



Editions ENI

Référentiel de compétences

## Description du métier, de l'activité ou de la situation professionnelle à partir duquel le dispositif visant la certification est initié :

Le langage PHP a été spécifiquement et uniquement conçu pour le développement web, offrant ainsi un ensemble riche de fonctionnalités et de bibliothèques dédiées à la création de sites web dynamiques. Sa grande communauté de développeurs contribue à un écosystème étendu d'outils, frameworks et bibliothèques qui facilitent et structurent le développement web. Des CMS populaires tels que WordPress, Joomla! et Drupal sont tous basés sur PHP.

La certification « Langage PHP – Développer des sites web » s'adresse aux développeurs et webmasters qui souhaitent maîtriser le langage PHP pour développer un site web, étendre les fonctionnalités d'un CMS ou personnaliser un site web réalisé avec un CMS.

En fonction du score obtenu à l'épreuve, les candidats valident la certification :

- **De 0 à 699/1000** : le candidat n'est pas certifié
- **De 700 à 1000/1000** : le candidat est certifié

En dessous de 700 points, la certification n'est pas validée.

Référentiel		
Référentiel de compétences	Référentiel de certification	
	Modalité(s)	Critères
<p><b>C1 - Créer des pages web simples</b> en recourant aux éléments fondamentaux d'algorithmie du langage PHP pour les rendre affichables grâce à un navigateur.</p> <p>Applications métiers :</p> <ul style="list-style-type: none"> <li>- Développeur : Ecrire des fonctions permettant de transformer des données métiers pour les publier sur un site Web</li> <li>- Webmaster : Créer une vitrine Web simple pour présenter l'activité d'une entreprise.</li> </ul>	<p>1 - Cas pratiques :</p> <ul style="list-style-type: none"> <li>• Le candidat écrit des portions de code PHP sous forme de scripts.</li> <li>• Le candidat complète des trous dans une structure de code existante.</li> </ul> <p>2 - Questionnaire de type QCM/QCU à visée professionnelle</p>	<p><b>Cr1</b> - La configuration du moteur PHP est modifiée pour agir sur l'exécution des scripts.</p> <p><b>Cr2</b> - Un fichier de script PHP simple est correctement écrit (balises <code>&lt;?php</code> et <code>?&gt;</code>, intégration avec du code HTML, règles de nommage respectées, fonction <code>echo()</code>).</p> <p><b>Cr3</b> - Les variables et constantes sont correctement déclarées et initialisées, une donnée est transtypée d'un type vers un autre type compatible et les fonctions <code>isset()</code> et <code>empty()</code> sont correctement utilisées.</p> <p><b>Cr4</b> - Les opérateurs de langage (arithmétiques, d'incrémentation et de décrémentation, d'affectation et d'affectation combinés, de comparaison, logiques) sont correctement mis en œuvre.</p> <p><b>Cr5</b> - Les mécanismes d'inclusion de fichiers sont utilisés pour inclure des fragments HTML ou des définitions PHP et les fonctions <code>require()</code>, <code>require_once()</code>, <code>include()</code> et <code>include_once()</code> sont différenciées d'un point de vue de leur comportement et de leur usage.</p> <p><b>Cr6</b> - Des instructions sont exécutées selon une ou plusieurs conditions (structure <code>if - else if - else</code>, opérateur ternaire, structure <code>witch</code>, expression <code>match</code>).</p> <p><b>Cr7</b> - Des itérations sont mises en œuvre sur des ensembles de données (<code>for</code>, <code>foreach</code>, <code>while</code>, <code>do - while</code>, mot-clé <code>break</code> et <code>continue</code>)</p> <p><b>Cr8</b> - Des tableaux (associatif, multidimensionnel...) sont déclarés et initialisés en utilisant la fonction <code>array()</code> ou les <code>[]</code> et les éléments d'un tableau sont accédés en utilisant la notation à crochets.</p> <p><b>Cr9</b> - Des fonctions et fonctions anonymes sont correctement déclarées et utilisées avec leurs paramètres.</p>

**C2 - Développer une application** en utilisant les mécanismes de programmation orientée objet de PHP pour maîtriser la complexité d'une application web.

Applications métier :

- Développeur : Concevoir et implémenter un progiciel de type CRM ou de gestion des stocks sur la base d'une analyse fonctionnelle métier.
- Webmaster : Structurer les produits proposés à la vente sur un site de commerce électronique.

- Cr1** - Une classe est créée et déclarée avec le mot-clé class.
- Cr2** - Les attributs sont déclarés dans le corps de la classe et correctement typés.
- Cr3** - Les variables d'instances, de classes et locales sont correctement utilisées (\$this, static).
- Cr4** - Une méthode `_construct()` est déclarée et les attributs sont initialisés avec les paramètres dans l'implémentation du constructeur.
- Cr5** - Des méthodes sont créées et implémentées.
- Cr6** - Un objet est instancié (opérateur `new`) et manipulé.
- Cr7** - L'encapsulation est mise en œuvre (mot-clé `private`, mot-clé `public`, `get`, `set`).
- Cr8** - L'héritage est mis en œuvre (mot-clé `extends`, instruction `parent::`, mot-clé `protected`).
- Cr9** - Le polymorphisme est mis en œuvre (opérateur de transtypage, mot-clé `parent`, opérateur `instanceof`).
- Cr10** - Les classes sont organisées dans des espaces de nom (mot-clé `namespace`, `use`).
- Cr11** - Des classes abstraites sont déclarées (mot-clé `abstract`) et utilisées.
- Cr12** - Des interfaces sont déclarées et utilisées (mot-clé `interface`, `implements`).
- Cr13** - La gestion des erreurs avec les exceptions est mise en œuvre (mot-clé `throw`, `structure try - catch - finally`).

**C3 - Exploiter les fonctionnalités de la bibliothèque PHP** en utilisant les fonctions sur les tableaux, les chaînes de caractères et les dates pour améliorer la gestion et la manipulation des données.

Applications métier :

- Développeur : Gérer des tableaux de données représentant des grilles tarifaires pour calculer des

- Cr1** - Les fonctions élémentaires de manipulation des chaînes de caractères sont utilisées.
- Cr2** - La syntaxe des expressions régulières est utilisée afin de créer des modèles de comparaison ou d'extraction de chaînes de caractères et les fonctions PHP exploitant les expressions régulières sont mises en œuvre (`preg_match()`, `preg_filter()` et `preg_replace()`).
- Cr3** - Les fonctions élémentaires de création et formatage de date sont utilisées (`strtotime()` et `date()`), un timestamp est créé à partir d'une date et inversement (`strtotime()`, `date()`) et les préférences locales sont ajustées afin de générer des données temporelles dans le format souhaité.

<p>prix remisés sur un site marchand.</p> <p>- Webmaster : Mettre en forme des dates pour un affichage selon le pays sur un site de publication éditorial.</p>		<p><b>Cr4</b> - Les tableaux sont manipulés avec PHP : nombre d'éléments comptés (count()), tris appliqués (sort(), rsort(), asort(), arsort(), ksort(), krsort()), valeurs recherchées ou remplacées (array_search(), array_replace())...</p>
<p><b>C4 - Mettre en œuvre les concepts spécifiques du Web en PHP</b> avec les formulaires, la gestion du protocole HTTP et la gestion des sessions pour construire une application web complète.</p> <p><u>Applications métier :</u></p> <p>- Développeur : Créer les différents formulaires permettant de confirmer une commande client dans une application métier.</p> <p>- Webmaster : Créer un système de gestion du panier d'achat sur un site de commerce électronique.</p>		<p><b>Cr1</b> - Des données transmises sont récupérées du protocole HTTP avec la méthode GET.</p> <p><b>Cr2</b> - Un formulaire est créé avec différents types de champs en HTML, l'URL d'envoi du formulaire ainsi que la méthode de transfert sont spécifiées, les données postées par le formulaire sont testées, filtrées et récupérées du tableau \$_POST et une redirection est effectuée après le traitement du formulaire.</p> <p><b>Cr3</b> - Le suivi utilisateur est mis en oeuvre avec les sessions (session_start(), tableau \$_SESSION...).</p>
<p><b>C5 - Mettre en oeuvre les techniques de manipulation de fichiers et répertoires</b> grâce aux fonctions de la bibliothèque PHP pour implémenter les fonctionnalités d'envoi de fichiers depuis un formulaire.</p> <p>Applications métier</p>		<p><b>Cr1</b> - L'existence d'éléments du système de fichiers du serveur est vérifiée avec file_exists() et les fonctions isXXX(), et le système de fichiers est manipulé avec les fonctions équivalentes aux commandes du monde Unix telles que chown(), chmod(), copy(), delete(), rename(), mkdir(), et rmdir().</p> <p><b>Cr2</b> - Les fonctions de base pour l'ouverture, l'écriture et la lecture, ainsi que la fermeture des fichiers sont utilisées (fopen(), fread(), fwrite() et fclose()) et les fonctions file_get_contents() et file_put_contents() sont utilisées en alternative aux fonctions précédentes.</p>

<p>- Développeur : Stocker et organiser sur le serveur, les bons de commande client transmis sous forme de fichier PDF.</p> <p>- Webmaster : Mettre en oeuvre un système d'envoi de photos pour les associer à un article dans un site de publication éditorial.</p>		<p><b>Cr3</b> - Le formulaire est correctement créé avec l'attribut enctype sur la balise form ainsi qu'un champ input de type file, les limites de taille d'envoi sont définies, les informations du tableau \$_FILES sont exploitées pour tester le bon transfert du fichier et vérifier qu'il ne pose aucun problème de sécurité (les erreurs sont gérées le cas échéant) et la fonction move_upload_file() est utilisée pour stocker les fichiers transmis dans un emplacement du système de fichiers.</p>
<p><b>C6 - Connecter une application PHP à une base de données relationnelle</b> en utilisant les classes et interfaces de l'API PHP Data Objects (PDO) pour implémenter des fonctionnalités de persistance de données.</p> <p><u>Applications métier :</u></p> <p>- Développeur : Enregistrer et mettre à jour les commandes clientes d'un CRM dans une base de données relationnelle.</p> <p>- Webmaster : Lire des articles depuis une base de données afin de les publier sur un blog.</p>		<p><b>Cr1</b> - Un pilote PDO est activé dans la configuration PHP pour la base de données à utiliser, l'URL de connexion est définie et la classe PDO est utilisée pour créer la connexion.</p> <p><b>Cr2</b> - Une requête simple est exécutée.</p> <p><b>Cr3</b> - Des requêtes SQL précompilées sont exécutées.</p> <p><b>Cr4</b> - Les méthodes fetch(), fetchAll() et fetchObject() sont utilisées pour parcourir un jeu d'enregistrement et traiter les résultats des requêtes SQL SELECT.</p> <p><b>Cr5</b> - Les exceptions de type PDOException sont gérées dans les instructions JDBC et des exceptions applicatives sont déclenchées.</p> <p><b>Cr6</b> - Les transactions sont démarrées (beginTransaction()), validées en fin de traitement (commit()) et annulées en cas d'exception (rollback()).</p>