The patient password must be a six-digit number. Also, it is recommended to encrypt the passwords in the system.

There are several employees at the clinic as shown in Tables 1 & 2.

staffId	firstName	lastName	userName	password
1	Eric	Some	manager	password
2	Victoria	Newton	nurse	password

Table 1 – Staff Entities

doctorId	firstName	LastName	registration	qualifications
1	Tan	Ming	S10011	BMBS
2	Clair	Hahn	S41221	MBBCh
3	Robert	Blake	S58201	MBBS

Table 2 – Doctor Entities

In addition, the clinic has also provided you with sample data of their patient database as shown in Table 3.

patientId	identityNumber	firstName	lastName	gender	age	phone	address	password
1	S9867027A	Sarah	Yi	F	22	93718799	13, Clementi Road	DY3ihrBrktQyJIz6uMDsqA
2	G1314207T	Rajesh	Singh	M	36	93506839	15, Mountbatten Road	Qa0Xm0UFdx3HZ6Xs7tyKKQ

Table 3 – Patient Entities

ARCHITECTURE AND DESIGN

Design a suitable **logical data model** consisting of a set of entity classes, their attributes, and the relationships among the entity classes to support the entire CARS software system. You are **NOT** required to draw a UML class diagram. You are only required to create the entity classes using JPA and then apply the **forward engineering** technique to generate the physical data model, i.e., the underlying relational database tables.

Java annotations from JPA must be used appropriately to decorate the entity classes in order to enforce the strictest possible integrity constraints on the logical data model. More explicitly, you are to ensure the correctness and integrity of the data that are eventually stored in the underlying relational database according to some well-defined business rules or assumptions.

For examples:

```
@Column(length = 32, nullable = false)
private String firstName;
@Column(length = 9, nullable = false, unique = true)
private String identificationNumber;
```

are preferred over:

```
private String firstName;
private String identificationNumber;
```

And:

```
@ManyToOne(optional = false)
@JoinColumn(nullable = false)
private Patient patient;
```

are preferred over:

```
@ManyToOne
private Patient patient;
```

Using Java EE, design a suitable **physical architecture** for the CARS software system taking reference from the high-level architecture shown in Figure 1. Your architecture should state clearly the various Java EE software elements that you are developing and their accompanying rationales.

Document the business rules, rationales and assumptions for your logical data model as well as a brief description of the physical architecture in a Microsoft Word document. You are **NOT** required to draw any formal software engineering diagram. If you wish to provide any illustrations, you may use any information block diagrams.

You are required to provide instructions for clinics to deploy and test your system with sample data.

CLINIC ADMIN TERMINAL

The UML use case diagram for the Clinic Admin Terminal is shown in Figure 2 below. A brief description of selected business use cases together with the associated business rules are given in Table 4.

If use case A includes use case B, it means that when performing the basic course of actions for use case A, the behaviour of use case B will be invoked at a point specified by the base use case, i.e., use case A. In Figure 3, “Register Consultation By Appointment” and “Cancel Appointment” both include “View Appointments”.

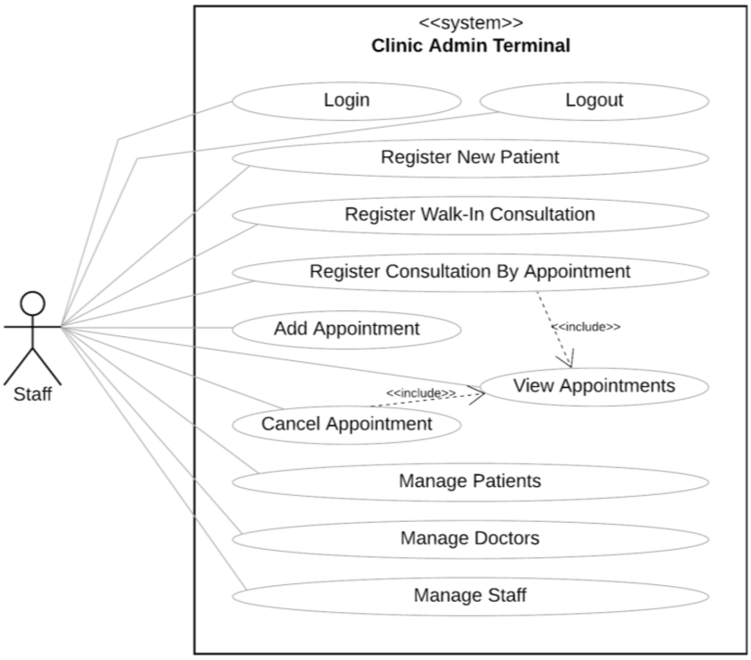

Figure 2 – UML use case diagram for Clinic Admin Terminal

S/N	Use Case	Description/Business Rules
1	Login	• May only be performed if the staff is not currently logged into the system. • Staff must be currently logged into the system to perform all other use cases.
2	Register New Patient	• If a patient is not registered, the admin staff registers a new patient record in the system. Admin staff needs to ask and enter patient information: o Identity Number (NRIC or Passport) o Password o First Name o Last Name o Gender o Age o Phone o Address
3	Register Walk-in Consultation	• Admin staff registers a walk-in consultation for a registered patient. • The system provides admin staff with 3-hour availability of all doctors for registration. • A queue number will be given to the patient once the consultation registration is done.
4	Register Consultation by Appointment	• Admin staff view appointments with the patient identity number • The system allows the admin staff to select which appointment to register. • A queue number will be given to the patient once the consultation registration is done.
5	View Appointments	• The system list down all appointments for the patient using his/her identity number.
6	Add Appointment	• The system shows a list of the doctors and allows the admin staff to select a doctor, and select the date for adding an appointment. • The system provides the availability of the doctors for the date.

		• The admin staff enters the time and patient identity number for finalizing the appointment.
7	Cancel Appointment	• The system shows the list of patient appointments, given his/her patient identity number. • Admin staff enters the appointment id to cancel the appointment.
8	Manage Patients	• The system provides functionalities for admin staff to perform: o Add Patient o View Patient Details o Update Patient o Delete Patient o View All Patients
9	Manage Doctors	• The system provides functionalities for admin staff to perform: o Add Doctor o View Doctor Details o Update Doctor o Delete Doctor o View All Doctors o Enter Doctor's leave requests
10	Manage Staff	• The system provides functionalities for admin staff to perform: o Add Staff o View Staff Details o Update Staff o Delete Staff o View All Staff
11	Logout	• Logout if the staff is currently logged in the system.

Table 4 – Use case descriptions and business rules of Clinic Admin Terminal

You are expected to create a simple console interface for the clinic staff to perform the necessary business use cases. At the bare minimum, the system shall produce the following outputs of key business use cases, but not limited to:

Login

```
*** Welcome to Clinic Appointment Registration System (CARS) ***

1: Login
2: Exit

>
```

```
*** CARS :: Login ***

Enter username> manager
Enter password> password
Login successful!
```

Main Menu

```
*** CARS :: Main ***

You are login as Linda Chua

1: Registration Operation
2: Appointment Operation
3: Administration Operation
4: Logout

>
```

Registration Operations

```
*** CARS :: Registration Operation ***
```

```
1: Register New Patient
2: Register Walk-In Consultation
3: Register Consultation By Appointment
4: Back

>
```

```
*** CARS :: Registration Operation :: Register New Patient ***

Enter Identity Number> S9867027A
Enter Password> 001001
Enter First Name> Sarah
Enter Last Name> Yi
Enter Gender> F
Enter Age> 22
Enter Phone> 93718799
Enter Address> 13, Clementi Road
Patient has been registered successfully!
```

```
*** CARS :: Registration Operation :: Register Walk-In Consultation ***

Doctor:
Id |Name
1  |Tan Ming
3  |Robert Blake
4  |Tan Min

Availability:
Time  |1 |3 |4 |
13:00 |O |X |X |
13:30 |O |O |X |
14:00 |O |O |O |
14:30 |O |X |O |
15:00 |O |X |O |
15:30 |X |O |O |

Enter Doctor Id> 4
Enter Patient Identity Number> S9867027A
Sarah Yi appointment with Dr. Tan Min has been booked at 14:00.
Queue Number is: 35.
```

```
*** CARS :: Registration Operation :: Register Consultation By Appointment ***

Enter Patient Identity Number> S9867027A

Appointments:
Id |Date       |Time  |Doctor
03 |2019-03-01 |09:30 |Robert Blake

Enter Appointment Id> 03
Sarah Yi appointment is confirmed with Dr. Robert Blake at 09:30.
Queue Number is: 10.
```

Appointment Operations

```
*** CARS :: Appointment Operation ***

1: View Patient Appointments
2: Add Appointment
3: Cancel Appointment
4: Back

>
```

```
*** CARS :: Appointment Operation :: View Patient Appointments ***
```

```
Enter Patient Identity Number> S9867027A

Appointments:
Id |Date       |Time  |Doctor
37 |2018-12-26 |15:30 |Robert Blake
63 |2019-03-01 |09:30 |Robert Blake
```

```
*** CARS :: Appointment Operation :: Add Appointment ***

Doctor:
Id |Name
1  |Tan Ming
2  |Clair Hahn
3  |Robert Blake
4  |Tan Min

Enter Doctor Id> 1
Enter Date> 2019-03-10

Availability for Tan Ming on 2019-03-10:
9:00 09:30 10:00 10:30 14:00 14:30 15:00 15:30

Enter Time> 10:30
Enter Patient Identity Number> S9867027A
Sarah Yi appointment with Tan Ming at 10:30 on 2019-03-10 has been added.
```

```
*** CARS :: Appointment Operation :: Cancel Appointment ***

Enter Patient Identity Number> S9867027A

Appointments:
Id |Date       |Time  |Doctor
37 |2018-12-26 |15:30 |Robert Blake
63 |2019-03-01 |09:30 |Robert Blake
105|2019-03-10 |10:30 |Tan Ming

Enter Appointment Id> 105
Sarah Yi appointment with Tan Ming at 10:30 on 2019-03-10 has been cancelled.
```

Administration Operations

```
*** CARS :: Administration Operation ***

1: Patient Management
2: Doctor Management
3: Staff Management
4: Back

>
```

```
*** CARS :: Administration Operation :: Patient Management ***

1: Add Patient
2: View Patient Details
3: Update Patient
4: Delete Patient
5: View All Patients
6: Back
```

```
*** CARS :: Administration Operation :: Doctor Management ***

1: Add Doctor
2: View Doctor Details
3: Update Doctor
4: Delete Doctor
5: View All Doctors
6: Leave Management
7: Back
```

```
*** CARS :: Administration Operation :: Staff Management ***

1: Add Staff
2: View Staff Details
3: Update Staff
4: Delete Staff
5: View All Staff
6: Back
```

The outputs for administration operations are omitted for brevity.

SELF-SERVICE KIOSK

The UML use case diagram for the CARS Self-Service Kiosk is shown in Figure 4 below. A brief description of selected business use cases together with the associated business rules are given in Table 5.

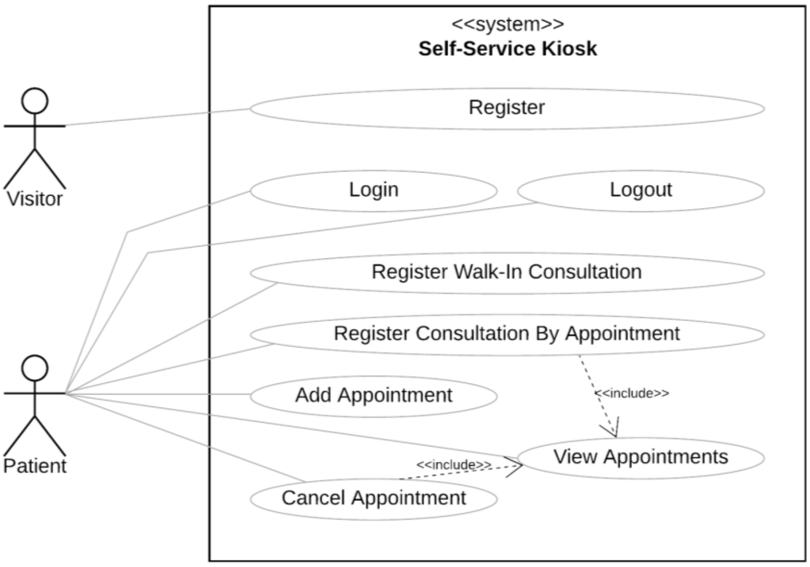

Figure 3 – UML use case diagram for Self-Service Kiosk

S/N	Use Case	Description/Business Rules
1	Register	• If a patient is not registered, the system will prompt for the following information: ○ Identity Number (NRIC or Passport) ○ Password ○ First Name ○ Last Name ○ Gender ○ Age ○ Phone

		○ Address
2	Login	• If a patient is registered, the patient will login into the system using: ○ Identity Number ○ Password
3	Register Walk-in Consultation	• The system provides the patient with 3-hour availability of all doctors for registration. • The patient selects a doctor for consultation. • A queue number will be given to the patient once the consultation registration is done.
4	Register Consultation by Appointment	• The system shows the list of patient appointments. • The system allows the patient to select which appointment to register. • A queue number will be given to the patient once the consultation registration is done.
5	View Appointments	• The system list down all appointments for the patient.
6	Add Appointment	• The system shows a list of the doctors and allows the patient to select a doctor, and select the date for adding an appointment. • The system provides the availability of the doctors for the date. • The patient enters the time for finalizing the appointment.
7	Cancel Appointment	• The system shows the list of patient appointments. • The patient enters the appointment id to cancel the appointment.
8	Logout	• Logout if the patient is currently logged in the system.

Table 5 – Use case descriptions and business rules of Self-Service Kiosk

At the bare minimum, the system shall produce the following outputs of key business use cases, but not limited to:

Main Menu

```
*** Welcome to Self-Service Kiosk ***

1: Register
2: Login
3: Exit

>
```

```
*** Self-Service Kiosk :: Register ***

Enter Identity Number> S9867027A
Enter Password> 001001
Enter First Name> Sarah
Enter Last Name> Yi
Enter Gender> F
Enter Age> 22
Enter Phone> 93718799
Enter Address> 13, Clementi Road
Registration is successful!
```

```
*** Self-Service Kiosk :: Login ***

Enter Identity Number> S9867027A
Enter Password> 001001
Login successful!
```

Patient Menu

```
*** Self-Service Kiosk :: Main ***

You are login as Sarah Yi

1: Register Walk-In Consultation
2: Register Consultation By Appointment
3: View Appointments
4: Add Appointment
5: Cancel Appointment
6: Logout

>
```

```
*** Self-Service Kiosk :: Register Walk-In Consultation ***

Doctor:
Id |Name
1  |Tan Ming
3  |Robert Blake
4  |Tan Min

Availability:
Time  |1 |3 |4 |
13:00 |O |X |X |
13:30 |O |O |X |
14:00 |O |O |O |
14:30 |O |X |O |
15:00 |O |X |O |
15:30 |X |O |O |

Enter Doctor Id> 3
Sarah Yi appointment with Dr. Tan Min has been booked at 14:00.
Queue Number is: 25.
```

```
*** Self-Service Kiosk :: Register Consultation By Appointment ***

Appointments:
Id |Date       |Time  |Doctor
03 |2019-03-01 |09:30 |Robert Blake

Enter Appointment Id> 03
Sarah Yi appointment is confirmed with Dr. Robert Blake at 09:30.
Queue Number is: 10.
```

```
*** Self-Service Kiosk :: View Appointments ***

Appointments:
Id |Date       |Time  |Doctor
37 |2018-12-26 |15:30 |Robert Blake
63 |2019-03-01 |09:30 |Robert Blake
```

```
*** Self-Service Kiosk :: Add Appointment ***

Doctor:
Id |Name
1  |Tan Ming
2  |Clair Hahn
3  |Robert Blake
4  |Tan Min

Enter Doctor Id> 1
Enter Date> 2019-03-10

Availability for Tan Ming on 2019-03-10:
9:00 09:30 10:00 10:30 14:00 14:30 15:00 15:30
```

```
Enter Time> 10:30
Sarah Yi appointment with Tan Ming at 10:30 on 2019-03-10 has been added.
```

```
*** Self-Service Kiosk :: Cancel Appointment ***

Appointments:
Id |Date       |Time  |Doctor
37 |2018-12-26 |15:30 |Robert Blake
63 |2019-03-01 |09:30 |Robert Blake
105|2019-03-10 |10:30 |Tan Ming

Enter Appointment Id> 105
Sarah Yi appointment with Tan Ming at 10:30 on 2019-03-10 has been cancelled.
```

AUTOMATED MACHINE SERVICE CLIENT

The UML use case diagram for the CARS Automated Machine Service Client is shown in Figure 5 below. A brief description of selected business use cases together with the associated business rules are given in Table 6.

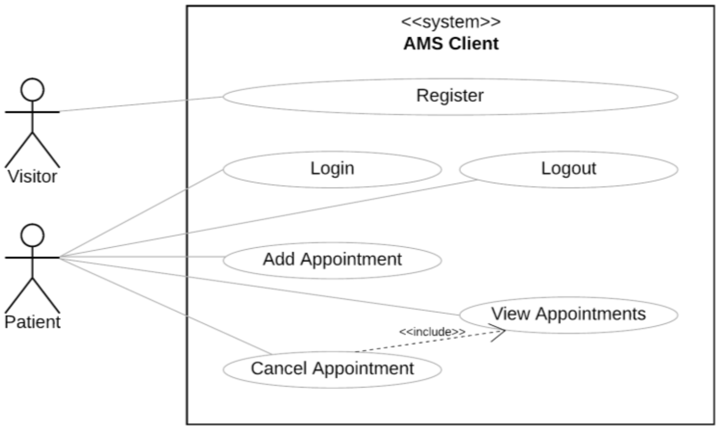

Figure 4 – UML use case diagram for AMS Client

S/N	Use Case	Description/Business Rules
1	Register	• If a patient is not registered, the system will prompt for the following information: o Identity Number (NRIC or Passport) o Password o First Name o Last Name o Gender o Age o Phone o Address
2	Login	• If a patient is registered, the patient will login into the system using: o Identity Number o Password
3	View Appointments	• The system list down all appointments for the patient.
4	Add Appointment	• The system shows a list of the doctors and allows the patient to select a doctor, and select the date for adding an appointment. • The system provides the availability of the doctors for the date.

		• The patient enters the time for finalizing the appointment.
5	Cancel Appointment	• The system shows the list of patient appointments. • The patient enters the appointment id to cancel the appointment.
6	Logout	• Logout if the patient is currently logged in the system.

Table 6 – Use case descriptions and business rules of Self-Service Kiosk

At the bare minimum, the system shall produce the following outputs of key business use cases, but not limited to:

Main Menu

```
*** Welcome to AMS Client ***

1: Register
2: Login
3: Exit

>
```

```
*** AMS Client :: Register ***

Enter Identity Number> S9867027A
Enter Password> 001001
Enter First Name> Sarah
Enter Last Name> Yi
Enter Gender> F
Enter Age> 22
Enter Phone> 93718799
Enter Address> 13, Clementi Road
Registration is successful!
```

```
*** AMS Client :: Login ***

Enter Identity Number> S9867027A
Enter Password> 001001
Login successful!
```

Patient Menu

```
*** AMS Client :: Main ***

You are login as Sarah Yi

1: View Appointments
2: Add Appointment
3: Cancel Appointment
4: Logout

>
```

```
*** AMS Client :: View Appointments ***

Appointments:
Id |Date       |Time  |Doctor
37 |2018-12-26 |15:30 |Robert Blake
63 |2019-03-01 |09:30 |Robert Blake
```

```
*** AMS Client:: Add Appointment ***

Doctor:
Id |Name
1  |Tan Ming
```

```
2  |Clair Hahn
3  |Robert Blake
4  |Tan Min

Enter Doctor Id> 1
Enter Date> 2019-03-10

Availability for Tan Ming on 2019-03-10:
9:00 09:30 10:00 10:30 14:00 14:30 15:00 15:30

Enter Time> 10:30
Sarah Yi appointment with Tan Ming at 10:30 on 2019-03-10 has been added.
```

```
*** AMS Client :: Cancel Appointment ***

Appointments:
Id |Date       |Time  |Doctor
37 |2018-12-26 |15:30 |Robert Blake
63 |2019-03-01 |09:30 |Robert Blake
105|2019-03-10 |10:30 |Tan Ming

Enter Appointment Id> 105
Sarah Yi appointment with Tan Ming at 10:30 on 2019-03-10 has been cancelled.
```

USER INTERFACE

For each client application, implement a suitable CLI interface consisting of menus, prompts, cues and feedback messages (both informative and error messages). Users should be able to interact with the CLI interface with relative ease.

SYSTEM QUALITY

All client applications should demonstrate some degree of input data validation and all server-side business processing should demonstrate some degree of business rules validation. Error handling should be done properly using Java checked exceptions. Error handling for each use case should reflect alternative and/or exceptional course(s) of action that deviates from the basic course of action as appropriate.

The quality of the coding will be assessed via criteria such as appropriate and correct use of the JPA EntityManager API methods, EJB component types, number of remote session bean business method invocation, etc.

ASSESSMENT CRITERIA

This group project is worth 35 marks out of the overall assessment for the whole module. In general, both members of the group will get the same score if there is no negative peer review.

The assessment criteria are listed in Table 5 below:

Component	**Marks**
Architecture and Design • Appropriateness and correctness of the logical data model, physical data model, and physical system architecture. • It also includes the implementation of the entity classes and the forward engineering of the underlying relational database tables.	10
System Functionalities – Clinical Admin Terminal • Implementation of use cases with respect to the logical data model as well as the complexity of the use cases that are implemented as measured by the fulfillment of the business rules.	10
System Functionalities – Self-Service Kiosk • Implementation of use cases with respect to the logical data model as well as the complexity of the use cases that are implemented as measured by the fulfillment of the business rules.	5
System Functionalities – Automated Machine Service Client • Implementation of use cases with respect to the logical data model as well as the complexity of the use cases that are implemented as measured by the fulfillment of the business rules. • It also includes web services exposed on the server-side.	5
System Quality • The quality of the user interface (Command Line Interface) including cues and input data validation. • The quality of the coding including general aspects such as proper exception handling and Java EE specific aspects such as correct use of annotations and the entity manager.	5
Total	**35**

Table 5 – Assessment Criteria

GENERAL ASSIGNMENT SCHEDULE

The general assignment schedule is shown in Table 6. Late submission will not be graded.

S/N	Week	Date	Activity	Remark
1	7	2 Mar	Release of the group project specification	
2	11	13 Apr	Submission of Project Deliverables Submission of Project Peer Review	Upload to LumiNUS Workbin by 23:59 (A grace period of 15 minutes)
3	11	14 Apr	Group Project Presentation	Presentation schedule, see Table 7.

Table 6 – General Assignment Schedule

PROJECT SUBMISSION INSTRUCTIONS

Place all the deliverables into a **single zip archive file** with the following folder structure:

- **doc folder**:
  - A Microsoft Word document containing the write-up of your solution's architecture and design.
- **src folder:**
  - Place all your NetBeans project folder(s) inside here.
  - You should "clean" the projects beforehand, the file size will be much smaller.
  - Test data, if any, such as the default system administrator account should be loaded using a singleton session bean decorated with the @Startup annotation. Do not use a SQL script to load the test data.
  - Ensure that your main enterprise application's EJB module contains a glassfish-resources.xml file and a persistence unit persistence.xml.
- **readme.txt:**
  - A text file containing the personal particular of all group members, including full name, matriculation number and email addresses.

Name the zip archive file with your project group name in the format "**GroupXX.zip**" where "XX" is your group number. Please double check your LumiNUS> class & group for the correct group number. Upload the zip archive file to the LumiNUS folder: Assignment 2 Submission. Each group should only need to upload one zip file. Should you upload more than 1 zip file, please delete those that are not required for evaluation, otherwise, the latest uploaded zip file will be used for evaluation.

The submission deadline is **Monday, 13 April 2019, 23:59 hrs** (A grace period of 15 minutes).

The Project Presentation will be held **Tuesday, 14 April.** The venue will be informed later**.**The schedule is given in the table below.

Group No.	Time (pm)	Group No.	Time (pm)
Group 01	2.00-2.20	Group 09	4.40-5.00
Group 02	2.20-2.40	Group 10	5.00-5.20
Group 03	2.40-3.00	Group 11	5.20-5.40
Group 04	3.00-3.20	Group 12	5.40-6.00
Group 05	3.20-3.40	Group 13	6.00-6.20
Group 06	3.40-4.00	Group 14	6.20-6.40
Group 07	4.00-4.20	Group 15	6.40-7.00
Group 08	4.20-4.40		

Table 7– Presentation Schedule