

Q C M de préparation pour l'examen de SEG 3502

Leçon 1 : Introduction

1. Qu'est-ce que le génie logiciel ?

- a) La création de systèmes à petite échelle.
- b) La résolution de problèmes clients en équipes, avec des systèmes de haute qualité et respectant des contraintes.
- c) L'évaluation des logiciels avant la mise en production.
- d) Le test des logiciels créés par d'autres équipes.

Réponse : b

2. Quelle est une conséquence potentielle d'une conception inadéquate ?

- a) Une exécution plus rapide des logiciels.
- b) Une meilleure satisfaction client.
- c) Une augmentation des risques et des coûts.
- d) Un cycle de développement plus court.

Réponse : c

3. Selon la définition IEEE, la conception logicielle inclut :

- a) La documentation utilisateur.
- b) La définition de l'architecture, des composants et interfaces du système.
- c) La maintenance des systèmes après leur création.
- d) L'optimisation des performances après la mise en service.

Réponse : b

4. Quel est l'avantage principal de la conception ?

- a) Permet de réduire les exigences.
- b) Simplifie le développement en fournissant des représentations abstraites.
- c) Diminue le besoin de tests finaux.
- d) Accélère la maintenance des systèmes existants.

Réponse : b

5. Pourquoi modéliser dans le génie logiciel ?

- a) Pour capturer des idées et faciliter la validation.
- b) Pour éviter les discussions d'équipe.
- c) Pour éliminer les phases de tests.
- d) Pour augmenter la complexité de la solution.

Réponse : a

6. **Quel type de vue est incluse dans la conception architecturale ?**

- a) Vue des utilisateurs.
- b) Vue de modules, de concurrence et de déploiement.

- c) Vue des prix de développement.
- d) Vue des utilisateurs finaux.

Réponse : b

7. Quelle est une caractéristique de la 'Clean Architecture' ?

- a) Basée uniquement sur la performance.
- b) Structure modulaire permettant l'évolution et des entités indépendantes.
- c) Se concentre uniquement sur la réduction des coûts.
- d) Implémentation rigide des règles métier.

Réponse : b

8. Quelle est la principale limitation de 'Big Design Up Front' (BDUF) ?

- a) Les problèmes apparaissent après la mise en œuvre.
- b) Trop de documentation est produite.
- c) Il n'est pas applicable aux grands projets.
- d) Il réduit les phases de conception.

Réponse : b

9. Quelle est la dette technique dans le contexte du génie logiciel ?

- a) Un coût lié à l'implémentation anticipée des exigences.
- b) Une refactorisation à faire en raison d'une conception limitée initiale.
- c) Une documentation manquante.
- d) Un coût anticipé de tests.

Réponse : b

10. Quel est l'objectif des diagrammes de contexte dans la conception ?

- a) Montrer l'interface utilisateur.
- b) Présenter le système dans son environnement.
- c) Dresser une liste des composants matériels.
- d) Décrire les choix technologiques du projet.

Réponse : b

#####

Leçon 2 : DomainFR

1. Qu'est-ce que le domaine dans le cadre de la conception dirigée par le domaine ?

- a) Un espace de solution uniquement.
- b) Un espace problème pour lequel une solution est créée.
- c) Un modèle d'organisation commerciale.
- d) Une base de données de l'application.

Réponse : b

2. Pourquoi est-il important que le domaine soit reflété dans le code ?

- a) Pour faciliter le déploiement.
- b) Pour améliorer l'apparence de l'interface utilisateur.
- c) Pour simplifier la maintenance et l'évolution.
- d) Pour augmenter les exigences de stockage.

Réponse : c

3. Le Domain-Driven Design (DDD) se concentre principalement sur :

- a) L'interface utilisateur.
- b) La gestion de la base de données.
- c) L'alignement du code avec le domaine du problème.
- d) La documentation du produit.

Réponse : c

4. Qu'est-ce que le langage omniprésent (Ubiquitous Language) ?

- a) Un langage de programmation spécifique.
- b) Un ensemble de termes et structures partagés pour la compréhension commune du domaine.
- c) Un jargon technique utilisé par les développeurs.
- d) Un modèle de données pour le domaine.

Réponse : b

5. Qu'est-ce qu'un sous-domaine ?

- a) Une petite base de données dédiée aux utilisateurs.
- b) Une partie de la fonctionnalité du système classée par importance.
- c) Un type de design d'interface utilisateur.
- d) Un modèle de solution générique pour les projets.

Réponse : b

6. Quel est le but de la conception stratégique dans le DDD ?

- a) Déterminer les partitions du domaine en sous-domaines.
- b) Concevoir des animations pour l'interface utilisateur.
- c) Tester la base de données.
- d) Documenter les spécifications techniques.

Réponse : a

7. Qu'est-ce qu'un contexte borné (Bounded Context) ?

- a) Une limite de protection qui encapsule les modèles et assure leur cohérence.
- b) Un espace de stockage des données.
- c) Un protocole de communication réseau.
- d) Une vue utilisateur de l'interface.

Réponse : a

8. Quel type de relation existe dans un contexte 'Client-Fournisseur' ?

- a) Les deux équipes sont indépendantes et travaillent sans concertation.
- b) Le contexte du client dépend du contexte du fournisseur, et les équipes collaborent pour définir une interface.
- c) Les équipes utilisent des données publiques sans dépendance mutuelle.
- d) Les équipes échangent uniquement des informations de manière unidirectionnelle.

Réponse : b

9. Quelle est la fonction de la 'Couche Anticorruption' dans le DDD ?

- a) Prévenir les conflits entre utilisateurs.
- b) Protéger un contexte consommateur contre la pollution due à l'adaptation à un autre modèle.
- c) Documenter les dépendances du système.
- d) Mettre à jour automatiquement les données partagées.

Réponse : b

10. Quel est le rôle d'une 'Factory' dans la conception tactique ?

- a) Créer des objets du domaine en encapsulant les vérifications nécessaires avant leur création.
- b) Assurer la sécurité des données.
- c) Fournir un modèle de conception pour les interfaces graphiques.
- d) Remplacer les fonctions obsolètes.

Réponse : a

11. À quoi sert un 'Repository' en DDD ?

- a) Stocker temporairement les données des utilisateurs.
- b) Abstraire le stockage et la récupération des concepts de domaine.
- c) Mettre à jour les informations d'identité des utilisateurs.
- d) Réduire les erreurs de saisie utilisateur.

Réponse : b

12. Les 'Événements du Domaine' sont utiles pour :

- a) Notifier des activités significatives dans le domaine.
- b) Remplacer des fonctionnalités inutilisées.
- c) Gérer les comptes utilisateurs.
- d) Rédiger la documentation d'une application.

Réponse : a

13. Quelles sont les étapes clés du DDD ?

- a) Développement d'interfaces graphiques et documentation.
- b) Conception stratégique et conception tactique.
- c) Test de performance et sécurité.
- d) Optimisation des bases de données et déploiement.

Réponse : b

14. Un 'Agrégat' en DDD est :

- a) Un ensemble de tables de bases de données.
- b) Un cluster d'entités et d'objets de valeur contrôlé par une racine.
- c) Un type d'interface utilisateur.
- d) Un fichier de configuration.

Réponse : b

15. Qu'est-ce qu'un 'Service de Domaine' ?

- a) Une interface utilisateur pour les clients.
- b) Un service implémentant la logique métier qui orchestre le comportement des entités.
- c) Un tableau de bord pour la gestion des utilisateurs.
- d) Une base de données de stockage temporaire.

Réponse : b

#####

Leçon 3 : OnlineFunctionDDD

1. Quel est l'objectif principal d'un système d'enchères en ligne ?

- a) Permettre uniquement aux vendeurs de vendre des objets.
- b) Permettre aux acheteurs et vendeurs d'effectuer des transactions commerciales.
- c) Gérer les données personnelles des utilisateurs.
- d) Calculer les coûts des enchères.

Réponse : b

2. Quel rôle joue le 'Diagramme de Contexte' dans la conception du système ?

- a) Définir les couleurs de l'interface utilisateur.
- b) Identifier les systèmes collaboratifs externes et spécifier les interfaces.
- c) Stocker les données des utilisateurs.
- d) Simplifier le code du système.

Réponse : b

3. Quel est le sous-domaine principal du système d'enchères ?

- a) Gestion des utilisateurs
- b) Gestion des enchères
- c) Gestion des paiements
- d) Gestion des notifications

Réponse : b

4. Dans le cas d'utilisation 'Create Auction', quelle information est requise pour créer une enchère ?

- a) Seulement le nom de l'utilisateur
- b) Les informations de l'enchère, y compris l'heure de début, la durée, et, si nécessaire, les informations de carte de crédit

- c) La localisation du vendeur
- d) Le nombre d'enchères actives

Réponse : b

5. Dans le DDD, qu'est-ce qu'une 'Commande d'Application' ?

- a) Une fonctionnalité d'interface utilisateur pour organiser les pages
- b) Une demande d'exécution d'une opération spécifique identifiée à partir des cas d'utilisation
- c) Une liste de toutes les fonctionnalités du système
- d) Un processus de gestion des erreurs dans le code

Réponse : b

6. Quel est le rôle du 'Data Transfer Object' (DTO) dans le système ?

- a) Décorer l'interface utilisateur
- b) Transférer des données entre les frontières de l'application pour éviter la pollution du domaine
- c) Sécuriser les informations utilisateur
- d) Tester la cohérence des données

Réponse : b

7. Quelles sont les préconditions pour créer une enchère ?

- a) Le vendeur doit être déconnecté et sans paiements en attente.
- b) Le vendeur doit être connecté et sans paiements en attente.
- c) L'acheteur doit être connecté et sans enchère active.
- d) Le vendeur doit fournir des informations d'expédition.

Réponse : b

8. Quelle est la post-condition principale lorsque l'encherre est créée ?

- a) Le système envoie un email de confirmation.
- b) L'encherre est ajoutée à la liste des enchères actives du vendeur.
- c) L'encherre est supprimée de la base de données.
- d) Le vendeur est déconnecté du système.

Réponse : b

9. Quel concept dans DDD permet de créer un ensemble d'entités ayant des responsabilités partagées ?

- a) Entité
- b) Agrégat
- c) Service de domaine
- d) Repository

Réponse : b

10. Dans un scénario où une carte de crédit est incluse dans les informations d'encherre, que fait le système ?

- a) Ignore les informations de carte de crédit.

- b) Crée une nouvelle carte de crédit associée au vendeur et à l'enchère.
- c) Crée une nouvelle enchère sans carte de crédit.
- d) Exige que le vendeur fournis un numéro de téléphone.

Réponse : b

11. Qu'est-ce que le 'Behavior Driven Development' (BDD) permet de faire dans le contexte du système d'enchères ?

- a) Créer des interfaces utilisateur.
- b) Tester les comportements de l'application en utilisant un langage compréhensible par l'humain.
- c) Automatiser la base de données.
- d) Améliorer la rapidité d'exécution de l'application.

Réponse : b

12. Quel est l'avantage d'utiliser une façade dans chaque contexte borné ?

- a) Simplifie la structure des bases de données.
- b) Contrôle l'accès au contexte en encapsulant les composants internes.
- c) Améliore la vitesse des transactions.
- d) Permet de supprimer les vérifications de sécurité.

Réponse : b

13. Dans le cadre des niveaux de diagrammes de séquence, quel niveau montre l'interaction de la façade avec les éléments de haut niveau d'un contexte borné ?

- a) Niveau 1
- b) Niveau 2
- c) Niveau 3
- d) Niveau 4

Réponse : b

14. À quoi sert le 'Repository' en DDD ?

- a) Simplifier le code source.
- b) Fournir une abstraction pour la persistance des concepts de domaine.
- c) Gérer les fichiers système.
- d) Décorer l'interface utilisateur.

Réponse : b

15. Dans le processus d'enchère, quel est le rôle du 'Service de Domaine' ?

- a) Fournir une interface utilisateur.
- b) Orchestrer le comportement des entités sans posséder d'identité.
- c) Décorer le profil du vendeur.
- d) Tester les enchères créées.

Réponse : b

#####

Leçon 4 : softwArchIntroFr

1. Selon Robert C. Martin, quel est l'objectif de l'architecture logicielle ?

- a) Minimiser les besoins de stockage de données.
- b) Minimiser les ressources humaines nécessaires pour créer et maintenir le système.
- c) Maximiser les fonctionnalités d'une application.
- d) Augmenter la vitesse de développement.

Réponse : b

2. Quelle est la définition IEEE de l'architecture logicielle ?

- a) Organisation des composants matériels d'un système.
- b) Ensemble des décisions de conception pour minimiser les coûts.
- c) Organisation fondamentale d'un système, ses composants, et les principes de conception.
- d) Documentation des processus de développement logiciel.

Réponse : c

3. Pourquoi l'architecture logicielle est-elle importante ?

- a) Elle se concentre uniquement sur les interfaces utilisateur.
- b) Elle permet la réutilisation et améliore la communication entre les parties prenantes.
- c) Elle se concentre exclusivement sur les coûts de développement.
- d) Elle n'est pas essentielle dans un projet mature.

Réponse : b

4. Quel est le rôle de l'architecte logiciel ?

- a) Écrire tout le code de l'application.
- b) Définir le problème du point de vue de l'ingénierie et gérer la dette technique.
- c) Faire des tests utilisateurs uniquement.
- d) Ne travailler que sur la documentation.

Réponse : b

5. Qu'est-ce qu'un framework d'architecture ?

- a) Une bibliothèque de code spécifique.
- b) Une pratique commune pour créer, interpréter et utiliser des descriptions d'architecture dans un domaine particulier.
- c) Une interface utilisateur.
- d) Un outil de gestion de base de données.

Réponse : b

6. Que décrit la Vue de Modules dans une architecture logicielle ?

- a) Les interactions des utilisateurs avec le système.
- b) La concurrence et la tolérance aux pannes.
- c) La modifiabilité et la testabilité.

- d) Les performances des serveurs uniquement.

Réponse : c

7. La Vue de Déploiement est principalement concernée par :

- a) La structure du code source.
- b) La redondance, la disponibilité et la fiabilité.
- c) La création des interfaces graphiques.
- d) La sécurité des utilisateurs.

Réponse : b

8. Quel est le but du processus de conception d'architecture ?

- a) Structurer, valider et améliorer l'architecture.
- b) Écrire des tests unitaires pour chaque module.
- c) Limiter les coûts de développement.
- d) Créer une interface utilisateur avancée.

Réponse : a

9. Qu'est-ce que le modèle '4+1' dans les frameworks d'architecture ?

- a) Un modèle pour organiser les bases de données.
- b) Un modèle de vue qui prend en compte les modules, la concurrence, le déploiement, et les scénarios d'utilisation.
- c) Un système de gestion de projet.
- d) Une structure de déploiement spécifique à une application.

Réponse : b

10. Quelle est l'étape finale du processus de conception ADD ?

- a) Choix des concepts de conception.
- b) Validation de la conception à l'aide d'une analyse.
- c) Création d'un prototype de l'interface.
- d) Déploiement de l'application.

Réponse : b

11. Quel est l'objectif d'une évaluation basée sur les scénarios ?

- a) Fournir un rapport de performance aux clients.
- b) Vérifier que la conception supporte les attributs de qualité requis.
- c) Concevoir des interfaces utilisateur pour les scénarios.
- d) Tester uniquement les contraintes de sécurité.

Réponse : b

12. Qu'est-ce que la dérive architecturale ?

- a) Une modification de l'architecture pour améliorer la sécurité.
- b) L'introduction de nouvelles décisions de conception sans violer les décisions existantes.
- c) La suppression des composants obsolètes de l'architecture.
- d) La mise à jour de la documentation de l'architecture.

Réponse : b

13. Quel est un exemple de scénario d'attribut de qualité en architecture logicielle ?

- a) Développer une application mobile en 3 mois.
- b) Assurer que le système reprend en moins de 30 secondes après une défaillance.
- c) Créer une interface utilisateur en plusieurs langues.
- d) Documenter tous les modules du système.

Réponse : b

14. Dans quel contexte utilise-t-on un modèle de référence ?

- a) Lorsqu'il est nécessaire de créer des bases de données.
- b) Pour fournir une structure logique globale à un type d'application.
- c) Pour concevoir des interfaces utilisateur spécifiques.
- d) Pour simplifier la documentation du code.

Réponse : b

15. Quels sont les composants d'une évaluation de l'architecture ?

- a) Tests de compatibilité et documentations techniques.
- b) Analyse de scénarios, évaluation quantitative et prototypage.
- c) Création d'interfaces et documentation des processus.
- d) Optimisation des performances et des coûts.

Réponse : b

#####

Leçon 5 : concurrencyFr

1. Qu'est-ce que la vue de concurrence dans l'architecture logicielle ?

- a) Une représentation des interfaces utilisateur.
- b) Une vue des composants et de leur communication pendant l'exécution.
- c) Un modèle de base de données.
- d) Un diagramme de déploiement des composants matériels.

Réponse : b

2. Quel attribut de qualité n'est pas associé à la vue de concurrence ?

- a) Performance
- b) Fiabilité
- c) Disponibilité
- d) Portabilité

Réponse : d

3. Quels sont les éléments principaux dans une vue de concurrence ?

- a) Interfaces utilisateur et serveurs

- b) Composants et connecteurs
- c) Bases de données et modèles de sécurité
- d) Processeurs et mémoires

Réponse : b

4. Quel est un exemple de composant dans une architecture concurrente ?

- a) Processeur
- b) Base de données
- c) Thread
- d) Connexion Wi-Fi

Réponse : c

5. Que permet le middleware dans une architecture distribuée ?

- a) De limiter la sécurité du réseau.
- b) De gérer l'hétérogénéité et d'assurer l'intégration des services.
- c) De restreindre l'accès aux données.
- d) D'améliorer les performances des processeurs.

Réponse : b

6. Quel est un avantage de la communication asynchrone ?

- a) Traçabilité accrue
- b) Couplage fort
- c) Blocage de l'appelant
- d) Permettre à l'appelant de continuer son travail sans attendre la réponse

Réponse : d

7. Qu'est-ce qu'un appel de procédure distante (RPC) ?

- a) Une méthode d'interrogation de base de données.
- b) Une technique pour appeler des procédures distribuées comme si elles étaient locales.
- c) Un protocole de communication pour la sécurité réseau.
- d) Un modèle de déploiement pour les applications mobiles.

Réponse : b

8. Quels sont les composants nécessaires pour une communication RPC ?

- a) Serveur et client
- b) Base de données et interface utilisateur
- c) Stub client et skeleton serveur
- d) Thread et processus

Réponse : c

9. Dans quel cas utiliserait-on une architecture orientée événements ?

- a) Pour une application de traitement par lots.
- b) Pour des systèmes nécessitant une tolérance aux pannes et une scalabilité élevée.
- c) Pour des logiciels en temps réel critique.

d) Pour des applications de bureau simples.

Réponse : b

10. Qu'est-ce que le Middleware Orienté Message (MOM) ?

- a) Une base de données pour la messagerie.
- b) Un service de gestion de sécurité pour les applications distribuées.
- c) Un middleware permettant la communication asynchrone entre producteurs et consommateurs.
- d) Un serveur de stockage de fichiers.

Réponse : c

11. Quelle est la différence entre le modèle point-à-point et le modèle publication/abonnement dans MOM ?

- a) Le modèle point-à-point est plus sécurisé.
- b) Le modèle publication/abonnement permet de diffuser un message à plusieurs consommateurs.
- c) Le modèle point-à-point utilise des requêtes HTTP.
- d) Le modèle publication/abonnement est plus rapide que le modèle point-à-point.

Réponse : b

12. Qu'est-ce que la spécification AMQP ?

- a) Un langage de programmation pour la sécurité des réseaux.
- b) Un protocole pour l'interopérabilité des brokers de messages.
- c) Une interface utilisateur pour les applications mobiles.
- d) Un modèle de développement client/serveur.

Réponse : b

13. Quelle est une caractéristique de l'architecture orientée services (SOA) ?

- a) Un couplage fort entre services.
- b) Utilisation de services indépendants et sans état.
- c) Exécution synchrone exclusivement.
- d) Utilisation de modèles de données spécifiques.

Réponse : b

14. Qu'est-ce que REST ?

- a) Un protocole de communication.
- b) Un style architectural pour les services web.
- c) Une base de données pour stocker des ressources.
- d) Un modèle de cryptographie pour le réseau.

Réponse : b

15. Que permet la méthode GET dans une API REST ?

- a) Supprimer une ressource.
- b) Créer une nouvelle ressource.

- c) Récupérer une représentation de la ressource.
- d) Mettre à jour une ressource existante.

Réponse : c

16. Quelle est une limite courante des architectures REST ?

- a) Utilisation excessive de données.
- b) Compatibilité avec tous les clients.
- c) Simplification de la gestion des erreurs.
- d) Flexibilité des types de réponse.

Réponse : a

17. Quel est l'avantage principal de GraphQL par rapport à REST ?

- a) Requêtes plus rapides
- b) Une API adaptée aux besoins spécifiques du client
- c) Moins de possibilités de personnalisation
- d) Compatibilité avec les bases de données relationnelles uniquement

Réponse : b

18. Quelle est la fonction d'une mutation dans GraphQL ?

- a) Lire les données.
- b) Mettre à jour ou modifier des données.
- c) Supprimer les utilisateurs non authentifiés.
- d) Mettre en cache les données pour une meilleure performance.

Réponse : b