



Jeux Olympiques



CAHIER DES CHARGES TECHNIQUES

Projet JO 2024



PARIS

Ville candidate
Jeux Olympiques de 2024



Dossier rédiger par Clément LIN



Jeux Olympiques



Sommaire

1.1. Présentation du projet.....	2
1.2. Date de rendu du projet.....	2
3.1. Ressources matérielles.....	3
3.2. Ressources logicielles.....	3
5.1. Le front-end.....	3
5.1.1. Wireframes.....	3
5.1.2. Maquettes.....	3
5.1.3. Arborescences.....	4
5.2. Le back-end.....	4
5.2.1. Diagramme de cas d'utilisation.....	4
5.2.2. Diagramme d'activités.....	4
5.2.3. Modèles Conceptuel de Données (MCD).....	4
5.2.4. Modèle Logique de Données (MLD).....	4
5.2.5. Modèle Physique de Données (MPD).....	4
6.1. Langages de développement Web.....	4
6.2. Base de données.....	4
7.1. Login.....	5
7.2. Cryptage des mots de passe.....	5
7.3. Protection des pages administrateurs.....	5
7.4. Protection contre les attaques XSS (Cross-Site Scripting).....	5
7.5. Protection contre les injections SQL.....	5



Jeux Olympiques



1. Contexte du projet

1.1. Présentation du projet

Votre agence web a été sélectionnée par le comité d'organisation des jeux olympiques de Paris 2024 pour développer une application web permettant aux organisateurs, aux médias et aux spectateurs de consulter des informations sur les sports, les calendriers des épreuves et les résultats des JO 2024.

Votre équipe et vous-même avez pour mission de proposer une solution qui répondra à la demande du client.

1.2. Date de rendu du projet

Le projet doit être rendu au plus tard le 22 mars 2024.

2. Besoins fonctionnels

Le site web devra avoir une partie accessible au public et une partie privée permettant de gérer les données.

Les données seront stockées dans une base de données relationnelle pour faciliter la gestion et la mise à jour des informations. Ces données peuvent être gérées directement via le site web à travers un espace administrateur.

3. Ressources nécessaires à la réalisation du projet

3.1. Ressources matérielles

Matériel nécessaire à la réalisation du projet, un **Unité Central - Clavier - Écran et Souris** où **PC** tout court.



Jeux Olympiques



3.2. Ressources logicielles

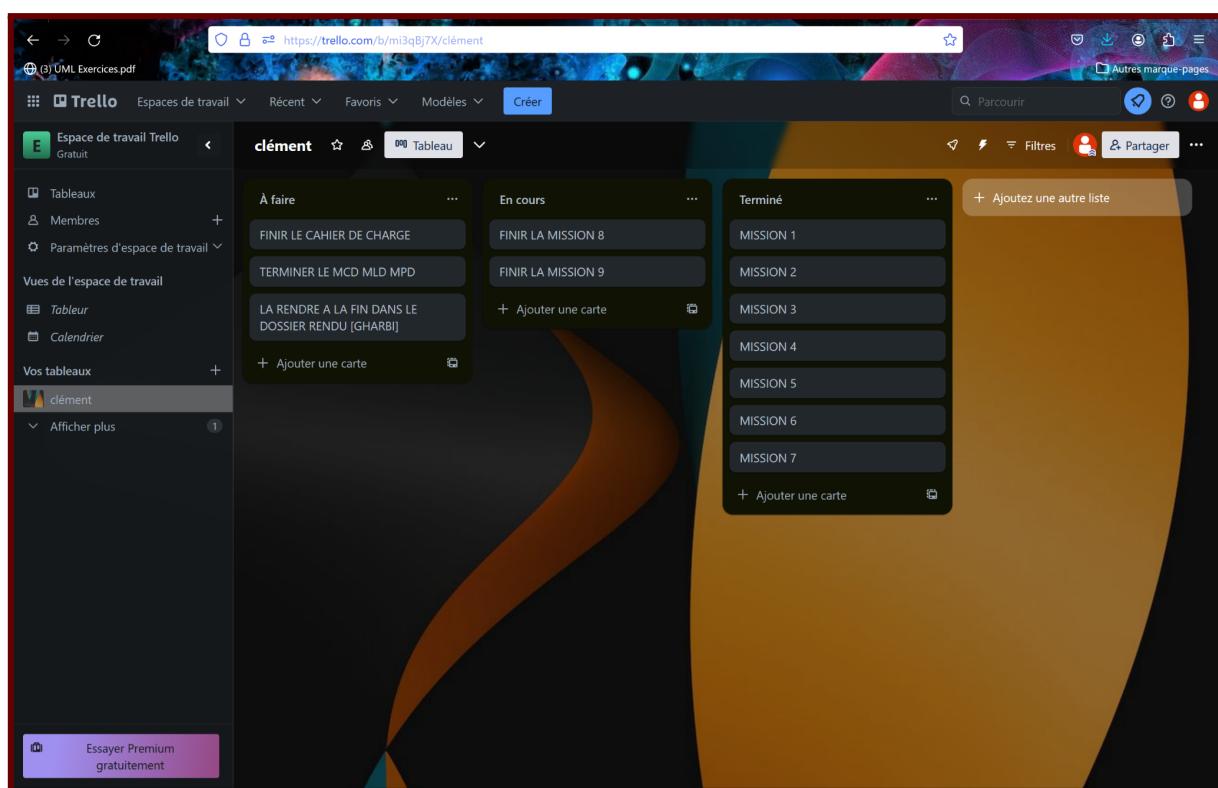
4. Gestion du projet

Pour réaliser le projet, nous utiliserons la méthode Agile Kanban.

Logiciel de développement utilisé à la réalisation, **MAMP** → **ServeurApach** (base de données **Mysql**) (IDE:**VisualStudio**) (Platform dev:**GitHub**) Nous utiliserons également l'outil de gestion de projet en ligne (**Trello**, éventuellement si vous le voulez).

(Apach permet de executer le script php au lieu de VisualCode).

INSÉREZ UNE CAPTURE D'ÉCRAN DE VOTRE TRELLO



Nous travaillons également sur GitHub, plateforme de développement collaboratif.

5. Conception du projet

5.1. Le front-end



Jeux Olympiques



REPRENDRE RÉPONSES MISSION 4

5.1.1. Wireframes

<p>Body B</p> <p>Ordi-Index</p> <p>Accueil Sports Calendrier des événements Résultats Accès Administrateur</p> <p>Sports Calendrier des épreuves</p> <p>Résultats</p> 	<p>Body B</p> <p>Responsive-Index</p> <p>Accueil Sports Calendrier des événements Résultats Accès Administrateur</p> <p>Sports</p> <p>Calendrier des épreuves</p> <p>Résultats</p> 
<p>Body B</p> <p>Ordi-Connexion</p> <p>Accueil Sports Calendrier des événements Résultats Accès Administrateur</p> <p>Connexion</p> <p>login : <input type="text"/> Mot de passe : <input type="password"/> Se connecter</p> 	<p>Body B</p> <p>Responsive-Connexion</p> <p>Accueil Sports Calendrier des événements Résultats Accès Administrateur</p> <p>Connexion</p> <p>login : <input type="text"/> Mot de passe : <input type="password"/> Se connecter</p> 



Jeux Olympiques



Ordi-Accueil	Responsive-Accueil
<p>Body B</p> <div style="background-color: #f0f0f0; padding: 10px;"> <div style="border-bottom: 1px solid black; margin-bottom: 10px;"> Accueil Administration Gestion Sports Gestion Lieux Gestion Calendrier Gestion pays Gestion Genres Gestion Athlètes Gestion Résultats Déconnexion </div> <p>Bonjour Super Admin</p> <div style="display: flex; justify-content: space-around; margin-top: 10px;"> <div style="text-align: center;"> <p>Frame 1</p> Gestion Administrateurs Gestion Lieux Gestion pays Gestion Athlètes </div> <div style="text-align: center;"> Gestion Sports Gestion Calendrier Gestion Genres Gestion Résultats </div> </div> <div style="text-align: center; margin-top: 20px;"> </div> </div>	<div style="background-color: #f0f0f0; padding: 10px;"> <div style="border-bottom: 1px solid black; margin-bottom: 10px;"> Accueil Administration Gestion Sports Gestion Lieux Gestion Calendrier Gestion pays Gestion Genres Gestion Athlètes Gestion Résultats Déconnexion </div> <p>Bonjour Super Admin</p> <div style="display: flex; justify-content: space-between; margin-top: 10px;"> <div style="width: 45%;"> <p>Frame 2</p> Gestion Administrateurs Gestion Sports Gestion Lieux Gestion Calendrier Gestion pays Gestion Genres Gestion Athlètes Gestion Résultats </div> <div style="width: 45%;"> </div> </div> <div style="text-align: center; margin-top: 20px;"> </div> </div>
Ordi	Responsive

5.1.2. Maquettes

Ordinateur	Responsive
<div style="background-color: #f0f0f0; padding: 10px;"> <div style="display: flex; justify-content: space-around; margin-bottom: 10px;"> Accueil Sports Calendrier des épreuves Résultats Accès administrateur </div> <div style="display: flex; justify-content: space-around; margin-bottom: 10px;"> <div style="text-align: center;"> Sports Calendrier des épreuves </div> <div style="text-align: center;"> Résultats </div> </div> <div style="text-align: center; margin-top: 20px;"> </div> </div>	<div style="background-color: #f0f0f0; padding: 10px;"> <div style="display: flex; justify-content: space-around; margin-bottom: 10px;"> Accueil Sports Calendrier des épreuves Résultats Accès administrateur </div> <div style="display: flex; justify-content: space-around; margin-bottom: 10px;"> <div style="text-align: center;"> Sports </div> <div style="text-align: center;"> Calendrier des épreuves </div> <div style="text-align: center;"> Résultats </div> </div> <div style="text-align: center; margin-top: 20px;"> </div> </div>



Jeux Olympiques



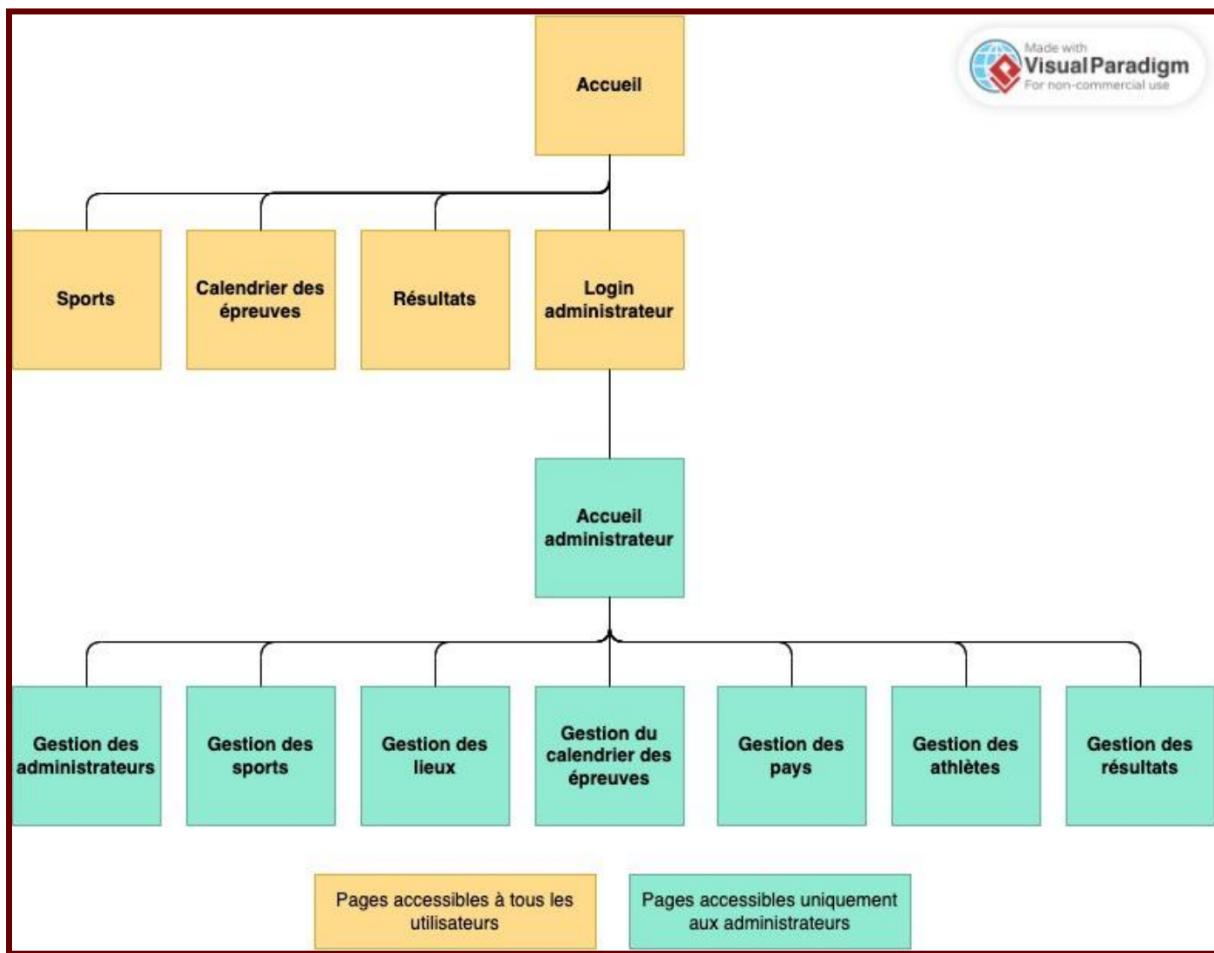
<p>Accueil Sports Calendrier des événements Résultats Accès administrateur</p> <p>Connexion</p> <p>Login : <input type="text"/> Mot de passe : <input type="password"/> Se connecter</p> <p></p>	<p>Accueil Sports Calendrier des événements Résultats Accès administrateur</p> <p>Connexion</p> <p>Login : <input type="text"/> Mot de passe : <input type="password"/> Se connecter</p> <p></p>
<p>Accueil Administration Gestion Sports Gestion Lieux Gestion Calendrier Gestion Pays Gestion Genres Gestion Athlètes Gestion Résultats Déconnexion</p> <p>Bonjour Super Admin</p> <p>Gestion Administrateurs Gestion Sports</p> <p>Gestion Lieux Gestion Calendrier</p> <p>Gestion Pays Gestion Genres</p> <p>Gestion Athlètes Gestion Résultats</p> <p></p>	<p>Accueil Administration Gestion Sports Gestion Lieux Gestion Calendrier Gestion Pays Gestion Genres Gestion Athlètes Gestion Résultats Déconnexion</p> <p>Bonjour Super Admin</p> <p>Gestion Administrateurs</p> <p>Gestion Sports</p> <p>Gestion Lieux</p> <p>Gestion Calendrier</p> <p>Gestion Pays</p> <p>Gestion Genres</p> <p>Gestion Athlètes</p> <p>Gestion Résultats</p> <p></p>
<p>Ordi</p>	<p>Responsive</p>



Jeux Olympiques

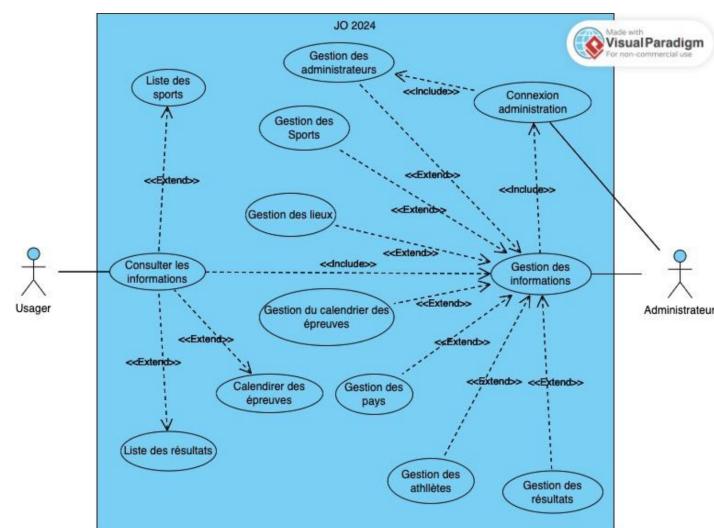


5.1.3. Arborescences



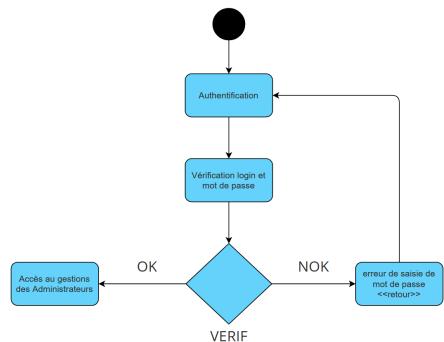
5.2. Le back-end

5.2.1. Diagramme de cas d'utilisation

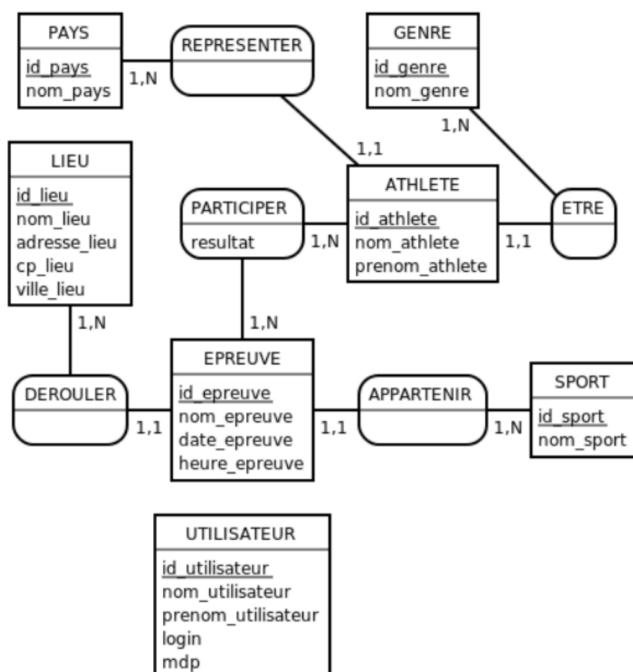




5.2.2. Diagramme d'activités



5.2.3. Modèles Conceptuel de Données (MCD)

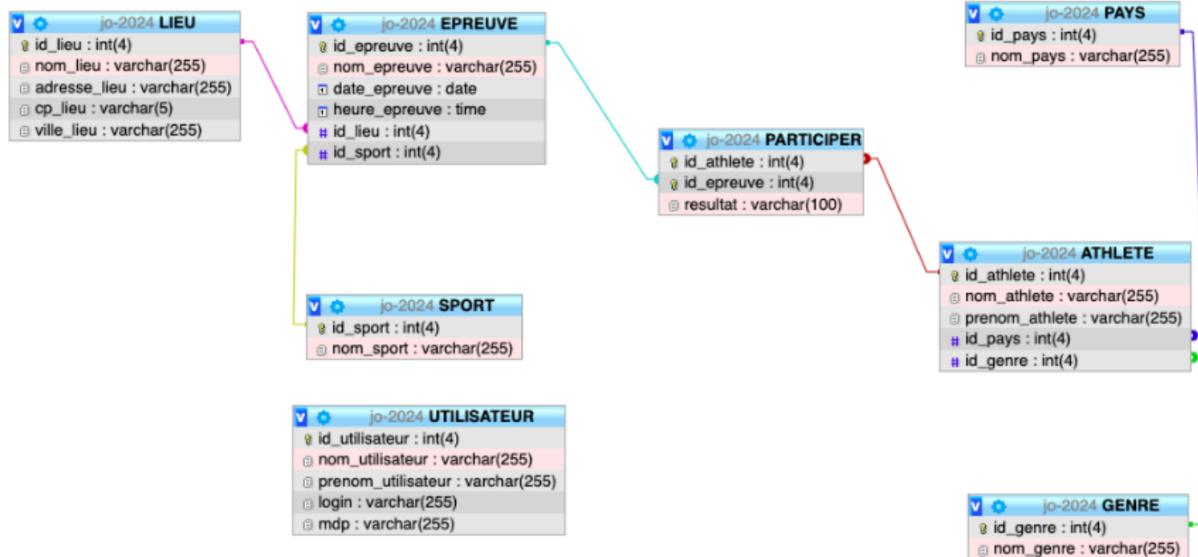


5.2.4. Modèle Logique de Données (MLD)

ATHLETE ([id_athlete](#), nom_athlete, prenom_athlete, #[id_pays](#), #[id_genre](#))
EPREUVE ([id_epreuve](#), nom_epreuve, date_epreuve, heure_epreuve, #[id_lieu](#), #[id_sport](#))
GENRE ([id_genre](#), nom_genre)
LIEU ([id_lieu](#), nom_lieu, adresse_lieu, cp_lieu, ville_lieu)
PARTICIPER (#[id_athlete](#), #[id_epreuve](#), resultat)
PAYS ([id_pays](#), nom_pays)
SPORT ([id_sport](#), nom_sport)
UTILISATEUR ([id_utilisateur](#), nom_utilisateur, prenom_utilisateur, login, mdp)



5.2.5. Modèle Physique de Données (MPD)



6. Technologies utilisées

6.1. Langages de développement Web

Les langages de codes utilisés à la réalisation d'un projet (**Html5.0**) (**Css3.0**) (**PHP7 où 8**) (**JS**) (**Sql** pour les base de données) (**Bdd et une base de données relationnelle et non relationnelle Oracle**).

6.2. Base de données

(**PHP7 où 8**) (**Sql** pour les base de données) (**Bdd et une base de données relationnelle et non relationnelle Oracle**).



7. Sécurité

7.1. Login

Le processus de connexion permet à un utilisateur de s'authentifier sur un site web en vérifiant son identité, généralement par un nom d'utilisateur et un mot de passe.

- Utiliser HTTPS pour sécuriser les données transmises.
- Valider les entrées côté serveur.
- Limiter les tentatives de connexion infructueuses pour prévenir les attaques par force brute.

7.2. Cryptage des mots de passe

Le cryptage des mots de passe consiste à transformer les mots de passe en une forme sécurisée qui ne peut pas être facilement déchiffrée, pour protéger les informations de l'utilisateur en cas de fuite de données.

- Utiliser des algorithmes de hachage robustes comme bcrypt.
- Implémenter des sels (valeurs aléatoires uniques) pour chaque mot de passe haché pour prévenir les attaques par dictionnaire et les attaques par tables de hachage.

```
import bcrypt

# Hachage du mot de passe
password = b"monSuperMotDePasse"
hashed = bcrypt.hashpw(password, bcrypt.gensalt())

# Vérification du mot de passe
if bcrypt.checkpw(password, hashed):
    print("Connexion réussie")
else:
    print("Mot de passe incorrect")
```



7.3. Protection des pages administrateurs

Cela implique de s'assurer que seuls les utilisateurs autorisés peuvent accéder aux pages d'administration.

- Authentification forte (par exemple, 2FA).
- Vérification des rôles et des permissions avant d'accorder l'accès.
- Journalisation des accès pour le suivi et l'analyse des activités suspectes.

7.4. Protection contre les attaques XSS (Cross-Site Scripting)

Les attaques XSS exploitent les vulnérabilités d'un site web en injectant des scripts malveillants dans les pages affichées aux autres utilisateurs.

- Échapper ou valider toutes les entrées utilisateur côté serveur et client.
- Utiliser des politiques de Content Security Policy (CSP).
- Ne jamais faire confiance aux données fournies par l'utilisateur.

7.5. Protection contre les injections SQL

Les injections SQL se produisent lorsqu'un attaquant insère ou manipule des commandes SQL dans une entrée de l'application pour accéder ou manipuler la base de données.

- Utiliser des requêtes préparées et des ORM pour éviter les injections SQL.
- Ne jamais construire des requêtes SQL directement à partir des entrées de l'utilisateur.
- Valider et échapper toutes les entrées utilisateurs.

```
import psycopg2

conn = psycopg2.connect("dbname=test user=postgres")
cur = conn.cursor()

# Utilisation sécurisée avec des requêtes préparées
cur.execute("INSERT INTO table (colonne) VALUES (%s)", (valeur,))
```