



# Helvetic Basket Documentation Technique

**Faites par :** Carlos Hidalgo

**Accompagné par :** Stephane Garchery

GitHub : <https://github.com/ohtpz/HelveticBasket>

2024 - 2025 | P3A

# Table des matières

- Introduction..... 3
- Cahier de charges..... 4
  - Environnement..... 4
  - Fonctionnalités attendues (extrait)..... 5
    - Gestion des matchs :..... 5
    - Gestion des joueurs et des équipes :..... 5
    - Système de favoris :..... 5
    - Gestion des permissions et accès aux données :..... 5
    - Expérience utilisateur :..... 6
    - Sécurité et validation des données :..... 6
  - Users stories..... 6
  - Maquette..... 7
  - MCD..... 8
- Planification..... 11
- Architecture du projet..... 12

# Introduction

## Description :

Le but de cette application est de permettre aux fans et aux joueurs de suivre les matchs de basketball ici en Suisse. Les utilisateurs peuvent chercher les matchs qui l'intéressent et ils peuvent filtrer les matchs en fonction de la date, du niveau ou de la région. Lorsque l'utilisateur a cliqué sur un match, il peut voir les statistiques du match. L'utilisateur peut voir le lieu du match, le score final, les joueurs ayant participé, ainsi que les minutes jouées et les points marqués par chacun. Si l'utilisateur est connecté, il peut mettre des joueurs ou des équipes en favoris pour les voir plus facilement.

## Public cible :

- Les fans de basketball ici en Suisse.
- Arbitres, joueurs, entraîneurs, etc...

## Types d'utilisateurs :

- Non Connecté - ils peuvent seulement voir les matchs
- User connecté - ils peuvent aussi mettre en favoris des joueurs et des équipes
- Administrateur - il a tous les droit de l'application (CRUD)

## Technologies et outils utilisés :

- Visual Studio Code
- GitHub
- HTML, CSS, PHP et Bootstrap
- phpMyAdmin

# Cahier de charges

## Environnement

L'application **Helvetic Basket** sera développée en utilisant les technologies suivantes :

- **Backend** : PHP avec le framework Slim pour la gestion des routes et des requêtes.
- **Base de données** : MariaDB/MySQL, gérée avec PDO et organisée en respectant les principes de normalisation.
- **Frontend** : HTML5, CSS3, et Bootstrap pour assurer un affichage responsive et ergonomique.
- **Environnement de développement** : WSL Debian avec serveur Apache.
- **Gestion des fichiers** : Upload sécurisé des médias avec restrictions sur les extensions et la taille des fichiers.
- **Authentification et permissions** : Système multi-utilisateurs avec gestion des rôles (administrateur, utilisateur standard).
- **Versionnement** : Utilisation de GitHub pour le suivi des modifications et la collaboration.

# Fonctionnalités attendues (extrait)

L'application **Helvetic Basket** doit permettre de :

## **Gestion des matchs :**

- Affichage et filtrage des listes de matchs accessible à tous les utilisateurs (connectés ou non).
- Ajout, modification et suppression des matchs par les administrateurs.
- Présence de boutons pour modifier et supprimer un match dans la liste des matchs (uniquement visible par les administrateurs).

## **Gestion des joueurs et des équipes :**

- Ajout, modification et suppression des joueurs par les administrateurs.
- Affichage des informations détaillées des joueurs.
- Lorsque l'utilisateur clique sur un joueur, il peut modifier ou supprimer le joueur (uniquement pour les administrateurs).

## **Système de favoris :**

- Un utilisateur connecté peut ajouter un ou plusieurs joueurs/équipes à ses favoris.
- Chaque utilisateur ne peut voir que ses propres favoris.
- Un utilisateur non connecté ne peut pas ajouter de favoris.
- Lorsqu'un joueur est mis en favori, l'utilisateur peut voir son match le plus récent ainsi que ses statistiques (Points et minutes et sa moyenne de minutes et points).
- Lorsqu'une équipe est mise en favori, l'utilisateur peut voir son historique de victoires et de défaites ainsi que ses prochains matchs.

## **Gestion des permissions et accès aux données :**

- Mise en place d'un système de permissions pour restreindre l'accès aux données en fonction du rôle de l'utilisateur (invité, utilisateur connecté, administrateur).
- Un utilisateur connecté ne peut pas voir les favoris des autres utilisateurs.

### **Expérience utilisateur :**

- Interface utilisateur claire et ergonomique avec une navigation fluide.
- Système de filtrage et de recherche des matchs par équipe, date ou statut.
- Affichage optimisé des statistiques et résultats des matchs.
- Compatibilité responsive pour un affichage optimal sur mobile et desktop.

### **Sécurité et validation des données :**

- Sécurisation des entrées utilisateur contre les attaques XSS et injections SQL.
- Vérification des permissions avant toute action de modification ou suppression.
- Gestion des sessions pour garantir la confidentialité des utilisateurs connectés.

## Users stories

### ☒ **En tant qu'utilisateur sans compte, je peux voir les listes de matchs.**

- Je ne peux pas préférer un joueur ni une équipe

### ☒ **En tant qu'utilisateur connecté, je peux voir et filtrer les listes de matchs, mets un ou plusieurs joueurs/équipes préféré.**

- Lors de la mise en favori d'un joueur, je peux voir le match le plus récent et ses statistiques.
- Lors de la mise en favori d'une équipe, je peux voir combien fois l'équipe a gagné et perdu. Je peux aussi voir les prochains matchs

### ☒ **En tant qu'administrateur, je peux ajouter, modifier ou supprimer des matchs et des joueurs.**

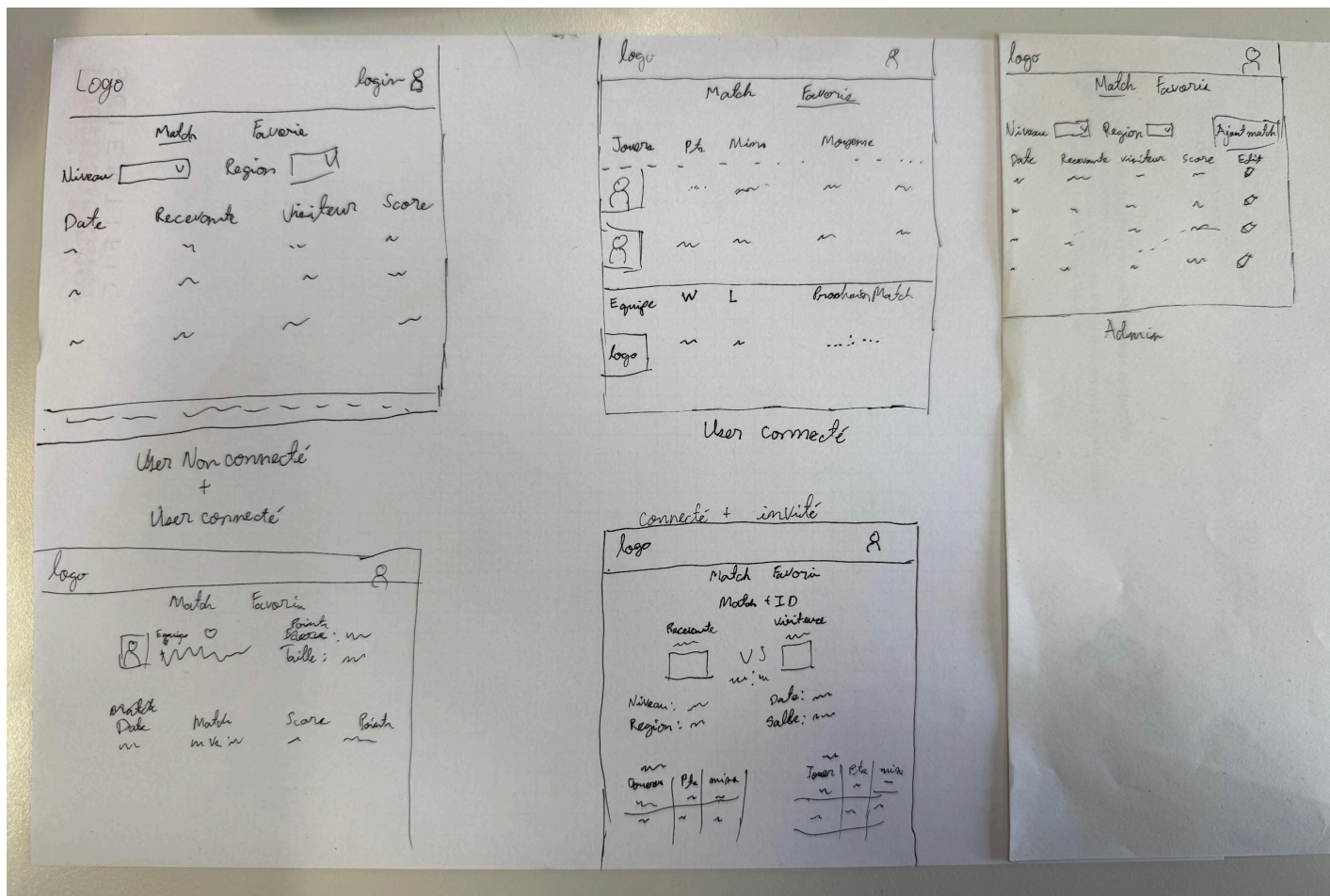
- Lorsque je peux voir la liste des matchs, il y a des boutons pour modifier et supprimer les matchs.
- Lorsque je clique sur un joueur, je peux modifier ou supprimer le joueur.

### ☒ **Connecté en tant qu'utilisateur A, je ne vois pas les favoris des autres utilisateurs.**

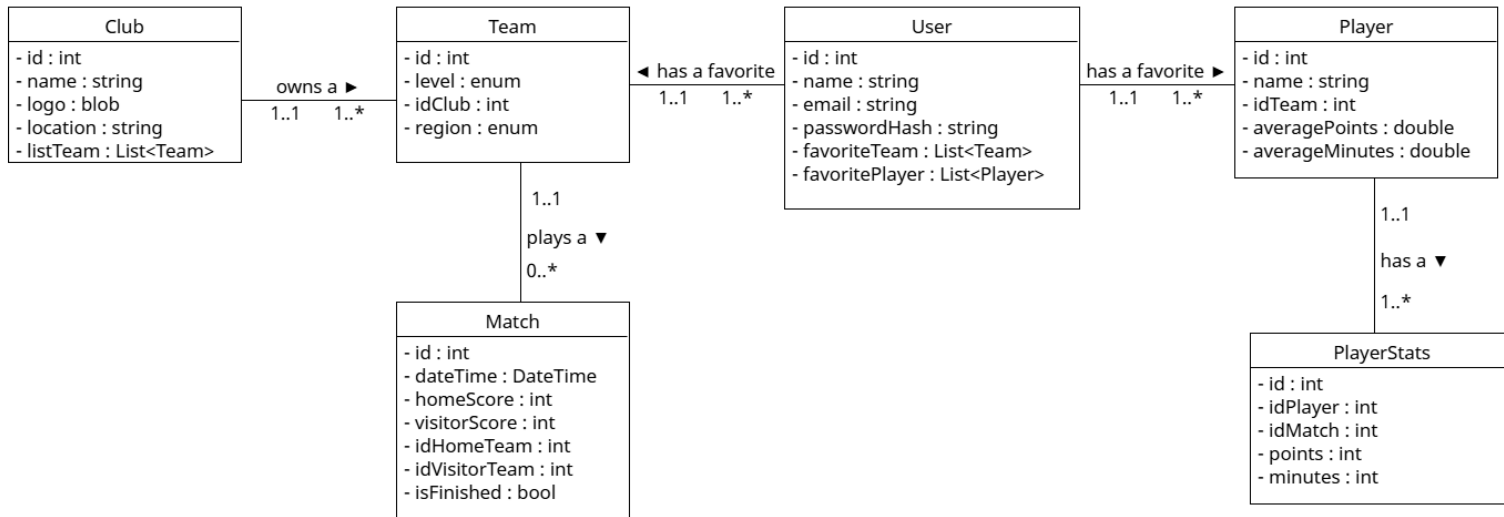
- Chaque utilisateur ne peut voir que les joueurs/équipes qu'il a mis en favori.

- Une gestion des permissions doit être mise en place pour restreindre l'accès aux données.

## Maquette



# MCD



## Club :

- ID : Identifiant du club
- Name : Nom du club
- Logo : Logo du club
- Location : Lieu où se déroule le match
- ListTeam : La liste des équipes associées à un club

**Relations :** Un club possède une ou plusieurs équipes

## Team :

- ID : Identifiant de l'équipe
- Level : Niveau de jeu (U16, U18, U20)
- IdClub : Clé étrangère liée au club
- Region : Niveau de compétition (Cantonal, Regional, National)



### **Relations :**

- Appartient à un club (1..1)
- Participe à des matchs (0..\*)
- A des joueurs.
- Peut être mise en favori par des utilisateurs.

### **User :**

- ID : Identifiant d'utilisateur
- name : nom de l'utilisateur
- email : adresse e-mail.
- passwordHash : mot de passe (haché).
- favoriteTeam : liste des équipes mises en favoris.
- favoritePlayer : liste des joueurs mis en favoris.

### **Relations :**

- Peut avoir plusieurs équipes et joueurs favoris (1..\*).

### **Player :**

- id : identifiant unique.
- name : nom du joueur.
- idTeam : clé étrangère vers l'équipe.
- averagePoints : moyenne de points marqués.
- averageMinutes : moyenne de minutes jouées.

### **Relations :**

- Est lié à une équipe (1..1).
- Possède des statistiques par match (1..\*).
- Peut être ajouté en favoris par des utilisateur

### **Match :**

- id : identifiant unique.
- dateTime : date et heure du match.

- homeScore : score de l'équipe à domicile.
- visitorScore : score de l'équipe visiteuse.
- idHomeTeam : équipe à domicile.
- idVisitorTeam : équipe visiteuse.

**Relations :**

- Deux équipes participent à un match : une à domicile, une visiteuse.
- Est lié à plusieurs lignes de statistiques de joueurs.

**PlayerStats :**

- id : identifiant unique.
- idPlayer : identifiant du joueur concerné.
- idMatch : identifiant du match.
- points : nombre de points marqués par le joueur.
- minutes : nombre de minutes jouées.

**Relations :**

- Lié un joueur à un match.
- Permet d'avoir l'historique de chaque joueur.

# Planification

Semaine 1 : Rédaction du cahier des charges.

Semaine 2 : Création de Vhost, création de la BDD et l'UML, début de codage

Semaine 3-7 : Implémentation des fonctionnalités de base (CRUD et Authentification)

Semaine 8 : Mise en oeuvre des tests fonctionnels

Semaine 9 : Finalisation de toutes les fonctionnalités

Semaine 10 : Présentation de l'application

# Architecture du projet

