

## NATIONAL UNIVERSITY OF SINGAPORE

**CS3219 – SOFTWARE ENGINEERING PRINCIPLES and PATTERNS**  
(Semester 1 AY2015/2016)

Time Allowed: 2 Hours

**INSTRUCTIONS TO CANDIDATES**

1. Write your Student Number only. Do not write your name.
2. This assessment paper contains **NINE** questions and comprises **EIGHTEEN** printed pages.
3. Answer **ALL** questions within the space in this booklet.
4. This is a **CLOSED** book assessment. An A4 size cheat-sheet (written both sides) is allowed.

STUDENT NO: \_\_\_\_\_

This portion is for examiner's use only

Question	Marks	Remarks
Q1	/5	
Q2	/2	
Q3	/3	
Q4	/5	
Q5	/5	
Q6	/8	
Q7	/7	
Q8	/7	
Q9	/8	
<b>Total</b>	<b>/50</b>	

**Question 2****2 marks**

Following is a requirement for a banking application, that is less than ideal. **In your answer, either briefly write the problems you spot or write a refined requirement.**

You can assume that the SRS of the application does not include any other requirement about Super Plan or Proof of Investment.

*If the Super plan is not selected and Proof of Investment is not provided , the customer should automatically default into the Ordinary plan.*

**Answer:**

**Question 3 :**

**3 marks**

You are designing a framework that allows for pre-packaged components such as session management, authentication, etc. **What design pattern(s)** would you choose for implementing the request execution in this framework i.e. its components would receive and send requests to other components; and would perform the appropriate operations in response to a received request.

**Answer:**

**Question 4****5 marks**

Assume that you are building an application for analysing source code. The application will allow transformations on source. There are 3 kind of components that will perform specific code transformations. :

Code Parsers (e.g., for C++, Java) that can read in source code and produce internal program representation

Code Analysers that work on the internal program representation to analyse it, and

Code Generators that can transform the internal program implementation back into source code.

The application should be easily extensible with new components designed to work with source code or internal representation(s).

**What architectural style would be most appropriate for this application? Why? Draw a sketch of the application architecture.**

**Answer:**

**Question 5****5 marks**

This is about designing a system 'Stadium' to provide multiplayer computer games that support virtual communities. 'Stadium' will support five types of users:

- Admin: should be able to define new games, define new matches, and manage users.
- Organizers : should be able to organize and announce new matches/tournaments.
- Players: should be able to register in a stadium and play matches/tournaments.
- Spectators: should be able to monitor any match and check score and statistics of past matches.
- Advertisers: should be able to upload new advertisements.

The 'Stadium' is to be designed to provide an infrastructure to support users in carrying out above and related tasks. Further, the system will support 10 parallel tournaments with 64 players each and 100s of spectators per tournament. The 'Stadium' server must be available 24 hours a day.

**Identify sub-systems for 'Stadium'. Write 1 to 2-sentence description to state the purpose of each sub-system. Fill the table with sub-systems and 1-2 sentence descriptions. You can make reasonable assumptions.**

**Answer** (*Add rows as needed. Write your assumptions, if any at the end of table*):

Sub-System	Purpose

**Blank page**

**Question 6****8 marks**

We have collected some metrics for two java packages as part of a quality review. Following tables summarize the values for LOC, WMC, RFC, DIT, and NOC metrics for the two packages. Table 1 and Table 2 represent Package1 and Package2 metrics respectively.

**Table 1 : Package 1 metrics**

Type	LOC	WMC	RFC	DIT	NOC
Class 1	41	7	22	3	1
Class 2	166	22	61	2	1
Class 3	12	2	6	3	0
Class 4	27	7	6	1	0
Class 5	295	25	58	4	0
Class 6	18	4	9	4	0
Class 7	3	1	0	1	0
Class 8	7	5	3	1	0
Class 9	7	6	4	1	0
Class 10	11	5	6	4	0
Class 11	17	4	12	2	0
Class12	35	6	13	3	0
Class 13	52	12	16	3	0
Class 14	6	3	3	1	0
Class 15	32	7	5	3	0
Class 16	19	8	5	3	0
Class 17	101	10	23	3	0
Class 18	54	30	19	1	0
Class 19	81	22	31	4	0
Class 20	18	2	7	1	0
Class 21	5	1	1	3	0
Class 22	68	10	20	3	0
Class 23	79	10	23	4	0
Class 24	17	4	2	1	0
Class 25	5	2	3	3	0
Total classes : 25, Total methods : 103 Total Lines of Code : 1023					

Table 2: Package 2 metrics

Type	LOC	WMC	RFC	DIT	NOC
Class 1	25	1	2	1	0
Class 2	86	16	3	1	0
Class 3	69	7	4	2	0
Class 4	90	14	21	1	0
Class 5	108	10	26	3	0
Class 6	4	1	1	2	0
Class 7	73	9	25	1	0
Class 8	175	31	60	1	0
Class 9	32	4	11	1	0
Class 10	187	21	43	1	1
Class 11	40	3	12	2	0
Class 12	9	2	2	2	0
Class 13	51	7	10	2	0
Class 14	25	3	5	1	0
Class 15	174	15	43	3	0
Class 16	13	2	6	2	0
Class 17	163	28	48	1	0
Class 18	26	5	3	2	0
Class 19	137	21	38	1	0
Class 20	38	3	8	3	0
Total classes : 20 , Total methods : 134 Total Lines of Code : 1729.					

**Suggest top 5 classes, in order of high to low priority, for each package you would prioritize for a detailed review and re-design. Write a reason(s) for your suggestions or describe how you arrived at your suggestions you have made in your answer.**

**Answer (continue on next page):**

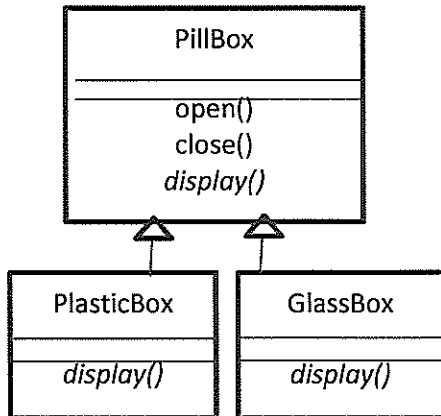


**Blank page**

## Question 7

5 + 2 marks

Bean, a software engineer, designed an App which requires a Pill Box to be manipulated. The initial design of the system used standard OO techniques and created following classes and methods.



Modify the design to accommodate following changes/requirements .

1. To add glow behavior to PillBox(es).
2. WoodenBox is to be added. They don't glow.
3. WoodenBox don't open or close in the same way as other boxes ie they pull open and push close instead of twist open and twist close.
4. DecorativeBoxes are to be added. They neither open/close nor glow .

**(a) Illustrate modified design by drawing the classes and relevant methods.**

**(b) Do briefly mention which design principles did you apply to modify the design.**

**Answer :**

Blank page

**Question 8****7 marks (4+3)**

Object oriented applications consist of a collection of components that need to communicate with each other. This interaction can be accomplished via a messaging model (or message design pattern).

- (a) **Identify key components of a messaging model, writing a 1-sentence description for each.**
- (b) **For a secured message transfer a Proxy could be included in the message request-reply sequence. Keeping messaging as the only mechanism of communication among participants(key components identified in (a) above and the Proxy), draw a sequence diagram to illustrate a message request-reply sequence. Use meaningful labels for the interactions among components.**

**Answer:**

**Question 9****8 marks (4+4)**

OAuth is an authentication protocol that allows you to approve one application interacting with another on your behalf without giving away your password. Following are descriptions of few of the terms: Roles, Tokens, and Grant Types

Roles: OAuth defines 4 roles

*Client:* application requesting access to a resource server (it can be your PHP website, a Javascript application or a mobile application).

*Resource Server:* server hosting protected data (for example Google hosting your profile and personal information).

*Resource Owner:* generally yourself.

*Authorization Server:* Server issuing access token to the client. This token will be used by the client to request data from the resource server. Resource server can be the same as the authorization server (same physical server and same application), and it is often the case.

Tokens: Tokens are random strings generated by the authorization server and are issued when the client requests them. There are 2 types of tokens:

*Access Token:* This token is sent by the client as a parameter or as a header in the request to the resource server. It has a limited lifetime, which is defined by the authorization server.

*Refresh Token:* This token is issued with the access token. It is not sent in each request from the client to the resource server. It is sent to the authorization server for renewing the access token when it gets expired.

If a Client wants to retrieve data from a resource server using OAuth, it has to send Redirect URLs (URLs of the client for receiving access token) in its request for authorization code.

Grant Types: OAuth defines several grant types. Given below are 2 of the grant types.

**Authorization Code Grant**

It is used if the client is a web server. It allows client to obtain a long-lived access token.

**Example:**

Resource Owner: you

Resource Server: a Google server

Client: any website

Authorization Server: a Google server

**Scenario:**

A Client wants to obtain information about your Google profile.

It seeks an authorization code.

You are redirected by the Client (the website) to the Authorization Server (Google) for consent.

If you authorize access, then, the authorization server sends an authorization code to the client in the callback response.

Next, the authorization code is exchanged for an access token between the client and the authorization server.

The website is now able to use this access token to query the resource server (Google again) and retrieve your profile data.

Authorization Server also sends other information with the access token, such as the token lifetime and eventually a refresh token.

**Implicit Grant**

It is typically used when the client is running in a browser using a scripting language such as Javascript. This grant type does not allow the issuance of a refresh token.

**Example:**

Resource Owner: you

Resource Server: a Facebook server

Client: a website using AngularJS for example

Authorization Server: a Facebook server

**Scenario:**

The client (e.g. AngularJS) wants to obtain information about your Facebook profile. It requests for an access token. You are redirected by the browser to the authorization server (Facebook).

If you authorize access, the authorization server returns with the access token.

This access token can now be retrieved and used by the client (AngularJS) to query the resource server (Facebook).

- a) **Discover and Document the API operations** for the Authorization Server based on the above description.
- b) **Illustrate interactions among different components** using any suitable notation e.g. a sequence diagram, **for the Implicit grant flow**  
*Use meaningful labels for the operations.*

**Answer:**

**Answer:**

Blank page



Blank page

--- End of paper---