

Questions (4 pts)

Choisir la bonne réponse

1. **Quel type de test vérifie le comportement externe du logiciel sans examiner le code interne ?**
 - a) Test unitaire
 - b) Test fonctionnel
 - c) Test structurel
 - d) Test de performance
2. **Quel type de test vérifie que le logiciel protège les données et résiste aux attaques ?**
 - a) Test de performance
 - b) Test de sécurité
 - c) Test d'intégration
 - d) Test unitaire
3. **Quel type de test est utilisé pour vérifier que toutes les parties du code sont exécutées au moins une fois ?**
 - a) Test fonctionnel
 - b) Test de couverture de code
 - c) Test de performance
 - d) Test d'acceptation
4. **Quel type de test est effectué pour vérifier que le logiciel fonctionne correctement dans son ensemble ?**
 - a) Test unitaire
 - b) Test système
 - c) Test d'intégration
 - d) Test de performance
5. **Qu'est-ce que la qualité ?**
 - a) La conformité aux exigences du client
 - b) Le respect des normes ISO
 - c) La satisfaction des attentes des parties intéressées
 - d) Toutes les réponses ci-dessus
6. **Quelle est la différence entre la qualité et l'assurance qualité ?**
 - a) La qualité est un concept, tandis que l'assurance qualité est un processus
 - b) La qualité concerne le produit, tandis que l'assurance qualité concerne le processus
 - c) L'assurance qualité vise à prévenir les défauts, tandis que la qualité mesure la satisfaction
 - d) Toutes les réponses ci-dessus
7. **Quel est l'objectif principal de l'assurance qualité ?**
 - a) Détecter les défauts dans les produits finis
 - b) Prévenir les défauts en amont des processus
 - c) Réduire les coûts de production
 - d) Améliorer la communication interne
8. **Qu'est-ce qu'un audit qualité ?**
 - a) Une inspection des produits finis
 - b) Une évaluation systématique des processus pour vérifier leur conformité
 - c) Une formation des employés sur les bonnes pratiques
 - d) Une analyse des coûts de production

Exercice 1 (12 pts)

Vous devez évaluer la qualité d'une **application de gestion de bibliothèque**. Cette application permet aux utilisateurs de :

- Ajouter, modifier et supprimer des livres.
- Rechercher des livres par titre, ou auteur .
- Gérer les emprunts et les retours de livres.
 1. soit les critères de qualité choisis (capacité fonctionnelle, fiabilité, potabilité) comment peut on les vérifier pour cette application. (6pts).
 2. Appliquez la méthode **GQM (Goal-Question-Metric) a cette exemple**. (donner les objectifs, les questions et les métriques)(6 pts)

Exercice 2 (4pts)

1. Faire le graphe de contrôle de l'extrait de code suivant (2pts)
if (x > 10) then a = a + 1; endif
if (x % 2 = 0) then b = b + 1; endif (où x % 2 donne le reste de la division entière de x par 2).
2. Donner 3 jeux d'essai satisfaisant les critères de couverture des chemins (2pts)

NB : les objectifs de l'application de l'exercice 1 sont :

Les objectifs représentent ce que vous souhaitez accomplir avec l'application. Voici quelques exemples d'objectifs pour votre système :

- **Objectif 1** : Améliorer l'expérience utilisateur en rendant l'application intuitive et facile à utiliser.
- **Objectif 2** : Assurer la fiabilité des opérations critiques (ajout, modification, suppression de livres, gestion des emprunts).
- **Objectif 3** : Optimiser les performances de recherche pour que les utilisateurs trouvent rapidement les livres.
- **Objectif 4** : Garantir la sécurité des données des utilisateurs et des livres.

Solution

Questions 0.5 pts par questions

1. **b) Test fonctionnel**
2. **b) Test de sécurité**
3. **b) Test de couverture de code**
4. **b) Test système**
5. d) Toutes les réponses ci-dessus
6. d) Toutes les réponses ci-dessus
7. b) Prévenir les défauts en amont des processus
8. b) Une évaluation systématique des processus pour vérifier leur conformité

Exercice 1

1.

1.1 Évaluer la Fonctionnalité

Scénario de test

- Ajoutez un livre avec les informations suivantes : titre = "Le Petit Prince", auteur = "Antoine de Saint-Exupéry", genre = "Littérature".
- Recherchez ce livre par titre, auteur et genre.
- Empruntez le livre, puis retournez-le.

Questions

1. Le livre a-t-il été correctement ajouté ?
2. La recherche fonctionne-t-elle pour tous les critères (titre, auteur, genre) ?
3. L'emprunt et le retour ont-ils été enregistrés correctement ?

1.2 Évaluer la Fiabilité

Scénario de test

- Essayez d'ajouter un livre sans titre ou sans auteur.
- Essayez d'emprunter un livre déjà emprunté.
- Essayez de supprimer un livre qui n'existe pas.

Questions

1. Le logiciel gère-t-il correctement les cas d'erreur (par exemple, affiche-t-il un message d'erreur clair) ?
2. Le logiciel plante-t-il ou se comporte-t-il de manière inattendue ?

1.3 Évaluer la Portabilité

Scénario de test

- Testez le logiciel sur différents systèmes d'exploitation (Windows, macOS, Linux).
- Vérifiez s'il fonctionne bien avec différentes versions de navigateurs (si c'est une application web).

Questions

1. Le logiciel fonctionne-t-il sur toutes les plateformes testées ?
2. Y a-t-il des problèmes de compatibilité ?

2.

2.1 Définir les Objectifs (Goals)

Les objectifs représentent ce que vous souhaitez accomplir avec l'application. Voici quelques exemples d'objectifs pour votre système :

- **Objectif 1** : Améliorer l'expérience utilisateur en rendant l'application intuitive et facile à utiliser.
- **Objectif 2** : Assurer la fiabilité des opérations critiques (ajout, modification, suppression de livres, gestion des emprunts).
- **Objectif 3** : Optimiser les performances de recherche pour que les utilisateurs trouvent rapidement les livres.
- **Objectif 4** : Garantir la sécurité des données des utilisateurs et des livres.

2.2 Poser des Questions (Questions)

Pour chaque objectif, posez des questions qui permettront d'évaluer si l'objectif est atteint.

Objectif 1 : Améliorer l'expérience utilisateur

- Q1.1 : Combien de temps un utilisateur met-il pour accomplir une tâche courante (ajouter un livre, rechercher un livre) ?
- Q1.2 : Combien d'erreurs les utilisateurs commettent-ils lors de l'utilisation de l'application ?
- Q1.3 : Quel est le niveau de satisfaction des utilisateurs après avoir utilisé l'application ?

Objectif 2 : Assurer la fiabilité des opérations critiques

- Q2.1 : Quel est le taux de succès des opérations d'ajout, de modification et de suppression de livres ?
- Q2.2 : Combien de fois des erreurs critiques surviennent-elles lors de la gestion des emprunts et des retours ?
- Q2.3 : Quel est le temps de réponse moyen pour les opérations critiques ?

Objectif 3 : Optimiser les performances de recherche

- Q3.1 : Quel est le temps moyen pour effectuer une recherche de livre par titre, auteur ou genre ?
- Q3.2 : Combien de recherches échouent ou retournent des résultats incorrects ?
- Q3.3 : Quel est le taux de satisfaction des utilisateurs concernant les résultats de recherche ?

Objectif 4 : Garantir la sécurité des données

- Q4.1 : Combien de tentatives d'accès non autorisé à l'application ont été détectées ?
- Q4.2 : Quel est le temps moyen pour détecter et résoudre une faille de sécurité ?
- Q4.3 : Les données des utilisateurs et des livres sont-elles correctement chiffrées et sauvegardées ?

2.3 Identifier les Métriques (Metrics)

Pour chaque question, définissez des métriques mesurables.

Objectif 1 : Améliorer l'expérience utilisateur

- M1.1 : Temps moyen pour accomplir une tâche (en secondes).
- M1.2 : Nombre d'erreurs commises par les utilisateurs par session.
- M1.3 : Score de satisfaction utilisateur (sur une échelle de 1 à 5).

Objectif 2 : Assurer la fiabilité des opérations critiques

- M2.1 : Taux de succès des opérations critiques (en pourcentage).
- M2.2 : Nombre d'erreurs critiques par jour/semaine.
- M2.3 : Temps de réponse moyen pour les opérations critiques (en millisecondes).

Objectif 3 : Optimiser les performances de recherche

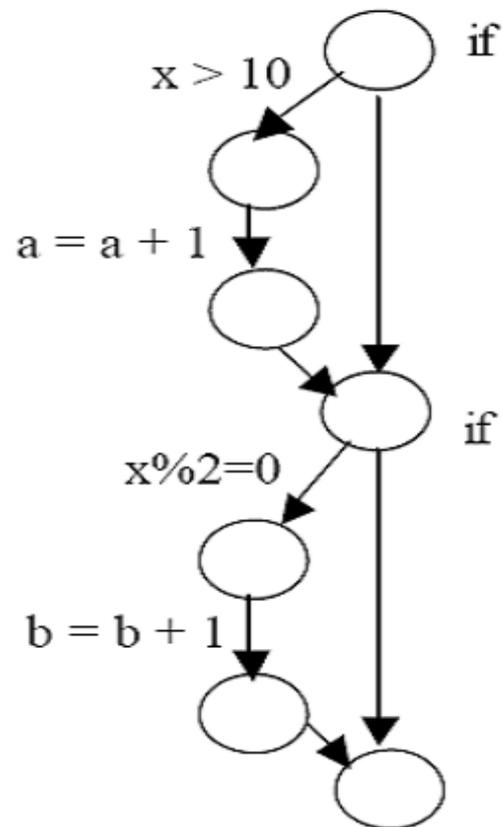
- M3.1 : Temps moyen de réponse pour une recherche (en millisecondes).
- M3.2 : Taux d'échec des recherches (en pourcentage).
- M3.3 : Score de satisfaction des utilisateurs concernant les résultats de recherche (sur une échelle de 1 à 5).

Objectif 4 : Garantir la sécurité des données

- M4.1 : Nombre de tentatives d'accès non autorisé détectées par mois.
- M4.2 : Temps moyen de détection et de résolution des failles de sécurité (en heures).
- M4.3 : Pourcentage de données chiffrées et sauvegardées correctement.

Exercice 2

Graphe de contrôle :



Couverture des chemins

On donne un nombre > 10 paire

On donne un nombre >10 impaire

On donne un nombre <10 impaire

On donne un nombre <10 paire

