

**République Algérienne Démocratique et Populaire Ministère
de l'Enseignement Supérieur et de la Recherche Scientifique**



جامعة بجاية
Tasdawit n Bgayet
Université de Béjaïa

Université A/Mira de Béjaia

Faculté des Sciences Exactes

Département d'Informatique

MÉMOIRE DE MASTER

En

Informatique

Option

Administration et Sécurité des Réseaux

Thème

**Conception et réalisation d'une application
Web Pour la ligue de natation de Bejaia**

Réaliser par :

- M. HEDDAOUI Salem
- M. KADDOUR Mohamed Djebri

Soutenu le 03/07/2024 Devant Le Jury Composé De :

Président : Dr. Saadi Mustapha	MCB	UNIVERSITE DE BEJAIA
Examineur : Dr. Khenous Lachemi	MCB	UNIVERSITE DE BEJAIA
Encadrant : Dr. Djebari Nabil	MCB	UNIVERSITE DE BEJAIA

Années universitaire :2023/2024

Remerciements

Nous tenons à exprimer notre profonde gratitude à tous ceux qui ont contribué, de près ou de loin, à l'aboutissement de ce travail.

*À notre encadreur, **Dr. Nabil Djebari**, pour sa guidance précieuse, ses conseils avisés et son soutien constant tout au long de notre projet.*

*Nous adressons également nos sincères remerciements aux membres du jury **Dr.Saadi Mustapha et Dr. Khenous Lachemi**. Votre évaluation et vos suggestions ont été d'une grande aide pour améliorer notre travail.*

*Un grand merci à la **Ligue de Natation de Béjaïa** pour son accueil chaleureux et sa coopération qui ont grandement facilité notre recherche.*

*Nous remercions également **l'Université de Béjaïa** pour les ressources et l'environnement académique propice à l'apprentissage et à la réalisation de ce mémoire.*

Enfin, nous exprimons notre reconnaissance à nos familles et amis pour leur soutien inébranlable et leur encouragement tout au long de cette aventure académique.

Merci à tous ceux que nous aimons et qui ont rendu ce travail possible.

Dédicace

À mon cher père, que Dieu ait son âme,

À ma chère mère, dont les prières sincères m'ont constamment soutenu,

À mes frères, pour leur soutien moral,

À mes sœurs, dont le soutien et l'encouragement m'ont toujours accompagné,

À ma famille, mes proches, et tous ceux qui m'ont entouré d'amour et de vitalité,

*À mon binôme, **Kaddour Mohamed Djebri**, pour sa collaboration exceptionnelle,
son travail acharné et notre succès partagé.*

*À **Dr. Azizi Tarek**, mon ancien encadreur, pour ses conseils et son soutien durant
les premières étapes de ce projet.*

*À tous mes amis, **Badjouda Soumia, Djafour Oussama, Alaeddine Bencheikh
Lehocine, Litim Selma, Houtia Mohamed Yacine, Debabi Siefeddine**, pour leurs
encouragements constants et à qui je souhaite encore plus de succès.*

À tous ceux que j'aime, un grand merci.

HEDDAOUI Salem

Dédicace

À mon cher père, dont le soutien indéfectible et les sacrifices au fil des ans ont été inestimables,

À ma chère mère, dont les prières sincères m'ont constamment porté,

À mon frère, pour son appui moral,

À mes sœurs, dont le soutien et l'encouragement ,

À ma famille et mes proches, qui m'ont entouré d'amour et de vitalité,

*À mon binôme, **Heddaoui Salem**, pour sa collaboration exceptionnelle, son travail acharné et notre succès partagés.*

*À **Dr. Azizi Tarek**, mon ancien encadreur, pour ses conseils et son soutien durant les premières étapes de ce projet.*

*À tous mes amis, **Djafour Oussama , Badjouda Soumia, Alaeddine Bencheikh Lehocine, Litim Selma, Houtia Mohamed Yacine, Debabi Siefeddine**, pour leurs encouragements constants et à qui je souhaite un avenir encore plus radieux.*

À tous ceux que j'aime, un grand merci.

KADDOUR Mohamed Djebril

Table des matières

Table de matières.....	I
Liste des figures	V
Liste des tableaux	VI
Liste des abréviations:.....	VII
Introduction générale.....	1
Chapitre I : Technologie de l'information et de la communication	3
1.1.Introduction	3
1.2.Le web	3
1.2.1.L'historique du web.....	3
1.2.2.Définition :.....	4
1.3.Différence entre web et Internet.....	5
1.3.1.Portée	5
1.3.2.Fonctionnalité	6
1.3.3.Composants	6
1.4.Les technologies de web	6
1.5.Comparaison entre site web et application web	9
1.5.1.Site Web	9
1.5.2.Application Web.....	9
1.6.Langages et les outils utilisé pour réaliser une application web	9
1.6.1.Les langages	10
1.6.1.1- Langage de balisage/affichage.....	10
1.6.1.2- Langage client.....	11
1.6.1.3- Langage serveur.....	11
1.6.1.4- Base de données.....	12
1.6.2.Les outils de développement	12
1.6.2. 1- Serveurs Web :.....	12
1.7.Comparaison entre les Framework les plus utilisés	16
1.8.Conclusion.....	16
Chapitre II : Présentation de l'organisme d'accueil	18
2.1.Introduction	18
2.2.Natation	18
2.2.1.Définition.....	18
2.2.2.Histoire de la Natation.....	18
2.2.3.Types de Natation	18
2.2.4.Plongée	19
2.3.Ligue sportive.....	19
2.3.1.Structure et Fonctionnement des Ligues Sportives	19

2.3.2.Présentation de la Ligue de Natation de Béjaïa	19
2.3.3.Structure Organisationnelle de la Ligue	20
2.3.4.Organigramme de la Ligue	23
2.4.Problématique et Objectifs	23
2.4.1.Problématique	23
2.4.2.Objectifs.....	23
2.5.Conclusion.....	24
Chapitre III : Analyse et Conception.....	25
3.1.Introduction :	25
3.2.Cahier de charge	25
3.2.1.Les besoins fonctionnels.....	25
3.2.2.Les besoins non fonctionnels.....	26
3.3.Analyse.....	26
3.3.1.Méthode d'analyse et de conception	26
3.3.1.1- Présentation d'UML	26
3.3.2.Démarche.....	27
3.3.2 - 1. Identification des acteurs :	27
3.3.2 – 2. Identification des messages échangés entre les acteurs et le système :	27
3.3.2 - 3. Identification des cas d'utilisation :	30
3.3.2 -4. Diagramme de cas d'utilisation:	31
3.4.Conception	34
3.4.1.Les diagrammes de séquence	34
3 .4.1 - 1. Définition.....	34
3.4.1 - 2. Diagramme de séquence « S'authentifier ».....	34
3.4.1- 4. Diagramme de séquence « créer compétition ».....	36
3.4.1 - 5. Diagramme de séquence « créer Session ».....	39
3.4.1 - 6. Diagramme de séquence « gérer athlète »	41
3.4.2.Le diagramme de classes	44
3.4.2 -1. Définition.....	44
3.4.3.Conception de la base de données	46
3.4.3 -1. Le Modèle relationnel.....	46
3.4.3 -2. Le modèle physique de données	47
3.5.Conclusion.....	50
Chapitre IV : Implémentation et mise en œuvre	51
4.1.Introduction	51
4.2.Architecture globale de l'application	51
4.2.1.Partie Admin : l'architecture globale pour l'admin	51

4.2.2.Partie Agent : l'architecture globale pour l'agent	52
4.3.L'environnement de développement	52
4.3.1.Les langages de programmation utilisés.....	52
4.3.1- 1. HTML5	52
4.3.1- 2. Css3	52
4.3.1- 3. JavaScripts	53
4.3.1- 4. Bootstrap.....	53
4.3.1- 5. PHP (Hypertext Preprocessor) :	53
4.3.1 -6. Apache	55
4.3.1 -7. MySQL	55
4.4.Outils de développement.....	55
4.4.1.Environnement de développement intégré Visual Studio	55
4.4.1- 1. Caractéristiques principales	55
4.4.1- 2. Performance et utilisation	56
4.4.2.Le serveur XAMPP	56
4.4.2- 1. PhpMyAdmin	57
4.4.2- 2. Le Git :.....	57
4.4.2- 3. Le Framework Laravel	58
4.4.2- 4. Node.js.....	58
4.5.Présentation des interfaces de l'application	59
4.5.1.L'interface (Authentification) :	59
4.5.2.L'interface (Dashboard) :	59
4.5.3.L'interface (Profil) :.....	60
4.5.4.Les scenario :	61
4.5.5.Les fichiers exportés :.....	68
4.6.Conclusion :.....	71
Conclusion générale et perspectives	72
Bibliographie.....	73

Liste des figures

Figure 1. 1: Circuit d'une page web statique.	7
Figure 1. 2: Circuit d'une page web dynamique.	8
Figure 1. 3: Frontend.	10
Figure 1. 4: Backend.	11
Figure 2. 1: Organigramme de la ligue de natation de Bejaia.	23
Figure 3. 1: Diagramme de cas d'utilisation globale pour l'administration (Admin-ligue).	32
Figure 3. 2: Diagramme de cas d'utilisation globale (Agent-club).	33
Figure 3. 3: Diagramme de séquence « S'authentifier ».	35
Figure 3. 4: Diagramme de séquence « création de l'épreuve».	36
Figure 3. 5: Diagramme de séquence « création de compétition».	38
Figure 3. 6: Diagramme de séquence « création de Session».	40
Figure 3. 7: Diagramme de séquence « gérer athlète».	43
Figure 3. 8: Diagramme de classes de l'application.	45
Figure 3. 9: Diagramme de classes de l'application.	45
Figure 4. 1: Représentation de l'architecture globale de l'application(Admin).	51
Figure 4. 2: Représentation de l'architecture globale de l'application(Agent).	52
Figure 4. 3: L'interface (Authentification).	59
Figure 4. 4: L'interface (Dashboard).	60
Figure 4. 5: L'interface (Profil).	60
Figure 4. 6: L'interface (Création).	61
Figure 4. 7: L'interface (Demande).	62
Figure 4. 8: L'interface Création (épreuve).	62
Figure 4. 9: L'interface (épreuves).	63
Figure 4. 10: L'interface (Création compétition).	63
Figure 4. 11: L'interface (Compétition vide).	64
Figure 4. 12: L'interface (Session).	64
Figure 4. 13: L'interface (Ajouter des épreuve).	65
Figure 4. 14: L'interface (Session avec épreuves).	65
Figure 4. 15: L'interface (Compétition).	66
Figure 4. 16: L'interface "Compétition" : Après la création des compétitions.	66
Figure 4. 17: L'interface "Participation"	67
Figure 4. 18: L'interface "Demande".	67
Figure 4. 19: L'interface "Athlètes".	68
Figure 4. 20: L'interface "Liste de Départ".	68
Figure 4. 21: Liste des athletes.	69
Figure 4. 22: Les Carte.	69
Figure 4. 23: Start liste.	70
Figure 4. 24: Liste des Athletes participer.	70
Figure 4. 25: Bulletin d'engagement.	71

Liste des tableaux

Tableau 1. 1: tableau comparatif entre les Framework..... 16

Tableau 3. 1: Identification des messages échangés(Admin)29

Tableau 3. 2: Identification des messages échangés(Agent).30

Liste des abréviations:

API : Application Programming Interface.

BDD : Base De Données.

CSS: Cascading Style Sheets.

CMS: Content Management System.

DJTS : Direction Méthodologique chargée des Jeunes Talents Sportifs.

DDF : Direction Méthodologique chargée du Développement Sportif et de la Formation.

DOOSC : Direction Méthodologique chargée de l'Organisation Sportive Organiser et Superviser les événements sportifs, assurer le respect du Calendrier sportif.

DOM : Document Object Model.

DTW : Directeur Technique de Wilaya.

GNN : Global Network Navigator.

HTTP: HyperText Transfer Protocol.

HTTPS : HyperText Transfer Protocol Secure.

HTML: HyperText Markup Language.

IP: Internet Protocol.

IDE : Integrated Development Environnement.

IDEA : Integrated Development Environment for All.

JS : JavaScript.

JSF : Java Server Faces.

MySQL: My Structured Query Language.

MVC : Model View Controller.

PDF : Portable Document Format.

PHP : Pré Hypertext Preprocessor.

SGBD : Système de Gestion de Base de Données.

SQL : Structured Query Language.

TCP: Transmission Control Protocol

UML: Unified Modeling Language.

URL : Uniform Resource Locator.

VSCode : Visual Studio Code.

Introduction générale

L'innovation technologique du XXI^e siècle a transformé tous les aspects de notre quotidien, en touchant notamment l'éducation, le commerce et l'administration. Grâce à son accessibilité et à sa polyvalence, elle offre une multitude de services qui facilitent la gestion et l'accomplissement des activités professionnelles et personnelles. En tirant parti de ces ressources, il est possible de créer des solutions innovantes qui optimisent les processus administratifs, les interactions et les échanges.

Actuellement, la Ligue de Natation de Béjaïa rencontre des difficultés majeures dans la gestion des inscriptions, des compétitions et des résultats, en raison de processus manuels laborieux et d'une fragmentation des informations. Ces problèmes entraînent des retards, des erreurs et une inefficacité générale qui impactent négativement les performances et l'organisation des événements.

Ce site a pour but de moderniser et d'optimiser les processus administratifs en réduisant considérablement l'utilisation de la paperasse, ce qui facilite le travail des gestionnaires et des membres de la ligue. Grâce à cette plateforme, la gestion des inscriptions, des compétitions et des résultats devient plus efficace et accessible, offrant une interface conviviale et intuitive qui simplifie les tâches quotidiennes. En somme, ce site web représente une avancée significative vers une administration plus moderne et plus efficace, répondant aux besoins spécifiques de la Ligue de Natation de Béjaïa.

Pour mener à bien ce projet, nous avons organisé le travail en chapitres afin de couvrir les aspects essentiels du développement d'une application conforme aux besoins de la Ligue de Natation de Béjaïa. Le premier chapitre, intitulé **Présentation de l'organisation actuelle**, décrit la structure et les pratiques existantes de la Ligue de Natation de Béjaïa, mettant en avant l'importance de la technologie dans l'amélioration des processus actuels. Le deuxième chapitre, **La technologie de l'information**, explore les concepts clés et les applications web et internet dans un contexte moderne. Le troisième chapitre, **Analyse et conception**, se concentre sur la création d'une

Introduction générale

application web pour la gestion de la Ligue de Natation de Béjaïa, en détaillant les étapes de conception et les besoins fonctionnels. Enfin, le quatrième chapitre, **Implémentation et mise en œuvre**, décrit les processus de développement et les tests effectués pour assurer le bon fonctionnement et la fiabilité de l'application.

En conclusion, ce projet a permis d'améliorer considérablement l'efficacité administrative de la Ligue de Natation de Béjaïa, tout en offrant des perspectives prometteuses pour l'avenir.

CHAPITRE I :

Technologie de l'information et de la communication

1.1. Introduction

L'Internet, souvent considéré comme l'infrastructure de communication la plus révolutionnaire de notre époque, a transformé la manière dont nous interagissons, travaillons et accédons à l'information.

En reliant des milliards d'appareils à travers le monde via le réseau, l'Internet a permis l'émergence de multiples services et applications, tels que le World Wide Web, les services de messagerie et les transferts de fichiers.

Cette technologie repose sur des principes fondamentaux comme la connectivité universelle, les protocoles standardisés et la résilience. L'Internet englobe également des concepts comme l'Intranet et l'Extranet, chacun ayant ses propres usages et objectifs spécifiques.

Dans cette exploration, nous détaillerons les composantes essentielles de l'Internet, les services qu'il offre, et ses principaux protocoles comme IP et TCP.

Nous examinerons également l'histoire et les concepts du Web, avant de différencier les sites web des applications web, et de discuter des langages et outils utilisés pour leur développement.

1.2. Le web

1.2.1. L'historique du web

Le World Wide Web (WWW), communément appelé le Web, a été inventé par Tim Berners-Lee en 1989 alors qu'il travaillait au CERN. Il a développé les premières spécifications pour URL (Uniform Resource Locator), HTTP (Hypertext Transfer Protocol) et HTML (Hypertext Markup Language), qui sont les fondements du Web. Le premier site web a été mis en ligne en 1991, et il fournissait des informations sur le projet WWW.

Le Web 1.0, souvent appelé le Web statique, était caractérisé par des pages web statiques et des contenus principalement en lecture seule. Les utilisateurs pouvaient consulter des informations mais n'avaient que peu d'interactions avec les sites web. Dale Dougherty, pionnier de l'Internet et créateur du GNN (Global Network Navigator), le premier site avec publicité en 1993, est une figure centrale de cette histoire. Le GNN fut racheté par AOL en 1995, rappelant la tendance des sites Web 2.0 à être acquis par de grandes entreprises. Craig Cline, décédé en 2006, a également participé à la formulation du terme, bien que son rôle soit moins documenté.

Tim O'Reilly a tenté de s'approprier le nom « Web 2.0 » en tant que marque, notamment pour protéger ses conférences Web 2.0, ce qui a suscité des controverses. O'Reilly a reconnu une erreur de jugement et a expliqué vouloir protéger l'organisation des conférences. Cependant, le débat reste ouvert sur la nature et le contenu intrinsèque du Web 2.0, car O'Reilly ne peut breveter une idée ou un concept, seulement une offre de service. [1]

1.2.2. Définition :

Le Web permet de consulter des pages sur des sites, grâce à des navigateurs adaptés, et via des machines comme des ordinateurs ou des smartphones. Il s'agit un peu d'une immense bibliothèque de documents, qui repose sur les fondations complexes d'Internet. Le Web utilise de nombreuses technologies et protocoles pour être simple d'utilisation. C'est par exemple le cas de l'hypertexte, qui permet d'aller d'un site à un autre en cliquant sur un lien, ou les URL, qui identifient simplement l'adresse d'un site ou d'un contenu. [2]

- **Hypertexte** : L'hypertexte est une méthode de structuration de l'information qui permet de lier différents documents ou parties de documents entre eux. Les liens hypertexte, ou "hyperliens", permettent aux utilisateurs de naviguer facilement d'une page à l'autre.
- **URL (Uniform Resource Locator)** : L'URL est l'adresse unique qui identifie une ressource sur le Web. Une URL peut pointer vers un document HTML, une image, une vidéo ou tout autre type de fichier. Par exemple, l'URL "https://www.example.com" pointe vers la page d'accueil du site web example.com.
- **HTTP/HTTPS (HyperText Transfer Protocol / Secure)** : est le protocole de communication utilisé pour transférer des données sur le Web. HTTPS est une version sécurisée de HTTP qui chiffre les données pour garantir la confidentialité et l'intégrité des informations échangées. Les sites web utilisent HTTPS pour protéger les communications des utilisateurs contre les interceptions et les attaques.

- **Serveurs Web** :Les serveurs web sont des ordinateurs qui hébergent des sites web et les rendent accessibles sur Internet. Lorsqu'un utilisateur demande une page web en entrant une URL dans un navigateur, le serveur web correspondant renvoie le contenu de la page au navigateur.
- **Navigateur Web** :Les navigateurs web sont des logiciels qui permettent aux utilisateurs d'accéder au Web. Ils interprètent les documents HTML et affichent le contenu formaté selon les règles définies par les CSS et les scripts JavaScript. Les navigateurs populaires incluent Google Chrome, Mozilla Firefox, Microsoft Edge et Safari.
- **Moteurs de Recherche** : les moteurs de recherche, comme Google, Bing et Yahoo, permettent aux utilisateurs de trouver des informations sur le Web. Ils utilisent des algorithmes sophistiqués pour indexer et classer les pages web en fonction de leur pertinence par rapport aux requêtes des utilisateurs.
- **CMS (Content Management System)** : un CMS est un logiciel qui permet de créer, gérer et modifier facilement le contenu d'un site web sans avoir besoin de connaissances techniques avancées en programmation. Des exemples de CMS populaires incluent WordPress, Joomla et Drupal.
- **API Web (Application Programming Interface)** : les API Web permettent aux développeurs d'interagir avec des services web et d'intégrer des fonctionnalités externes dans leurs applications. Les API RESTful et les API GraphQL sont des types courants d'API utilisées sur le Web.
- **Responsive Design** :Le responsive design est une approche de conception web qui vise à créer des sites web qui offrent une expérience utilisateur optimale sur une large gamme d'appareils (ordinateurs de bureau, tablettes, smartphones). Il utilise des techniques comme les grilles fluides et les media queries CSS pour adapter la mise en page en fonction de la taille de l'écran. [3]

1.3. Différence entre web et Internet

La différence entre le Web et l'Internet est essentielle pour comprendre le fonctionnement du monde numérique.

1.3.1. Portée

- **Internet** : Un réseau de réseaux, englobant une gamme de services et d'applications, y compris mais non limité au Web.

- **Web** : Un système de documents hypertextes interconnectés accessibles via l'Internet.

1.3.2. Fonctionnalité

- **Internet** : Fournit l'infrastructure sous-jacente qui permet diverses activités en ligne et méthodes de communication.
- **Web** : Un service construit sur l'Internet, spécifiquement pour accéder et partager des informations à travers des pages web.

1.3.3. Composants

- **Internet** : Inclut le matériel (serveurs, routeurs, câbles) et les protocoles logiciels (TCP/IP).
- **Web** : Implique principalement les navigateurs web, les serveurs web et HTTP. [4]

1.4. Les technologies de web

1.4.1. Site web

Un site web est défini comme un objet numérique composé de pages web, de contenu multimédia et de fonctionnalités interactives, tous liés ensemble sous un même domaine, et qui change au fil du temps en fonction des mises à jour et des modifications . [5]

1.4.2. Applications Web

Les applications web sont des programmes accessibles via un navigateur web et fonctionnant sur un serveur web. Elles offrent des fonctionnalités interactives et peuvent être utilisées pour diverses tâches, comme la gestion de contenu, le commerce électronique, ou la communication.

- **Les types d'applications web** : Les applications web se déclinent en plusieurs types, chacune répondant à des besoins spécifiques en matière de contenu et d'interaction utilisateur. Cette section explore trois principaux types d'applications web : les applications web statiques, dynamiques et les portails web, en soulignant leurs caractéristiques distinctes et leurs usages courants.

- **Application web statique:** Les applications web statiques se composent de pages web dont le contenu ne change pas en réponse aux actions de l'utilisateur. Elles sont principalement construites en HTML et CSS, et peuvent inclure des images et des vidéos. [6]

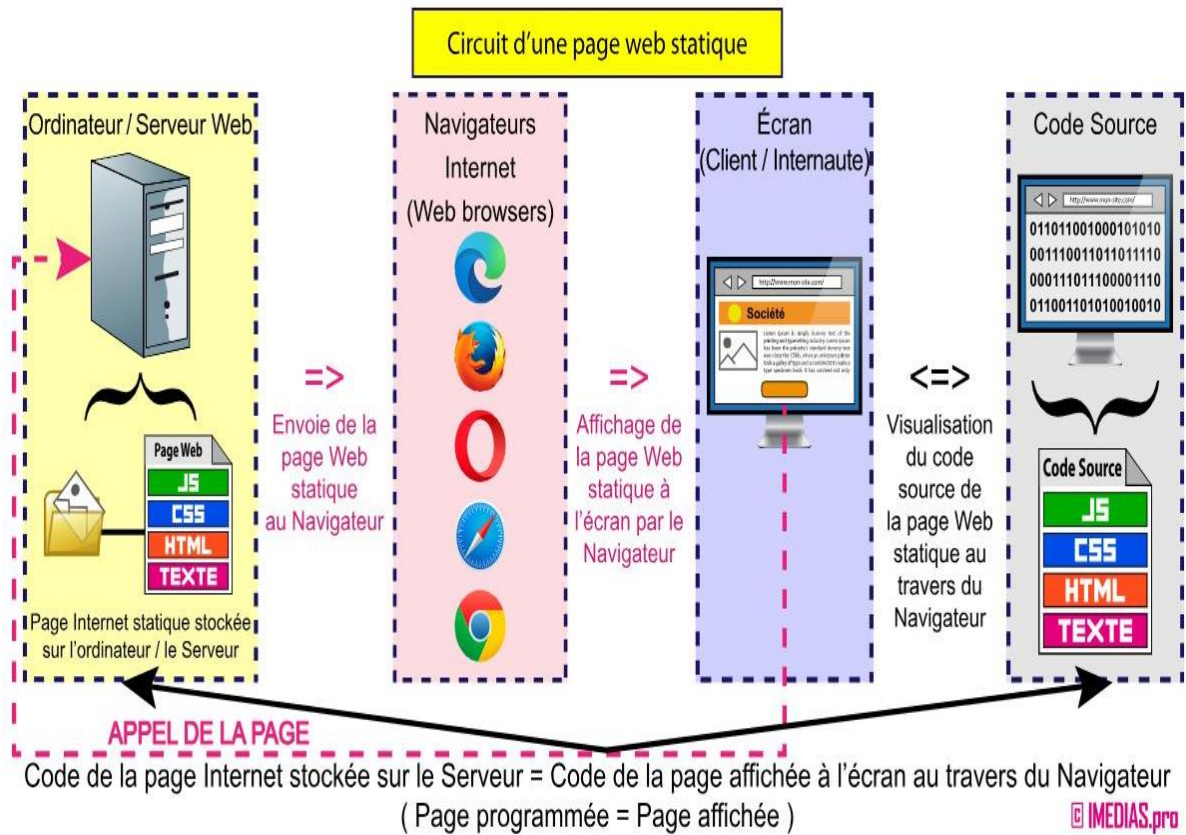


Figure 1. 1: Circuit d'une page web statique.

- **Application web dynamique** : les applications web dynamiques génèrent du contenu en réponse aux interactions des utilisateurs. Elles utilisent des technologies comme JavaScript, PHP, Python, ou Ruby, souvent en conjonction avec des bases de données pour stocker et récupérer des données. [7]

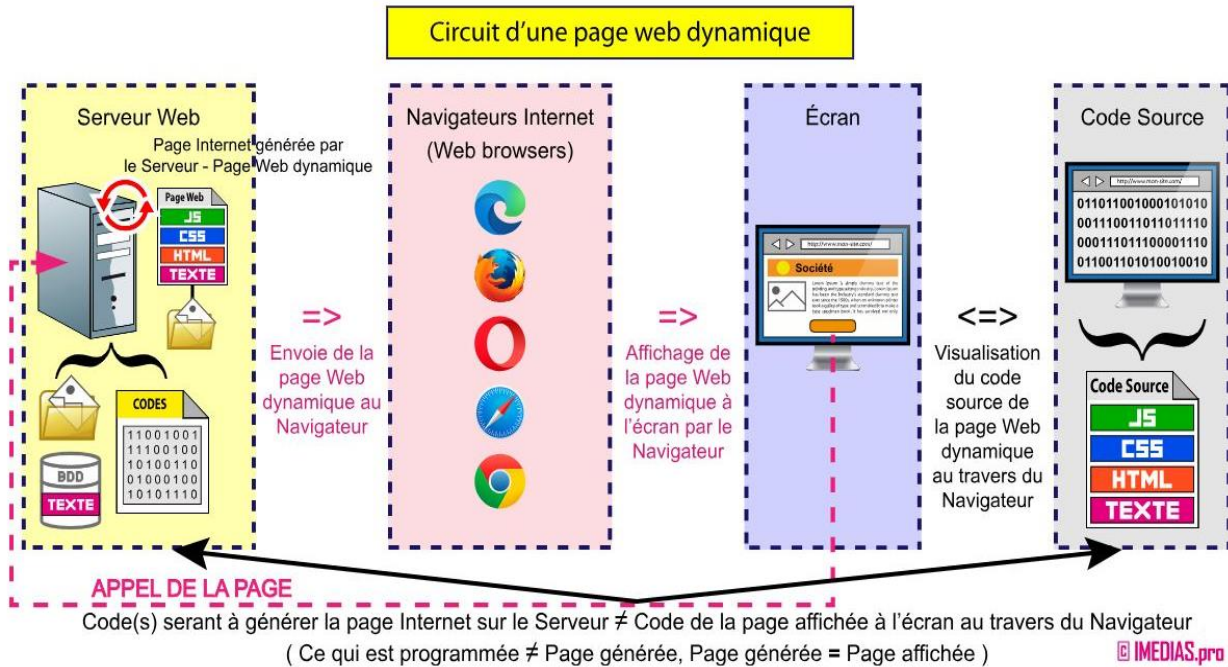


Figure 1. 2: Circuit d'une page web dynamique.

- **Application web portail** : est un environnement en ligne qui sert d'interface pour différents utilisateurs et propose des fonctionnalités telles que la recherche sémantique des ressources d'apprentissage, la notation des ressources par les utilisateurs, et un système de filtrage collaboratif pour recommander des ressources en fonction des préférences passées des utilisateurs. [5]
- **Application web avec gestionnaire de contenu** : est une solution logicielle qui simplifie la création, la gestion et la modification de contenu numérique sur un site web. Elle permet aux utilisateurs, même sans compétences techniques avancées, de gérer le contenu du site via une interface intuitive. Ces systèmes supportent divers types de contenu et offrent des fonctionnalités comme le stockage, la récupération, la publication et la gestion des flux de travail de contenu. [8]

1.4.3. Serveur web

Un serveur web est un logiciel qui répond aux requêtes HTTP/HTTPS des clients (navigateurs web) en fournissant les ressources demandées, comme des pages HTML, des images, ou des fichiers. Les serveurs web populaires incluent Apache, Nginx, et Microsoft IIS. [9]

1.5. Comparaison entre site web et application web

Un site web et une application web sont tous deux des interfaces utilisateur accessibles via un navigateur web, mais ils diffèrent principalement dans leur fonctionnalité, leur structure et parfois leur accessibilité.

1.5.1. Site Web

- Un site web est généralement constitué de pages web liées entre elles par des hyperliens.
- Il est généralement conçu pour être consulté sur divers appareils, y compris les ordinateurs de bureau, les tablettes et les smartphones.
- Les sites Web peuvent être statiques (avec un contenu fixe qui ne change pas fréquemment) ou dynamiques (avec un contenu généré dynamiquement à partir d'une base de données ou d'autres sources).
- Exemples : Blogs, sites d'actualités, sites de commerce électronique.

1.5.2. Application Web

- Une application Web est une application logicielle hébergée sur un serveur distant et accessible via un navigateur Web.
- Contrairement à un site web, une application web est généralement plus interactive et offre souvent des fonctionnalités avancées telles que des jeux, des outils de productivité, des réseaux sociaux, etc.
- Les applications web peuvent offrir des fonctionnalités hors ligne grâce à l'utilisation de technologies de stockage local.
- Elles peuvent également être conçues pour être responsives, c'est-à-dire qu'elles s'adaptent à différents appareils et tailles d'écran.
- Exemples : Google Docs, Trello, Gmail. [10]

1.6. Langages et les outils utilisé pour réaliser une application web

La création d'une application web nécessite l'utilisation de divers langages et outils spécialisés pour concevoir, développer et déployer des fonctionnalités efficaces et interactives. Cette section

présente les principaux langages de programmation et outils de développement utilisés dans le processus de création d'applications web.

1.6.1. Les langages

1.6.1.1- Langage de balisage/affichage

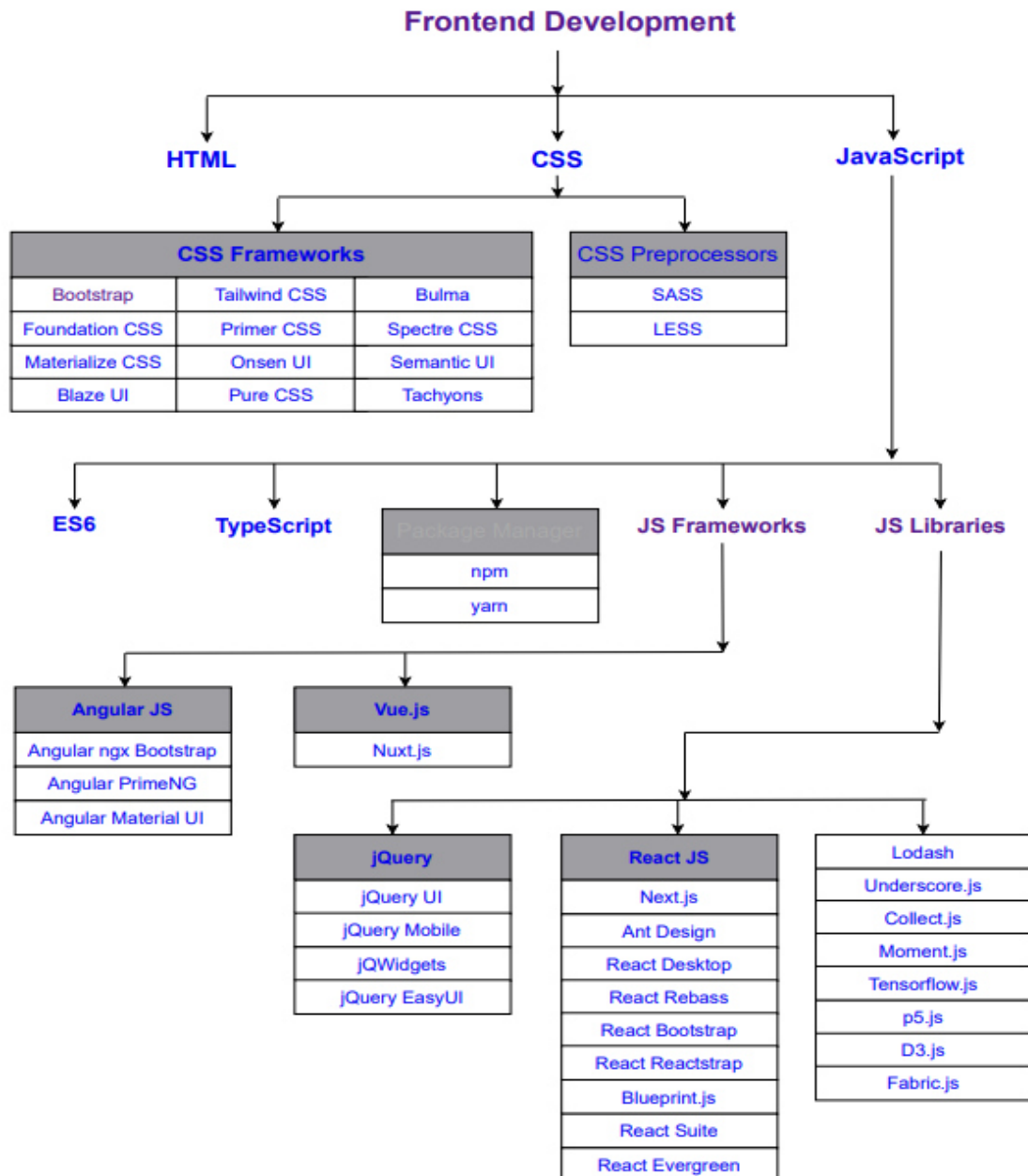


Figure 1. 3: Frontend.

- **HTML (HyperText Markup Language)** : Il s'agit du langage de balisage standard utilisé pour créer la structure et le contenu des pages web. Il permet de définir la hiérarchie des éléments et leur disposition sur la page.
- **CSS (Cascading Style Sheets)** : Utilisé pour définir la présentation et le style des éléments HTML. CSS permet de contrôler les aspects visuels tels que la couleur, la police, la taille et la mise en page des éléments HTML.

1.6.1.2- Langage client

JavaScript : langage de programmation côté client utilisé pour rendre les sites web interactifs. Il est interprété par le navigateur web et permet d'ajouter des fonctionnalités dynamiques aux pages web, telles que la manipulation du DOM (Document Object Model), la gestion des événements utilisateur et les requêtes AJAX.

1.6.1.3- Langage serveur

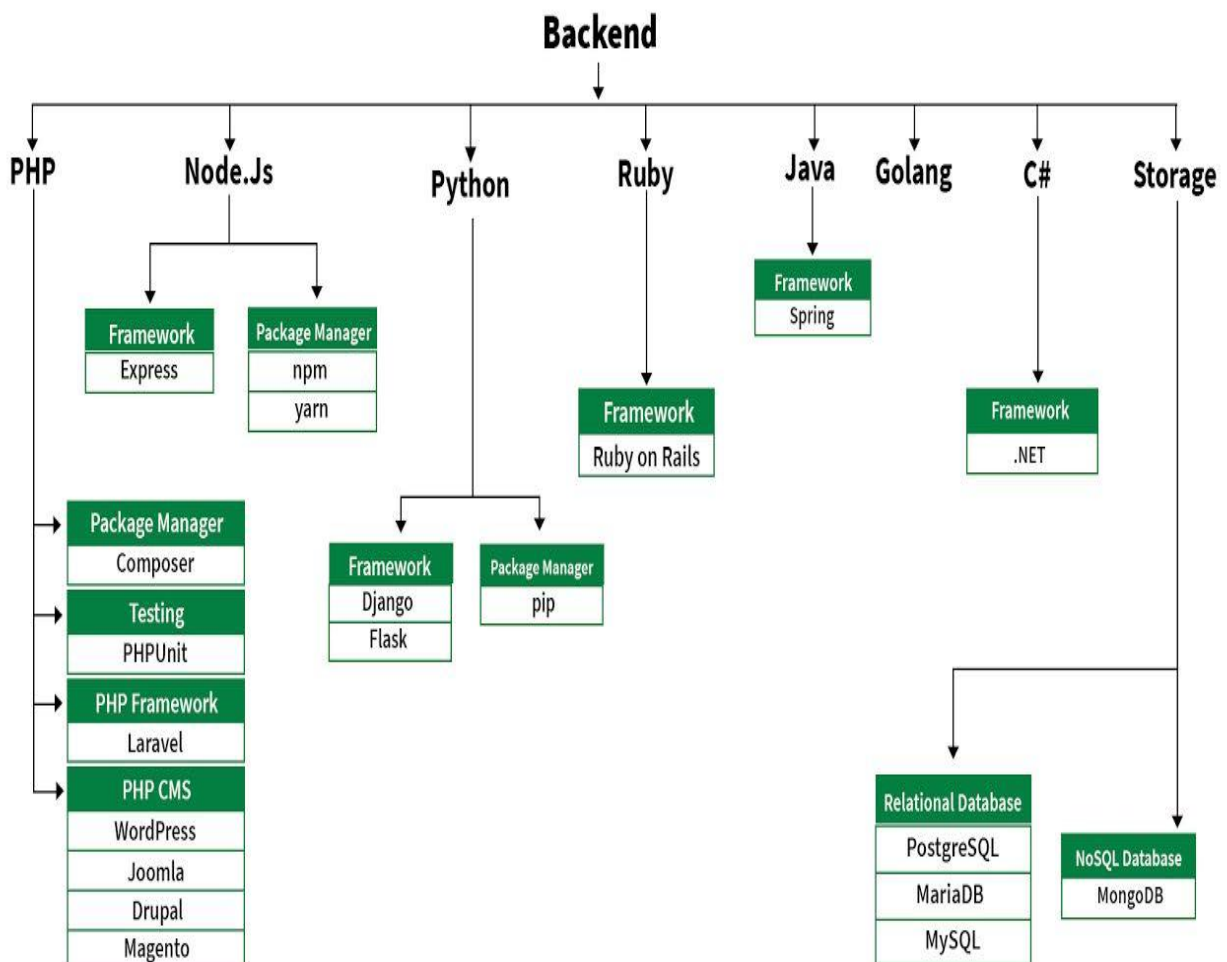


Figure 1. 4:Backend.

- **PHP** : Langage de script côté serveur largement utilisé pour le développement web. PHP est souvent intégré dans du code HTML et s'exécute sur le serveur, générant du contenu dynamique pour les pages web. Il est compatible avec de nombreux systèmes de gestion de bases de données, ce qui en fait un choix populaire pour les applications web dynamiques.
- **Node.js**: Environnement d'exécution JavaScript côté serveur basé sur le moteur JavaScript V8 de Google Chrome. Node.js permet d'écrire des applications serveur en JavaScript, offrant une approche unifiée du développement web avec JavaScript pour le côté client et le côté serveur.

1.6.1.4- Base de données

- **MySQL** : Système de gestion de base de données relationnelles (SGBDR) open-source, largement utilisé dans le développement web pour stocker et gérer les données des applications.
- **SQL Server** : SGBDR développé par Microsoft, offrant une solution professionnelle pour le stockage et la gestion de grandes bases de données dans des environnements web.
- **MongoDB** : Moteur de base de données NoSQL, orienté document, utilisé pour stocker des données non structurées ou semi-structurées. Il est souvent utilisé dans les applications web modernes pour sa flexibilité et sa scalabilité.
- **PostgreSQL** : SGBDR et objet relationnel, offrant des fonctionnalités avancées pour la gestion des données dans les applications web. Il est utilisé comme alternative à MySQL dans certains cas, offrant une performance et une sécurité élevées.

1.6.2. Les outils de développement

Les outils de développement sont essentiels pour créer, tester et déployer des applications web. Voici une description des principaux outils utilisés par les développeurs :

1.6.2. 1- Serveurs Web :

- Apache, Nginx, IIS : Des serveurs web populaires utilisés pour héberger des sites web et des applications web.
- Solutions clé en main : WampServer, XAMP, MAMP, Docker : Des solutions prêtes à l'emploi pour installer un environnement de développement web sur différentes plateformes (Windows, Linux, Mac OS X).

➤ Éditeurs de texte :

- Notepad++, Sublime Text 3, Atom, Visual Studio Code : Des éditeurs de texte populaires offrant des fonctionnalités de développement basiques telles que la coloration syntaxique, l'autocomplétion et la gestion de projets.

➤ Gestion de code :

- Git : Un système de contrôle de version populaire utilisé pour suivre les modifications du code source et collaborer avec d'autres développeurs sur des projets web.

➤ Débogage et Tests :

- XDebug : Un outil de débogage puissant pour PHP, permettant de mettre des points d'arrêt dans le code et d'inspecter les variables pendant l'exécution.
- PHPUnit : Un framework de test unitaire pour PHP, permettant de tester le code de manière automatisée pour garantir sa qualité.
- MailHog, MailDev : Des outils pour le test d'envoi d'e-mails dans des environnements de développement, permettant de capturer et de visualiser les e-mails envoyés par une application web.

➤ Gestion des dépendances :

- Composer : Un gestionnaire de dépendances pour PHP, permettant d'installer et de gérer les bibliothèques et les packages nécessaires à un projet web.
- IDE : Un IDE (Integrated Development Environment) est un logiciel qui regroupe plusieurs outils nécessaires au développement de logiciels. L'avantage principal des IDE réside dans le fait qu'ils intègrent une multitude d'outils visant à améliorer la productivité des développeurs, le tout au sein d'une interface unique

Voici une liste des IDE (Environnements de Développement Intégré) les plus populaires, classés par langage de programmation et usage spécifique :

- IDEs Multilinguages

➤ Visual Studio Code (VS Code)

- Développeur : Microsoft
- Caractéristiques : Léger, extensible via des extensions, support intégré de Git, terminal intégré.
- Langages : JavaScript, Python, C++, Java, PHP, et bien d'autres.

➤ **Eclipse**

- Développeur : Eclipse Foundation
- Caractéristiques : Extensible avec des plugins, supporte de nombreux langages de programmation.
- Langages : Java, C++, PHP, Python, et bien d'autres.

➤ **IntelliJ IDEA**

- Développeur : JetBrains
- Caractéristiques : Très puissant pour le développement Java, support pour d'autres langages avec des plugins.
- Langages : Java, Kotlin, Groovy, Scala, et d'autres.

➤ **NetBeans**

- Développeur : Apache Software Foundation
- Caractéristiques : Support pour le développement d'applications Java SE, Java EE, PHP, HTML5.
- Langages : Java, PHP, JavaScript, HTML5, et d'autres.

➤ **Atom**

- Développeur : GitHub
- Caractéristiques : Éditeur de texte hackable pour le 21ème siècle, extensible via des packages.
- Langages : Multilingue (via packages).

➤ **Sublime Text**

- Développeur : Sublime HQ
- Caractéristiques : Éditeur de texte sophistiqué pour le code, le balisage et la prose.
- Langages : Multilingue (via packages). [11]
- Frameworks : Les frameworks sont des structures de développement de logiciels fournissant une base pour les développeurs afin de créer des applications plus facilement. Voici une liste des frameworks les plus populaires. [12]

➤ **Frameworks Web**

- **JavaScript:**
 - React : Bibliothèque pour construire des interfaces utilisateur.

- Angular : Framework développé par Google pour la création de SPAs.
- Vue.js : Framework progressif pour la construction d'interfaces utilisateur.
- Node.js : Environnement d'exécution pour JavaScript côté serveur.
- Next.js : Framework pour les applications React avec rendu côté serveur et génération statique.
- Nuxt.js : Framework basé sur Vue.js avec rendu côté serveur. [11]
- **PHP:**
 - Laravel : Framework PHP populaire pour le développement web avec MVC.
 - Symfony : Framework PHP pour des applications web robustes.
 - CodeIgniter : Framework léger pour le développement rapide en PHP.
 - Zend Framework : Framework PHP orienté objet pour applications web. [12]
- **Python:**
 - Django : Framework web de haut niveau pour un développement rapide et propre.
 - Flask : Microframework léger pour les applications web.
 - Pyramid : Framework flexible et open-source pour Python.
 - Bottle : Microframework simple pour de petites applications web.
- **Ruby:**
 - Ruby on Rails : Framework web full-stack pour des applications web rapides.
 - Sinatra : Microframework minimaliste pour Ruby.
- **Java:**
 - Spring : Framework complet pour les applications Java, incluant Spring Boot pour le développement rapide.
 - JSF (JavaServer Faces) : Framework pour la construction d'interfaces utilisateur web.
 - Struts : Framework web pour des applications Java basées sur MVC.
 - ASP.NET Core : Framework open-source pour la construction d'applications modernes basées sur le cloud et internet.
- **TypeScript**
 - NestJS : Framework progressif pour la construction de serveurs Node.js efficaces, fiables et évolutifs. [11]

1.7. Comparaison entre les Framework les plus utilisés

Dans le tableau suivant, nous avons fait une comparaison entre les Framework les plus utilisés pour choisir le meilleur framework pour créer notre application.

Framework	Langage	Fonctionnalités clés	Avantages	Inconvénients
Django	Python	Principes DRY, ORM, Interface d'administration	Développement rapide, Sécurité robuste	Courbe d'apprentissage plus rapide
Laravel	PHP	Eloquent ORM, Blade templating, CLI Artisan	Syntaxe élégante, Communauté active	Peut être complexe pour les débutants
Spring Boot	Java	Injection de dépendances, Services RESTful, Sécurité	Scalabilité élevée, Documentation étendue	Connaissance nécessaire en Java
Ruby on Rails	Ruby	Convention over configuration, Active Record ORM	Développement rapide, Bon pour les startups	Moins performant pour les applications à grande échelle
ASP.NET Core	C#	Support multiplateforme, Blazor pour les UIs, Entity Framework Core	Performance élevée, Bon pour les applications d'entreprise	Communauté moins étendue que d'autres

Tableau 1. 1: tableau comparatif entre les Framework

Après avoir comparé différents Framework, nous avons choisi Laravel. Écrit en PHP, Laravel se distingue par sa syntaxe élégante et sa communauté active. Ses fonctionnalités clés, telles que l'Eloquent ORM, le templating Blade et l'interface en ligne de commande Artisan, facilitent et accélèrent le développement. Bien que complexe pour les débutants, Laravel offre des outils robustes pour la gestion des bases de données, la sécurité et l'optimisation des performances, faisant de lui un choix idéal pour notre application.

1.8. Conclusion

En conclusion, l'Internet et le Web sont des éléments centraux de notre monde numérique, chacun ayant des rôles et des fonctionnalités distinctes mais complémentaires.

Tandis que l'Internet constitue la base physique et logicielle permettant une variété de services de communication et d'information, le web offre un cadre structuré pour l'accès et la navigation parmi ces informations via des pages hypertextes.

La distinction entre sites web et applications Web repose principalement sur leur interactivité et leurs fonctionnalités : les sites web fournissent souvent du contenu statique ou dynamique destiné à l'information, tandis que les applications web offrent des fonctionnalités interactives et complexes, allant des outils de productivité aux plateformes de communication.

La création et la gestion de ces outils reposent sur une diversité de langages de programmation et d'outils de développement sophistiqués, chacun contribuant à l'enrichissement de notre expérience numérique quotidienne

CHAPITRE II :

Présentation de l'organisme d'accueil

2.1. Introduction

Ce chapitre nous permettra d'avoir une vue générale sur notre domaine d'application, à savoir la gestion des activités de la Ligue de Natation de Béjaïa. Nous utiliserons la structure globale de la ligue pour établir les objectifs de notre application.

2.2. Natation

2.2.1. Définition

La natation est une activité sportive et récréative qui consiste à se déplacer dans l'eau en utilisant diverses techniques de propulsion. Elle peut être pratiquée à des fins de loisir, de compétition, de remise en forme ou de sauvetage.

2.2.2. Histoire de la Natation

La natation a une longue histoire qui remonte à l'Antiquité. Des bas-reliefs égyptiens datant de 2500 av. J.-C. montrent déjà des nageurs. Au fil du temps, la natation est devenue un sport structuré avec des compétitions organisées, notamment les Jeux Olympiques modernes où elle est présente depuis 1896.

2.2.3. Types de Natation

a. Natation de Compétition

- Nage Libre : Aussi appelée crawl, c'est la nage la plus rapide.
- Dos : Nage sur le dos, offrant une vue vers le ciel.
- Brasse : Nage qui nécessite une coordination précise des bras et des jambes.
- Papillon : Considérée comme la nage la plus technique et exigeante.
- Natation Synchronisée
- Combinaison de Natation, Danse et Gymnastique : Réalisée en équipe avec des routines chorégraphiées.
- Compétitions et Spectacles : Jugées sur la synchronisation, la difficulté et la présentation artistique.

b. Water-Polo

- **Sport d'Équipe** : Combinaison de natation, football et basketball dans l'eau.

- **Stratégie et Endurance** : Nécessite des compétences en natation, en tactique et une grande endurance.

2.2.4. Plongée

- Plongée de Haut Vol et Acrobatie : Exécutions de sauts et figures complexes à partir de tremplins ou plateformes.

2.3. Ligue sportive

2.3.1. Structure et Fonctionnement des Ligues Sportives

La ligue sportive est une association régie par les dispositions de la loi sur les associations, celles de la présente loi ainsi que par ses statuts et les statuts de la fédération sportive nationale à laquelle elle est affiliée. Elle peut être constituée sur proposition de la ou des fédérations sportives nationales concernées et après avis conforme de l'administration chargée des sports.

La ligue sportive peut être :

- selon la nature de ses activités, une ligue sportive omnisports ou spécialisée,
- selon l'importance de ses missions et de sa compétence territoriale, une ligue sportive nationale, régionale, de wilaya ou communale.

La ligue sportive regroupe en son sein les clubs sportifs et, le cas échéant, les ligues dûment constituées et qui lui sont affiliées conformément à ses statuts. Elle assure la coordination des clubs et ligues sportives qui lui sont affiliés.

2.3.2. Présentation de la Ligue de Natation de Béjaïa

La Ligue de Natation de Béjaïa est une organisation dédiée à la promotion et au développement de la natation dans la région de Béjaïa. Elle vise à encourager la pratique de la natation, organiser des compétitions locales et régionales, et détecter de jeunes talents pour les former et les intégrer aux équipes nationales.

- **Structure et Organisation** : La ligue est structurée pour assurer une gestion efficace de ses activités. Elle comprend des membres élus qui occupent différents postes clés, tels que le président, le vice-président, le secrétaire général, le trésorier et le directeur technique. Chacun de ces postes a des responsabilités spécifiques pour garantir le bon fonctionnement de la ligue.
- **Activités et Compétitions** : La ligue organise régulièrement des compétitions pour diverses catégories d'âge et niveaux de compétence. Ces compétitions servent à évaluer les performances des nageurs, à les préparer pour des compétitions nationales et internationales, et à promouvoir un esprit de compétition saine et de fair-play.

Organiser une compétition par la ligue de natation nécessite plusieurs étapes :

➤ **Détermination du Calendrier et du Lieu**

La Ligue de Natation fixe d'abord la date et le lieu de la compétition. Ensuite, elle établit l'horaire des sessions, la durée limite pour chaque épreuve, ainsi que les catégories d'âge et de genre concernées.

➤ **Annnonce et Préparatifs**

Une fois les préparatifs terminés, la ligue publie l'annonce de la compétition aux clubs et les informe de la possibilité d'y participer.

➤ **Sélection des Participants**

Les clubs sélectionnent les nageurs qu'ils souhaitent inscrire à la compétition. Chaque club soumet les noms des participants, les épreuves choisies et les temps d'engagement à la ligue.

➤ **Filtrage et Répartition**

La ligue reçoit les inscriptions, filtre les nageurs par épreuve et organise les séries. Chaque série comprend huit nageurs, répartis selon leurs temps d'engagement : le meilleur temps est assigné au couloir 4, le suivant au couloir 5, puis 3, 6, 2, 7, 1, et enfin 8.

➤ **Création de la Start List**

Après répartition, une liste de départ (start list) est générée, incluant tous les nageurs classés selon l'ordre mentionné, ainsi que les cartes de participation.

➤ **Déroulement de la Compétition**

La compétition commence et les arbitres enregistrent les résultats, qui sont ensuite envoyés à la ligue. La ligue compile les résultats et les communique aux clubs.

❖ **Formation et Développement** : Un des rôles principaux de la ligue est la formation des jeunes talents. Elle travaille en étroite collaboration avec des clubs locaux, des écoles et des centres de formation pour offrir des programmes de formation technique, méthodologique et médico-sportif adaptés aux besoins des jeunes nageurs.

➤ **Partenariats et Collaborations** : La ligue de natation de Béjaia collabore avec diverses organisations sportives, fédérations nationales, et associations scientifiques pour améliorer constamment ses méthodes et offrir les meilleures opportunités de développement à ses athlètes.

2.3.3. Structure Organisationnelle de la Ligue

A. Président : le Président est le chef de l'organisation, responsable de la supervision globale et de la direction stratégique de l'association ou de la ligue.

- Représenter l'organisation dans toutes les instances.

Chapitre II : Présentation de l'organisme d'accueil

- Superviser les activités de l'organisation.
 - Présider les réunions et les assemblées générales.
 - Assurer la mise en œuvre des décisions prises par les organes de gouvernance.
- B. Vice-président :** le vice-président assiste le président et le remplace en cas d'absence ou d'empêchement.
- Assister le Président dans ses fonctions.
 - Remplacer le Président en cas d'absence.
 - Coordonner certaines activités spécifiques selon les délégations reçues du Président.
- C. Secrétaire générale :** le secrétaire général est responsable de la gestion administrative de l'organisation.
- Gérer les correspondances et documents officiels.
 - Organiser et préparer les réunions et les assemblées générales.
 - Rédiger les procès-verbaux des réunions.
 - Maintenir les archives de l'organisation.
- D. Trésorier :** le trésorier est responsable de la gestion financière de l'organisation.
- Gérer les comptes et les finances.
 - Préparer les budgets et les rapports financiers.
 - Assurer la transparence et la conformité financière.
 - Collecter les cotisations et autres revenus.
- E. DJTS :** la direction méthodologique chargée des jeunes talents sportifs identifier et sélectionner les jeunes talents sportifs en définissant des critères spécifiques.
- Définir les critères de détection et sélection des jeunes talents.
 - Assurer le suivi des jeunes talents dans les centres de formation.
 - Organiser des compétitions pour l'émergence de nouveaux talents.
 - Diffuser des guides techniques.
 - Collaborer pour la prospection et sélection des talents nationaux.
- ❖ **Peut comprendre**
- Un Bureau des Études et de l'Orientation Sportive.
 - Un Bureau de Suivi et de l'Évaluation.
- F. DDF :** la direction méthodologique chargée du développement sportif et de la formation mettre en œuvre des programmes pour développer le sport, encourager la pratique sportive généralisée.
- Mettre en œuvre les programmes de développement sportif.

Chapitre II : Présentation de l'organisme d'accueil

- Généraliser la pratique sportive.
- Organiser des séminaires pour homogénéiser les méthodes de travail.
- Mettre à jour les fichiers techniques.

❖ Peut comprendre

- Un Bureau des Méthodes, Programmes et du Contrôle.
- Un Bureau du Suivi du Développement en Milieu Scolaire, Universitaire et de Proximité

G. DOSC : la direction méthodologique chargée de l'organisation sportive organiser et superviser les événements sportifs, assurer le respect du calendrier sportif établi.

- Organiser et contrôler les manifestations sportives.
- Assurer l'exécution du calendrier sportif.
- Homologuer les infrastructures sportives.
- Promouvoir l'éthique et le fair-play.

❖ Peut comprendre

- Un Bureau de la Programmation et du Suivi.
- Un Bureau des Homologations et du Contrôle.

H. DTW : le Directeur Technique de Wilaya est chargé de superviser les aspects techniques et sportifs au niveau régional (wilaya).

- Coordonner les activités techniques et sportives au niveau de la wilaya.
- Encadrer les entraîneurs et les équipes locales.
- Assurer la mise en œuvre des programmes de formation.
- Suivre les performances des clubs et des athlètes dans la wilaya. [13]

2.3.4. Organigramme de la Ligue

Pour montrer la structure de la direction, l'organigramme de la ligue de natation illustre la hiérarchie et les différentes responsabilités au sein de l'organisation

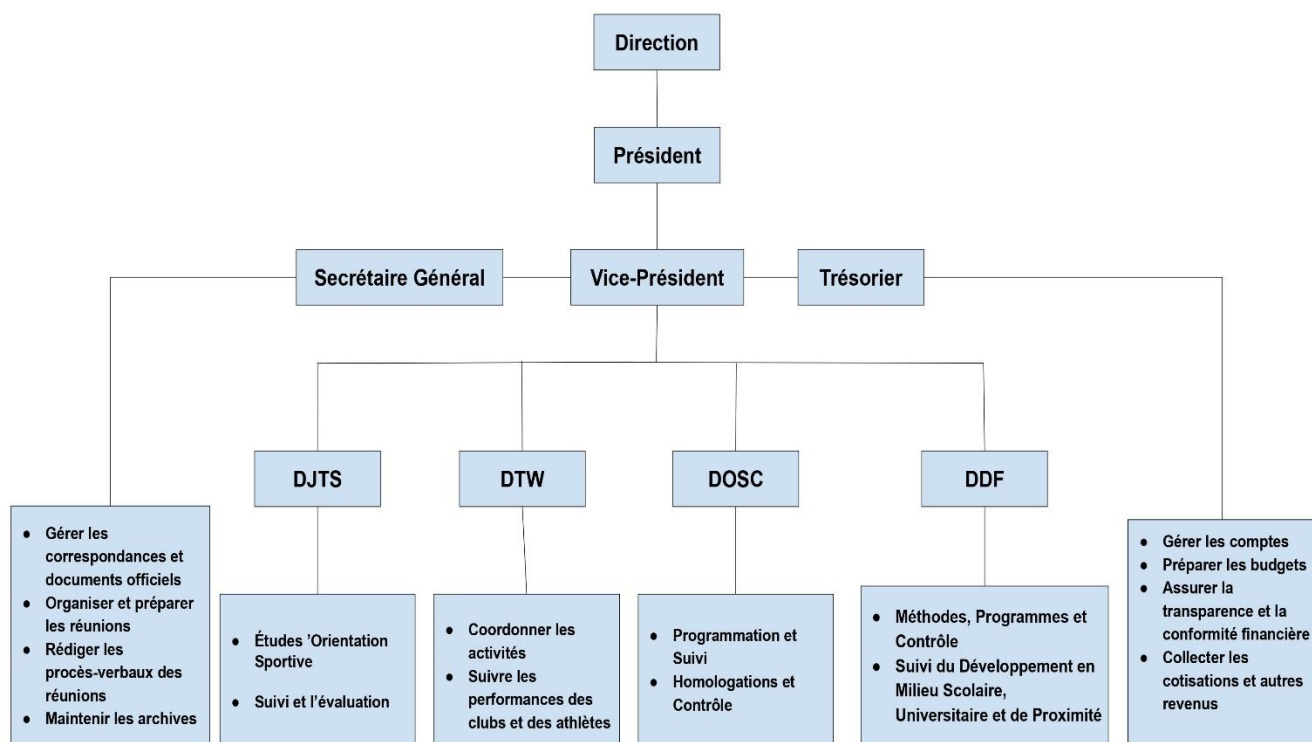


Figure 2. 1: Organigramme de la ligue de natation de Bejaia.

2.4. Problématique et Objectifs

2.4.1. Problématique

Le processus actuel de gestion des compétitions au sein de la ligue de natation repose sur de nombreuses transactions papier, ce qui peut entraîner une fatigue administrative considérable et des risques d'erreurs. La gestion manuelle des inscriptions, des résultats, ainsi que la gestion des athlètes, des clubs et des présidents, et la communication avec les clubs, est laborieuse et inefficace.

2.4.2. Objectifs

Pour améliorer l'efficacité de la gestion et simplifier l'administration des compétitions, nous avons décidé de développer une plateforme en ligne. Ce site web permettra d'automatiser les transactions et de rendre la gestion des compétitions plus fluide. Les clubs pourront inscrire leurs nageurs en ligne, les résultats seront automatiquement enregistrés et publiés, et la communication

entre la ligue et les clubs sera nettement améliorée. Cette solution vise à réduire la charge de travail manuel, à minimiser les erreurs et à faciliter l'accès aux informations pour toutes les parties prenantes.

2.5. Conclusion

La Ligue de Natation de Béjaïa se distingue non seulement par son engagement envers le sport mais aussi par son rôle dynamique dans le développement personnel et athlétique des jeunes au sein de la communauté. À travers ses initiatives diversifiées et son leadership organisé, elle continue de nourrir la passion pour la natation et d'encourager l'excellence sportive dans la région.

CHAPITRE III :

Analyse et Conception

3.1. Introduction :

Nous présentons dans ce chapitre « analyse et conception » notre travail qui consiste à développer une application web pour la gestion de la ligue de natation de Béjaïa.

Nous commencerons par présenter les fonctionnalités de notre projet, puis nous détaillerons la modélisation orientée objet à l'aide du langage UML (Unified Modeling Language). Enfin, nous terminerons par la conception de la base de données nécessaire à notre application.

3.2. Cahier de charge

L'application vise à moderniser et à faciliter la gestion de la ligue de natation de Béjaïa en automatisant les processus administratifs, en améliorant la communication entre les équipes et la Ligue, et en fournissant des outils avancés pour la gestion des tournois et des athlètes et des compétitions.

Créer des compétitions entre les équipes, réduire la paperasse en extrayant automatiquement la start liste, et permettre aux équipes de gérer leurs propres athlètes et de les inscrire aux compétitions créés par la Ligue. Cela facilite le processus entre les équipes et la Ligue. Enfin, l'envoi des résultats des joueurs aux équipes est automatisé.

3.2.1. Les besoins fonctionnels

Ce système inclut diverses fonctionnalités adaptées à la gestion d'une ligue de natation.

Il répond aux besoins en matière de comportement d'entrée et de sortie. Il doit permettre à l'administrateur de s'authentifier pour accéder à ses interfaces spécifiques et gérer l'espace des utilisateurs. De plus, le système doit permettre à l'administrateur de la ligue de natation de gérer les compétitions, les inscriptions des nageurs, les résultats et les classements, avec les opérateurs suivants :

- Gérer le profil (mettre à jour) .
- Gérer les clubs (créer, modifier, supprimer).

- Gérer les présidents (créer, modifier, supprimer).
- Gérer les athlètes (accepter, refuser) .
- Gérer les compétitions (créer, modifier, supprimer).

Le système doit également permettre à l'administrateur de club de gérer les athlètes, avec les opérateurs suivants :

- Gérer le profil (mettre à jour) .
- Gérer les athlètes (ajouter, inscrire).

3.2.2. Les besoins non fonctionnels

Il s'agit des besoins qui caractérisent le système, incluant des exigences de performance, de matériel, et de conception. Ces besoins peuvent également concerner les contraintes d'implémentation, telles que le langage de programmation, le type de SGBD, et le système d'exploitation. Parmi les besoins non fonctionnels, nous pouvons citer :

- **Extensibilité** : Capacité d'ajouter ou de modifier des fonctionnalités.
- **Sécurité des données** : Toutes les informations doivent être sécurisées et accessibles uniquement aux utilisateurs autorisés.
- **Compatibilité** : Fonctionnement avec n'importe quel système d'exploitation.
- **Auditabilité** : Vérification des éléments et des champs.
- **Performance** : Temps de réponse rapide, chargement de l'application, ouverture des écrans et rafraîchissement rapide. Cela inclut le traitement des fonctions.

3.3. Analyse

3.3.1. Méthode d'analyse et de conception

Avant de développer une application, il est important de suivre une méthode bien définie. Pour assurer une réalisation précise et organisée de notre projet, nous devons passer par différentes étapes. Pour cela, nous avons choisi d'utiliser le langage UML.

3.3.1. 1- Présentation d'UML

- **Définition UML** : L'UML (Unified Modeling Language) est un langage standardisé utilisé pour visualiser, spécifier, construire et documenter des systèmes logiciels. Il permet de représenter graphiquement différents aspects d'un logiciel, facilitant ainsi la compréhension et la communication entre les membres d'une équipe de développement. UML est géré par

l'Object Management Group (OMG) et comprend des outils pour modéliser des classes, des composants, des activités, des cas d'utilisation, et bien plus encore, afin de soutenir la conception et le développement de logiciels de manière cohérente et extensible.

En résumé, l'UML est :

- **Exact** : Décrivant précisément et correctement le système qu'il représente.
- **Compréhensible** : Aussi simple que possible tout en étant précis.
- **Consistant** : Différentes vues ne doivent pas se contredire.
- **Modifiable** : Facile à changer et à mettre à jour. [14]

➤ Quelques définitions de base

- **Acteur** : un acteur est un ensemble cohérent de rôles que les utilisateurs des cas d'utilisation jouent lorsqu'ils interagissent avec ces cas d'utilisation.
- **Classe** : une classe est une description d'un ensemble d'objets qui partagent les mêmes attributs, opérations, relations et sémantiques.
- **Diagramme** : un diagramme est une représentation graphique d'un ensemble d'éléments, le plus souvent rendu sous forme de graphe connecté de sommets (éléments) et d'arcs (relations).
- **Scénario** : un scénario est une séquence spécifique d'actions qui illustre un comportement. [15]

3.3.2. Démarche

3.3.2 - 1. Identification des acteurs :

- **Admin (ligue)** : Dans notre cas, l'admin de la ligue de natation est responsable de la gestion des compétitions et des nageurs. Il possède les droits d'accès complets à l'espace utilisateur et gère l'ensemble de l'application.
- **Agent (club)** : L'agent de club est responsable de la gestion des membres et des activités au sein de son club de natation. Il possède les droits d'accès nécessaires pour gérer les athlètes et les compétitions au niveau du club.

3.3.2 – 2. Identification des messages échangés entre les acteurs et le système :

- **Admin** : Le tableau suivant décrit les différents rôles et interactions de l'administrateur au sein du système de gestion de la ligue de natation. Chaque section détaille les actions principales de l'admin, les messages entrants qu'il reçoit et les messages sortants qu'il envoie

Chapitre III : Analyse et Conception

Acteur		Messages entrants	Messages sortants
Admin	Authentification	<p>M1 : Accéder à login M3 : Saisir l'email ou nom et le mot de passe.</p>	<p>M2 : Demande de s'authentifier M4 : Vérification. M5 : L'affichage du Dashboard.</p>
	Creation des presidents	<p>M1 : Accéder a l'espace président M3 : Demande d'ajouter un président M5 : Saisir les information de président et sauvegarder</p>	<p>M2 : L'affichage de l'espace président M4 : L'affichage du formulaire pour ajouter président M6 : L'affichage de l'espace président avec tableau des présidents.</p>
	Creation des clubs	<p>M1 : Accéder a l'espace club M3 : Demande d'ajouter un club M5 : Saisir les information du club et choisir président et sauvegarder</p>	<p>M2 : L'affichage de l'espace club M4 : L'affichage du formulaire pour ajouter club M6 : L'affichage de l'espace club avec tableau des clubs .</p>
	Creation des Epreuves	<p>M1 : Accéder a l'espace épreuve M3 : Demande d'ajouter un épreuve M5 : Saisir les information d'épreuve et sauvegarder</p>	<p>M2 : L'affichage de l'espace épreuve M4 : L'affichage du formulaire pour ajouter épreuve M6 : L'affichage de l'espace épreuve avec tableau des épreuves.</p>

Chapitre III : Analyse et Conception

	Creation des competitions	<p>M1 : Accéder a l'espace compétition</p> <p>M3 : Demande d'ajouter un compétition</p> <p>M5 : Saisir les information du compétition avec le nombre de session et sauvegarder</p> <p>M7 : Accéder a l'espace session</p> <p>M9 : choisir un session</p> <p>M11 : Saisir les information du session et choisir l'épreuve et sauvegarder</p> <p>M13 : Accéder a l'espace compétition</p>	<p>M2 : L'affichage de l'espace compétition</p> <p>M4 : L'affichage du formulaire pour ajouter compétition</p> <p>M6 : L'affichage de l'espace compétition avec tableau des competitions.</p> <p>M8 : L'affichage de l'espace session</p> <p>M10 : L'affichage du formulaire pour modifier un session</p> <p>M12 : L'affichage de l'espace session avec tableau des sessions</p> <p>M14 : L'affichage du tableau de compétition avec les sessions et les epreuve pour chaque compétition</p>
	Acceptation ou refusant des athletes dans le ligue et compétition	<p>M1 : Accéder a l'espace demande inscription</p> <p>M3 : accept ou refuse athlète</p>	<p>M2 : L'affichage de l'espace demande inscription</p>
	Start liste	<p>M1 : Accéder a l'espace athlète compétition</p> <p>M3 : Demande d'imprimer la liste des athlète</p> <p>M5 : Demande d'accéder au start liste</p> <p>M7 : Demande d'imprimer la start liste</p>	<p>M2 : L'affichage de l'espace athlète compétition</p> <p>M4 : L'affichage de liste des athletes (PDF)</p> <p>M6 : L'affichage du start liste</p> <p>M8 : L'affichage du start liste (PDF)</p>

Tableau 3. 1: Identification des messages échangés(Admin)

- **Agent** : Le tableau suivant décrit les différents rôles et interactions de l'agent au sein du système de gestion de la ligue de natation. Chaque section détaille les actions principales de l'agent, les messages entrants qu'il reçoit et les messages sortants qu'il envoie

Acteur		Messages entrants	Messages sortants
Agent	Authentification	M1 : Accéder à login M3 : Saisir l'email ou nom et le mot de passe.	M2 : Demande de s'authentifier M4 : Vérification. M5 : L'affichage du Dashboard.
	Creation des athletes	M1 : Accéder a l'espace athlète M3 : Demande d'ajouter un athlète M5 : Saisir les information de l'athlète et sauvegarder M7 : Attendant l'admin pour accept ou refuse l'athlète	M2 : L'affichage de l'espace athlète M4 : L'affichage du formulaire pour ajouter athlète M6 : L'affichage de l'athlète dans l'espace athlète en attente. M8 : L'affichage de l'athlète dans l'espace accept ou refuse
	Inscrit des athletes dans compétition	M1 : Accéder a l'espace compétition M3 : Demande d'inscrit un athlète M5 : choisir compétition et l'athlète et l'preuve et sauvegarder M7 : Attendant l'admin pour accept ou refuse l'athlète	M2 : L'affichage de l'espace compétition M4 : L'affichage du formulaire pour inscrit athlète M6 : L'affichage de l'athlète dans l'espace en attente. M8 : L'affichage de l'athlète dans l'espace accept ou refuse

Tableau 3. 2: Identification des messages échangés(Agent).

3.3.2 - 3. Identification des cas d'utilisation :

- **Définition d'un cas d'utilisation** : un cas d'utilisation est une description détaillée d'un ensemble de séquences d'actions, incluant des variantes, qu'un système exécute pour atteindre un objectif spécifique. Chaque séquence décrit une interaction entre un acteur (utilisateur ou autre système) et le système, visant à produire un résultat observable et de valeur pour l'acteur.

Un cas d'utilisation fournit une vue complète des différents scénarios possibles, y compris les chemins principaux et alternatifs, et définit clairement les interactions nécessaires pour atteindre l'objectif.

3.3.2 -4. Diagramme de cas d'utilisation:

- **Définition :** Un diagramme de cas d'utilisation est une représentation graphique qui montre un ensemble de cas d'utilisation et les acteurs impliqués ainsi que leurs relations. Il fournit une vue statique des interactions possibles dans le système et sert de point de départ pour la modélisation des fonctionnalités du système.

Les éléments clés d'un diagramme de cas d'utilisation comprennent :

- **Acteurs :** Représentés par des figures en bâton, ils indiquent les rôles extérieurs qui interagissent avec le système.
- **Cas d'utilisation :** Représentés par des ovales, ils montrent les différentes fonctions ou services que le système offre aux acteurs.
- **Système :** Délimité par un rectangle qui englobe tous les cas d'utilisation pertinents.
- **Relations :** Des lignes qui connectent les acteurs aux cas d'utilisation, indiquant les interactions. Ces relations peuvent inclure des associations simples, des inclusions (<<include>>), des extensions (<<extend>>), et des généralisations. [15]
- **Diagramme de cas d'utilisation global pour l'administration :** les diagrammes ci-dessus nous montre une vue globale de l'application :

▪ **Partie Admin :**

Le diagramme de cas d'utilisation pour l'admin présente les diverses fonctions que ce rôle peut accomplir dans le système de gestion de la ligue de natation de Béjaïa. L'administrateur a la capacité de gérer les athlètes, les clubs, les présidents et les compétitions. Dans la gestion des athlètes, l'administrateur peut accepter ou refuser des athlètes, ainsi qu'ajouter, modifier ou supprimer leurs informations. La gestion des clubs permet à l'administrateur d'ajouter, modifier ou supprimer des clubs, tout en gérant les présidents associés à chaque club. En ce qui concerne la gestion des présidents, l'administrateur peut effectuer des actions d'ajout, de modification et de suppression des présidents des clubs. Enfin, l'administrateur peut créer des compétitions, ainsi que gérer les épreuves et les sessions associées à ces compétitions

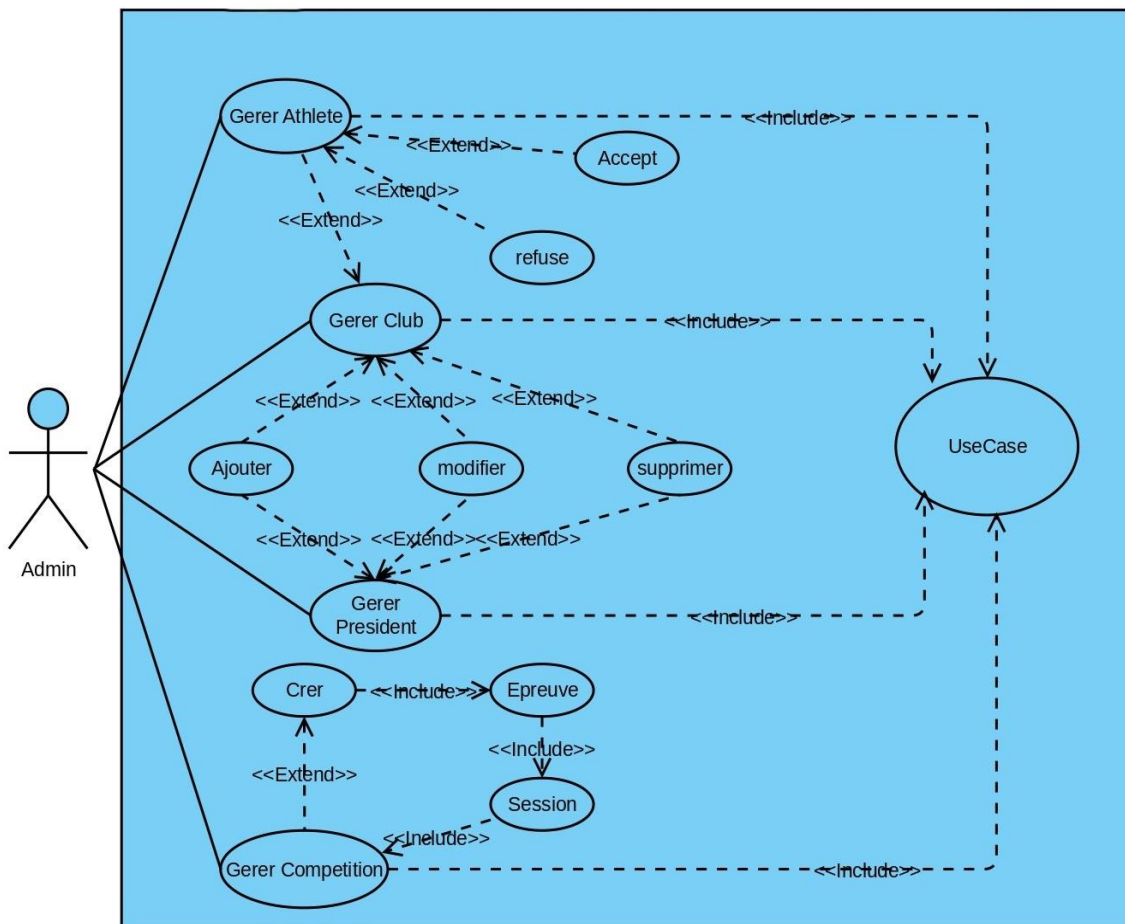


Figure 3. 1: Diagramme de cas d'utilisation globale pour l'administration (Admin-ligue).

▪ **Partie Agent :**

Le diagramme de cas d'utilisation pour l'agent détaille les tâches spécifiques que ce rôle peut accomplir dans le système de gestion. L'agent doit d'abord s'authentifier pour accéder au système. Ensuite, il peut gérer les athlètes en ajoutant ou en modifiant leurs informations et en choisissant les athlètes à inscrire dans des compétitions. L'inscription des athlètes dans des compétitions inclut la sélection des épreuves et la gestion des sessions pour chaque compétition. De plus, l'agent peut gérer les compétitions en sélectionnant les épreuves et en définissant les sessions ainsi que les temps d'engagement.

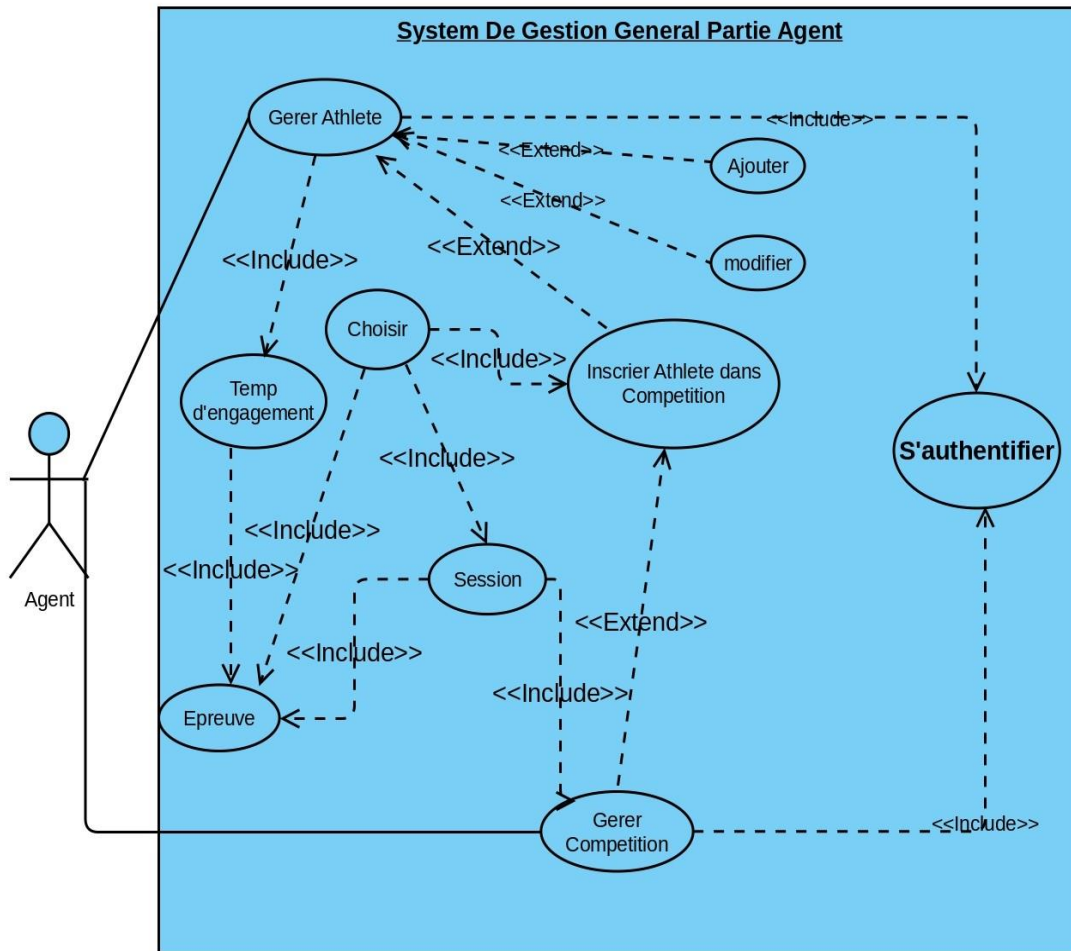


Figure 3. 2: Diagramme de cas d'utilisation globale (Agent-club).

3.4. Conception

3.4.1. Les diagrammes de séquence

3.4.1 - 1. Définition

Le diagramme de séquence est un outil de modélisation utilisé pour représenter visuellement l'ordre temporel des messages échangés entre les objets d'un système logiciel. Dans un diagramme de séquence :

- Les objets participant à l'interaction sont placés le long de l'axe horizontal (X), généralement de gauche à droite, avec l'objet initiateur de l'interaction à gauche.
- Les messages échangés entre les objets sont représentés le long de l'axe vertical (Y), de haut en bas, pour montrer l'ordre dans lequel ils sont envoyés et reçus.
- Chaque objet est représenté par une ligne de vie verticale pointillée, qui montre son existence pendant la durée de l'interaction.
- Les messages envoyés entre les objets sont représentés par des flèches reliant les lignes de vie des objets concernés.
- Les contraintes de temps ou d'espace peuvent être spécifiées pour chaque message, si nécessaire.
- Les focus de contrôle peuvent être utilisés pour montrer quand un objet est activement impliqué dans une action.

3.4.1 - 2. Diagramme de séquence « S'authentifier »

➤ Description textuelle de diagramme de séquence «S'authentifier»

L'authentification vise à garantir la confidentialité des données en vérifiant les informations associées à un utilisateur, telles que le nom d'utilisateur et le mot de passe. Ces informations sont préalablement enregistrées dans une base de données. Lors d'une tentative d'authentification, deux situations peuvent se présenter :

- ✓ Les informations fournies par l'utilisateur sont incomplètes ou incorrectes : dans ce cas, un message d'erreur apparaît et la même interface est rechargée pour un nouvel essai.
- ✓ Les informations fournies par l'utilisateur sont complètes et correctes : l'interface correspondante est affichée.
- ✓ Cela explique l'utilisation de l'opérateur « alt ».

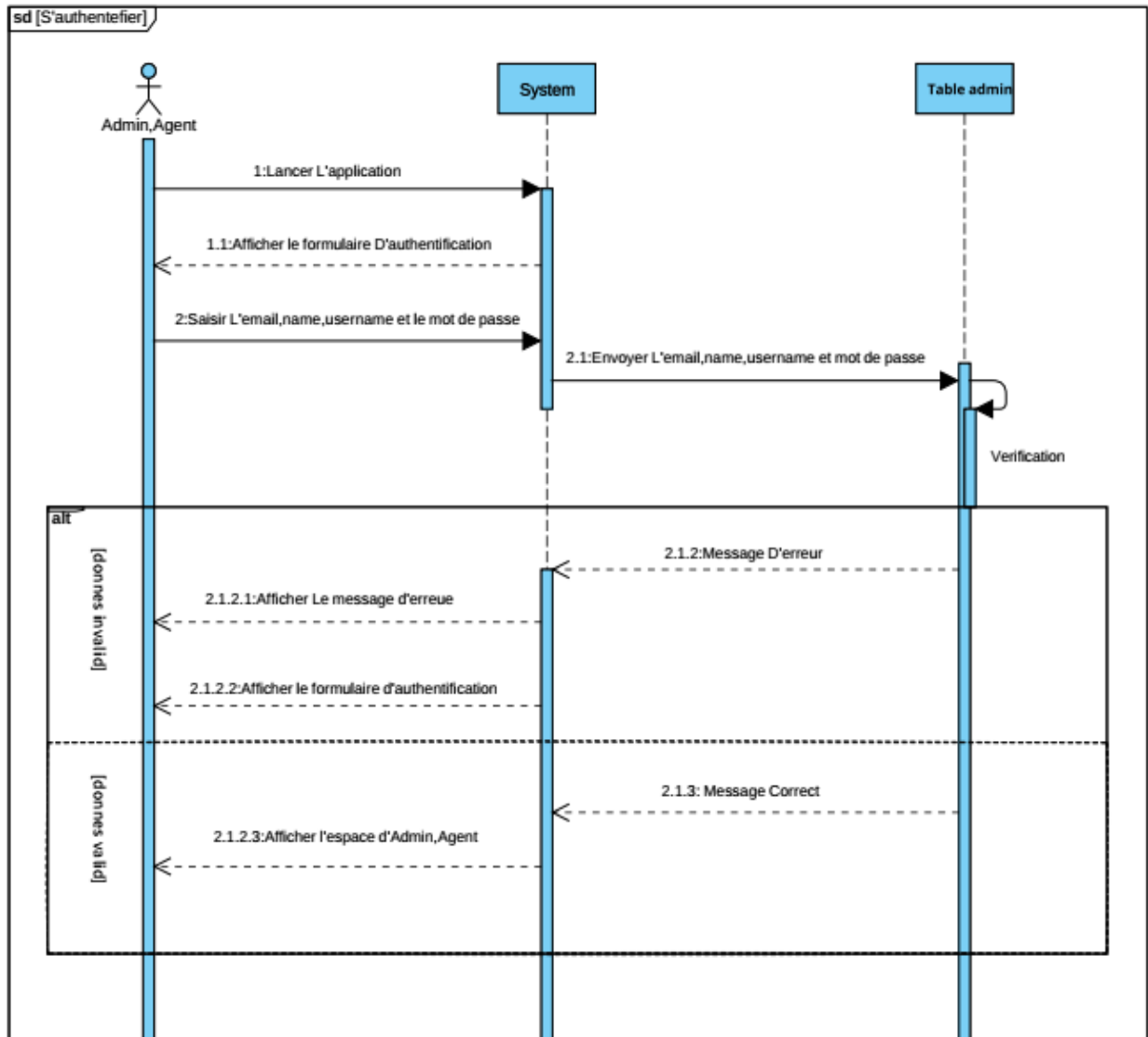


Figure 3. 3: Diagramme de séquence « S'authentifier ».

3.4.1 -3. Diagramme de séquence « créer l'épreuve »

➤ **Description textuelle de diagramme de séquence « créer l'épreuve »**

- Après s'être authentifié, l'administrateur demande l'accès à l'espace épreuves.
- Le système affiche alors cet espace.
- L'administrateur choisit ensuite d'ajouter une nouvelle épreuve.
- Le système lui présente un formulaire d'ajout.

- L'administrateur remplit les informations nécessaires à la création de l'épreuve et les sauvegarde.
- Le système enregistre les données dans la base de données, puis affiche la nouvelle épreuve dans le tableau des épreuves de l'espace épreuves.

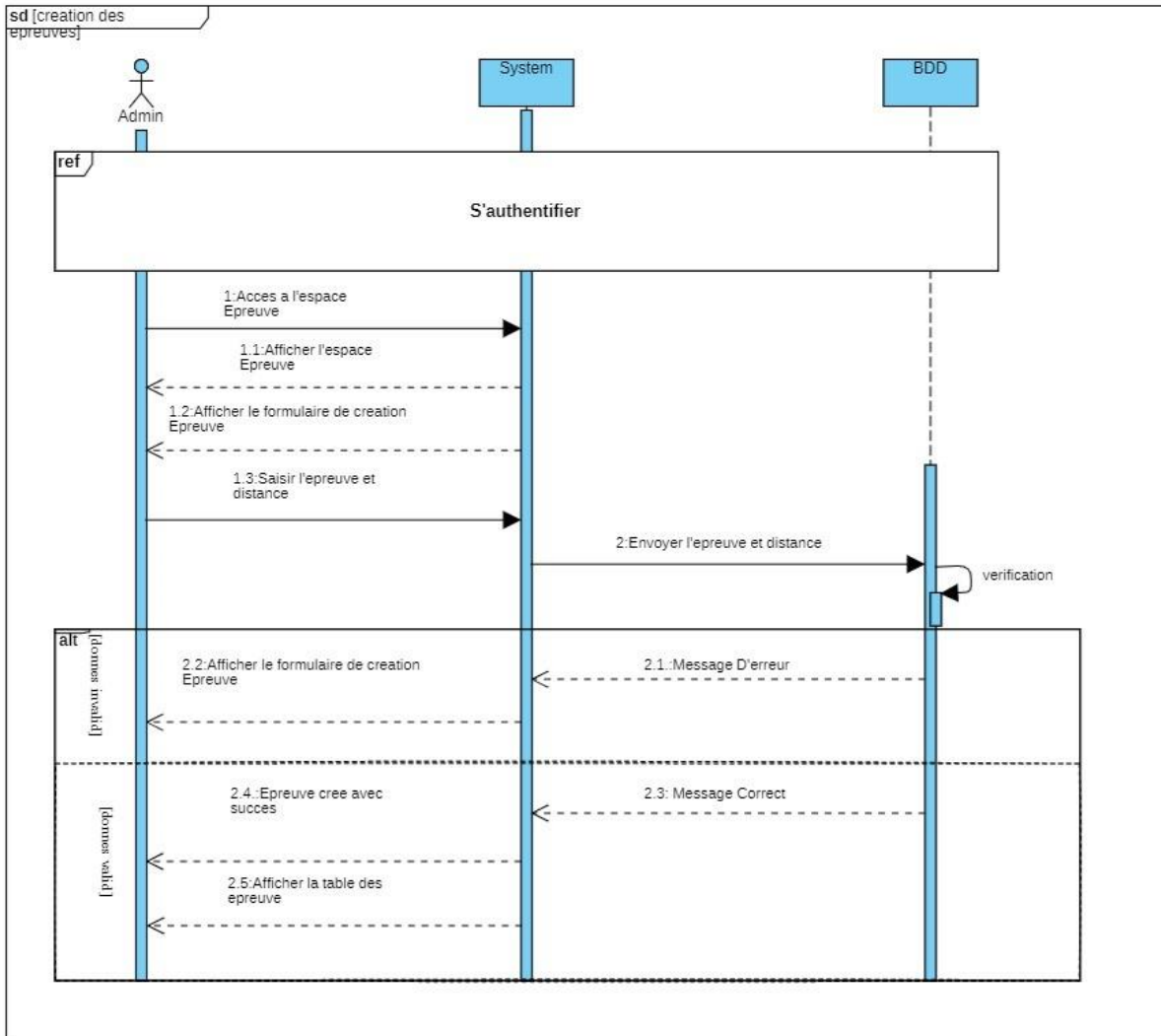


Figure 3. 12: Diagramme de séquence « créer l'épreuve ».

3.4.1- 4. Diagramme de séquence « créer compétition »

➤ Description textuelle de diagramme de séquence « créer compétition »

Après la création de l'épreuve, l'administrateur demande l'accès à l'espace compétition. Le système répond en affichant cet espace.

Chapitre III : Analyse et Conception

L'administrateur choisit ensuite d'ajouter une nouvelle compétition. Le système affiche le formulaire d'ajout de compétition. L'administrateur remplit les informations nécessaires, dont un champ pour le nombre de sessions.

Il saisit le nombre de sessions pour cette compétition, puis soumet le formulaire. Le système enregistre les informations dans la base de données et crée automatiquement les sessions correspondant au nombre indiqué. Enfin, le système affiche la compétition dans le tableau des compétitions.

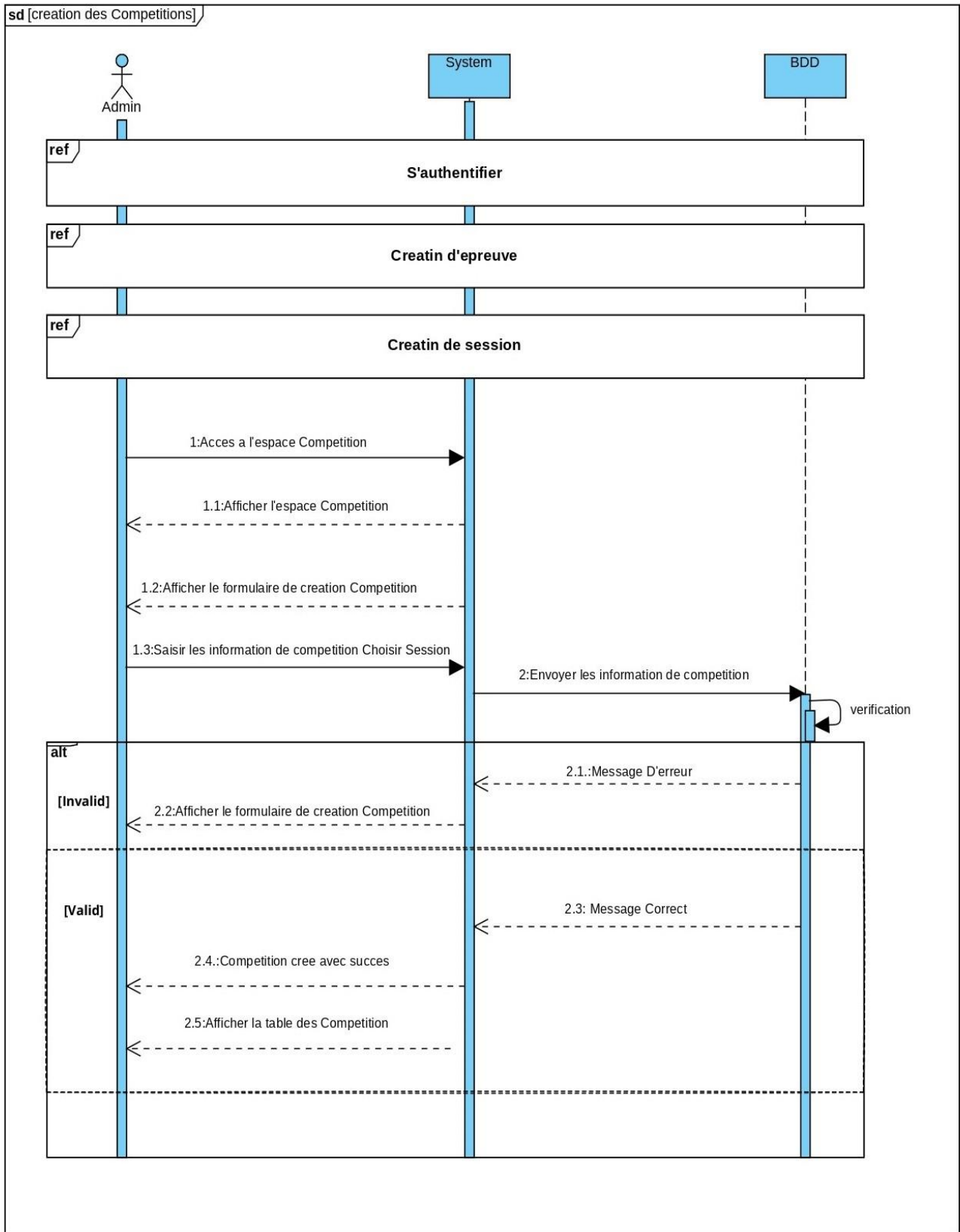


Figure 3. 13: Diagramme de séquence « créer compétition».

3.4.1 - 5. Diagramme de séquence « créer Session »

➤ Description textuelle de diagramme de séquence « créer Session »

Après la création de la compétition, l'administrateur demande l'accès à l'espace des sessions. Le système affiche cet espace.

L'administrateur sélectionne une session et demande d'ajouter des épreuves. Le système affiche alors le formulaire d'ajout d'épreuve. L'administrateur remplit le formulaire, ajoute les épreuves, puis le soumet.

Le système enregistre les informations dans la base de données et affiche le tableau de la session avec les épreuves ajoutées.

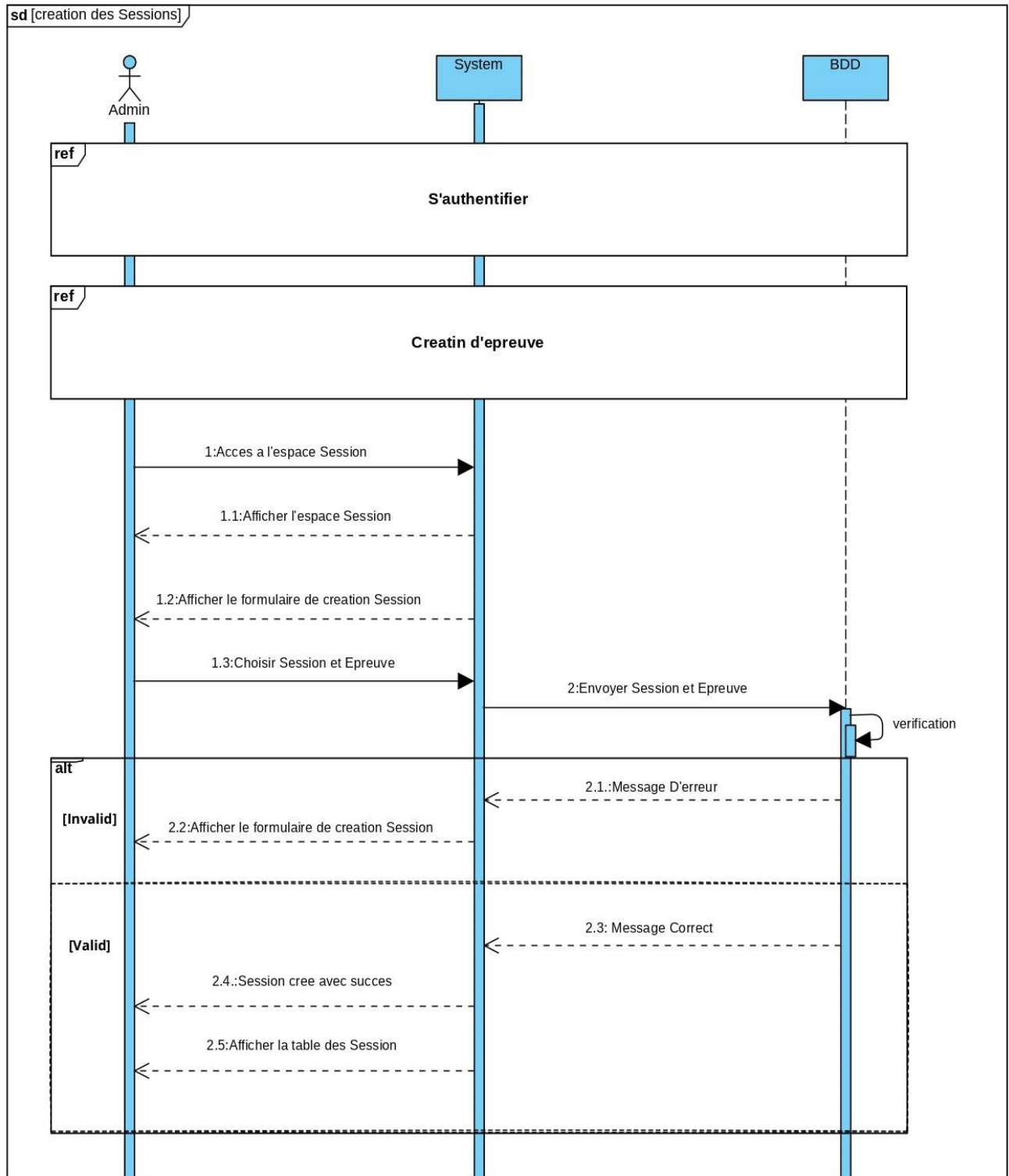


Figure 3. 14: Diagramme de séquence « créer Session».

3.4.1 - 6. Diagramme de séquence « gérer athlète »

➤ Description textuelle de diagramme de séquence « gérer athlète »

❖ **Authentification:**

- Agent : L'agent doit s'authentifier.
- Système : Procède à l'authentification de l'agent.

❖ **Accès à l'espace Athlète:**

- Agent : L'agent demande l'accès à l'espace Athlète.
- Système : Le système affiche l'espace Athlète pour l'agent.

❖ **Demande d'ajout d'un Athlète:**

- Agent : L'agent demande à ajouter un nouvel athlète.
- Système : Le système affiche le formulaire pour ajouter un athlète.

❖ **Saisie des informations de l'athlète :**

- Agent : L'agent saisit les informations de l'athlète dans le formulaire.
- Système : Les informations de l'athlète sont envoyées à la base de données (BDD).

❖ **Traitement des informations par l'Admin :**

- Système : Affiche les informations de l'athlète dans l'espace en attente pour l'administrateur.
- Admin : L'administrateur choisit d'accepter ou de refuser l'athlète.

❖ **Scénarios de réponse :**

- Refus de l'athlète :
 - Système : Affiche l'athlète dans l'espace refusé.
 - Agent: L'administrateur peut corriger les informations de l'athlète et renvoyer les informations pour une nouvelle vérification.
 - Système : Demande à l'administrateur de confirmer l'acceptation de l'athlète.
- Acceptation de l'athlète :

- Admin : L'administrateur accepte l'athlète.
- Système : Affiche l'athlète dans l'espace accepté.
- ❖ **Vérification et Sauvegarde :**
 - Système : Vérifie les informations de l'athlète et les sauvegarde dans la base de données.

Chapitre III : Analyse et Conception

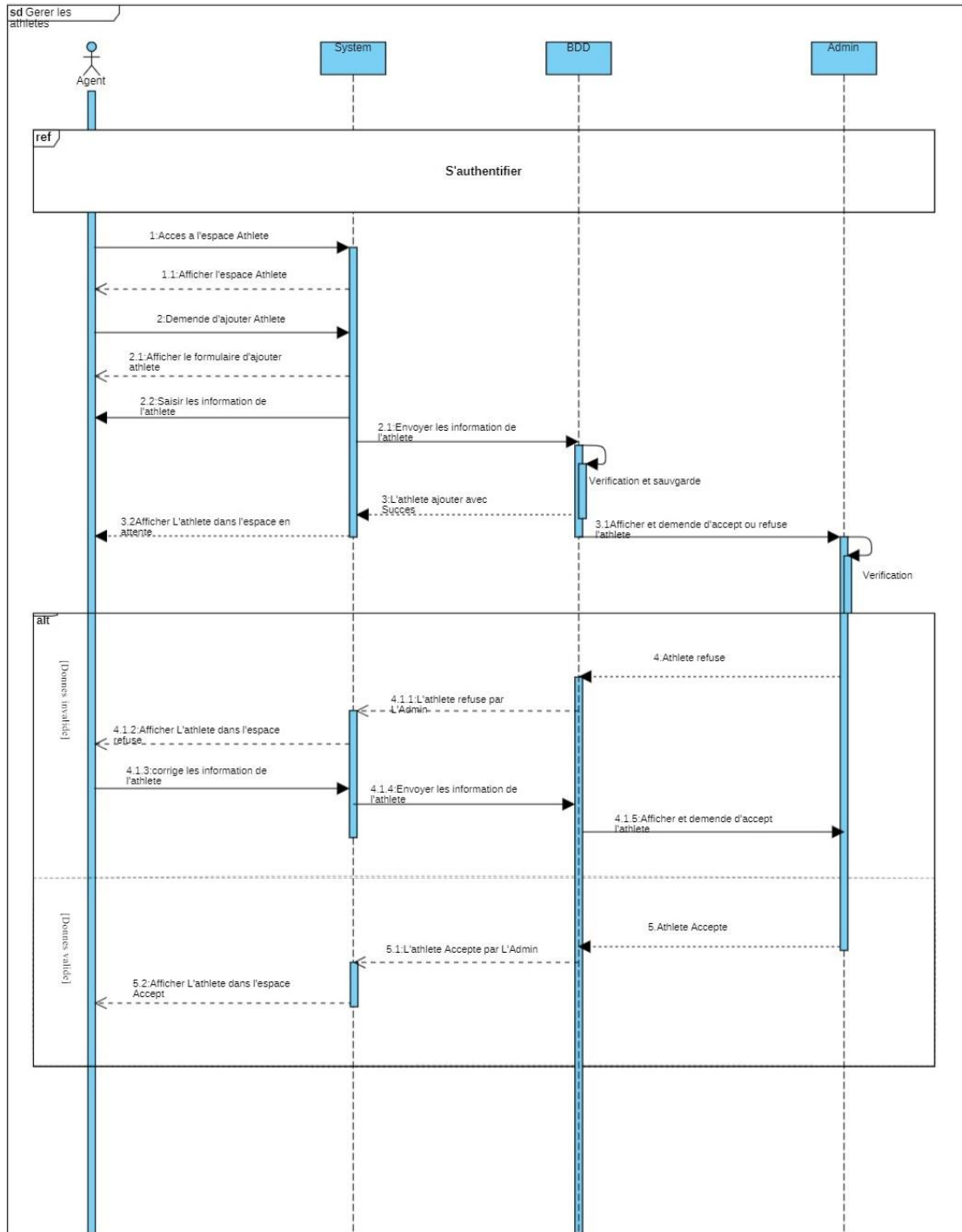


Figure 3. 15: Diagramme de séquence « gérer athlète ».

3.4.2. Le diagramme de classes

3.4.2 -1. Définition

Un diagramme de classe est un outil de modélisation utilisé en génie logiciel pour représenter les composants statiques d'un système logiciel, tels que les classes, les interfaces et les relations entre elles. Il fournit une représentation visuelle des entités du système et de leurs interactions, ce qui permet aux développeurs de comprendre la structure et l'organisation du système avant sa mise en œuvre. [15]

Chapitre III : Analyse et Conception

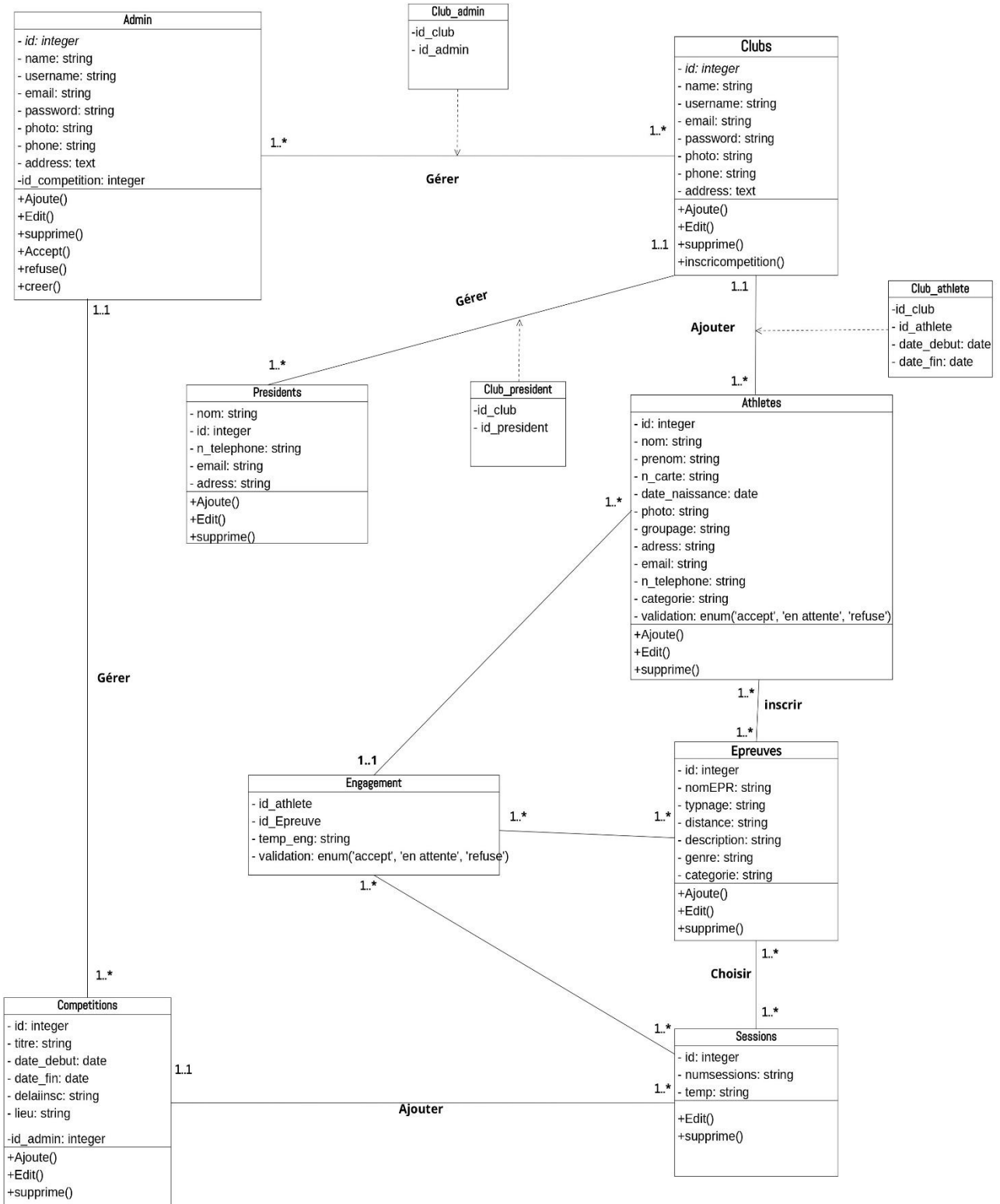


Figure 3. 24: Diagramme de classes de l’application.

3.4.3. Conception de la base de données

Après avoir établi le diagramme de classes pour notre application, nous passons à la représentation physique de nos classes d'objets persistants en utilisant le modèle relationnel de données.

3.4.3 -1. Le Modèle relationnel

Le modèle relationnel décrit l'univers des données représenté dans le modèle entité/association, en tenant compte du type de base de données choisi (relationnelle ou autre, comme les bases de données en réseau ou hiérarchiques). En d'autres termes, le modèle relationnel convertit le modèle entité/association en un format compréhensible par la machine, facilitant ainsi l'implémentation de la base de données.

➤ Règle de passage du diagramme de classes vers le modèle relationnel

Pour obtenir un schéma relationnel à partir d'un diagramme de classes, il est nécessaire de suivre ces étapes :

- **Traduction des classes en relations** : Chaque classe du diagramme est convertie en une table dans le schéma relationnel.
- **Traduction des associations**
 - Pour les associations où une multiplicité maximale est inférieure ou égale à 1 (désignons la classe avec cette multiplicité par A), on ajoute la clé primaire de la classe A comme attribut dans la table de l'autre classe.
 - Pour les autres types d'associations, on crée une table associative dont la clé primaire est composée des clés primaires des tables représentant les classes impliquées dans l'association. Les autres attributs de cette table associative sont ceux de la classe liée à cette association, si elle existe.
- **Traduction de l'héritage** : On partage la même clé primaire entre les classes parent et enfant. La clé primaire de la classe enfant est déclarée comme clé étrangère correspondant à la clé primaire de la classe parent. [16]
- **Schéma relationnel** :

Admin(id(Primary Key),name,username,email,password,photo,phone,address)

Presidents(id(Primary Key),nom,n_telephone,email,address)

Clubs(id(Primary Key), name,username,email,password,photo,phone,address)

Chapitre III : Analyse et Conception

Athletes(id(Primary Key), nom,prenom,n_carte,date_naissance,photo,groupage,address, email, n_telephone, categorie, validation)

Competitions (id(Primary Key),titre,date_debut,date_fin,delaiinsc,lieu, #id_admin)

Engagement (#id athlete (Foreign Key references Athletes(id)),#id epreuve (Foreign Key references Epreuves(id)),temp_eng,validation)

Epreuves (id (Primary Key),nomEP,distance,description,genre,categorie)

Sessions (id (Primary Key), numsession, temp,#id_competition)

Club_athlete (#id club (Foreign Key references Clubs(id)),#id_athlete (Foreign Key references Athletes(id)),date_debut,date_fin)

3.4.3 -2. Le modèle physique de données

Le modèle physique de données est la conversion du modèle logique de données en un langage de description spécifique au système de gestion de bases de données utilisé. Il s'agit de l'implantation concrète des données (tables) dérivées du modèle logique sur la machine, aboutissant ainsi à la description des fichiers de la base de données. [17]

➤ Les tables de notre base de données :

- **Table Admin:**

Nom du champ	Type de donnée	Taille	Description	Observation
id	Int		Identifiant Admin	Clé primaire
name	Varchar	30	Le nom de l'Admin	
username	Varchar	30	Nom d'utilisation	
email	Varchar	100	Email de l'Admin	
password	Varchar	150	Mot de passe de compte de l'Admin	
photo			Photo de compte de l'Admin	
phone	Int	10	Numero de telephone de l'Admin	
address	Varchar	150	Address de l'admin	

Chapitre III : Analyse et Conception

- **Table Presidents:**

Nom du champ	Type de donnée	Taille	Description	Observation
id	Int		Identifiant Presidents	Clé primaire
nom	Varchar	50	Le nom de president	
n_telephone	Int	10	Numero de telephone de l'Admin	
email			Email de president	
address	Varchar	150	Address de president	

- **Table Clubs:**

Nom du champ	Type de donnée	Taille	Description	Observation
id	Int		Identifiant Clubs	Clé primaire
name	Varchar	30	Nom du club	
username	Varchar	30	Abriviation de club	
email	Varchar	100	Email de club	
password	Varchar	150	Mot de passe de compte de club	
photo			Logo de club	
phone	Int	10	Numero de telephone de club	
address	Varchar	150	Address de club	

- **Table Athletes:**

Nom du champ	Type de donnée	Taille	Description	Observation
id	Int		Identifiant Athletes	Clé primaire
nom	Varchar	30	Le nom de l'Athlete	
prenom	Varchar	30	Le prenom de l'Athlete	
n_carte	Int	18	Numero de la lesence	
date_naissance	date		Date de naissqnce de l'Athlete	
photo			Photo de l'Athlete	
groupage	Int		Groupe sanguin de l'Athlete	
address	Varchar	150	Adrdress de l'Athlete	
email	Varchar		Email de l'Athlete	
n_telephone	Int	10	Numero de telephone de l'Athlete	
categorie	Varchar		La categorie de l'Athlete	
validation	Varchar		La validation des athletes	

Chapitre III : Analyse et Conception

- **Table Competitions:**

Nom du champ	Type de donnée	Taille	Description	Observation
id	Int		Identifiant de Athlete	Clé primaire
titre	varchar		Le titre de competition	
date_debut	date		Date debut de competition	
date_fin	date		Date fin de competition	
date_dinc	date		Date debut d'inscription	
date_finc	date		Date fin d'inscription	
delaiinc	varchar	2	Delai d'inscription	
lieu	varchar		Le lieu de competition	
N_session	varchar	1	Le nombre de session	

- **Table Engagement:**

Nom du champ	Type de donnée	Taille	Description	Observation
id_athlete	Int		Identifiant de Athlete	Clé étrangère
<u>id_epreuve</u>	Int		Number de epreuve	Clé étrangère
temp_eng	Reel	3	Temp d'engagement	
validation	varchar		Validation des athlete dans competition	

- **Table Epreuves:**

Nom du champ	Type de donnée	Taille	Description	Observation
<u>id</u>	Int		Identifiant de epreuve	Clé primaire
nomEP	varchar		Number de session	
typnage	varchar		Le type de nage	
typeEpr	varchar		Le type de l'epreuve	
distance	Int	4	Temp de session	
description	varchar		Description de epreuve	
genre	varchar		Genre de l'epreuve	
categorie	varchar		Categorie de l'epreuve	

- **Table Sessions:**

Nom du champ	Type de donnée	Taille	Description	Observation
<u>id</u>	Int		Identifiant de session	Clé primaire
numsession	Int	2	Nombre de session	
temp	varchar		Temp de session	

3.5. Conclusion

Ce chapitre a permis de développer une application web pour la gestion de la succession, en utilisant UML pour la modélisation et en détaillant les différentes fonctionnalités et la conception de la base de données.

Cette approche méthodique et structurée permet de créer un système efficace et adaptable, répondant aux besoins spécifiques de la gestion de la ligue de natation.

CHAPITRE IV :

Implémentation et mise en œuvre

4.1. Introduction

Ce chapitre se concentre sur les aspects techniques de l'implémentation et du déploiement de notre application. Nous commencerons par décrire l'environnement de développement, les technologies et les outils que nous avons choisis et utilisés. Ensuite, nous présenterons les interfaces principales de l'application ainsi que les fonctionnalités qu'elles englobent.

4.2. Architecture globale de l'application

Les schémas suivants montrent l'architecture globale de l'application, qu'il s'agisse de l'administration ou de l'agent

4.2.1. Partie Admin : l'architecture globale pour l'admin

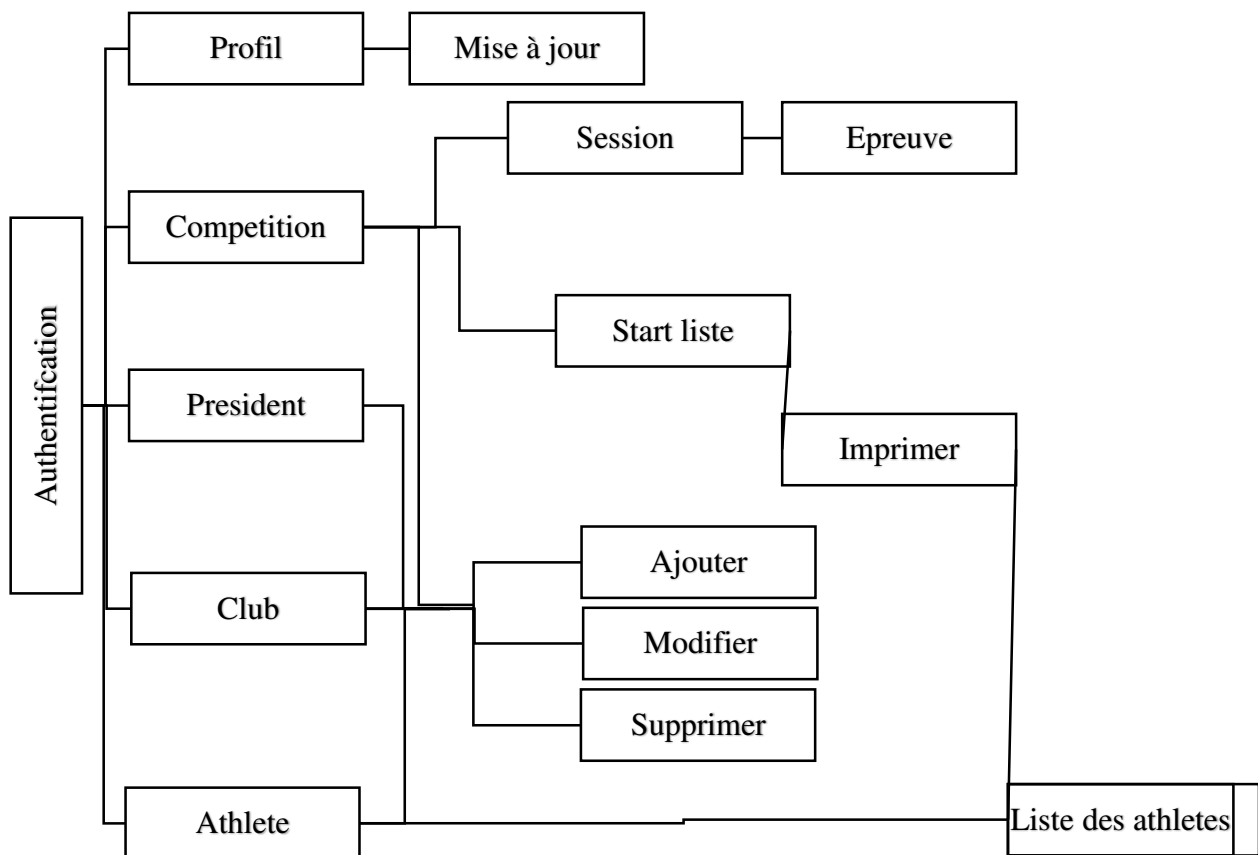


Figure 4. 1: Représentation de l'architecture globale de l'application(Admin).

4.2.2. Partie Agent : l'architecture globale pour l'agent

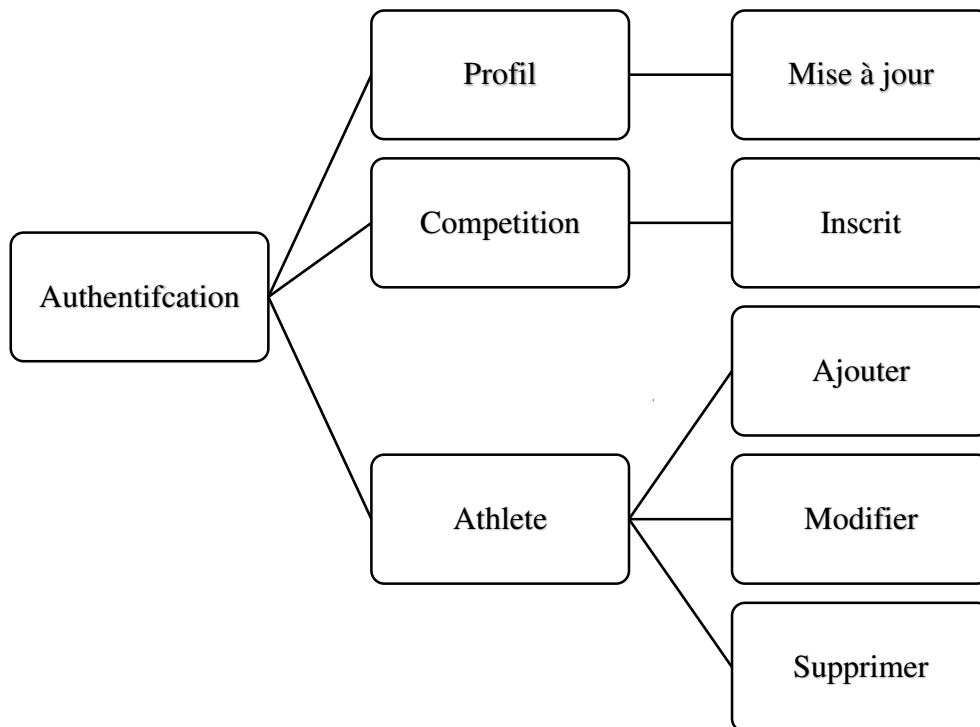


Figure 4. 2: Représentation de l'architecture globale de l'application(Agent).

4.3. L'environnement de développement

4.3.1. Les langages de programmation utilisés

Les langages de programmation utilisés dans notre projet comprennent HTML5, CSS3, JavaScript, Bootstrap, PHP, Apache Tomcat, et MySQL. Chacun de ces langages et technologies joue un rôle crucial dans la création d'applications web dynamiques, réactives et sécurisées.

4.3.1- 1. HTML5

- HTML5 est la cinquième version majeure du langage HTML (HyperText Markup Language), conçue pour améliorer les fonctionnalités et l'interopérabilité des pages et applications web.
- HTML5 facilite la création d'applications web dynamiques, réactives et accessibles, répondant aux besoins modernes du développement web. [18]

4.3.1- 2. Ccss3

- CSS3 (Cascading Style Sheets, niveau 3) est la dernière version du langage de feuilles de style en cascade, utilisé pour décrire la présentation des documents HTML ou XML.

- CSS3 est essentiel pour la conception web moderne, offrant des outils pour créer des interfaces riches, réactives et performantes. [19]

4.3.1- 3. JavaScripts

- JavaScript est un langage de programmation interprété, orienté objet et basé sur des prototypes, principalement utilisé pour rendre les pages web interactives. Il est un composant essentiel de l'architecture des pages web modernes et est utilisé à la fois côté client et côté serveur.
- JavaScript est essentiel pour le développement web moderne, offrant les outils nécessaires pour créer des applications web interactives et performantes. [20]

4.3.1- 4. Bootstrap

- Bootstrap est un framework front-end open-source conçu pour développer des sites web et des applications web réactifs et mobiles. Créé par Twitter, il facilite la conception web avec des modèles HTML et CSS pour divers éléments et des composants JavaScript pour des fonctionnalités interactives.
- Bootstrap permet de créer rapidement des sites et des applications modernes, réactifs et esthétiques, avec une compatibilité et une accessibilité optimales. [21]

4.3.1- 5. PHP (Hypertext Preprocessor) :

est un langage de programmation de scripts côté serveur, principalement utilisé pour le développement web dynamique et interactif. Selon Olivier Heurtel dans son ouvrage "PHP y MySQL" (2014), PHP est l'une des technologies les plus populaires pour créer des sites web complexes, en particulier lorsqu'il est utilisé conjointement avec MySQL pour la gestion des bases de données.

PHP a été créé en 1994 par Rasmus Lerdorf pour simplifier la gestion de son site personnel. Rapidement, il a évolué grâce à la contribution d'une large communauté de développeurs.

- **Dernier version de PHP :** PHP 8, publié le 26 novembre 2020, introduit des fonctionnalités majeures et des améliorations de performance significatives. Cette nouvelle version marque une évolution importante pour le langage de programmation PHP utilisé principalement pour le développement web.
- ❖ **Utilisation et Applications :** PHP est utilisé dans divers contextes de développement web, notamment :
 - Sites Web Dynamiques

- Utilisé pour générer des pages web dynamiques, gérer des sessions utilisateur, et interagir avec des bases de données pour présenter des contenus personnalisés.
- Systèmes de Gestion de Contenu (CMS)
- Des CMS populaires tels que WordPress, Joomla et Drupal sont basés sur PHP, offrant des solutions puissantes pour la gestion de contenu en ligne.
- Applications Web Complètes
- PHP est utilisé pour développer des plateformes de commerce électronique, des forums, des réseaux sociaux, et d'autres types d'applications web interactives et complexes.

a. Avantages de PHP :

➤ **Open Source**

PHP est gratuit et open source, ce qui le rend accessible à tous les développeurs sans frais de licence.

➤ **Performance et Efficacité**

Les versions récentes de PHP offrent des améliorations significatives en termes de vitesse et d'efficacité, notamment grâce à des optimisations internes et à la compilation JIT.

➤ **Flexibilité et Extensibilité**

PHP peut être facilement étendu avec des extensions et des bibliothèques tierces, ce qui permet de l'adapter à divers besoins de développement.

➤ **Compatibilité avec Diverses Bases de Données**

PHP fonctionne bien avec une variété de systèmes de gestion de bases de données, y compris MySQL, PostgreSQL, SQLite, et Oracle, offrant ainsi une flexibilité maximale pour la gestion des données.

➤ **Sécurité**

Bien que la sécurité dépende en grande partie de la manière dont le code est écrit, PHP offre de nombreuses fonctionnalités de sécurité intégrées et des meilleures pratiques pour aider les développeurs à créer des applications sécurisées. [22]

4.3.1 -6. Apache

Apache Tomcat est un serveur web open-source développé par la Fondation Apache, utilisé principalement comme conteneur servlet pour les applications web Java. Il fournit un environnement d'exécution pour les servlets Java, les JavaServer Pages (JSP) et d'autres technologies Java utilisées pour créer des applications web dynamiques. En résumé, Apache Tomcat permet de déployer et d'exécuter des applications web Java de manière efficace et sécurisée. [23]

4.3.1 -7. MySQL

MySQL est un système de gestion de base de données relationnelle open-source largement utilisé. Il permet de stocker, d'organiser et de récupérer des données de manière efficace. MySQL est souvent utilisé en combinaison avec des langages de programmation web comme PHP pour créer des sites web dynamiques et interactifs. Il offre des fonctionnalités telles que la gestion des utilisateurs, la gestion des transactions, les jointures de tables et bien d'autres, ce qui en fait un choix populaire pour de nombreux développeurs et organisations. [22]

4.4. Outils de développement

Les outils de développement utilisés dans notre projet incluent Visual Studio Code (VS Code), XAMPP, Git, le framework Laravel, et Node.js. Chacun de ces outils est crucial pour la création, la gestion, et le déploiement d'applications web modernes et efficaces.

4.4.1. Environnement de développement intégré Visual Studio

Visual Studio Code (VS Code) est un éditeur de code source développé par Microsoft pour Windows, macOS et Linux. Lancé en 2015, il est devenu l'un des outils de développement les plus populaires grâce à sa légèreté, sa flexibilité et son large éventail de fonctionnalités. VS Code est open-source et peut être personnalisé pour répondre à une variété de besoins de développement.

4.4.1- 1. Caractéristiques principales

➤ Interface utilisateur intuitive

- Explorateur de fichiers : Permet une navigation facile à travers les fichiers et dossiers de votre projet.
- Terminal intégré : Accès direct à une ligne de commande, éliminant le besoin de basculer entre différentes applications.
- Minimap : Vue d'ensemble de votre code pour une navigation rapide.

➤ **Support multiplateforme**

- Disponible sur Windows, macOS et Linux, permettant une expérience de développement cohérente sur différentes plateformes.

➤ **Extensions et personnalisations**

- Market Place : Un écosystème riche d'extensions pour ajouter de nouvelles fonctionnalités, allant des langages de programmation supplémentaires aux outils de débogage et aux thèmes de couleur.
- Personnalisation de l'interface : Thèmes, icônes et paramètres personnalisables pour adapter l'éditeur à vos préférences.

➤ **Intégration de Git**

- Contrôle de version intégré : Support natif pour Git, permettant de gérer les dépôts, de visualiser les modifications et de résoudre les conflits directement dans l'éditeur.
- Extensions Git : Extensions supplémentaires pour améliorer l'expérience Git, telles que GitLens pour des annotations détaillées sur le code.

➤ **Débogage puissant**

- Débogueur intégré : Support pour déboguer le code directement depuis l'éditeur, avec des points d'arrêt, des étapes et des inspections de variables.
- Extensions de débogage : Prise en charge de nombreux langages et environnements grâce à des extensions spécialisées.

➤ **Support de nombreux langages de programmation**

- Langages natifs : Support intégré pour JavaScript, TypeScript et Node.js.
- Extensions linguistiques : Prise en charge de nombreux autres langages via des extensions, y compris Python, C++, Java, PHP, et bien d'autres.

4.4.1- 2. Performance et utilisation

VS Code est conçu pour être léger et performant, ce qui le rend adapté à des projets de toutes tailles. Il démarre rapidement et consomme moins de ressources système par rapport à des environnements de développement intégrés (IDE) plus lourds. Cela permet aux développeurs de travailler efficacement même sur des machines moins puissantes. [24]

4.4.2. Le serveur XAMPP

XAMPP, lancé en 2002 dans le cadre du projet Apache Friends, a pour objectif de simplifier l'installation d'un serveur web complet. Cet outil est crucial pour le développement et la mise en

œuvre d'applications web dynamiques, en regroupant plusieurs logiciels (Apache, PHP, MySQL, Perl, etc.) en un seul package compatible avec les principaux systèmes d'exploitation.

4.4.2- 1. PhpMyAdmin

PhpMyAdmin est un utilitaire qui facilite la gestion des bases de données MySQL. Il s'agit d'un ensemble de scripts PHP permettant l'administration des bases de données MySQL via un navigateur web.

➤ **Fonctionnalités de PhpMyAdmin**

- Création et suppression de bases de données
- Création, modification, copie et suppression de tables
- Édition, ajout et suppression de champs
- Exécution de commandes SQL
- Création d'index
- Importation de fichiers dans des tables

➤ **Utilisation de PhpMyAdmin**

Pour accéder à PhpMyAdmin, assurez-vous que EasyPHP.exe est en cours d'exécution et que les serveurs sont actifs. Ensuite, faites un clic droit sur l'icône et sélectionnez "administration". Une page web s'ouvrira avec un bouton PhpMyAdmin au centre. Cliquez sur ce bouton pour afficher la page d'accueil de PhpMyAdmin dans le navigateur, où vous pourrez sélectionner les bases de données disponibles sur le serveur MySQL par défaut.

Pour visualiser le contenu d'une base de données personnelle, cliquez sur la ligne correspondante dans la partie gauche de la fenêtre du navigateur. Les tables de cette base de données seront alors affichées. [25]

4.4.2- 2. Le Git :

Git, conçu par Linus Torvalds en 2005 pour le noyau Linux, est un système de contrôle de version distribué qui se distingue par sa capacité à gérer minutieusement les révisions de logiciel. Contrairement aux systèmes traditionnels, Git considère les révisions comme primordiales, offrant ainsi une meilleure abstraction et favorisant la collaboration entre les développeurs . [26]

4.4.2- 3. Le Framework Laravel

Laravel est un framework open-source de développement web écrit en PHP. Créé par Taylor Otwell en 2011, il est devenu l'un des frameworks les plus populaires pour le développement web. Laravel se distingue par sa structure MVC (Modèle-Vue-Contrôleur) qui permet une séparation claire des préoccupations et une organisation logique du code. Il offre des fonctionnalités élégantes et un code clair et concis, permettant aux développeurs de travailler rapidement et efficacement.

❖ Avantages de Laravel

- ✓ Productivité accrue : Laravel est conçu pour améliorer la productivité des développeurs grâce à sa syntaxe claire et élégante, sa structure organisée et ses fonctionnalités intégrées. Il permet aux développeurs de travailler plus rapidement et plus efficacement.
- ✓ Sécurité améliorée : Laravel offre un système de sécurité robuste qui protège les applications web contre les attaques courantes telles que les attaques CSRF et XSS. Les développeurs peuvent se concentrer sur le développement de l'application sans se soucier de la sécurité.
- ✓ Évolutivité : Laravel est conçu pour être évolutif, ce qui permet aux développeurs d'ajouter facilement de nouvelles fonctionnalités et d'étendre l'application en fonction des besoins. Cela permet aux entreprises de s'adapter rapidement à l'évolution des exigences et de rester compétitives.
- ✓ Documentation complète : Laravel dispose d'une documentation complète et à jour qui facilite l'apprentissage du framework pour les nouveaux développeurs. La documentation est un excellent point de départ pour apprendre Laravel et pour résoudre les problèmes rencontrés lors du développement de l'application.
- ✓ Communauté active : Laravel bénéficie d'une communauté active de développeurs qui partagent des connaissances et des ressources utiles. Les développeurs peuvent trouver rapidement de l'aide en cas de problème.
- ✓ Facilité de déploiement : Laravel facilite le déploiement de l'application web grâce à des outils intégrés tels que Laravel Forge et Laravel Envoyer. Les développeurs peuvent déployer leur application en quelques minutes seulement, ce qui leur permet de se concentrer sur le développement de l'application. [27]

4.4.2- 4. Node.js

Node.js est un environnement d'exécution JavaScript open-source et multiplateforme. C'est un outil populaire pour presque tous types de projets !

Node.js exécute le moteur JavaScript V8, le cœur de Google Chrome, en dehors du navigateur. Cela permet à Node.js d'être très performant.

Une application Node.js s'exécute dans un seul processus, sans créer un nouveau thread pour chaque requête. Node.js fournit un ensemble de primitives d'E/S asynchrones dans sa bibliothèque standard qui empêchent le code JavaScript de bloquer, et généralement, les bibliothèques en Node.js sont écrites en utilisant des paradigmes non bloquants, rendant le blocage de comportement l'exception plutôt que la norme. [28]

4.5. Présentation des interfaces de l'application

Dans cette partie, nous allons montrer les différentes interfaces de notre application pour illustrer la gestion de la ligue à l'aide de l'application.

4.5.1. L'interface (Authentification) :

La première interface à laquelle accèdent l'administrateur de la ligue et l'agent de club. Ils saisissent leur nom et leur mot de passe pour accéder à l'application et la gérer.

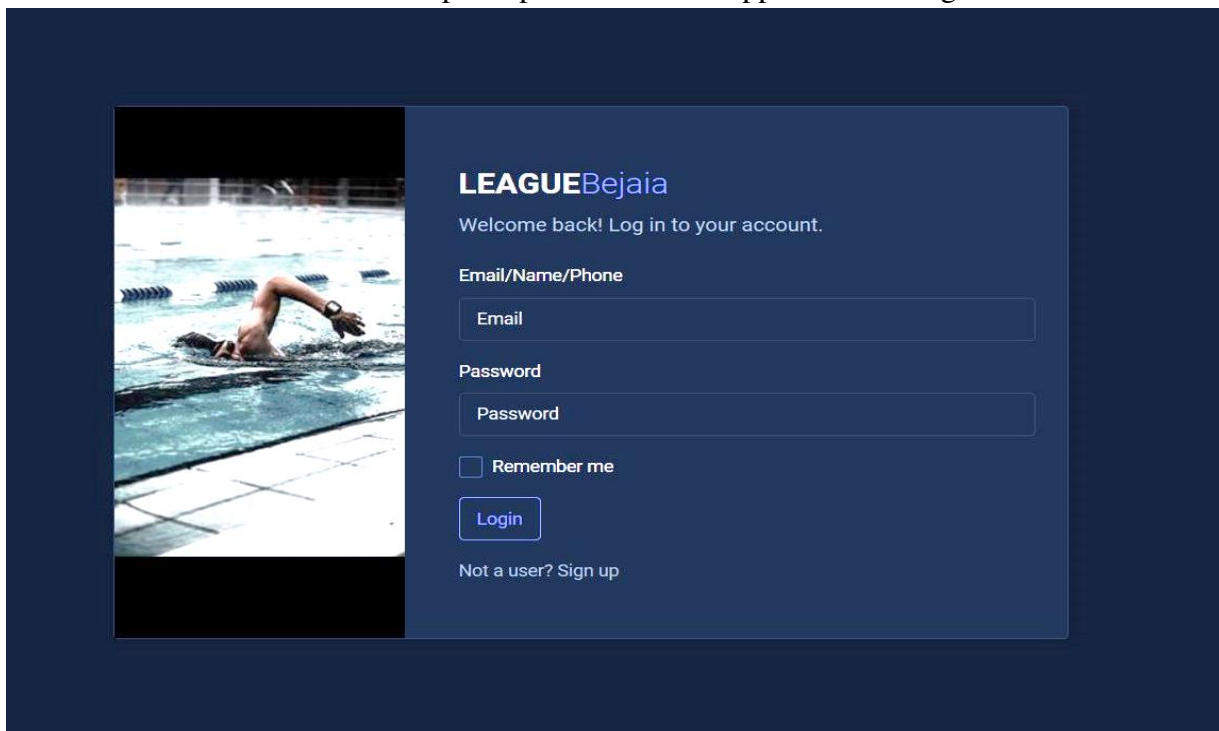


Figure 4. 3:L'interface (Authentification).

4.5.2. L'interface (Dashboard) :

La deuxième interface fournie par l'application après l'authentification est Dashboard. Il contient des statistiques sur les athlètes, les clubs, les compétitions, etc.

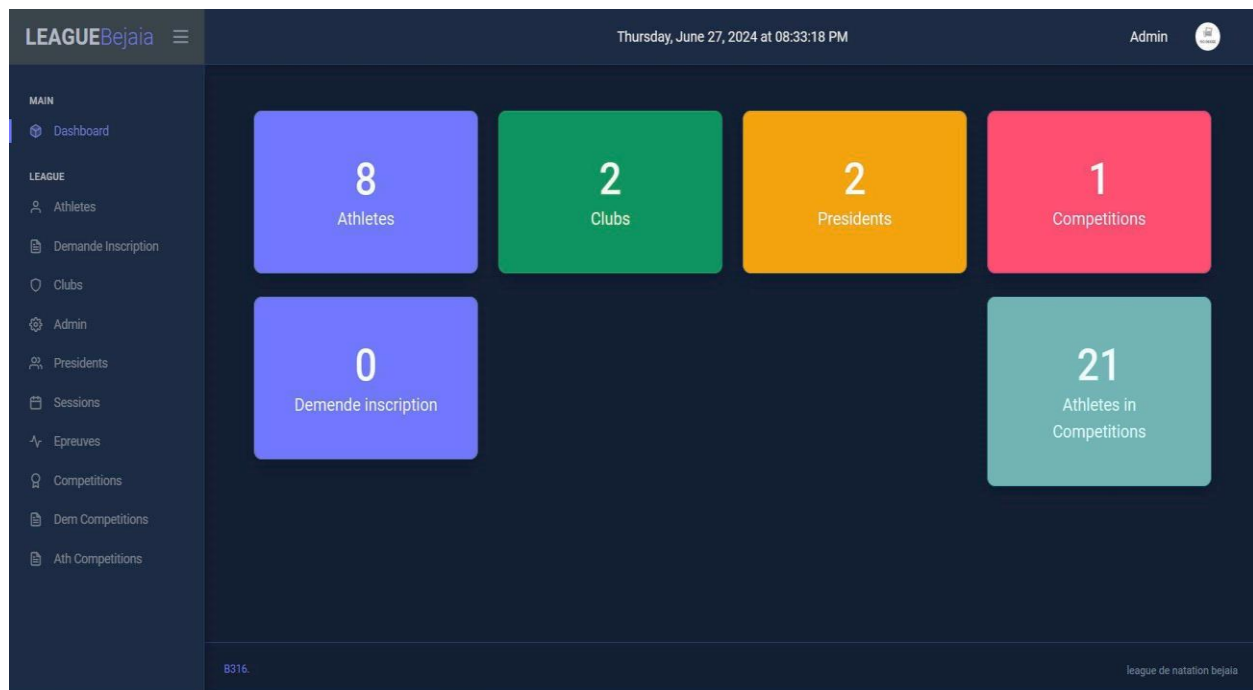


Figure 4. 4:L’interface (Dashboard).

4.5.3.L’interface (Profil) :

L'administrateur de la ligue et l'agent de club peuvent accéder aux profils pour modifier ce qu'ils souhaitent, comme le nom, le numéro de téléphone, la photo, le mot de passe, etc.

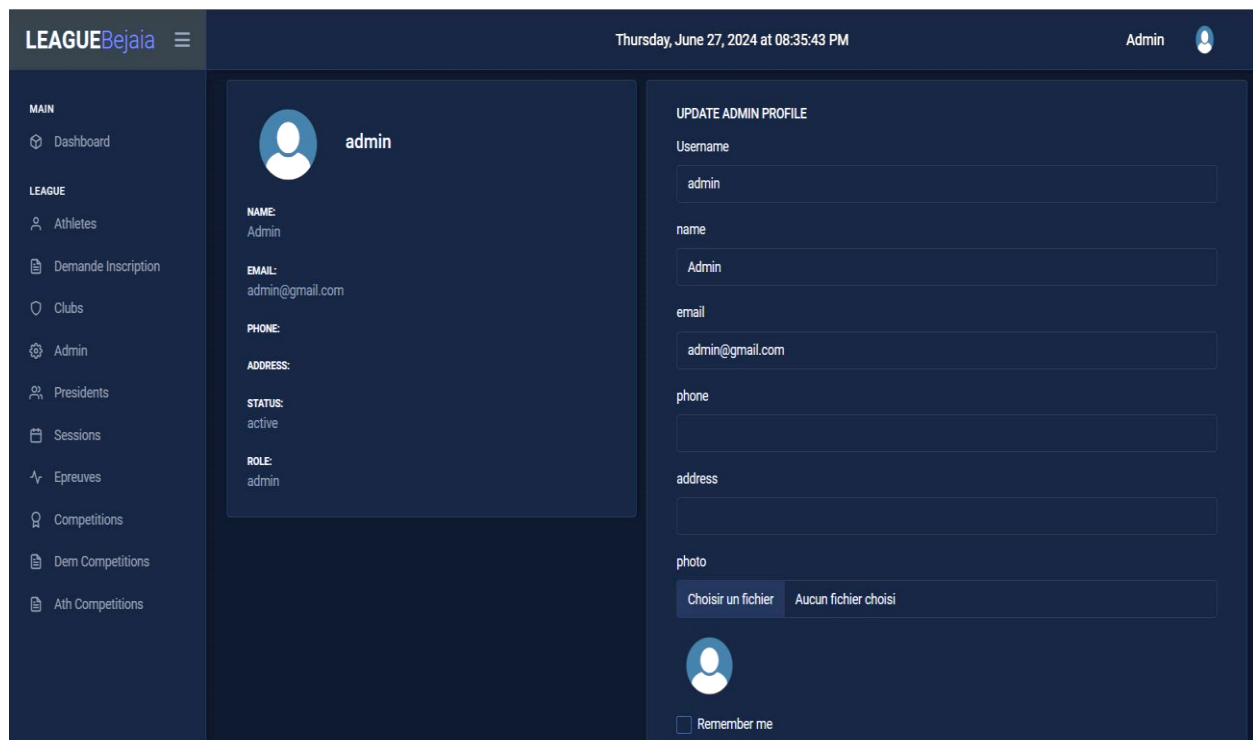


Figure 4. 5:L’interface (Profil).

4.5.4. Les scenario :

Dans notre application, il y a trois scénarios principaux. Le premier est la création et l'acceptation des athlètes, le deuxième est la création des compétitions, et le troisième est la participation des athlètes aux compétitions. C'est ce que nous allons montrer dans cette partie.

4.5.4 - 1. L'inscription :

Le premier scénario est l'inscription des athlètes dans la ligue de natation.

- **L'interface (Création) :**

Le club crée des athlètes en entrant toutes les informations les concernant dans cette interface et en les sauvegardant. Ensuite, les informations sont automatiquement envoyées à la ligue.

The screenshot shows a web application interface for managing athletes. The main content area is titled 'MODIFIER UN ATHLETE' and contains a form with the following fields and values:

Field	Value
Nom:	kaddour
Prenom:	Mohamed djebri
N° de carte:	346587236587236598
Date de naissance:	01/08/1998
Groupe sanguin:	O+
Address:	tamanrasset
Email address:	mohamed@gmail.com
N° telephone:	0668401285
Categorie:	Senior
Genre:	homme
Date de debut:	03/06/2024
Date de fin:	31/10/2025
Club:	mcbejaia
photo:	Choisir un fichier

At the bottom of the form, there are two buttons: 'Enregistrer' (green) and 'Annuler' (blue). The left sidebar shows navigation options for MAIN, LEAGUE, and ATHLETE COMPETITIONS.

Figure 4. 6:L'interface (Création).

- **L'interface (Demende) :**

La ligue reçoit les informations de l'athlète et peut les accepter ou les refuser.

The screenshot shows the 'Demande' (Request) interface in the LEAGUEBejaia application. It features a sidebar with navigation options like Dashboard, Athletes, and Demande Inscription. The main content area displays a table of 8 athlete entries. Each entry includes a status icon, a photo, and various personal and contact details.

ACTIONS	PHOTO	NOM	PRENOM	NUM CARTE	DATE DE NAISSANCE	GROUPAGE	ADRESSE	EMAIL	N_TELEP
<input checked="" type="checkbox"/> <input type="checkbox"/>		kaddour	Mohamed djibril	346587236587236598	1998-08-01	O+	tamanrasset	mohamed@gmail.com	066840
<input checked="" type="checkbox"/> <input type="checkbox"/>		heddaoui	salem	278364782465783659	1996-07-27	O+	tamanrasset	salem@gmail.com	079265
<input checked="" type="checkbox"/> <input type="checkbox"/>		djafour	oussama	782367836594782648	2000-08-18	AB+	tamanrasset	oussama@gmail.com	380466
<input checked="" type="checkbox"/> <input type="checkbox"/>		benchikh	alaa	475678936537896589	2001-10-18	B+	tamanrasset	alaa@gmail.com	734782
<input checked="" type="checkbox"/> <input type="checkbox"/>		houtia	yassine	756783657893647825	2002-10-27	B+	adrar	houtia@gmail.com	645763
<input checked="" type="checkbox"/> <input type="checkbox"/>		debabi	sifo	374637856789236578	1998-06-27	AB+	bejaia	sifo@gmail.com	465236
<input checked="" type="checkbox"/> <input type="checkbox"/>		saad	saad	263764823648254136	2000-06-27	AB+	tamanrasset	saad@gmail.com	374856
<input checked="" type="checkbox"/> <input type="checkbox"/>		wyetw	wqjueyfwie	783467863487625345	2001-06-27	AB+	bejaia	wgrfyu@gmail.com	456782

Figure 4. 7:L'interface (Demande).

4.5.4 - 2. Compétition :

Le deuxième scénario consiste à créer des compétitions.

Dans cette partie, nous allons montrer les différentes étapes pour créer des compétitions.

➤ Epreuve :

- L'interface (Création) :

Sur cette interface, la première étape montre que l'administrateur crée les épreuves.

The screenshot shows the 'Création (épreuve)' (Event Creation) interface. The form is titled 'AJOUT UN EPREUVE' and includes several input fields and buttons. The 'Type de l'épreuve' (Event Type) is set to 'Individual'. The 'Nom epreuve' (Event Name) is 'Epreuve1'. The 'Distance (M)' is '200', and the 'Type de Nage' (Swimming Style) is 'dos'. The 'Categorie' (Category) is 'Senior' and the 'Genre' (Gender) is 'Homme'. There are 'Enregistrer' (Register) and 'Annuler' (Cancel) buttons at the bottom.

Figure 4. 8:L'interface Création (épreuve).

- L'interface (épreuves):

Dans cette interface, une liste des épreuves créées est affichée.

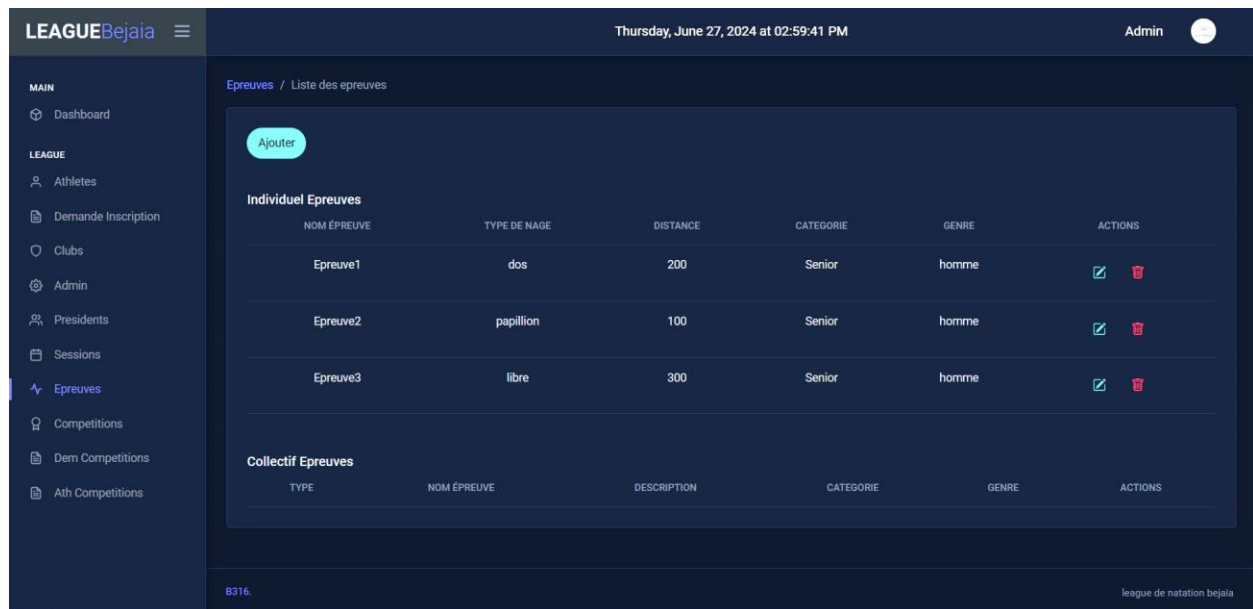


Figure 4. 9:L'interface (épreuves).

➤ **Compétition :**

- **L'interface (Création) :**

Sur cette interface, la deuxième étape montre que l'administrateur crée les compétitions. Lors de la création d'une compétition, l'administrateur entre le nombre de sessions de cette compétition.

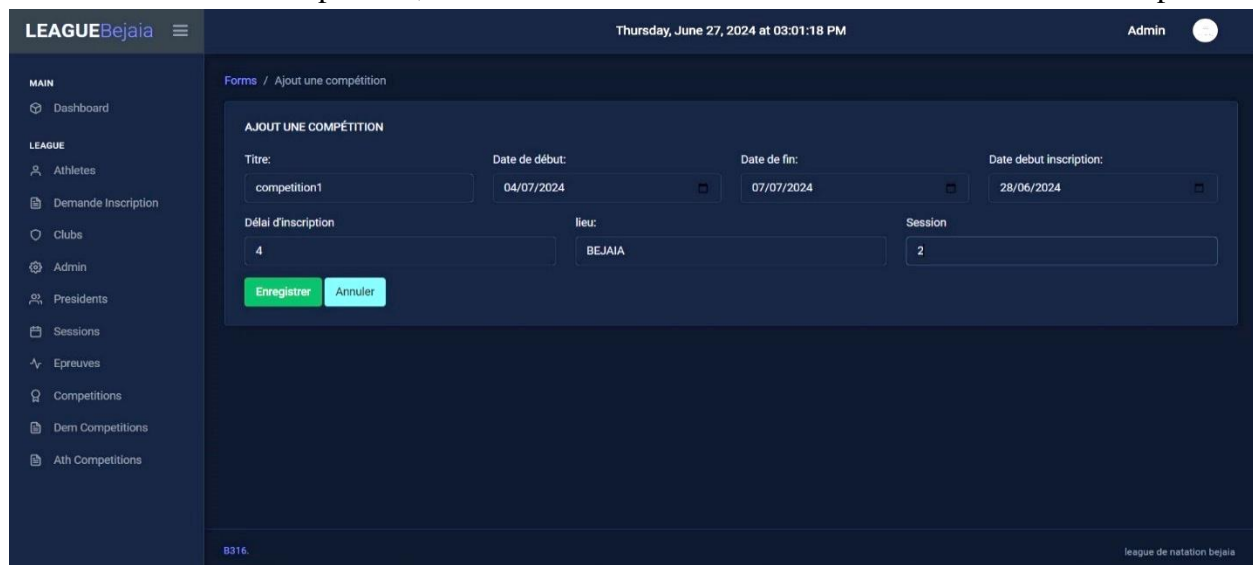


Figure 4. 10:L'interface (Création compétition).

- **L'interface (Compétition vide) :**

Après avoir créé la compétition et entré le nombre de sessions lors de la création, l'application affiche dans cette interface les compétitions créées, vides, avec seulement les sessions et sans les épreuves.

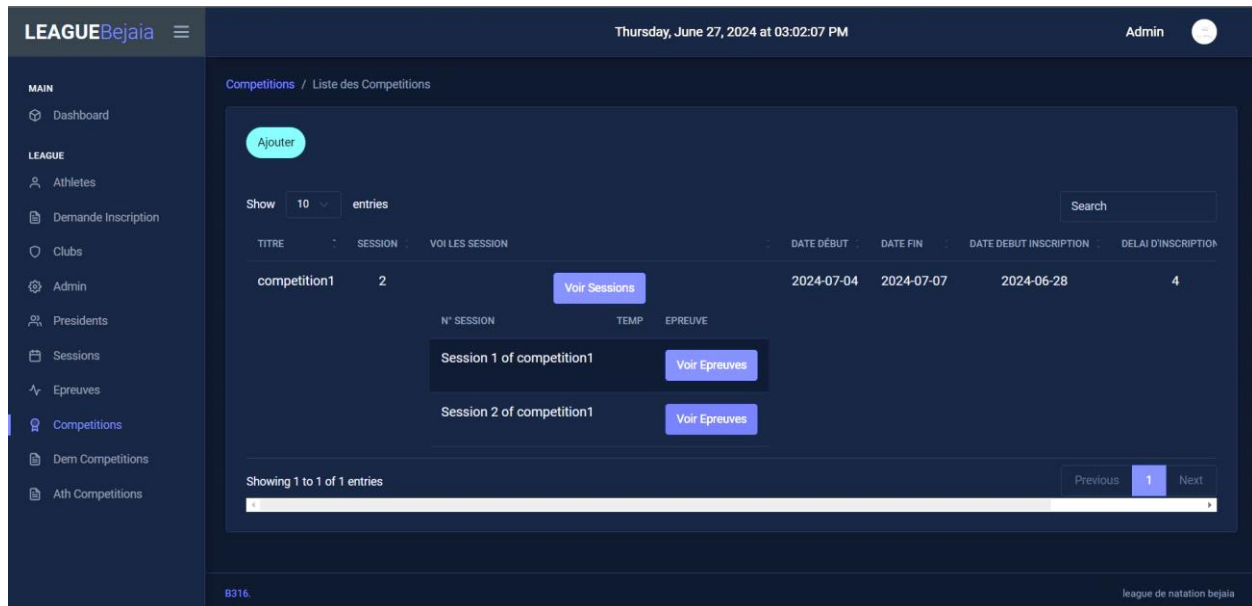


Figure 4. 11:L'interface (Compétition vide).

➤ Session :

Dans la troisième étape, l'administrateur ajoute des épreuves à la session

- **L'interface (Session):** Cette interface montre des sessions vides sans épreuves, car celles-ci sont créées automatiquement lors de la création des compétitions

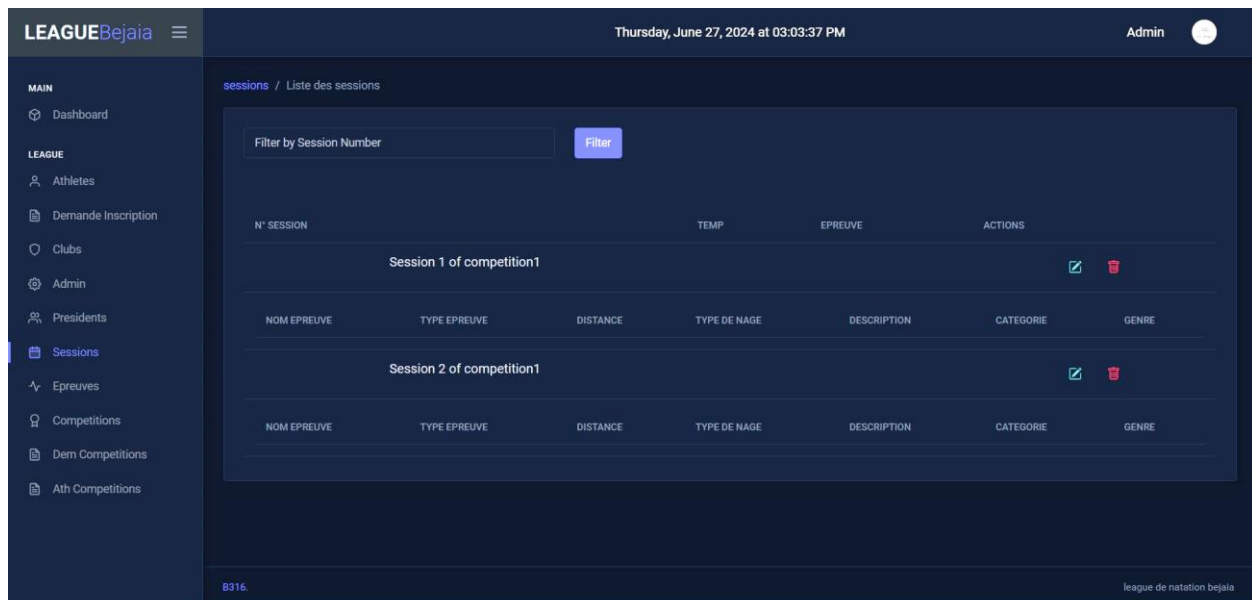


Figure 4. 12:L'interface (Session).

- **L'interface (Ajouter des épreuve) :**

Cette interface montre comment l'administrateur ajoute des épreuves à une session.

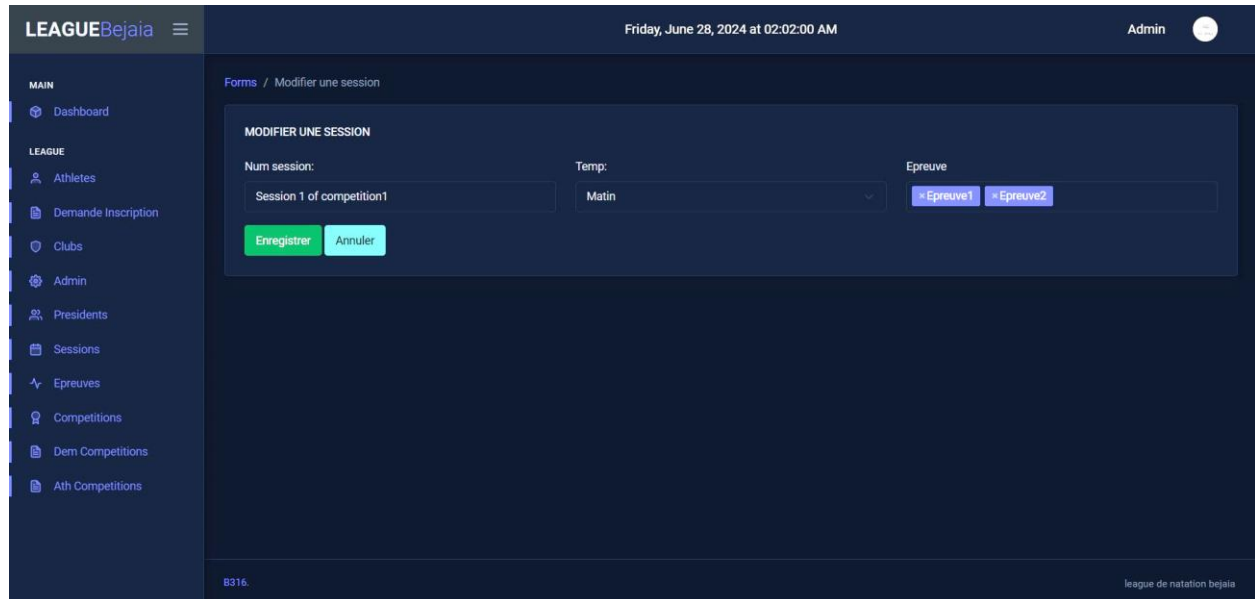


Figure 4. 13:L'interface (Ajouter des épreuve).

- **L'interface (Session avec épreuves) :**

Après avoir ajouté les épreuves, l'application retourne à l'interface de la session, cette fois-ci avec les épreuves ajoutées

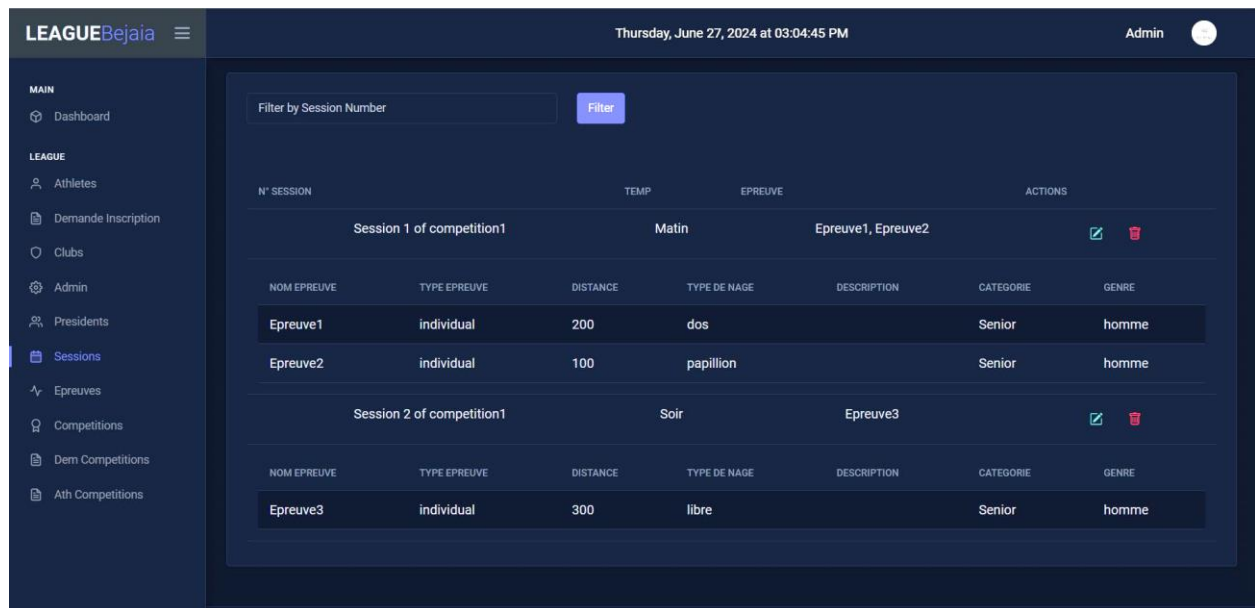


Figure 4. 14:L'interface (Session avec épreuves).

- **L'interface (Compétition):**

À la dernière étape, l'accès à l'interface des compétitions affiche toutes les informations, y compris les sessions et les épreuves.

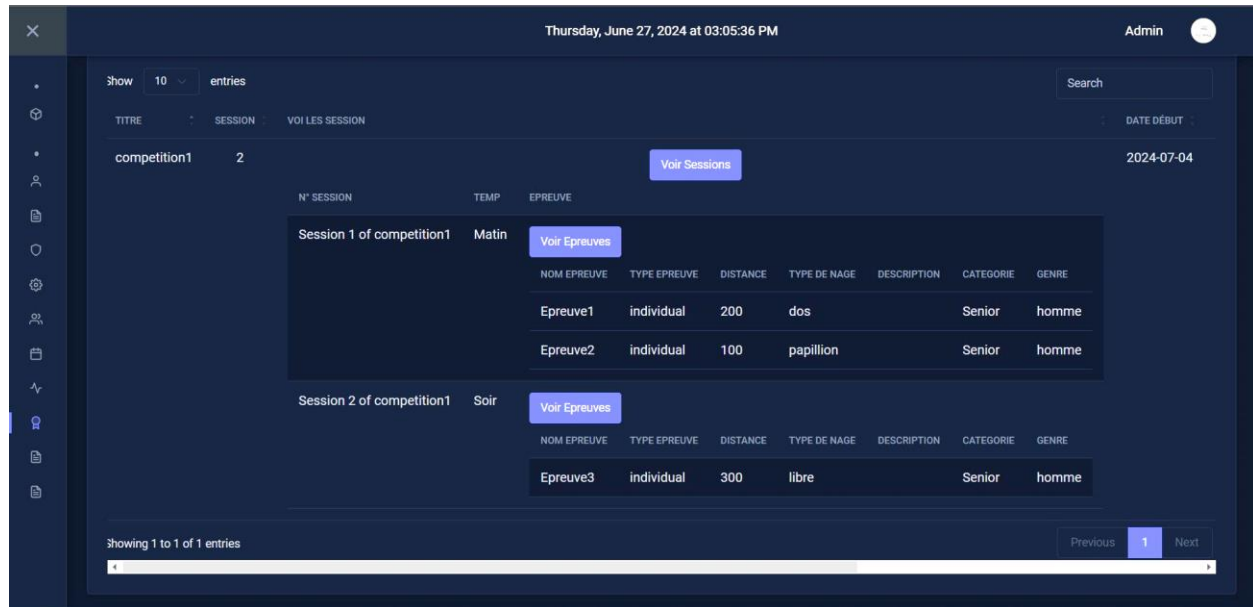


Figure 4. 15:L'interface (Compétition).

➤ Participation :

Le troisième scénario concerne la participation des athlètes dans les compétitions

- **L'interface "Compétition"** : Après la création des compétitions, l'application affiche celles-ci dans la section réservée aux clubs, où ils peuvent inscrire des athlètes pour participer à la compétition.

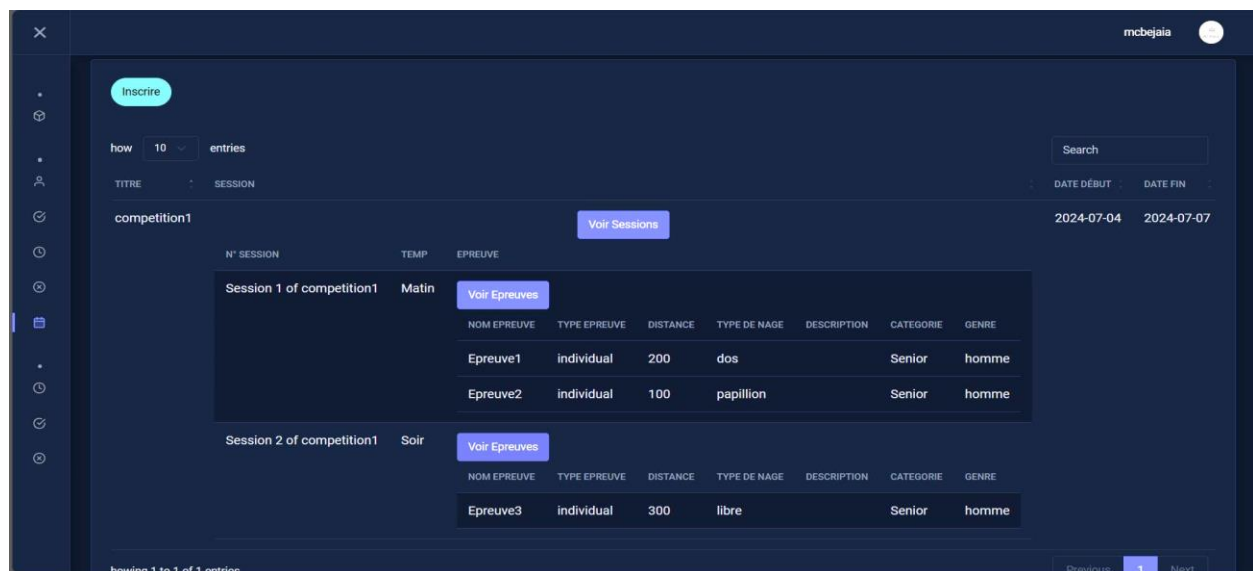


Figure 4. 16:L'interface "Compétition" : Après la création des compétitions.

- **L'interface "Participation"** : Cette interface permet aux clubs de sélectionner la compétition et les athlètes souhaités pour participer, avec les épreuves et les temps d'engagement pour chaque athlète.

Figure 4. 17: L'interface "Participation".

- **L'interface "Demande"** : Les athlètes choisis par les clubs sont affichés dans cette interface, où l'administrateur peut accepter ou refuser leur participation.

TITRE COMPETITION	EPREUVE	ATHLETE	TEMP D'ENGAGEMENT	VALIDATION	ACTION
competition1	Epreuve1	heddaoui	00:00:05	en attente	<input checked="" type="checkbox"/> <input type="checkbox"/>
competition1	Epreuve2	heddaoui	00:00:02	en attente	<input checked="" type="checkbox"/> <input type="checkbox"/>
competition1	Epreuve3	heddaoui	00:00:07	en attente	<input checked="" type="checkbox"/> <input type="checkbox"/>

Showing 1 to 3 of 3 entries

Previous 1 Next

Figure 4. 18: L'interface "Demande".

- **L'interface "Athlètes"** : Cette interface affiche les athlètes acceptés et inscrits pour participer à la compétition.

Thursday, June 27, 2024 at 04:39:45 PM Admin

Athlètes compétition / Liste des athlètes compétition

competition1 Rechercher... Imprimer Voir Liste de Départ

TITRE COMPETITION	CLUB	ATHLETE	EPREUVE	TEMP D'ENGAGEMENT	VALIDATION
competition1	mcbejaia	benchikh	Epreuve1	00:00:08	accept
competition1	mcbejaia	benchikh	Epreuve2	00:00:06	accept
competition1	mcbejaia	kaddour	Epreuve1	00:00:05	accept
competition1	mcbejaia	kaddour	Epreuve3	00:00:03	accept
competition1	mcbejaia	kaddour	Epreuve2	00:00:10	accept
competition1	mcbejaia	heddaoui	Epreuve1	00:00:09	accept
competition1	mcbejaia	heddaoui	Epreuve2	00:00:02	accept
competition1	mcbejaia	heddaoui	Epreuve3	00:00:08	accept
TITRE COMPETITION	CLUB	ATHLETE	EPREUVE	TEMP D'ENGAGEMENT	VALIDATION
competition1	mcbejaia	djafour	Epreuve1	00:00:05	accept
competition1	mcbejaia	djafour	Epreuve2	00:00:07	accept

Figure 4. 19:L'interface "Athlètes".

- **L'interface "Liste de Départ"** : Ici, l'administrateur peut facilement afficher la liste de départ qui contient les athlètes avec leurs couloire et toutes les informations nécessaires pour cette compétition.

Thursday, June 27, 2024 at 04:40:54 PM Admin

Imprimer

Epreuve1
Série 1

#	NOM	CLUB	TEMPS D'ENGAGEMENT
1	benchikh	mcbejaia	00:00:08
2	kaddour	mcbejaia	00:00:05
3	heddaoui	mcbejaia	00:00:09
4	djafour	mcbejaia	00:00:05
5	houtia	mobejaia	00:00:04
6	debabi	mobejaia	00:00:09
7	saad	mobejaia	00:00:05
8	wyetw	mobejaia	00:00:08

Epreuve2
Série 1

#	NOM	CLUB	TEMPS D'ENGAGEMENT
1	benchikh	mcbejaia	00:00:06
2	kaddour	mcbejaia	00:00:10
3	heddaoui	mcbejaia	00:00:02

Figure 4. 20:L'interface "Liste de Départ".

4.5.5. Les fichiers exportés :

- **Liste des athletes :**

Chaque club peut imprimer la liste des athlètes de ce club

Print Liste

Liste des Athletes





Photo	Nom	Prenom	Num carte	Date de naissance	Genre	Groupage	Adresse	Catégorie
	kaddour	Mohamed djebri	346587236587236598	1998-08-01	homme	O+	tamanrasset	Senior
	heddaoui	salem	278364782465783659	1996-07-27	homme	O+	tamanrasset	Senior
	djafour	oussama	782367836594782648	2000-08-18	homme	AB+	tamanrasset	Senior
	benchikh	alaa	475678936537896589	2001-10-18	homme	B+	tamanrasset	Senior

Figure 4. 21:Liste des athletes.

- **Les Carte :**

La ligue peut imprimer la carte pour chaque athlète

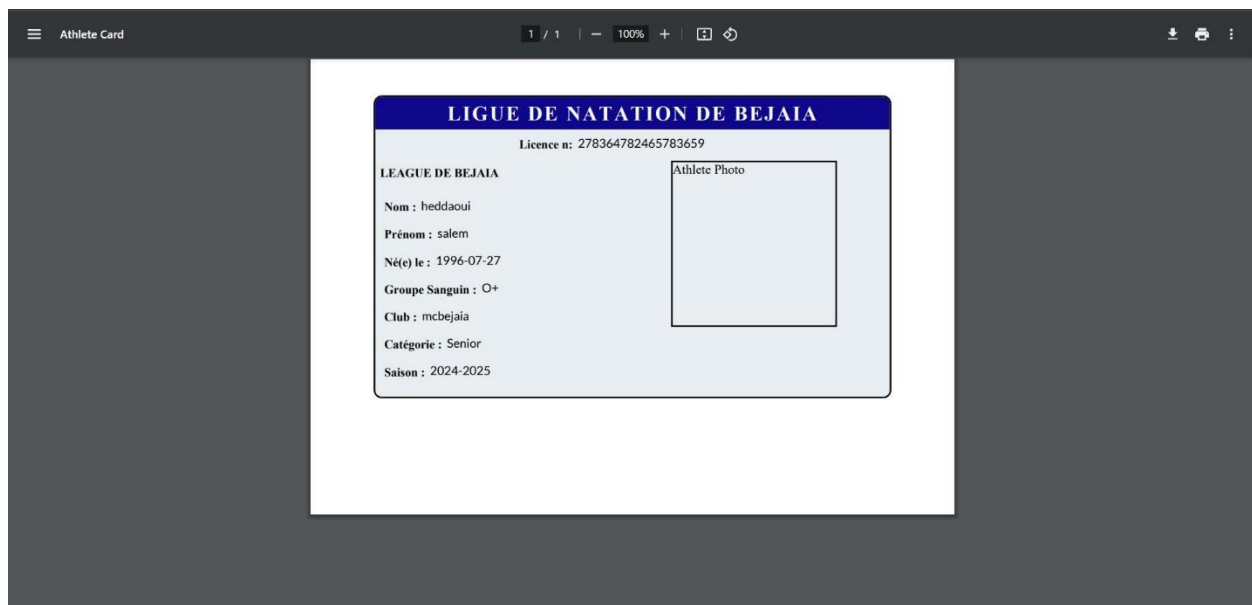


Figure 4. 22:Les Carte

- **Start liste :**

La liste de départ peut être imprimée et utilisée lors de la compétition.

The screenshot displays a 'Start list' for three events. The interface is divided into a main table area and a right-hand print menu.

Table 1: Epreuve1 Série 1

#	NOM	CLUB	TEMPS D'ENGAGEMENT
1	benchikh	mbejaia	00:00:08
2	kaddour	mbejaia	00:00:05
3	heddaoui	mbejaia	00:00:09
4	djafour	mbejaia	00:00:05
5	houtia	mbejaia	00:00:04
6	debabi	mbejaia	00:00:09
7	saad	mbejaia	00:00:05
8	wyethw	mbejaia	00:00:08

Table 2: Epreuve2 Série 1

#	NOM	CLUB	TEMPS D'ENGAGEMENT
1	benchikh	mbejaia	00:00:6
2	kaddour	mbejaia	00:00:10
3	heddaoui	mbejaia	00:00:02
4	djafour	mbejaia	00:00:07
5	houtia	mbejaia	00:00:07
6	debabi	mbejaia	00:00:04
7	saad	mbejaia	00:00:08
8	wyethw	mbejaia	00:00:03

Table 3: Epreuve3 Série 1

#	NOM	CLUB	TEMPS D'ENGAGEMENT
1	kaddour	mbejaia	00:00:03
2	heddaoui	mbejaia	00:00:08
3	djafour	mbejaia	00:00:10
4	debabi	mbejaia	00:00:09

The print menu on the right includes options for 'Destination' (Enregistrer au format Pdf), 'Pages' (Tous), and 'Mise en page' (Portrait). It also features 'Enregistrer' and 'Annuler' buttons.

Figure 4. 23:Start liste.

- **Liste des Athletes participer :**

La ligue peut imprimer la liste des athlètes participant à la compétition.

The screenshot displays a 'Liste des Athletes participer' for a competition. The interface is divided into a main table area and a right-hand print menu.

Table 1: Liste des Athletes participer

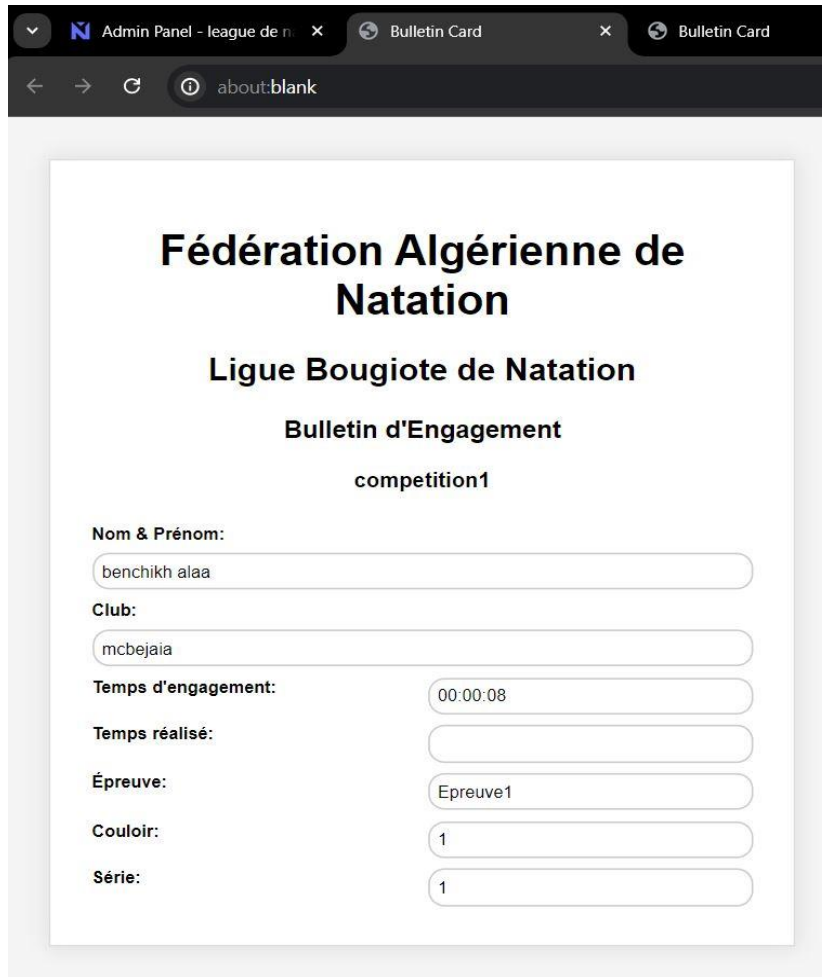
TITRE COMPETITION	CLUB	ATHLETE	EPREUVE	TEMP D'ENGAGEMENT	VALIDATION
competition1	mbejaia	benchikh	Epreuve1	00:00:08	accept
competition1	mbejaia	benchikh	Epreuve2	00:00:6	accept
competition1	mbejaia	kaddour	Epreuve1	00:00:05	accept
competition1	mbejaia	kaddour	Epreuve3	00:00:03	accept
competition1	mbejaia	kaddour	Epreuve2	00:00:10	accept
competition1	mbejaia	heddaoui	Epreuve1	00:00:09	accept
competition1	mbejaia	heddaoui	Epreuve2	00:00:02	accept
competition1	mbejaia	heddaoui	Epreuve3	00:00:08	accept
competition1	mbejaia	djafour	Epreuve1	00:00:05	accept
competition1	mbejaia	djafour	Epreuve2	00:00:07	accept
competition1	mbejaia	djafour	Epreuve3	00:00:10	accept
competition1	mbejaia	houtia	Epreuve1	00:00:04	accept
competition1	mbejaia	houtia	Epreuve2	00:00:07	accept
competition1	mbejaia	debabi	Epreuve1	00:00:09	accept
competition1	mbejaia	debabi	Epreuve2	00:00:04	accept
competition1	mbejaia	debabi	Epreuve3	00:00:09	accept
competition1	mbejaia	saad	Epreuve1	00:00:05	accept
competition1	mbejaia	saad	Epreuve2	00:00:08	accept

The print menu on the right includes options for 'Destination' (Enregistrer au format Pdf), 'Pages' (Tous), and 'Mise en page' (Portrait). It also features 'Enregistrer' and 'Annuler' buttons.

Figure 4. 24:Liste des Athletes participer.

- **Bulletin d'engagement :**

Pour la compétition, après avoir imprimé la liste de départ, la ligue doit imprimer les bulletins d'engagement pour chaque athlète afin d'organiser correctement l'événement et de consigner les résultats de chaque participant.



The image shows a web browser window with two tabs: 'Admin Panel - league de n...' and 'Bulletin Card'. The address bar shows 'about:blank'. The main content is a commitment card for the 'Fédération Algérienne de Natation' and 'Ligue Bougiote de Natation'. The card is titled 'Bulletin d'Engagement' for 'competition1'. It contains a form with the following fields:

Nom & Prénom:	<input type="text" value="benchikh alaa"/>
Club:	<input type="text" value="mcbejaia"/>
Temps d'engagement:	<input type="text" value="00:00:08"/>
Temps réalisé:	<input type="text"/>
Épreuve:	<input type="text" value="Epreuve1"/>
Couloir:	<input type="text" value="1"/>
Série:	<input type="text" value="1"/>

Figure 4. 25:Bulletin d'engagement.

4.6. Conclusion :

Nous avons présenté les différents outils utilisés et l'environnement de développement de notre application. Enfin, nous avons détaillé les interfaces principales ainsi que les scénarios d'utilisation de l'application pour gérer la ligue.

Conclusion générale et perspectives

En conclusion, la mise en place de cette application web a significativement amélioré la gestion administrative de la Ligue de Natation de Béjaïa. Les principaux apports du projet incluent une réduction notable de la paperasse, une gestion plus efficace des inscriptions, des compétitions et des résultats, ainsi qu'une interface conviviale qui simplifie les tâches quotidiennes des gestionnaires.

Les tests effectués ont confirmé la fiabilité et la robustesse de l'application, permettant ainsi une adoption sereine et efficace par les utilisateurs. En résolvant les problèmes majeurs rencontrés auparavant, cette plateforme a non seulement modernisé les processus existants mais a également ouvert la voie à de nouvelles possibilités d'optimisation et d'innovation.

ce projet a démontré qu'une approche technologique bien pensée peut transformer et dynamiser la gestion administrative, offrant ainsi des bénéfices durables et évolutifs à la Ligue de Natation de Béjaïa.

Pour l'avenir, nous envisageons d'intégrer de nouvelles fonctionnalités telles que des modules de formation en ligne pour les athlètes et les entraîneurs, ainsi que des outils d'analyse avancée pour un suivi plus précis des performances. Ces perspectives prometteuses permettront à la Ligue de Natation de Béjaïa de rester à la pointe de la technologie et de continuer à améliorer ses opérations et ses performances.

Bibliographie

- [1] O. LE DEUFF, "Le succès du web 2.0: histoire, techniques et controverse," *Publications, Hermès Science*, 2007.
- [2] "Quelle est la différence entre le web et Internet ?", Available: <https://www.lefigaro.fr/secteur/high-tech/2019/03/11/32001-20190311ARTFIG00004-quelle-est-la-difference-entre-le-web-et-internet.php>. [Accessed 2 Juin 2024].
- [3] T. Berners-Lee, "The World Wide Web: Past, Present and Future," *Information, Journal of Digital*, 1997.
- [4] B. M. Leiner, V. G. Cerf, D. D. Clark, R. E. Kahn, L. Kleinrock, D. C. Lynch, J. Postel, L. G. Roberts and S. Wolff, "A brief history of the internet," *Review, ACM SIGCOMM Computer Communication*, 2009.
- [5] D. R. D. S. James J. Townsend, *Building Portals, Intranets, and Corporate Web Sites Using Microsoft Servers*, Professional, Addison-Wesley ed., 2004.
- [6] J. & C. d. a. b. w. DUCKETT, *HTML & CSS: design and build websites.*, Sons, John Wiley ed., 2011.
- [7] J. DUCKETT, *Javascript and jquery: Interactive front-end web development.*, Sons, John Wiley ed., 2014.
- [8] V. D. W. I. V. J. B. S. Souer J, "Situational requirements engineering for the development of content management system-based web applications," *Technology., International Journal of Web Engineering and*, 01 Jan 2007.
- [9] D. GOURLEY and B. TOTTY, *HTTP: the definitive guide.*, Media, O'Reilly ed., 2002.
- [10] "Site web vs Application web", Available: <https://www.mbiance.com/fr/blogue/32/site-web-vs-application-web>. [Accessed 2 juin 2024].
- [11] "Apprendre la programmation,", Available: apprendre-la-programmation.net. [Accessed 2 juin 2024].
- [12] "Qu'est-ce qu'un framework ?", Available: <https://www.vigicorp.fr/qu-est-ce-qu-un-framework/>. [Accessed 2 juin 2024].
- [13] Facebook: "LIGUE DE NATATION DE BEJAIA". [Accessed 2 juin 2024].
- [14] H.-E. e. a. ERIKSSON, "UML 2 toolkit.," Wiley ed., 2003.

Bibliographie

- [15] G. Booch, "The unified modeling language user guide.," Addison-Wesley ed., 2005.
- [16] G. Richard, "Introduction aux bases de données, modèle relationnel," Paris, Springer-Verlag ed., 1999.
- [17] "Conception et réalisation d'un site web pour le département d'informatique", Available: https://www.memoireonline.com/01/13/6844/m_Conception-et-realisation-dun-site-web-pour-le-departement-dinformatique30.html. [Accessed 1 juin 2024].
- [18] P. e. a. LUBBERS, Pro HTML5 programming., Apress ed., 2011.
- [19] D. MCFARLAND, CSS3: the missing manual., Media, O'Reilly ed., 2012.
- [20] D. CROCKFORD, JavaScript: The Good Parts: The Good Parts., O'Reilly Media ed., 2008.
- [21] "Bootstrap", Available: <https://getbootstrap.com/>. [Accessed 3 juin 2024].
- [22] O. Heurtel, "PHP y MySQL., Ediciones Eni. ed., 2014.
- [23] "Développons en Java", Available: <https://www.jmdoudoux.fr/java/dej/chap-tomcat.htm>. [Accessed 3 juin 2024].
- [24] "Fonctionnalités de développement Visual Studio", Available: <https://visualstudio.microsoft.com/fr/vs/features/>. [Accessed 3 juin 2024].
- [25] "XAMPP", Available: Manuel d'aide fourni avec le logiciel XAMPP. [Accessed 3 juin 2024].
- [26] D. Spinellis, "'Git.," *IEEE software*, pp. 100-101, 2012.
- [27] M. STAUFFER, Laravel: Up & Running., o'reilly ed., 2023.
- [28] " Introduction to Node.js ". Available: <https://nodejs.org/en/learn/getting-started/introduction-to-nodejs>. [Accessed 3 juin 2024]

Résumé

Ce travail se concentre sur le développement et l'implémentation d'une **application web** de gestion pour la **ligue de natation de Béjaïa**. L'objectif principal est de fournir aux administrateurs, agents des clubs et présidents un outil centralisé pour gérer les compétitions, les inscriptions, les résultats des nageurs, ainsi que la gestion des athlètes et des clubs. L'application vise à simplifier et moderniser les processus de gestion grâce à une interface conviviale et des fonctionnalités automatisées. Elle a été développée en utilisant **Laravel** comme framework principal et **PHP** pour la logique de serveur. La modélisation **UML** a été utilisée pour structurer le projet, et **MySQL** pour la base de données. Cette approche méthodique a permis de créer une application efficace et adaptable, améliorant la communication entre les parties prenantes et facilitant l'organisation des événements.

Mots clés : ligue de natation de Béjaïa, application web, Laravel, UML, PHP, MySQL.

Abstract:

This work focuses on the development and implementation of a **web application** for the management of the **swimming league of Béjaïa**. The main objective is to provide administrators, club agents, and presidents with a centralized tool to manage competitions, registrations, swimmers' results, as well as the management of athletes and clubs. The application aims to simplify and modernize management processes through a user-friendly interface and automated features. It was developed using **Laravel** as the main framework and **PHP** for server logic. **UML** modeling was used to structure the project, and **MySQL** for the database. This methodical approach enabled the creation of an efficient and adaptable application, improving communication between stakeholders and facilitating the organization of events.

Keywords: Béjaïa swimming league, web application, Laravel, UML, PHP, MySQL.