# GLOSSAIRE

I. Général

--------------------------------------------------------------------------------------------------------------------------------

1. Quel est l’environnement à installer pour exécuter un script PHP ? Citer 2 exemples de logiciels permettant ce contexte

- Il faut installer un environnement côté serveur. 2exemples : Laragon que l'on utilise en cours, et XAMPP

2. Qu’est-ce qu’un algorithme

- Un algorithme est une suite finie et non ambiguë d'instructions et d’opérations permettant de résoudre une classe de problèmes.

3. Qu’est-ce qu’une variable ? Par quel symbole est préfixée une variable en PHP ?

- En informatique, une variable est un symbole (habituellement un nom) qui renvoie à une position de mémoire dont le contenu peut prendre successivement différentes valeurs pendant l'exécution d'un programme ; mathématiques et en logique, une variable marque un rôle dans une formule, un prédicat ou un algorithme.

- Une variable en PHP est préfixée par un symbole dollar $.

- Le nom est sensible à la casse. Les noms de variables suivent les mêmes règles de nommage que les autres entités PHP.

- Un nom de variable valide doit commencer par une lettre ou un souligné (\_), suivi de lettres, chiffres ou soulignés .

- Examples: $var = 'Bob'; $Var = 'Joe';.

4. Qu’est-ce que la portée d’une variable ?

- La portée d'une variable est la zone de code où elle a été déclarée et peut être utilisée. En programmation, une variable peut avoir une portée locale ou une portée globale. La portée variable générale s'applique à tous les blocs de code, y compris les classes. En JS: lle sera accessible dans les blocs intérieurs. Pour contrôler l'accès aux variables et aux fonctions, il est possible d'utiliser des niveaux de contrôle : public, protected, package-protected et private.

5. Qu’est-ce qu’une constante ? Quelle est la différence avec une variable ?

- Une constante est une variable dont la valeur ne peut pas être modifiée.

- Tenter de modifier la valeur d’une constante provoque une erreur.

6. Qu’est-ce qu’une superglobale, combien en existent-ils et donner un exemple d’utilisation

- Une superglobale PHP est une variable PREDEFINIE, disponible partout et tout le temps dans un script.

Il en existe 8 :

- $\_GET

- $\_POST

- $\_REQUEST

- $\_COOKIE

- $\_SESSION

- $\_SERVER

- $\_FILES

- $\_ENV

7. Quels sont les différents types (primitifs) que l’on peut associer à une variable en PHP ?

Les citer et en donner des exemples (ne pas oublier le type d’une variable sans valeur)

- $string = "Bonjour";

- $number = 13;

- $float = 12.34;

- $boolean = true;

- $array = ["azerty", 2, 45.4, "uiop"]

- $undefined = undefined;

- $dateTime = new Date('1995-07-03')

8. Existe-t-il plusieurs types de tableaux en PHP, si oui lesquels ?

- Oui, il existe différents types de tableaux.

- Il existe les tableaux simples de variables tel que :

array(1, 2, 3, "a", "b", "c")

- Et il existe les tableaux "associatifs" avec la forme clé => valeur, où l'on attribue à chaque clé une valeur :

array("nom" => "Louërat", "prenom" => "floris", "age" => 28)

9. Quelles sont les différentes structures de contrôles qu’il existe en algorithmie ?

Donner un exemple pour chacune d’entre elles.

-En algorithmie, les structures de contrôle principales sont :

Séquence : Exécution d’instructions l’une après l’autre.

x = 1;

y = 2;

z = x+y;

Sélection : Structures conditionnelles comme if, else, switch, etc. qui permettent d’exécuter des instructions selon des conditions.

if(x<15) {

console.log(x)

} else {

null

}

Itération : Boucles comme for, while, et do-while qui répètent des instructions jusqu’à ce qu’une condition soit remplie.

while (x<15) {

console.log(x); x++

}

Les boucles for pour indiquer un nombre prédéfini de répétitions de la boucle :

for(i=0, i<5, i++){

console.log(i)

}

Et enfin les forEach qui permettent d'accéder a un ensemble d'éléments ou de variables et de faire une action sur chacune d'elles :

table = [1,2,3,4,5];

table.forEach(element, () => {

console.log(element)

}

10. Quelle est la fonction PHP permettant de demander la longueur d’une chaîne de caractères ?

- La fonction est : "strlen(string $string): int", elle retourne la taille de la chaîne string en int

ou' strlen(phrase)

11. Qu’est-ce qu’une session ? Quelle fonction permet de démarrer une session en PHP ?

Donner un exemple d’utilisation en PHP.

- Une session est démarrée par session\_start() en PHP. Elle peut être utilisée pour recueillir des informations sur les actions de l'utilisateur pour lui renvoyer les données personnalisée à travers la superglobale $\_SESSION, comme des notifications après

une action par exemple:

$\_SESSION['user'] = 'Bob';

echo 'Bonjour, '.$\_SESSION['user'];

12. Qu’est-ce qu’un cookie ? Donner un exemple d’utilisation en PHP.

- Un cookie appelé aussi témoin de connexion ou témoin, est une petite quantité de données échangées entre un serveur HTTP et un client HTTP, et qui permet de créer une session avec état lors de la visite d'un site Web.

Un cookiesert à stocker des informationsqui peuvent être réutilisées plus tard par le site web. Un exemple d'utilisation :

setcookie("user", "Bob", time() + 86400);

créait un cookie qui avec pour la clé 'user', la valeur 'Bob' jusqu'à (time() + 86400) = 24h.

On accède au cookie avec la superglobale : $\_COOKIE["user"];

13. Quelle est la différence entre les instructions « require » et « include » en PHP ?

- La différence entre les deux se passe seulement lorsque le fichier en question est introuvable.

" require " provoque une erreur bloquante (fatal error) (E\_COMPILE\_ERROR) et arrêtera le script,tandis que " include " provoque un avertissement (warning) (E\_WARNING) mais le script continue de fonctionner (utile si ce fichier est optionnel).

* "require" et "include" sont toutes les 2 utilisées pour inclure le contenur d'un autre fichier.

14. Comment effectuer une redirection en PHP ?

- Avec la fonction header() de PHP on peut renvoyer l'internaute d'une page à l'autre en PHP.

header('Location:monchemin')

15. Définir la partie « front-end » et « back-end » d’une application

- La partie " front-end " (côté client, ou' on se consentre sur le UX-UI) d'une application fait référence à ce que voient les utilisatuers :texte, images, boutons... et avec lesquels les utilisateurs peuvent intéragir (menus de navigation...),

- on utilise le HTML, le CSS ou encore le JS pour contrôler la structure, le visuel ou encore le responsive d'une page internet.

- La partie " back-end " (côté serveur, la partie invisible) d'une application fait référence aux fonctionnalités générales qu'elle comporte. Elle traite les demandes dues aux interactions des users (remplir un champ de texte avec des données personnelles), des requêtes envoyées au serveur, la base de donnée.

16. Définir le contrôle de version ? Qu’est-ce que Git ?

- Le contrôle de version, également appelé contrôle de source, désigne la pratique consistant à suivre et à gérer les changements apportés au code d'un logiciel.

- Git est un système de contrôle de version, un contrôleur de version "décentralisé". Il s’agit d’un outil de développement qui aide une équipe de développeurs à gérer les changements apportés au code source au fil du temps.

17. Qu’est-ce qu’un CMS ? Citer au moins 2 exemples.

- C'est un Système de Gestion de Contenu (Content Management System), cela permet de créer, modifier facilement un site internet (blog, site de vente en ligne...).

Exemples : WordPress, TYPO3, Drupal, PrestaShop...

--------------------------------------------------------------------------------------------------------------------------------------

II. Front-end

--------------------------------------------------------------------------------------------------------------------------------------

1. Définir HTML

- Le HyperText Markup Language (langage de balises pour l'hypertexte) est le langage de balisage conçu pour représenter une page internet et sa structure.

2. Définir CSS

- Les Cascading Style Sheets (feuilles de style en cascade) forment un langage qui décrit la présentation des documents HTML et XML, elles permettent de donner du visuel et de la personnalité à une page internet afin de la rendre attractive. Le CSS est également utilisé afin de rendre une page web responsive.

3. Définir Javascript

- Javascript, aussi appelé JS, est le langage qui a rendu possible le « dynamisme côté client » de toutes les applications web.

- Depuis, grâce à Node.js, il peut également être utilisé côté serveur, ce qui en fait le seul langage pouvant être utilisé en frontend et en backend. ( à compléter...)

- JavaScript est un langage de programmation permettant d'ajouter des scripts pour rendre l'application plus responsive.

4. Définir JSON. Dans quel contexte ce format est-il utilisé ?

- JSON est une structure utilisée pour conserver des données, utilisé donc en base de données.

JSON : JavaScript Object Notation, est un format de données textuel dérivé de la notation des objets du langage JavaScript. C'est basé sur des paires nom/valeur et des listes ordonnées.

5. Peut-on interpréter du Javascript côté serveur ? Si oui, comment ?

- Je pense que oui, en envoyant un exécutable en PHP via le JavaScript

6. Qu’est-ce qu’un sélecteur CSS ?

- Les sélecteurs CSS définissent les éléments sur lesquelles s'applique un ensemble de règles CSS, voici les 3 plus utilisés :

Les sélecteurs de types : input {};

Les sélecteurs de classes : .classe {};

Les sélecteurs d'identifiants : #id {} (l'ID est unique).

Ils permettant d'identifier un élément dans le HTML pour y attribuer des styles.

7. Quelle balise HTML permet de créer un lien hypertexte ?

- La balise "<a href="> description du lien </ a>.

8. Qu’est-ce qu’une requête AJAX ?

- C'est utilisé pour la communication asynchrone : envoyer des requêtes vers le serveur et déclencher des opérations lors de la réception de réponses de celui-ci.

9. Quel sélecteur CSS permet de sélectionner tous les éléments d’une classe spécifique ?

D’un identifiant spécifique ?

- "." pour sélectionner une classe et "#" pour un identifiant

(C'est le sélecteur de classe : ".nomclasse {}" ; ou le sélecteur d'identifiant : "#valeurid {}").

10. Définir le responsive design.

- Le responsive design s'adapte aux différents types et tailles d'appareils pour que l'utilisateur ait une meilleure expérience. C'est un ensemble de pratique qui permet d'offrir une consultation confortable sur des écrans de tailles très différentes.

11. Qu’est-ce que le templating ?

- C'est l'utilisation de modèle ou de patron qui permettent une organisation du code plus ordonnée afin d'être mieux organisé et de gagner en efficience.

12. Qu’est-ce qu’une fonction anonyme en Javascript ?

- Une fonction anonyme en JavaScript est une fonction qui est définie sans nom.

- Ces fonctions sont souvent utilisées pour créer des fonctions qui ne seront pas réutilisées ailleurs ou pour passer une fonction en tant qu’argument à une autre fonction.

- Voici un exemple simple :

setTimeout(function() {

console.log('Ceci est une fonction anonyme');

}, 1000);

- Dans cet exemple, setTimeout utilise une fonction anonyme qui affiche un message dans la console après un délai d’une seconde. La fonction est exécutée une fois et n’a pas besoin d’être nommée car elle n’est pas utilisée ailleurs.

- Autre exemple forEach

- forEach prend en paramètre une fonction dite "callback (ou fonction de rappel) qui sera exécuté à chaque tour de boucle.

- Dans cette fonction anonyme, chaque "li" sera représenté par la variable passée en argument "element" (qui peut etre nommée autrement si besoin)

- let listElements = list.querySelectorAll("li")

// pour modifier des éléments du DOM (la propriété style de chaque élément en effectuant un boucle forEach( sur la NodeList))

listElements.forEach(function(element){

element.style.color = "red"

})

13. Quelle méthode JavaScript est utilisée pour ajouter un élément à la fin d'un tableau ?

- Pour ajouter un élément à la fin du tableau indexé on utilise la méthode .push(newElement) de la classe Array par example:

tableauIndexe.push(2);

- Nous ne pouvons pas ajouter seulement un élément au tableau associatif, on peut ajouter une/des pairs (duos) clef/valeur, par example:

tabAssoc.cle1 = "Valeur1";

14. Qu’est-ce qu’un « media query » ?

- C'est un moyen utilisé en CSS afin d'attribuer des styles en fonction du type ou de la taille de l'appareil de l'utilisateur. Les requêtes média (media queries) permettent de modifier l'apparence d'un site ou d'une application en fonction du type d'appareil (impression, écran, ...) et de ses caractéristiques (la résolution d'écran ou la largeur de la zone d'affichage (viewport) par exemple). Les media queries permettent d’appliquer certains styles de manière conditionnelle avec le CSS grâce aux règles @media et @import.

15. Qu’est-ce qu’un pseudo élément en CSS ?

- Mot-clé ajouté à un sélecteur qui permet de mettre en forme certaines parties de l'élément ciblé par la règle. Ainsi, le pseudo-élément "::first-line" permettra de ne cibler que la première ligne d'un élément visé par le sélecteur :

"p ::first-line { color: blue; text-transform: uppercase; }".

16. Qu’est-ce que Bootstrap ? Donner d’autres exemples équivalent.

- C'est une collection d'outils utiles à la création du design de site et d'applications web. Il contient des codes HTML et CSS, des formulaires, boutons... Il existe de nombreux équivalent, notamment :

Tailwind CSS, Bulma, Materialize, Foundation by Zurb, Skeleton...

17. Quand un formulaire HTML est créé, quelles sont les 2 méthodes qui peuvent lui être associées ? Donner la différence entre ces 2 méthodes.

* L'attribut "method" définit la méthode HTTP qui sera utilisée pour envoyer les données au serveur, il peut prendre les valeurs suivantes :

POST : la méthode POST, utilisée pour envoyer des données au serveur ; les paramètres (données saisies par l'utilisateur) sont passés dans la requête elle-même.

GET : la méthode GET, utilisée pour récupérer les données, qui après passent par l'URL.

--------------------------------------------------------------------------------------------------------------------------------

III. UX UI

--------------------------------------------------------------------------------------------------------------------------------

1. Quelle est la différence entre UX Design et UI Design ?

- UI (User Interface) se concentre sur l'interface, l'aspect visuel pour l'utilisateur.

- UX (User eXperience) se concentre sur l'experience, les interactions de l'utilisateur avec les diffrents objets. L'UX Design : User eXperience Design (UXD) est une méthode de conception centré sur l'user afin de lui proposer **une expérience unique** où l'ergonomie et la navigation sont adaptées à une cible définie. Les 3 pilliers de l'UX sont : contenter l'utilisateur, écouter l'entreprise et maîtriser la technologie.

L'UI Design **: User Interface** (UID) est l'étape de conception de l'interface utilisateur : l'expérience utilisateur (UX) est liée au design graphique de l'interface (UI). Le but est de créer une interface agréable et pratique, facile à prendre en main. L'UI s'attaque plus particulièrement aux éléments perceptibles : éléments graphiques, boutons, navigation, typographie...

2. Qu’est-ce qu’un wireframe ?

- C'est une représentation visuelle d'une maquette dite, "fil de fer", en niveau de gris. Elle représente la structure et la fonctionnalité d'une seule page web ou d'un écran d'application mobile.

3. Qu’est-ce qu’un prototype ?

- Un prototype en UX permet de simuler le fonctionnement de notre dispositif digital afin de le tester, avant de le produire.

4. Qu’est-ce que la hiérarchie visuelle en UI Design ?

- C'est un principe de design qui fait référence à la manière dont les éléments sont disposés dans un design afin que le visuel soit assimilé correctement.

5. Qu’est-ce que l’accessibilité en UX Design ?

L'accessibilité permet à toutes les personnes, y compris celles ayant des limitations physiques, sensorielles ou cognitives, d'utiliser des produits numériques sans obstacles. Cela promeut l'équité et l'inclusion, assurant que personne ne soit exclu de l'utilisation de vos services.

6. Qu’est-ce qu’une grille de mise en page ?

- La grille, c’est un ensemble de repères servant à organiser les éléments d’une page : elle aide le webdesigner dans son travail de composition.

7. Qu’est-ce que la notion d’affordance en UX Design ?

- L'affordance est la capacité d'un objet ou d'un système à évoquer son utilisation, sa fonction. Par définition, l'affordance provoque une interaction spontanée entre un environnement et son utilisateur. En ergonomie, elle permet de rendre l'utilisation d'un objet ou d'un service « intuitive »

8. Qu’est-ce qu’un « mobile first design » ?

- C'est un concept de web design optimisé pour le mobile qui va au-delà du responsive web design. Il consiste à concevoir un site en mettant la priorité sur la version mobile et en adaptant progressivement le web design pour les écrans plus large. En bref : C'est quand on vise à adapter notre application pour mobile en priorité, et ensuite on mettra les @mediaQueries par exemple, pour les autres écrans.

-----------------------------------------------------------------------------------------------------

IV. Programmation orientée objet (POO)

------------------------------------------------------------------------------------------------------

1. Donner une définition de la programmation orientée objet

- C'est un modèle de programmation qui repose sur le concept de classes et d'objets. C'est utilisé pour structurer un programme logiciel en éléments de code simples et réutilisables, généralement appelés classes, qui sont utilisés pour créer des instances individuelles d'objets.

2. Qu’est-ce qu’une classe ? Comment la déclare-t-on ?

- C'est un ensemble de code contenant des variables et des fonctions permettant de créer des objets. Elle peut contenir plusieurs objets. On la déclare avec le mot-clé "class" suivi du nom de la classe et des {} ; on déclare ensuite ses attributs et méthodes. En bref: Une classe est un ensemble d'attributs et se déclare avec un constructor.

3. Qu’est-ce qu’un objet ?

- On peut prendre une classe concrète au moyen d'un intance de classe (1 objet) : on instancie la classe. Un objet est un exemple concret de la classe. En PHP, on instancie une classe avec le mot-clé new (puis le nom de la classe derrière. Ex : "new Voiture()"), c'est à dire que l'on qu’on crée 1 objet.

4. Définir la notion de propriété / attribut / méthode

- Propriété : ?

Attribut : c'est un élément de description d'un objet, les caractéristiques (variables dites définies ou déclarées) d'un objet d'une classe.

Méthode : ce sont les fonctions (dites définies ou déclarées) à l'intérieur d'une classe.

5. Qu’est-ce que la visibilité d’une propriété ou d’une méthode ? Citer les différents types de visibilité.

- La visibilité permet de définir comment une propriété ou une méthode pourra être utilisée, les différents types de visibilité sont :

- public : permet d'indiquer que la propriété ou la méthode sera accessible à l'intérieur mais aussi à l'extérieur de la classe,

- private : permet d'indiquer que la propriété ou la méthode sera accessible à l'intérieur de la classe seulement,

- protected : permet d'indiquer que la propriété ou la méthode sera accessible à l'intérieur de la classe et des classes héritées.

6. Quelle est la méthode spécifique utilisée pour créer un nouvel objet à partir d’une classe .

- Le constructeur "\_\_construct($valeur1, $valeur2...)" permet de créer une nouvelle instance de classe. On peut y passer des arguments dans la méthode construct.

7. Qu’est-ce que l’encapsulation ?

- L'encapsulation consiste à restreindre l'accès à certains éléments d'une classe (le plus souvent ses attributs). Le but est de ne laisser accessible que le strict NÉCESSAIRE pour que la classe soit utilisable. En modifiant la visibilité d'une méthode (public, private, protected), on modifie le niveau d'encapsulation.

8. Que signifie « étendre une classe » ? Quelle est le concept clé mis en œuvre ? Donner un exemple.

- Cela signifie créer une classe à partir d’une autre existante. C'est le principe de l’héritage. La nouvelle classe va hériter des méthodes et propriétés de la classe qu’elle étend :

ex : 1 classe Personnage, 3 classes Guerrier, Archer et Magicien qui l’étendent.

9. Définir l’opérateur de résolution de portée.

- l’opérateur de résolution de portée (double deux-points (::) en hébreu) permet d’accéder à une constante, une propriété statique, une méthode d’une classe ou d’une de ses classes parentes. **::id**

10. Définir une méthode / propriété statique

- Le fait de déclarer des propriétés ou des méthodes comme statiques nous permet d'y accéder sans avoir besoin d'instancier la classe.

11. Définir le polymorphisme en POO

- Le nom de polymorphisme vient du grec et signifie qui peut prendre plusieurs formes. Cette caractéristique est un des concepts essentiels de la programmation orientée objet. Alors que l'héritage concerne les classes (et leur hiérarchie), le polymorphisme est relatif aux méthodes des objets. Des fonctions et des méthodes de mêmes noms peuvent avoir des comportements différents ou effectuer des opérations sur des données de types différents.

On distingue généralement trois types de polymorphisme :

- Le polymorphisme ad hoc (également surcharge ou en anglais overloading)

- Le polymorphisme paramétrique (également généricité ou en anglais template)

- Le polymorphisme d'héritage (également redéfinition, spécialisation ou en anglais overriding)

12. Définir une méthode / classe abstraite ?

- Une classe abstraite est une classe qui ne peut pas être directement instanciée.

Une classe abstraite encapsule des attributs et méthodes qui peuvent être utilisés par les instances des classes qui en héritent. L’intérêt des classes abstraites est de regrouper plusieurs classes sous un même nom de classe (par polymorphisme) ainsi que de décrire partiellement des attributs et méthodes communs à plusieurs classes.

13. Définir le chaînage de méthodes

- Chainer des méthodes nous permet d’exécuter plusieurs méthodes d’affilée de façon simple et plus rapide en les écrivant à la suite les unes des autres, « en chaine ». Exemple : "$topic->getMember() . " inscrit le " . $topic->getInscriptionDate()".

14. Qu’est-ce que la méthode \_\_toString() ? Existe-t-il d’autres méthodes « magiques »

- La méthode magique \_\_toString() va être appelée dès que l’on va traiter un objet comme une chaine de caractères (par exemple lorsqu’on tente d’echo un objet). Il existes d'autres méthodes magiques :

\_\_construct(), \_\_destruct(), \_\_call(), \_\_callStatic(), \_\_get(),

\_\_set(), \_\_isset(), \_\_unset(), \_\_clone(), \_\_sleep(),

\_\_wakeup(), \_\_invoke(), \_\_set\_state(), \_\_debugInfo().

15. Qu’est-ce qu’un « autoload » ?

- C'est une fonction qui va aller chercher toutes les classes existantes dans les autres fichiers php :

spl\_autoload\_register(function ($class\_name) {

require 'classes/'. $class\_name .'.php';} );

16. Comment appelle-t-on en français les « getters » et les « setters » ?

- Les « getters » sont appelés des « accesseurs » et les « setters » sont appelés des « mutateurs ». Un accesseur est une méthode qui permet de récupérer la valeur d'un attribut d'un objet, tandis qu'un mutateur est une méthode qui permet de modifier la valeur d'un attribut d'un objet.

En bref : Les getters renvoient la valeur de la variable et les setters permettent de la changer.

17. Qu’est-ce que la sérialisation en PHP ?

La sérialisation permet de stocker dans une variable des données complexes comme un tableau ou un objet, sous forme de chaines de caractères. C'est utilisé afin de les stocker en base de données (par exemple). Ex :

$data = array("name" => "John", "age" => 30, "city" => "New York");

$serialized\_data = serialize($data);

echo $serialized\_data;

On peut ensuite les désérialiser afin de retrouver l'objet / tableau d'origine.

$original\_data = unserialize($serialized\_data);

print\_r($original\_data);

----------------------------------------------------------------------------------------------------------------------

V. Architecture

----------------------------------------------------------------------------------------------------------------------

1. Qu’est-ce que l’architecture client / serveur ? Grâce à quel type de requête peut-on interroger le serveur ? Définir l’acronyme de ce type de requête. Si on ajoute un « S » à cet acronyme, expliquer la différence.

- Dans l'architecture serveur / client, on distingue les deux entités comme suit :

Le client envoie des requêtes au serveur.

Le serveur reçoit les requêtes du client, effectue des opérations demandées et renvoie les résultats au client.

Le client traite les réponses.

Pour interroger le serveur, on envoie des requêtes "HTTP" (Hypertext Transfer Protocol). Avec un "S", HTTPS devient "Hypertext Transfer Protocol Secure". La différence est que cette version est plus sécurisée en ajoutant une couche de sécurité avec des chiffrements pour éviter que les données passantes entre le serveur et le client ne soient interceptées.

2. Donner la définition d’un design pattern. Citer au moins 3 exemples de design pattern.

- Un design pattern est un patron d'architecture, ne solution réutilisable qui permet avoir quelques avantages :

respecter des méthodes de conception professionnellement reconnues ;

développer plus rapidement en suivant des modèles architecturaux ayant fait elurs prevues ;

permettre au code d'être relu plus facilement par un autre développeur sensibilisé aux pattenrs utilisés.

Exemple : Factory, Composite, Iterator, Observer, Singleton, Strategy, Template.

Singleton : Assure qu'une classe n'a qu'une seule instance et fournit un point d'accès global à cette instance.

Observer : Permet à un objet de notifier d'autres objets lorsqu'un changement d'état se produit.

Factory : Définit une interface pour créer des objets, mais laisse aux sous-classes le soin de décider quelle classe instancier.

3. Qu’est-ce que l’architecture MVC ?

- C'est un patron de conception concernant l'agencement du code, segmenté en trois sections :

modèle, vue, controlleur.

- L'architecture MVC est une façon d'organiser le code pour séparer les différentes responsabilités d'une application web. Cela permet d'avoir un code plus clair et plus facile à maintenir.

4. Quel est le rôle de chaque couche du design pattern MVC : Model, View, Controller ?

- Modèle : cette partie gère ce qu'on appelle la logique métier du site. Elle comprend notamment la gestion des données qui sont stockées, mais aussi tout le code qui prend des décisions autour de ces données. Son objectif est de fournir une interface d'action la plus simple possible au contrôleur. On y trouve donc entre autres des algorithmes complexes et des requêtes SQL.

- Vue : cette partie se concentre sur l'affichage. Elle ne fait presque aucun calcul et se contente de récupérer des variables pour savoir ce qu'elle doit afficher. On y trouve essentiellement du code HTML mais aussi quelques boucles et conditions PHP très simples, pour afficher par exemple une liste de messages.

- Contrôleur : cette partie gère les échanges avec l'utilisateur. C'est en quelque sorte l'intermédiaire entre l’utilisateur, le modèle et la vue. Le contrôleur va recevoir des requêtes de l'utilisateur. Pour chacune, il va demander au modèle d'effectuer certaines actions (lire des articles de blog depuis une base de données, supprimer un commentaire) et de lui renvoyer les résultats (la liste des articles, si la suppression est réussie). Puis il va adapter ce résultat et le donner à la vue. Enfin, il va renvoyer la nouvelle page HTML, générée par la vue, à l'utilisateur.

5. Quels sont les avantages de l’architecture MVC ?

- Séparation des responsabilités : Chaque couche a un rôle spécifique, ce qui rend le code plus clair et plus organisé.

- Facilité de maintenance : Il est plus facile de modifier une partie de l'application sans affecter les autres parties.

6. Existe-t-il des variantes à l’architecture MVC ?

- Modèle-Vue-Présentateur (MVP)

- Modèle-Vue-VueModèle (MVVM)

- Modèle-Adaptateur-Vue (MAV)

- Modèle-Vue-Contrôleur-Frontière (MVCF)

- Modèle-Interactions-Vues-État (MIVES)

7. Qu’est-ce qu’une API ? Définir l’architecture REST.

- C'est une interface de traitement d'applications entre un serveur web et un navigateur web.

- L'architecture REST (Representational State Transfer) définit les principes REST par 4 contrôles d'interface, notamment l'identification des ressources, la gestion des ressources via des représentations, l'activation des communications autodescriptives et la transformation de l'hypermédia en moteur de l'état de l'application.

Autre réponse plus précise:

- API (Application Programming Interface) : Une API est un ensemble de règles qui permet à des applications de communiquer entre elles.

- C'est comme un traducteur qui permet à différents logiciels de se parler.

- Architecture REST (Representational State Transfer) : REST est un style d'architecture pour les API web.

- Il utilise des méthodes HTTP standard comme GET, POST, PUT et DELETE pour effectuer des opérations sur les ressources.

- Les ressources sont identifiées par des URL, et les opérations sont réalisées de manière stateless, ce qui signifie que chaque requête est indépendante des autres.

--------------------------------------------------------------------------------------------------------------------------------------

VI. Modélisation - Base de données

--------------------------------------------------------------------------------------------------------------------------------------

1. Qu’est-ce que la modélisation de données ? Définir la méthode Merise

- C'est le processus de création d'une représentation visuelle ou d'un plan qui définit les systèmes de collecte et de gestion de l'information de toute organisation. La méthode Merise est une méthode d'analyse, de conception et de réalisation de systèmes d'informations.

2. Quelles sont les 3 étapes principales de la méthode Merise ?

- a. Analyse, conception et réalisation - Cette réponse OK.

- b. Planification, exécution et contrôle

- c. Création, modification et suppression

3. Qu’est-ce qu’un modèle conceptuel de données (MCD) en Merise ?

- Le MCD fournit une description graphique pour représenter des modèles de données sous la forme de diagrammes pouvant contenir des entités ou des associations.

4. Qu’est-ce qu’un modèle logique de données (MLD) en Merise ?

- C'est une représentation structurée et normalisée des données d'un système d’information, conçue pour être directement implémentable dans une base de données relationnelle.

5. Donner la définition des mots suivants :

a.Entité; b.Relation; c.Cardinalité; d.Clé primaire / clé étrangère

- a. Entité : Une entité est un objet ou un concept identifiable qui peut être représenté dans une base de données. Exemple : 'Livre' et 'Auteur'. Elle exprime une personne / chose / lieu / (une classe). "Ensemble d'informations à traiter"

b. Relation : Association relative entre les entités, relation qu'il y a entre elles. (porte bien son nom)

- c. Cardinalité : Sert à compter le nombre minimum et maximum de possibilités que chaque classe contient dans la relation liant deux ou plusieurs objets. C'est la relativité entre les entités, précisant le lien et la quantité minimale et maximale qu'une entité puisse avoir d'une autre entité.

d. Clé primaire : garanti l'unicité d'un enregistrement dans une table; identifiant unique d'une entité

Clé étrangère : garanti le lien avec une autre table; valeur récupérée de l'identifiant d'une autre entité.

Clé primaire dans table associative : c'est l'association des clés étrangères qui fait que la table associatve est la clé primaire.

6. Que devient une relation de type « Many To Many » dans le modèle logique de données ?

- elle devient une nouvelle table intermédiaire (table de liaison) entre ces entités, et portant les clés étrangères des entités en question

7. Qu’est-ce qu’une base de données ?

- Une base de données est un système organisé qui permet de stocker, gérer et récupérer des informations. Elle contient des collections de données structurées de manière à faciliter leur accès, leur gestion et leur mise à jour.

8. Définir les notions suivantes : a. SQL b.MySQL c.SGBD (donner 2 exemples de SGBD)

- a. SQL : Structured Query Language, c'est un langage informatique normalisé servant à exploiter des bases de données relationnelles.

- b. MySQL : MySQL est un système de gestion de bases de données (SGBD) relationnelles.

- c. SGBD (donner 2 exemples de SGBD) : Système de gestion de base de données, c'est un logiciel système permettant aux utilisateurs et programmeurs de créer et de gérer des bases de données, exemple : MySQL, Oracle et SQL Server.

9. Dans une base de données, les données sont stockées dans des \_\_\_.

Celles-ci sont constituées de lignes appelées \_\_\_ et de colonnes appelées \_\_\_

**- Dans une base de données, les données sont stockées dans des tables.**

Celles-ci sont constituées de lignes appelées enregistrements et de colonnes appelées champs.

10. Quelle est la différence entre une base de données relationnelle et non relationnelle ?

- Base de données relationnelle : Organise les données en tables avec des relations entre elles (clés primaires / clés étrangères). Les données sont structurées de manière rigide et on utilise SQL pour les manipuler.

- Base de données non relationnelle : Organise les données sous forme de documents, de paires clé-valeur (comme JSON) ou de graphes. Elles sont plus flexibles et peuvent gérer des types de données variés sans suivre de règles strictes.

11. Qu’est-ce qu’une jointure dans une base de données ? En existe-t-il plusieurs ? Si oui lesquelles ?

- Une jointure est une opération en SQL qui permet de combiner des lignes de deux ou plusieurs tables en fonction d'une condition commune:

\* INNER JOIN : Retourne les lignes où il y a une correspondance dans les deux tables.

\* LEFT JOIN : Retourne toutes les lignes de la table de gauche, et les lignes correspondantes de la table de droite, ou NULL si aucune correspondance n'est trouvée.

\* RIGHT JOIN : Retourne toutes les lignes de la table de droite, et les lignes correspondantes de la table de gauche, ou NULL si aucune correspondance n'est trouvée.

\* FULL JOIN : Retourne toutes les lignes lorsqu'il y a une correspondance dans une des tables.

12. A quoi sert une vue dans une base de données ?

-Pour des raisons de rapidité d’accès et de simplification, il existe les vues. Les vues sont des requêtes préparées à l’avance dont les paramètres (mais pas les résultats) sont stockés dans la base de données. Les interrogations complexes sont ainsi optimisées : en enregistrant un modèle plutôt qu’en le préparant de A à Z à chaque fois, l’utilisateur va plus vite et fait moins d’erreurs. Concrètement, la base de données stocke la requête qui peut ensuite être appelée (une vue a un nom) et exécutée. Grâce à ce nom, on peut ensuite les utiliser tels des tables dans les requêtes. C’est pratique, rapide et génère des requêtes plus lisibles tout en masquant la complexité sous-jacente.

Les vues ont d’autres caractéristiques :

Une vue peut masquer certaines données. Étant une requête, la vue n’affiche que les informations souhaitées.

Elle permet ainsi d’extraire des données choisies et de masquer des données non souhaitées.

Certains SGBD permettent de stocker les résultats des vues et peuvent faire office de mémoire tampon (cache).

Les vues ne prennent pas de place. C’est la requête qui est enregistrée (pas le résultat).

On peut réaliser des regroupements de données (somme, moyenne…) qui peuvent être réutilisés très facilement.

Les vues sont habituellement en lecture seule (d’ou le terme de « vue ») mais dans certains systèmes et sous certains conditions, on peut manipuler les données d’une vue.

\*\*\* Création d’une vue SQL \*\*\*

CREATE VIEW ToutesLesFactures AS

SELECT fac\_id, fac\_ref, cli\_nom

FROM facture, client

WHERE facture.cli\_id = client.cli\_id ;

\*\*\* Utilisation d’une vue SQL \*\*\*

Plutôt que de retaper la requête ci-dessus avec sa jointure, on peut directement appeler la vue :

SELECT \* FROM ToutesLesFactures ;

Une base de données n’a pas besoin d’interface graphique pour fonctionner. Et c’est très perturbant lorsque l’on débute avec ces outils. Alors il existe des outils variés qui permettent de représenter la structure d’une base de données (le MCD de Merise) et les données qui se trouvent à l’intérieur

(PhpMyAdmin pour MySql par exemple). Mais cela ne suffit pas et ne doit pas être confondu avec les vues.

13. Qu’est-ce que l’intégrité référentielle dans une base de données ?

-L'intégrité référentielle garantit que les relations entre les tables restent cohérentes.

Par exemple, si une table contient une clé étrangère reliant à une autre table, l'intégrité référentielle assure que cette clé étrangère correspond à une clé primaire valide dans la table liée.

14. Quelles sont les fonctions d’agrégation en SQL ?

-Les fonctions d'agrégation en SQL permettent de réaliser des calculs sur un ensemble de valeurs et de retourner une valeur unique. Exemples :

COUNT() : Compte le nombre de lignes.

SUM() : Calcule la somme des valeurs.

AVG() : Calcule la moyenne des valeurs.

MAX() : Retourne la valeur maximale.

MIN() : Retourne la valeur minimale.

15. Qu’est-ce qu’un CRUD dans le contexte d’une base de données ?

-CRUD est un acronyme pour les quatre opérations de base qu'on peut réaliser sur les données dans une base de données :

Create (Créer) : Ajouter de nouvelles données.

Read (Lire) : Lire ou récupérer des données existantes.

Update (Mettre à jour) : Modifier des données existantes.

Delete (Supprimer) : Effacer des données existantes.

16. Quelles sont les clauses qui permettent de :

a. Insérer un nouvel enregistrement dans une table

INSERT INTO table (key1, key2, key3)

VALUES (valeur1, valeur2, valeur3)

b. Modifier un enregistrement dans une table

UPDATE table

SET key = valeur, key = valeur, ...

WHERE ...

c. Supprimer un enregistrement dans une table

DELETE FROM table

WHERE ...

d. Supprimer la base de données

DROP DATABASE nom\_DB

e. Filtrer les résultats d’une requête SQL

SELECT colonne1, colonne2, ...

FROM table

WHERE ...;

f. Trier les résultats d’une requête SELECT

ORDER BY colonne ( DESC/ASC, colonne2 DESC/ASC )

g. Regrouper les résultats d'une requête SELECT en fonction d'une colonne spécifique

SELECT colonne1, COUNT(colonne2)/SUM(colonne2)

FROM table

GROUP BY colonne1;

h. Concaténer 2 chaînes de caractères

-SELECT CONCAT(blabla1, blabla2) AS blablabla

17. Comment se connecter à une base de données en PHP ? Quelle est la classe native utilisée ?

-La classe native utilisée est PDO. On créait donc un nouvel objet PDO en renseignant serveur, nom de la bdd, et toute information encore nécessaire à la connexion.

---------------------------------------------------------------------------------------------------------------------------

VII. Symfony

--------------------------------------------------------------------------------------------------------------------------

1. Qu’est-ce que Symfony ?

- Symfony est un framework PHP "open source" utilisé par les développeurs pour créer des sites ou applications web complexes, robustes, fiables, évolutifs, maintenables et performants.

2. Sur quel langage de programmation et design pattern repose Symfony ?

- Le langage de programmation sur lequel repose Symfony est le PHP. Le design pattern sur lequel repose Symfony est le modèle de conception MVC (Modèle-Vue-Contrôleur). Les données de l'application (le modèle) sont séparées de l'interface utilisateur (la vue) et de la logique de contrôle (le contrôleur), ce qui permet une meilleure organisation du code et une meilleure maintenabilité de l'application.

3. Quelle est la dernière version en date de Symfony ?

- La dernière version en date de Symfony est là 7.0.5 (Stable Release, 6.4.5 pour la LTS (Long-Term Support)).

4. Qu’est-ce qu’un bundle ?

- Un bundle est une offre groupée, un ensemble de produits vendus ensemble.

5. Quel est le moteur de template utilisé par défaut dans Symfony ?

- Le moteur de template par défaut est Twig. c'est un moteur de template flexible, rapide et sécurisé, spécialement conçu pour être utilisé avec PHP. Il est simple à utiliser et a une bonne capacité à séparer efficacement la logique de présentation de la logique métier dans les applications web.

6. Qu’est-ce qu’un ORM ? Quel est son utilité et comment s’appelle-t-il au sein de Symfony ?

7. Qu’est-ce que l’injection de dépendances ? Quel est l’outil utilisé dans ce contexte et quel fichier contient l’intégralité des dépendances du projet ?

8. Que permet le bundle Maker au sein de Symfony ?

9. Quel est le langage de requêtage exploité au sein d’un projet Symfony ?

10. Quel est le composant qui garantit l’authentification et l’autorisation des utilisateurs ?

------------------------------------------------------------------------------------------------------------------------

VIII. Sécurité

------------------------------------------------------------------------------------------------------------------------

1. Qu’est-ce que l’injection SQL ? Comment s’en prémunir ?

- Elle consiste à faire de l'insertion de code SQL non autorisé dans des requêtes SQL afin d'avoir accès à la base de données. On s'en défend en utilisant prepare(), puis execute() afin de préparer un moule non modifiable de la requête SQL avant de l'exécuter en y insérant des variables qui seront au préalable filtrées.

2. Qu’est-ce que la faille XSS ? Comment s’en prémunir ?

- La faille XSS consiste à injecter du script dans les pages web.

3. Qu’est-ce que la faille CSRF ? Comment s’en prémunir ?

4. Définir l’attaque par force brute et l’attaque par dictionnaire

5. Existe-t-il d’autres failles de sécurité ? Citer celles-ci et expliquer simplement leur comportement

6. A quoi servent l’authentification et l’autorisation dans un contexte d’application web ?

7. Définir la notion de hachage d’un mot de passe et citer des algorithmes de hachage

8. Qu’est-ce qu’une politique de mots de passe forts ?

9. Qu’est-ce que l’hameçonnage ?

10. Définir la « validation des entrées »

-----------------------------------------------------------------------------------------------------------------------

IX. RGPD

------------------------------------------------------------------------------------------------------------------------

1. Qu’est-ce que le RGPD ?

- C'est le Règlement Général de la Protection des Données.

2. Quel est son objectif principal ?

3. Quelle est la date d’entrée en vigueur du RGPD ?

4. Quelles sont les sanctions possibles en cas de non-respect du RGPD ?

5. En France, quel est l’autorité administrative qui s’occupe de faire appliquer le RGPD ?

6. Quel est le consentement valide selon le RPGD ?

7. Qu’est-ce qu’une politique de confidentialité ?

8. Quelle est la durée de conservation maximale des données personnelles selon le RGPD ?

9. Quels sont les droits des utilisateurs selon le RGPD ?

10. Qu’est-ce que le principe de minimisation des données selon le RGPD ?

-------------------------------------------------------------------------------------------------------------------------

X. SEO

-------------------------------------------------------------------------------------------------------------------------

1. Qu’est-ce que le SEO ?

- Le SEO (Search Engine Optimization) est l'acronyme qui signifie « Optimisation pour les moteurs de recherche » en français.

2. Quel est l’objectif principal du SEO ?

- C'est un ensemble de techniques permettant d'améliorer le positionnement d'un site web dans les moteurs de recherche (Google, Bing, Yahoo, etc.) afin de le rendre visible auprès des internautes.

3. Existe-t-il plusieurs types de référencement ? Lesquels ?

- Les trois différents types de référencement sont le SEO sur-page, le SEO hors-page et le SEO technique.

4. Qu’est-ce que la densité de mots-clés en SEO ?

5. Qu’est-ce qu’une balise « alt » ?

- La balise ALT, également connue sous le nom d'« attribut ALT », correspond au texte alternatif d'une image ou d'un visuel sur une page Internet. Il fait partie des champs possibles à remplir en codage HTML et permet de donner une description de l'image ou du visuel si ce dernier n'apparaît pas à l'écran.

6. Qu’est-ce que la balise « meta description » ?

- La Meta Description est une des balises HTML les plus connues en référencement. Elle permet de fournir une description aux moteurs de recherche pour l’affichage des résultats au sein des SERP ou aux applications web ayant besoin d’un petit résumé rapide. De plus, lorsqu’elle est de qualité, elle peut inciter l’internaute à cliquer sur votre site ce qui augmente le trafic naturel.

7. Qu’est-ce que le « nofollow » en SEO ?

- Un lien nofollow est un lien inséré par l'éditeur sur un contenu de son site internet. Ce lien a la particularité de donner l'indication aux moteurs de recherche de ne pas suivre son contenu et donc de ne pas le prendre en compte pour le référencement naturel du site.

8. Quelle est l'importance du contenu de qualité pour le référencement d'un site web ?

9. Pourquoi est-il important d'utiliser des balises de titre (h1, h2, h3…) de manière structurée ?

10. Quelle est la recommandation pour les URL d'un site web bien référencé ?

11. Qu'est-ce que le maillage interne et pourquoi est-il important pour le référencement ?

12. Qu'est-ce que l'optimisation des images pour le référencement ?

13. Qu'est-ce qu'un plan de site (sitemap) et pourquoi est-il important pour le référencement ?

-----------------------------------------------------------------------------------------------------------------

XI. Gestion de projets - DevOps

-----------------------------------------------------------------------------------------------------------------

1. Qu’est-ce que la gestion de projet ?

- La gestion de projet, ou le management de projet, est un mode de réalisation d'un projet, ou' l'application des techniques de gestion pendant le cycle de vie du projet permet d'atteindre des objectifs précis. Le gestionnaire de projet est le responsable de l'organisation, de la coordination et de suivi de tous les acteurs et les ressources impliqués dans le projet. Il doit aussi gérer les contraintes, les risques et les impacts du projet.

Les principales étapes de la gestion de projets: Conception->Planification->Exécution->Contrôle->Clôture->Bilan.

2. Qu’est-ce qu’une méthode Agile de gestion de projet ?

- La méthode Agile est une méthode flexible qui consiste à découper un projet en plusieurs petits projets, de sorte que le client puisse valider chaque étape au fur et à mesure. La méthode Agile est un parfait moyen de s’assurer la satisfaction de ses clients et un minimum de retours à la fin du projet. Elle implique une collaboration constante tant entre les membres de l'équipe qu'avec les parties prenantes du projets, une amélioration et une itération continues à chaque étape. Une fois le travail commencé, les équipes suivent un processus de planification, d'exécution et d'évaluation, qui permet de modifier facilement le livrable final pour mieux répondre aux besoins du client.

3. Expliquer la méthode MoSCoW en quelques lignes et citer ses avantages.

- La méthode MoSCoW est une technique de priorisation utilisée par les Chefs de projet pour travailler plus intelligemment. MoSCoW est un acronyme (en réalité, seuls MSCW désignent une catégorie spécifique, les « o » étant utilisés pour faciliter la prononciation).

Voici ce que signifie l’acronyme MoSCoW.

- M : Must have this (les initiatives qui sont essentielles à la réussite du projet. Elles peuvent apporter une vraie valeur ajoutée).

- C : Could have this if it does not affect anything else (les initiatives qui ne sont pas nécessaires au cœur d’un produit, elles sont souvent dépriorisés lorsqu’un autre projet prend plus de temps que prévu.

- W : Won't have this time but would like in the future (plusieurs initiatives non-essentielles.

Cette catégorie montre à l’équipe que le projet n’est pas une priorité à ce moment précis).

Voici quelques-uns des avantages de la méthode de priorisation MoSCoW :

- Flexibilité lors de la hiérarchisation des tâches : La méthode fait la distinction entre les fonctionnalités « indispensables » et celles qui ne le sont pas. Grâce à cela, on peut donner la priorité aux fonctionnalités urgentes et nécessaires pour construire le produit.Vous pouvez ensuite mettre ces dernières de côté pendant un certain temps au profit de celles qui sont moins prioritaires.

- Transparence pendant la période de développement : MoSCoW permet à tous les acteurs clés impliqués dans le développement d’un produit de se concentrer sur une seule liste de priorités. C’est essentiel, car tout le monde comprend chaque élément « indispensable ». Avec ce niveau de transparence, les développeurs réalisent rapidement quels éléments inclure dans chaque sprint ou itération.

- Division des ressources : Un autre avantage de MoSCoW est une division claire de l’allocation des ressources pendant le projet.

Le fait d’avoir des « Must Have » séparés des « Should Have » et des « « Could Have » permet à votre équipe d’affecter clairement des ressources aux fonctionnalités qui sont des ajouts indispensables et d’affecter des ressources aux autres fonctionnalités possibles après les avoir déléguées aux sections indispensables. Cette répartition des ressources est essentielle lors de la gestion d’un projet complexe et aide l’ensemble de l’équipe à comprendre ses attentes et à répartir son temps.

- La méthode est efficace pour: hiérarchiser en fonction des contraintes budgétaires; établir des priorités en fonction des compétences de l’équipe; définir des priorités en fonction des besoins concurrents de l’entreprise.

4. A quoi sert la méthodologie MVP ? Citer les caractéristiques clés.

- Né dans la Silicon Valley, le concept fait partie de Lean Startup, une méthode agile très appréciée lors d’une création d’entreprise et par les startups. Son processus itératif agit comme un accélérateur de lancement d’un produit sur le marché, et ce à moindre coût. La méthodologie MVP, qui signifie “Minimum Viable Product” (Produit Minimum Viable), sert à développer un nouveau produit avec suffisamment de fonctionnalités pour attirer les premiers utilisateurs et valider une idée de produit tôt dans le cycle de développement. Cela permet aux équipes de collecter des données sur les réactions des utilisateurs pour itérer et améliorer le produit.

"Minimum" - répondre au besoin principal; objectif : le développement rapide du produit en diminuant le temps de dev et coûts; astuce : automatiser rapidement et sans développement certaines fonctionnalités.

"Viable" - mettre un MVP sur le marché, tester sa capacité à générer de la valeur; objectif :les ressources mises en œuvre pour le développement génèrent un retour sur investissement; astuce: réinvestir les revenus générés par un produit minimum.

"Product" MVP est une version succincte du produit fini mais il a déjà les bons stabilité technique&expérience utilisateur); objectif : cette première version doit séduire ses users; astuce : récoltez des retours utilisateurs pour la suite du dév.

5. Qu’est-ce que la planification itérative ?

- Tout comme la méthode Agile, les processus itératifs peuvent vous aider à réduire les risques, travailler plus efficacement et aborder les problèmes de manière plus souple et dynamique. La planification itérative désigne la pratique qui consiste à créer, affiner et améliorer un projet, un produit ou une initiative. Les équipes créent, testent et révisent jusqu’à ce qu’elles soient satisfaites du résultat final. Vous pouvez considérer le processus itératif comme une méthode essai-erreur qui rapproche votre projet de son objectif final. La méthodologie Lean et la gestion de projet Agile intègrent ces processus itératifs, absolument essentiels. Opter pour la planification itérative implique d’améliorer constamment la conception, produit ou projet jusqu’à ce votre équipe soit satisfaite du livrable final. Dans un processus autre qu’itératif, l'équipe cherche à concevoir son produit final sans tester de nouvelles idées en cours de route. Les démarches qui ne sont pas itératives connaissent des phases de conceptualisation et de création plus longues.

6. Citer 3 méthodes Agiles dans le cadre d’un projet informatique.

1) Scrum est un cadre dans lequel les gens peuvent résoudre des problèmes adaptatifs complexes, tout en fournissant de manière productive et créative des produits de la plus haute valeur possible. L’équipe de développement découpe son produit en tâches. Elle devient plus agile et découvre comment réagir rapidement et répondre aux imprévus.

2) Kanban, comme Scrum, encourage le travail à être décomposé en petites tâches.Mais plutôt que de planifier le travail dans une itération ou un Sprint, les membres de l’équipe récupèrent la tâche la plus prioritaire dans le backlog qui est prête à être développée. Il repose sur un travail effectué en toute transparence et une communication en temps réel de la capacité. Les tâches sont représentées visuellement sur un tableau Kanban. Les membres de l'équipe peuvent voir l'état de chaque tâche à tout moment. Un tableau Kanban est un outil de gestion de projet Agile conçu pour aider à visualiser le travail, limiter le travail en cours et maximiser l'efficacité (ou le flux). Ces tableaux peuvent aider les équipes Agile et DevOps à mettre de l'ordre dans leur travail quotidien. Les tableaux Kanban ont recours à des cartes, à des colonnes et à l'amélioration continue pour aider les équipes technologiques et de service à s'engager sur une quantité de travail appropriée, puis à la réalise

3) Extreme Programming (XP) est une méthode qui se concentre sur l’excellence technique et une bonne conception tout en répondant aux besoins changeants du client, elle privilége l'aspect réalisation d'une application, sans négliger l'aspect gestion de projet. L'originalité de la méthode est de les pousser à l'extrême : la revue de code sera faite en permanence (par un binôme) ; les tests seront faits systématiquement avant chaque mise en œuvre ; le code sera retravaillé tout au long du projet (refactoring ou remaniement du code) ; la solution la plus simple sera toujours celle qui sera retenue ; des métaphores seront définies et évolueront en concomitance ; les modifications seront faites plusieurs fois par jour ; des cycles de développement très rapides faciliteront l'adaptation au changement.

7. Qu’est-ce qu’une réunion de revue de projet ?

- Une réunion de revue de projet est une rencontre organisée pour évaluer l’avancement d’un projet. Elle permet aux parties prenantes de discuter des progrès réalisés, des obstacles rencontrés et des étapes suivantes. C’est un moment clé pour s’assurer que le projet reste sur la bonne voie et respecte les délais et le budget prévus. Elle peut se composer, par exemple, comme suit : le commanditaire du projet, le chef de projet, l'équipe projet, des experts et/ou partenaires, un sponsor.

Objectifs : contrôler que les objectifs d'une étape sont bien atteints , vérifier que les échéances et délais sont respectés , valider les différentes phases du projet, gérer les incidents, problèmes ou risques éventuels, décider d'éventuels réajustements ou réorientations , déclencher un appel de fonds , mais également : fédérer à nouveau les acteurs autour de la réussite du projet, communiquer en face à face, renforcer l'intelligence collective ,favoriser la coopération entre les différents protagonistes, donner une vision globale, unique et précise du projet à chacune des parties en présence. Plusieurs types de revues de projet, parmi lesquels :

Comité de pilotage : composé de dirigeants opérationnels dans la maîtrise d'ouvrage du projet, il permet de chapeauter le projet dans sa globalité : constitution de l'équipe, désignation du chef de projet, définition des phases, échéance et diverses réunions de suivi, validation des étapes, prise de décision, etc.

Revue budgétaire : implémentée dans le cadre de la gestion budgétaire, elle permet le contrôle et le suivi des dépenses et recettes :suivi des écarts, révisions, réajustements, prévisions.

Revue de sprint : dans le cadre de projets agiles , elle se tient à la fin de chaque sprint et permet de présenter ce qui a été développé au cours du sprint écoulé, faire un point sur l'avancée du projet, valider ou réadapter les options.

Revue de fabrication : dans le cadre de la fabrication d'un nouveau produit, elle permet d'en valider toutes les étapes, faire les ajustements éventuels, valider la conformité, etc.

Déroulement de ce type de réunion: préparation; invitation; conduite de la réunion; précision, concision et positivité; conclusion; compte-rendu; dossier de revue de projet; suivi.

8. Qu’est-ce qu’un livrable dans un projet ?

- La conduite d'un projet débouche sur un produit, un service, etc. Cette finalité, appelée "livrable", est le résultat tangible d'une production réelle, appréhendable, mesurable attendue par le client final. Un projet peut, bien sûr, avoir plusieurs livrables.

9. Quels sont les 3 piliers SCRUM ? Définir chacun d’entre eux

-Les trois piliers de Scrum sont:

1) Transparence : les faits sont présentés tels qu'ils sont à toutes les parties prenantes qui sont concernées directement ou indirectement.

2)Inspection : le travail est vérifié au fur et à mesure pour éviter de devoir refaire deux fois la même chose.

3)Adaptation : l'équipe Scrum s'adapte continuellement au projet, à son contexte, et à ce qu'elle apprend en cours de route grâce à l'empirisme.

10. Qu’est-ce que le DevOps et quel est son objectif principal ?

- le DevOps est une approche conciliant le développement et les opérations.

— est un mouvement en ingénierie informatique et une pratique technique visant à l'unification du développement logiciel (dev) et de l'administration des infrastructures informatiques (ops), notamment l'administration système. Couramment utilisée dans le domaine du développement logiciel, elle est aussi adoptée pour la Data Science et le Machine Learning. DevOps est une pratique de collaboration entre Dev et Ops. Elle vise à supprimer les contraintes entre ces équipes, pour accélérer la livraison de valeur aux utilisateurs. En deux mots, il s’agit de mettre la collaboration entre les équipes au service de l’amélioration continue du système de production logicielle.

11. Qu’est-ce que l’intégration continue ?

- L'intégration continue est une méthode de développement de logiciel DevOps par laquelle les développeurs intègrent régulièrement leurs modifications de code à un référentiel centralisé, suite à quoi des opérations de création et de test sont automatiquement exécutées.

12. Qu’est-ce que Docker ? Et en quoi est-il utile dans le cadre du DevOps ?

- Docker est une plateforme permettant de lancer certaines applications dans des conteneurs logiciels lancée en 2013. Docker est un outil qui peut empaqueter une application et ses dépendances dans un conteneur isolé, qui pourra être exécuté sur n'importe quel serveur ». Il ne s'agit pas de virtualisation, mais de conteneurisation, une forme plus légère qui s'appuie sur certaines parties de la machine hôte pour son fonctionnement. Cette approche permet d'accroître la flexibilité et la portabilité d’exécution d'une application, laquelle va pouvoir tourner de façon fiable et prévisible sur une grande variété de machines hôtes, que ce soit sur la machine locale, un cloud privé ou public, une machine nue, etc. Il s'agit actuellement du moteur de conteneurisation le plus utilisé. Un grand avantage de Docker est la possibilité de modéliser chaque conteneur sous la forme d'une image que l'on peut stocker localement ou sur le Docker Hub, endroit public où de nombreuses images sont publiées et mises à jour régulièrement.

13. Qu’est-ce qu’un test unitaire ?

- une procédure permettant de vérifier le bon fonctionnement d'une partie précise d'un logiciel ou d'une portion d'un programme (appelée « unité » ou « module »).

14. Quelle est l'unité de code testée lors d'un test unitaire ?

- Ces unités peuvent être des fonctions, des méthodes, ou des classes. L'objectif principal est d'isoler chaque partie du programme et de vérifier son bon fonctionnement de manière indépendante.

15. Quelles sont les caractéristiques d'un bon test unitaire ?

1. Isolé Chaque test unitaire doit pouvoir être autonome, ce qui signifie qu’il peut exister indépendamment d’autres facteurs. ...

2. Rapide Tenez compte du volume de code à tester et du temps qu’il faudrait pour effectuer suffisamment de tests pour obtenir des résultats satisfaisants. ...

3. Cohérent Les tests unitaires doivent renvoyer des résultats identiques à chaque fois. ...

4. Autocontrôle .

16. Qu'est-ce qu'une assertion dans un test unitaire ?

- Les tests unitaires ne sont utiles que s'ils incluent des assertions qui valident que la méthode testée renvoie la bonne valeur et/ou ajuste l'état des autres objets comme prévu.

------------------------------------------------------------------------------------------------------------------

XII. English

------------------------------------------------------------------------------------------------------------------

1) What does JavaScript enable you to do on a website ?

a. Add interactive behavior and dynamic content - OK

b. Define the layout and design of web pages

c. Handle server-side operations

2) Which programming language is primarily used for server-side web development ?

a. PHP - OK

b. JavaScript

c. HTML

3) What is the purpose of a web browser ?

a. To render and display web pages - OK

b. To execute serve-side code

c. To manage databases

4) What is the difference between GET and POST methods in HTTP ?

a. GET retrieves data from a server, while POST submits data to a server - OK

b. GET submits data to a server, while POST retrieves data from a server

c. GET and POST methods are interchangeable

5) What is the purpose of version control systems (e.g., Git) in web development ?

a. To track changes and manage collaborative development - OK

b. To optimize website loading speed

c. To handle server-side scripting

6) What is the purpose of a framework in web development ?

a. To provide a structured environment for building web applications - OK

b. To handle network protocols and data transfer

c. To create visual designs and layouts for websites

7) What does NoSQL stand for ?

a. Not Only SQL - OK

b. Non-Structured Query Language

c. New Object-Oriented Language

8) Which of the following is a characteristic of NoSQL databases ?

a. Strict schema enforcement

b. Support for complex transactions

c. Scalability and flexible data models - OK