

[**I**nstitut **I**nternational **d**e **T**echnologie **- S**fax](http://www.iit-nau.com/)

**المدرسة العليا الدولية الخاصة**

**للتكنولوجيا بصفاقس**

**Rapport du Projet**

**Microservice**

Réalisé par

**Ellouze feres**

**Ellouze Mohamed**

Encadré par

**Mr JMAL Ahmed**

Spécialité

**Génie logiciel et informatique décisionnelle**

**Conception et développement d’une application web de gestion des résultats**

**de matches de football**

Année universitaire 2023-2024

**Tables des matières**

1. Introduction générale

1. Chapitre 1 : Cadre générale

2.1. Présentation du sujet

2.2. Analyse de l’existant

2.3. Conclusion

1. Chapitre 2 : Analyse et spécifications des besoins

Introduction

3.1. Définition des acteurs

3.2. Les besoins fonctionnels

3.3. Les besoins non fonctionnels

3.4. Technologies utilisées

Conclusion

1. Chapitre 3 : Conception

Introduction

* 1. Diagramme de classe
  2. Diagramme de cas d’utilisation

Conclusion

1. Chapitre 4 : Réalisation

Introduction

5.1. Environnement matériel

5.2. Environnement logiciel

5.3. Présentation de projet

Conclusion

1. Conclusion générale

# INTRODUCTION GENERALE

Dans le paysage dynamique du football tunisien, la gestion efficiente des résultats de matchs et des données liées aux équipes est cruciale pour maintenir l'intégrité et l'organisation du championnat de la Ligue Pro1. Traditionnellement, ces tâches ont été effectuées manuellement, ce qui peut entraîner des inefficiences, des erreurs et une complexité accrue dans la gestion des informations.

Notre projet, baptisé "GestChamp", se positionne comme une solution innovante visant à moderniser et à rationaliser le processus de gestion des résultats de matchs de football. À travers une approche microservices basée sur la plate-forme JEE et Spring Cloud, nous aspirons à fournir une plate-forme robuste et évolutive pour la saisie, la gestion et la visualisation des données liées à la Ligue Pro1.

Dans le premier chapitre, nous dresserons un cadre général pour le projet "GestChamp". Nous présenterons le contexte du championnat de la Tunisie de Football (Ligue Pro1) et expliquerons la nécessité d'une solution moderne pour la gestion des résultats de matchs. Nous effectuerons une étude approfondie de l'existant, examinant les méthodes traditionnelles de gestion des données de football, mettant en lumière les lacunes et les inefficiences. Enfin, nous discuterons des solutions proposées, justifiant ainsi le besoin d'une approche microservices avec JEE et Spring Cloud.

Dans le deuxieme chapitre, nous détaillerons les besoins fonctionnels et non fonctionnels de l'application "GestChamp". Nous identifierons les différentes fonctionnalités. Nous utiliserons un diagramme de cas d'utilisation général pour modéliser ces besoins, fournissant ainsi une vision claire des interactions entre les utilisateurs et le système. Ce chapitre servira de base solide pour le processus de conception ultérieur.

Le troisième chapitre sera consacré à la conception de "GestChamp". Nous expliquerons en détail certains cas d'utilisation spécifiques à travers un diagramme de classe général de l'application. Nous illustrerons la structure des classes, les relations entre les entités, et les différentes couches de l'application. Cette section fournira une vision technique approfondie de la manière dont le système sera mis en œuvre.

Dans le dernier chapitre, nous présenterons la réalisation concrète de "GestChamp". Nous détaillerons l'architecture utilisée pour développer l'application, mettant en avant les choix technologiques tels que Java Enterprise Edition (JEE) et Spring Cloud. Nous décrirons également les différents outils employés pendant le processus de développement et présenterons les composantes applicatives réalisées. Ce chapitre offre une vue pratique de l'implémentation du système, mettant en évidence les résultats tangibles obtenus.

# 

# CHAPITRE 1 : CADRE GENERAL

## 1.1. Présentation du sujet

Le monde du football moderne est en constante évolution, marqué par des compétitions féroces, des moments palpitants et des supporters passionnés. Dans ce contexte, la gestion efficace des résultats de matchs, des équipes et des classements est cruciale pour maintenir l'intégrité et l'organisation des championnats. Le championnat de la Tunisie de Football, en particulier la Ligue Pro1, ne fait pas exception à cette nécessité de gestion rigoureuse.

Notre projet, intitulé "GestChamp", vise à répondre à cette exigence en développant une application complète de gestion des résultats de matchs de football pour la Ligue Pro1. Notre approche se base sur les principes de l'architecture microservices, en utilisant la plate-forme Java Enterprise Edition (JEE) et le framework Spring Cloud pour assurer une extensibilité, une évolutivité et une facilité de maintenance optimales.

## 1.2. Analyse de l’existant

Après l’étude du système actuel, nous avons constaté que pour procéder à une évaluation de l'enseignant, l'administration a pris l'habitude de distribuer aux étudiants des formulaires à remplir dans lesquels on trouve un ensemble de questions auxquelles les étudiants sont censés répondre.

Ensuite, ils recueillent tous ces formulaires pour les consulter un par un pour chaque matière séparément, puis ils en tirent une conclusion finale à propos de l'évaluation de l'enseignant.

**1.3. Critiques de l’existant**

Dans le paysage actuel des applications de gestion des résultats de football, il est notable qu'aucune n'est spécifiquement dédiée au championnat tunisien. Il manque une plateforme centralisée pour fournir les dernières actualités, gérer les compositions d'équipes lors des matchs, et détailler les événements de chaque rencontre dans le cadre du championnat tunisien.

## 1.4. Solution proposée

## Le projet "GestChamp" a pour objectif de combler ce manque en développant une application web dédiée. Cette application vise à offrir un accès facile et rapide aux informations pertinentes, permettant aux amateurs de football de suivre de près l'évolution du championnat tunisien. Les fonctionnalités incluent des mises à jour en temps réel sur les résultats des matchs, des analyses détaillées, des informations sur les compositions d'équipes, les statistiques des joueurs, et d'autres aspects spécifiques au championnat de Tunisie.

## 1.5. Objectif à atteindre

L'objectif principal est de concevoir, développer et déployer une application spécifique pour la gestion des résultats de football du championnat tunisien. L'application doit centraliser les informations, offrir des mises à jour en temps réel, et fournir des détails approfondis sur chaque match, les compositions d'équipes, les statistiques des joueurs, ainsi que des analyses complètes des rencontres.

## Conclusion

En concluant ce premier chapitre consacré à la gestion des résultats de football, il apparaît clairement que l'environnement actuel présente des défis à relever. L'absence d'une plateforme centralisée dédiée au championnat tunisien ouvre la voie à la nécessité d'une solution innovante. Les lacunes identifiées, telles que la dispersion des informations et le manque de mises à jour en temps réel, soulignent l'urgence d'une approche modernisée.

Cette analyse initiale constitue la pierre angulaire de notre projet, préparant le terrain pour la spécification des besoins, la conception, et finalement, la réalisation d'une application de gestion des résultats de football qui répondra aux attentes des utilisateurs tout en surmontant les limitations actuelles. Notre objectif est clair : offrir une expérience sportive en ligne complète, immersive et instantanée pour les passionnés du championnat tunisien.

# 

# CHAPITRE 2 : ANALYSE ET SPECIFICATIONS

## Introduction

Ce chapitre nous permet d'identifier les différentes fonctionnalités de notre futur système pour chaque type d'utilisateur.

Pour ce faire, nous recensons les besoins fonctionnels afin de comprendre la liste des exigences qui se traduisent par des besoins non fonctionnels.

Cette démarche implique l'identification des acteurs et la définition de tous les besoins qui seront ensuite représentés dans un diagramme de cas d'utilisation général.

## 2.1. Définition des acteurs :

## Administrateurs (Gestion de l'Application Web) :

## Les Administrateurs sont les utilisateurs responsables de la gestion globale de l'application web. Ils ont la capacité d'ajouter, modifier et supprimer des données liées aux joueurs, équipes, entraîneurs, planification de matchs, arbitres, et d'afficher le classement en temps réel.

## Utilisateurs (Consultation des Informations) :

## Les Utilisateurs sont les personnes qui utilisent l'application web pour consulter diverses informations. Ils peuvent filtrer les disponibilités du personnel, consulter l'effectif des équipes, la composition des équipes lors des matchs, les détails des matchs, les résultats des matchs, et le classement des équipes de la ligue pro 1 après chaque journée. Ils peuvent également effectuer des recherches sur les équipes et les joueurs.

## 2.2. Les besoins fonctionnels :

|  |  |
| --- | --- |
| **Acteur** | **Rôle** |
| Administrateur | * Gestion des Joueurs : Ajouter, modifier et supprimer des joueurs de la base de données. * Gestion des Équipes : Ajouter, modifier et supprimer des équipes participant au championnat. * Gestion des Entraîneurs : Possibilité d'ajouter, modifier et supprimer des informations sur les entraîneurs des équipes. * Planification des Matchs : Permettre la planification des matchs, y compris les détails tels que la date, le stade, et les joueurs. * Gestion des Arbitres : Ajouter, modifier et supprimer des informations sur les arbitres officiant lors des matchs. * Affichage du Classement en Temps Réel : Offrir une vue instantanée et constamment mise à jour du classement des équipes dans la ligue pro 1. |
| Utilisateur | * Consultation de l'Effectif des Équipes : Permettre aux utilisateurs de consulter la liste complète des joueurs dans chaque équipe. * Consultation des Compositions de Match : Afficher les compositions des équipes qui participent à un match donné, y compris les titulaires et les remplaçants. * Consultation des Détails de Match : Fournir des informations détaillées sur un match spécifique, telles que la date, la journée, le stade, et les arbitres. * Consultation des Résultats : Permettre aux utilisateurs de consulter les résultats des matchs de chaque journée du championnat. * Consultation du Classement : Offrir un accès facile au classement des équipes de la ligue pro 1 après chaque journée. |

Ces besoins fonctionnels constituent une base solide pour la création de cette application web, en garantissant une expérience utilisateur fluide, une évaluation transparente et un système de notation équitable.

## 2.3. Les besoins non fonctionnels

Notre application doit répondre aux besoins suivants :

|  |  |
| --- | --- |
| **Besoin** | **Fonctionnalité** |
| La performance | Un site web doit être performant c'est-à-dire à travers ses fonctionnalités, répond à toutes les exigences d’une manière optimale. |
| La Scalabilité | L'application doit pouvoir être adaptée à l'évolution des besoins. |
| La convivialité | L’application doit fournir des interfaces conviviales c'est-à- dire simples et ergonomiques. Elle doit présenter un enchaînement logique entre les pages. |
| La rapidité du traitement | Vu le nombre important des opérations quotidiennes, il est impérativement nécessaire que la durée d’exécution des traitements soit la plus courte possible. |
| La compatibilité | L’application doit être compatible avec les différentes machines et navigateurs. |
| L’accessibilité | Plusieurs utilisateurs utilisent notre système simultanément. |

En intégrant ces besoins non fonctionnels dans la conception et le développement du l’application web, on garantit une expérience utilisateur satisfaisante et une performance optimale du système.

## 2.4. Technologies utilisées

### 2.4.1. Frontend : Angular



* **Framework Frontend Moderne :** Angular est choisi comme framework frontend en raison de sa modernité, de sa robustesse, et de son support actif par une large communauté de développeurs.
* **Interfaces Utilisateur Dynamiques :** Angular permet de créer des interfaces utilisateur dynamiques, réactives et esthétiques, offrant ainsi une expérience utilisateur améliorée.
* **Structuration Modulaire :** La structuration modulaire d'Angular facilite le développement, la maintenance, et l'évolutivité de l'application frontend

### 

### 2.4.2. Backend : JEE microservices, MySQL, Spring Cloud

****

* **JEE Microservice :**
* Les services côté backend sont mis en œuvre en utilisant des microservices basés sur la spécification Java EE (JEE), permettant une conception modulaire et distribuée de l'application.

****

* **MySQL :**
* La base de données MySQL est choisie comme système de gestion de base de données relationnelle pour assurer la persistance des données liées aux matchs, équipes, joueurs, etc.



#### 

* **Spring Cloud :**
* L'utilisation de Spring Boot avec Spring Cloud facilite la création d'une architecture de microservices, offrant des fonctionnalités telles que la gestion de la configuration, la découverte des services, et la tolérance aux pannes.

## Conclusion

L’activité d’analyse des différents cas d’utilisations a permis de fournir une spécification complète des besoins issus du différentes technologies utilisées.

# CHAPITRE 3 : CONCEPTION

## Introduction

Après l'étape de l'étude préalable et la spécification des besoins, nous réservons cette partie à la conception détaillée, et l'étude technique de notre projet.

## Diagramme de classes

Le diagramme de classe représente la structure statique de l'application, montrant les classes, leurs attributs, opérations, et les relations entre elles. Il est essentiel pour modéliser l'organisation des données et des concepts dans le système.

Ci-dessous, le diagramme de classe de notre système :

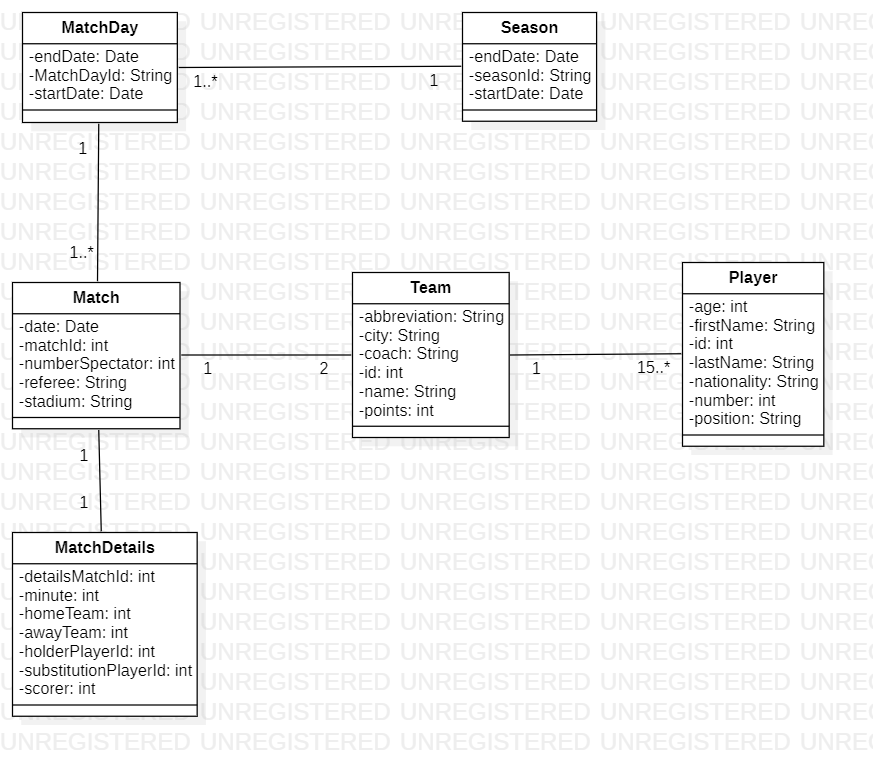


Figure 1. Diagramme de classe générale

Le diagramme montre une architecture micro-services pour une application de gestion des résultats de matches de football. L'application est composée de plusieurs services indépendants, chacun responsable d'une tâche spécifique.

Les services principaux de l'application sont les suivants :

* MatchService : Ce service est responsable de la gestion des données des matches, y compris le nom des équipes, le score, le nombre de buts marqués par chaque équipe et le classement des équipes après le match.
* TeamService : Ce service est responsable de la gestion des données des équipes, y compris le nom de l'équipe, le logo de l'équipe, la ville de l'équipe, etc.
* PlayerService : Ce service est responsable de la gestion des données des joueurs, y compris le nom du joueur, le numéro du maillot, la position du joueur, etc.

## Diagramme de cas d’utilisation

Le diagramme de cas d'utilisation est un outil de modélisation visuelle utilisé dans l'ingénierie logicielle pour représenter les interactions entre différents acteurs (utilisateurs ou systèmes) et un système informatique.

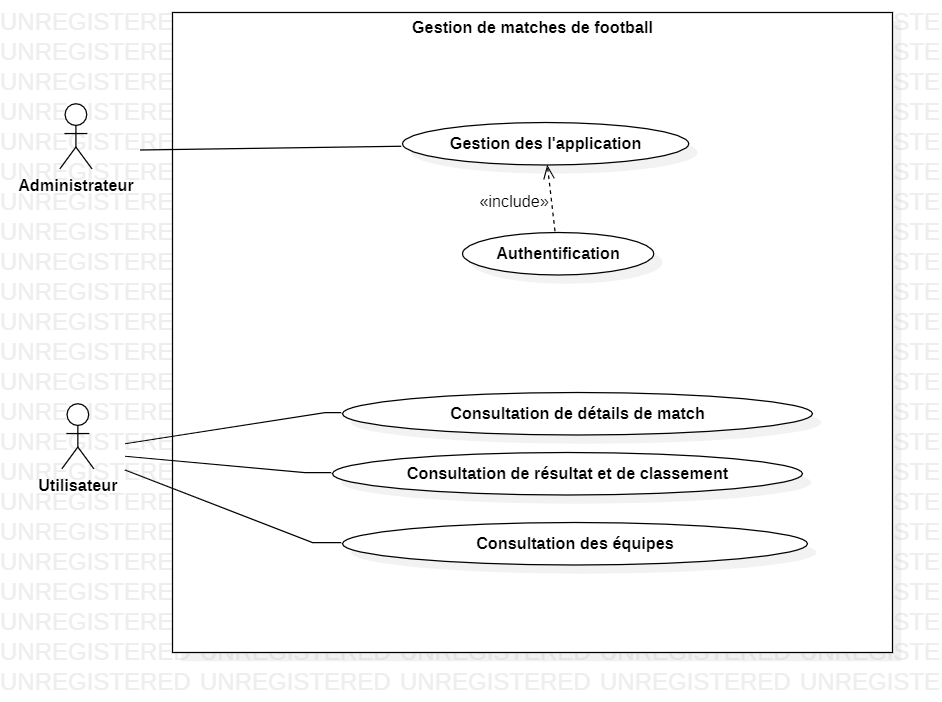


Figure 2. Diagramme de cas d’utilisation générale

## Conclusion

Comme nous pouvons le constater, l’activité de la conception a facilité la compréhension de notre système par le diagramme de classe générale.

Le chapitre suivant sera dédié pour la phase de réalisation. Cette phase va englober la création de notre application.

# CHAPITRE 4 : RÉALISATION

## Introduction

Dans ce chapitre, nous allons présenter l'architecture sur laquelle nous avons développé notre application, les différents outils utilisés ainsi que les composantes applicatives réalisées.

# 4.1. Environnement matériel

Nous avons élaboré ce travail sur nos PC personnel, dont la configuration est la suivante :

- Système d’exploitation : Windows 10 et 11.

- RAM : Varie entre 16 GO et 24 GO.

- Disque dur : Varie entre 500 GO et 1 TO.

## 4.2. Environnement logiciel

### 4.2.1. IDEs

* **Visual Studio Code :**

Visual Studio Code (VS Code) est un éditeur de code source léger et performant, développé par Microsoft. Il est largement utilisé par les développeurs pour divers langages de programmation. Les caractéristiques notables de VS Code comprennent une interface utilisateur minimaliste, une intégration native avec Git pour le contrôle de version, un support étendu pour des langages via des extensions, et une multitude de fonctionnalités telles que le débogage, l'autocomplétions, et la gestion des extensions.

* **Spring Tool Suite :**

Spring Tool Suite est un environnement de développement intégré (IDE) basé sur Eclipse, spécialement conçu pour le développement d'applications Java Spring.

Il offre des fonctionnalités avancées telles que la création rapide de projets Spring, l'intégration avec les outils de déploiement de serveurs, la gestion des dépendances Maven, un support robuste pour Spring Boot, et des outils de débogage et de profilage dédiés aux projets Spring.

* **Intellij Idea :**

IntelliJ IDEA est un environnement de développement intégré (IDE) développé par JetBrains, largement utilisé pour le développement d'applications Java.

Il offre une gamme complète de fonctionnalités pour améliorer la productivité des développeurs Java.

* **Laragon :**

Laragon est une plateforme de développement web locale tout-en-un qui simplifie la configuration et la gestion des environnements de développement. Il est principalement utilisé pour les applications web basées sur PHP et MySQL.

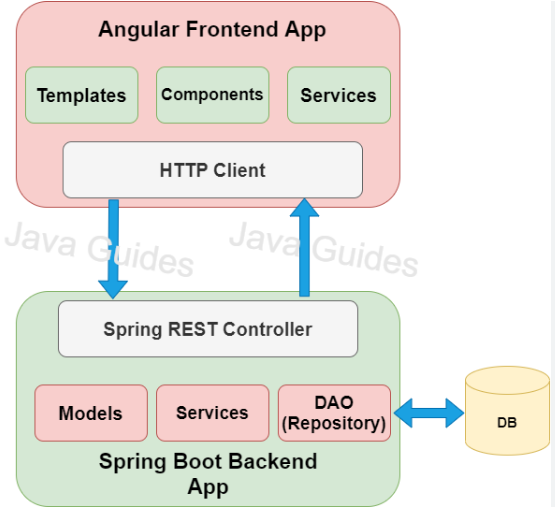
Laragon offre une installation facile et rapide d'un environnement complet comprenant Apache, MySQL, PHP, ainsi que des fonctionnalités telles que la gestion des bases de données, la configuration des domaines virtuels, et la possibilité d'intégrer des services tels que Redis ou Node.js.

* **StarUML :**

StarUML est un outil de modélisation UML (Unified Modeling Language) qui permet aux développeurs de créer des diagrammes pour visualiser, concevoir et documenter des systèmes logiciels.

### 4.2.4. Architecture et relation entre le Frontend et le Backend

Nous avons développé l'application selon l'architecture suivante :



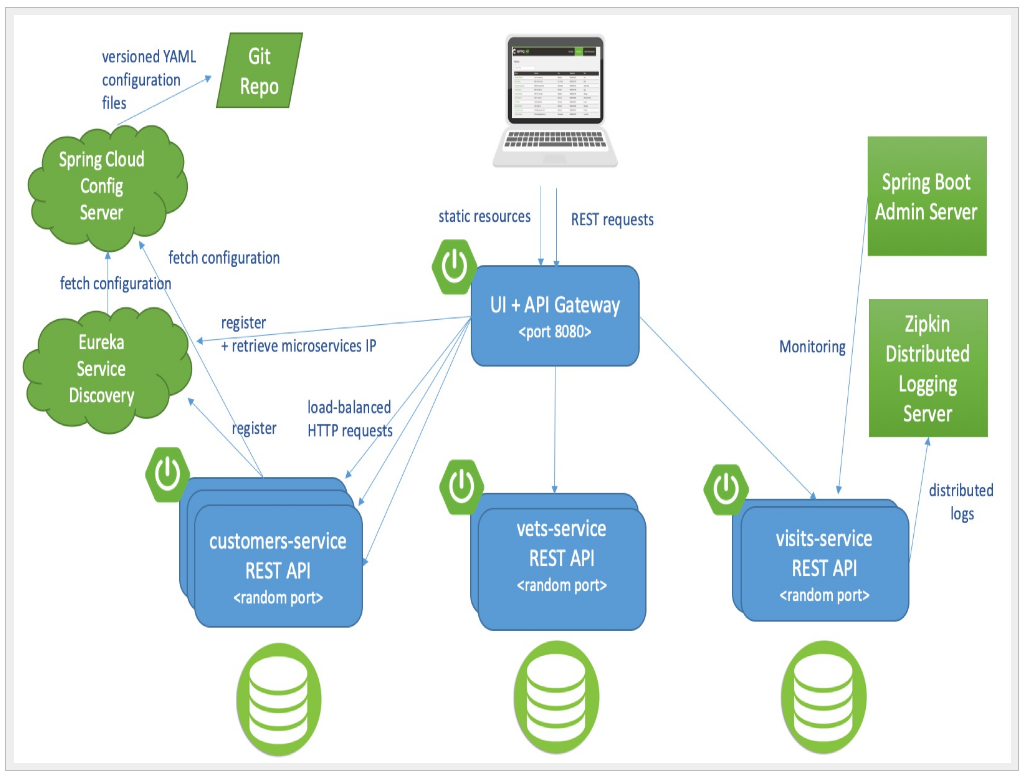


Figure 3. Architecture et relation entre le Frontend et la Backend.

## 4.3. Présentation de l’application

Nous présentons l’enchaînement de quelques interfaces accompagnées par leurs scénarios descriptifs :

* **Authentification :**

L’administrateur saisit son mail et son mot de passe pour pouvoir se connecter.

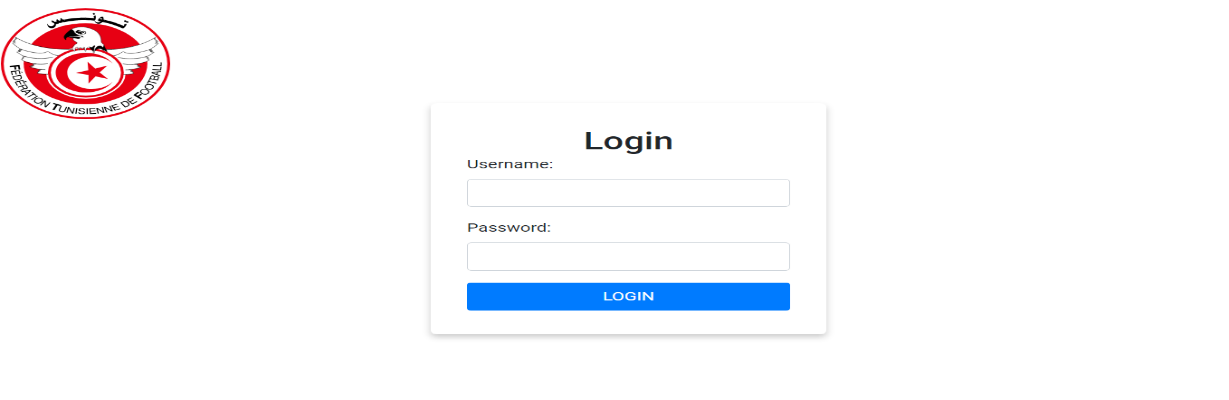
****

Figure 4. Authentification

* **Tableau de bord :**

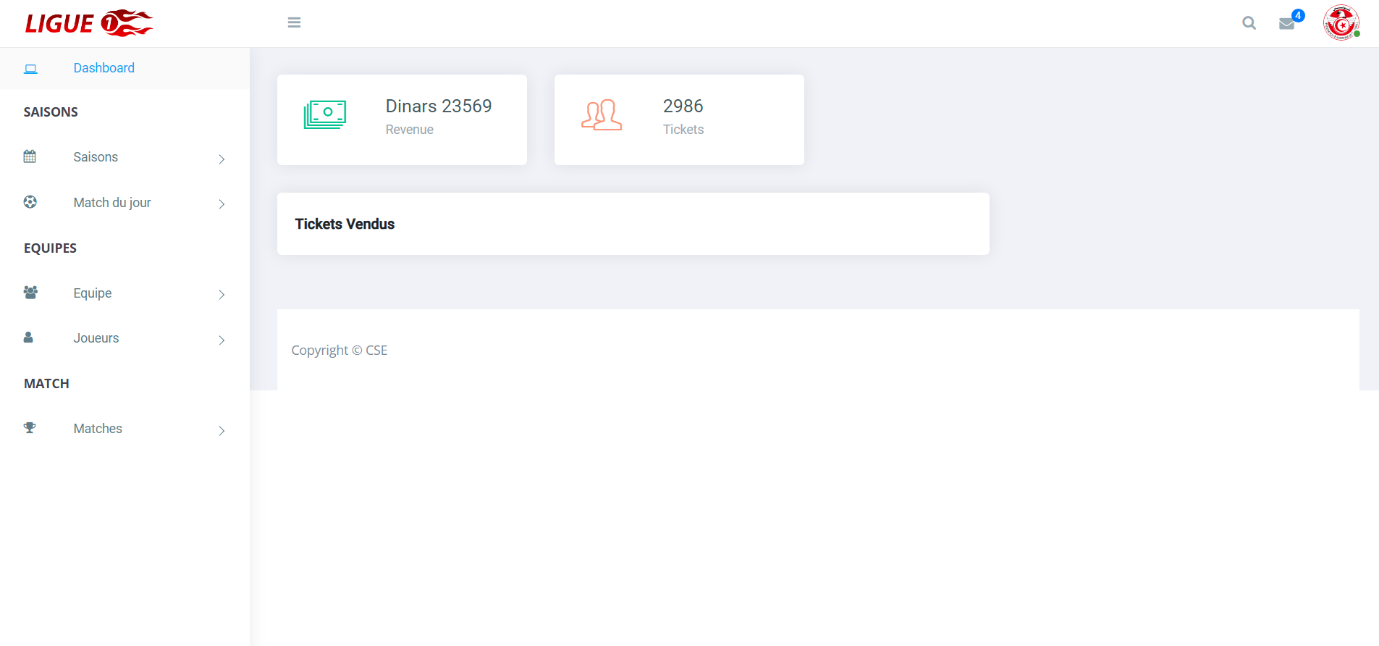
Voici la page qui s'affiche lorsque l'administrateur se connecte.

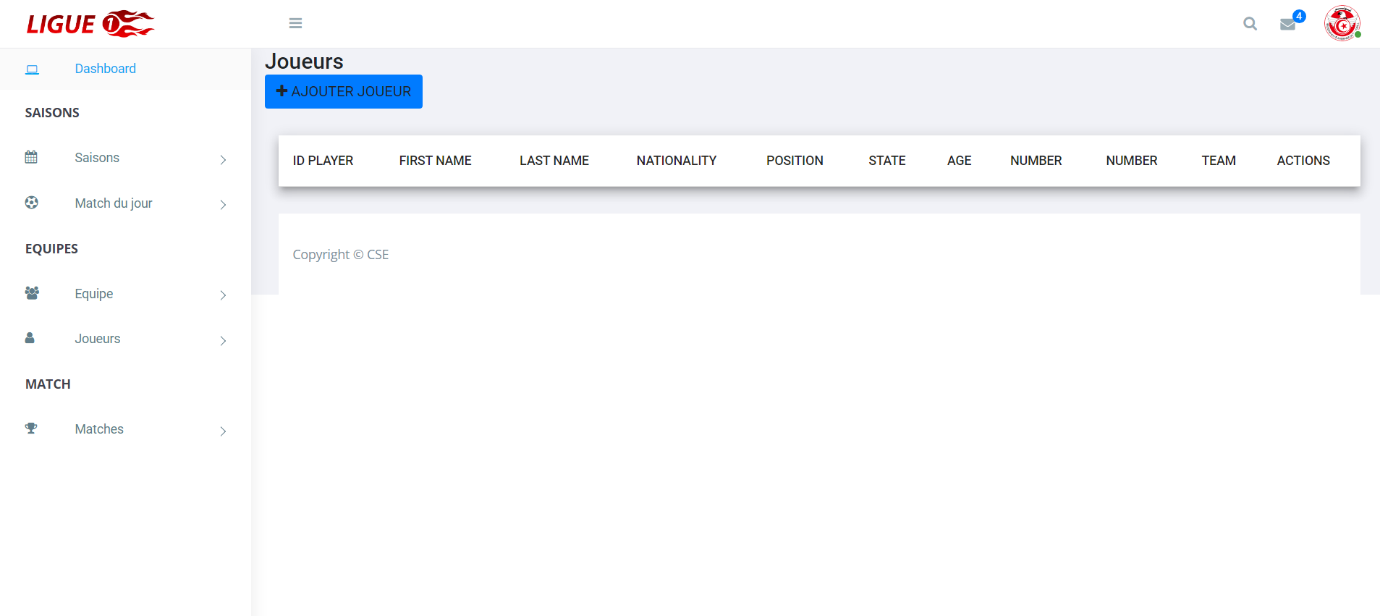
Figure 5. Tableau de bord

* **Vue utilisateur :**

Voici la page qui s'affiche lorsque l’utilisateur visite le site.



Figure 5. Vue utilisateur

* **Gestion des joueurs :**

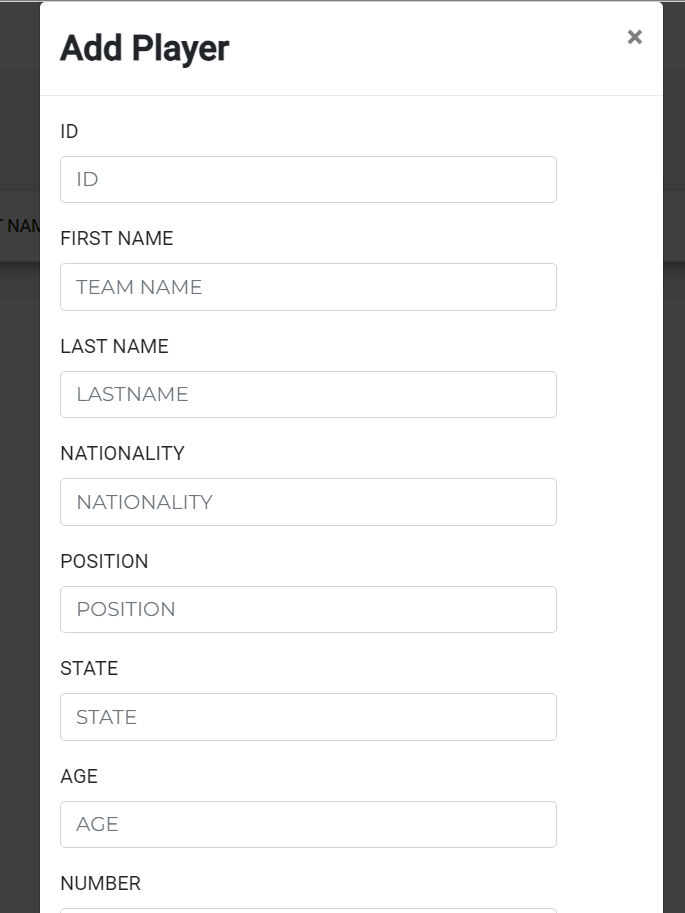
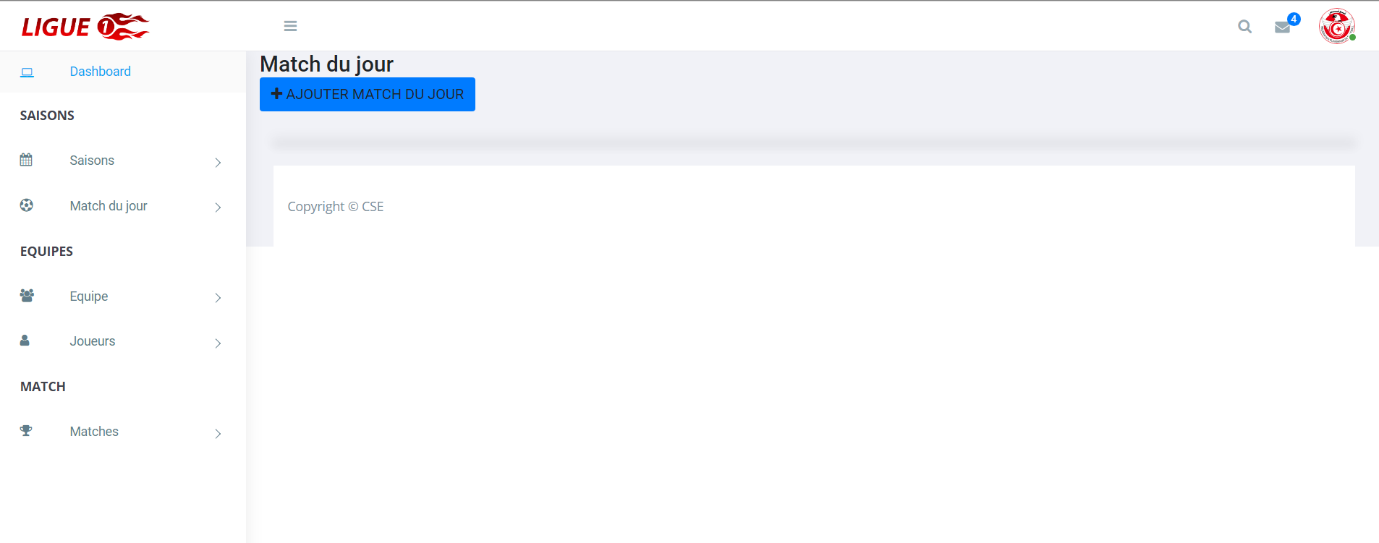


Figure 6. Gestion des joueurs

* **Gestion des matches :**

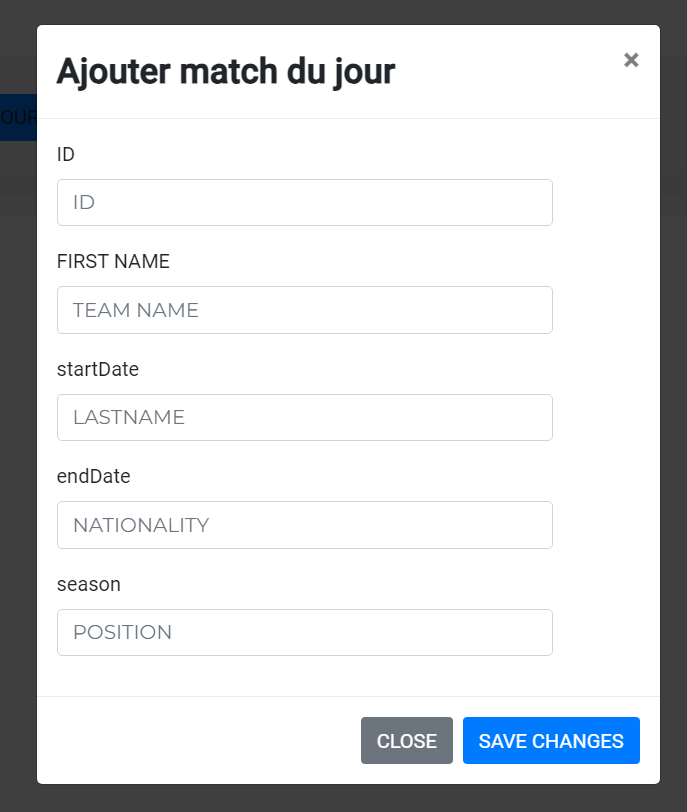
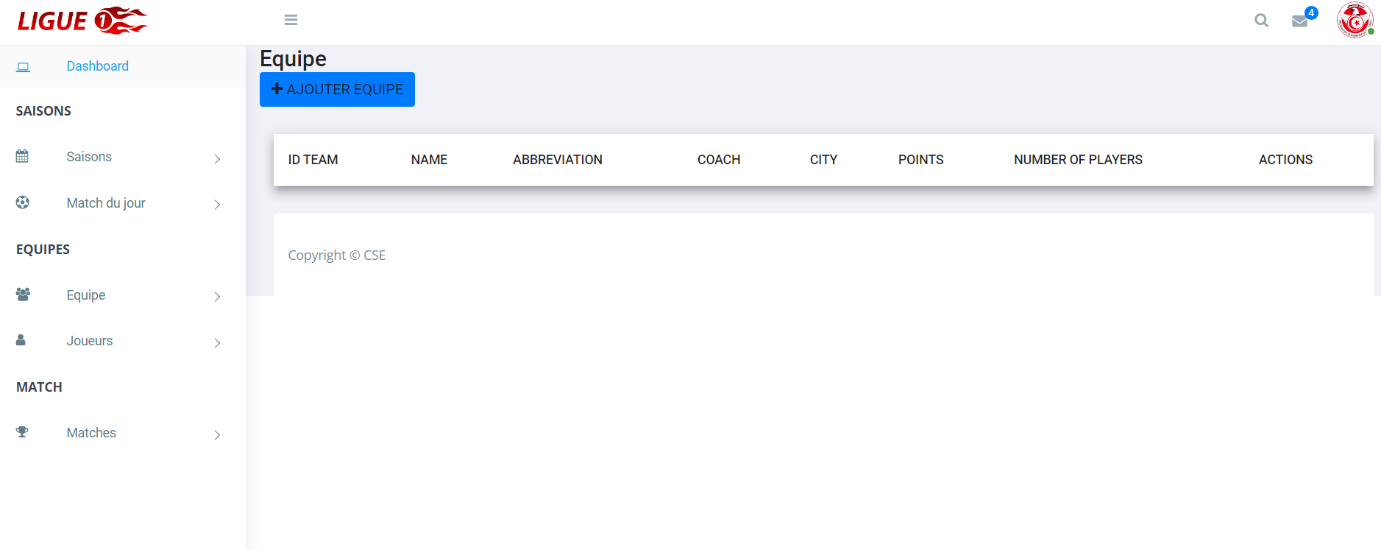
****

Figure 7. Gestion des matches

* **Gestion des équipes :**

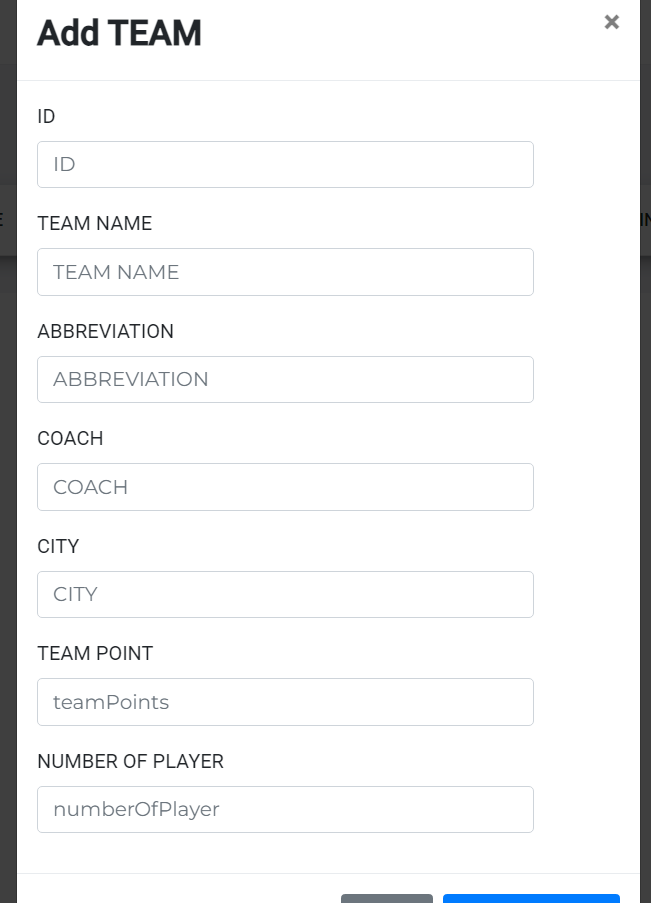
****

Figure 8. Gestion des équipes

## Conclusion

Dans ce chapitre, nous avons exposé la conception de notre application en expliquant les raisons derrière nos choix technologiques. Nous avons également présenté des captures d'écran des interfaces graphiques les plus significatives et fourni une brève description de la manière dont nous avons planifié notre projet.

# Conclusion Générale

Le projet "GestChamp" se positionne comme une réponse pragmatique aux défis de gestion des résultats de matchs de football dans le cadre du championnat tunisien. En se basant sur une infrastructure solide reposant sur la base de données MySQL, l'application aspire à centraliser de manière efficiente les informations, fournissant ainsi des mises à jour en temps réel et offrant des fonctionnalités de gestion complètes pour la Ligue 1 du championnat tunisien.

L'objectif principal de "GestChamp" est d'assurer une expérience utilisateur fluide et complète en consolidant les données liées aux matchs, aux saisons, aux équipes, aux joueurs et aux classements. Grâce à une architecture microservices basée sur JEE et Spring Cloud, le projet se distingue par sa capacité à évoluer avec les besoins futurs du championnat.

La centralisation des informations, les mises à jour en temps réel sur les événements des matchs, la gestion détaillée des équipes, et la consultation facile des classements positionnent "GestChamp" comme une solution exhaustive pour les amateurs de football, les parieurs et tous les acteurs impliqués dans le suivi et la gestion du championnat tunisien.

En somme, le projet "GestChamp" aspire à transformer la gestion des résultats de matchs de football en offrant une plateforme complète, transparente et adaptée aux besoins spécifiques de la Ligue 1 du championnat tunisien.