

# Cahier des charges techniques



## Sommaire

### [1. Contexte du projet](#)

#### [1.1. Présentation du projet](#)

#### [1.2. Date de rendu du projet](#)

### [2. Besoins fonctionnels](#)

### [3. Ressources nécessaires à la réalisation du projet](#)

#### [3.1. Ressources matérielles](#)

#### [3.2. Ressources logicielles](#)

### [4. Gestion du projet](#)

### [5. Conception du projet](#)

#### [5.1. Le front-end](#)

##### [5.1.1. Wireframes](#)

##### [5.1.2. Maquettes](#)

##### [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 \(MLD\)](#)

##### [5.2.5. Modèle Physique de Données \(MPD\)](#)

## 6. Technologies utilisées

### 6.1. Langages de développement Web

### 6.2. Base de données

## 7. Sécurité

### 7.1. Login et protection des pages administrateurs

### 7.2. Cryptage des mots de passe avec Bcrypt

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

### 7.4. Protection contre les injections SQL

# 1. Contexte du projet

## 1.1. Présentation du projet

Votre agence web a été sélectionnée par le comité d'organisation des jeux olympiques de Los Angeles 2028 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 2028.

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 8 novembre 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

Les ressources matérielles sont:

- Ordinateur portable
- Ordinateur fixe (avec souris, clavier, écran et unité centrale)
- Connexion Ethernet ou Wifi

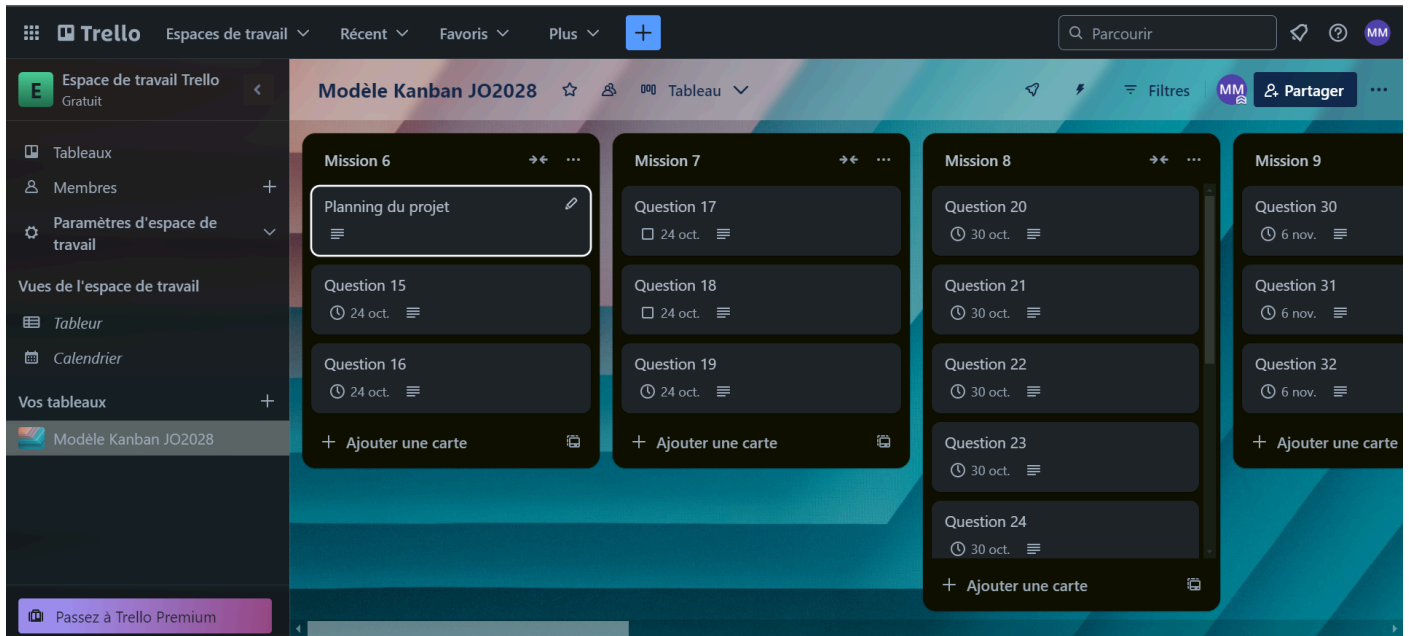
## 3.2. Ressources logicielles

Les ressources logicielles sont:

- Visual Studio Code (environnement de développement)
- Plateforme de développement collaborative: Github
- Serveur Web: Apache (contenu dans MAMP)
- Système de gestion de base de données relationnels: MySQL (contenu dans MAMP)
- Outil de gestion de projet: Trello
- Conception d'UML et arborescence: Visual Paradigm Online
- Maquettage: Figma
- Conception de base de données: Mocodo

## 4. Gestion du projet

Pour réaliser le projet, nous utiliserons la méthode Agile Kanban. Nous utiliserons également l'outil de gestion de projet en ligne Trello.



Nous travaillons également sur GitHub, plateforme de développement collaboratif.

## 5. Conception du projet

### 5.1. Le front-end

- HTML 5
- CSS 3
- JavaScript

#### 5.1.1. Wireframes

Version ordinateur:

www.url.com

☆ M ⋮

AccueilSportsCalendrier des évènementsRésultatsAccès administrateur

Connexion

Email

Value

Password

Value

Sign In

[Forgot password?](#)

logo jo

AccueilSportsCalendrier des évènementsRésultatsAccès administrateur

Liste des sports

Sport
Athlétisme
Boxe
Cyclisme
Escalade
Gymnastique
Lancer
Natation
Saut en hauteur
Saut en longueur
VTT

Retour Accueil

logp jo

Accueil

Sports

Calendrier des évènements

Résultats

Accès administrateur

Sports

Calendrier des épreuves

Résultats

logo  
jo

Version téléphone:



Accueil

Sports

Calendrier des événements

Résultats

Accès administrateur

## Liste des sports

### Sport

Athlétisme

Boxe

Cyclisme

Escalade

Gymnastique

Lancer

Natation

Saut en hauteur

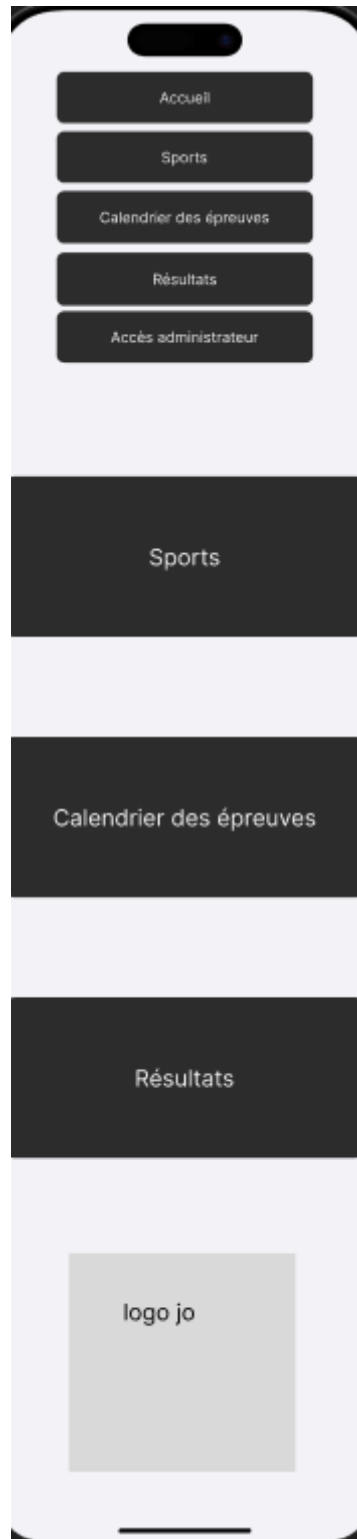
Saut en longueur

VTT

Retour Accueil

logo jo





Version tablette:

Accueil

Sports

Calendrier des événements

Résultats

Accès administrateur

## Connexion

Login

Mot de passe

Se connecter

[Mot de passe oublié?](#)

logo jo

Accueil

Sports

Calendrier des événements

Résultats

Accès administrateur

## Liste des sports

Sport

Athlétisme

Boxe

Cyclisme

Escalade

Gymnastique

Lancer

Natation

Saut en hauteur

Saut en longueur

VTT

Retour Accueil

logo jo



### 5.1.2. Maquettes

Version ordinateur:



#### Connexion

Login :

Mot de passe :

Se connecter



[Accueil](#)[Sports](#)[Calendrier des événements](#)[Résultats](#)[Accès administrateur](#)

## Liste des Sports

Sport
Athlétisme
Boxe
Cyclisme
Escalade
Gymnastique
Lancer
Natation
Saut en hauteur
Saut en longueur
VTT

[Retour Accueil](#)[Accueil](#)[Sports](#)[Calendrier des épreuves](#)[Résultats](#)[Accès administrateur](#)[Sports](#)[Calendrier des épreuves](#)[Résultats](#)

Version responsive:

Accueil

Sports

Calendrier des évènements

Résultats

Accès administrateur

## Connexion

Login :

Mot de passe :

Se connecter



[Accueil](#)[Sports](#)[Calendrier des évènements](#)[Résultats](#)[Accès administrateur](#)

## Liste des Sports

Sport
Athlétisme
Boxe
Cyclisme
Escalade
Gymnastique
Lancer
Natation
Saut en hauteur
Saut en longueur
VTT

[Retour Accueil](#)

Accueil

Sports

Calendrier des épreuves

Résultats

Accès administrateur

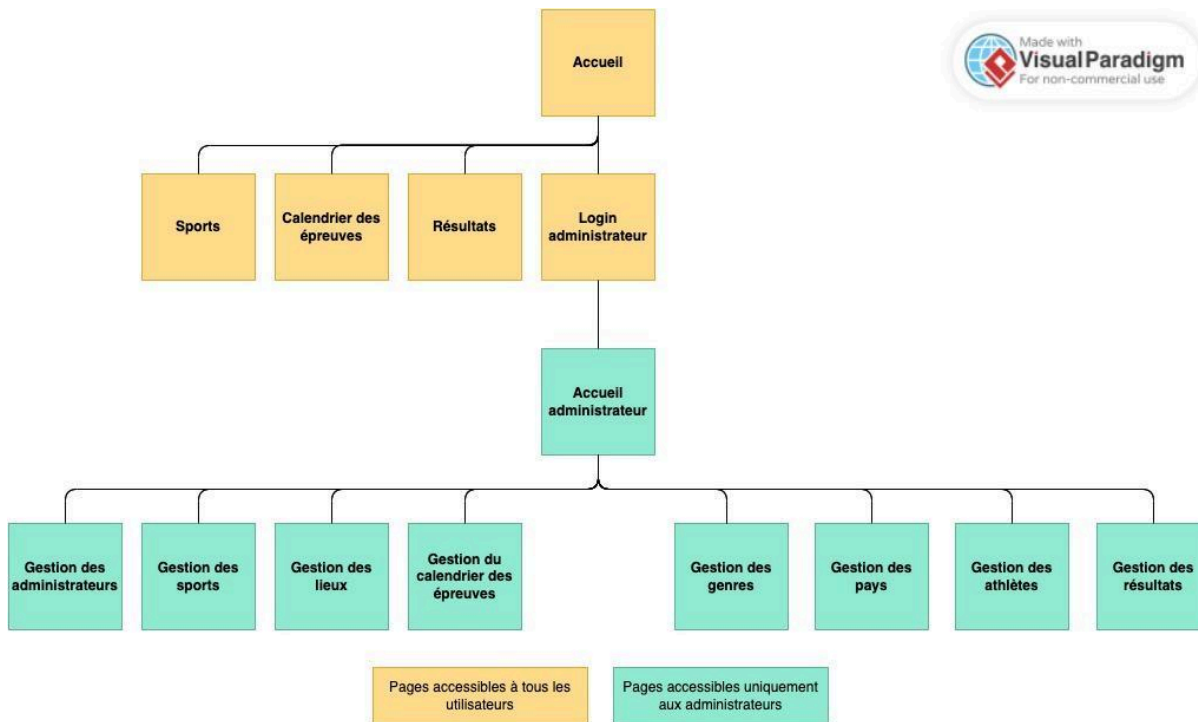
Sports

Calendrier des épreuves

Résultats



### 5.1.3. Arborescences

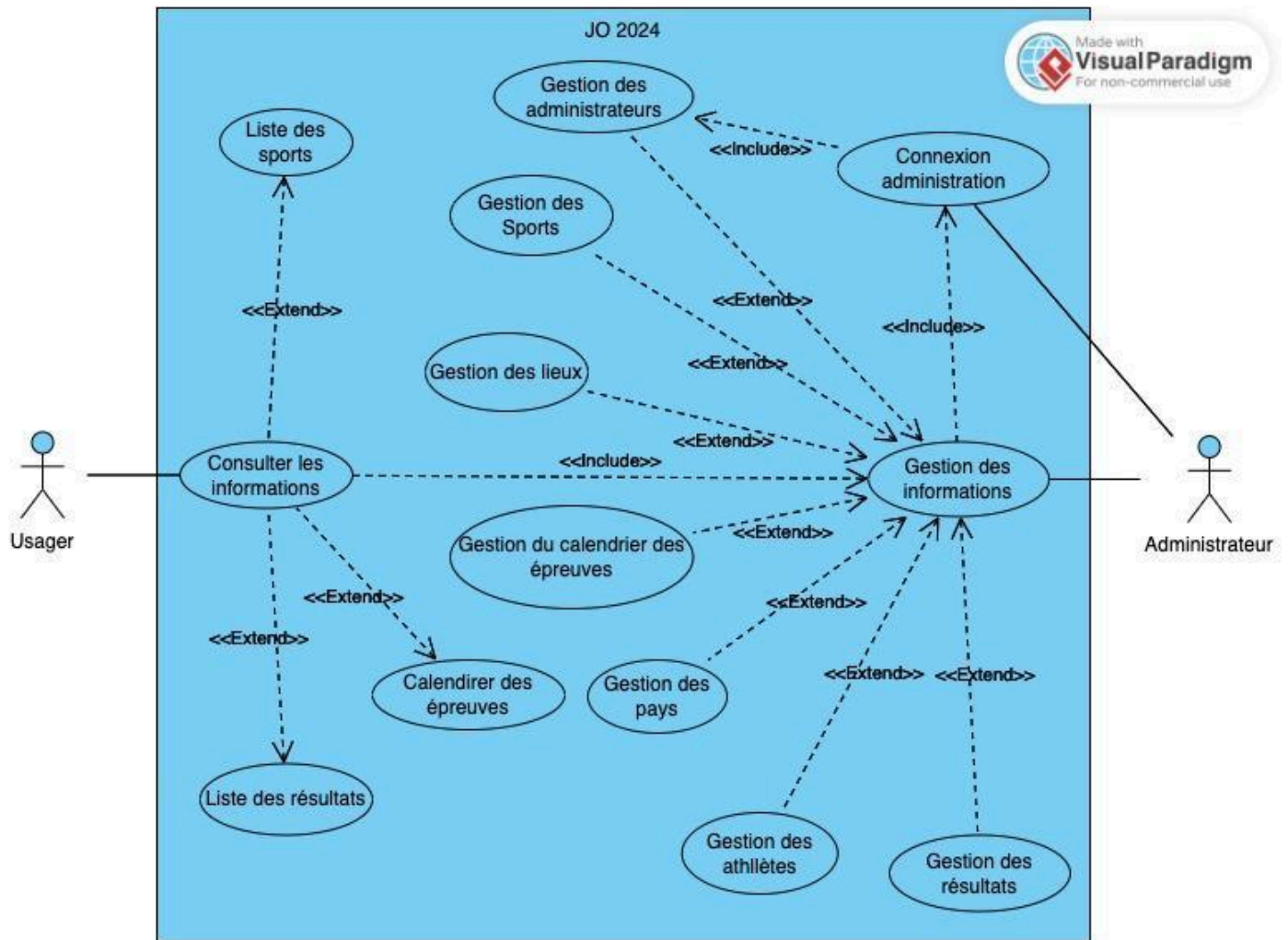


## 5.2. Le back-end

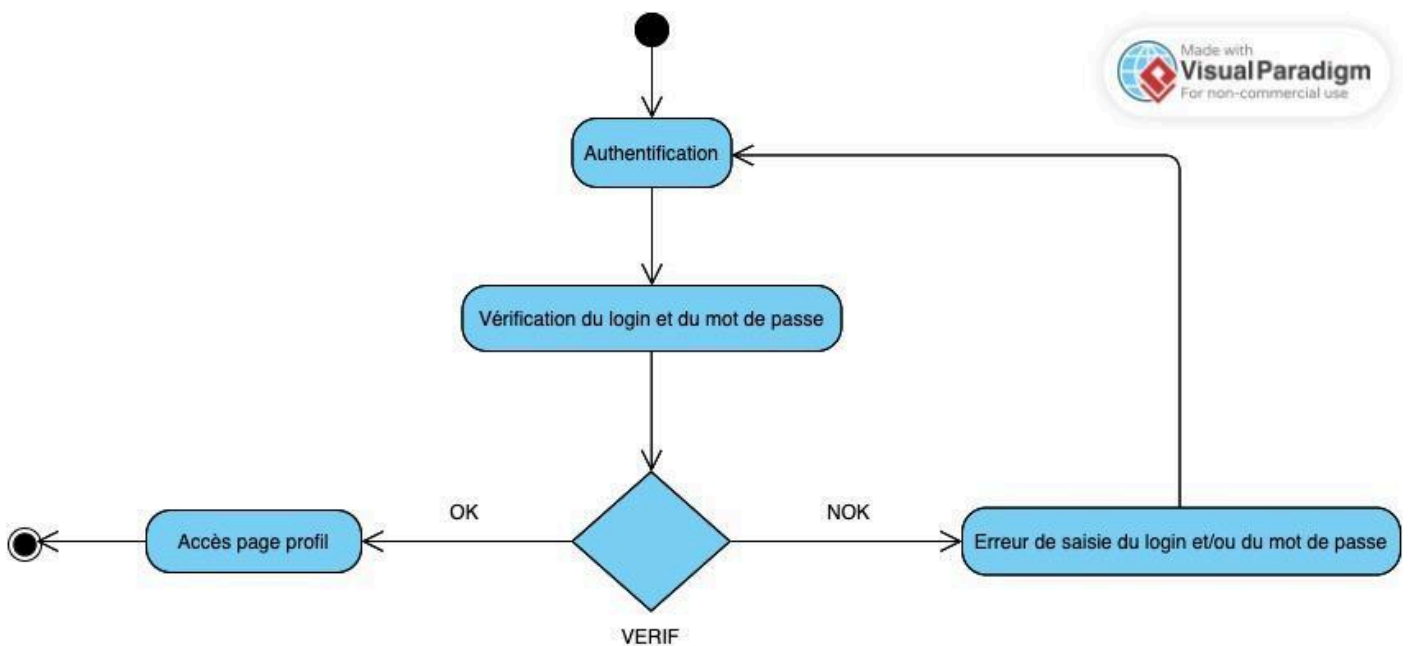
- PHP (version de PHP: 8)
- Langage Interrogation base de données: SQL



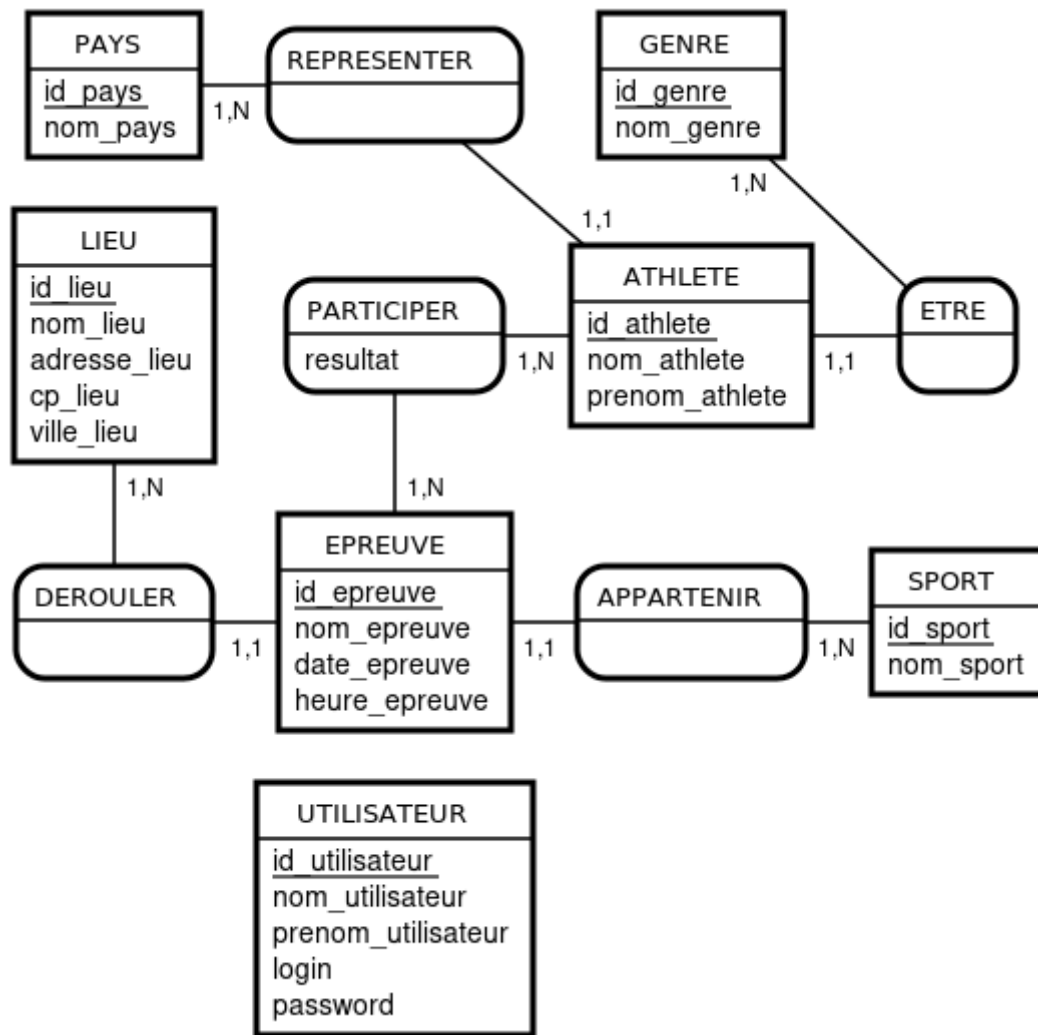
### 5.2.1. Diagramme de cas d'utilisation



### 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, password)

#### Correction version BTS

- ATHLETE (id\_athlete, nom\_athlete, prenom\_athlete, id\_pays, id\_genre)  
Clé primaire : id\_athlete  
Clés étrangères : id\_pays en référence à id\_pays de PAYS  
id\_genre en référence à id\_genre de GENRE
- EPREUVE (id\_epreuve, nom\_epreuve, date\_epreuve, heure\_epreuve, id\_lieu, id\_sport)

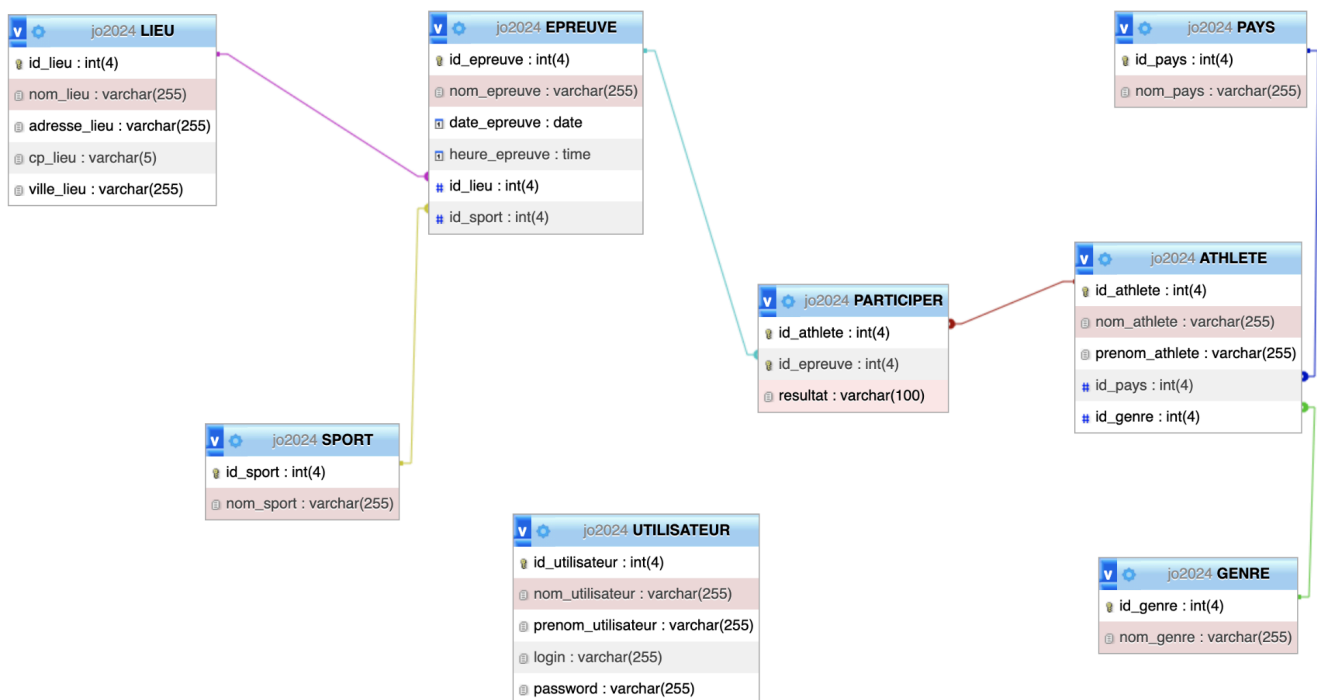
Clé primaire : id\_epreuve

Clés étrangères : id\_lieu en référence à id\_lieu de LIEU

id\_sport en référence à id\_sport de SPORT

- GENRE (id\_genre, nom\_genre)  
Clé primaire : id\_genre
- LIEU (id\_lieu, nom\_lieu, adresse\_lieu, cp\_lieu, ville\_lieu)  
Clé primaire : id\_lieu
- PARTICIPER (id\_athlete, id\_epreuve, resultat)  
Clé primaire : id\_athlete, id\_epreuve  
Clés étrangères : id\_athlete en référence à id\_athlete de ATHLETE  
id\_epreuve en référence à id\_epreuve de EPREUVE
- PAYS (id\_pays, nom\_pays)  
Clé primaire : id\_pays, id\_epreuve
- SPORT (id\_sport, nom\_sport)  
Clé primaire : id\_sport
- UTILISATEUR (id\_utilisateur, nom\_utilisateur, prenom\_utilisateur, login, password)  
Clé primaire : id\_utilisateur, id\_epreuve

### 5.2.5. Modèle Physique de Données (MPD)



## 6. Technologies utilisées

### REPRENDRE RÉPONSES MISSION 1

#### 6.1. Langages de développement Web

Pour ce projet, le langage de développement web sera Visual Studio Code

#### 6.2. Base de données

Notre base de données sera développée sur myphpAdmin

## 7. Sécurité

### DÉFINISSEZ (SI POSSIBLE) ET EXPLIQUEZ BRIÈVEMENT VOTRE SOLUTION.

### AJOUTEZ SI VOUS LE SOUHAITEZ DES COURS EXTRAITS DE CODE.

#### 7.1. Login et protection des pages administrateurs

Pour le login, on fera un formulaire en PHP avec la méthode POST pour pas que les informations saisies soient révélées dans la barre de recherche. Pour protéger cela on peut utiliser htmlspecialchars().

#### 7.2. Cryptage des mots de passe avec Bcrypt

Expliquez à quoi sert Bcrypt comment vous faites en PHP.

Bcrypt est un algorithme de hachage sécurisé utilisé pour le cryptage des mots de passe. Il est conçu pour être résistant aux attaques par force brute grâce à deux mécanismes clés :

- Salage : Ajoute un élément aléatoire (sel) au mot de passe pour le rendre unique.
- Facteur de coût : Permet d'ajuster la difficulté du calcul, ralentissant les tentatives de piratage.

Pour le hachage, on utilise la commande password\_hash() et pour la vérification d'un mot de passe on utilise la commande password\_verify().

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

Donnez la définition et expliquez comment vous faites en PHP.

Les attaques XSS (Cross-Site Scripting) permettent à un attaquant d'injecter du code malveillant dans une page web, pouvant voler des données ou manipuler l'interface pour d'autres utilisateurs. Comme solution, on peut utiliser htmlspecialchars() pour empêcher l'exécution d'un code malveillant en convertissant les caractères spéciaux en html.

## 7.4. Protection contre les injections SQL

Les injections SQL sont des attaques où un utilisateur malveillant manipule des requêtes SQL pour accéder ou modifier des données sensibles dans une base de données. Comme solution on peut utiliser les requêtes préparées avec PDO, cela va empêcher toute modification notamment les malveillantes