Test Technique

1- Les designs patterns : MVC, Singleton, Injection des dépendances etc

Un design pattern est un moyen de conception répondant à un problème récurrent.

Par exemple :

Le pattern singleton permet de pouvoir instancier une classe une seule et unique fois.

Le pattern injection de dépendances a pour but de rendre le plus indépendantes possible les classes.

Le pattern MVC est un pattern avec lequel le code est découpé en trois parties logiques qui communiquent entre elles : Le modèle (données),La vue (fenêtre), Le contrôleur qui lie les deux.

L'implémentation du pattern observer permet au modèle de tenir informés ses observateurs.

L'implémentation du pattern strategy permet à la vue d'avoir des contrôles différents.

Utiliser ce pattern permet de découpler trois acteurs d'une application, ce qui permet plus de souplesse et une maintenance plus aisée du code.

2- La modélisation UML : digramme des classes, digramme des use-case, les relations (composition/ agrégation)

UML est un langage visuel constitué d’un ensemble de schémas, appelés des diagrammes, qui donnent chacun une vision différente du projet à traiter. UML nous fournit donc des diagrammes pour représenter le logiciel à développer : son fonctionnement, sa mise en route, les actions susceptibles d’être effectuées par le logiciel.

3- La méthodologie Agile : Scrum , les fondements du Scrum , la différence entre le cycle en V et Scrum

Agile représente un ensemble de “méthodes et pratiques basées sur les valeurs et les principes du Manifeste Agile”, qui repose entre autre sur la collaboration, l’autonomie et des équipes pluri-disciplinaires.

Scrum est un framework qui est utilisé pour implémenter la méthode Agile de développement et de gestion de projet.

L’avantage principal du cycle en V est qu’elle présente des gains d’échelle (planning et/ou charge) en traitant d’un bloc l’ensemble du périmètre du projet mais comme inconvénients y a le manque de communications et de souplesse Quand au Scrum le processus itératif est vertueux et permet d’avoir un produit très proche des besoins du client tenant compte des évolutions de ces derniers. Il facilite également la communication entre l’ensemble des acteurs du projet et responsabilise toute l’équipe.Comme inconvénients y a le manque de documentation.

4-Qu'est-ce que DevOps?

Le DevOps est un mouvement, une approche qui privilégie l’étroite collaboration entre les équipes de développement (dev) et d’exploitation (ops) pour toute solution TI. Par ce rapprochement, le but visé est d’améliorer la qualité du travail et la relation entre ces deux équipes qui ont chacun leur vision pour atteindre la satisfaction du client – livrer du nouveau code sur demande et maintenir la disponibilité des services.

5- Qu’est-ce que le TDD(test driven developpement) et le BDD(behavior driven developpement ) ?

Le Test-Driven Development (TDD est une méthode de développement de logiciel qui consiste à écrire chaque test avant d'écrire le code source d'un logiciel, de façon itérative

6-C’est quoi le BDD (behavor driven developpement ?

Le Behavior-Driven Development, ou BDD, est une méthode de développement logicielle dérivée du Test-Driven Development – TDD. Elle incite à la collaboration des différentes parties prenantes au projet logiciel, équipes de développement, qualification et management en instaurant que le comportement d’une fonctionnalité sera décrit par des phrases basées sur un canevas composé de mots clés du langage courant.

7-Les règles du SOLID

**Responsabilité unique** (Single responsibility principle)

une classe, une fonction ou une méthode doit avoir une et une seule responsabilité

**Ouvert/fermé** (Open/closed principle)

une entité applicative (class, fonction, module ...) doit être ouverte à l'extension, mais fermée à la modification

**Substitution de Liskov** (Liskov substitution principle)

une instance de type T doit pouvoir être remplacée par une instance de type G, tel que G sous-type de T, sans que cela ne modifie la cohérence du programme

**Ségrégation des interfaces** (Interface segregation principle)

préférer plusieurs interfaces spécifiques pour chaque client plutôt qu'une seule interface générale

**Inversion des dépendances** (Dependency inversion principle)

il faut dépendre des abstractions, pas des implémentations

8-Techniques d’optimisation des requêtes SQL

\*Filtrer les données directement via le filtre WHERE et/ou LIMIT, en évitant que cela ne soit fait par l’application.

\*Eviter d’utiliser des fonctions dans les clauses de recherche, telle que WHERE.

\*Eviter les lectures via “SELECT \*” en privilégiant plutôt de lister uniquement les colonnes qui seront exploitées.

\*Eviter d’utiliser le wildcard “%” au début d’une recherche LIKE.

Exemple : une requête “SELECT \* FROM table WHERE title LIKE ‘%abc’ est consommateur en performance

\*Utiliser des requêtes préparées ou procédures stockées (stored procedure) pour mettre en cache certaines requêtes ou créer un script directement du côté du système de gestion de base de données.

9-Techniques d’optimisation du temps d’affichage des pages Web

Compression de fichiers exemple (Gzip)

Optimisation des images

Minification de code

La mise en cache

La minification et le regroupement(Bundling) du CSS et JavaScript et CSS.

10-Comment vous optimisez votre code ?

Conception

Garbage Collector

11-Comment vous testez votre projet ? Quels sont les différents types de test ?.

Avec les test unitaires.

Types de test : Les tests d'intégration, Les tests fonctionnels, Les tests de non-régression, Les tests IHM, Les tests de configuration, Les tests de performance, Les tests d'installation.

12-En quoi consiste l’intégration continue ?

L’Intégration Continue (continuous integration) consiste à intégrer les changements apportés au code informatique d’un projet logiciel de façon continuelle, afin de détecter et de corriger immédiatement les éventuelles erreurs.

13-Comment vous gérer les exceptions ?

Lancer des exceptions : throw new Exception

Intercepter des exceptions : try { } catch {}

14-Docker et VM , définition et différence

Docker est une technologie basée sur des conteneurs et les conteneurs ne sont qu'un espace utilisateur du système d'exploitation. Au niveau inférieur, un conteneur est simplement un ensemble de processus isolés du reste du système, exécutés à partir d'une image distincte fournissant tous les fichiers nécessaires à la prise en charge des processus. Il est construit pour exécuter des applications. Dans Docker, les conteneurs en cours d'exécution partagent le noyau du système d'exploitation hôte.

En revanche, une machine virtuelle n’est pas basée sur la technologie de conteneur. Ils se composent de l’espace utilisateur et de l’espace noyau d’un système d’exploitation. Sous les ordinateurs virtuels, le matériel du serveur est virtualisé. Chaque machine virtuelle dispose d'un système d'exploitation (OS) et d'applications. Il partage les ressources matérielles de l'hôte.

15-C’est quoi l’injection SQL , donnez un exemple

L'injection SQL est une méthode d'attaque très connue. Il consiste à modifier une requête SQL en injectant des morceaux de code non filtrés, généralement par le biais d'un formulaire.

Exemple :

Soit un formulaire d’authentification ou on saisit le login et le mot de passe , avec une requete "SELECT \* FROM utilisateurs WHERE login='$login' AND password='$password'" si l’utilisateur décide de mettre en login "personne' #" et laisser le password vide. Notre requête deviendrait donc : "SELECT \* FROM utilisateurs WHERE login='jean' # AND password=''" , grâce à cette injection l'utilisateur va pouvoir se connecter à n'importe quel compte sans connaitre son mot de passe

16-Qu’est-ce qu’un design pattern ?

Un design pattern est un moyen de conception répondant à un problème récurrent.

17-Quels design patterns connaissez-vous et avez-vous déjà utilisé ?

Mvc , Singleton , Factory , Repository ,Unit of Work.

18-Avez-vous déjà utilisé l’injection de dépendance ? Si oui quels frameworks avez-vous utilisé ?

Non

19-Que signifie MVC ?

Le pattern MVC est un pattern avec lequel le code est découpé en trois parties logiques qui communiquent entre elles : Le modèle (données),La vue (fenêtre), Le contrôleur qui lie les deux.

20-Quels sont les diagrammes UML que vous utilisez ?

UsesCase , Diagramme de classes , diagramme de séquences , diagramme d’activité.

21-Quel framework javascript utilisez-vous ? (2)

Bootstrap , Angular

22-Qu’est-ce que Ajax (1) ?

Il ne s'agit pas d'une technologie en soi mais plutôt d'un ensemble de technologies qui permettent la mise à jour du contenu d'une page Web d'une manière rapide et sans chargement complet de celle ci.

23-Qu’est-ce que SPA, Citez plusieurs framework SPA (2)

SPA signifie Single Page Application,c’est une application web accessible via une page web unique. Le but est d’éviter le chargement d’une nouvelle page à chaque action demandée, et de fluidifier ainsi l’expérience utilisateur. Fraworks :AngularJS , reactJS , EmberJS

24-Ou peut-t-on cacher les données ?

25-C’est quoi le bundle (regroupement) il sert a quoi ?

Le regroupement est une technique qui permet de combiner plusieurs fichiers JavaScript et CSS en un seul fichier, pour réduire le nombre d'éléments à télécharger, et de ce fait le nombre de requêtes HTTP du navigateur.Il sert à optimiser le temps de chargement des pages.

26-C’est quoi un Closure en JavaScript ?

On appelle closure une fonction interne qui va pouvoir continuer d’accéder à des variables définies dans la fonction externe à laquelle elle appartient même après que cette fonction externe ait été exécutée.

27-C’est quoi les Jquery validator ?

jQuery Form Validator est un plugin jQuery riche en fonctionnalités et multilingue qui facilite la validation des entrées utilisateur tout en maintenant votre balise HTML propre du code javascript.

28-Comment vous optimisez le temps d’affichage d’une page web ?

Compression de fichiers exemple (Gzip)

Optimisation des images

Minification du code.

La mise en du cache.

La minification et le regroupement(Bundling) du CSS et JavaScript et CSS.

29-Quelle est la particularité de GIT ?

Git ne repose pas sur un serveur centralisé, mais il utilise un système de connexion pair à pair.

30-C’est quoi la différence entre SOAP et Rest ?

L’une des principales différences entre SOAP et REST est que SOAP est un protocole et que REST est un modèle d’architecture.

Une autre différence essentielle entre les protocoles SOAP et REST réside dans le fait que les requêtes envoyées via REST ont tendance à être beaucoup plus légères que SOAP. Pour cette raison, les applications ne nécessitent pas beaucoup de bande passante pour utiliser les services Web REST sur SOAP.

REST peut disposer d’une solution de mise en cache qui permettra de sauvegarder les réponses reçues du serveur. Dans ce cas, le client n’a pas besoin de faire la même requête au serveur et peut utiliser le cache pour obtenir la réponse souhaitée.

31-C’est quoi WebServise Restful ?

RESTful Web Service sont basés sur la structure REST qui est basée sur le protocole http.

RESTful Web Service utilisent les actions http (Get , Post ,Delete ..) pour effectuer des opérations sur des ressources.

32-La différence entre le cycle en V et la méthodologie agile ?

Adopter une méthode agile comme scrum plutôt qu’une gestion de projet waterfall va apporter de la souplesse, qui est notamment nécessaire pour prendre en compte la complexité croissante des applications demandées. Les tests en continu et les objectifs courts vont améliorer la qualité finale et augmenter la productivité de l’équipe. Au final, le projet prendra moins de temps, et son budget sera mieux contrôlé.

33-Le code review consiste en quoi ?

Il consiste à convoquer consciemment et systématiquement les autres programmeurs pour vérifier les erreurs de code. Il a été démontré à maintes reprises qu’il accélérait et rationalisait le processus de développement logiciel .

34-C’est quoi la TMA ?

La tierce maintenance applicative est la maintenance appliquée à un logiciel et assurée par un prestataire externe dans le domaine des technologies de l'information et de la communication.

35-C’est quoi prototype en JavaScript ?

Les prototypes sont un mécanisme au sein de JavaScript qui permettent aux objets JavaScript d'hériter des propriétés d'autres objets. Tous les objets JavaScript héritent des propriétés et des méthodes d'un prototype.

36-Comment vous définissez le format d’envoie des données d’un WebService Json ou Xml ?

JSON est un format léger d’échange de données indépendant du langage. Il est basé sur JavaScript et est facile à comprendre et à générer.

Le langage XML est un langage de balisage qui définit un ensemble de règles de codage des documents dans un format lisible

37-SQL : Donnez la définition de chaque mot : trigger, vue, table, index, clé primaire

**Un trigger** (ou déclencheur) est un bloc PL/SQL associé à une table permettant de déclencher une action avant ou après un INSERT, UPDATE ou DELETE sur cette table.

**Une vue** dans une base de données est une synthèse d'une requête d'interrogation de la base. On peut la voir comme une table virtuelle, définie par une requête.

**une table** est un ensemble de données organisées sous forme d'un tableau où les colonnes correspondent à des catégories d'information

**Un index** est une structure qui reprend la liste ordonnée des valeurs auxquelles il se rapporte.

**une clé primaire** est la donnée qui permet d'identifier de manière unique un enregistrement dans une table

38-Erreur http : 500, 400, 300, 200 donnez la signification de chaque plage et des exemples

200 : succès de la requête

301 et 302 : redirection, respectivement permanente et temporaire

401 : utilisateur non authentifié

403 : accès refusé

404 : page non trouvée

500 et 503 : erreur serveur

**Remarque : Lors des entretiens clients, vous auriez des questions sur l’orienté objet, l’architecture logicielle, la sécurité des application et l’optimisation des performances.**