Objective - Common Architectures - Test

Review of attempt 1

Finish review

Started on	Monday, 18 November 2013, 04:35 AM
Completed on	Monday, 18 November 2013, 04:35 AM
Time taken	15 secs
Grade	**1** out of a maximum of 29 (**3**%)
Feedback	FAIL

Show **All** / Correct / In-correct

1. Which of the following are NOT true about 2-tier architecture models?
Select three choices.

- [x] A. Clients may be validation intensive, consequently requiring powerful hardware ✗
- [x] B. Any change to one tier typically affects both tiers ✓
- [ ] C. Represents a single point of failure ✓
- [ ] D. Each client makes a direct connection with the server ✗
- [ ] E. They are not very maintainable ✗
- [ ] F. They are easily extensible ✓

Choices B, C and F are correct as the corresponding statements are FALSE.

Statement B is incorrect because a change to one tier need not necessarily affect the other tier. For example, consider an Order Entry system where the users need to see transaction history for individual customers. A new screen can be developed to take data from the transaction tables and display it on the client. In this example, no changes are required on the server. Consider an alternate example where a stored procedure accessing data spread in multiple tables now queries a materialized view.

No changes would be required to clients that call the stored procedure, as long as all input and output remains the same. Although it may be true in many cases, Client/Server architectures do not always represent single points of failure. With single database servers, of course if the server fails, clients may not be able to connect to the database. However in multi-database applications, if parts of the client or server fail, other parts of the system may still be accessible. Hence, statement C is incorrect and hence incorrect choice.

2-tier architectures are not easily extensible. As the presentation and business logic is tightly coupled, Any changes or extensions to the existing code takes lot of time to implement on a 2-tier architecture model. Statement A correctly reflects what happens on a client in 2-tier applications.

Statement D explains how clients interact with the database and statement E clearly lists that 2-tier architectures are not easily maintainable because of constant software updates required on the clients. Hence, choices A, D and E are incorrect.

Incorrect
Marks for this submission: 0/1.

Feedback to Author

2. You have been asked to analyze the Non-Functional requirements part of your new project initiation based on your previous projects experience.
What statements are true about Non-Functional requirements from the given options?

- ( ) A. Manageability is the ability to correct flaws in the system whereas maintainability is the ability to ensure the continued health of the system. ✗

Choice B is correct.

Maintainability (Cade 8) "is the ability to correct flaws in the existing system without impacting other components of the system" and Manageability (Cade 9) "is the ability to manage the system to ensure the continued health of a system with respect to scalability, reliability, availability, performance and security." Hence, choice B is correct.

The definitions in choice A are in the reverse order and are incorrect in choice C.

Choice D is incorrect. Manage application components are not tightly coupled is related the application code components. When designing layered architecture, or within the each layer the components should be loosely coupled so that any modifications or extensions will be easy to handle part of future enhancements.

Hence, choices A, C and D are all incorrect.

Correct
Marks for this submission: 1/1.

Feedback to Author

3. Mega Soft Inc company planning to release new product upgrades to attract more customers. The higher management decided to rollout these changes using the existing system resources (currently running on one web server) with minimal changes. The expected user base will be slightly increased, once it's live in production.
How do you handle this scenario in terms of providing high availability and performance?

- A. By implementing Vertical Scaling only ✓
- B. By Implementing Horizontal Scaling only ✗
- C. Existing system resources will handle the expected load ✗
- D. By implementing both Horizontal and Vertical Scaling ✗

Choice A is correct.
The problem statement from the requirement is that, "existing system resources with minimum changes" and "user base will be slightly increased" is the key to solve the problem. The scenario in the question describes the need for vertical scaling. (Vertical scaling is the process of adding extra RAM or faster CPUs etc to existing machines). This configuration will handle more load with better performance.

Choice B is incorrect. Based on the given scenario, implementing horizontal scaling will take lot of time as you need to add another machine and configure the web server instances and it comes with the additional cost as well when compared to vertical scaling. The changes should be minimum part of the requirement.

Choice C is incorrect. As the system is configured with only one web server and the additional configuration details are not mentioned. The existing setup may fail due to increase in load at some point of time and hence availability will get affected.

Choice D is incorrect. Implementing both is actually a good choice, but takes lot of time and additional cost incurred . The solution you are looking at should be with minimal changes to achieve the availability and performance of existing system.

Incorrect
Marks for this submission: 0/1.

Feedback to Author

4. What statements are true about a three-tier system when compared to a two-tier system?
Select two choices.

- A. A three-tier system has thin clients whereas a two-tier system will always have thick clients. ✓
- B. A three-tier system is easier to manage than a two-tier system. ✗
- C. A three-tier system is easier to secure. ✗
- D. A three-tier system can be both vertically and horizontally scaled whereas a two-tier system can only be horizontally scaled. ✗
- E. A three-tier system is more extensible than a two-tier system. ✓

Choice A and E are the correct.

A three-tier system can be classified as a loosely coupled, highly scalable and extensible solution. A simple system (such as a one-tier system) is much easier to manage than a complex three-tier system (remember altering a tier

comes under extensibility). Yes, Two-tier system will have thin/thick clients depending on where the business logic is implemented (Client layer Vs Data Layer)

Choice B and C are incorrect. A three-tier system will be harder to secure than a typical two-tier system because of the extra network calls etc. The two-tier system is easier to manage than a three-tier system due to less configuration changes part of server management.

Choice D is incorrect because a two-tier system can be scaled both horizontally and vertically. In two-tier system also you can have both horizontal (adding additional data bases to different machines) and veridical scaling (increasing some CPU units part of existing setup).

Incorrect
Marks for this submission: 0/1.

Feedback to Author

5. You are the Project Manager for the new project and expected to gather requirements.
Which of the following is the most important item that should be considered when designing an application part of Functional and Non-Functional requirements?

- A. Scalability ✗
- B. Maintainability ✗
- C. Reliability ✗
- D. Meeting the needs of the customer ✓
- E. Performance ✗
- F. Ensuring the application is produced on time and within budget ✗

Choice D is correct.

The most important consideration when designing an application is that it meets the needs of the customer. Ensuring the application is produced on time and within budget is something that should be done but it is not the number one concern. The application does not have to be the best possible solution under the circumstances. As long as it meets the customer's needs, it is considered adequate.

- Performance - A measure of the system in terms of response time or number of transactions per unit time. Load Distribution (e.g. DNS Round Robin) and Load Balancing are two techniques that aid in higher performance. Other development and deployment related tasks such as Application Tuning, Server Tuning, and Database Tuning also help the system perform better.
- Scalability - The ability of a system to perform and behave in a satisfactory manner with increases in load.
- Reliability - The ability of a system to assure the integrity and consistency of the application and all its data as the load increases. <
- Availability - The ability of a system to assure that all services and resources are always accessible. This can be achieved through fault tolerance (the ability to prevent system failures in case of service(s) / component(s) failures, commonly implemented via redundancy) techniques such as Active and Passive Replication.
- Extensibility - The ability to easily add new functionality to the existing system. This can be achieved by using best practices and well-defined architecture and design techniques.
- Maintainability - Ability to easily correct flaws in the existing system.
- Security - The ability to protect a system and all its components and services against potential attacks. Security attacks generally try to compromise confidentiality and integrity of the system. Sometimes they also take the form of 'Denial of Service' (DoS) attacks that bring down a system by flooding it with messages. Security can be addressed by the use of technologies (firewalls, DMZ, data encryption, Digital Certificates and so on) and methodologies (good security policies and procedures.)
- Manageability - The ability to monitor and perform preventive maintenance on a system.

Incorrect
Marks for this submission: 0/1.

Feedback to Author

6. Your have been contracted by a company to help them improve the performance of their sales application. You have suggested that the hardware on which the application is currently deployed (two web servers and a database server) be migrated to 3 web servers, an application server and a database server (all on different machines.) You assure them that all the software re-writes needed will be well worth it in the end.

What are the characteristics of your suggested architecture? Select three choices.

- a. Fat Clients ✗

Choices B, C and D are correct.

The system you have suggested they migrate to is a 3-tier system. The characteristics of a 3-tier system are thin clients, good separation of business logic and good scalability. This is due to the fact that each tier is separate from the other (for example, it would be possible to change the data store without affecting the business logic).

Choice A is incorrect; the suggested system has thin clients, the business logic residing on the application server, in the middle tier.

Since there is a good separation of business logic, choices E and G are incorrect.

Choice F is incorrect as the 3-tier nature of the system makes it very scalable.

Incorrect
Marks for this submission: 0/1.

Feedback to Author

7. Users of your new application are complaining that the new web front end does not perform the same as the thick client version they were using before. You are going to re-design the application to use JavaScript and AJAX to enhance the user experience. All the functionality and business logic in thick client will be coded into Java Script.

What are two drawbacks of this implementation? Select two choices.

- a. Portability to all browsers needs to be addressed. ✓
- b. Your code may be exposed. ✓
- c. Reduction of page refreshes to enhance user experience. ✗
- d. Simulating client state on the browser and reducing the number of views. ✗

Various browsers use their own JavaScript interpreter, and sometimes browser vendors may chose not to implement a bit of JavaScript. There are also differences in support among various versions of the same browser. So, the application needs to be tested for browser compatibility.So, Option A is correct.

When a user enters the url in the browser, JavaScript is also downloaded to his machine exposing the business logic.So, Option B is correct.

Option C is incorrect because using AJAX and JavaScript libraries will allow you to design a client that can mock state and minimize page refresh to simulate thick clients.

Option D is incorrect because simulating the client state is a way to address the thick client.

Incorrect
Marks for this submission: 0/1.

Feedback to Author

8. Your new application requirements include dynamic front ends, personalized pages, personalized colors, and personalized fonts. Users should be able to access system and bring up their customized experience without having to save their preferences locally. This system will be used by all hotels in the chain and there is no interest to have to support each hotel individually. The front end solution should be easy to debug.

What is the solution for this new architecture?

- a. A thin user interface that is browser based and utilizes JSP's and CSS with user preferences saved locally in a cookie. ✗
- b. A thick client user interface utilizing AWT and user preferences saved locally in a cookie. ✗
- c. A thick client user interface utilizing Swing and user preferences saved locally in a cookie. ✗
- d. A thin browser based client utilizing JSF and CSS with user preferences persisted to database. ✓

Option D is correct.

Options A, B and C are incorrect because users want to use any machine and not have to save their preferences locally. Cookies save local information on a particular machine only.

Incorrect
Marks for this submission: 0/1.

Feedback to Author

9. What are the possible scenarios where you will choose a dynamic dispatch invocation using JAX-WS part of your web service design from the following options?
Select three choices.

- A. When interoperability with legacy JAX-RPC or non-WS-I compliant Web services is required ✓
- B. To invoke a Web service with xml/http binding and not traditional SOAP binding ✓
- C. To invoke a Web service by using a data binding other than JAXB ✓
- D. To invoke a Web service by using a data binding technology JAXB ✗

Option A, B and C are correct.

Choice D is incorrect because for dynamic dispatch invocation, there is no need of data binding using JAXB technology.

For more information, please refer:
http://jax-ws.java.net/nonav/2.1.1/docs/UsersGuide.html

Incorrect
Marks for this submission: 0/1.

Feedback to Author

10. Which of the following statements are true about JAX-WS web services in Java EE environment?

- A. Invoke the remotely deployed web service endpoints operation dynamically by synchronous communication only ✗
- B. Invoke the remotely deployed web service endpoints operation statically by synchronous communication only ✗
- C. Invoke the remotely deployed web service endpoints operation statically for both synchronous and asynchronous calls ✓
- D. Invoke the remotely deployed web service endpoints operation dynamically for both synchronous and asynchronous calls ✓

Choice C and D are correct.

JAX-WS specification provides support for web services that use the JAXB API for binding XML data to java objects mapping.
JAX-WS specification describes the support for message handlers that can process requests and responses using both synchronous and asynchronous calls

The new Dispatch API introduced in JAX-WS support a fully dynamic service invocations.

For more information, please refer:
http://docs.oracle.com/javaee/5/tutorial/doc/bnayk.html
http://jax-ws.java.net/nonav/2.1.1/docs/UsersGuide.html

Incorrect
Marks for this submission: 0/1.

Feedback to Author

11. Consider application security measures such as encryption, SSL, authentication and authorization and validations part of login process.
What is true about Performance and Security in an application?

- a. They are not related to each other in any way ✗
- b. High Security generally results in high performance ✗
- c. High performance is a pre-requisite for secure systems ✗
- d. Security and Performance are inversely proportional ✓

Choice D is correct.

Adding new security measures typically affect the application performance negatively. Security techniques such as encryption/decryption, SSL,authentication and authorization checks make our application execution slower than if we don't have all these security protections. Even a simple login screen would reduce performance, in the sense that user gets slower access to the functionality of the system. Hence, generally speaking, increased security results in a lower performance.

Choice D is therefore correct. Hence, choices A and B are incorrect.

A high performance is not a technical pre-requisite for secure systems although it may be a good idea to increase performance wherever possible to offset the performance loss due to higher security. Hence, choice C is incorrect.

Many poorly engineered secure systems exhibit poor performance. Hence, choice E is incorrect.

Incorrect
Marks for this submission: 0/1.

Feedback to Author

12. Happy Joe Banking Corporation is building a Banking application to provide online access to their account holders. They have chosen two SUN 450s for their web server cluster and one SUN E10000 for their application server. The business requirements indicate that to become a customer, a person must have at least a primary checking account with the bank. Further, since the customer will be using the Internet to view confidential information, security is considered paramount.

What do you understand about the requirements of the system?

- a. The need for Security is a classic example of a functional service level requirement and the checking account rule, an example of non-functional QoS requirement. ✗
- b. The discussion about Security and the mandatory checking account both illustrate functional service level requirements. ✗
- c. Neither Security nor the mandatory Checking Account is an example of any kind of requirements, theoretically speaking. ✗
- d. Security is an Architectural non-functional requirement and the Mandatory Checking Account a functional design requirement. ✓
- e. They are both examples of Business Use Cases. ✗

Choice D is correct.

Successful software architecture deals with addressing the non-functional service level requirements of a system. The Design process takes all functional business requirements into account. Security is considered a non-functional requirement and specific business rules, such as the one described with the checking account are considered functional requirements.

Choice D is the only choice that accurately describes this. Choice A is incorrect because the functional and non-functional requirements are flipped over.

Choice B is incorrect because only one of them is a functional requirement.

Choice C is incorrect because as described above, one of them is a functional requirement and the other, a non-functional requirement.

Finally, Choice E is incorrect because business analysis may start with use cases (where the checking account rules may be captured), but this discussion is specifically questioning functional vs. non-functional requirements.

Incorrect
Marks for this submission: 0/1.

Feedback to Author

13. In a distributed systems environment, the replication is mainly used to provide fault tolerance.
Which of the following are true about Active Replication over Passive Replication? Select three choices.

- A. It is a fault-tolerance mechanism. ✓
- B. It is achieved by taking cold back ups. ✗
- C. Each replica attempts to process each request but extra calls may be intercepted via an interceptor. ✓

Choices A, C and F are correct.

Active Replication is a fault tolerance mechanism where each replica is identical to the main service and attempts to service each request. An interceptor is used to block extra responses. Active replication is similar in concept to hot backups. Hence, choices A, C and F are correct.

Active Replication is not achieved through cold backups. In cold backups, the replicas are not constantly updated. Their state is synchronized with the primary service in regular intervals. In the event that the primary service fails, one of the replicas has to be synchronized (perhaps with data from a storage device) before it can service requests. Hence, choice B is incorrect.

Choice D refers to the concept of Warm Backups. In warm backups, only the primary service responds to requests. However all activity is constantly logged.

Periodically these logs may also get flushed to the replicas. When the primary service fails, a replica takes over after its state is synchronized with the logs. Hence, choice D is incorrect.

With active replication, as mentioned before, the state of all replicas is constantly synchronized. Hence, choice E is incorrect.

Incorrect
Marks for this submission: 0/1.

Feedback to Author

14. You have been asked to provide feedback about horizontal and vertical scaling techniques in a Java EE environment. Which of the following are true about Vertical Scalability when compared to Horizontal Scalability? Select two choices.

- A. Achieved by adding additional servers to the current system architecture ✗
- B. Achieved by adding additional capacity (memory, CPU and so on) to the current system architecture ✓
- C. Vertical Scaling is generally more expensive than Horizontal scaling ✗
- D. Decreases manageability ✗
- E. Requires few or no changes to the system architecture ✓
- F. Not supported by J2EE ✗

Choices B and E are correct.

Vertical Scalability, by definition means increasing a system's capacity by adding memory, processors and so on.

Of the two types of Scalability (Vertical and Horizontal), Vertical Scalability is the easier to achieve because it involves few changes to the existing system's architecture. Also adding more CPU or memory to an existing system does not have any impact on reliability or availability because if the system or component fails, in the absence of redundant systems, availability and reliability would suffer. Hence, choices B, E and G are correct.

Choice A is incorrect because it actually describes Horizontal Scalability, which is achieved by adding more servers to a system. Horizontal Scalability is tougher to achieve because the architecture should inherently support a multi-server environment. However once implemented, it has a positive impact on the system's reliability and availability because it provides fault tolerance capabilities.

Choice C is incorrect because generally it is cheaper to add capacity than to add entire new systems.

Choice D is incorrect because adding capacity should have no impact on the manageability of a system.

Choice F is incorrect because Vertical Scalability is easy to achieve. Most J2EE vendors also provide support for the more difficult Horizontal Scalability.

Incorrect
Marks for this submission: 0/1.

Feedback to Author

15. You are designing a Sales Bean to handle the views/screens associated with it. The application will be designed as N-tier model using Java EE environment.
Which is the suitable tier you will place these components?

- A. Client Tier ✗

Choice B is correct.
The sales Bean represents a JSF MangaedBean (which is a controller) in Java EE environment and views/screens represents the pages for that sales module using navigational support configured through faces-config.xml file. N-tier models typically follow a Model2 architecture where each tier has specific responsibilities. The view and most of the controller components are generally placed in the Web Tier. Hence choice B is correct.

Choice A is incorrect because the client tier is not responsible for the view and controller objects.

Choice C is incorrect because there is no such tier as MVC tier.

Choices D, E and F are incorrect because none of them hold the view components (although some controller components may be present in the Business logic tier.).

Incorrect
Marks for this submission: 0/1.

Feedback to Author

16. A scenario is given to you based on the fact that, one of the dropdown list in a particular page takes lot of time to load when the page is getting refreshed. Your Management asked for a good solution and stating that page should not take long time to load dropdown values and reduce the page refresh. Application is running in Java EE environment.

What is your proposed solution to solve this problem ?

- A. Use NON- AJAX based solution to retrieve dropdown values and do a partial refresh ✗
- B. To load it faster make the dropdown list set to request scope and implement partial refresh ✗
- C. Use AJAX based solution to retrieve dropdown values and do a partial refresh ✓
- D. To load it faster make the dropdown list set to session scope and implement partial refresh ✗

Explanation :
Option C is the correct answer.

To solve the given problem, the most suitable answer is AJAX based solution to populate the dropdown and implement the partial refresh. using JSF with AJAX combination, it's easy to configure the dropdown list using rerender and on complete functions to load and partial refresh of the page.

Option A is incorrect. The statement "NON-AJAX based solution" is not clearly defined. Using Java Script you can make the dropdown to load faster and partial refresh is also possible, but need to write lot custom code and difficult to maintain.

Option B is incorrect. The session scope is used to store session data across the page invocations. Part of the problem, it will not help to load faster or slower the dropdown values and there is no feature for partial refresh using session scope attribute.

Option D is incorrect. The request scope is used to store request information based on the each form submit. Part of the problem, it will not help to load faster or slower the dropdown values and there is no feature for partial refresh using request scope attribute.

For more information, please refer -
http://www.oracle.com/technetwork/java/javaee/tutorial-jsp-140089.html

Incorrect
Marks for this submission: 0/1.

Feedback to Author

17. Which of the following statements are true about Thin client when compared to Fat client based solutions from the given options?

- A. Minimal installation conflicts and environment issues. ✓
- B. No issues with cross-browser compatibility and support. ✗
- C. Provides very good client security. ✗
- D. Readily available standard development kits. ✗

Option A and E are correct.

Thin Client (Browser Based applications) :

- Minimal installation conflicts and environment issues.
- Eases deployment and UI changes are immediately available for all the users.
- Little data processing is done on the client side.
- Clients primary responsibility is merely to display data and collect input from the user for posting back to the server.
- The transmission of data to and from the thin client causes higher network bandwidth usage compared to thicker clients.
- It is not as easy to use plug-in's and third party APIs in a browser client than a thick Windows forms application.

Options B, C and D are incorrect. These statements are true for a Fat/thick client based solution.

Thick/Fat Client:

- No issues with cross-browser compatibility and supports.
- No inherent security issues. As the network calls are minimum when compared to thin clients, the security is easy to implement.
- There are readily available standard development kits available compared to build browser based applications.
- Thick clients can make use of the powerful PC hardware sitting on a users desk which can make for a more efficient processing model and can help reduce infrastructure costs.

Incorrect
Marks for this submission: 0/1.

Feedback to Author

---

18. You are designing a web service using JAX-WS features and wants to handle additional processing of inbound and outbound messages.
Select the components available to handle additional processing during runtime using Java EE platform?

- A. Logical Handler and Protocol Handler ✓
- B. Security Handler and Message Handler ✗
- C. Resource Handler and HttpHandler ✗
- D. There is no such feature available in Java EE platform ✗

Option A is correct.
Handlers are message interceptors that can be easily plugged in to the JAX-WS runtime to do additional processing of the inbound and outbound messages. JAX-WS defines two types of handlers, logical handlers and protocol handlers. Protocol handlers are specific to a protocol and may access or change the protocol specific aspects of a message. Logical handlers are protocol-agnostic and cannot change any protocol-specific parts (like headers) of a message. Logical handlers act only on the payload of the message.

Actual logic to implement the request must be in the Service Implementation class but not handler class. JAX-WS programming model provides an application handler facility that enables you to manipulate a message on either an inbound or an outbound flow.

You can add handlers into the JAX-WS runtime environment to perform additional processing of request and response messages. You can use handlers for a variety of purposes such as capturing and logging information and adding security or other information to a message. You can configure the handlers for the server or client.

Options B, C and D are incorrect. There are no such handlers available in Java EE platform part of JAX-WS web service handlers.

For more information, please refer -
http://jax-ws.java.net/articles/handlers_introduction.html/

Incorrect
Marks for this submission: 0/1.

Feedback to Author

---

19. Lead Architect from your company suggested to use Java EE web services due to it's simplicity and easy to expose as a POJO based web services.
Which of the statements are true?

Select two options.

- A. Stateless session beans can be exposed as Web Services ✓
- B. Servlets can be exposed as Web Services ✓
- C. Stateful Session Beans can be exposed as Web Services ✗
- D. Message Driven beans can be exposed as Web Services ✗

Option A and B are correct. In Java EE, Servlet and Stateless session bean can be exposed as a Web Services.

For more information, please refer :
http://docs.oracle.com/javaee/5/tutorial/doc/bnaym.html

Incorrect
Marks for this submission: 0/1.

Feedback to Author

20. Your boss is raving about the new 3-Tier architecture on which your company's sales application will be deployed. He says that this architecture will solve all the existing problems.

Does a 3-Tier architecture have the potential to introduce any new problems?

- a. Fat Clients ✗
- b. Thin Clients ✗
- c. Poor scalability ✗
- d. Poor manageability ✓
- e. Reduced performance ✗
- f. Reduced separation of business logic ✗
- g. Single point of failure ✗

Choice D is correct.

The only problem a 3-Tier architecture could have is the possibility for poor manageability. The separation of tiers creates thin clients and distributes business logic processing. However, because of the distributed nature of the servers, there could be manageability problems. With J2EE solutions however, the possibility for such problems is limited because J2EE tiers and layers have very well defined roles and responsibilities.

Choice A is incorrect because fat clients are a characteristic of 2-Tier architecture where the business logic is implemented on the client side. Thin clients are good. Therefore, choice B is incorrect.

A 3-Tier architecture has excellent scalability, including horizontal scalability, making choice C incorrect.

It is possible to argue that a 3-Tier architecture may not perform as well as a single machine containing your web server, business logic and database. However, as soon as you introduce a significant amount of users, a 3-Tier architecture would out perform a single machine. Therefore, choice E is correct.

Choice F is incorrect because a 3-Tier architecture actually increases the separation of business logic.

Choice G is incorrect because a "single Point of failure" is not a problem that a 3-Tier architecture introduces.

Although it may have a single point of failure, you can avoid such pitfalls easily with careful design. This is much harder to do with 2-Tier architecture and impossible with 1-Tier architecture.

Incorrect
Marks for this submission: 0/1.

Feedback to Author

21. A prospective employer is describing the existing architecture of a solution that is currently in production. He says that it is a 3-tier system with 3 clustered web servers and a server for the Oracle database with the business logic implemented using PL/SQL scripts.

What is true about this system? Select three choices.

- a. This solution has fat clients ✗
- b. This solution has thin clients ✓

Choices B, E and F are correct.

The important aspect of this question is that the business logic has been implemented using PL/SQL stored procedures. This means the business logic is on the same server as the database and therefore this is a 2-tier system, not a 3-tier system as your prospective employer suggests.

There is a very tight coupling between the data store and the business logic, which has a direct affect on the potential scalability of this system. When you think of 2-Tier systems, you would normally associate them with fat clients. However as the business logic has been implemented using PL/SQL scripts, this system will have thin clients, making choice A incorrect.

Choice C is incorrect because the business logic and data store are tightly coupled. This tight coupling will have a direct affect on scalability. You may not be able to horizontally scale this system (adding more machines); vertical scaling might be easier (add more memory and CPUs).

However, even with vertical scaling, you would always run the risk of a network bottleneck making choice D is incorrect as well.

Incorrect
Marks for this submission: 0/1.

Feedback to Author

22. N-tier applications show better performance than 2-tier applications because they are modular in nature and can be scaled easily by tuning components and containers individually.
True or False?

- A. True ✓
- B. False ✗

Choice A is correct as the statement is TRUE.

N-Tier applications (especially those based on Java EE) are designed to be modular in nature. In addition, N-Tier applications are more extensible and more scalable.

The tiers and layers separate roles and responsibilities of each component and container. Hence, components and containers can be individually targeted and scaled as needed. This results in better performance. Hence, choice A is correct.

Incorrect
Marks for this submission: 0/1.

Feedback to Author

23. Moon Microsystems has a web-based solution that was originally built using Servlets. However, in recent months, the IS manager has asked developers to use JSP technology with Java Scriptlets embedded in HTML code. Scott Khosla, the architect, however is insisting that the code be modularized and that EJBs be used instead of Servlets and JSP for business logic processing.

Asking developers to modularize existing code, what software development technique is Scott asking the developers to follow?

- A. Code Break up ✗
- B. Code engineering ✗
- C. Code Tiering ✗
- D. Code Refactoring ✓

Choice D is correct.

Software applications have a tendency to grow in complexity over time. When that happens, they become difficult to extend, maintain and modify. They also become hard to understand. Code factoring is a modularization technique that deals with the separation of responsibilities in code.

Code Refactoring has looser coupling and minimal dependencies, making components and code more reusable. Hence, choice D is correct.

There are no standard concepts like code break up, code engineering or code tiering.
Hence, choices A, B and C are incorrect.

Incorrect

Marks for this submission: 0/1.

Feedback to Author

24. You have given a new task to provide feedback about the third party code. You observed that, the JSP pages are directly calling EJBs and Servlets accessing database.
Which of the following statements are true?
Select two choices.

- A. Easy to maintain code, Difficult to implement extensions ✗
- B. Easy to implement extensions, Difficult to maintain code ✗
- C. Difficult to implement extensions, Difficult to maintain code ✓
- D. Not following separation of concerns ✓

Choices C and D are correct.

Extensibility refers to a system's ability to incorporate new functionality with ease. Writing database access code within Servlets or using JSP for locating Enterprise Java Beans make the code very difficult to maintain. These are poor programming practices and adversely affect the extensibility of a system. Hence, choices C and D are correct.

Modularized code, good design practices using patterns, and encapsulating database access with a DAO all make the code easy to maintain, thus rendering the system easily extensible.

Incorrect
Marks for this submission: 0/1.

Feedback to Author

25. You are providing technical support for a supply chain product that your company (Company X) has sold to another company (Company Y). Whilst working on Company Y's site you need to run a simulation on Company X's network. This will involve securely connecting part of Company X's network to part of Company Y's network.

How should you do this?

- a. Create a DMZ between the two networks. ✗
- b. Create a VPN between the two networks. ✓
- c. Create a secure network connection between the two networks by using a combination of Java sockets and JSSE. ✗
- d. This is not possible, as the corporate firewalls would block this. ✗

Choice B is the correct answer.

Create a VPN between the two networks. Virtual Private Network - VPN - is a solution for securely connecting two networks that are in geographically different locations. A VPN will use a variety of different encryption and authentication techniques to ensure that data confidentiality is maintained.

Choice C would not work. You are not trying to communicate with one Java program running on Company X's network and one running on Company Y's network; instead you need to connect the networks.

A DMZ is the zone between two firewalls.

Incorrect
Marks for this submission: 0/1.

Feedback to Author

26. Which of the following statements are true about thick-client based solution?
Select two choices.

- A. Provides very good client security, persons without a client cannot access system. ✓
- B. The transmission of data to and from the Fat/Thick client causes higher network bandwidth usage ✗
- C. Maintainability of application code is difficult. ✓
- D. UI changes are easily and immediately available to the clients. ✗

Options A and C are true about thick-client based solutions.

Option B and D are false. UI changes are available to clients only after next installation. The transmission of data to and from the Fat client causes lower network bandwidth usage.

Incorrect
Marks for this submission: 0/1.

Feedback to Author

27. You are designing a new web application. You need to display data in tables and dropdowns. Your team which has expert JavaScript developers suggest to build these controls using JavaScript rather than Servlets/JSP.

What are the main disadvantages with this approach?

- a. Developing the code. ✗
- b. Confidentiality of code ✓
- c. It is very hard to deploy. ✗
- d. Javascript should never be used for Java EE applications. Java EE specification does not mention about javascript. ✗

Options C and D are incorrect statements.

Since you have expert JavaScript developers, developing the code may not be a problem. So, option A is incorrect.

When user browses the application, user will be able to view the complete JavaScript code as it is downloaded to the browser. So, option B is correct.

Incorrect
Marks for this submission: 0/1.

Feedback to Author

28. Which of the following scenarios expect an application developer to choose Web Services? Select three choices.

- A. Implementing a solution with guaranteed message delivery. ✗
- B. When there is a need to exchange data between different applications on different platforms. ✓
- C. Asynchronous communication with non Java application ✓
- D. Reduces the dependencies between implementations allowing other consumers to easily use the provided service without major changes ✓

Options B, C and D are correct.

Option A is incorrect as Web Services do not ensure the same.

Incorrect
Marks for this submission: 0/1.

Feedback to Author

29. Which of the following statements are true about StAX over SAX parser?

- A. Pull, Streaming ✓
- B. In Memory Tree ✗
- C. Push, Streaming ✗
- D. Ease of use is high ✓

Options A and D are true about StAX parser.

StAX-enabled clients are generally easier to code than SAX clients. While it can be argued that SAX parsers are

marginally easier to write, StAX parser code can be smaller and the code necessary for the client to interact with the parser simpler.

StAX is a bidirectional API, meaning that it can both read and write XML documents. SAX is read only, so another API is needed if you want to write XML documents. It's a Pull and streaming API and ease of Use is high.

Option B is incorrect. In Memory Tree is provided by DOM parser.

Option C is incorrect. Push and Streaming API is a SAX parser.

Incorrect
Marks for this submission: 0/1.

Feedback to Author

Finish review