



Projet 2024 “ PadelMatch Pro”

LADJ Hadil & KARTOUT Ahmed Yacine

I. Cahier des charges

1. Étude de l'existant pour le projet "PadelMatch Pro"

Contexte et technologies existantes :

La gestion des installations sportives, en particulier les terrains de padel, est actuellement soutenue par une variété de plateformes numériques. Ces systèmes facilitent la réservation de terrains, la gestion des membres, et parfois l'organisation de tournois. Des exemples notables incluent Playtomic et OpenPlay, qui offrent des interfaces utilisateurs simplifiées permettant aux utilisateurs de réserver des créneaux horaires pour diverses installations sportives. Ces plateformes utilisent généralement des technologies web standard pour fournir leurs services.

Innovations et limites :

Toutefois, ces systèmes ont souvent des limitations, notamment l'absence de fonctionnalités avancées telles que le matchmaking basé sur les compétences et préférences des joueurs, qui peut améliorer l'engagement et l'expérience utilisateur. De plus, la gestion intégrée des tournois et des membres est fréquemment rudimentaire, sans options de personnalisation profonde pour les gestionnaires de complexes ou les utilisateurs finaux.

Technologies sous-jacentes :

Les plateformes existantes s'appuient principalement sur des solutions de base de données relationnelles et des frameworks frontend comme React ou Vue.js. Elles intègrent des systèmes de paiement en ligne et offrent souvent des applications mobiles pour accéder aux services de manière plus flexible.

Opportunités d'innovation pour "PadelMatch Pro" :

Le projet "PadelMatch Pro" se distingue par son utilisation d'un algorithme de matchmaking sophistiqué, qui n'est pas seulement conçu pour associer des joueurs de manière aléatoire, mais pour optimiser les matchs en fonction des niveaux de compétence et des préférences. De plus, l'intégration d'Angular pour le développement frontend offre une gestion dynamique des données en temps réel, essentielle pour les réservations instantanées et les mises à jour des statuts des terrains. Le backend en Java avec MySQL assure non seulement une base solide pour la performance et la

sécurité mais aussi une grande évolutive pour gérer un volume accru de données et d'interactions utilisateurs à mesure que la plateforme grandit.

En résumé, bien que des solutions existent dans le domaine de la gestion des installations sportives, "PadelMatch Pro" propose des améliorations significatives en termes de technologie et de fonctionnalités utilisateur, plaçant le projet dans une position favorable pour répondre aux besoins non comblés par les offres actuelles.

2. Étude des besoins pour le projet "PadelMatch Pro"

Fonctionnalités essentielles :

Le projet "PadelMatch Pro" vise à adresser plusieurs besoins spécifiques non couverts efficacement par les solutions existantes, en se concentrant sur les fonctionnalités qui amélioreront l'expérience utilisateur et optimiseront la gestion des complexes de padel. Voici les besoins identifiés :

Réservation de terrains :

- Interface utilisateur intuitive : Une interface en ligne facile à utiliser pour réserver des terrains à des horaires spécifiques.
- Gestion des annulations et des modifications : Capacité pour les utilisateurs de modifier ou annuler leurs réservations facilement.

Matchmaking sophistiqué :

- Appariement basé sur le niveau de compétence et préférences : Un système qui analyse les données des joueurs pour créer des matchs équilibrés, augmentant ainsi la satisfaction des utilisateurs.
- Suggestions dynamiques : Suggestions de partenaires de jeu et de créneaux horaires basées sur les préférences et historiques des joueurs.

Gestion des membres :

- Enregistrement et suivi des membres : Système pour inscrire de nouveaux membres, mettre à jour les informations des membres existants, et gérer les cotisations de manière sécurisée.
- Gestion des profils utilisateurs : Profils personnalisables qui permettent aux joueurs de définir leurs préférences et de recevoir des recommandations personnalisées.

Organisation de tournois :

- Gestion complète des tournois : Outils pour créer, organiser et gérer des tournois, incluant l'inscription des joueurs, le suivi des matchs et la publication des résultats.

- Flexibilité des formats de tournoi : Support pour différents formats de tournoi pour accommoder diverses configurations et tailles d'événements.

Gestion de l'inventaire :

- Suivi en temps réel : Gestion de l'état des équipements disponibles (raquettes, balles), y compris leur disponibilité et maintenance.
- Alertes automatiques : Notifications pour le réapprovisionnement ou la maintenance nécessaire des équipements.

Reporting et statistiques :

- Analyse des données : Statistiques détaillées sur l'utilisation des terrains, les performances des joueurs, et les revenus générés.
- Tableaux de bord personnalisés : Tableaux de bord pour les gestionnaires de complexes et les utilisateurs, permettant un accès rapide aux informations pertinentes.

3. Étude de faisabilité pour le projet "PadelMatch Pro"

Choix des méthodologies et justifications :

Analyse et Conception :

- Méthode Merise : Utilisée pour la conception de la base de données, Merise est particulièrement adaptée pour structurer et modéliser les données dans des environnements complexes. Elle permet une approche claire et systématique, divisant le système en modèles de données et de traitements, ce qui facilite la gestion des relations entre les différentes entités (terrains, utilisateurs, réservations, etc.).
- Avantages : Clarifie la structure des bases de données, facilite la maintenance et l'évolution du système.
- Inconvénients : Peut être rigide dans des environnements où les besoins changent rapidement.
- **UML (Unified Modeling Language)** : Utilisée pour la modélisation générale du système, UML aidera à visualiser, spécifier, construire et documenter les artefacts du projet logiciel. Elle est essentielle pour la conception des interactions entre les différents composants du système.
- Avantages : Standard de l'industrie, offre une variété de diagrammes pour couvrir presque tous les aspects de la conception.
- Inconvénients : Peut être complexe et excessif pour des projets de petite taille.

Outils et langages de développement :

Front-end Angular : Choisi pour sa robustesse et son écosystème riche, Angular permet un développement structuré et une gestion efficace des états de l'application, ce qui est crucial pour les fonctionnalités en temps réel comme les réservations et les mises à jour de matchs.

- Avantages : Binding bidirectionnel, large communauté, intégration facile avec divers back-ends.
- Inconvénients : Courbe d'apprentissage relativement élevée.

Back-end Java avec Spring Boot : Java est choisi pour sa robustesse, sa sécurité et sa portabilité entre différents environnements d'exploitation. Spring Boot facilite la configuration et le déploiement, permettant une mise en marché rapide et une scalabilité.

- Avantages : Maturité, performances élevées, vaste écosystème de bibliothèques et cadre de sécurité.
- Inconvénients : Peut nécessiter plus de ressources système comparé à d'autres langages plus légers.

Base de données MySQL : MySQL est un système de gestion de base de données relationnelle robuste et éprouvé, idéal pour les transactions impliquant des réservations et la gestion des données utilisateur.

Avantages : Large adoption, support communautaire étendu, coût efficacité.

- Inconvénients : Peut rencontrer des limites de performance sous charges extrêmement élevées sans optimisation appropriée.

Alternatives envisagées :

PHP avec Laravel : Une alternative pour le backend pourrait être PHP utilisé avec le framework Laravel, qui offre également une structure robuste pour le développement rapide d'applications web.

- Avantages : Facile à apprendre, large base d'utilisateurs, bonnes capacités d'intégration.
- Inconvénients : Moins performant que Java pour les applications à grande échelle.

PostgreSQL au lieu de MySQL : PostgreSQL est une autre option de système de gestion de base de données qui offre des fonctionnalités avancées en termes de support des transactions complexes et de la conformité aux standards SQL.

- Avantages : Plus conforme aux standards SQL, supporte mieux les charges de travail complexes.
- Inconvénients : Peut être plus complexe à administrer que MySQL.

En somme, le choix des outils et des méthodes pour "**PadelMatch Pro**" est guidé par la nécessité de combiner une robustesse technologique à une flexibilité et une efficacité de développement. Cette combinaison assure non seulement la faisabilité technique du projet, mais également sa capacité à évoluer et à s'adapter aux besoins futurs.

4. Analyse préalable pour le projet "PadelMatch Pro"

L'analyse préalable consiste à détailler la conception du système de "PadelMatch Pro", définissant clairement les caractéristiques des différentes entités, leurs interactions, et les processus spécifiques à implémenter. Cette phase est cruciale pour garantir une planification adéquate avant le développement.

Entités principales et leurs caractéristiques :

Terrain de Padel :

Identifiant du terrain : Un identifiant unique pour chaque terrain.

Localisation : Emplacement spécifique du terrain.

Statut : Disponible, réservé, en maintenance.

Type de surface : Herbe, terre battue, dur.

Utilisateur (Membre) :

Identifiant utilisateur : Un identifiant unique pour chaque membre.

Nom : Nom complet de l'utilisateur.

Niveau de compétence : Débutant, intermédiaire, avancé.

Préférences : Préférences de réservation, disponibilité.

Réservation :

Identifiant de réservation : Un identifiant unique pour chaque réservation.

Identifiant utilisateur : Lié à l'utilisateur qui fait la réservation.

Identifiant du terrain : Terrain réservé.

Date/heure de début et de fin : Les créneaux horaires de la réservation.

Match :

Identifiant de match : Un identifiant unique pour chaque match.

Joueurs : Liste des identifiants des joueurs participants.

Score : Résultat du match.

Commentaires : Observations ou commentaires sur le match.

Tournoi :

Identifiant de tournoi : Un identifiant unique pour chaque tournoi.

Nom du tournoi : Nom descriptif du tournoi.

Statut : Planifié, en cours, terminé.

Résultats : Classement ou résultats du tournoi.

Équipement :

Identifiant d'équipement : Un identifiant unique pour chaque pièce d'équipement.

Type : Raquette, balle.

Quantité : Nombre d'articles disponibles.

État : Bon, besoin de réparation, remplacé.

Interactions et processus :

Processus de réservation : Les utilisateurs sélectionnent un terrain et un créneau horaire disponibles, effectuent la réservation qui est ensuite enregistrée dans la base de données.

Matchmaking : Le système analyse les niveaux des joueurs disponibles et leurs préférences pour proposer des matchs équilibrés.

Gestion des tournois : Création et gestion de tournois avec inscription automatique ou manuelle, gestion des matches du tournoi, et mise à jour des résultats en temps réel.

Maintenance de l'équipement : Suivi de l'état de l'équipement, notifications de maintenance préventive et gestion des remplacements.

II. Analyse détaillée

1) Partie Conception

Entités et Associations :

Pour le projet "**PadelMatch Pro**", les entités principales et leurs associations sont définies comme suit :

Entités Principales :

- Terrain de Padel
- Utilisateur (Membre)
- Réservation
- Match
- Tournoi
- Équipement

Terrain de Padel:

- Identifiant du terrain (ID_Terrain)
- Localisation

- Statut (Disponible, Réserve, En Maintenance)
- Type de surface (Herbe, Terre battue, Dur)

Utilisateur (Membre):

- Identifiant utilisateur (ID_Utilisateur)
- Nom
- Niveau de compétence (Débutant, Intermédiaire, Avancé)
- Préférences de réservation
- Disponibilité

Réservation:

- Identifiant de réservation (ID_Réservation)
- Identifiant utilisateur (ID_Utilisateur)
- Identifiant du terrain (ID_Terrain)
- Date/heure de début et de fin

Match:

- Identifiant de match (ID_Match)
- Joueurs (Liste des ID_Utilisateur participants)
- Score
- Commentaires

Tournoi:

- Identifiant de tournoi (ID_Tournoi)
- Nom du tournoi
- Statut (Planifié, En cours, Terminé)
- Résultats (Classement ou résultats du tournoi)

Équipement:

- Identifiant d'équipement (ID_Équipement)
- Type (Raquette, Balle)
- Quantité
- État (Bon, Besoin de réparation, Remplacé)

Associations :

- Un Utilisateur peut faire plusieurs Réservations.
- Une Réservation concerne un Terrain de Padel.
- Un Match implique plusieurs Utilisateurs et peut être associé à une Réservation.
- Un Tournoi comprend plusieurs Matches.
- Un Terrain de Padel peut nécessiter plusieurs types d'Équipements.

Dictionnaire des données :

Entité	Attribut	Type	Description
Terrain	ID_Terrain	Integer	Identifiant unique du terrain
Terrain	Localisation	String	Emplacement spécifique du terrain
Terrain	Statut	String	Statut actuel du terrain
Terrain	Type_Surface	String	Type de surface du terrain
Utilisateur	ID_Utilisateur	Integer	Identifiant unique de l'utilisateur
Utilisateur	Nom	String	Nom complet de l'utilisateur
Utilisateur	Niveau_Compétence	String	Niveau de compétence du joueur
Utilisateur	Préférences	String	Préférences de réservation
Utilisateur	Disponibilité	String	Disponibilité du joueur
Réservation	ID_Réservation	Integer	Identifiant unique de la réservation
Réservation	ID_Utilisateur	Integer	Identifiant de l'utilisateur
Réservation	ID_Terrain	Integer	Identifiant du terrain
Réservation	Date_Heure_Début	DateTime	Début de la réservation
Réservation	Date_Heure_Fin	DateTime	Fin de la réservation
Match	ID_Match	Integer	Identifiant unique du match
Match	Joueurs	List	Liste des identifiants des joueurs
Match	Score	String	Résultat du match

Match	Commentaires	String	Observations ou commentaires
Tournoi	ID_Tournoi	Integer	Identifiant unique du tournoi
Tournoi	Nom_Tournoi	String	Nom du tournoi
Tournoi	Statut	String	Statut actuel du tournoi
Tournoi	Résultats	String	Résultats ou classement du tournoi
Équipement	ID_Équipement	Integer	Identifiant unique de l'équipement
Équipement	Type	String	Type d'équipement
Équipement	Quantité	Integer	Quantité disponible
Équipement	État	String	État de l'équipement

Attributs et Propriétés

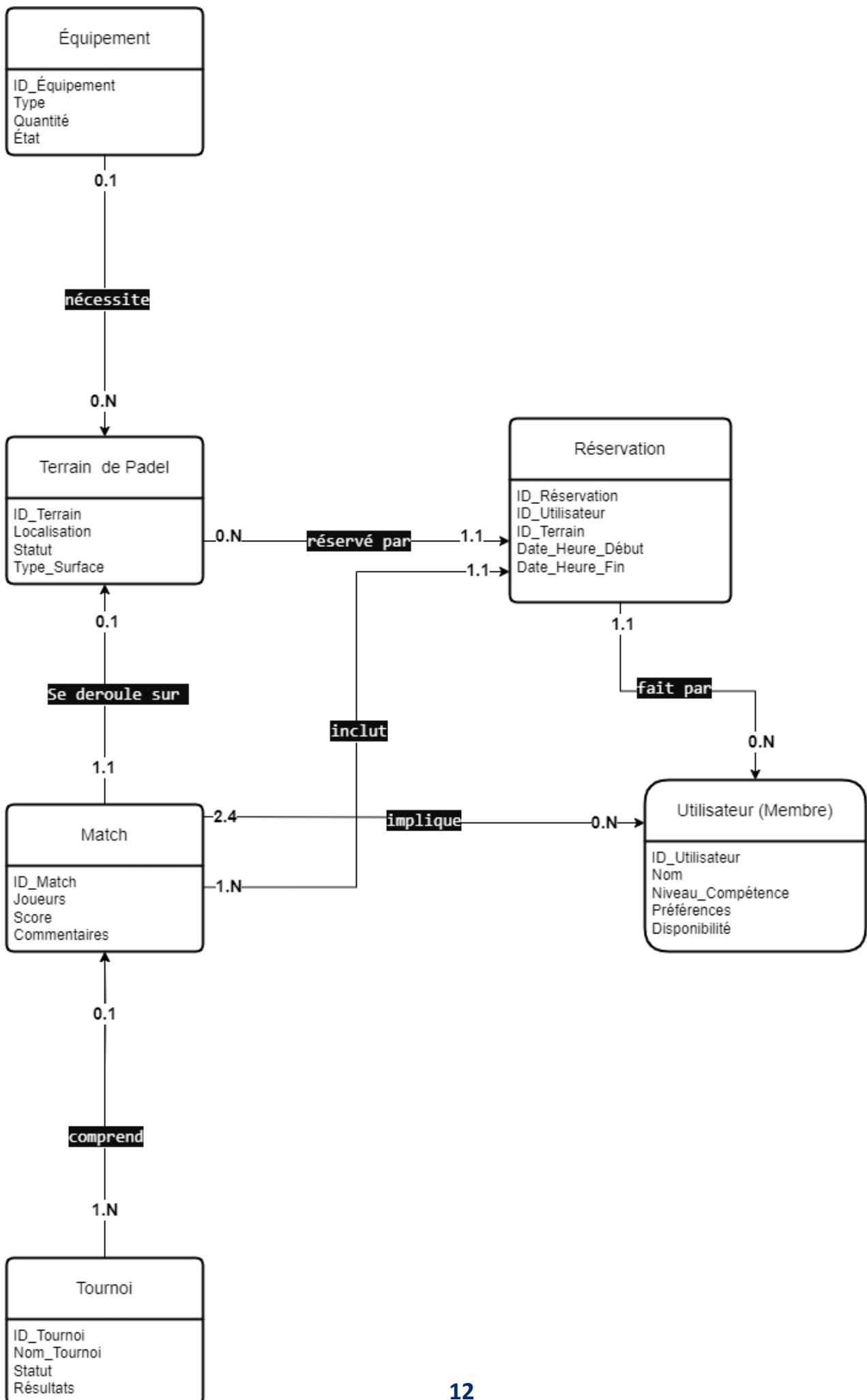
Attribut	Propriétés
ID_Terrain	Clé primaire, unique, auto-incrémenté
Localisation	Chaîne de caractères, non vide
Statut	Enumération (Disponible, Réserve, En Maintenance)
Type_Surface	Enumération (Herbe, Terre battue, Dur)
ID_Utilisateur	Clé primaire, unique, auto-incrémenté
Nom	Chaîne de caractères, non vide
Niveau_Compétence	Enumération (Débutant, Intermédiaire, Avancé)
Préférences	Chaîne de caractères
Disponibilité	Chaîne de caractères
ID_Réservation	Clé primaire, unique, auto-incrémenté
Date_Heure_Début	Date et heure, non vide
Date_Heure_Fin	Date et heure, non vide

ID_Match	Clé primaire, unique, auto-incrémenté
Joueurs	Liste de clés étrangères (ID_Utilisateur)
Score	Chaîne de caractères
Commentaires	Chaîne de caractères
ID_Tournoi	Clé primaire, unique, auto-incrémenté
Nom_Tournoi	Chaîne de caractères, non vide
Résultats	Chaîne de caractères
ID_Équipement	Clé primaire, unique, auto-incrémenté
Type	Chaîne de caractères
Quantité	Entier, non négatif
État	Enumération (Bon, Besoin de réparation, Remplacé)

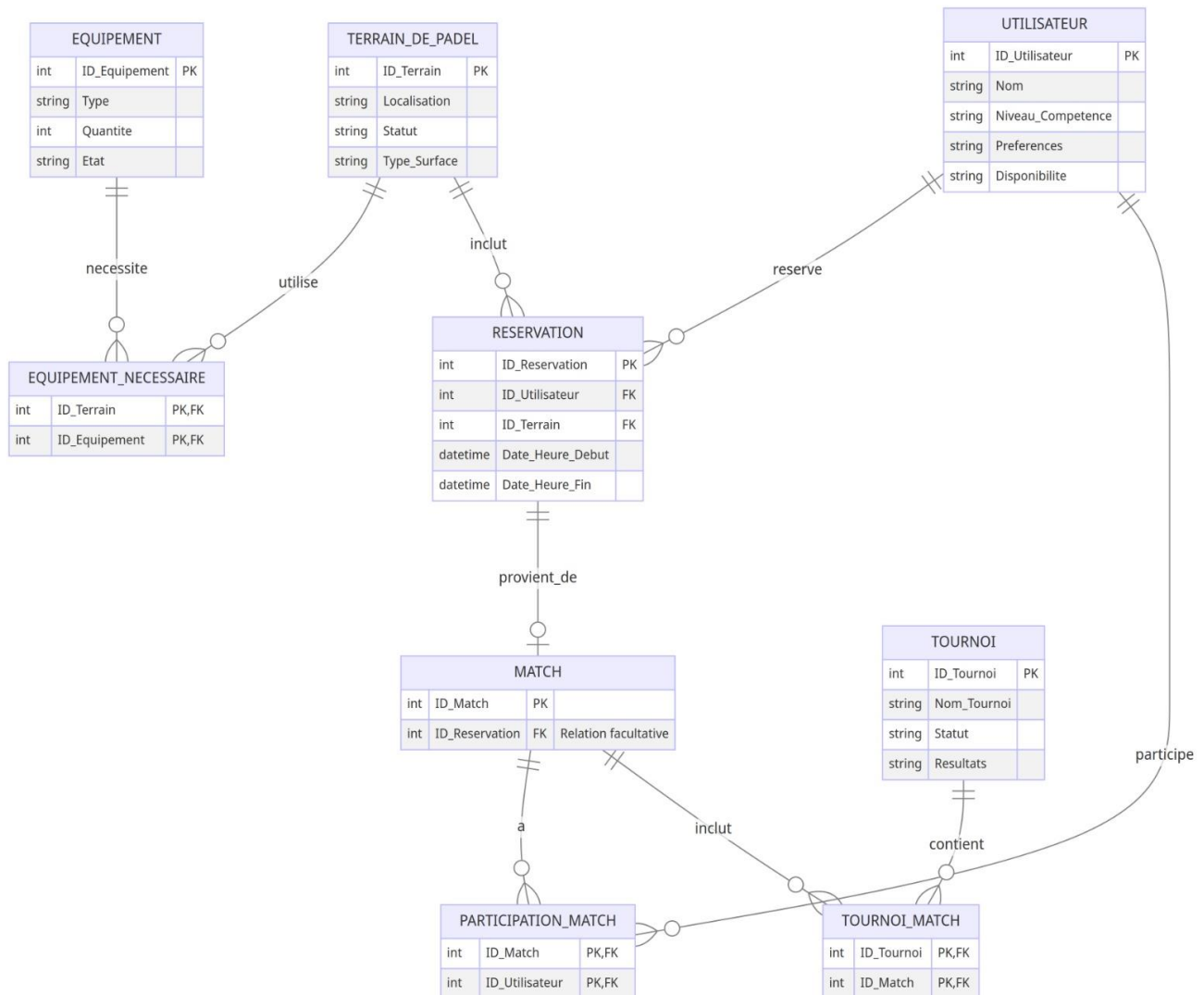
Règles de Gestion :

- Réservations : Un utilisateur ne peut pas réserver plus de deux terrains simultanément. Les réservations doivent être faites au moins 24 heures à l'avance.
- Matchmaking : Les matchs doivent être équilibrés en termes de niveaux de compétence des joueurs. Les suggestions doivent prendre en compte les disponibilités et préférences des joueurs.
- Tournois : Les inscriptions aux tournois doivent être validées par l'administrateur du club. Les résultats des matchs doivent être enregistrés dans les 24 heures suivant la fin du match.
- Équipement : Les équipements doivent être vérifiés et maintenus régulièrement. Les notifications de maintenance doivent être envoyées automatiquement en cas de besoin.

Modèle Conceptuel des Données (MCD)



Modèle Relationnel :



2) Partie fonctionnalités :

Algorithmes à Implémenter pour "PadelMatch Pro"

1. Processus de Réservation:

Objectif: Permettre aux utilisateurs de réserver un terrain à un créneau horaire spécifique.

Pseudo-code:

Fonction réserverTerrain(ID_Utilisateur, ID_Terrain, date_heure_debut, date_heure_fin)

Si utilisateur et terrain existent

Si le terrain est disponible pendant le créneau spécifié

Générer ID_Réservation unique

Insérer dans la table Réservation (ID_Réservation, ID_Utilisateur, ID_Terrain, date_heure_debut, date_heure_fin)

Mettre à jour le statut du terrain à "Réservé" pour le créneau spécifié

Retourner "Réservation réussie" et ID_Réservation

Sinon

Retourner "Terrain non disponible"

Fin Si

Sinon

Retourner "Utilisateur ou Terrain non trouvé"

Fin Si

Fin Fonction

Explications:

- Vérifier l'existence de l'utilisateur et du terrain.
- Vérifier la disponibilité du terrain pour le créneau horaire spécifié.
- Générer un identifiant unique pour la réservation.
- Insérer la réservation dans la base de données.
- Mettre à jour le statut du terrain comme réservé pour le créneau horaire choisi.
- Retourner un message de succès avec l'identifiant de réservation ou un message d'erreur en cas de non-disponibilité.

2. Matchmaking Basé sur les Compétences:

Objectif: Associer des joueurs en fonction de leur niveau de compétence et de leurs préférences pour des matchs équilibrés.

Pseudo-code:

Fonction matchmaking()

Récupérer liste des joueurs disponibles avec leurs niveaux de compétence et préférences

Trier les joueurs par niveau de compétence

Associer les joueurs par paires de niveaux similaires

Pour chaque paire de joueurs

Générer ID_Match unique

Insérer dans la table Match (ID_Match, ID_Joueur_1, ID_Joueur_2, date_match, statut =

"Planifié")

Retourner ID_Match et les détails du match

Fin Pour

Fin Fonction

Explications:

- Récupérer les joueurs disponibles avec leurs niveaux de compétence et préférences.
- Trier les joueurs par niveau de compétence.
- Associer les joueurs par paires ayant des niveaux similaires.
- Pour chaque paire, générer un identifiant unique de match.
- Insérer les détails du match dans la base de données.
- Retourner les informations du match créé.

3. Gestion des Tournois

Objectif: Créer et gérer des tournois, gérer les inscriptions, organiser les matchs et mettre à jour les résultats en temps réel.

Pseudo-code:

Fonction créerTournoi(nomTournoi, date_debut, date_fin)

Générer ID_Tournoi unique

Insérer dans la table Tournoi (ID_Tournoi, nomTournoi, date_debut, date_fin, statut = "Planifié")

Retourner ID_Tournoi et "Tournoi créé avec succès"

Fin Fonction

Fonction inscrireJoueurTournoi(ID_Tournoi, ID_Joueur)

Si tournoi existe et est "Planifié"

Si joueur n'est pas déjà inscrit

Insérer dans la table InscriptionTournoi (ID_Tournoi, ID_Joueur)

Retourner "Inscription réussie"

Sinon

Retourner "Joueur déjà inscrit"

Fin Si

Sinon

Retourner "Tournoi non trouvé ou inscriptions fermées"

Fin Si

Fin Fonction

Fonction mettreAJourResultatMatch(ID_Match, score)

Si match existe et est "En cours"

Mettre à jour le score dans la table Match

Si tous les matchs du tournoi sont terminés

Mettre à jour le statut du tournoi à "Terminé"

Fin Si

Retourner "Résultat mis à jour"

Sinon

Retourner "Match non trouvé ou déjà terminé"

Fin Si

Fin Fonction

Explications:

- **Créer Tournoi:** Générer un identifiant unique pour le tournoi, insérer les détails du tournoi dans la base de données, et retourner un message de succès.
- **Inscrire Joueur Tournoi:** Vérifier l'existence du tournoi et son statut, vérifier si le joueur est déjà inscrit, ajouter le joueur à la liste des inscriptions, et retourner un message de succès ou d'erreur.
- **Mettre à Jour Résultat Match:** Vérifier l'existence et le statut du match, mettre à jour le score, vérifier si tous les matchs du tournoi sont terminés, mettre à jour le statut du tournoi, et retourner un message de succès ou d'erreur.

4. Maintenance de l'Équipement:

Objectif: Suivre l'état des équipements, envoyer des notifications de maintenance et gérer les remplacements.

Pseudo-code:

Fonction verifierEtatEquipement()

Récupérer la liste des équipements

Pour chaque équipement

Si état = "Besoin de réparation"

Envoyer notification de maintenance à l'administrateur

Sinon Si état = "Remplacé"

Mettre à jour l'état à "Bon"

Fin Si

Fin Pour

Fin Fonction

Fonction remplacerEquipement(ID_Equipement)

Si équipement existe

Mettre à jour l'état de l'équipement à "Remplacé"

Mettre à jour la quantité si nécessaire

Retourner "Équipement remplacé"

Sinon

Retourner "Équipement non trouvé"

Fin Si

Fin Fonction

Explications:

- *Vérifier État Équipement: Parcourir la liste des équipements, envoyer des notifications de maintenance pour les équipements nécessitant une réparation, et mettre à jour l'état des équipements remplacés.*
- *Remplacer Équipement: Vérifier l'existence de l'équipement, mettre à jour son état et sa quantité si nécessaire, et retourner un message de succès ou d'erreur.*
- *Ces algorithmes décrivent les processus clés pour la gestion des terrains, des joueurs et des équipements dans le système "PadelMatch Pro", offrant une solution complète pour améliorer l'expérience utilisateur et l'efficacité opérationnelle.*