

Compétences transversales développées dans tous les projets de la formation CDA

Les projets réalisés dans le cadre de la formation "Concepteur Développeur d'Applications" permettent à chaque stagiaire de mobiliser l'ensemble des compétences professionnelles attendues en entreprise. Ces compétences sont transversales et doivent être présentes dans tous les projets proposés, quel que soit leur thème.

Sécurité et conformité

- Mise en place d'un système d'authentification sécurisé (par sessions ou via un système de jetons JWT).
- Hachage sécurisé des mots de passe (utilisation d'algorithmes tels que bcrypt ou Argon2i).
- Gestion fine des droits d'accès et des autorisations selon les rôles utilisateurs (utilisation de systèmes de contrôle d'accès tels que les firewalls Symfony ou les voters).
- Intégration d'outils de test de sécurité comme OWASP ZAP pour détecter les failles potentielles.
- Application des recommandations de sécurité issues des référentiels de l'ANSSI.
- Respect des obligations imposées par le RGPD (mentions légales, consentement, droit à l'oubli, anonymisation, portabilité des données).
- Intégration d'un module de paiement sécurisé (Stripe ou équivalent) avec gestion des retours d'échec ou succès.

Web design, accessibilité et qualité du code HTML/CSS

- Réalisation de maquettes et prototypes fonctionnels à l'aide d'outils professionnels comme Figma.
- Mise en œuvre de l'accessibilité numérique conformément au référentiel RGAA.
- Développement d'interfaces web responsives (compatibilité mobile, tablette, desktop).
- Respect des normes HTML5/CSS3 et validation auprès des outils W3C.
- Mise en place des éléments techniques contribuant au référencement naturel (balises sémantiques, balises meta, performance des pages).

Architecture logicielle et structuration du code

- Organisation du projet en couches (modèle MVC, services injectables, séparation des responsabilités).
- Mise en place de services personnalisés dans Symfony ou React.
- Gestion de la persistance des données avec MariaDB (relationnelle) ou MongoDB (documentaire).
- Utilisation d'un ORM comme Doctrine pour structurer l'accès aux bases relationnelles.

- Création d'API RESTful robustes et bien documentées (documentation Swagger/OpenAPI).

Qualité, tests et maintenabilité

- Écriture de tests unitaires automatisés (PHPUnit pour PHP, Jest pour JavaScript).
- Développement de composants réutilisables (dans Symfony ou React).
- Application des standards de codage (PSR-1, PSR-4, PSR-12 pour PHP).
- Mise en œuvre des bonnes pratiques de développement (code lisible, réutilisable, structuré).

Déploiement et environnement de travail

- Conteneurisation complète des projets avec Docker et docker-compose.
- Utilisation d'un gestionnaire de versions (Git) et gestion de branches avec GitFlow.
- Mise en place de chaînes d'intégration continue (CI/CD) via GitHub Actions ou GitLab CI.
- Création de scripts de déploiement ou utilisation de plateformes PaaS pour la mise en production.
- Installation et configuration d'un environnement de travail professionnel sous Linux Mint.

Compétences professionnelles transversales

- Rédaction de documentations techniques claires et structurées (en Markdown ou HTML).
- Préparation d'un dossier professionnel complet décrivant le projet, les choix techniques, les étapes de réalisation.
- Présentation orale du projet (soutenance) incluant une démonstration interactive.
- Développement de l'autonomie professionnelle par la mise en œuvre d'une veille technologique active et la consultation de la documentation officielle des outils utilisés.