



## Rapport de travaux pratique transversal

« Mise en place d'un pari en ligne multiplateforme Bet Soccer »

Par :

RANDRIAMANDIMBITIANA Lovanirina Murelle

ANDRIANARISON Tokiniaina Hervé

RAKOTOARISOA Andrianirina Mamitiana

## Résumé

Le thème du travail s'intitule : « Pari de foot en ligne ». Pari de foot en ligne « Bet Soccer » est une application multiplateforme qui comprends : Une application mobile, web et de bureau ont été créées pour répondre aux besoins du client. L'application permet de parier sur des matchs de football en ligne et de permettre à tout joueur de parier de façon simple n'importe où n'importe quand sur mobile ou navigateur.

Le long du projet, on a utilisé le scrum, une méthode agile qui consiste à la communication à chaque achèvement d'une étape.

Vous allez voir dans ce document les démarches que nous avons entrepris pour la réalisation de ce projet, depuis sa conception jusqu'à son développement final

## Liste des figures

Figure 1 : Cas d'utilisation - Connexion et consultation d'un match.....	16
Figure 2: Diagramme de séance - Connexion et visualisation du détail d'un match .....	17
Figure 3: Diagramme d'utilisation – Parier.....	17
Figure 4: Diagramme de séquence - Parier .....	18
Figure 5: Diagramme de séquence - Terminer un match.....	19
Figure 6: Liste des matchs (Application Android).....	21
Figure 7: Détails d'un match (Application Android) .....	22
Figure 8: Connexion et inscription utilisateur (Application Android) .....	23
Figure 9: Profil utilisateur (Application Android) .....	24
Figure 10: Rechargement de compte (Application Android) .....	24
Figure 11: Historique des paris (Application Android) .....	25
Figure 12: Page d'accueil (Site Web Front Office).....	26
Figure 13: Bannière (Site Web Front Office) .....	26
Figure 14 : Liste des paris avec la recherche multicritère (Site Web Front Office).....	26
Figure 15: Liste des différents types de pari (Site Web Front Office) .....	27
Figure 16: Statistiques (Site Web Front Office).....	27
Figure 17: Détails d'un match (Site Web Front Office) .....	28
Figure 18: Profil utilisateur (Site Web Front Office).....	29
Figure 19: Connexion administrateur (Site Web Back Office).....	29
Figure 20: Liste des matchs (Site Web Back Office) .....	30
Figure 21: Création d'un match (Site Web Back Office).....	31
Figure 22: Se connecter (Application Desktop).....	32
Figure 23: Liste des matchs (Application Desktop) .....	33
Figure 24: Les filtres de recherche (Application Desktop) .....	33
Figure 25: Création d'un match (Application Desktop).....	34
Figure 26: Architecture système .....	35
Figure 27: Code Source - vue statique.....	37
Figure 28: Cas d'utilisation - Terminer match .....	38

## Liste des tableaux

Tableau 1: Tableau comparatif des solutions.....	12
Tableau 2: Contraintes et risques sur le projet .....	15
Tableau 3: Diagramme de Gantt – planification .....	16

## Liste des acronymes

API : Application Programming Interface

IDE : Integrated Development Environment

IHM : Interface Homme Machine

MBDS : Mobilité Base de données et intégration de Systèmes

UML : Unified Modeling Language

VPS : Virtual Private Server

MVC : Modèle Vue Controller

## Glossaires

API : Application Programming Interface est un ensemble de classes, méthodes et fonctions permettant d'accéder aux services d'une application, via un langage de programmation.

IDE : Integrated Development Environment, ou environnement de développement intégré en français, un logiciel regroupant plusieurs outils dans le but de faciliter le développement d'une application, d'un projet informatique

IHM : Interface Homme Machine

MBDS : Master MBDS (Mobilité Base de Données et intégration de Systèmes), Master 2

Professionnel - Spécialité Sciences et Technologies de l'Information et de la Communication

UML : Unified Modeling Language, en français : Langage de modélisation unifié spécialement utilisé lors du développement des logiciels et de la conception orientée, il sert à créer divers diagrammes pour visualiser et représenter la conception d'un système

VPS : Virtual Private Server ou Virtual Dedicated Server (appellation commerciale), Serveur dédié virtuel en français, une méthode de partitionnement d'un serveur en plusieurs serveurs virtuels indépendants, chacun possède les caractéristiques d'un serveur dédié en utilisant des techniques de virtualisation

## Table des matières

Résumé .....	2
Liste des figures .....	3
Liste des tableaux .....	4
Liste des acronymes .....	5
Glossaires .....	6
1. Introduction générale.....	10
2. Présentation du projet .....	11
1.1. Présentation de l'entreprise.....	11
1.2. Présentation du sujet et objectif du projet (problème traité) .....	11
3. Etat de l'art sur le sujet traité.....	11
3.1. Critère de comparaison .....	11
3.2. Etude de chaque solution au vu des critères .....	12
3.3. Tableau comparatif des solutions au vu des critères .....	12
4. Etude de l'existant et solution envisagée.....	12
4.1. Etude de l'existant.....	12
4.2. Critique de l'existant .....	12
4.3. Solutions envisagées .....	12
4.4. Objectifs principaux et livrables .....	13
5. Démarche du projet .....	13
5.1. Principes de la démarche du projet .....	13
5.1.1. Activités d'ingénierie logicielle.....	13
5.1.2. Méthode de gestion de projet utilisée.....	14
5.1.3. Rôles et responsabilités.....	14
5.1.4. Outils .....	14
5.1.5. Outils de configuration.....	14
5.2. Contraintes et risques sur le projet.....	14
5.3. Démarche projet et mise en œuvre .....	15
5.4. Planification.....	15
5.5. Budget du pari .....	16
6. Exigences réalisées dans le projet (vision externe/utilisateur) .....	16
6.1. Exigences fonctionnelles – Cas d'utilisation.....	16
6.1.1. Cas d'utilisation – Connexion et consultation d'un match.....	16

6.1.2.	Cas d'utilisation – parier .....	17
6.1.3.	Cas d'utilisation – Terminer un match .....	18
6.2.	Exigences non fonctionnelles transverses.....	19
6.3.	Interfaces détaillées .....	20
6.3.1.	IHM .....	20
6.3.1.1.	Android.....	20
6.3.1.1.1.	Liste des matchs .....	20
6.3.1.1.2.	Détails d'un match.....	21
6.3.1.1.3.	Connexion et inscription utilisateur.....	22
6.3.1.1.4.	Profil utilisateur .....	23
6.3.1.1.5.	Historique des paris .....	25
6.3.1.2.	Front Office Angular (Site Web).....	25
6.3.1.2.1.	Page d'accueil.....	25
6.3.1.2.2.	Détails d'un match.....	27
6.3.1.2.3.	Connexion et inscription utilisateur.....	28
6.3.1.2.4.	Profil utilisateur .....	28
6.3.1.3.	Back Office Angular (Site Web) .....	29
6.3.1.3.1.	Connexion administrateur .....	29
6.3.1.3.2.	Liste des matchs .....	29
6.3.1.3.3.	Création d'un match .....	30
6.3.1.3.4.	Modification, suppression et terminaison d'un match.....	31
6.3.1.4.	Back Office Winform (Application Desktop) .....	31
6.3.1.4.1.	Connexion administrateur .....	31
6.3.1.4.2.	Liste des matchs .....	32
6.3.1.4.3.	Création d'un match .....	34
6.3.1.4.4.	Modification, suppression et terminaison d'un match.....	34
7.	Architecture(s) système .....	35
8.	Conception du système logiciel réalisée dans le projet (vision interne/développeur).....	36
8.1.	Plate-forme technique.....	36
8.2.	Conception du logiciel développé .....	36
8.2.1.	Conception du code source.....	36
8.2.2.	Le code source – vue statique.....	36
8.2.3.	Modélisation de données .....	37
8.2.4.	Réalisation des cas d'utilisation – terminer match .....	38
8.2.5.	Les composants et leur déploiement.....	38
9.	Tests du système logiciel.....	39



10.	Conclusion générale .....	39
11.	Références et Bibliographie .....	39

## 1. Introduction générale

Le présent projet a pour principal objectif de permettre à un utilisateur de faire un pari en ligne sur le sport de football sur différentes plateformes :

Premièrement, il y a l'application web, permettant à un utilisateur de parier depuis son navigateur, de visualiser les matchs disponibles et aussi de consulter le détail d'un match qui l'intéresse comme la date et l'endroit où le match se fera.

Deuxièmement, il y a l'application mobile, permettant à l'utilisateur de consulter les matchs disponibles et les détails de ces matchs n'importe où et n'importe quand, il peut aussi parier depuis son téléphone mobile. Il peut consulter les matchs en mode hors ligne.

Enfin, une application Windows et une application web pour l'administrateur pour gérer et créer les matchs et paris

## 2. Présentation du projet

### 1.1. Présentation de l'entreprise

### 1.2. Présentation du sujet et objectif du projet (problème traité)

Le présent projet a pour principal objectif de permettre à un utilisateur de faire un pari en ligne sur le sport de football sur différentes plateformes :

Premièrement, il y a l'application web, permettant à un utilisateur de parier depuis son navigateur, de visualiser les matchs disponibles et aussi de consulter le détail d'un match qui l'intéresse comme la date et l'endroit où le match se fera.

Deuxièmement, il y a l'application mobile, permettant à l'utilisateur de consulter les matchs disponibles et les détails de ces matchs n'importe où et n'importe quand, il peut aussi parier depuis son téléphone mobile. Il peut consulter les matchs en mode hors ligne.

Enfin, une application Windows et une application web pour l'administrateur pour gérer et créer les matchs et paris

## 3. Etat de l'art sur le sujet traité

### 3.1. Critère de comparaison

Dans cette partie, on va prendre 3 sites selon les critères et après on établira un tableau comparatif de ces 3 sites de paris sportifs en ligne.

Les 3 sites que l'on étudiera sont parmi les meilleurs d'après un article publié dans un site <sup>1</sup>de comparatif des meilleurs sites de pari en ligne le 6 août 2021

- Zebet

Zebet possède des paris sur plus de 10 sports et propose des probabilités de victoire aux joueurs.

- Parions Sports

Parions Sports n'offre que 3 sports (Basketball, Football, Rugby) pour parier mais propose aux joueurs de parier sur un point de vente.

- Winamax

Winamax possède des paris sur plus de 10 sports mais est centrée principalement sur le jeu du poker.

Les 3 sites présentent beaucoup trop d'informations dans leurs sites qui peut perturber les joueurs, surtout les nouveaux parieurs.

Ils manquent des détails sur leur match sur leur match pour l'utilisateur comme la localisation du match.

---

<sup>1</sup> Site de pari en ligne : <https://www.guidedupari.com/meilleur-site-de-pari.html>

### 3.2. Etude de chaque solution au vu des critères

On a montré deux critères de comparaison de Bet Soccer par rapport aux 3 meilleurs sites de pari sportif en ligne.

La première est que les 3 sites présentent trop d'informations superflus et peuvent paraître incompréhensible pour l'utilisateur. Pour pallier ce problème, Bet Soccer propose un affichage des matchs et de parier de façon simplifiée.

La deuxième est que dans le détail d'un match, on ne retrouve pas des informations nécessaires comme la localisation du match. Pour cela Bet Soccer montrera une carte et un QR code à scanner pour de meilleures informations sur le match.

### 3.3. Tableau comparatif des solutions au vu des critères

Voici le tableau comparatif des solutions vu ci-dessus

Critères de comparaison	Solution envisagée	Solution de Bet Soccer
Trop d'informations superflus pour les 3 sites (Zebet, Parions Sport, Winamax)	Des affichages plus simplifiés	Affichage des matchs avec les informations nécessaires et minimum à l'utilisateur
Le détail d'un match manque d'informations sur le match comme la localisation	Affichage du lieu du match sur une carte	Ajout d'un module de mapping (carte) dans les détails de matchs, QR code pour mieux récupérer ses informations.

Tableau 1: Tableau comparatif des solutions

## 4. Etude de l'existant et solution envisagée

### 4.1. Etude de l'existant

Jusqu'ici nous n'avons pas d'application de pari sportif en ligne ou de base pour commencer.

Nous avons juste des idées sur les préférences des parieurs concernant les paris sportifs en ligne.

C'est pourquoi nous avons décidé de créer l'application Bet Soccer à partir de ses idées.

### 4.2. Critique de l'existant

Comme vue dans la partie 4.1, nous n'avons aucune base pour commencer le projet à part les sites concurrents présentés dans la partie 3.1, qui présentent des problématiques pour les nouveaux parieurs du fait de leurs informations superflus

### 4.3. Solutions envisagées

A cause des problématiques de la section 4.2, nous avons décidé de créer plusieurs applications simples pour permettre aux joueurs de parier sereinement et sur n'importe quel platform (Web ou Android)

#### 4.4. Objectifs principaux et livrables

L'objectif principale étant de créer une application multi-plateforme.

Les livrables seront :

- Application Android

1 fichier .APK pour l'application mobile Android disponible sur :

<https://drive.google.com/file/d/1o0g2Z2BGtAKOCTfZ--cGNA9KZhmbfn3j/view?usp=sharing>

- 2 sites hébergés en ligne

1 site en ligne pour le front office (coté client) disponible sur : <https://pari-angular.herokuapp.com/>

1 site en ligne back office (côté administrateur) disponible sur : <https://pari-angular-back-office.herokuapp.com/>

- Une application .exe (Windows)

Une application desktop pour l'administrateur disponible sur :

<https://drive.google.com/file/d/1nYIdwHDXYKmf53aIjd5949kse20wg-2y/view?usp=sharing>

- 2 serveurs webservices hébergés en ligne :

Node.js disponible sur : <https://pari-node.herokuapp.com/>

Grails disponible sur : <https://mbdsp7-grails-app.herokuapp.com/>

- 2 bases de données hébergées en ligne : Oracle et MongoDB

### 5. Démarche du projet

#### 5.1. Principes de la démarche du projet

##### 5.1.1. Activités d'ingénierie logicielle

Le domaine de connaissance du génie logiciel couvre en particulier le cycle de la vie d'un logiciel, les activités clés du cycle de vie depuis la demande d'un maître d'ouvrage jusqu'à la mise hors service définitive du produit. Voici les listes de l'activité d'ingénierie concernée :

- Analyse des besoins : Définir quelles sont les problématiques existantes, c'est-à-dire déterminer les points faibles du système courant pour ensuite offrir de meilleures solutions pour remédier à ces faiblesses.
- Conception : Consiste à déterminer et schématiser les grandes lignes des mécanismes qui devront être programmés en vue de satisfaire les exigences requises à la réalisation du projet.
- Gestion de projets : Une activité réalisée tout au long des travaux sur le projet, qui consiste à organiser l'équipe, répartir les tâches et veiller à l'avancée des travaux en vue de finir dans les délais prévus.

- Test : Une suite de vérifications qui servent à déceler un maximum de bugs, des défauts de programmation qui provoquent des pannes ou des résultats incorrects.

#### 5.1.2. Méthode de gestion de projet utilisée

Nous avons adopté la méthode SCRUM ou bien méthode agile, ce dernier implique au maximum le demandeur (client) et permet une grande réactivité à ses demandes. La réalisation du projet a été divisée en sprints de deux semaines suivant le planning estimé. En cas de retard ou d'imprévu, les heures supplémentaires étaient nécessaires, tout ça pour que le planning de réalisation soit suivi. Concernant la méthode de communication, on organisait des réunions Skype afin d'étudier les facteurs bloquant à la réalisation technique du projet.

#### 5.1.3. Rôles et responsabilités

Nous sommes 3 à se pencher sur le projet. Nous nous chargeons de développer, déployer les différentes applications et de les tester.

Nous avons aussi élaboré ensemble la conception et l'architecture du projet

#### 5.1.4. Outils

Les technologies et outils qu'on a utilisé lors du développement de l'application sont les suivantes :

- **Angular** : Un Framework officiel de Javascript. Développé pour le côté client, open source, basé sur TypeScript. Avec Visual Studio Code ou Web Storm comme IDE
- **Visual Studio** : un logiciel permettant de développer des applications .NET notamment C# Winform pour l'application Windows
- **Android Studio** : Android Studio est un IDE gratuit, spécialisé pour le développement des applications Android Native et Java. Il présente beaucoup de communauté dorénavant.
- **Postman** : Un outil qui nous a servi à exécuter des appels HTTP directement depuis une interface graphique pour tester les fonctionnalités depuis la partie Back End

#### 5.1.5. Outils de configuration

Concernant la gestion des livrables, voici donc les spécifications :

- Les codes sources devront être mis en ligne sur GitHub
- Les sites et base de données devront être déployés en ligne sur Heroku
- Les comptes administrateurs des applications doivent être fournis au moment de livrer

### 5.2. Contraintes et risques sur le projet

Libellé du risque	Facteur contribuant	Solution envisagée	STATUT (en cours, réalisé, solution choisie...)
Déploiement de base de données Oracle en ligne.	Le manque de matériel pour héberger la base. La plupart des VPS sont payants.	Utiliser Oracle Free Cloud Tier.	Utiliser la base de données fournie par l'Université Nice Sophia Antipolis

Tableau 2: Contraintes et risques sur le projet

### 5.3. Démarche projet et mise en œuvre

Dans cette partie, nous verrons le découpage en étapes du projet et on les détaillera une à une :

#### – **Étape 1 : Spécification des exigences**

##### • **Objectifs**

Dans cette étape, on déterminera surtout la vision externe du projet, c'est-à-dire, répondre à cette question : Qu'attendons-nous du système ?

##### • **Tâches à faire**

Définir les exigences fonctionnelles :

- ✓ Quelles sont les informations à traiter et d'où proviennent-elles ?
- ✓ Quelles sont les fonctionnalités attendues ?
- ✓ Qui fait quoi ?

Définir les exigences non fonctionnelles

#### – **Étape 2 : Conception**

Faire la conception des bases de données. Cette étape est nécessaire avant le développement des fonctionnalités

#### – **Étape 3 : Mise en ligne**

Tester le déploiement des sites et des bases de données en ligne

#### – **Étape 4 : Développement**

Programmer une à une les fonctionnalités et les tester.

#### – **Étape 5 : Tests**

Tester toutes les applications du projet et corriger les éventuels retours

### 5.4. Planification

Voici les grandes lignes du diagramme de GANTT du projet


			
Nom		Date de début	Date de fin
☐	• Creation des prototypes	10/05/2021	17/05/2021
	• Conception	10/05/2021	11/05/2021
	• Creation des base de donnees	11/05/2021	17/05/2021
	• Premier deploiement des sites	10/05/2021	12/05/2021
	• Creation des repository Git	10/05/2021	10/05/2021
	• Developpement des fonctionnalites	18/05/2021	09/08/2021
	• Test globale des fonctionnalites	10/08/2021	10/08/2021
	• Correction des retours	10/08/2021	12/08/2021
	• Livraison	16/08/2021	16/08/2021

Tableau 3: Diagramme de Gantt – planification

### 5.5. Budget du pari

Le budget nécessaire à l'élaboration du projet est le montant du tarif de connexion des développeurs soit 200.000 Ar/mois soit 43,47€/mois (devise du 16 août 2021).

## 6. Exigences réalisées dans le projet (vision externe/utilisateur)

### 6.1. Exigences fonctionnelles – Cas d'utilisation

Nous allons parler des exigences fonctionnelles et non fonctionnelles réalisées dans le projet.

#### 6.1.1. Cas d'utilisation – Connexion et consultation d'un match

Cas d'utilisation pour simuler la connexion et la consultation du détail d'un match

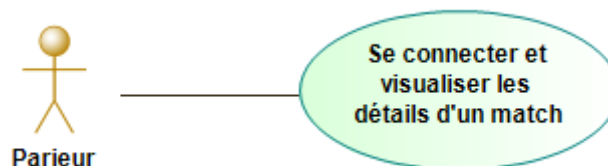


Figure 1 : Cas d'utilisation - Connexion et consultation d'un match

#### Scénario normal : Se connecter et visualiser les détails d'un match

Le scénario décrit les cas d'utilisation pour simuler la connexion et la visualisation des détails d'un match, les utilisateurs, qu'ils soient membres ou pas peuvent visualiser les détails d'un match ; ce cas d'utilisation commence lorsqu'un utilisateur est dans la page de connexion de l'application :



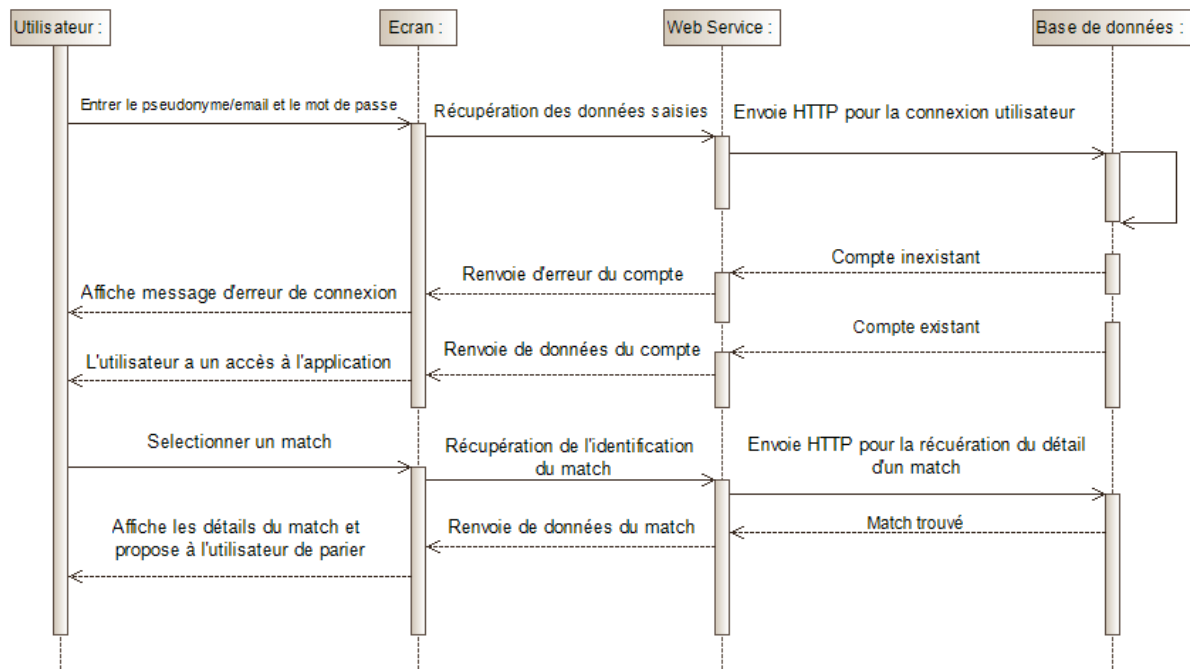


Figure 2: Diagramme de séance - Connexion et visualisation du détail d'un match

### 6.1.2. Cas d'utilisation – parier

#### Cas d'utilisation pour parier

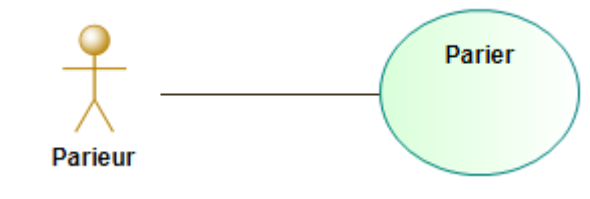


Figure 3: Diagramme d'utilisation – Parier

#### Scénario normal : parier

Ce scénario décrit les cas d'utilisation pour simuler la démarche pour parier sur un des paris d'un match proposé, ce cas d'utilisation commence lorsque l'utilisateur est sur l'écran du détail d'un match

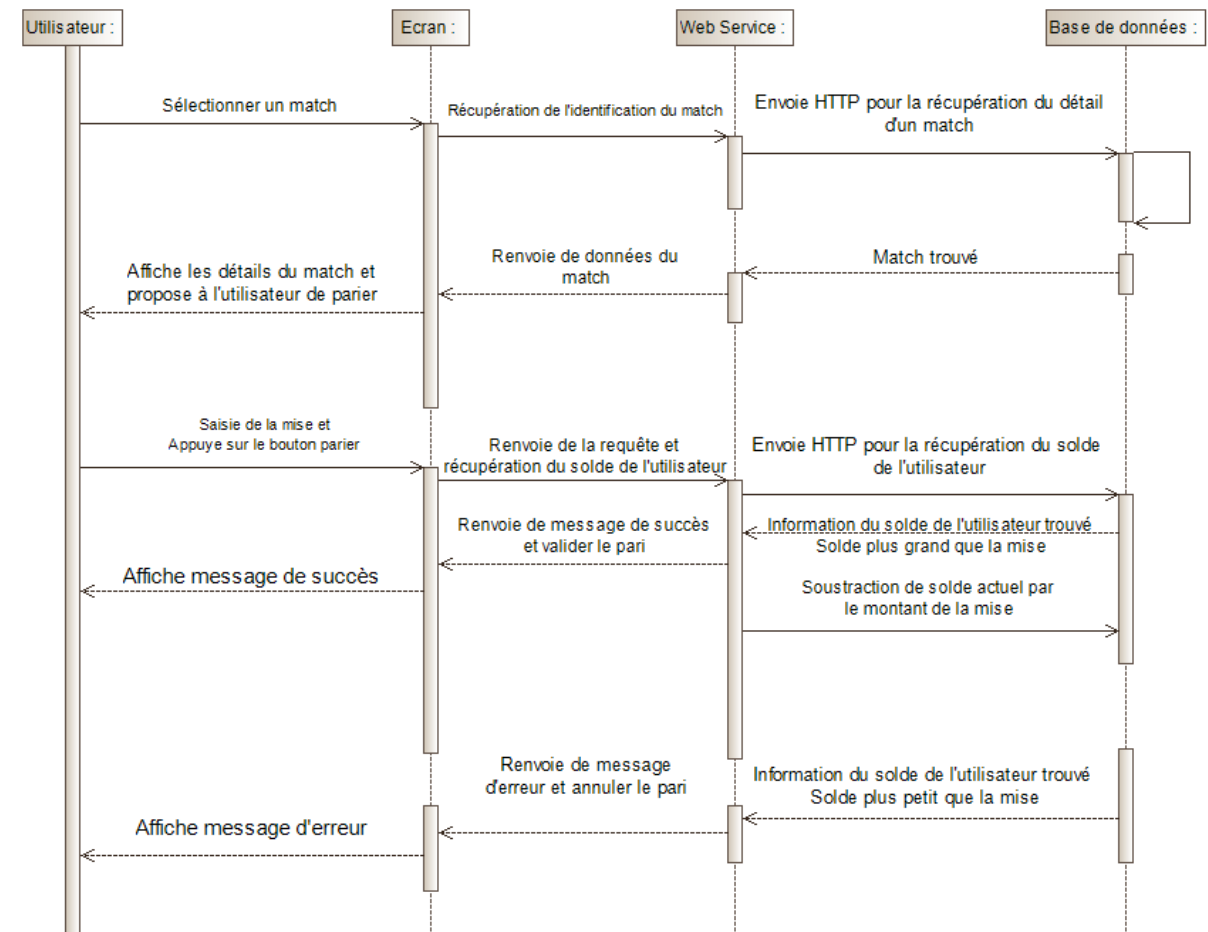


Figure 4: Diagramme de séquence - Parier

### 6.1.3. Cas d'utilisation – Terminer un match

Ce scénario décrit les cas d'utilisation pour simuler la distribution des gains et de terminer un match par l'administrateur. Ce cas d'utilisation commence lorsque l'administrateur est sur l'écran de la liste des matchs :

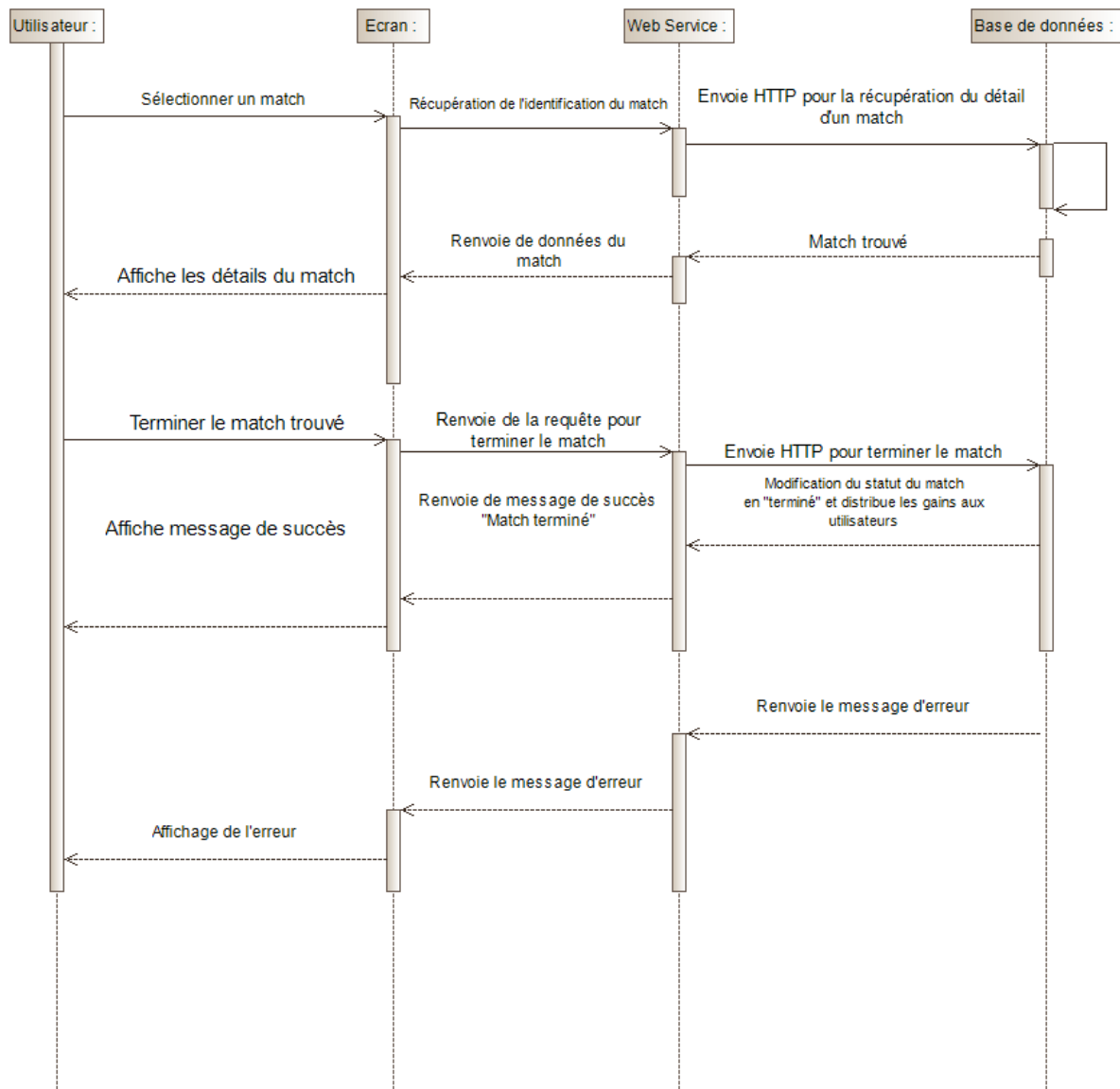


Figure 5: Diagramme de séquence - Terminer un match

## 6.2. Exigences non fonctionnelles transverses

Les exigences « non fonctionnelles » sont les exigences, les critiques pour les utilisateurs qui assurent le fonctionnement du système.

### Utilisabilité :

- Les fonctionnalités sont accessibles aux utilisateurs membres (parieurs) et les visiteurs
- L'application est accessible par plusieurs utilisateurs à la fois
- L'application est très facile à comprendre avec des interfaces intuitives.

#### Disponibilité :

- L'application est accessible à distance : disponible 24 heures sur 24

#### Performance :

- Optimisation de la base de données Oracle et puis la mise en place de la pagination depuis la base de données.

#### Maintenabilité :

- Facilité de déploiement
- Utilisable avec différents systèmes d'exploitation

### 6.3. Interfaces détaillées

#### 6.3.1. IHM

Ci-dessous la liste des interfaces de chaque plateforme

##### 6.3.1.1. Android

##### 6.3.1.1.1. Liste des matchs

La liste des matchs est la page d'accueil de l'application. Il s'agit de la liste des matchs selon les filtres de recherches (aucun par défaut). On peut rechercher un match selon les filtres suivant : entre deux dates, match d'aujourd'hui, match terminé ou selon un critère de recherche sur une équipe ou un pari. Il sera possible de consulter des matchs en cours (100 maximum) en mode hors ligne.

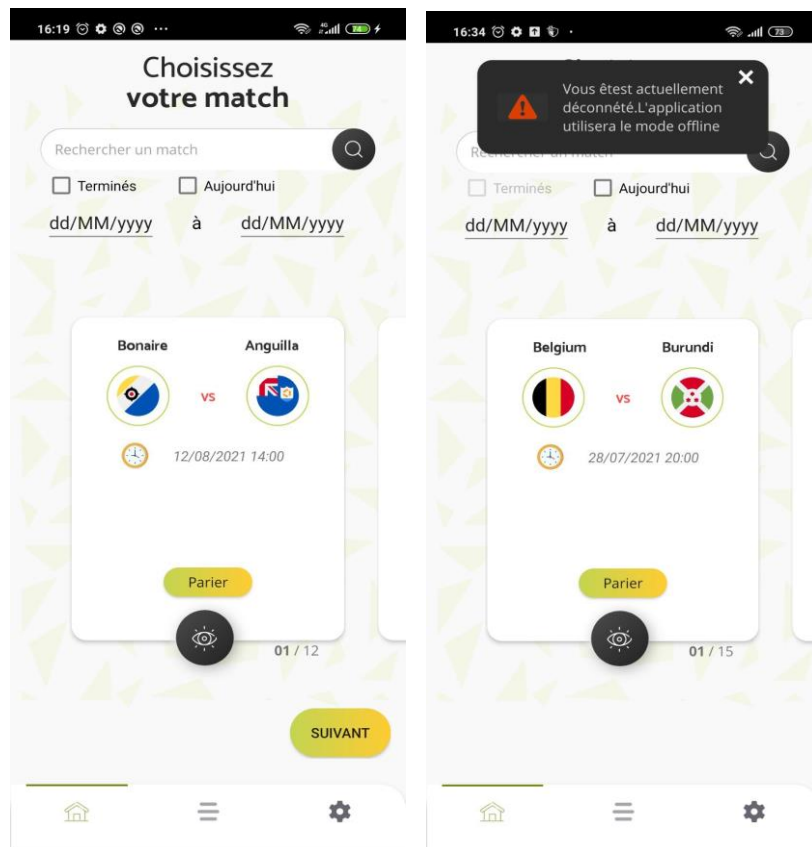


Figure 6: Liste des matchs (Application Android)

#### 6.3.1.1.2. Détails d'un match

Lorsque l'on clique sur "parier" ou un match dans la liste des matchs, un nouvel écran s'affiche contenant les informations d'un match comme les deux équipes qui s'affrontent, la date du match et la liste des paris avec un bouton parier chacun et le QR code concernant le match. Le bouton « parier » ne seront utilisables que si l'utilisateur est connecté. Dans le cas où l'utilisateur est connecté et que son solde est insuffisant par rapport à sa mise, un message d'erreur est alors affiché.



Figure 7: Détails d'un match (Application Android)

#### 6.3.1.1.3. Connexion et inscription utilisateur

Un écran de connexion sera affiché si l'utilisateur clique sur « Connexion » dans la section Profil utilisateur ou qu'il tente de parier ou insérer de l'argent (voir section Profil utilisateur) sans être connecté. Dans l'écran de connexion, un bouton "inscription" pour rediriger vers l'écran d'inscription.

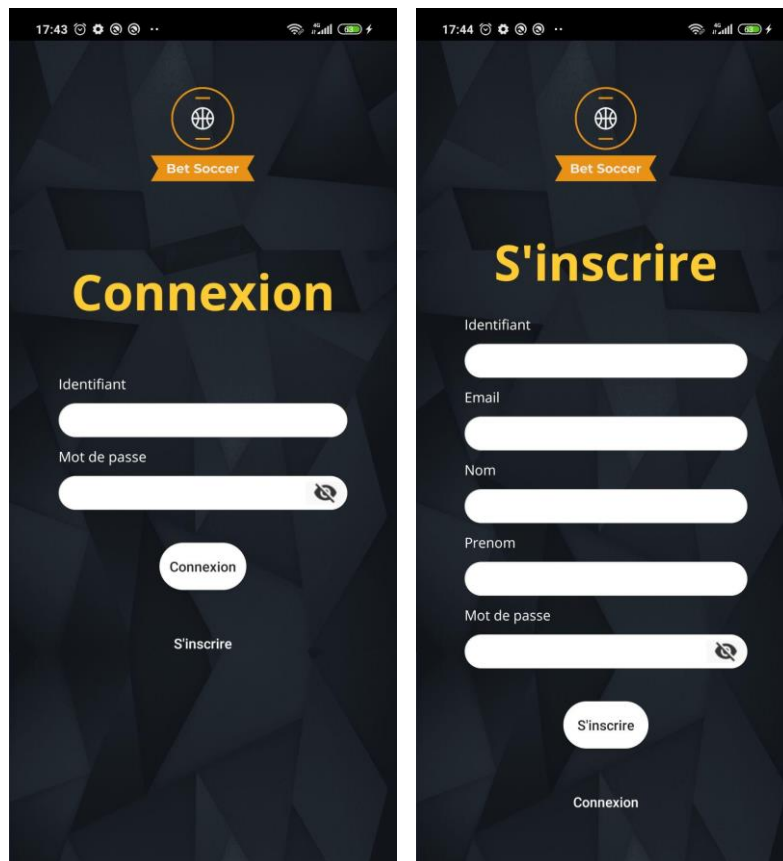



Figure 8: Connexion et inscription utilisateur (Application Android)

#### 6.3.1.1.4. Profil utilisateur

Un écran paramètre sera affiché si on clique sur  dans le menu en bas. Les informations sur l'utilisateur seront affichées s'il est connecté

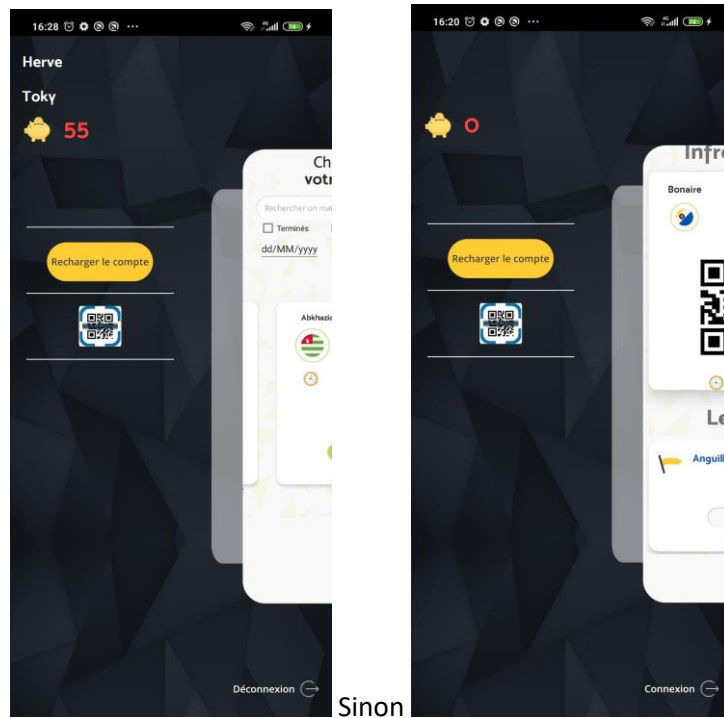


Figure 9: Profil utilisateur (Application Android)




Dans cet écran, le bouton “Connexion” deviendra "Déconnexion" si le client est connecté. Le bouton  permet de lancer un scan de QR code qui va rediriger vers un match. Le bouton  permet de recharger le compte de l'utilisateur.



Figure 10: Rechargement de compte (Application Android)



#### 6.3.1.1.5. Historique des paris

Si l'utilisateur est connecté, il peut consulter son historique de paris en cliquant sur l'icône  .  
L'historique peut être affiché sous deux types : les paris en cours et terminés.

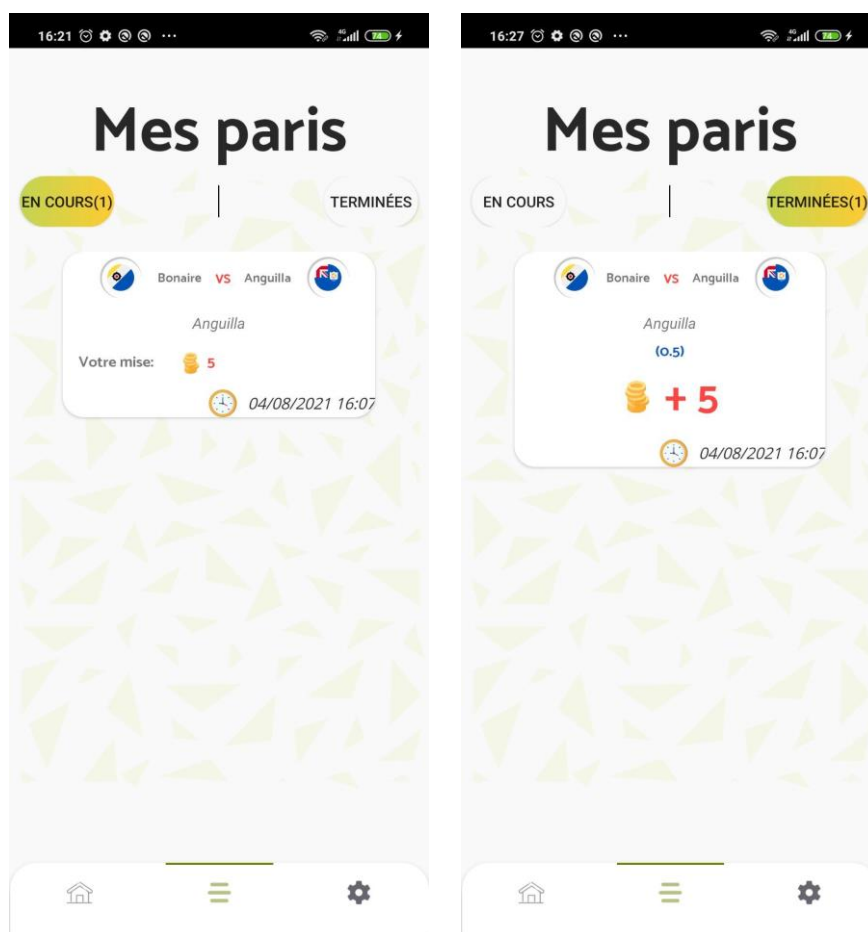


Figure 11: Historique des paris (Application Android)

#### 6.3.1.2. Front Office Angular (Site Web)

##### 6.3.1.2.1. Page d'accueil

La page d'accueil regroupe plusieurs sections.

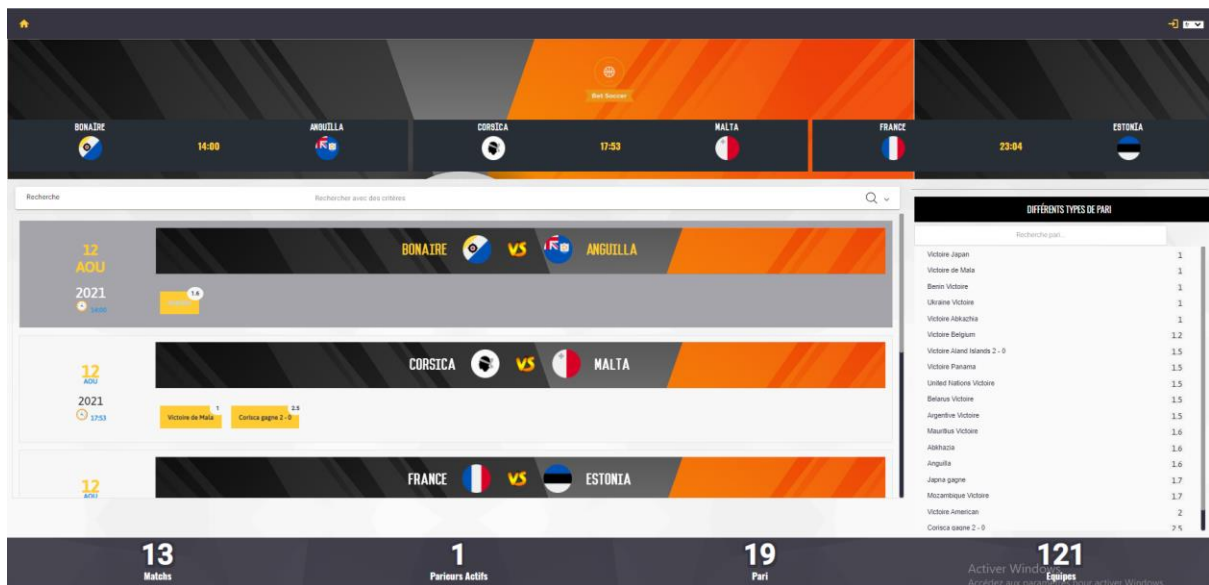


Figure 12: Page d'accueil (Site Web Front Office)

La bannière qui contient la liste des matchs du jour et un bouton de connexion ou déconnexion de l'utilisateur en haut à droite et le choix de la langue.

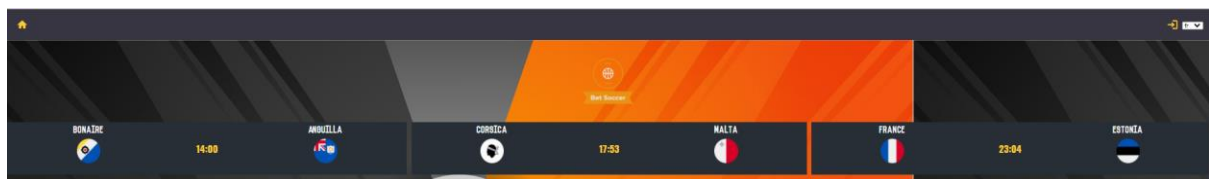


Figure 13: Bannière (Site Web Front Office)

La liste des matchs en fonction de filtres de recherches (nom de l'équipe ou texte du pari et entre deux dates) peut être déplié. Par défaut, sans critère, les 6 prochains matchs seront affichés.

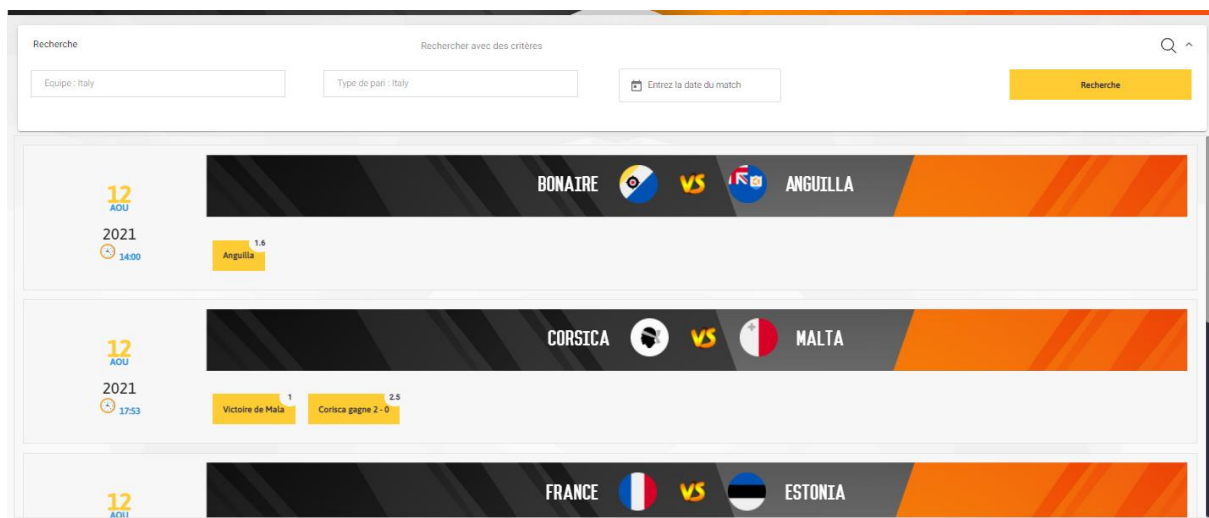


Figure 14 : Liste des paris avec la recherche multicritère (Site Web Front Office)

DIFFÉRENTS TYPES DE PARI	
Recherche pari...	
Victoire Japan	1
Victoire de Mala	1
Benin Victoire	1
Ukraine Victoire	1
Victoire Abkazia	1
Victoire Belgium	1.2
Victoire Aland Islands 2 - 0	1.5
Victoire Panama	1.5
United Nations Victoire	1.5
Belarus Victoire	1.5
Argentine Victoire	1.5
Mauritius Victoire	1.6
Abkhazia	1.6
Anguilla	1.6
Japna gagne	1.7
Mozambique Victoire	1.7
Victoire American	2
Corisca qagne 2 - 0	2.5

Figure 15: Liste des différents types de pari (Site Web Front Office)

Et un pied de page regroupant le nombre de matchs, le nombre de parieurs actifs, le nombre de paris et le nombre d'équipes.

<b>13</b> Matches	<b>1</b> Parieurs Actifs	<b>19</b> Paris	<b>121</b> Equipes
----------------------	-----------------------------	--------------------	-----------------------

Figure 16: Statistiques (Site Web Front Office)

#### 6.3.1.2.2. Détails d'un match

En cliquant sur un match dans la page d'accueil on est redirigé dans la page des détails d'un match

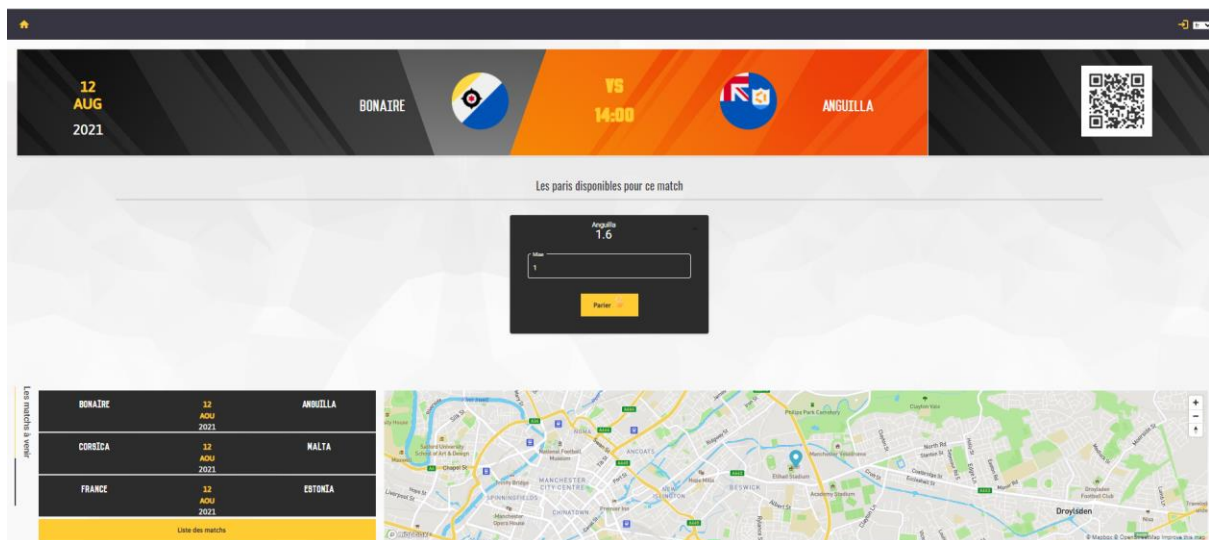










Figure 17: Détails d'un match (Site Web Front Office)

Les détails concernant les deux équipes, la date de match, le QR Code du match et la localisation sur la carte du match mais aussi la possibilité de parier sur une des listes de paris. L'utilisateur ne peut parier que s'il est connecté.

#### 6.3.1.2.3. Connexion et inscription utilisateur

Si l'utilisateur n'est pas connecté, le bouton  permet de le monter un formulaire de connexion pour se connecter ou s'inscrire. S'il est connecté, de nouveaux boutons apparaissent à la place

     est le solde de l'utilisateur.  permet de rediriger vers son profil et  pour le déconnecter

#### 6.3.1.2.4. Profil utilisateur

Le page de profil permet de visualiser son solde, ses historiques de paris et d'ajouter son solde.

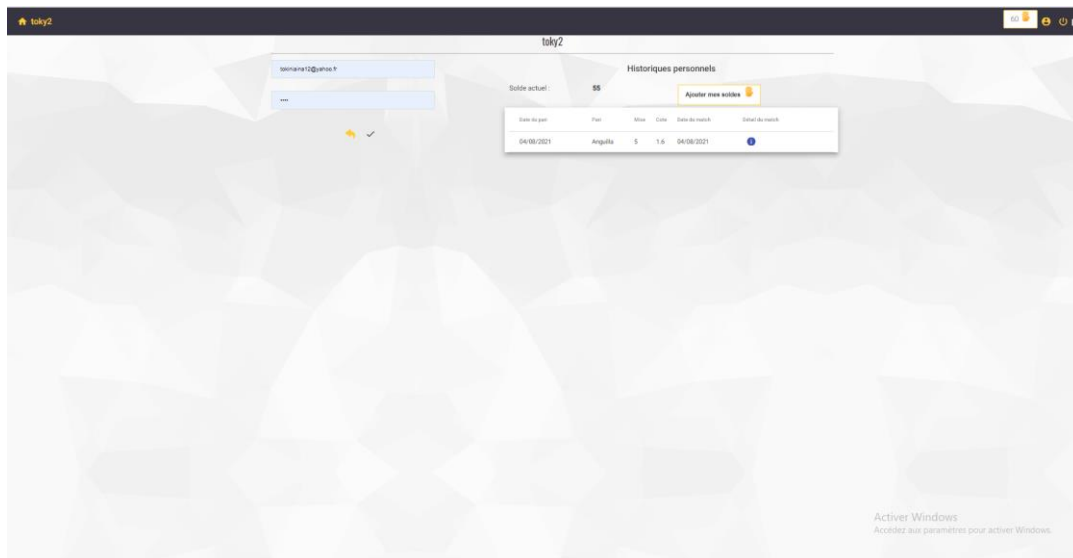


Figure 18: Profil utilisateur (Site Web Front Office)

### 6.3.1.3. Back Office Angular (Site Web)

#### 6.3.1.3.1. Connexion administrateur

Une connexion en tant qu'administrateur est requise pour accéder au site.



Figure 19: Connexion administrateur (Site Web Back Office)

#### 6.3.1.3.2. Liste des matchs

Après la connexion, l'écran contenant la liste des matchs avec pagination est affiché.

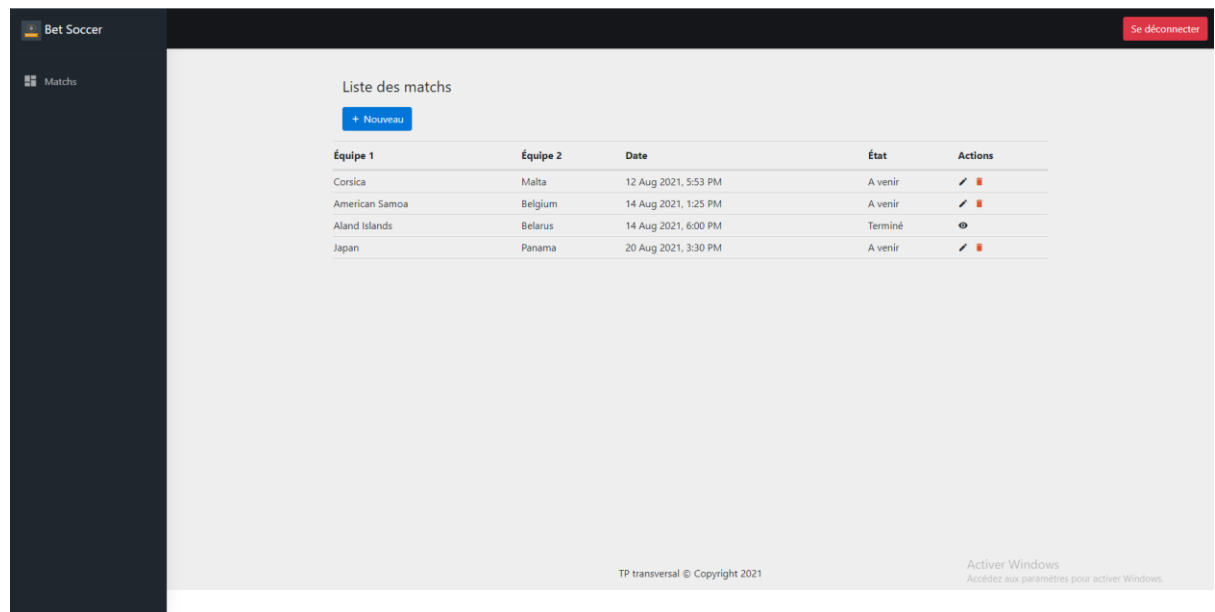


Figure 20: Liste des matchs (Site Web Back Office)

Il existe deux types de matchs, les matchs à venir et les matchs terminés. Seuls les matchs à venir peuvent être modifiés ou supprimés. Les matchs terminés ne peuvent qu'être consultés. L'icône permet de modifier un match, permet de supprimer le match et de consulter les détails du match. Et enfin le bouton pour déconnecter l'administrateur

#### 6.3.1.3.3. Création d'un match

En cliquant sur nouveau dans la liste des matchs, on est redirigé vers un formulaire de création de matchs. Les informations nécessaires pour une création de match sont : 2 équipes adverses différentes, latitude et longitude du lieu du match sur la carte, la date du match et la possibilité d'insérer des paris. L'enregistrement se fait lors de l'appui du bouton et une erreur sera produite si les données sont invalides. Lors de la première création de match le bouton restera désactivée.

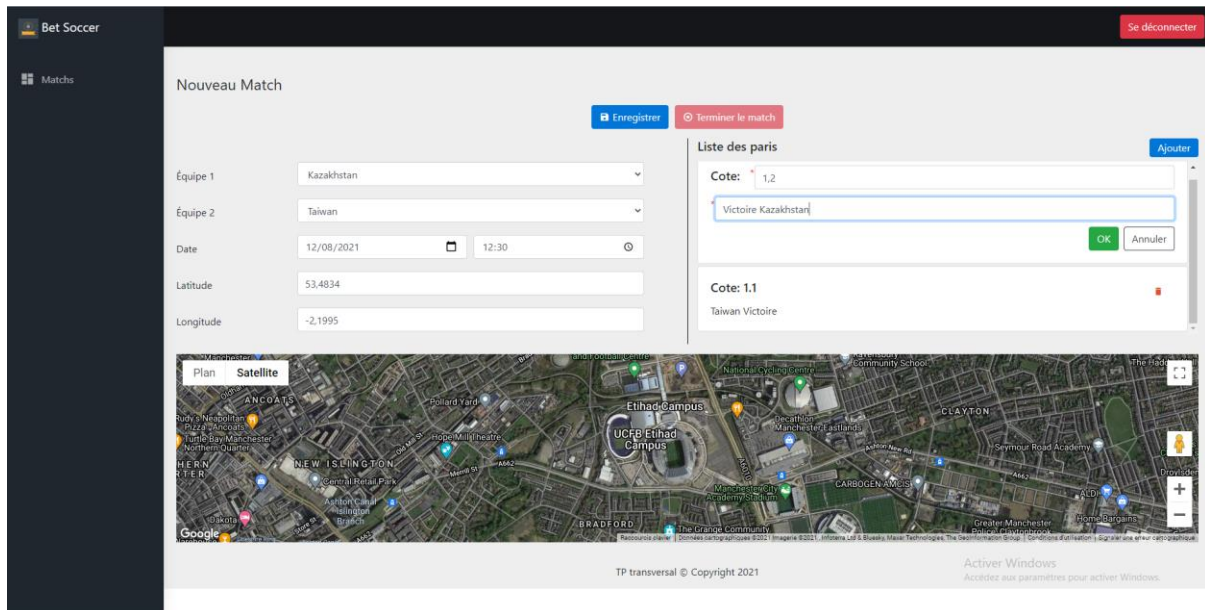

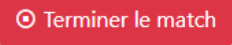


Figure 21: Création d'un match (Site Web Back Office)

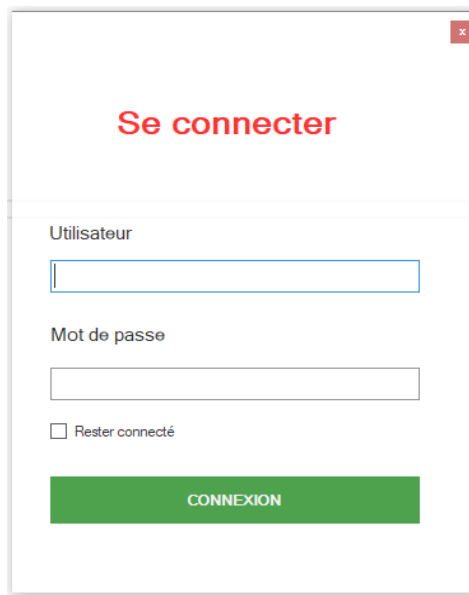
#### 6.3.1.3.4. Modification, suppression et terminaison d'un match

En cliquant sur  dans la liste des matchs, on retourne vers l'écran de création qui permettra de modifier le match. Lorsque l'on modifie un match, le bouton  est disponible et permet de terminer le match et distribuer les gains aux parieurs en fonction des paris gagnants. Seuls les matchs en cours peuvent être modifiés. Le gain est calculé en fonction de la cote et la mise du parieur  $gain = cote \times mise$

#### 6.3.1.4. Back Office Winform (Application Desktop)

##### 6.3.1.4.1. Connexion administrateur

Une connexion en tant qu'administrateur est requise pour accéder à l'application.



The image shows a desktop application window for logging in. The window has a title bar with a close button in the top right corner. The main heading is 'Se connecter' in red. Below it, there are two input fields: 'Utilisateur' (User) and 'Mot de passe' (Password). The 'Utilisateur' field has a blue border, and the 'Mot de passe' field has a grey border. Below the password field is a checkbox labeled 'Rester connecté' (Stay connected). At the bottom is a green button labeled 'CONNEXION'.

Figure 22: Se connecter (Application Desktop)

#### 6.3.1.4.2. Liste des matchs

Après la connexion, l'écran contenant la liste des matchs avec pagination est affiché.



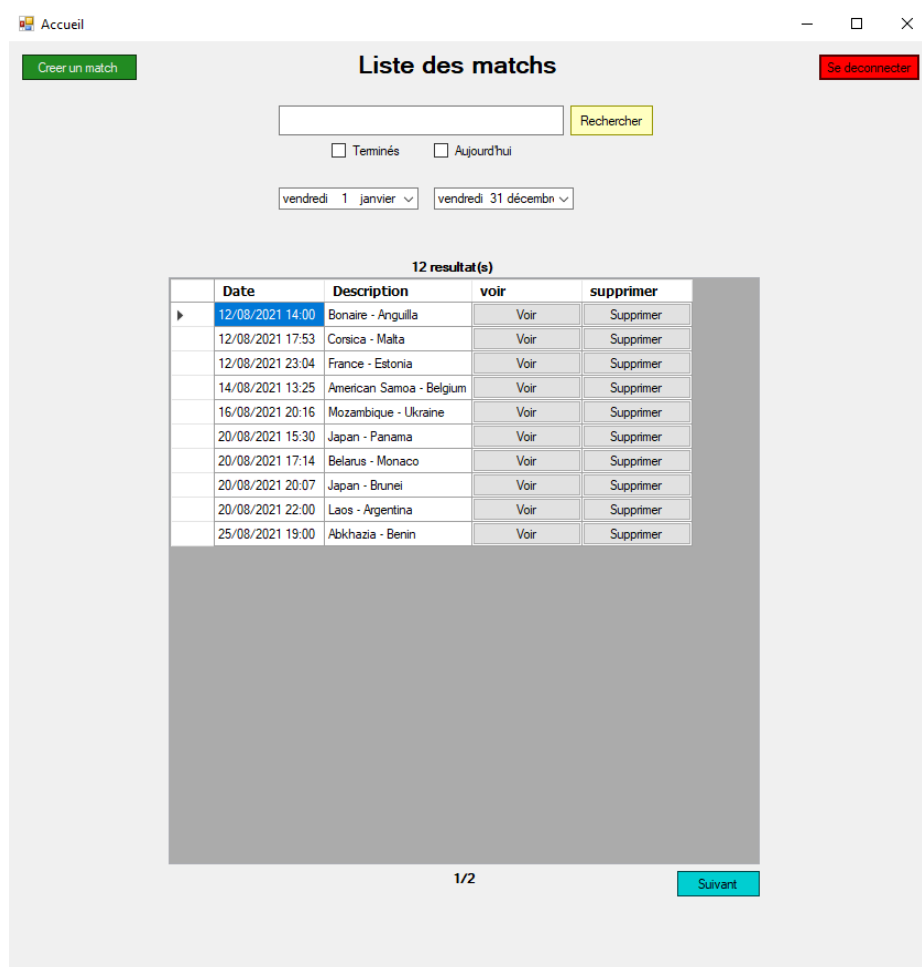

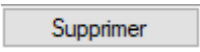



Figure 23: Liste des matchs (Application Desktop)

L'icône  permet de consulter un match et le modifier s'il n'est pas terminé,  permet de supprimer le match. Et enfin le bouton  pour déconnecter l'administrateur.

On peut rechercher un match selon les filtres suivants : entre deux dates, match d'aujourd'hui, match terminé ou selon un critère de recherche sur une équipe ou un pari.

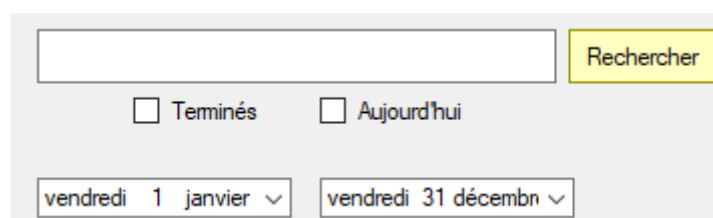




Figure 24: Les filtres de recherche (Application Desktop)

#### 6.3.1.4.3. Création d'un match

En cliquant sur nouveau  dans la liste des matchs, on est redirigé vers un formulaire de création de matchs. Les informations nécessaires pour une création de match sont : 2 équipes adverses différentes, latitude et longitude du lieu du match sur la carte, la date du match et la possibilité d'insérer des paris. Lors de la première création de match le bouton  restera désactivée.

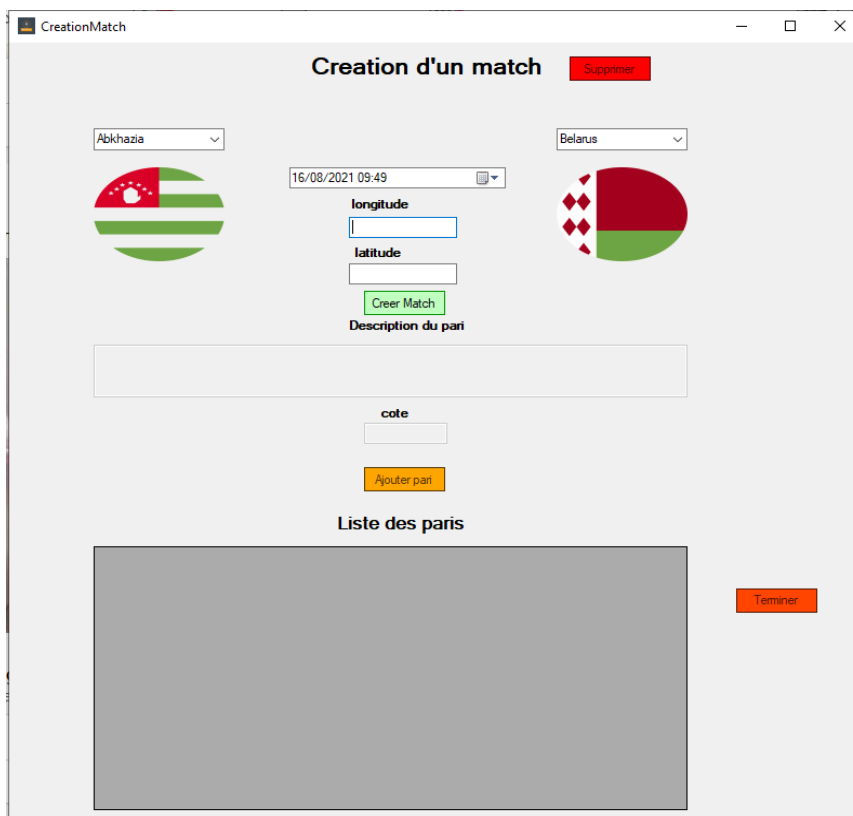




Figure 25: Création d'un match (Application Desktop)

#### 6.3.1.4.4. Modification, suppression et terminaison d'un match

En cliquant sur  dans la liste des matchs, on retourne vers l'écran de création qui permettra de modifier le match. Lorsque l'on modifie un match, le bouton  est disponible et permet de terminer le match et distribuer les gains aux parieurs en fonction des paris gagnants. Seuls les matchs en cours peuvent être modifiés. Le gain est calculé en fonction de la cote et la mise du parieur  $gain = cote \times mise$ .

## 7. Architecture(s) système

Voici l'architecture du système. Une architecture multi tiers

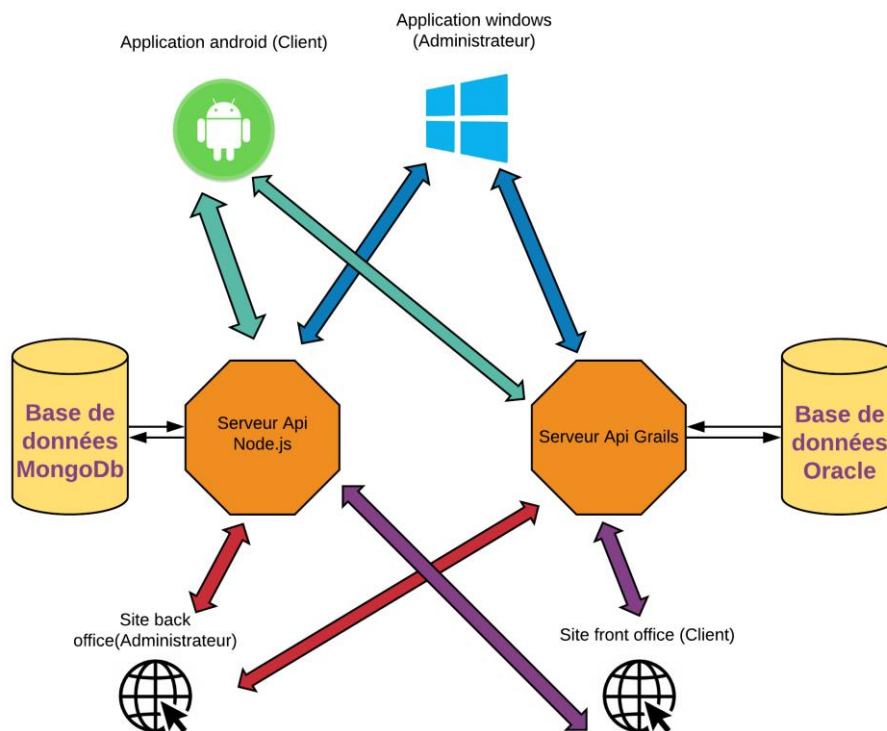


Figure 26: Architecture système

- 1 application mobile Android développé en JAVA
- 1 application Windows développé en .NET C# Winform
- 1 serveur Api développée en Node.js (Framework Javascript)
- 1 serveur Api développée en Grails (Framework JAVA)
- 2 bases de données Oracle et MongoDB pour stocker les données
- 1 site front office pour le client et 1 site backoffice pour l'administrateur, développés en Angular (Framework TypeScript)

## 8. Conception du système logiciel réalisée dans le projet (vision interne/développeur)

### 8.1. Plate-forme technique

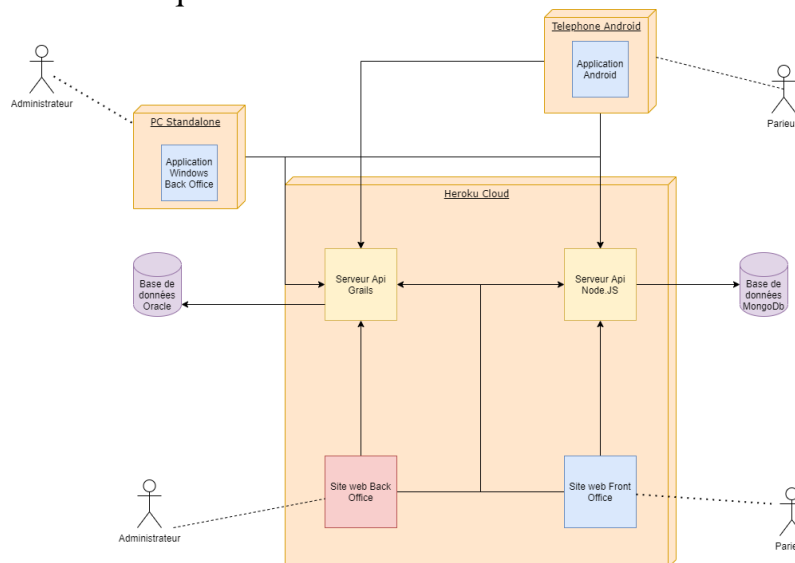


Figure 27: Plate-forme technique

Ci-dessus se présente l'architecture logiciel du système au grand complet. On peut remarquer le besoin de ressources matérielles pour utiliser pleinement Bet Soccer :

- 1 ou plusieurs clients utilisant un téléphone Android ou son navigateur
- 1 ou plusieurs administrateurs utilisant son navigateur ou un ordinateur Windows pour gérer et créer les matchs
- 2 serveurs api pour permettre d'accéder aux bases de données, non visible par les utilisateurs ou administrateurs
- 2 bases de données oracle et MongoDB pour stocker les données

### 8.2. Conception du logiciel développé

#### 8.2.1. Conception du code source

Les codes sources suivent le modèle MVC (Modèle Vue Contrôleur) pour mieux séparer les logiques métiers, et de vues. Pour cela nous avons séparé chaque couche par des packages dans les 5 différentes applications.

#### 8.2.2. Le code source – vue statique

L'architecture du code source des 5 applications sont différentes.

Il existe 4 couches :

- **Vue** : Une couche pour afficher l'écran à l'utilisateur
- **Controller** : couche permettant d'appeler une vue

- **Modèles** : couche qui contient les classes du mapping des tables de la base de données
- **Service** : couche permettant de récupérer les données depuis la base de données au webservice

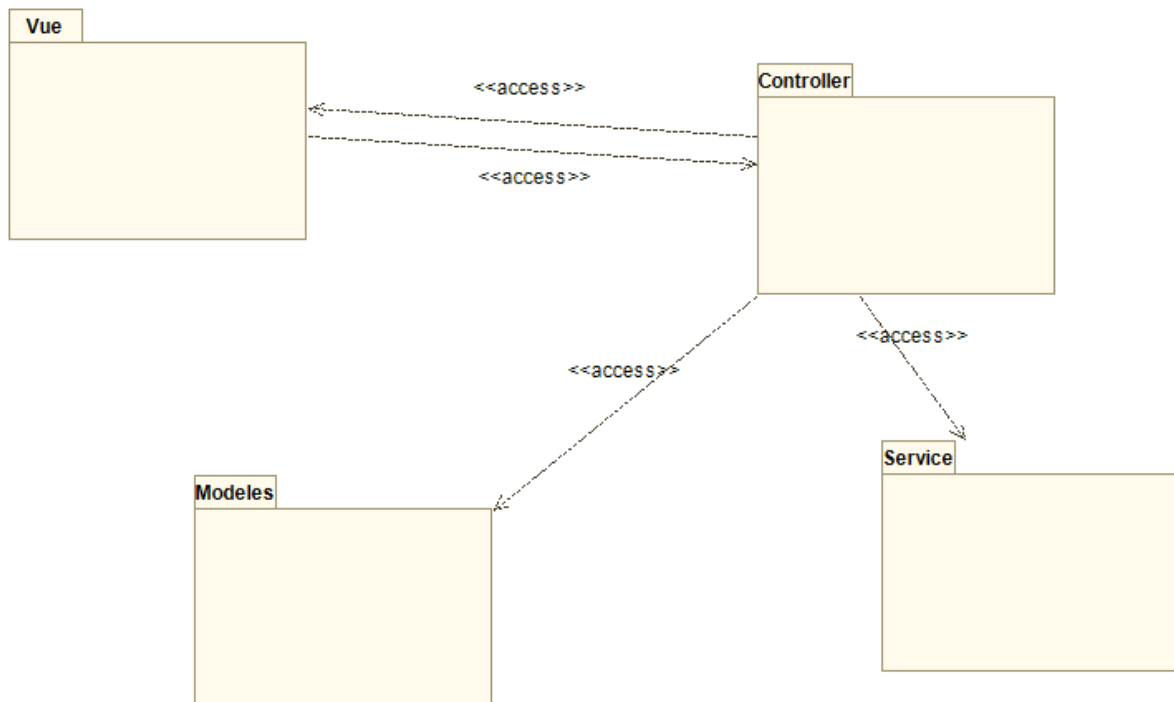


Figure 28: Code Source - vue statique

Pour les 2 serveurs Webservices Grails et Node.js, la couche vue n'existe pas.

### 8.2.3. Modélisation de données

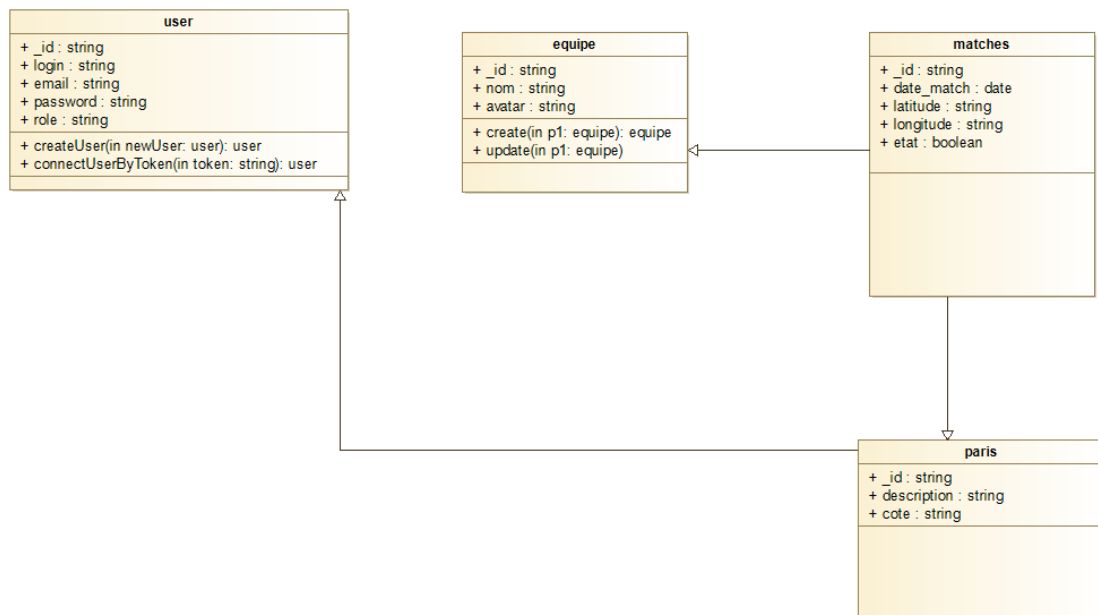


Figure 29: Diagramme de classe

### 8.2.4. Réalisation des cas d'utilisation – terminer match

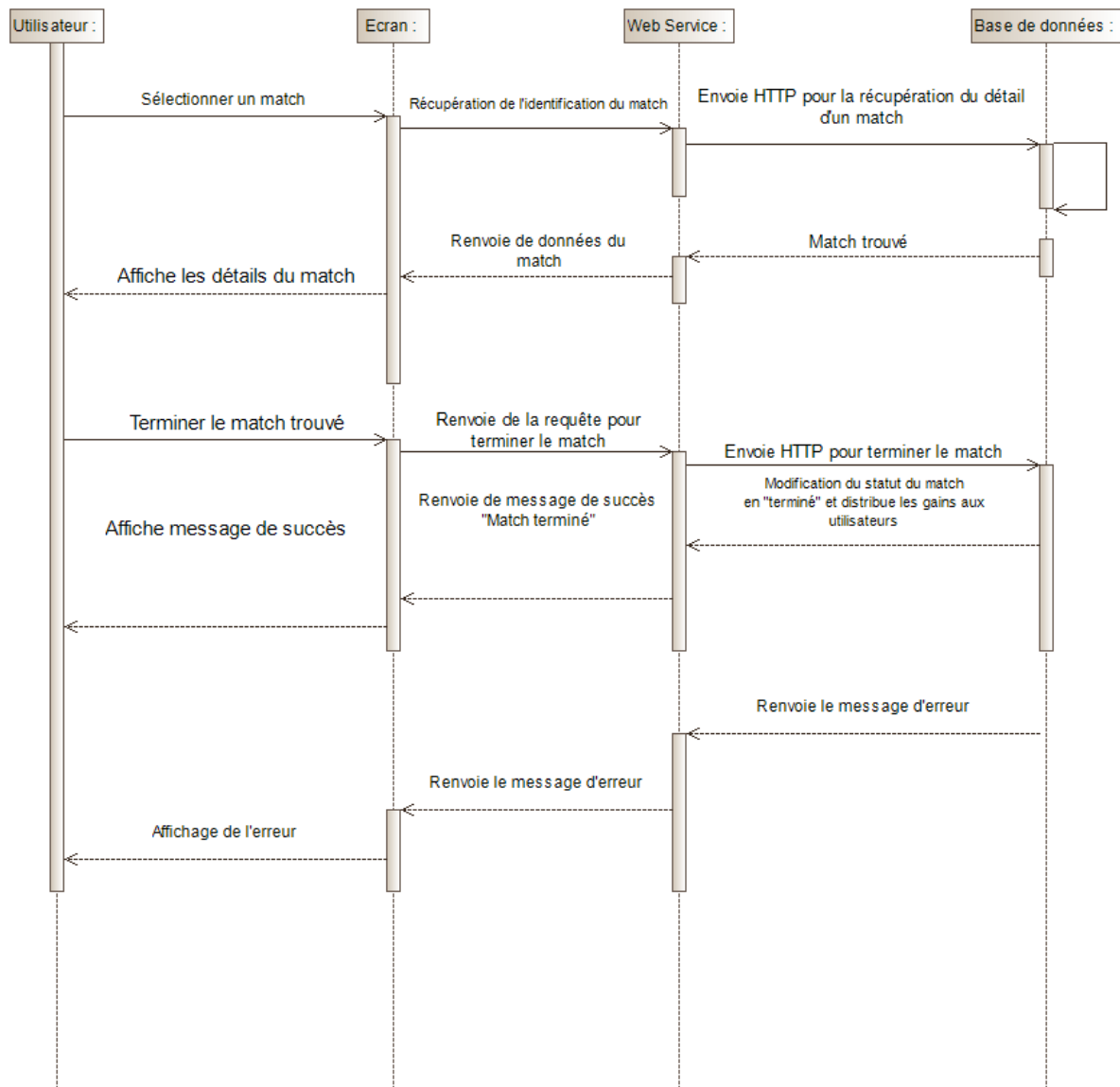


Figure 30: Cas d'utilisation - Terminer match

- Du côté administrateur, sur l'application Windows ou site administrateur, l'administrateur recherche le match à terminer. Ensuite dans le détail du match il sélectionne le ou les paris gagnants et décide de terminer le match ce qui enverra une requête au webservice pour changer l'état du match en terminé.
- Du côté client, sur l'application Android ou site client, si le parieur a misé sur un des paris gagnants. Son solde augmentera par rapport à sa mise ( $gain = cote \times mise$ ) quand le match est terminé par l'administrateur

### 8.2.5. Les composants et leur déploiement

Les 2 sites, 2 serveurs api sont déployés en ligne sur Heroku.

Les 2 base de données sont aussi déployés en ligne

L'application mobile est générée sous forme de fichier « .apk ».

L'application Windows est déployée en fichier compressé .zip contenant l'installation de l'application.

## 9. Tests du système logiciel

Les API web services sont d'abord testées sous « Postman » avant d'être testées en situation réelle.

Les applications doivent être testées ensemble lors du test final.

Les tests sont faits par les développeurs lors des développements.

Les tests seront faits sur les versions en ligne des applications avant la livraison.

## 10. Conclusion générale

En conclusion, le projet développé nous a permis de créer une application de pari en ligne sur plusieurs plateformes. Du fait que de nos jours tout est connecté, il est impératif que l'utilisateur puisse utiliser l'application de pari en ligne sur son smartphone ou ordinateur ou site web.

Les fonctionnalités du projet peuvent être étendues avec de nouveaux services du fait de son architecture multi-tiers et l'ajout d'une application IOS peut être envisagé ou l'ajout d'autres sports.

Durant le projet, nous avons eu de la difficulté à déployer la base de données Oracle en ligne du fait que la plupart des serveurs en ligne sont payants et que Oracle Free Cloud Tier est difficile à intégrer avec Grails.

## 11. Références et Bibliographie

### Le fonctionnement du pari

- **Wikipédia** *Pari sportif* [en ligne] Disponible sur [https://fr.wikipedia.org/wiki/Pari\\_sportif](https://fr.wikipedia.org/wiki/Pari_sportif) (consulté le 11-08-2021)
- **Quentin Toulemonde**, *Les paris sportifs en ligne : comprendre, jouer, gagner* : Comment faire un pari sportif
- **Guide du pari** *Quel est le meilleur site de paris sportifs ?* [En ligne] Disponible sur <https://www.guidedupari.com/meilleur-site-de-pari.html> (consulté le 11-08-2021).

### Autres ressources

- **Heroku Dev Center** *Deploying Grails App on Heroku* [En ligne] Disponible sur <https://blog.heroku.com/grails> (consulté le 10-05-2021)
- **MongoDB** *Mongo DB Atlas* [en ligne] Disponible sur <https://cloud.mongodb.com/> (consulté le 15-05-2021)
- **Heroku Dev Center** *Deploying Node.js App on Heroku* [En ligne] Disponible sur <https://devcenter.heroku.com/articles/deploying-nodejs> (consulté le 03-08-2021)

- **Modelio** *Support Modelio des diagrammes* [en ligne] Disponible sur <https://www.modeliosoft.com/fr/ressources/exemples-de-diagrammes.html> (consulté le 01-07-2021)
- **QR Code** *Code QR : qu'est-ce que c'est ?* [en ligne] Disponible sur <https://www.futura-sciences.com/tech/definitions/informatique-code-qr-11969/> (consulté le 01-07-2021)
- **Postman** *Débuter avec Postman* [en ligne] Disponible sur <https://practicalprogramming.fr/postman> (consulté le 02-07-2021)