# Cahier des Charges : Projet Pluridisciplinaire sur les Jeux Olympiques

## 1. Introduction

Le projet proposé s'inscrit dans le cadre d'une semaine "piscine", visant à favoriser l'autonomie, la montée en compétences, le challenge et le dépassement de soi des étudiants. Chaque groupe d'étudiants travaillera sur un des sujets proposés, avec une division des tâches entre développeurs et administrateurs systèmes et réseaux. Ce projet permettra de mettre en pratique les compétences en développement web, administration de systèmes et réseaux, gestion de projet, et collaboration en équipe.

## 2. Objectifs du Projet

- Développer une application web fonctionnelle avec des fonctionnalités CRUD (Create, Read, Update, Delete) pour un objet en lien avec les Jeux Olympiques.  
- Mettre en place une infrastructure serveur pour héberger l'application.  
- Assurer la sécurité et la performance de l'application.  
- Collaborer efficacement en utilisant des outils de gestion de projet et de versionnement de code.

## 3. Fonctionnalités Requises

1. Page d'Accueil  
 - Accueil standard.  
  
2. Page d'Inscription  
 - Formulaire de création de compte avec validation des données (email, mot de passe, etc.).

3. Page de Connexion  
 - Formulaire de connexion pour les utilisateurs existants.  
  
4. Page de Profil  
 - Affichage et modification des informations personnelles de l'utilisateur.  
  
5. Gestion CRUD  
 - Ajouter, consulter, modifier et supprimer un objet spécifique en fonction du projet choisi (événement, athlète, médaille, site olympique, volontaire, billet, sponsor, record, conférence de presse, préparation olympique).

## 4. Structure du Projet

1. Frontend  
 - Technologies : HTML, CSS, Bootstrap, JavaScript.  
 - Responsables : Développeurs Frontend.  
 - Tâches : Conception et réalisation des interfaces utilisateur.  
  
2. Backend  
 - Technologies : PHP.  
 - Responsables : Développeurs Backend.  
 - Tâches : Développement des fonctionnalités serveur, gestion des données.  
  
3. Base de Données  
 - Technologies : MySQL ou MariaDB.  
 - Responsables : Développeurs Backend.  
 - Tâches : Conception et gestion de la base de données.  
  
4. Déploiement  
 - Technologies : Linux (Ubuntu), Apache/Nginx, Certbot pour HTTPS.  
 - Responsables : Administrateurs Systèmes et Réseaux.  
 - Tâches : Configuration du serveur, déploiement de l'application, gestion de la sécurité et des backups.  
  
5. Collaboration et Gestion de Projet  
 - Outils : Git/GitHub, Trello.  
 - Responsables : Tous les membres de l'équipe.  
 - Tâches : Suivi des tâches, versionnement du code, réunions de coordination.

## 5. Contraintes et Exigences

1. Autonomie et Montée en Compétences  
 - Les étudiants doivent travailler de manière autonome et pro-active, recherchant des solutions par eux-mêmes et s'aidant mutuellement au sein de l'équipe.  
  
2. Sécurité  
 - Utilisation de HTTPS pour sécuriser les échanges de données.  
 - Protection contre les injections SQL et les attaques XSS.  
  
3. Gestion de Projet  
 - Utilisation d'un outil de gestion de projet (Trello, Planner, Jira etc.) pour planifier et suivre les tâches.  
 - Réunions journalières pour faire le point sur l'avancement et ajuster les priorités.  
  
4. Versionning du Code  
 - Utilisation de Git/GitHub pour le contrôle de version.  
 - Respect des bonnes pratiques de commit et de gestion des branches.  
  
5. Déploiement  
 - Déploiement sur un serveur Linux avec Apache ou Nginx.  
 - Mise en place d'un système de backup régulier de la base de données.  
  
6. Collaboration  
 - Communication régulière entre les développeurs et les administrateurs systèmes pour assurer l'intégration et le déploiement fluide de l'application.  
 - Partage de la documentation et des ressources techniques.

## 6. Évaluation

L'évaluation du projet se fera sur les critères suivants :  
- Fonctionnalité : L'application répond-elle aux spécifications fonctionnelles ?  
- Qualité du Code : Le code est-il propre, bien structuré et commenté ?  
- Collaboration : Le travail d'équipe a-t-il été efficace et bien coordonné ?  
- Déploiement : L'application est-elle correctement déployée et sécurisée ?  
- Présentation : La présentation finale est-elle claire et complète ?

## 7. Livrables

- Code source de l'application sur GitHub.  
- Documentation technique et utilisateur.  
- Planification du projet sur Trello, Planner.  
- Présentation finale du projet.

## 8. Planning à venir