

EXPÉRIENCES

1. Ingénieur Développement des Automatisations – Tutorax, Canada

Tâches réalisées :

- **Optimisation des processus d'archivage numérique** : Analyse et amélioration des processus d'archivage pour réduire les inefficacités et automatiser les tâches manuelles.
- **Organisation et centralisation des fichiers sensibles** : Mise en place d'un système sécurisé pour centraliser les documents confidentiels.

Outils utilisés et leur fonctionnement :

- **Systèmes de gestion documentaire (DMS)** : Exemples : Alfresco, SharePoint.
 - **Fonctionnement** : Ces outils permettent de stocker, classer et retrouver rapidement des documents numériques grâce à des métadonnées.
 - **Pourquoi utilisés** : Assurent une gestion efficace, la sécurité des données et la conformité avec les normes.
 - **Outils d'automatisation (Python, scripts Shell)** :
 - **Fonctionnement** : Automatisent des tâches répétitives telles que l'organisation de fichiers ou l'ajout de métadonnées.
 - **Pourquoi utilisés** : Économisent du temps, réduisent les erreurs humaines et optimisent les flux de travail.
-

2. Ingénieur Développement FullStack – Webloo, Californie

Tâches réalisées :

- **Développement de fonctionnalités de paiement via l'application** : Intégration de passerelles de paiement pour faciliter les transactions en ligne.
- **Développement de pages web adaptatives** : Création de pages web s'adaptant à tous types d'écrans (ordinateurs, tablettes, mobiles).

Outils utilisés et leur fonctionnement :

- **Frontend : React.js, HTML/CSS** :
 - **Fonctionnement** : React.js permet de développer des interfaces dynamiques grâce à des composants réutilisables, tandis que HTML/CSS définissent la structure et le style des pages.
 - **Pourquoi utilisés** : Pour créer des interfaces interactives et conviviales.
- **Backend : Node.js, Express.js** :
 - **Fonctionnement** : Node.js exécute du JavaScript côté serveur, et Express.js structure la logique backend pour gérer les API et les requêtes.

- **Pourquoi utilisés** : Pour traiter les paiements, gérer les données des utilisateurs et assurer la communication entre le frontend et les bases de données.
 - **API de passerelles de paiement (Stripe, PayPal)** :
 - **Fonctionnement** : Ces API permettent de traiter les transactions de manière sécurisée.
 - **Pourquoi utilisées** : Pour simplifier et sécuriser les paiements en ligne.
-

3. Archiviste - système de gestion documentaire – InteraktAgency, Rabat

Tâches réalisées :

- **Audit des besoins documentaires** : Analyse des besoins pour concevoir une stratégie efficace de gestion documentaire.
- **Implémentation d'un système d'archivage électronique** : Mise en place de solutions pour gérer et conserver les documents numériques.

Outils utilisés et leur fonctionnement :

- **Systèmes de gestion documentaire électronique (EDMS)** : Exemples : OpenText, M-Files.
 - **Fonctionnement** : Ces systèmes permettent de numériser les documents, de les enrichir avec des métadonnées et de les organiser pour une récupération rapide.
 - **Pourquoi utilisés** : Pour assurer une préservation à long terme et une accessibilité efficace.
 - **Outils de classification et d'indexation** : Développement de taxonomies pour catégoriser les fichiers.
 - **Fonctionnement** : Permet d'organiser les fichiers par mots-clés ou catégories spécifiques.
 - **Pourquoi utilisés** : Facilite la recherche et réduit le temps de récupération des documents.
-

4. Stage Étude : Développeur Web – Mobile – AlbaridBank, Rabat

Tâches réalisées :

- **Intégration de services backend** : Développement et intégration d'APIs pour collecter et traiter des données dans des applications.

Outils utilisés et leur fonctionnement :

- **Outils backend** : Python, Java, Node.js :

- **Fonctionnement** : Utilisés pour écrire des scripts et développer des fonctionnalités côté serveur, comme l'authentification utilisateur ou la collecte de données.
- **Pourquoi utilisés** : Garantissent une communication fluide entre l'interface utilisateur et la base de données.
- **Bases de données : SQL, NoSQL (MySQL, MongoDB) :**
 - **Fonctionnement** : SQL gère les données structurées avec des requêtes complexes, tandis que NoSQL est adapté aux données semi-structurées ou non structurées.
 - **Pourquoi utilisés** : Permettent une gestion efficace des données en fonction des besoins du projet.