|  |
| --- |
| * Toky Ny Aina RAFIDIMANANA * Miora Gabrielle RANOMENJANAHARY * Heggy Randy LOYENS |

|  |
| --- |
|  |
| ZE-BET |
| Plateforme de paris en ligne |

Table des matières

[1. Introduction générale 2](#_Toc79487652)

[2. Présentation du projet 3](#_Toc79487653)

[3. Etat de l’art sur le sujet traité 4](#_Toc79487654)

[3.1 Critères de comparaison 4](#_Toc79487655)

[3.2 Etude de chaque solution au vu des critères 4](#_Toc79487656)

[3.3 Tableau comparatif des solutions au vu des critères 4](#_Toc79487657)

[4. Etude de l’existant et solution envisagée 5](#_Toc79487658)

[4.1 Etude de l’existant 5](#_Toc79487659)

[4.1.1 Description externe du système logiciel existant (vision utilisateur) 5](#_Toc79487660)

[4.2 Critique de l’existant 5](#_Toc79487661)

[4.3 Solutions envisagée 5](#_Toc79487662)

[4.4 Objectifs principaux et livrables 5](#_Toc79487663)

[5. Démarche projet 6](#_Toc79487664)

[5.1 Principes de la démarche projet 6](#_Toc79487665)

[5.1.1 Activités d’ingénierie logicielle 6](#_Toc79487666)

[5.1.2 Méthode de gestion de projet utilisée 6](#_Toc79487667)

[5.1.3 Rôles et responsabilités 7](#_Toc79487668)

[5.1.5 Gestion de la configuration 7](#_Toc79487669)

[5.2 Contraintes et risques sur le projet 8](#_Toc79487670)

[5.3 Démarche projet mise en œuvre 8](#_Toc79487671)

[5.4 Planification 9](#_Toc79487672)

[5.5 Budget du projet 9](#_Toc79487673)

[6. Exigences réalisées dans le projet (vision externe/utilisateur) 9](#_Toc79487674)

[6.1 Exigences fonctionnelles – Cas d’utilisation 9](#_Toc79487675)

[6.1.1 Cas d’utilisation du site de pari en ligne et de l’application mobile 9](#_Toc79487676)

[6.1.2 Cas d’utilisation pour l’application de la banque 10](#_Toc79487677)

[6.1.3 Cas d’utilisation pour le back office du site de pari 11](#_Toc79487678)

[6.2 Exigences non fonctionnelles transverses 12](#_Toc79487679)

[6.2.1 Utilisabilité 12](#_Toc79487680)

[6.2.2 Performances 13](#_Toc79487681)

[6.2.3 Sécurité : 13](#_Toc79487682)

[6.3 Interfaces détaillées 13](#_Toc79487683)

[6.3.1 IHM 13](#_Toc79487684)

[6.3.2 Interfaces avec d’autres systèmes 27](#_Toc79487685)

[7. Architectures système 27](#_Toc79487686)

[8. Conception du système logiciel réalisée dans le projet (vision interne/développeur) 28](#_Toc79487687)

[8.1 Plateforme technique 28](#_Toc79487688)

[8.2 Conception du logiciel développé 29](#_Toc79487689)

[8.2.1 Conception du code source 29](#_Toc79487690)

[8.2.2 Le code source – vue statique 29](#_Toc79487691)

[8.2.3 Modélisation de données 32](#_Toc79487692)

[8.2.4 Réalisation des cas d’utilisation 33](#_Toc79487693)

[8.2.5 Les composants et leur déploiement 34](#_Toc79487694)

[9. Tests du système logiciel 34](#_Toc79487695)

[10. Conclusion générale 35](#_Toc79487696)

[11. Références et Bibliographie 35](#_Toc79487697)

[12. Annexes 35](#_Toc79487698)

# Listes des figures :

[Figure 1:Architecture de la plateforme de pari en ligne 10](#_Toc79832236)

[Figure 2:Diagramme de séquence correspondant au client web et mobile 14](#_Toc79832237)

[Figure 3:Diagramme de séquence correspondant à l’application de la banque 14](#_Toc79832238)

[Figure 4:Diagramme de séquence correspondant au back office du site de pari 15](#_Toc79832239)

[Figure 5:Ecran de login de l’application 16](#_Toc79832240)

[Figure 6:Login par QR Code 17](#_Toc79832241)

[Figure 7:Module mapping des cash point 17](#_Toc79832242)

[Figure 8:Page d’accueil de l’application avec la liste des paris disponibles 19](#_Toc79832243)

[Figure 9:Panier sur un pari sportif et liste des paris terminé et paris en cours parier par l’utilisateur. 19](#_Toc79832244)

[Figure 10:Historiques pour chaque pari effectué par l’utilisateur le loader qui met en attente l’utilisateur pour charger les vues. 20](#_Toc79832245)

[Figure 11:Page de Login 20](#_Toc79832246)

[Figure 12:Page d’accueil 21](#_Toc79832247)

[Figure 13:Page de transaction 21](#_Toc79832248)

[Figure 14:Ecran de login pour le client 22](#_Toc79832249)

[Figure 15:Ecran d’accueil 22](#_Toc79832250)

[Figure 16:Ecran détails d’un pari 23](#_Toc79832251)

[Figure 17:Ecran pour les informations du client 23](#_Toc79832252)

[Figure 18:Ecran (2) pour les informations du client 24](#_Toc79832253)

[Figure 19:Ecran d’inscription 24](#_Toc79832254)

[Figure 20:Ecran pour type de pari Tennis 25](#_Toc79832255)

[Figure 21:Ecran d’achat de jeton 25](#_Toc79832256)

[Figure 22:Authentification des membres 26](#_Toc79832257)

[Figure 23:Création pari 26](#_Toc79832258)

[Figure 24:Liste des paris 27](#_Toc79832259)

[Figure 25:Modification d’un pari 27](#_Toc79832260)

[Figure 26:Insertion résultat en fonction d’une catégorie 28](#_Toc79832261)

[Figure 27:Architectures système de la plateforme de pari en ligne 28](#_Toc79832262)

[Figure 28:Architectures système de la plateforme du système bancaire 29](#_Toc79832263)

[Figure 29:Architecture matérielle pour le système de pari en ligne 29](#_Toc79832264)

[Figure 30:Architecture matérielle pour la banque 30](#_Toc79832265)

[Figure 31:Packaging de l’api Banking et l’api du site 30](#_Toc79832266)

[Figure 32:Packaging du projet mobile sur Android studio 30](#_Toc79832267)

[Figure 33:Packaging du projet application lourde sur Visual Studio 31](#_Toc79832268)

[Figure 34:MCD de la base de données Oracle pour la banque 31](#_Toc79832269)

[Figure 35:Schémas de la base de données NoSQL pour la gestion des paris 32](#_Toc79832270)

[Figure 36:Diagramme de séquence correspondant au site de pari en ligne 33](#_Toc79832271)

[Figure 37:Diagramme de séquence pour l’application mobile 34](#_Toc79832272)

# Listes des tableaux :

[Tableau 1:Tableau comparatif des solutions au vu des critères 6](#_Toc79832964)

[Tableau 2:Contraintes et risques sur le projet 9](#_Toc79832965)

[Tableau 3:Budget du projet 11](#_Toc79832966)

# 1. Introduction générale

De nos jours, l’internet est devenu une banalité dans nos vies quotidiennes et de plus en plus de jeune s’intéresse davantage aux jeux en lignes. Le pari sport fait partie de ces attractions qui leur permet de jouer tout en restant chez eux depuis leurs smartphones ou leurs ordinateurs.

Le pari sport est un moyen de distraction car il s’agit d’un jeu, mais aussi, il permet à autrui de gagner un peu ou beaucoup d’argent en faisant des pronostiques gagnantes.

Cependant, ils existent de nombreux sites de pari sportif que nous pouvons trouver en ligne mais peu d’entre eux sont attractifs à cause des interfaces “vieillot” et ils ne proposent pas souvent d’intégrer de nouvelles fonctionnalités en rapport avec les nouvelles technologies.

C’est pour cela que nous avons lancé un défi et introduit le concept de “Ze Bet” une plateforme de pari en ligne. Elle propose une interface utilisateur moderne et qui intègre des fonctionnalités de mapping, la facilitation de connexion à partir des QR Codes et surtout l’intégration d’une banque propre à la plateforme.

# 2. Présentation du projet

Le projet consiste à créer une plateforme de paris en ligne. La plateforme manipulera les monnaies des clients dans une banque propre à l’application. D’une part, les clients pourront stocker de l’argent au sein de la banque, et d’autre part, ils pourront utiliser cet argent pour échanger avec des jetons de pari sur le site.

La plateforme sera composée d’un site web de pari sportif, une application bancaire pour placer l’argent des clients, une application mobile du site et un site de gestion des contenus du site web. Les monnaies d’échanges seront des jetons que l’on achètera à travers l’application bancaire.

L’application bancaire sera une application lourde pour la gestion des comptes bancaires des clients.

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

## 3.1 Critères de comparaison

On est parti d’une plateforme en ligne déjà existante « Bet 261 ». Cette plateforme propose plusieurs disciplines.  Il réunit à la fois le jeu du hasard et les paris en lignes (tout catégorie confondu). C’est cette diversité qui fait tout son potentiel. Le joueur n’a plus besoin de passé entre différentes plateformes pour faire des paris en lignes, ou s’engager dans un jeu de hasard. Aux vues de ses nombreuses qualités nous avons décidé d’établir deux critères de comparaison par rapport à notre site. Nous allons comparer en premier **nos fonctionnalités** par rapport à ceux de « Bet261 », et le en second la **compatibilité** sur différentes plateformes.

## 3.2 Etude de chaque solution au vu des critères

Suivant la tendance actuelle, plus de personnes sont sur mobile que sur ordinateur. Il serait donc plus judicieux de fournir un apk pour pouvoir cibler plus de public. Côté pratique il serait plus facile pour le client de juste scanner son Qr Code au lieu de se remettre à saisir sont mot de passe et son identifiant chaque fois. Fournir un point de localisation de zone de retraite aussi sera plus bénéfique pour le client.

## 3.3 Tableau comparatif des solutions au vu des critères

|  |  |  |
| --- | --- | --- |
| **CRITERES** | **Bet261** | **ZeBet** |
| Fonctionnalités :   * L’application mobile sera munie d’un système de géolocalisation. Ainsi, Il est possible de repérer un cash point aux alentours de votre position pour récupérer l’argent gagné. * Authentification via QR Code * Différents types de panier (panier simple, panier multiple) | * Non                      * Non      * Oui | * Oui                      * Oui      * Panier simple seulement |

|  |  |  |
| --- | --- | --- |
| Compatibilité :   * Existence d’une version mobile pour la plateforme | * Non | * Oui |

Tableau 1:Tableau comparatif des solutions au vu des critères

# 4. Etude de l’existant et solution envisagée

## 4.1 Etude de l’existant

### 4.1.1 Description externe du système logiciel existant (vision utilisateur)

Le site web bet261 est une application de pari sportif en ligne.

Le client peut créer un compte sur l’application à travers une inscription, ensuite la validation du compte se fait par numéros mobile du client. Le site web n’accepte que les numéros Telma.

Le client peut charger ou faire un retrait de son solde de pari (monnaies virtuels utilisée pour les paris) à travers des comptes mobiles Banking comme MVola ou Orange Money, ou bien à partir des points de vente de bet261.

A la page d’accueil, le site propose plusieurs types de paris sportifs tels que le foot, le basket, le tennis, les courses de chevaux, rugby, etc.    
Chaque type de sport possède diverses catégories de pari à savoir ; Pour le foot : On peut parier sur le résultat du match ou les nombres de buts marqués.

Et sur chaque pari, le client peut attribuer une mise selon son choix. On utilise les soldes de pari comme monnaie de mise.

Les paris du client sont enregistrés dans l’application sous forme de panier.

Pour valider la commande le client doit avoir la somme requise dans son solde pari.

Le client peut lister les historiques de ses paris en cours ou auparavant. Il peut aussi regarder les transactions effectuer des mouvements (recharges ou retrait) de son solde pari.

## 4.2 Critique de l’existant

Le site « Bet 261 » propose plusieurs disciplines en partant du pari sportif et des jeux dans les casinos.

Le parie en **direct** est aussi l’une des spécificités de ce site. C’est-à-dire que vous pouvez encore parier durant le match.

Ce site est aussi dédié qu’au fin connaisseur. Dans la catégorie foot par exemple il propose des choses dont je suis sûre que certaines personnes ne connaissent pas le sens le champ « double champ par exemple »

## 4.3 Solutions envisagée

Les applications de pari en ligne sont assez complètes mais n’intègrent pas des fonctionnalités et des technologies qui sont indispensables en ce moment. 

Par exemple :

* Ils devraient intégrer un module de mapping avec géolocalisation pour les cash points.
* Intégrer l’utilisation des authentifications par QR Code pour les applications mobiles.

## 4.4 Objectifs principaux et livrables

Notre objectif sur ce projet est de mettre en place une plateforme de pari en ligne « Ze Bet » avec la gestion des monnaies des clients qui est propre à la plateforme.   
La plateforme se divise en quatre parties :

* La partie site web de pari en ligne.
* La partie back office pour la gestion des contenues du site
* Une application mobile pour le site.
* Une application de banque pour la gestion des monnaies des clients.

# 5. Démarche projet

## 5.1 Principes de la démarche projet

### 5.1.1 Activités d’ingénierie logicielle

Pour rendre le projet maintenable, robuste et performant, nous avons :

* Travaillé sur une base NoSQL pour le site de pari afin d’accroitre les performances. Ainsi, nous avons optimisé les temps de réponses pour chaque requête et permis au système de gérer plusieurs utilisateurs en même temps.
* Respecté les exigences du client tout en maintenant le système évolutif avec le respect des règles de conception en ingénierie logicielle :
* Conception de l’architecture global
* Mise en place des cas d’utilisations à travers des diagrammes UML
* Mise en place des diagrammes de séquences
* Mise en place des diagrammes de classes avec les packages
* Conception des schémas de base de données (MCD et MPD)
* Optimisation des requêtes SQL
* Choisi les langages de programmation les plus performants de nos jours : Java, JS, C#
* Effectué des séries de Test pour chaque étape de l’application et après son déploiement (test en développement et test en production).

Une image contenant texte, parking

Description générée automatiquement

Figure 1:Architecture de la plateforme de pari en ligne

### 5.1.2 Méthode de gestion de projet utilisée

On a utilisé la méthode agile pour gérer le projet. En partitionnant le projet, nous pouvons ainsi nous concentrer sur une partie à la fois.

A chaque parti entamé du projet, nous nous sommes assurés de pouvoir présenter une partie fonctionnelle.

Les démarches de communications établies sont des réunions et la présentation des résultats à chaque fin de semaine.

### 5.1.3 Rôles et responsabilités

Nous étions une équipe de trois pour la réalisation de ce projet :

**Toky :**

Création de l’application lourde en Winform

Création de l’application mobile en JAVA

Développement de certaines parties dans l’api NodeJS

Conception logiciel

**Miora :**

Création de l’api banking.

Intégration du back office.

Intégration du panier et des détails de chaque pari dans le front-office.

Développement d'une partie de l’api NodeJS.

Conception logiciel

**Heggy :**

Intégration jwt token dans certains partis de l’api.

Développement d’une partie de l’api NodeJs

Création de l’interface statique back-office et front- office.

Intégration partie front office.

Conception logiciel

5.1.4 Outils

Pour chaque démarche du projet on a utilisé des outils et logiciels qui sont surtout faciles à utiliser et efficaces

* Entreprise Architect : modélisation des diagrammes de séquences, cas d’utilisation, diagramme de packages et de classes
* Power AMC : pour la création des modèles conceptuels de données et des modèles physiques de données des bases de données
* Intellij IDE : est un environnement de développement intégré destiné au développement de logiciels informatiques reposant sur la technologie Java.
* Notepad : éditeur de texte libre.
* Visual Studio : éditeur de code développé par Microsoft.
* Excel : un logiciel de la suite bureautique Office de Microsoft et permet la création de tableaux, de calculs automatisés, de plannings, de graphiques et de bases de données
* Gliffy: est un logiciel de création de diagrammes via une application basée sur le cloud HTML5

### 5.1.5 Gestion de la configuration

Les livrables du projet sont mise à disposition dans un dépôt en ligne git, ce dépôt comprend :

* Code source du site front-office, du site back-office, de l’application mobile du site de pari et de l’application lourd pour la banque.
* Code source de l’api de la banque.
* Code source de l’api pour la gestion du site.

Pour les règles de nommage des fichiers, on a nommé nos fichiers avec les noms de package en préfixe des noms de fichiers. Dans les packages modèles, les fichiers sont nommés selon le nom des tables, des vues ou des objets.

## 5.2 Contraintes et risques sur le projet

|  |  |  |  |
| --- | --- | --- | --- |
| N° | Les risques et contraintes | Facteurs Contribuants | Solutions proposées |
| 1 | Coût d’hébergement d’une base de données Oracle | Une base de données Oracle est très couteuse | Migration vers une base de données PostgreSQL ou MariaDB qui sont beaucoup moins chères mais aussi performants |
| 2 | Couverture de l’application mobile | Application mobile a été développé en JAVA sur Android donc incompatible avec IOS | Utiliser un framework (Ionic, flutter) cross-platform afin de rendre compatible l’application à la fois IOS et Android |

Tableau 2:Contraintes et risques sur le projet5.3 Démarche projet mise en œuvre

* En tant qu’utilisateur il doit pouvoir acheter des jetons pour parier.

BackLog :

Création api pour convertir de l’argent en jetons (api banking)

Affichage Achat de jetons avec numéro de compte.

* L’utilisateur pourra s’authentifier via un login et un mot de passe ou par QrCode

BackLog :

Création api pour vérifier utilisateur.

Affichage page d’authentification

* Deux choix de type de pari sont offerts aux utilisateurs.

BackLog :

Création api pour récupérer la liste des types de paris

Afficher liste des choix de types de pari.

* Des champs à parier seront proposés aux utilisateurs, après avoir choisir entre le type de pari.

BackLog :

Création api pour récupérer la liste des champs à parier.

Création api pour pouvoir récupérer cote de chaque champ en fonction d’une catégorie.

Afficher champ avec cote.

* L’authentification via un compte sera obligatoire si l’utilisateur veut miser dans l’un des champs.

BackLog :

Api authentification d’utilisateur.

Affichage authentification avec contrôle de valeur.

* L’utilisateur aura ensuite accès à son panier dans lequel il pourra insérer différents paris mais aussi supprimer celui-ci.

BackLog :

Création panier

Suppression élément dans panier

Insertion mise dans panier.

Totaliser les mises et les gains potentiels.

Validation panier.

Création api pour insérer les résultats prédits par l’utilisateur.

Affichage liste éléments des paniers.

* L’utilisateur pourra depuis son mobile savoir les cash point près de sa position actuelle.

BackLog :

Création api pour récupérer point de vente en fonction de la position actuel.

Affichage du périmètre correspondant.

## 5.4 Planification

**Diagramme de Gantt**

## 5.5 Budget du projet

Le coup du projet est évalué à 17 000 000 Ariary selon le devis suivant :

|  |  |
| --- | --- |
| Salaire des développeurs | 1 500 000 Ariary |
| Local + Internet | 4 000 000 Ariary |
| Electricité | 200 000 Ariary |
| Machines | 12 000 000 Ariary |
| **Total** | 17 000 000 Ariary |

Tableau 3:Budget du projet

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

## 6.1 Exigences fonctionnelles – Cas d’utilisation

### 6.1.1 Cas d’utilisation du site de pari en ligne et de l’application mobile

Le client web peut :

* S’authentifier et se connecter sur le site web
* Créer un compte sur le site
* Avoir ses informations personnelles sur le site
* Avoir la liste des paris disponibles
* Effectuer des échanges de jetons (échange de jeton en argent et vice versa)
* Parier sur des matchs
* Manipuler son panier de pari
* Lister les historiques des échanges de jetons
* Générer un QR Code pour pouvoir se connecter sur une application mobile
* Voir les résultats des paris

Le client mobile peut :

* S’authentifier et se connecter via un formulaire sur l’application
* S’authentifier et se connecter rapidement via le scan d’un QR Code générer par le site
* Activer sa position et voir la liste des cash point aux alentours et demander un itinéraire vers ce cash point via google map
* Parier sur des matchs
* Avoir ses informations personnelles
* Avoir la liste des paris disponibles
* Lister les historiques des paris terminés et des paris en cours
* Manipuler son panier de pari
* Voir les résultats des paris

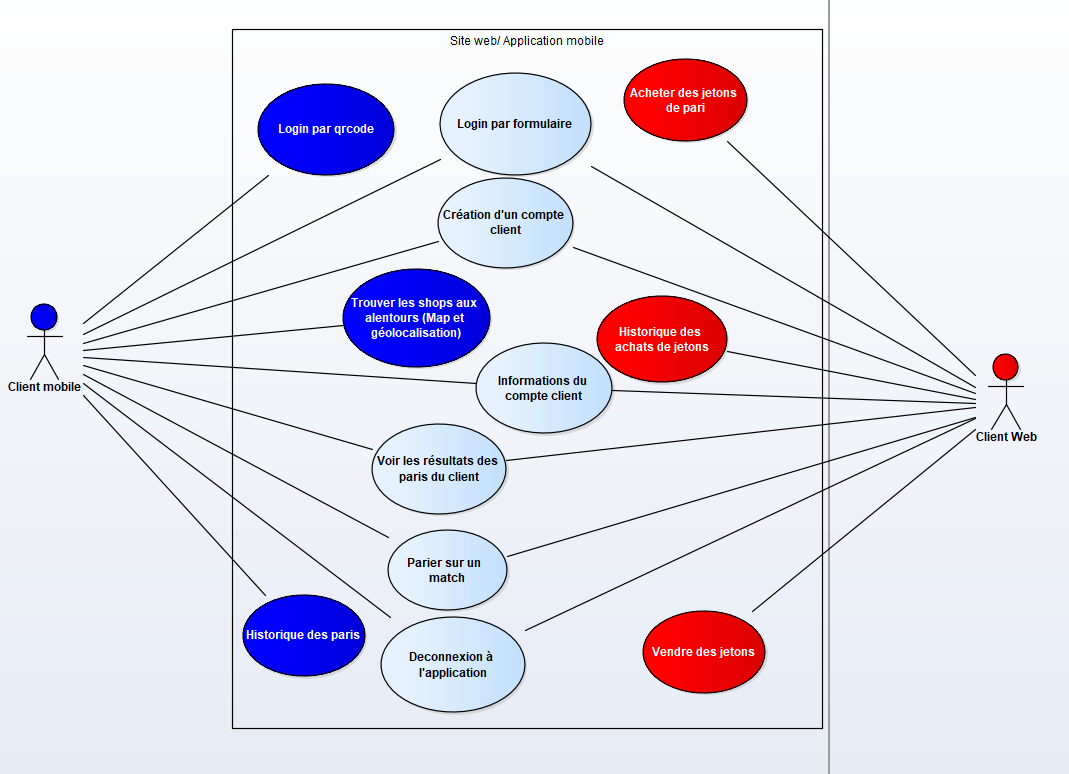


Figure 2:Diagramme de séquence correspondant au client web et mobile

### 6.1.2 Cas d’utilisation pour l’application de la banque

L’employé de la banque peut :

* S’authentifier et se connecter à l’application de la banque
* Vérifier l’existence ou la validité du compte bancaire du client
* Lister les historiques des comptes bancaires du client
* Effectuer les mouvements bancaires (débit et crédit) sur un compte bancaire

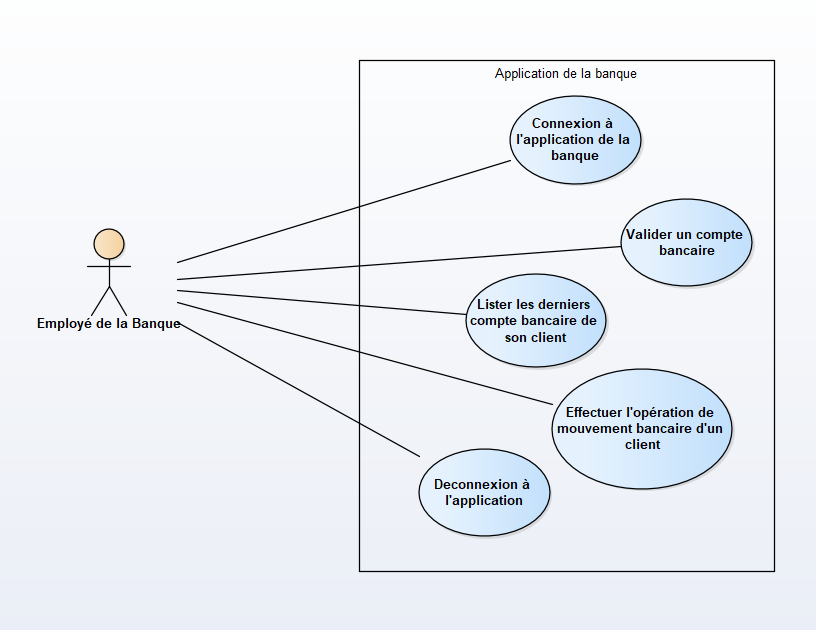


Figure 3:Diagramme de séquence correspondant à l’application de la banque

### 6.1.3 Cas d’utilisation pour le back office du site de pari

L’administrateur peut :

* S’identifier et s’authentifier à l’application.
* Gérer les paris disponibles.
* Gérer les informations sur les paris c’est à dire les catégories de paris (nombre de but marqué, équipe gagnante), les champs à pariés ainsi que les côtes.
* Gérer le système de géolocalisation des points de ventes de “Ze Bet”.
* Gérer l’insertion des résultats des paris.
* Gérer l’espace client.

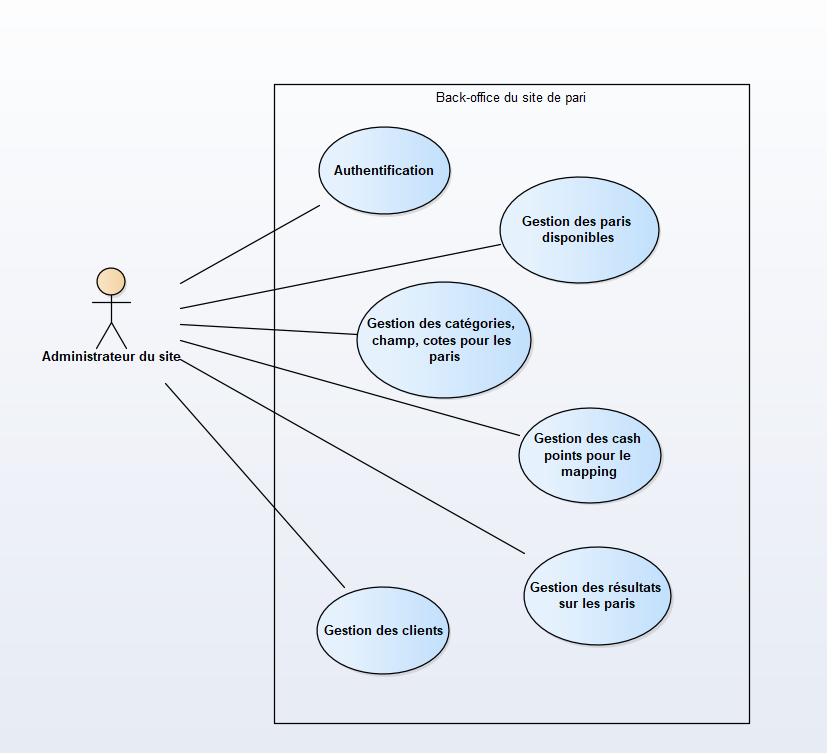


Figure 4:Diagramme de séquence correspondant au back office du site de pari

## 6.2 Exigences non fonctionnelles transverses

### 6.2.1 Utilisabilité

Charte graphique :  rouge, noir, orange, verte.

Logo :

Une image contenant texte, assiette, vaisselle, arts de la table

Description générée automatiquement

La typographie ou police d’écriture : « Open sans » Sans serif

Ergonomie : Site très ergonomique car les menus sont accessibles facilement, les polices sont bien lisibles.

### 6.2.2 Performances

Le site est plutôt réactif mais la connexion à la plateforme de Web Hosting Heroku est lente à cause du forfait gratuit.

### 6.2.3 Sécurité :

Sécurisation de l’api par l’utilisation de la librairie JWT Token

Utilisation des hashs SHA-2 pour l’hashage des mots de passe

## 6.3 Interfaces détaillées

### 6.3.1 IHM

#### 6.3.1.1 Client mobile

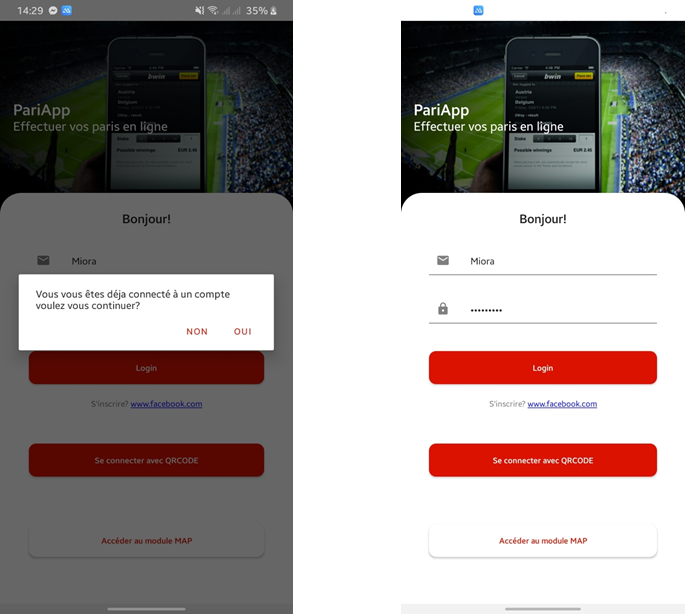


Figure 5:Ecran de login de l’application

Si l’utilisateur s’est déjà connecté sur l’application alors l’application le reconnaitra automatiquement et propose de se connecter sans saisir le login.

**Login par QR Code**



Figure :Login par QR Code

Une image contenant carte

Description générée automatiquement

Figure 7:Module mapping des cash point

Une image contenant carte

Description générée automatiquement**L’application « Ze bet » ouvre automatiquement google Map pour l’itinéraire vers un cash point en fonction de votre position**



Figure 8:Page d’accueil de l’application avec la liste des paris disponibles

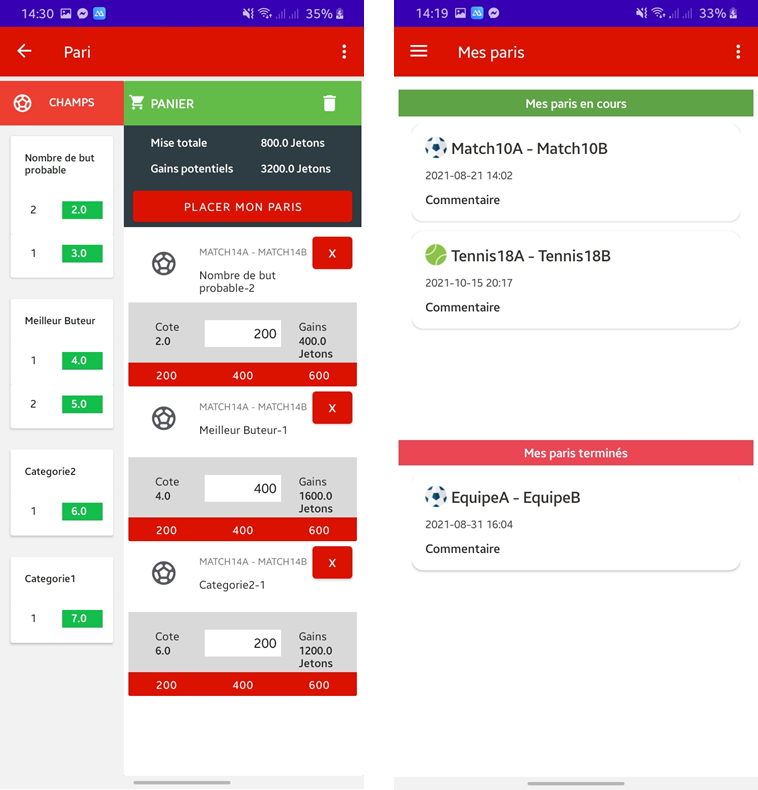


Figure 9:Panier sur un pari sportif et liste des paris terminé et paris en cours parier par l’utilisateur.

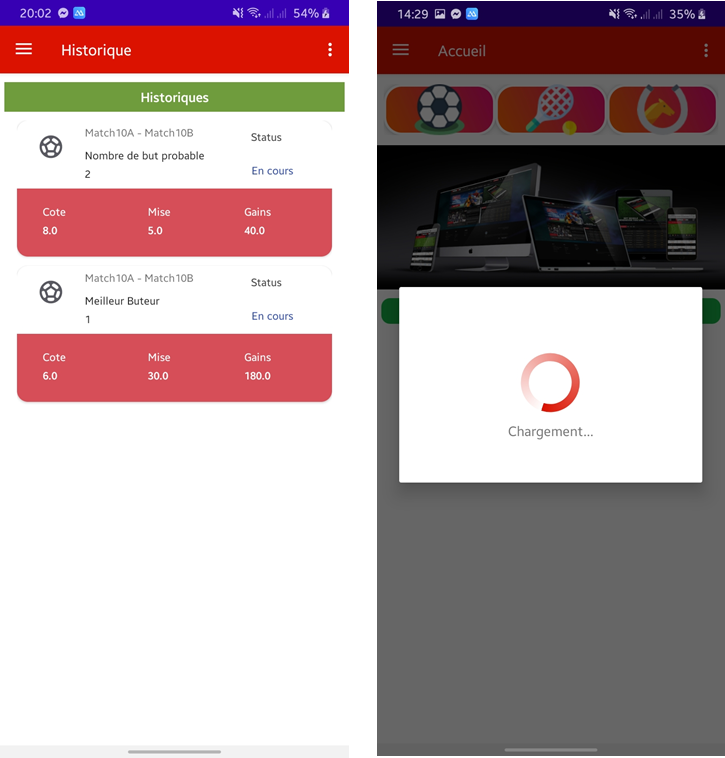
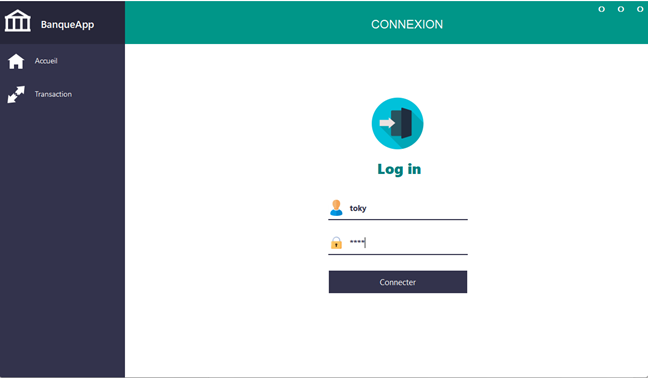


Figure 10:Historiques pour chaque pari effectué par l’utilisateur le loader qui met en attente l’utilisateur pour charger les vues.

#### 6.3.1.2 Client lourd

   
 Figure 11:Page de Login

Une image contenant texte

Description générée automatiquement

Figure 12:Page d’accueil

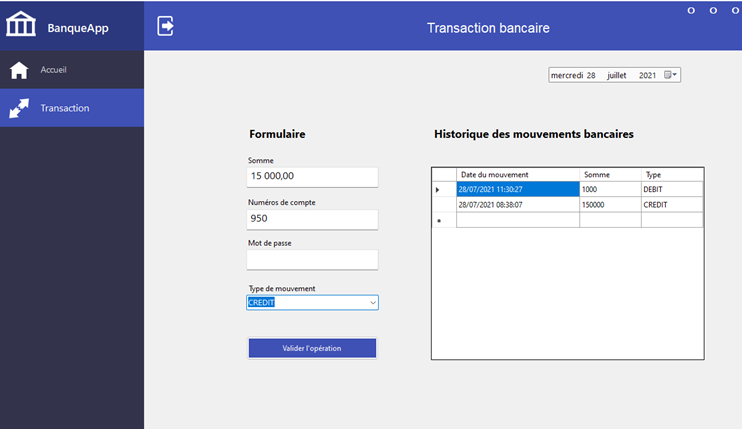


Figure 13:Page de transaction

#### 6.3.1.3 Front office

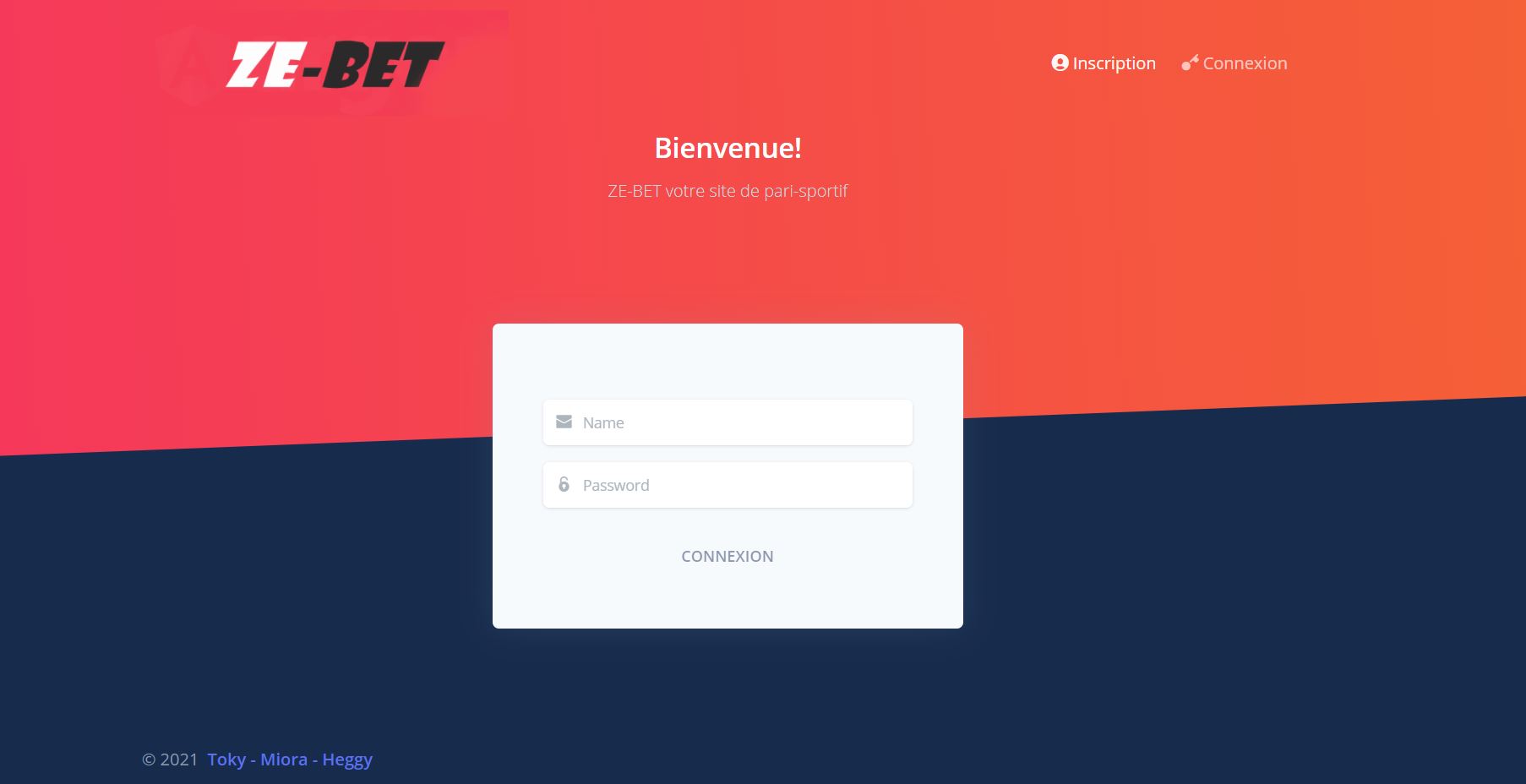


Figure 14:Ecran de login pour le client

Le client peut se connecter à l’application ou s’inscrire

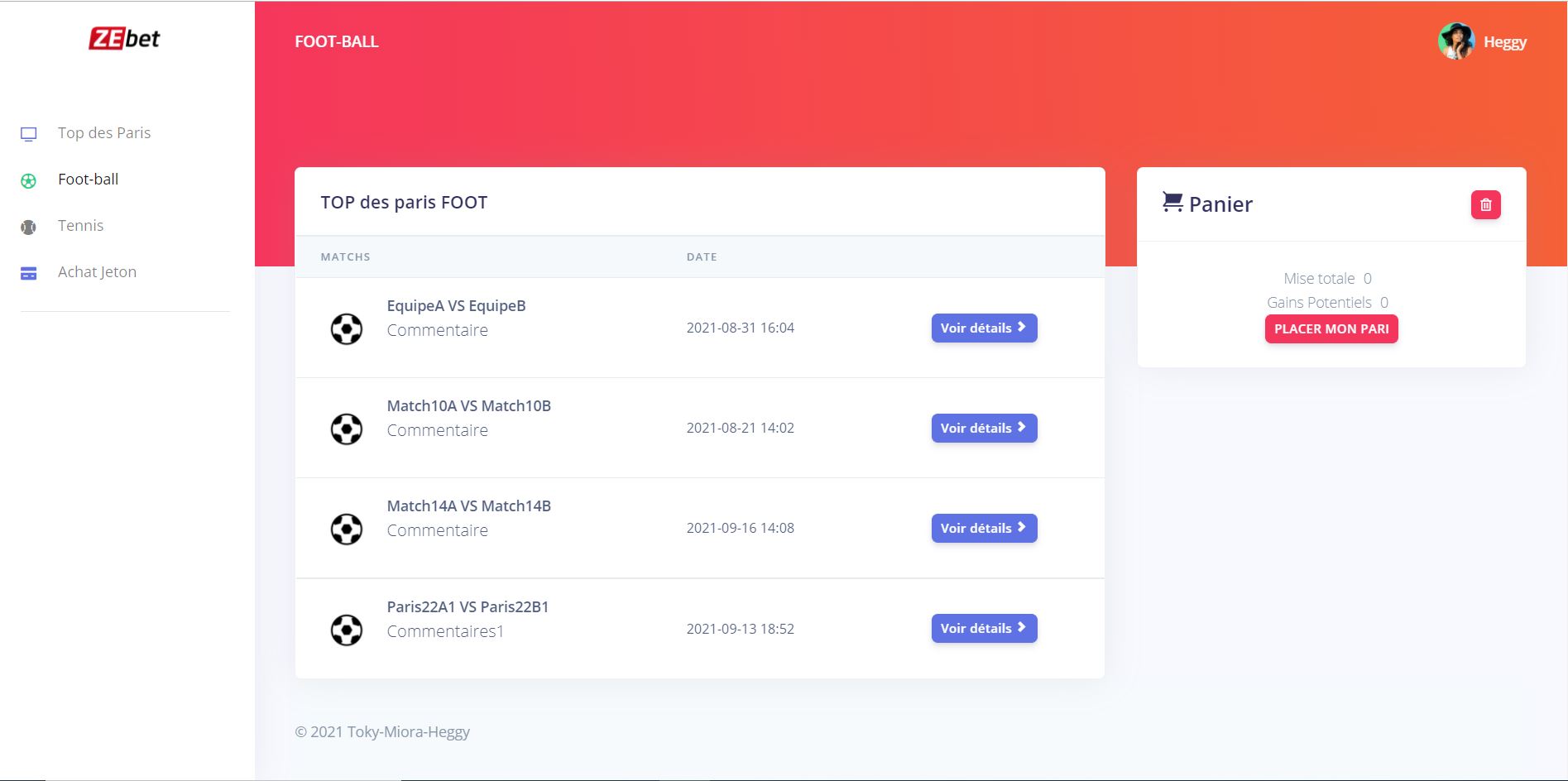


Figure 15:Ecran d’accueil

Cet écran contient la liste des paris disponibles

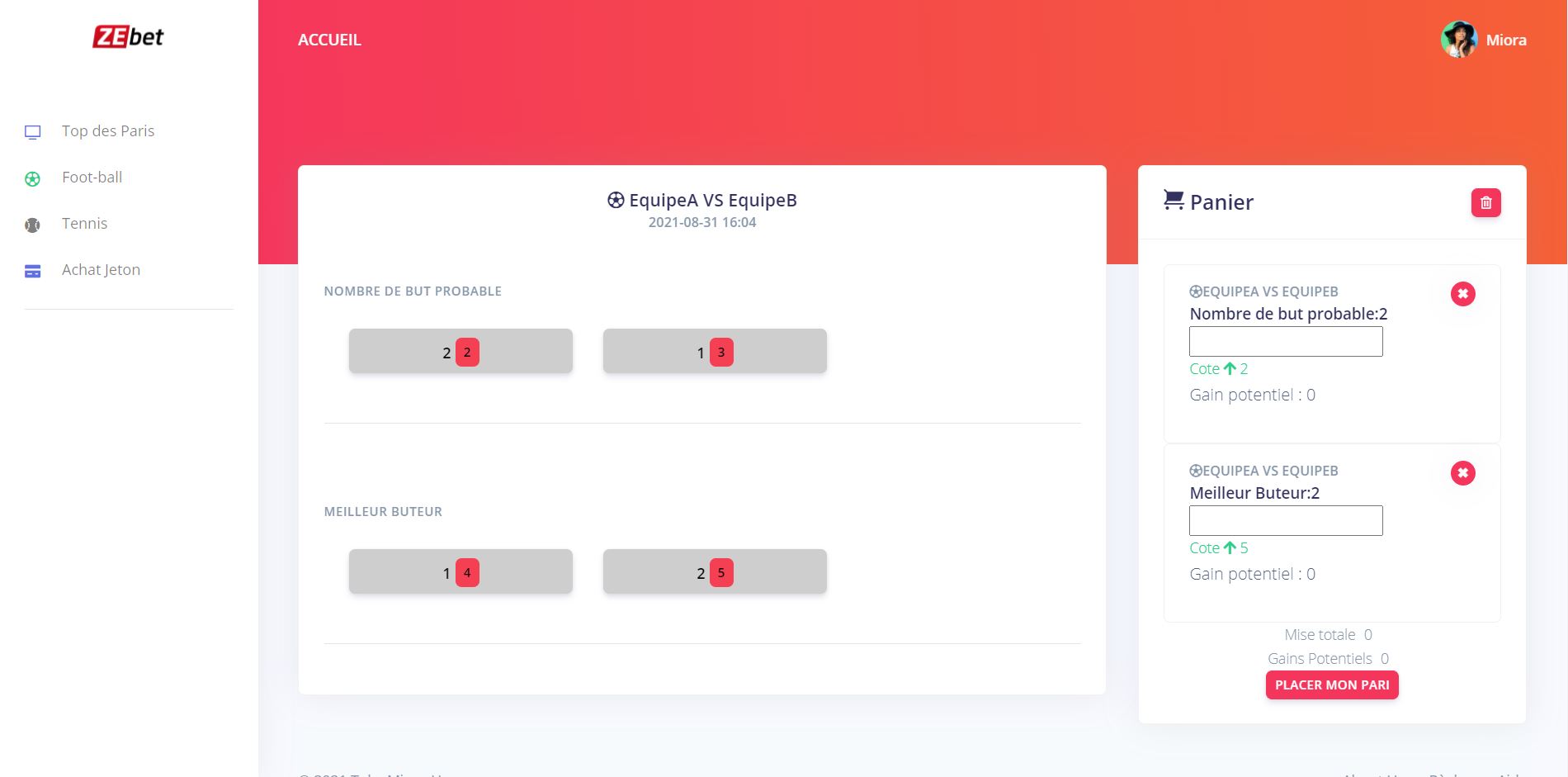


Figure 16:Ecran détails d’un pari

Le client peut voir le côté de chaque champ en fonction du catégorie et match.

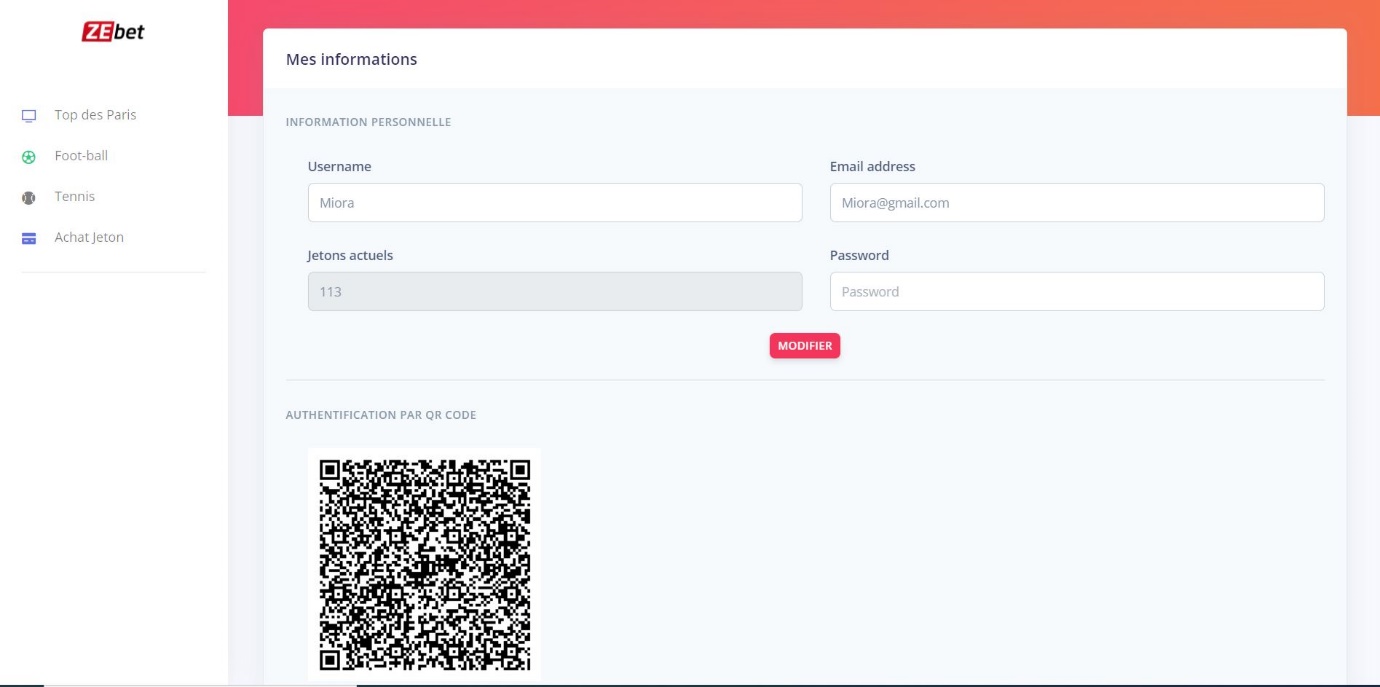


Figure 17:Ecran pour les informations du client

Ecran contenant les informations personnelles du client connecté (nombres de jetons, mail..) avec une possibilité de les modifié.

CodeQR contenant les informations du client.

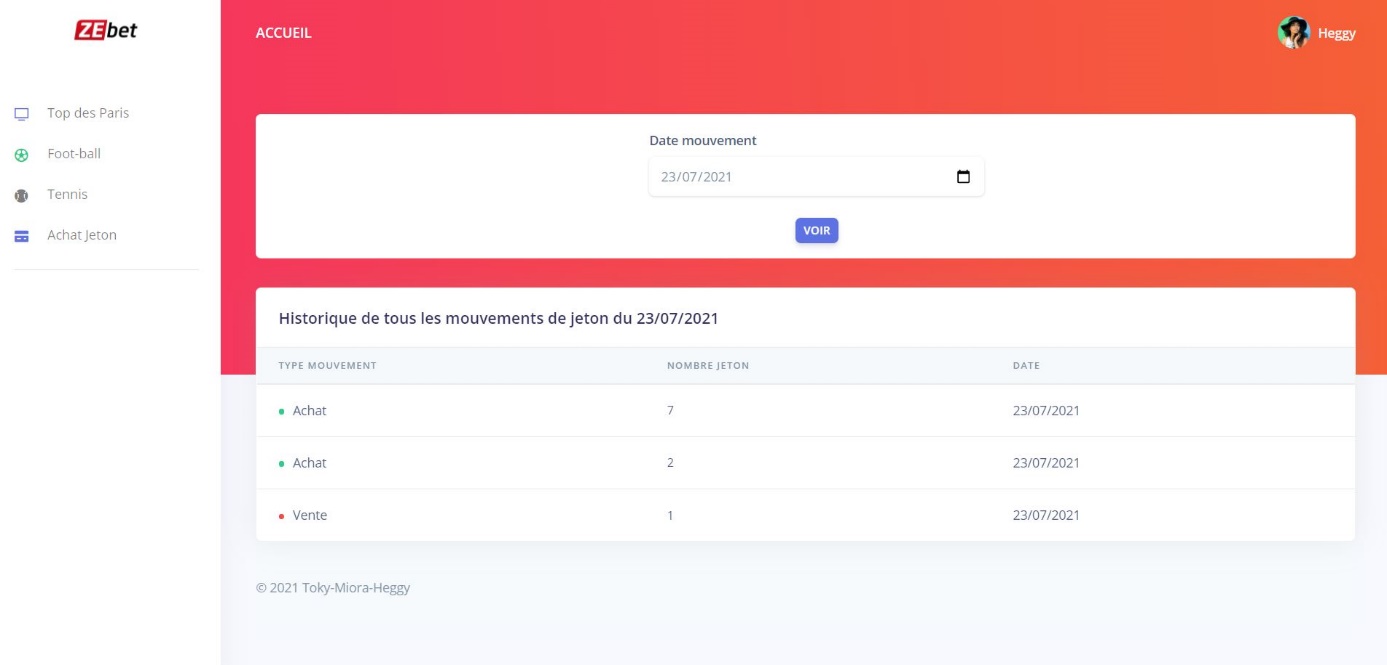


Figure 18:Ecran (2) pour les informations du client

Ecran contenant la liste des mouvements des jetons

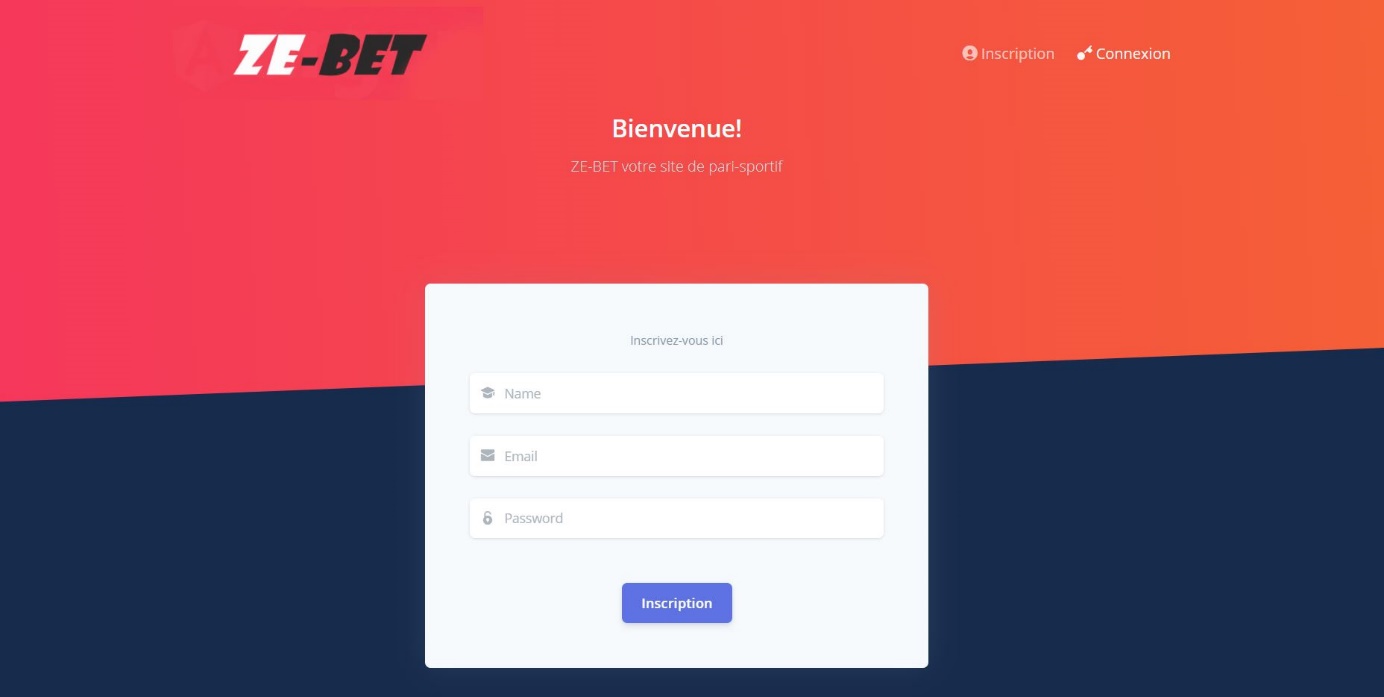


Figure 19:Ecran d’inscription

Ecrans contenant un formulaire d’inscription pour les nouveaux clients.



Figure 20:Ecran pour type de pari Tennis

Ecran contenant uniquement la liste des paris de type Tennis.

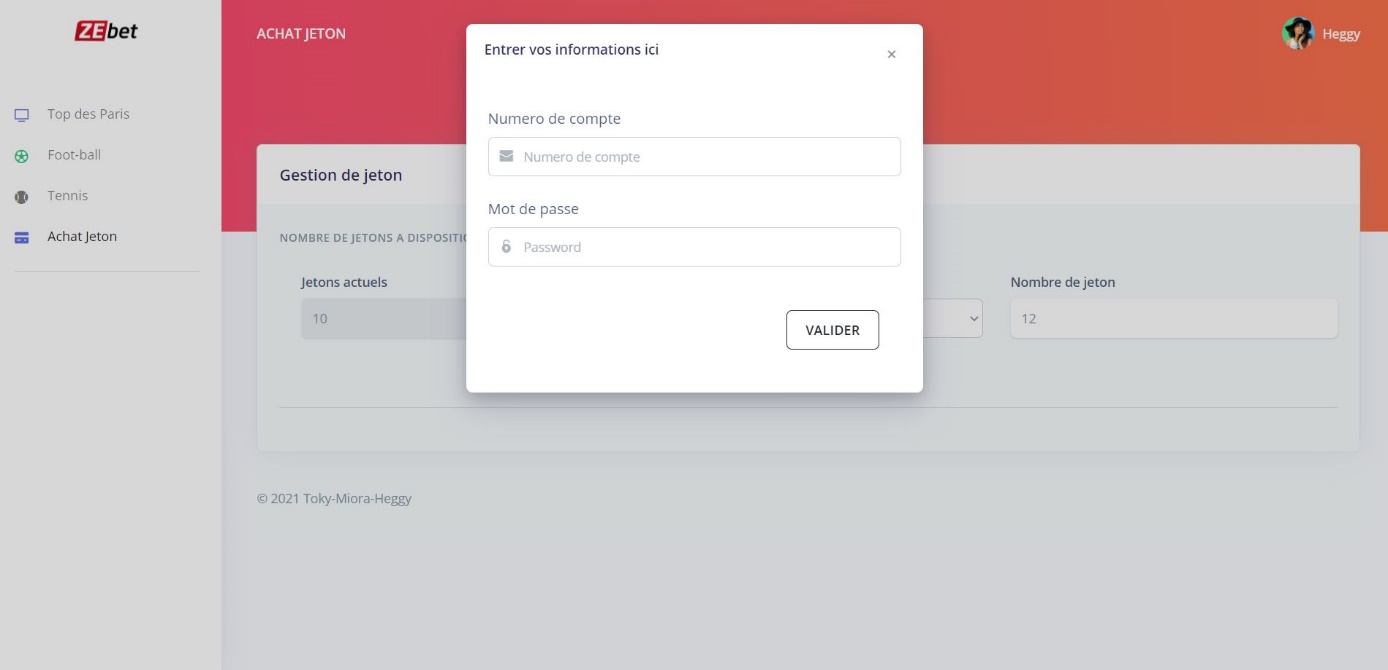


Figure 21:Ecran d’achat de jeton

Ecran permettant au client de convertir leur monnaie en jeton.

#### 6.3.1.4 Back office

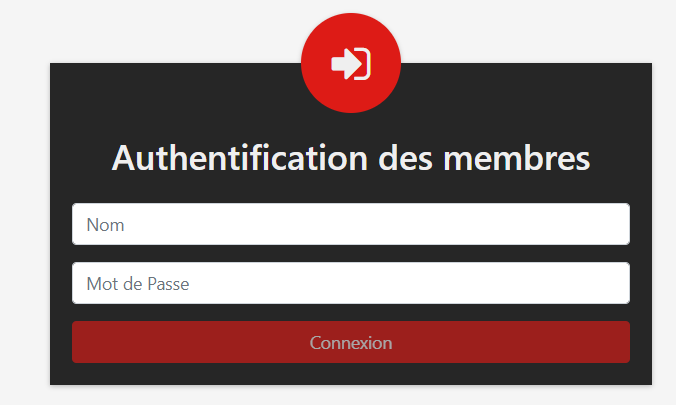


Figure 22:Authentification des membres

La première page qui s’ouvre est l’interface d’authentification des membres. Cette page permet de sécuriser le site en entier

Une image contenant texte

Description générée automatiquement

Figure 23:Création pari

Dans cette page sera insérer les informations des paris à venir. Ainsi que les côtes de chaque champs.



Figure 24:Liste des paris

Dans cette page l’administrateur pourra supprimer, modifier un pari. Mais aussi insérer le résultat d’un match et modifier ainsi par ricochet les jetons des clients ayant parier.

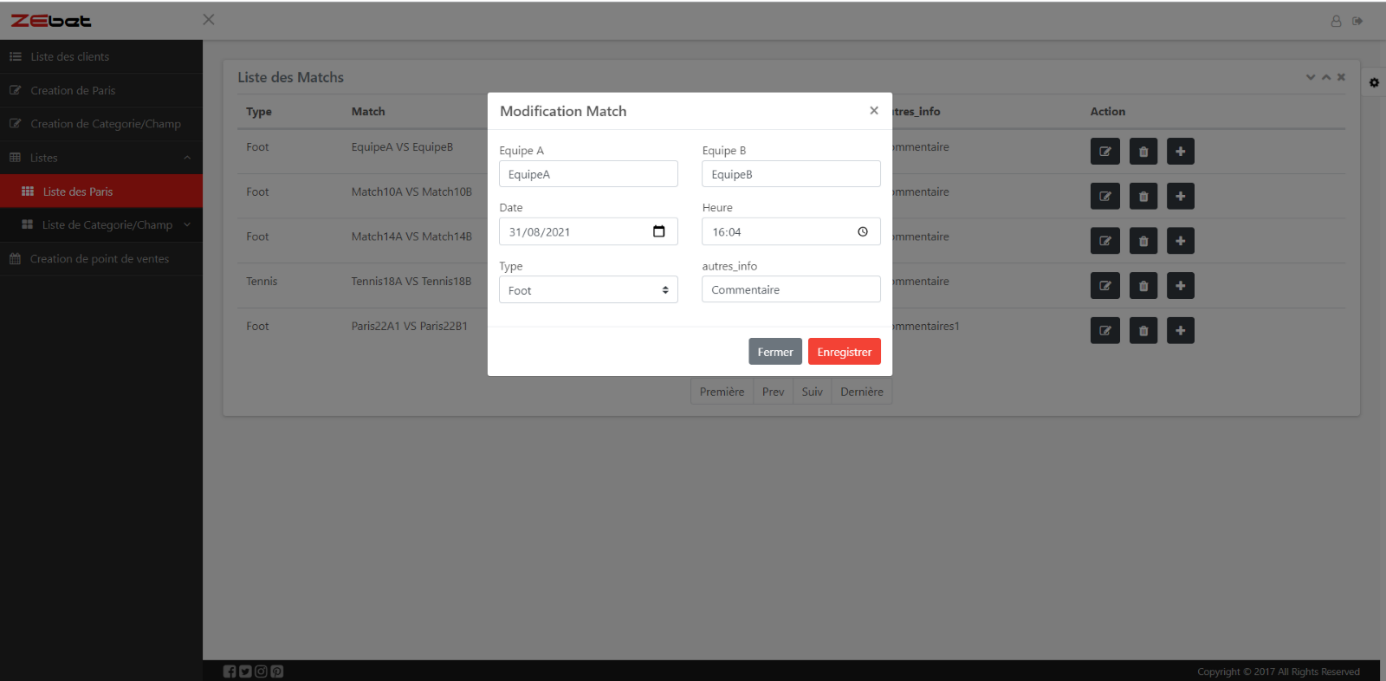


Figure 25:Modification d’un pari

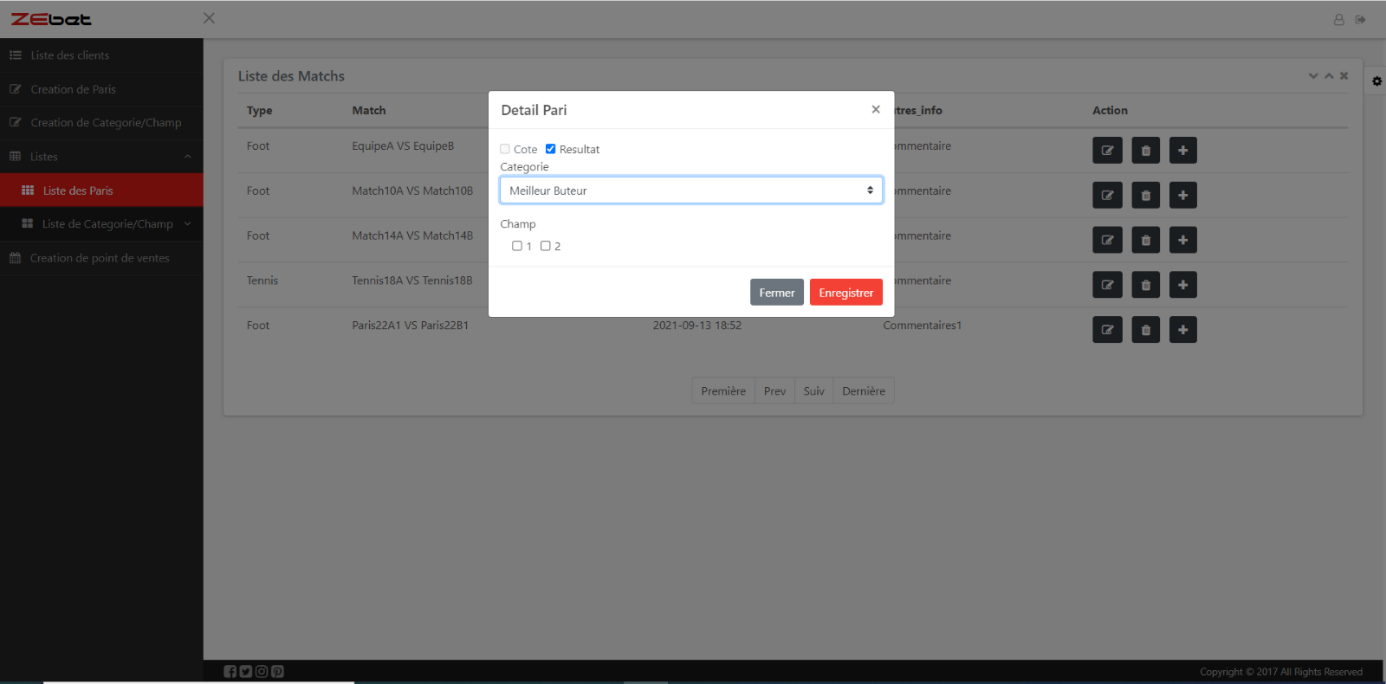


Figure 26:Insertion résultat en fonction d’une catégorie

### 6.3.2 Interfaces avec d’autres systèmes

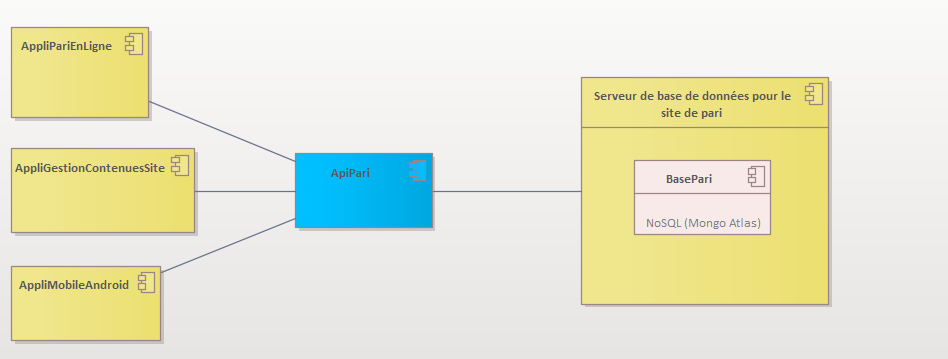
L’api développé avec **"Grails”** gère le service lié à la banque. Cette api communique avec l’application windows de la Banque.

L’api développé avec **“Node”** gère le service lié au site de pari en ligne, à l’application mobile et à la gestion des contenus du site

# 7. Architectures système

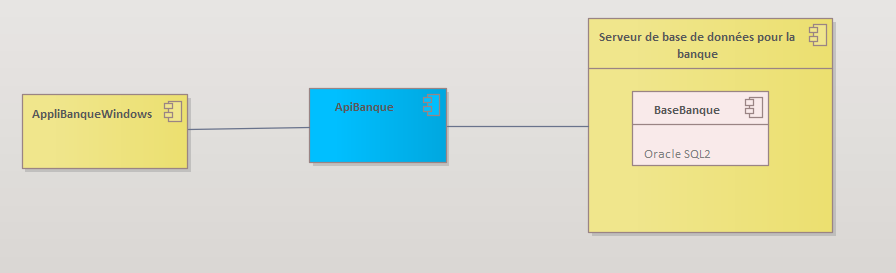
Nous avons deux architectures système pour la plateforme de pari en ligne :

* Le système de pari en ligne est produit sous la forme de cinq composants :
* Le serveur ainsi que la base de données NoSQL sur le cloud Mongo Atlas
* L’api qui interagit avec le serveur de base de données NoSQL
* Le site de pari en ligne
* Un site back office pour la gestion des paris en ligne
* Une application mobile Android

 Figure 27:Architectures système de la plateforme de pari en ligne

Le système de banque est produit sous la forme de 3 composants :

* Le serveur ainsi que la base de données Oracle SQL sur le serveur de l’université
* L’api qui interagit avec le serveur de base de données Oracle
* L’application Windows pour la gestion de la banque

 Figure 28:Architectures système de la plateforme du système bancaire

Les deux systèmes de la plateforme adoptent l’architecture en ***3 couches*** pour bénéficier des avantages de vitesse, d’évolutivité, des performances et de disponibilité.

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

## 8.1 Plateforme technique

Une image contenant texte, blanc, parking

Description générée automatiquement Figure 29:Architecture matérielle pour le système de pari en ligne

Une image contenant texte, shoji, fenêtre

Description générée automatiquement Figure :Architecture matérielle pour la banque

## 8.2 Conception du logiciel développé

### 8.2.1 Conception du code source

### 8.2.2 Le code source – vue statique

Une image contenant texte

Description générée automatiquement

Figure 31:Packaging de l’api Banking et l’api du site

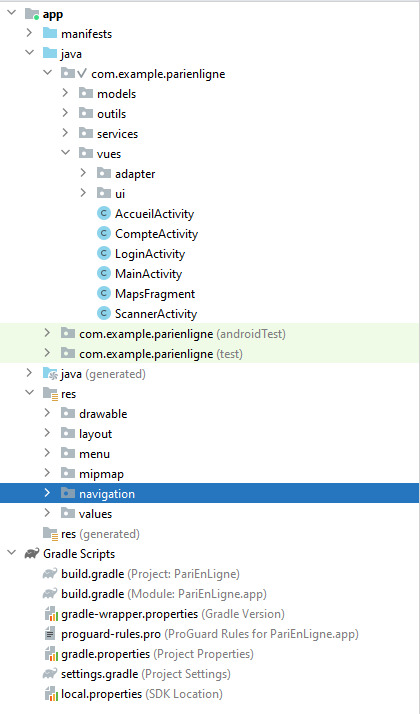


Figure 32:Packaging du projet mobile sur Android studio

Une image contenant texte

Description générée automatiquement

Figure 33:Packaging du projet application lourde sur Visual Studio

### 8.2.3 Modélisation de données

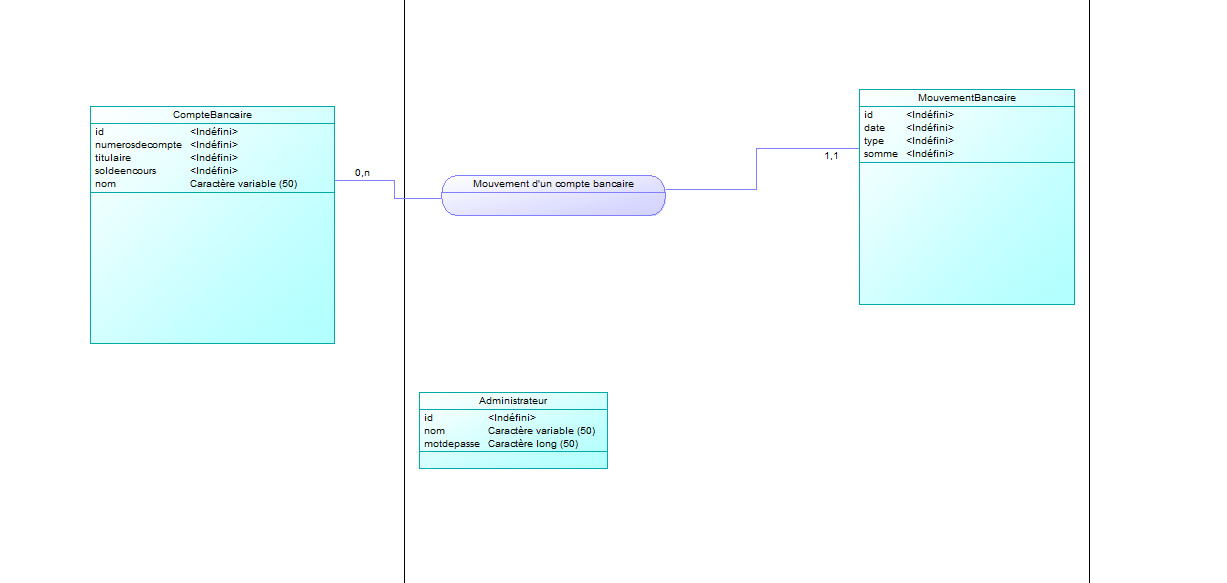
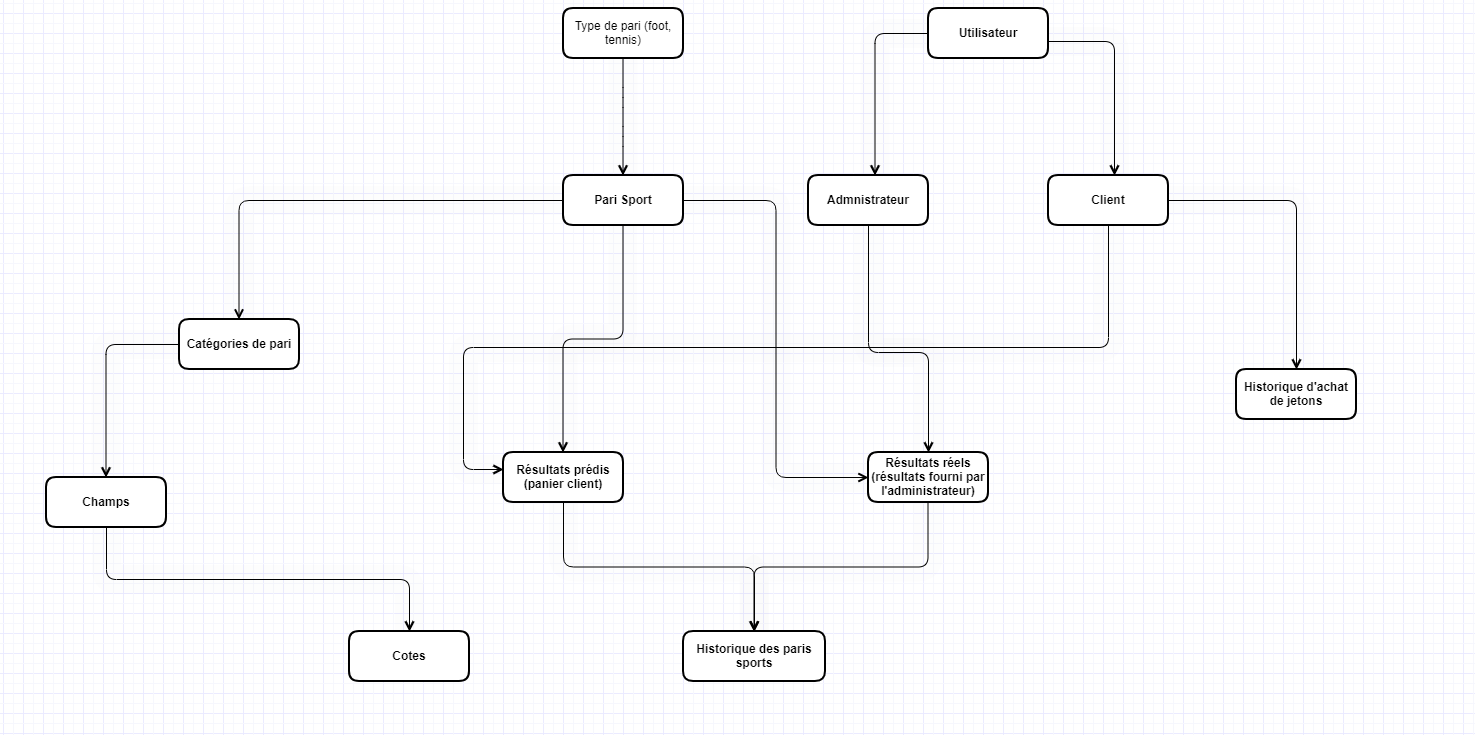
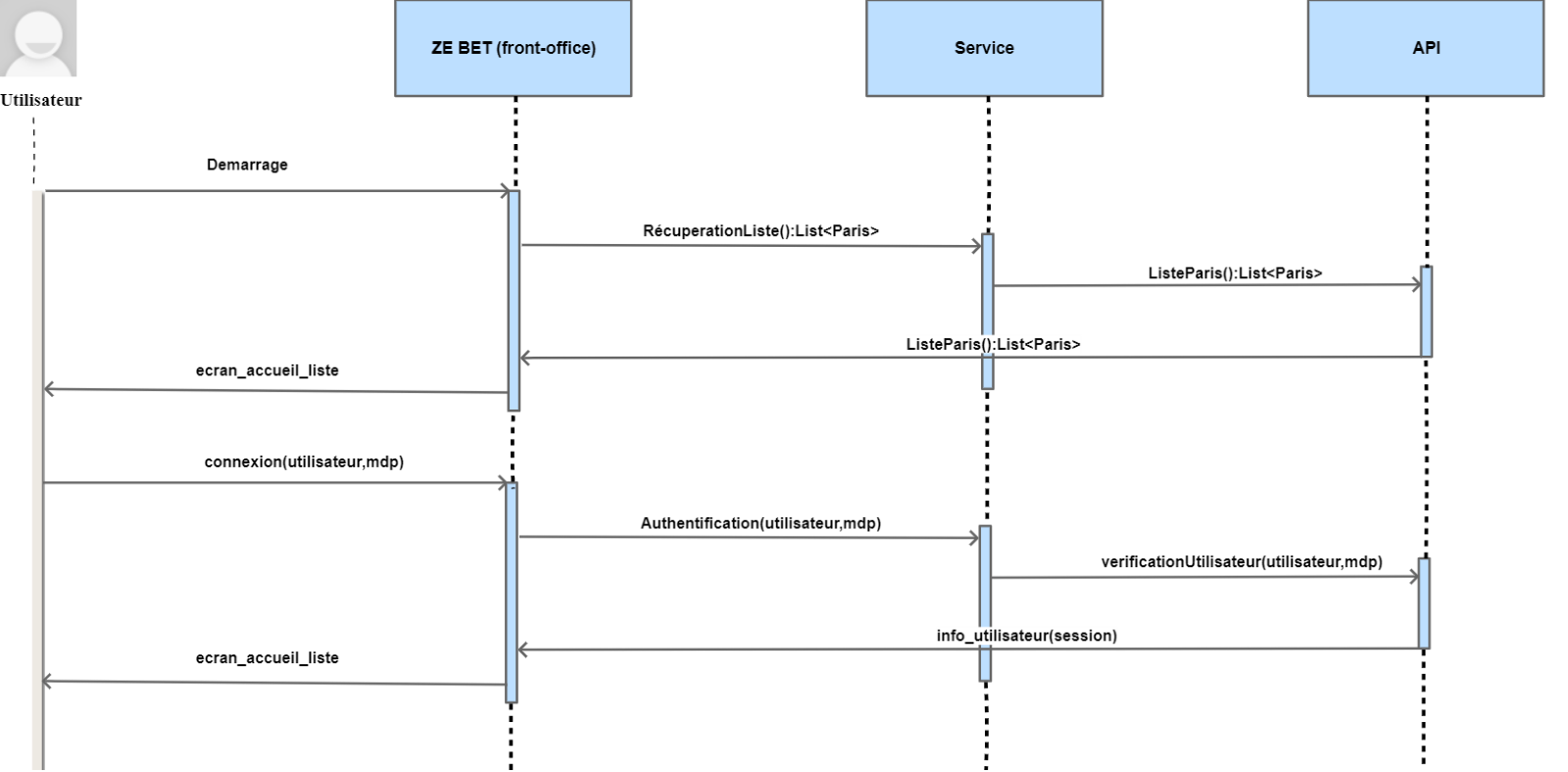
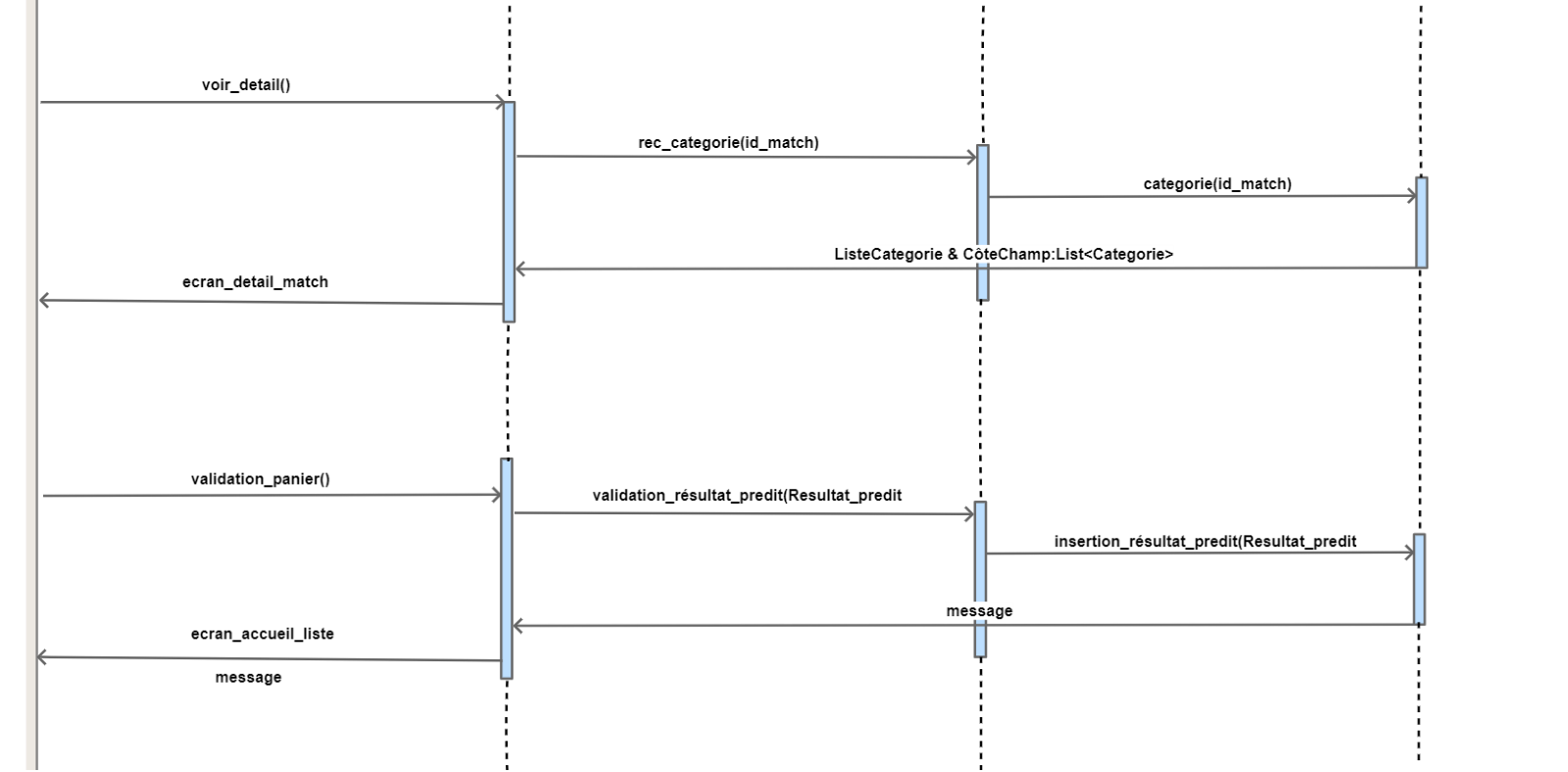


Figure 34:MCD de la base de données Oracle pour la banque

Figure 35:Schémas de la base de données NoSQL pour la gestion des paris

### 8.2.4 Réalisation des cas d’utilisation

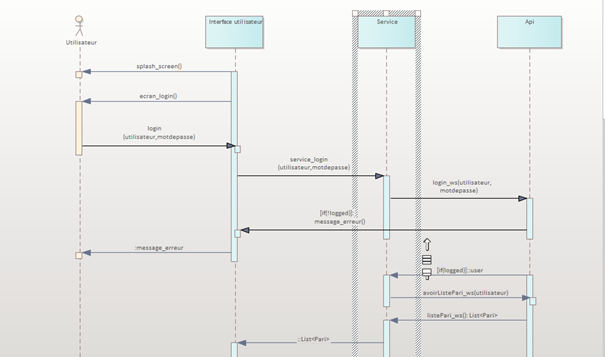




Une image contenant texte, intérieur, sombre, capture d’écran

Description générée automatiquement

Figure 36:Diagramme de séquence correspondant au site de pari en ligne



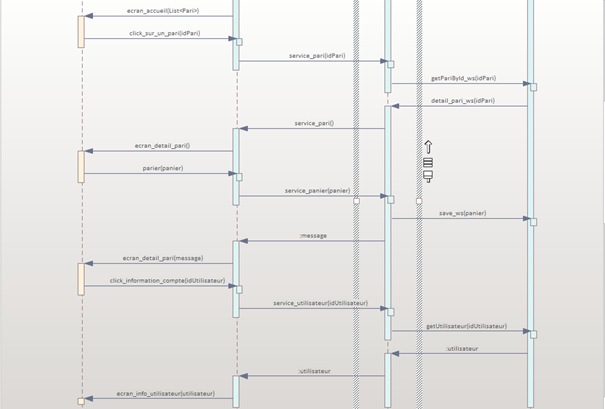


Figure 37:Diagramme de séquence pour l’application mobile

### 8.2.5 Les composants et leur déploiement

**Api grails (Banking)**

Site de déploiement : Heroku

Base de données : Base de données de l’université.

**Api node**

Site de déploiement : Heroku

Base de données : Mongo Atlas

**Back-office :**

Site de déploiement : Heroku

**FrontOffice :**

Site de déploiement : Heroku

**Application mobile :**

Application : .apk

**Client lourd :**

Application : .exe

# 9. Tests du système logiciel

Test Api Node :  ok

Test Api Java : ok

Test du système de localisation : ok

Test d’authentification via QR code: ok

Test des fonctionnalités de pari : ok

Test de panier simple pour le site et l’application mobile : ok

Test de gestion de contenu back-office : ok

# 10. Conclusion générale

Pour conclure, ce projet nous a permis la mise en œuvre et le perfectionnement des connaissances acquises dans notre cursus académique. Il nous a également permis de gérer un grand projet informatique en groupe en employant la méthodologie agile, mais surtout d’améliorer notre capacité à travailler en équipe.

# 11. Références et Bibliographie

Modélisation des bases de données NoSQL :

<https://www.gliffy.com/>

Documentation officielle du framework Angular :

<https://angular.io/>

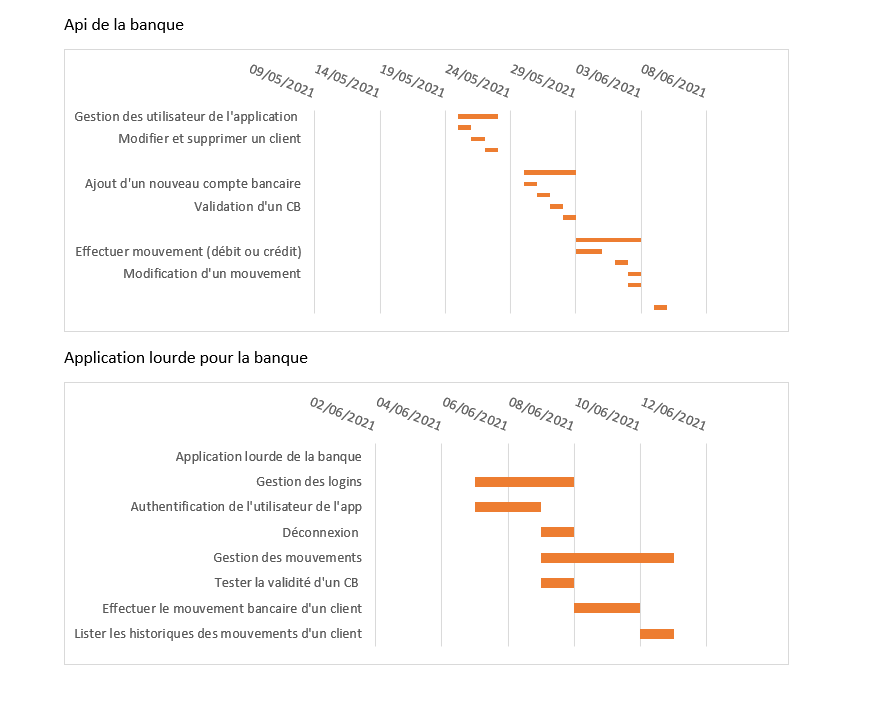
Implémentation de google Maps dans android studio :

<https://developers.google.com/maps/documentation/android-sdk/start>

Implémentation de JWT Token dans NodeJS :

<https://www.section.io/engineering-education/how-to-build-authentication-api-with-jwt-token-in-nodejs/>

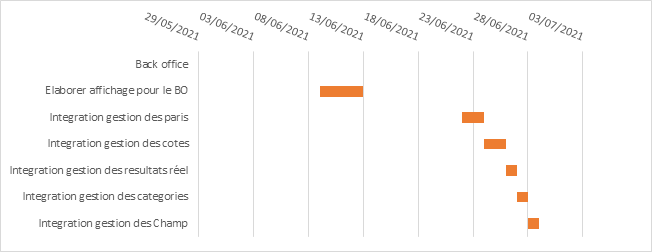
# 12. Annexes



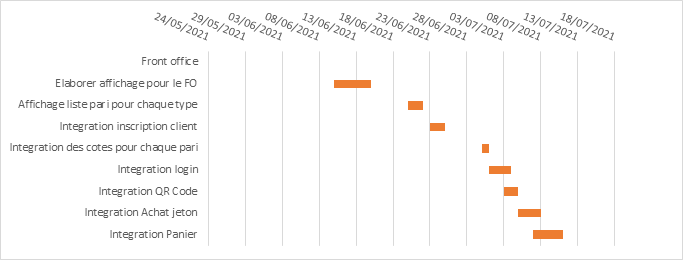
Api pour la gestion site de pari (front office et back office) et l’application mobile



Back office du site de pari



Front office du site de pari



Application mobile pour le pari

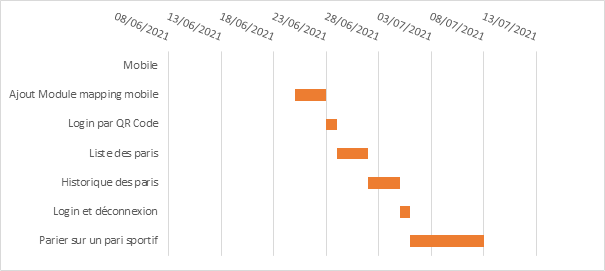
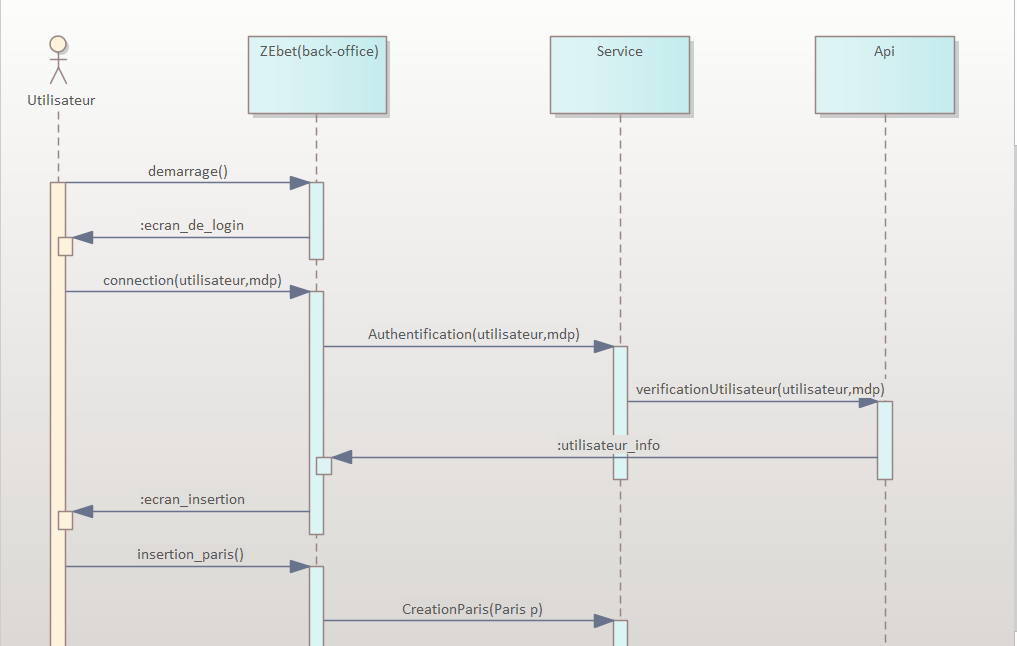
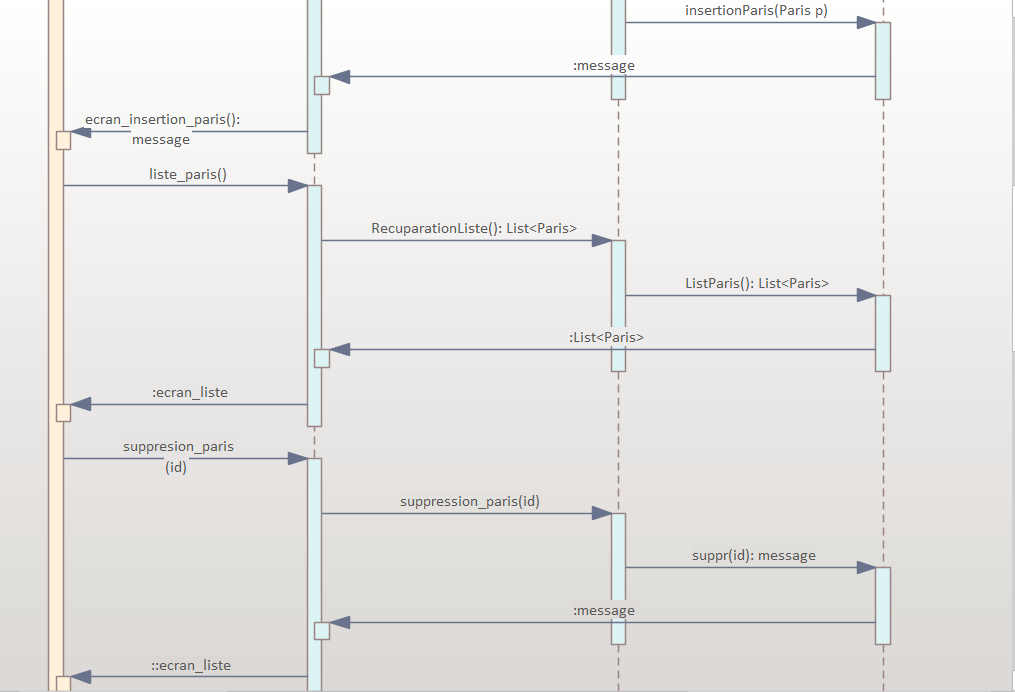


Diagramme de séquence correspondant au back-office:





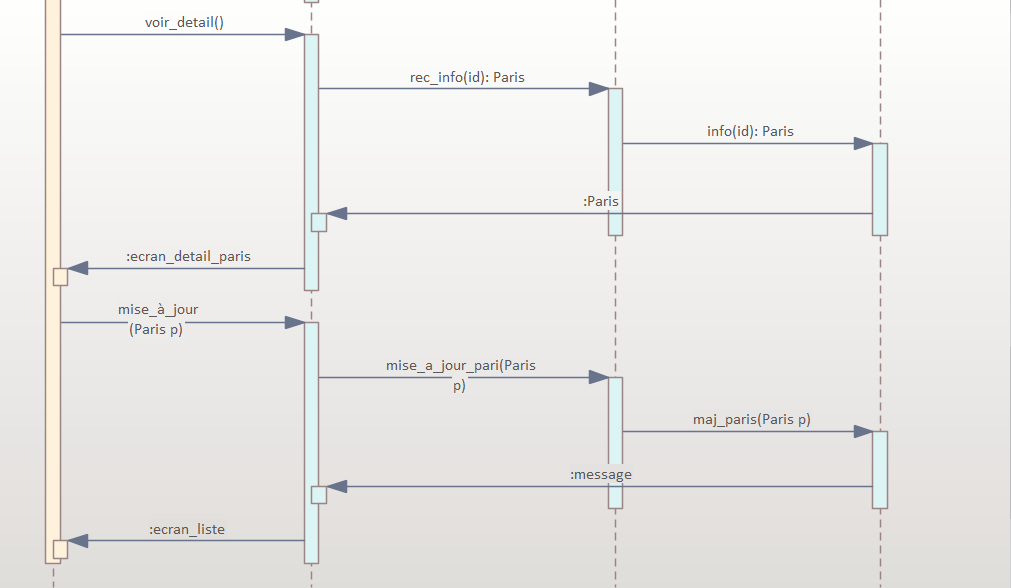
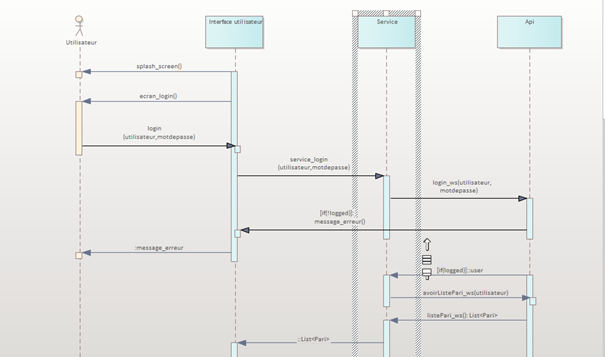


Diagramme de séquence correspondant à l’application mobile :



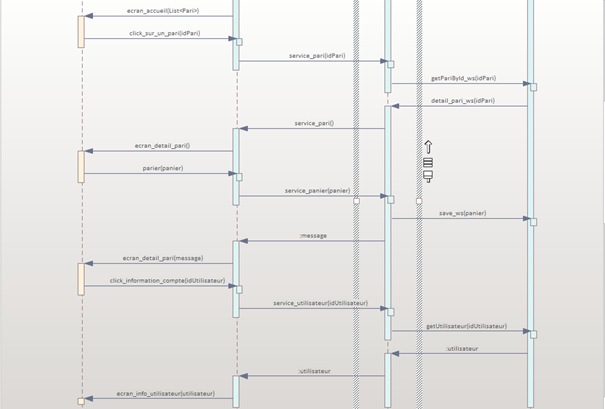
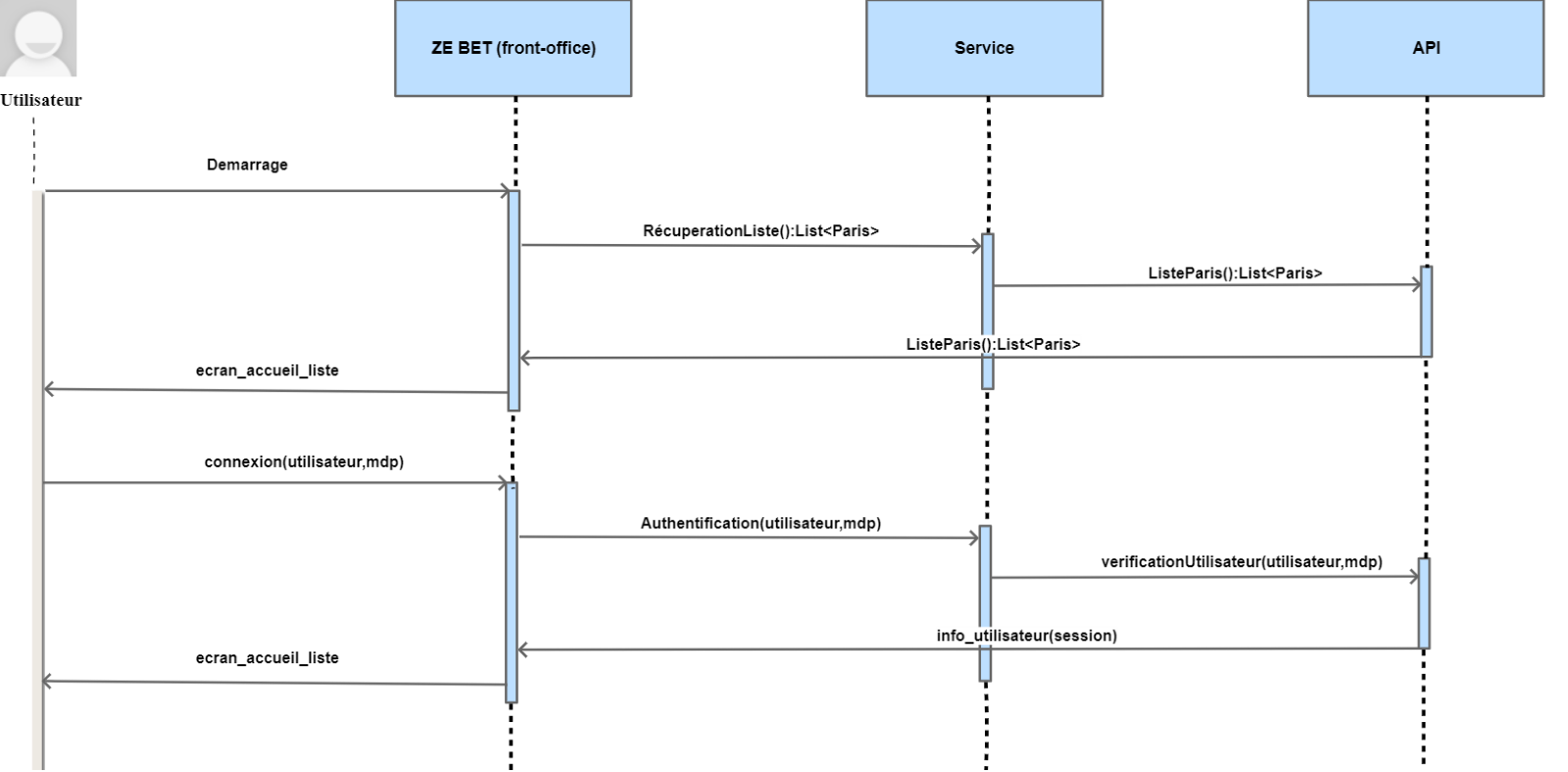
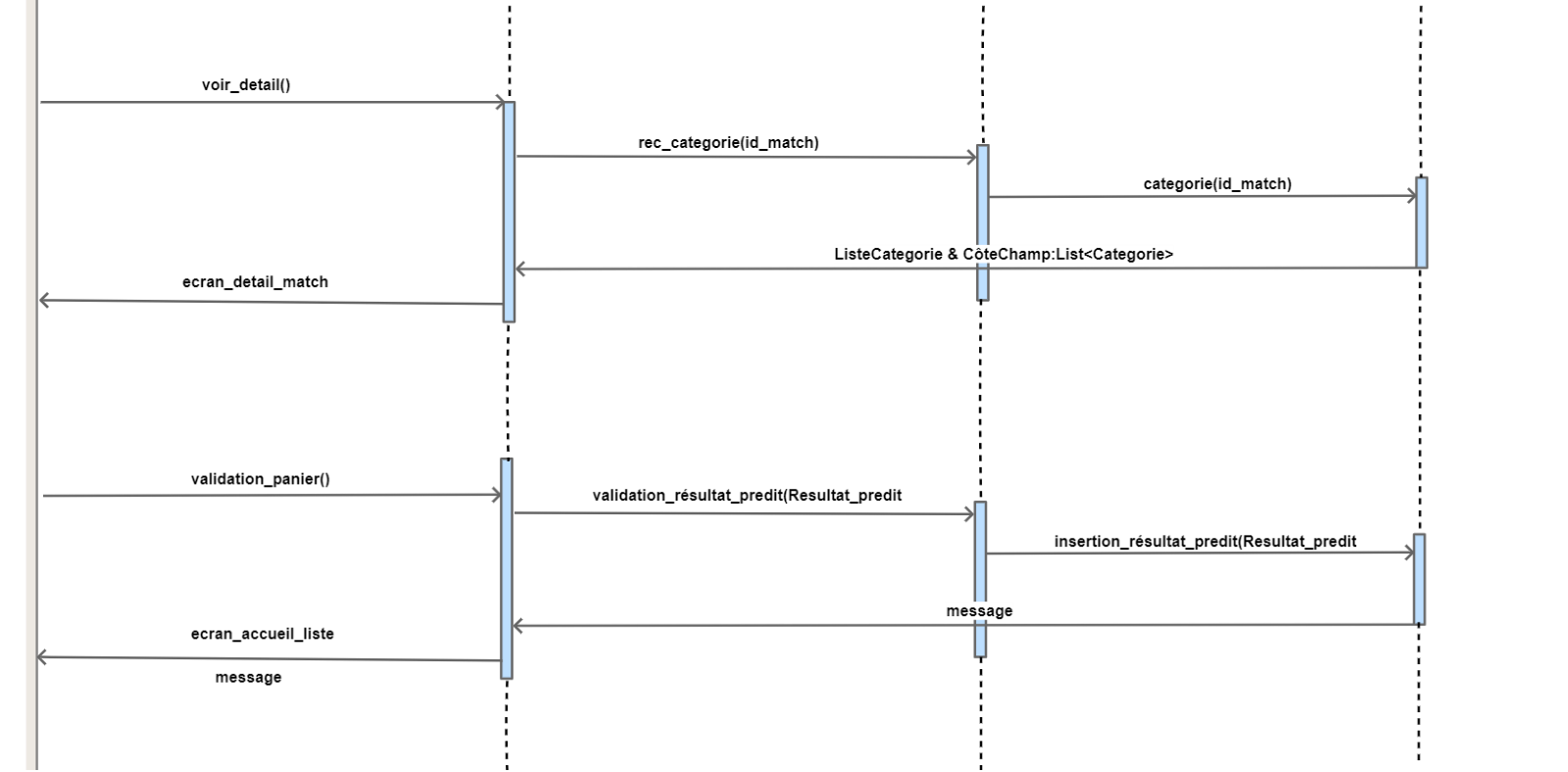


Diagramme de séquence correspondant au site de pari en ligne





Une image contenant texte, intérieur, sombre, capture d’écran

Description générée automatiquement