

# Cahier des charges - OpportuniStage

## 1. Introduction

### 1.1 Titre du projet

Développement de l'application web *OpportuniStage* pour la gestion des stages.

### 1.2 Présentation générale

Le projet *OpportuniStage* vise à créer une application web de gestion de stages permettant aux étudiants de postuler à des offres, aux entreprises de soumettre des offres de stages, et à l'administrateur de gérer les utilisateurs et les offres. L'application inclura une **base de données MySQL** et sera réalisée avec **HTML, CSS, JavaScript, PHP et MySQL**.

#### ➤ Contexte

Ce projet répond à la nécessité de simplifier la gestion des stages pour les étudiants, les entreprises et les établissements scolaires. Une solution centralisée et numérique peut remplacer les systèmes manuels existants.

#### ➤ Problématique à résoudre

Comment développer une plateforme simple, intuitive et sécurisée qui facilite les interactions entre étudiants et entreprises, tout en assurant un suivi et une gestion efficaces des candidatures de stages ?

### 1.3 Objectifs du projet

#### ➤ Objectif général :

Développer une plateforme web permettant aux étudiants de rechercher et postuler à des offres de stages, aux entreprises de publier leurs offres, et à l'administrateur de gérer ces processus.

#### ➤ Objectifs spécifiques :

- Créer un système d'authentification sécurisé pour les trois types d'utilisateurs (étudiant, entreprise, administrateur).
- Permettre aux entreprises de publier et supprimer leurs offres de stages.
- Offrir aux étudiants la possibilité de postuler et suivre l'état de leurs candidatures.
- Fournir une interface d'administration pour la gestion des utilisateurs et des offres.
- Intégrer une base de données MySQL pour stocker les informations des utilisateurs, des offres, et des candidatures.

## 1.4 Public cible et utilisateurs finaux

- **Étudiants** : Recherche et postulation à des stages.
- **Entreprises** : Publication et gestion des offres de stages.
- **Administrateurs** : Gestion des utilisateurs, des offres et des candidatures.

# 2. Analyse des besoins

## 2.1 Besoins fonctionnels

Le système devra permettre :

- **Authentification des utilisateurs** (étudiants, entreprises, administrateur).
- **Gestion des offres de stages** (création, suppression).
- **Postulation à des offres de stages** par les étudiants.
- **Consultation des candidatures** et gestion des utilisateurs par l'administrateur.
- **Tableau de bord personnalisé** pour chaque type d'utilisateur (étudiant, entreprise, admin).
- **Notifications** concernant l'état des candidatures.

## 2.2 Besoins non fonctionnels

- **Performance** :
  - La plateforme doit être réactive avec un temps de réponse inférieur à 200 ms pour les actions courantes.
  - Les processus de sauvegarde des candidatures doivent être rapides et sans impact visible sur l'utilisateur.
- **Compatibilité** :
  - L'application doit être accessible sur les principaux navigateurs web (Chrome, Firefox, Edge, Safari).
- **Accessibilité** :
  - Interface conviviale avec des éléments visuels adaptés aux utilisateurs non techniques.

# 3. Description des fonctionnalités

## 3.1 Authentification

- **Acteur** : Étudiant, Entreprise, Administrateur.
- **Action** : Se connecter avec un email et mot de passe.
- **Résultat attendu** : Accès à l'interface utilisateur spécifique selon le rôle.

### 3.2 Gestion des offres de stages

- **Acteur** : Entreprise
- **Action** : Créer ou supprimer une offre de stage.
- **Résultat attendu** : Offre visible sur la section offre des étudiants.

### 3.3 Postulation à une offre

- **Acteur** : Étudiant
- **Action** : Postuler à une offre de stage.
- **Résultat attendu** : Candidature enregistrée et envoyée à l'entreprise.

### 3.4 Gestion des utilisateurs et des offres

- **Acteur** : Administrateur
- **Action** : Voir ou supprimer le profil des utilisateurs et des offres.
- **Résultat attendu** : Mise à jour de la base de données et modification de l'affichage.

## 4. Technologies et outils

### 4.1 Langages de programmation

- **Front-end** : HTML5, CSS3, JavaScript (pour une interface dynamique et interactive)
- **Back-end** : PHP (pour la gestion des requêtes côté serveur)
- **Base de données** : MySQL (pour stocker les utilisateurs, les offres, les candidatures)

## 5. Contraintes et limites

### 5.1 Contraintes techniques

- **Temps de réponse** : Les actions utilisateurs doivent être traitées rapidement, avec un délai de réponse inférieur à 200 ms.
- **Stockage** : La base de données doit contenir des informations sur les utilisateurs, les offres et les candidatures, nécessitant un espace minimal.

### 5.2 Contraintes de développement

- **Temps limité** : Le projet doit être terminé d'ici fin juin 2025.
- **Technologies imposées** : Le projet sera développé en utilisant HTML, CSS, JavaScript, PHP, et MySQL.

## 6. Planification et livrables

### 6.1 Planning

- **Phase 1** : Authentification et gestion des utilisateurs (1 semaine)
- **Phase 2** : Gestion des offres de stages pour les entreprises (1 semaine)
- **Phase 3** : Postulation et gestion des candidatures (1 semaine)
- **Phase 4** : Interface d'administration et finalisation (1 semaine)

### 6.2 Livrables attendus

- **Version fonctionnelle** de l'application avec une interface utilisateur testée et fonctionnelle.
- **Code source documenté.**