Tests & Quizzes

CS425-SWE-202204-FinalExam

Return to Assessment List

Part 1 of 2 - / 14.0 Points

Theory and Knowledge-based Questions

Question 1 of 11 2.0 Points

For large and complex software systems, Orchestration works better than Choreography

✓
○ True
○ False

Answer Key: False

Question 2 of 11 2.0 Points

In a layered architectural design/solution, Classes in the service layer are allowed to call classes of the data access layer

✓
○ True
○ False

Answer Key: True

Question 3 of 11 2.0 Points

In a layered architectural design/solution, Classes in the data access layer are allowed to call classes of the domain layer

✓
○ True
○ False

Answer Key: True

Question 4 of 11 2.0 Points

In a layered architectural design/solution, Classes in the domain layer are allowed to call classes of the data access layer

✓
○ True
○ False

Answer Key: False

Question 5 of 11 2.0 Points

A characteristic of a good Class design is one that supports high cohesion and tight coupling.
(1 point) Give a rationale for your answer.

✖
○ True
○ False

Rationale:
because within the given class there should be communication between elements in that class and should get as much data as possible within themselves not from another class

Answer Key: False

Question 6 of 11 2.0 Points

In Software Testing, Mocking, is a technique that makes it possible to write and execute a Unit Test in isolation, even when the unit of functionality has a dependency. True or False?

✔
○ True
○ False

Answer Key: True

Question 7 of 11 2.0 Points

The JUnit Testing framework is for used for Unit Testing only, and cannot be used for implementing Integration tests. True or False?

✔
○ True
○ False

Answer Key: False

Part 2 of 2 - Part2 / 46.0 Points

Software Requirements Analysis and Design skills questions

Question 8 of 11 10.0 Points

Download the attached SpringBoot/SpringWebMVC application project (hcmcwebapp.zip) which is not secured. Add basic spring security to the application such that when the homepage is accessed, the application should prompt the user for valid credential (i.e. username and password). In your implementation, make it so that the valid username/password pair is 'hcmc-admin/hcmc1234'.

Take screenshots of your work (showing the steps you followed/changes you have applied). i.e. Specifically, take two (2) screenshots of your IDE tool, showing the code or configuration additions or changes that you have made.

hcmcwebapp.zip 75 KB

question8.zip (1,469.74 KB)

Question 9 of 11 14.0 Points

Given below, is the Problem Statement/Description of a Software system for MIU Computer Science department, named "eRegistrar", which is a web application for managing Student-Course Registrations. Assume you have been hired by the Department to design and develop this software solution. Your task for this question will include:

- Elicit and present the Software Requirements in a Use-case diagram

- Create the system Analysis & Design artifacts (UML diagrams) for the solution.

Problem Statement/Description:

The eRegistrar is a system that will enable students to register online for courses, as well as track their progress toward completion of their degree.

The CS department conducts admission of students into its degree programs, in 4 quarterly entries. Upon admission, students are offered enrollment into one of (currently) two Computer Science areas of specialization, namely Data Sciences and Web Applications. In addition to the areas of specialization, students are also placed into one of two tracks (FPP or MPP), at the inception of their program, depending on the grades they score in a pretest.

When a student first enrolls, he/she uses the eRegistrar to create a plan of study that lists the courses he/she plans on taking to satisfy his/her particular degree program, and is also assigned a faculty advisor. The eRegistrar will verify whether or not the proposed plan of study satisfies the requirements of the degree that the student is seeking.

Once a plan of study has been established, then, during the registration period preceding each semester, students are able to view the schedule of classes online and choose whichever classes they wish to attend, indicating their preferred section (i.e. the course as well as the block and faculty), especially if the class is offered by more than one professor.

When a student submits their registration for classes, the eRegistrar system will verify whether or not the student has satisfied the necessary prerequisites for each requested course by referring to the student's online transcript of courses completed and grades received (the student may also review his/her transcript online at any time).

A student is registered for their selected class(es), if and only if: (a) the prerequisites for the requested course(s) are satisfied, (b) the course(s) comply/complies with the student's plan of study requirements, and (c) there is room available in each of the class(es).

If (a) and (b) are satisfied, but (c) is not, the student is placed on a first-come, first-served wait-list. If a class/section that he/she was previously waitlisted for becomes available (either because some other student has dropped the class or because the seating capacity for the class has been increased), the student is automatically enrolled in the waitlisted class, and an email message to that effect is sent to the student. It is the student’s responsibility to drop the class if it is no longer desired; otherwise, he/she will be billed for the course. Students may drop a class up to the end of the first week of the semester in which the class is being taught.

The CSRegistrar, who is a Computer Science department staff, will be the main operator of the eRegistrar system, and she is responsible for initially assigning students to their respective tracks and faculty advisors. She is also responsible for allocating classes to respective classrooms, which are located in various buildings across the university campus. She is also responsible for receiving and managing requests for change of class, which are sent-in by students, through the eRegistrar system.

Your tasks for this question are:

2.1 Draw the use case diagram

2.2 Draw the Analysis/Design-level UML class diagram, including the following architecture and design choices in your consideration:

1. The data for the eRegistrar system will be stored/persisted in a Database.
2. A Layered architectural style
3. The system will have a Graphical User Interface (GUI)

For this diagram, make sure to clearly group the classes into their specific architectural layers.

For your class diagrams, make sure to show all important aspects in the diagram. i.e. Show the important classes, attributes, methods, relationships, multiplicity, role etc.

For all your classes also show what kind of class it is (i.e. entity, domain, service, etc.)

For submission, put your diagrams/files into a single .zip file, which you upload here.

Question 10 of 11 20.0 Points

Implementing testing with JUnit

Download the attached zip file which contains a Java Command Line Interface (CLI) application named, PatientsMgmtApp. It is a Maven project.

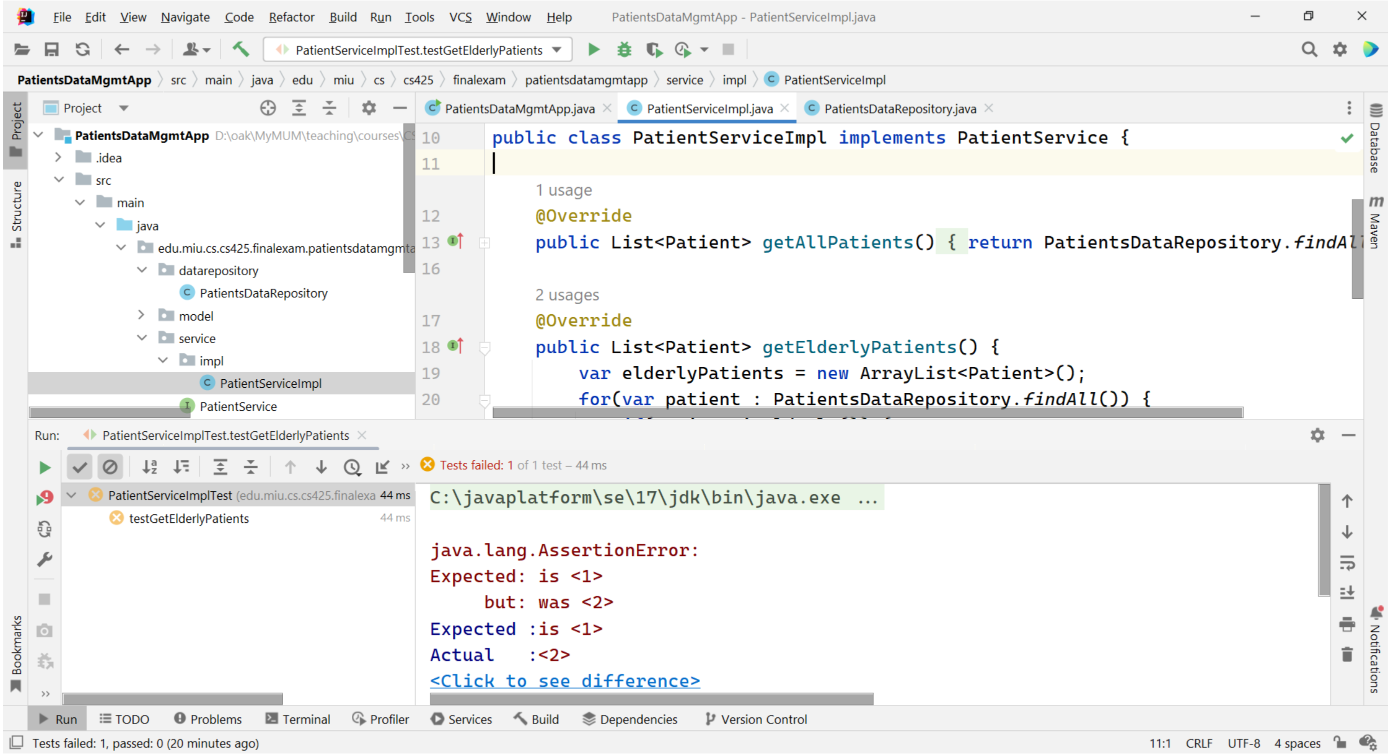
Load the project into your Java Application Development IDE or Code Editor tool.

For this question, you are expected to do the following:

Using the JUnit Testing Framework, implement Test for the method named, getElderlyPatients, found in the class named, PatientServiceImpl. Note: The method as implemented, is faulty. Therefore, execute your test and take a screenshot of the result, as presented in your IDE, showing a Failing test (See sample screenshot below).

Next, correct the faulty implementation of the method. Then execute your test once again and take a screenshot of the result, as presented in your IDE, showing a Passing test.

To submit, put your two (2) screenshots into a folder/directory named “screenshots” created inside your project folder and then zip the project folder and upload the zip file as your submission for this question.


File Edit View Navigate Code Refactor Build Run Tools VCS Window Help
PatientsDataMgmtApp - PatientServiceImpl.java
PatientServiceImplTest.testGetElderlyPatients
PatientsDataMgmtApp 〉 src 〉 main 〉 java 〉 edu 〉 miu 〉 cs 〉 cs425 〉 finalexam 〉 patientsdatamgmtapp 〉 service 〉 impl 〉 PatientServiceImpl
Project
PatientsDataMgmtApp D:\oak\MyMUM\teaching\courses\
.idea
src
main
java
edu.miu.cs.cs425.finalexam.patientsdatamgmta
datarepository
PatientsDataRepository
model
service
impl
PatientServiceImpl
PatientService
PatientsDataMgmtApp.java
PatientServiceImpl.java
PatientsDataRepository.java
Database
Maven
Structure
Bookmarks
Notifications
Run:
PatientServiceImplTest.testGetElderlyPatients
PatientServiceImplTest (edu.miu.cs.cs425.finalexa 44 ms
testGetElderlyPatients 44 ms
Run TODO Problems Terminal Profiler Services Build Dependencies Version Control
11:1 CRLF UTF-8 4 spaces

```
public class PatientServiceImpl implements PatientService {

    1 usage
    @Override
    public List<Patient> getAllPatients() { return PatientsDataRepository.findAl

    2 usages
    @Override
    public List<Patient> getElderlyPatients() {
        var elderlyPatients = new ArrayList<Patient>();
        for(var patient : PatientsDataRepository.findAll()) {
```

Tests failed: 1 of 1 test – 44 ms

```
C:\javaplatform\se\17\jdk\bin\java.exe ...

java.lang.AssertionError:
Expected: is <1>
     but: was <2>
Expected :is <1>
Actual   :<2>
<Click to see difference>
```

Tests failed: 1, passed: 0 (20 minutes ago)

PatientsDataMgmtApp.zip 17 KB

PatientsDataMgmtApp.zip (1,501.06 KB)

Question 11 of 11 2.0 Points

Is the Test you have implemented in Part2 Question 3 above a Unit Test or an Integration Test?

That is an Integration test because it depends on the execution of the real Patients repository