## OOSD rehearsal

Wednesday, December 30, 2020 8:30 PM

Good morning to Dr Chai and fellow audience. We're from Group 7 presenting Online Hospital Blood Test Appointment.

First of all lemme introduce my team member, thy are ... And myself ...

DEMO CODE

Q&A

In our class diagram, we have 5 class. Thy are Homepage class, Admin class, Patient class, Appointment class and Payment class, inheritance

1. How can u relate ur class diagram to ur code?

GG

2. How do u implement ur objectives based on ur case study into ur class diagram?

GG 3. Based on question 2, how did u implement those objectives within ur code?

4. What relationship did u have in ur class diagram? Association- In our project, the relationship between Admin and Patient is shown

which is register.

Aggregation- For our project, the class Patient is made up of one or more Appointment. In aggregation, the contained classes are never totally dependent on the lifecycle of the container. Here, the Patient class will remain even if the Appointment is not available.

Composition- For our project, if Appointment is composed of class payment. The

appointment. So, if appointment is not functioning all the payment also removed.

appointment could contain payment, while each payment belongs to only one

5. Based on ur class diagram, what do u mean by multiplicity from class xx to class xx?

Each patient registered by one admin. Each patient book one or many appointments

Each admin register one or many patients

Each appointment booked by one patient

Each appointment has one payment

6. Which class is inherited by other class? classes Admin and Patient inherit class Homepage

class Homepage, Admin, Patient-> inheritance...

Each payment has one or many appointment

7. Give me a reason why u create this class...

Every attribute and method in class Homepage

8. For class Admin, which attributes are inherited from class Homepage?

9. Which class did u apply concept polymorphism (override, overload)?

class appointment and payment-> function for patient to fulfil objectives

Overloading -> same method difference parameter (Homepage.h [code line 18 & 19] Overriding -> homepage.login() in admin and patient implementation file

10. Why u encapsulate these attributes and operands to private, protected, public?

11. Based on ur code, which method can be related to ur objectives in case study?

14. which part in ur code apply with oo concept -- encapsulation, polymorphism,

Encapsulation -> built on the idea of data hiding. Restrict access of certain attribute

Polymorphism -> direct translation: many form. Allows you to determine what

you want to built a new class that uses the stuff from the previous class but you

-> allow code reusability. For example, you've an existing class and

Protected-> can be accessed in derived classes. Public-> can be accessed by any function or other class.

Private-> To avoid modifying member/ declaration.

Class Homepage, Admin, and Patient.

Command Line Interface

12. What concept did u apply for ur User Interface design?

abstraction, inheritance, class, object?

13. What is the functionality of ur object in code?

Protected-> can be accessed in derived classes. Public-> can be accessed by any function or other class.

or method of object to whatever is calling that object.

separate header file and implementation file from main file.

Private-> To avoid modifying member/ declaration.

function to run. Overloading -> same method difference parameter (Homepage.h [code line 18 & 19]

Overriding -> homepage.login() in admin and patient implementation file -> only show the necessary details to user of the object. Example

wanna add additional feature. Classes Admin and Patient inherit class Homepage Extra information

- holi

- holi

- holi

register patient - holi view user list

homepage

TOD

GG

GG

view all appointment patient login(ali) - chris edit patient detail - chris view profile - chris

Why empty constructor?

For integer declare only.

change patient password - amar make appointment - amar view appointment - amar patient login(ahkao) - zaki edit patient detail - zaki view profile - zaki make appointment - lee ji view appointment - lee ji

atient login(muthu) - lee ji edit patient detail - miza view profile - miza make appointment - miza view appointment - miza view all appointment - holi