Table des matières

[1 Général 7](#_Toc153193718)

[1.1 Quel est l’environnement à installer pour exécuter un script PHP ? Citer 2 exemples de logiciels permettant ce contexte 7](#_Toc153193719)

[1.2 Qu’est-ce qu’un algorithme ? 7](#_Toc153193720)

[1.3 Qu’est-ce qu’une variable ? Par quel symbole est préfixée une variable en PHP ? 7](#_Toc153193721)

[1.4 Qu’est-ce que la portée d’une variable ? 7](#_Toc153193722)

[1.5 Qu’est-ce qu’une constante ? Quelle est la différence avec une variable ? 7](#_Toc153193723)

[1.6 Qu’est-ce qu’une superglobale, combien en existent-ils et donner un exemple d’utilisation 7](#_Toc153193724)

[1.7 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) 7](#_Toc153193725)

[1.8 Existe-t-il plusieurs types de tableaux en PHP, si oui lesquels ? 7](#_Toc153193726)

[1.9 Quelles sont les différentes structures de contrôles qu’il existe en algorithmie ? Donner un exemple pour chacune d’entre elles 7](#_Toc153193727)

[1.10 Quelle est la fonction PHP permettant de demander la longueur d’une chaîne de caractères ? 7](#_Toc153193728)

[1.11 Qu’est-ce qu’une session ? Quelle fonction permet de démarrer une session en PHP ? Donner un exemple d’utilisation en PHP 7](#_Toc153193729)

[1.12 Qu’est-ce qu’un cookie ? Donner un exemple d’utilisation en PHP 7](#_Toc153193730)

[1.13 Quelle est la différence entre les instructions « require » et « include » en PHP 7](#_Toc153193731)

[1.14 Comment effectuer une redirection en PHP ? 7](#_Toc153193732)

[1.15 Définir la partie « front-end » et « back-end » d’une application 7](#_Toc153193733)

[1.16 Définir le contrôle de version ? Qu’est-ce que Git ? 7](#_Toc153193734)

[1.17 Qu’est-ce qu’un CMS ? Citer au moins 2 exemples 7](#_Toc153193735)

[2 Front-end 8](#_Toc153193736)

[2.1 Définir HTML 8](#_Toc153193737)

[2.2 Définir CSS 8](#_Toc153193738)

[2.3 Définir Javascript 8](#_Toc153193739)

[2.4 Définir JSON. Dans quel contexte ce format est-il utilisé ? 8](#_Toc153193740)

[2.5 Peut-on interpréter du Javascript côté serveur ? Si oui, comment ? 8](#_Toc153193741)

[2.6 Qu’est-ce qu’un sélecteur CSS ? 8](#_Toc153193742)

[2.7 Quelle balise HTML permet de créer un lien hypertexte ? 8](#_Toc153193743)

[2.8 Qu’est-ce qu’une requête AJAX ? 8](#_Toc153193744)

[2.9 Quel sélecteur CSS permet de sélectionner tous les éléments d’une classe spécifique ? D’un identifiant spécifique ? 8](#_Toc153193745)

[2.10 Définir le responsive design 8](#_Toc153193746)

[2.11 Qu’est-ce que le templating ? 8](#_Toc153193747)

[2.12 Qu’est-ce qu’une fonction anonyme en Javascript ? 8](#_Toc153193748)

[2.13 Quelle méthode JavaScript est utilisée pour ajouter un élément à la fin d'un tableau ? 8](#_Toc153193749)

[2.14 Qu’est-ce qu’un « media query » ? 8](#_Toc153193750)

[2.15 Qu’est-ce qu’un pseudo élément en CSS ? 8](#_Toc153193751)

[2.16 Qu’est-ce que Bootstrap ? Donner d’autres exemples équivalent 8](#_Toc153193752)

[2.17 Quand un formulaire HTML est créé, quelles sont les 2 méthodes qui peuvent lui être associées ? Donner la différence entre ces 2 méthodes 8](#_Toc153193753)

[3 UX / UI 9](#_Toc153193754)

[3.1 Quelle est la différence entre UX Design et UI Design ? 9](#_Toc153193755)

[3.2 Qu’est-ce qu’un wireframe ? 9](#_Toc153193756)

[3.3 Qu’est-ce qu’un prototype ? 9](#_Toc153193757)

[3.4 Qu’est-ce que la hiérarchie visuelle en UI Design ? 9](#_Toc153193758)

[3.5 Qu’est-ce que l’accessibilité en UX Design ? 9](#_Toc153193759)

[3.6 Qu’est-ce qu’une grille de mise en page ? 9](#_Toc153193760)

[3.7 Qu’est-ce que la notion d’affordance en UX Design ? 9](#_Toc153193761)

[3.8 Qu’est-ce qu’un « mobile first design » ? 9](#_Toc153193762)

[4 Programmation orientée objet (POO) 10](#_Toc153193763)

[4.1 Donner une définition de la programmation orientée objet 10](#_Toc153193764)

[4.2 Qu’est-ce qu’une classe ? Comment la déclare-t-on ? 10](#_Toc153193765)

[4.3 Qu’est-ce qu’un objet ? 10](#_Toc153193766)

[4.4 Définir la notion de propriété / attribut / méthode 10](#_Toc153193767)

[4.5 Qu’est-ce que la visibilité d’une propriété ou d’une méthode ? Citer les différents types de visibilité 10](#_Toc153193768)

[4.6 Quelle est la méthode spécifique utilisée pour créer un nouvel objet à partir d’une classe ? 10](#_Toc153193769)

[4.7 Qu’est-ce que l’encapsulation ? 10](#_Toc153193770)

[4.8 Que signifie « étendre une classe » ? Quelle est le concept clé mis en œuvre ? Donner un exemple 10](#_Toc153193771)

[4.9 Définir l’opérateur de résolution de portée 10](#_Toc153193772)

[4.10 Définir une méthode / propriété statique 10](#_Toc153193773)

[4.11 Définir le polymorphisme en POO 10](#_Toc153193774)

[4.12 Définir une méthode / classe abstraite ? 10](#_Toc153193775)

[4.13 Définir le chaînage de méthodes 10](#_Toc153193776)

[4.14 Qu’est-ce que la méthode \_\_toString() ? Existe-t-il d’autres méthodes « magiques » 10](#_Toc153193777)

[4.15 Qu’est-ce qu’un « autoload » ? 10](#_Toc153193778)

[4.16 Comment appelle-t-on en français les « getters » et les « setters » ? 10](#_Toc153193779)

[4.17 Qu’est-ce que la sérialisation en PHP 10](#_Toc153193780)

[5 Architecture 11](#_Toc153193781)

[5.1 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 11](#_Toc153193782)

[5.2 Donner la définition d’un design pattern. Citer au moins 3 exemples de design pattern 11](#_Toc153193783)

[5.3 Qu’est-ce que l’architecture MVC ? 11](#_Toc153193784)

[5.4 Quel est le rôle de chaque couche du design pattern MVC : Model, View, Controller ? 11](#_Toc153193785)

[5.5 Quels sont les avantages de l’architecture MVC ? 11](#_Toc153193786)

[5.6 Existe-t-il des variantes à l’architecture MVC ? 11](#_Toc153193787)

[5.7 Qu’est-ce qu’une API ? Définir l’architecture REST 11](#_Toc153193788)

[6 Modélisation / Base de données 12](#_Toc153193789)

[6.1 Qu’est-ce que la modélisation de données ? Définir la méthode Merise 12](#_Toc153193790)

[6.2 Quelles sont les 3 étapes principales de la méthode Merise ? a. Analyse, conception et réalisation b. Planification, exécution et contrôle c. Création, modification et suppression 12](#_Toc153193791)

[6.3 Qu’est-ce qu’un modèle conceptuel de données (MCD) en Merise ? 12](#_Toc153193792)

[6.4 Qu’est-ce qu’un modèle logique de données (MLD) en Merise ? 12](#_Toc153193793)

[6.5 Donner la définition des mots suivants : a. Entité b. Relation c. Cardinalité d. Clé primaire / clé étrangère 12](#_Toc153193794)

[6.6 Que devient une relation de type « Many To Many » dans le modèle logique de données ? 12](#_Toc153193795)

[6.7 Qu’est-ce qu’une base de données ? 12](#_Toc153193796)

[6.8 Définir les notions suivantes : a. SQL b. MySQL c. SGBD (donner 2 exemples de SGBD) 12](#_Toc153193797)

[6.9 Dans une base de données, les données sont stockées dans des \_\_\_. Celles-ci sont constituées de lignes appelées \_\_\_ et de colonnes appelées \_\_\_ 12](#_Toc153193798)

[6.10 Quelle est la différence entre une base de données relationnelle et non relationnelle ? 12](#_Toc153193799)

[6.11 Qu’est-ce qu’une jointure dans une base de données ? En existe-t-il plusieurs ? Si oui lesquelles ? 12](#_Toc153193800)

[6.12 A quoi sert une vue dans une base de données ? 12](#_Toc153193801)

[6.13 Qu’est-ce que l’intégrité référentielle dans une base de données ? 12](#_Toc153193802)

[6.14 Quelles sont les fonctions d’agrégation en SQL ? 12](#_Toc153193803)

[6.15 Qu’est ce qu’un CRUD dans le contexte d’une base de données ? 12](#_Toc153193804)

[6.16 Quelles sont les clauses qui permettent de : a. Insérer un nouvel enregistrement dans une table b. Modifier un enregistrement dans une table c. Supprimer un enregistrement dans une table d. Supprimer la base de données e. Filtrer les résultats d’une requte SQL f. Trier les résultats d’une requête SELECT g. Regrouper les résultats d'une requête SELECT en fonction d'une colonne spécifique h. Concaténer 2 chaînes de caractères 12](#_Toc153193805)

[6.17 Comment se connecter à une base de données en PHP ? Quelle est la classe native utilisée ? 12](#_Toc153193806)

[7 Symfony 13](#_Toc153193807)

[7.1 Qu’est-ce que Symfony ? 13](#_Toc153193808)

[7.2 Sur quel langage de programmation et design pattern repose Symfony ? 13](#_Toc153193809)

[7.3 Quelle est la dernière version en date de Symfony ? 13](#_Toc153193810)

[7.4 Qu’est-ce qu’un bundle ? 13](#_Toc153193811)

[7.5 Quel est le moteur de template utilisé par défaut dans Symfony ? 13](#_Toc153193812)

[7.6 Qu’est-ce qu’un ORM ? Quel est son utilité et comment s’appelle-t-il au sein de Symfony ? 13](#_Toc153193813)

[7.7 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 ? 13](#_Toc153193814)

[7.8 Que permet le bundle Maker au sein de Symfony ? 13](#_Toc153193815)

[7.9 Quel est le langage de requêtage exploité au sein d’un projet Symfony ? 13](#_Toc153193816)

[7.10 Quel est le composant qui garantit l’authentification et l’autorisation des utilisateurs ? 13](#_Toc153193817)

[8 Sécurité 13](#_Toc153193818)

[8.1 Qu’est-ce que l’injection SQL ? Comment s’en prémunir ? 13](#_Toc153193819)

[8.2 Qu’est-ce que la faille XSS ? Comment s’en prémunir ? 13](#_Toc153193820)

[8.3 Qu’est-ce que la faille CSRF ? Comment s’en prémunir ? 13](#_Toc153193821)

[8.4 Définir l’attaque par force brute et l’attaque par dictionnaire 13](#_Toc153193822)

[8.5 Existe-t-il d’autres failles de sécurité ? Citer celles-ci et expliquer simplement leur comportement 13](#_Toc153193823)

[8.6 A quoi servent l’authentification et l’autorisation dans un contexte d’application web ? 13](#_Toc153193824)

[8.7 Définir la notion de hachage d’un mot de passe et citer des algorithmes de hachage 13](#_Toc153193825)

[8.8 Qu’est-ce qu’une politique de mots de passe forts ? 13](#_Toc153193826)

[8.9 Qu’est-ce que l’hameçonnage ? 13](#_Toc153193827)

[8.10 Définir la « validation des entrées » 13](#_Toc153193828)

[9 RGPD 14](#_Toc153193829)

[9.1 Qu’est-ce que le RGPD ? 14](#_Toc153193830)

[9.2 Quel est son objectif principal ? 14](#_Toc153193831)

[9.3 Quelle est la date d’entrée en vigueur du RGPD ? 14](#_Toc153193832)

[9.4 Quelles sont les sanctions possibles en cas de non-respect du RGPD ? 14](#_Toc153193833)

[9.5 En France, quel est l’autorité administrative qui s’occupe de faire appliquer le RGPD ? 14](#_Toc153193834)

[9.6 Quel est le consentement valide selon le RPGD ? 14](#_Toc153193835)

[9.7 Qu’est-ce qu’une politique de confidentialité ? 14](#_Toc153193836)

[9.8 Quelle est la durée de conservation maximale des données personnelles selon le RGPD ? 14](#_Toc153193837)

[9.9 Quels sont les droits des utilisateurs selon le RGPD ? 14](#_Toc153193838)

[9.10 Qu’est-ce que le principe de minimisation des données selon le RGPD ? 14](#_Toc153193839)

[10 SEO 15](#_Toc153193840)

[10.1 Qu’est-ce que le SEO ? 15](#_Toc153193841)

[10.2 Quel est l’objectif principal du SEO ? 15](#_Toc153193842)

[10.3 Existe-t-il plusieurs types de référencement ? Lesquels ? 15](#_Toc153193843)

[10.4 Qu’est-ce que la densité de mots-clés en SEO ? 15](#_Toc153193844)

[10.5 Qu’est-ce qu’une balise « alt » ? 15](#_Toc153193845)

[10.6 Qu’est-ce que la balise « meta description » ? 15](#_Toc153193846)

[10.7 Qu’est-ce que le « nofollow » en SEO ? 15](#_Toc153193847)

[10.8 Quelle est l'importance du contenu de qualité pour le référencement d'un site web ? 15](#_Toc153193848)

[10.9 Pourquoi est-il important d'utiliser des balises de titre (h1, h2, h3, etc.) de manière structurée ? 15](#_Toc153193849)

[10.10 Quelle est la recommandation pour les URL d'un site web bien référencé ? 15](#_Toc153193850)

[11 Gestion de projets / DevOps 15](#_Toc153193851)

[11.1 Qu’est-ce que la gestion de projet ? 15](#_Toc153193852)

[11.2 Qu’est-ce qu’une méthode Agile de gestion de projet ? 15](#_Toc153193853)

[11.3 Expliquer la méthode MoSCoW en quelques lignes et citer ses avantages 15](#_Toc153193854)

[11.4 A quoi sert la méthodologie MVP ? Citer les caractéristiques clés 15](#_Toc153193855)

[11.5 Qu’est-ce que la planification itérative ? 15](#_Toc153193856)

[11.6 Citer 3 méthodes Agiles dans le cadre d’un projet informatique 15](#_Toc153193857)

[11.7 Qu’est-ce qu’une réunion de revue de projet ? 15](#_Toc153193858)

[11.8 Qu’est-ce qu’un livrable dans un projet ? 15](#_Toc153193859)

[11.9 Quels sont les 3 piliers SCRUM ? Définir chacun d’entre eux 15](#_Toc153193860)

[11.10 Qu’est-ce que le DevOps et quel est son objectif principal ? 15](#_Toc153193861)

[11.11 Qu’est-ce que l’intégration continue ? 15](#_Toc153193862)

[11.12 Qu’est-ce que Docker ? Et en quoi est-il utile dans le cadre du DevOps ? 15](#_Toc153193863)

[11.13 Qu’est-ce qu’un test unitaire ? 15](#_Toc153193864)

[11.14 Quelle est l'unité de code testée lors d'un test unitaire ? 15](#_Toc153193865)

[11.15 Quelles sont les caractéristiques d'un bon test unitaire ? 15](#_Toc153193866)

[11.16 Qu'est-ce qu'une assertion dans un test unitaire ? 15](#_Toc153193867)

[12 English 16](#_Toc153193868)

[12.1 What does JavaScript enable you to do on a website ? a. Add interactive behavior and dynamic content b. Define the layout and design of web pages c. Handle server-side operations 16](#_Toc153193869)

[12.2 Which programming language is primarily used for server-side web development ? a. PHP b. JavaScript c. HTML 16](#_Toc153193870)

[12.3 What is the purpose of a web browser ? a. To render and display web pages b. To execute serve-side code c. To manage databases 16](#_Toc153193871)

[12.4 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 b. GET submits data to a server, while POST retrieves data from a server c. GET and POST methods are interchangeable 16](#_Toc153193872)

[12.5 What is the purpose of version control systems (e.g., Git) in web development ? a. To track changes and manage collaborative development b. To optimize website loading speed c. To handle server-side scripting 16](#_Toc153193873)

[12.6 What is the purpose of a framework in web development ? a. To provide a structured environment for building web applications b. To handle network protocols and data transfer c. To create visual designs and layouts for websites 16](#_Toc153193874)

[12.7 What does NoSQL stand for ? a. Not Only SQL b. Non-Structured Query Language c. New Object-Oriented Language 16](#_Toc153193875)

[12.8 Which of the following is a characteristic of NoSQL databases ? a. Strict schema enforcement b. Support for complex transactions c. Scalability and flexible data model 16](#_Toc153193876)

# Général

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

Le script PHP s’exécute sur un serveur Web.

Il faut installer un serveur Apache

Un moteur PHP

Un serveur de base de données MySQL

Il est possible de les installer séparément ou d’utiliser des packages préconfigurer comme :

Laragon

WampServer

## Qu’est-ce qu’un algorithme ?

Un algorithme est une série d’instruction bien définie et organisée qui représente une méthode pour résoudre un problème.

Exemple : Une recette de cuisine.

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

Une variable est un espace de stockage dont la valeur peut être modifier.

En PHP on les déclare avec le symbole **$**

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

La portée d’une variable définit ou une variable peut être utilisée et accédé

Exemple :

* Local :

Les variables locales sont déclarées à l’intérieur d’une fonction ou d’un bloc de code spécifiques. Elles ne sont accessibles et utilisables qu’à l’intérieur de cette fonction ou de ce bloc de code ou elles sont déclarées.

* Global :

Les variables globales sont déclarées en dehors de toute fonction ou bloc spécifiques généralement dès le début.  
Elles sont accessible et utilisable dans l’ensemble du programme.

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

Une constante est un espace de stockage tous comme la variable leurs différence est que dans une constante la valeur reste fixe et ne peux être changé

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

Il existe 9 superglobales.

Une superglobale est une variable accessible partout $GLOBAl( $\_POST $\_GET, $\_SESSION ,….)

Exemple :

Si $\_POST et $\_GET sont utilisé lors de la soumission d’un formulaire d’inscription

$\_GET récupérera les donner inclus dans l’url de l’action après le symbole **?**

<form action= ‘index.php ?registerUser’ method=’post’

$\_POST : Utiliser pour récupérer les données envoyées via une méthode POST dans un formulaire HTML. Ces données ne sont pas visibles dans l’URL, mais envoyées de manière invisible. (pseudo email password\*2 )

## 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)

Les type de variable sont :

* string (chaine de caractère délimité par des guillemet simples ou doubles)

Exemple : $nom = "Mon Nom" ;

* int un nombre entier positif ou négatif

Exemple : $age = 34 ;

* bool une valeur booléenne, soit vrais(true) soit false(faux)

Exemple : $estConecter = false ;

* float un nombre décimal

Exemple : $pris = 10.99 ;

* Objet: Une instance de classe, ou une variable fait référence à un objet

Exemple : $objet = new MaClasse ; //création d'une instance d'un objet

* Array est un tableau contenant plusieurs éléments

Exemple : $tableauPays = [France, Argentine, Brésil, Angleterre] ;

* Null est une variable sans valeur assignée

Exemple : $varibleNul = null ;

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

Oui il existe plusieurs type de tableau en PHP:

* Les tableaux indexés

$array = ["salut","bonjours","aurevoir"]

* Les tableaux associatifs

$array =

[

"France" => "Paris",

"Allemagne" => "Berlin",

"Espagne" => "Madrid"

];

* Les tableaux multidimensionnels

$array =

[

"Arthur"

[

"Cp" => 68800,

"ville" => "Thann"

],

"Noémie"

[

"Cp" => 68800,

"ville" => "Thann"

]

]

## Quelles sont les différentes structures de contrôles qu’il existe en algorithmie ? Donner un exemple pour chacune d’entre elles

* Structure conditionnelle (if, else)

Utilisé pour exécuter du code selon certaine condition spécifiques

Si l'utilisateur est connecté alors

Instruction a exécuté si c'est vrai

Sinon

Instruction a exécuté si la condition est fausse

}

* Boucle (for, while, do-while)

Pour i de 1 à 10 faire

Résultat = 2\* i

Afficher résultat

Fin Pour

i = 1

Tant que i est inférieur ou égal a 10 faire

Résultat = 2\*i

Afficher résultat

i = i +1

Fin Tant Que

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

La fonction PHP qui retournera la longueur d'une chaine de caractère est :

Strlen();

Exemple : $chaine = "Bonjours" ;

Echo strlen($chaine) ;

Affichera 8

## Qu’est-ce qu’une session ? Quelle fonction permet de démarrer une session en PHP ? Donner un exemple d’utilisation en PHP

Une Session correspond à une façon de stocker les donné différentes pour chaque Utilisateur en utilisant un identifiant de session unique qui va généralement être envoyer au navigateur sous forme de cookie sous le nom PHPSESSID. On démarre une session grâce à : **Session\_start**.

Lors d’une connexion sur un forum l’user pour être connecté sera enregistrer en session

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

Un cookie est un petit fichier texte stocké sur l'ordinateur d'un utilisateur par le navigateur web. Il est utilisé pour enregistrer des informations spécifiques, telles que ces préférences, des données de session, des identifiants, etc.

Les cookies permettent au serveur web de conserver des informations sur l'utilisateur pour une utilisation ultérieur, même après la fermeture du navigateur.

En PHP, la fonction setcookie() est utilisée pour définir un cookie.

Exemple :

Setcookie("nom\_utilisateur"(nom du cookie), "Bernard"(valeur), time() + 3600 (durée de vie), "/"(accessibilité "/" pour tous le site) ;

Définie un cookie nommé "nom\_utilisateur" avec la valeur "Bernard" d'une durée de vie de 1 heure accessible dans tout le site

## Quelle est la différence entre les instructions « require » et « include » en PHP

Require: Arrêtera l'exécution du script a la ligne ou l'erreur sera rencontré et génèreras une erreur fatal.

Include: Il ne s'arrêteras pas et continuera d'exécuter la suite du script en générant un warning la ou l'erreur a été rencontré

## Comment effectuer une redirection en PHP ?

Pour effectuer une redirection en PHP on peut utiliser la fonction header().

Cette fonction envoie un en-tête http e redirection vers le navigateur du client

Exemple : Header("Location: nouvelle\_page.php")

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

Front-end: Le front-end est la partie visible par l’utilisateur il est constitué des pages du site web ou l'utilisateur va interagir.

Back-end : partie invisible qui gère toute la logique, la sécurité et les redirections d'après les actions de l'utilisateur.

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

Le contrôle de version est un système qui enregistre les modifications apportées à un ensemble de fichier au fil du temps. Il permet de garder un historique des modifications, de suivre l'évolution et de revenir en arrière si nécessaire.

Git est un système de contrôle de version populaire et largement utilisé. Conçu pour gérer tout type de projet petit, ou grand avec rapidité et efficacité. Il permet de créer des branches pour travailler sur des fonctionnalités séparément, de fusionner les modifications entre différentes branches et facilité la collaboration entre plusieurs développeurs.

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

Un CMS(Content Management System) est une plateforme logicielle permettant de créer, gérer et organiser facilement le contenu d'un site web sans nécessiter de compétences avancées en programmation.

Exemple de CMS: WordPress, Joomla.

## Différence chiffrement, encodage, et hachage ?

**L'encodage** permet uniquement de **traduire/convertir des données** pour les rendre plus facilement échangeables entre différents systèmes.

**Le chiffrement** permet de **sécuriser** des données qui ne seront déchiffrables que par des utilisateurs en possession d'une **clé secrète** (ou privée).

**Le hachage**, contrairement au chiffrement, est une technique **unidirectionnelle** : on ne peut pas « dé-hacher » des données hachées. Il est fréquemment utilisé pour les mots de passe avant leurs enregistrements en base de données.

# Front-end

## Définir HTML

Le Html HyperText Markup Language, est un langage dit de balises, il est utilisé pour créer et représenter le contenu d’une page web et sa structure. S’exécute côté client.

## Définir CSS

Cascading Style Sheets ou feuilles de style en cascade est le code utilisé pour mettre en forme une page Web. Couleurs et tailles du texte, taille des images, position des éléments, couleurs de fond. Il est associé directement avec les balises du HTML.

C’est en utilisant les sélecteur CSS qu’on met en forme les éléments HTML. On peut directement sélectionner une balise HTML (**div {}, .maClasse** {}, un id, **#monId** {}). S’exécute côté client.

## Définir Javascript

Langage de script léger utilisé pour certaines parties dynamiques de mon application.

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

JSON(JavaScript Object Notation) est un format de données léger et lisible par les humains. Il est largement utilisé pour l'échange de données structurées entre un serveur et un client web.

Contexte :

Echange de données entre serveur et client : Le JSON est utilisé dans les applications web pour échanger des données. Les requêtes Ajax sont souvent utilisées pour récupérer des données au format JSON depuis un serveur pour les afficher dynamiquement dans une page web.

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

Oui il est possible d'interpréter du javascript coté serveur grâce a Node.js Qui est une plateforme logicielle construite sur le moteur de Goolge Chrome. Node.js utilise un modèle asynchrone et évènementiel qui lui permet de gérer un grand nombre de connexion simultanées sans bloquer l'exécution du code. Node.js permet d'écrire des script JavScript coté serveur qui peuvent gérer les fichiers, manipuler la base de données, la création de serveur web et d'API.

## Qu’est-ce qu’un sélecteur CSS ?

Un sélecteur CSS est une expression qui permet de cibler les élément HTML auquel s'appliquera le style. Il en existe plusieurs dont :

* Les sélecteurs d'éléments : Ils ciblent des élément HTML spécifique en utilisant le nom de la balise (exemple : h1, p, span)
* Les sélecteurs de classe : Ils ciblent des éléments ayant une classe spécifiques définie dans le HTML en utilisant le préfix " **.** " (exemple. maClasse**)**
* **Les sélecteurs d'identifiant :** Il cible l'élément par son identifiant unique définis dans le HTML en utilisant le préfix " **#** "

## Quelle balise HTML permet de créer un lien hypertexte ?

La balise <a>

## Qu’est-ce qu’une requête AJAX ?

Les requête AJAX servent à envoyer et récupérer des données d'un serveur de manière asynchrone sans recharger la pages. AJAX permet de modifier partiellement la page affichée par le navigateur pour la mettre à jour. Il utilise le Javascript pour capturer les évènements du client et le XML ou le JSON pour échanger avec le serveur.

## Quel sélecteur CSS permet de sélectionner tous les éléments d’une classe spécifique ? D’un identifiant spécifique ?

Le sélecteur de classe est : **.maClasse**

Le sélecteur d'identifiant est : **#monId**

## Définir le responsive design

Adapter une page web en fonction de la taille de l’écran

## Qu’est-ce que le templating ?

Le templating est la création de modèles de structure de pages contenant des élément statiques(similaire a toute les pages) tels que l'en-tête, le pied de page, la barre de navigation, etc tous en laissant des espaces réservé pour le contenu dynamique qui sera injecté plus tard.

Il servent a maintenir une cohérences visuelle et structurelle sur l'ensemble du site ou de l'application.

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

Une fonction anonyme n'a pas de nom définie.

Elle sont souvent utilisé dans des contexte ou une fonction est nécessaire temporairement ou localement (qui ne nécessite pas d'être réutilisé ailleurs).

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

La méthode push(). Cette méthode permet d'ajouter un ou plusieurs élément a la fin d'un tableau existant.

Exemple : let tableau = [1,2,3,4]

tableau.push(5);

Cosole.log(tableau); affiche [1,2,3,4,5]

## Qu’est-ce qu’un « media query » ?

Un média querry (requête multimédia) est une fonctionnalité du langage CSS permettant de définir des règles de style conditionnelles basé sur la caractéristique du périphérique sur lequel la pages web est affiché. Ils sont souvent utilisés pour le responsive design.

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

Un pseudo élément en CSS est une méthode permettant de cibler et de styliser une partie spécifique d'un élément HTML.

Les pseudo-éléments commencent par deux points ::

Exemple: ::first-letter Permet de sélectionner la première lettre de l'élément.

::First-line Permet de sélectionner la première ligne de texte a l'intérieur de l'élément

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

Bootstrap est un framewrk CSS populaire et très utilisé pour créer des site web et des aplication web responsive. Ils offre une collection d'outils et de composant prédéfinis basé sur HTML, CSS et JavaScript.

Il permet de simplifier et d'accélérer le processus de création d'interface web moderne et responsive.

Autre framework : Foundation, Bulma

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

Les deux méthodes sont GET et POST

Dans la méthode GET les données du formulaire sont transmise par l'url en tan que chaine de requête (query string), les données sont visibles dans l'url, a n'utilisé que pour des données non sensibles.

Dans la méthode POST les données sont envoyées dans le corp de la requête http de manière invisible dans l'Url, utilisé généralement dans les formulaire au données sensible tel que les formulaire d'enregistrement d'utilisateur

# UX / UI

## Quelle est la différence entre UX Design et UI Design ?

Ux design (User Expérience Design) se concentre sur l'expérience global de l'utilisateur lorsqu'il interagit avec un produit ou un service. Cela englobe tous les aspects de l'interaction de l'utilisateur y compris ses émotion perception et réactions. Les concepteur UX s'efforcent d'apporter une expérience agréable a l'utilisateur sur le site ou l'application.

Ui Design (User Interface Design) se concentre sur la conception visuelle et interactive de l'interface utilisateur. Cela inclut la création d'éléments visuel tels que les boutons, les icones, les formulaires, la mise en pages, les couleurs, la typographie, etc. Les concepteur Ui s'efforcent à rendre l'interface belle, claire, facile à lire accessible et fonctionnelle.

## Qu’est-ce qu’un wireframe ?

Un wireframe est une représentation visuelle de la conception des pages de notre site Web (sans couleur, sans image). On utilise les wireframes au début d’un projet pour établir une structure de base d’une page avant l’ajout de son visuel. Réalisable sur papier, en HTML et CSS directement ou à l’aide de logiciel ou site spécialisé

## Qu’est-ce qu’un prototype ?

Un prototype est une représentation initiale, souvent interactive d'un produit ou d'une application avant sa production ou mise en œuvre

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

L a hiérarchie visuelle fait référence à l'organisation et à la disposition des éléments sur une interface pour guider l'attention de l'utilisateur, créer une structure claire et faciliter la compréhension des informations présenter.

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

L'accessibilité fait référence à la conception et au développement de produit numérique tels que des site web, des applications mobiles, des logiciels afin de les rendre utilisable et accessible à un large éventail d'utilisateur y compris ceux ayant des capacité physique ou cognitives différentes.

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

Un e grille de mise en pages est une structure composée de lignes horizontales et verticales qui organisent visuellement le contenu d'une pages web ou d'une interface utilisateur. Elle divise l'espace en colonnes, lignes ou modules pour aider les concepteurs à placer les éléments de manière cohérentes.

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

La notion d'affordances fait références aux caractéristiques ou aux propriété visuelle et fonctionnelles d'un élément qui suggèrent ou indiquent intuitivement son utilisation ou sa fonction.

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

Le concept Mobile First est une approche de conception ou on conçoit d'abord l'expérience utilisateur sur les appareils mobile, puis à étendre cette conception aux écran plus larges.

# Programmation orientée objet (POO)

## Donner une définition de la programmation orientée objet

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

## Qu’est-ce qu’un objet ?

## Définir la notion de propriété / attribut / méthode

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

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

## Qu’est-ce que l’encapsulation ?

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

## Définir l’opérateur de résolution de portée

## Définir une méthode / propriété statique

## Définir le polymorphisme en POO

## Définir une méthode / classe abstraite ?

## Définir le chaînage de méthodes

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

## Qu’est-ce qu’un « autoload » ?

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

## Qu’est-ce que la sérialisation en PHP

# Architecture

## 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

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

## Qu’est-ce que l’architecture MVC ?

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

## Quels sont les avantages de l’architecture MVC ?

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

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

# Modélisation / Base de données

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

## Quelles sont les 3 étapes principales de la méthode Merise ? a. Analyse, conception et réalisation b. Planification, exécution et contrôle c. Création, modification et suppression

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

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

## Donner la définition des mots suivants : a. Entité b. Relation c. Cardinalité d. Clé primaire / clé étrangère

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

## Qu’est-ce qu’une base de données ?

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

## Dans une base de données, les données sont stockées dans des \_\_\_. Celles-ci sont constituées de lignes appelées \_\_\_ et de colonnes appelées \_\_\_

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

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

## A quoi sert une vue dans une base de données ?

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

## Quelles sont les fonctions d’agrégation en SQL ?

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

## Quelles sont les clauses qui permettent de : a. Insérer un nouvel enregistrement dans une table b. Modifier un enregistrement dans une table c. Supprimer un enregistrement dans une table d. Supprimer la base de données e. Filtrer les résultats d’une requête SQL f. Trier les résultats d’une requête SELECT g. Regrouper les résultats d'une requête SELECT en fonction d'une colonne spécifique h. Concaténer 2 chaînes de caractères

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

# Symfony

## Qu’est-ce que Symfony ?

## Sur quel langage de programmation et design pattern repose Symfony ?

## Quelle est la dernière version en date de Symfony ?

## Qu’est-ce qu’un bundle ?

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

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

## 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 ?

## Que permet le bundle Maker au sein de Symfony ?

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

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

# Sécurité

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

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

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

## Définir l’attaque par force brute et l’attaque par dictionnaire

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

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

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

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

## Qu’est-ce que l’hameçonnage ?

## Définir la « validation des entrées »

# RGPD

## Qu’est-ce que le RGPD ?

## Quel est son objectif principal ?

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

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

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

## Quel est le consentement valide selon le RPGD ?

## Qu’est-ce qu’une politique de confidentialité ?

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

## Quels sont les droits des utilisateurs selon le RGPD ?

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

# SEO

## Qu’est-ce que le SEO ?

## Quel est l’objectif principal du SEO ?

## Existe-t-il plusieurs types de référencement ? Lesquels ?

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

## Qu’est-ce qu’une balise « alt » ?

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

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

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

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

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

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

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

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

# Gestion de projets / DevOps

## Qu’est-ce que la gestion de projet ?

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

## Expliquer la méthode MoSCoW en quelques lignes et citer ses avantages

## A quoi sert la méthodologie MVP ? Citer les caractéristiques clés

## Qu’est-ce que la planification itérative ?

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

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

## Qu’est-ce qu’un livrable dans un projet ?

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

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

## Qu’est-ce que l’intégration continue ?

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

## Qu’est-ce qu’un test unitaire ?

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

## Quelles sont les caractéristiques d'un bon test unitaire ?

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

# English

## What does JavaScript enable you to do on a website ? a. Add interactive behavior and dynamic content b. Define the layout and design of web pages c. Handle server-side operations

## Which programming language is primarily used for server-side web development ? a. PHP b. JavaScript c. HTML

## What is the purpose of a web browser ? a. To render and display web pages b. To execute serve-side code c. To manage databases

## 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 b. GET submits data to a server, while POST retrieves data from a server c. GET and POST methods are interchangeable

## What is the purpose of version control systems (e.g., Git) in web development ? a. To track changes and manage collaborative development b. To optimize website loading speed c. To handle server-side scripting

## What is the purpose of a framework in web development ? a. To provide a structured environment for building web applications b. To handle network protocols and data transfer c. To create visual designs and layouts for websites

## What does NoSQL stand for ? a. Not Only SQL b. Non-Structured Query Language c. New Object-Oriented Language

## Which of the following is a characteristic of NoSQL databases ? a. Strict schema enforcement b. Support for complex transactions c. Scalability and flexible data model