



**MINISTERE DE L'ENSEIGNEMENT SUPERIEURET DE LA  
RECHERCHE SCIENTIFIQUE  
DIRECTION GENERALE DES ETUDES TECHNOLOGIQUES**



Institut Supérieur des Etudes Technologiques de Djerba  
Département TECHNOLOGIES de l'Informatique

**Rapport de  
STAGE DE FIN D'ETUDES**

Présenté en vue de l'obtention du diplôme de  
Licence Appliquée en TECHNOLOGIES de  
l'Informatique

**Parcours : DSI**

**CONCEPTION ET DEVELOPEMENT D'UNE APPLICATION WEB  
POUR LA GESTION D'ASSOCIATION**

Elaboré par :

Sekouba Doumbouya  
Kaba Condé

Encadré par :

M. Mohamed Raouf Ourir

Effectué à :

Entreprise : Association de Djerba pour la Science et la Technologie

Encadrant : M. Mejdi Blaghi

**Année Universitaire : 2023/2024**

# Dédicaces

*Ce projet de fin d'étude est dédié*

*A nos très chers parents, qui nous ont toujours soutenu tant financièrement que moralement dans nos études, qui nous ont aussi boosté et motivé quand on doutait de nos capacités à y arriver, qui sans relâche ont prié jour et nuit pour notre réussite, sans eux on ne saura atteindre cet objectif.*

*A nos très chères mères qui s'est donné corps et âme afin qu'on ne manque de rien et qu'on puisse étudier dans les meilleures conditions.*

*A nos très chers pères qui depuis tout petit ont su nous imprégner une éducation de qualité tant au niveau professionnel que celui de la vie en communauté.*

*A nos très chers frères et sœurs ainsi qu'à nos professeurs de l'ISE7 Djerba, plus particulièrement à ceux du département informatique merci pour vos encouragements, assistances et soutiens inconditionnels.*

*Avec tout cet amour dont vous nous avez comblé chers papas, chères mamans, recevez ce projet de fin d'étude en guise de fruit de votre dur labeur qui représente l'aboutissement du soutien et des encouragements dont vous nous avez prodigués.*

*Sekouba Doumbouya*

*Kaba Conde*

## REMERCIEMENTS

Nous tenons avant toute chose à exprimer notre profonde gratitude à notre cher professeur et encadrant M. Mohamed Raouf Ourir pour son suivi, son énorme soutien et ses conseils qu'il nous a prodigués tout au long de ce stage. Nous tenons également à remercier notre encadrant M. Mejdi Blaghi pour le temps et pour les précieux conseils qu'il nous a prodigués avec intérêt et compréhension.

Enfin, Nous remercions toutes ces personnes qui ont fait de ce stage une réussite.

# Sommaire

<b>INTRODUCTION GENERALE.....</b>	<b>2</b>
<b>CHAPITRE 1 : ETUDE PREALABLE .....</b>	<b>3</b>
INTRODUCTION .....	3
1.1 CADRE DU TRAVAIL .....	3
1.2 PRESENTATION DE L'ORGANISME D'ACCUEIL .....	3
1.2.1 Organigramme de l'entreprise d'accueil .....	4
1.3 ETUDE DE L'EXISTANT .....	5
1.3.1 Gestions des adhésions.....	5
1.3.2 Gestion des bénéficiaires.....	5
1.3.3 Gestion des actualités .....	6
1.3.4 Gestions des revenus et des dépenses .....	6
1.3.5 Gestion des sessions de formations .....	7
1.3.6 Gestions des événements.....	8
1.4. EVALUATION DE L'EXISTANT .....	8
1.5 SOLUTION PROPOSEE .....	9
1.6 CHOIX TECHNOLOGIQUES.....	9
1.7 CHOIX METHODOLOGIQUE .....	9
1.7.1 Unified Modeling Language .....	9
1.7.2 Scrum .....	10
CONCLUSION.....	12
<b>CHAPITRE 2 : ANALYSES ET SPECIFICATIONS DES BESOINS.....</b>	<b>13</b>
INTRODUCTION .....	13
2.1 SPECIFICATION DES BESOINS .....	13
2.1.1 Besoins fonctionnels .....	13
2.1.2. Besoins non fonctionnels .....	13

2.2 PRODUCT BACKLOG .....	14
2.2.1 Identification des acteurs.....	14
2.2.2 Eléments du Product Backlog .....	15
2.2.3 Planning des sprints.....	17
CONCLUSION.....	18
<b>CHAPITRE 3 : ETUDE CONCEPTUELLE.....</b>	<b>19</b>
INTRODUCTION .....	19
3.1 MODELISATION DYNAMIQUE.....	19
3.1.1 Diagramme de séquences .....	19
3.1.1.1 Composants et symboles élémentaire .....	19
3.1.2 Diagramme de séquence « Ajout d’une actualité».....	22
3.1.3 Diagramme de séquence « Ajout d’un événement».....	24
3.1.4 Diagramme de séquence « Ajout d’une session de formation» .....	24
3.1.5 Diagrammes d'activité.....	25
3.2 MODELISATION STATIQUE.....	29
3.2.1 Diagramme de cas d’utilisation global.....	29
3.2.2 Description détaillée du cas d’utilisation « S’authentifier ».....	29
3.2.3 Description détaillée du cas d’utilisation « Affecter rôles» .....	30
3.2.4 Description détaillée du cas « Supprimer utilisateur » .....	31
3.2.2 Diagramme de classes global .....	31
.....	32
CONCLUSION.....	33
<b>CHAPITRE 4 : REALISATIONS .....</b>	<b>34</b>
INTRODUCTION .....	34
4.1 SPRINT 0 : MODULE DES OUTILS DE DEVELOPPEMENT .....	34
4.1.1 ENVIRONNEMENT DE TRAVAIL .....	34

4.1.2 Environnement matériel .....	35
4.1.3 Environnement logiciel .....	35
4.1.3.1 Environnements de développement.....	35
4.2.2.2 Framework de développement .....	36
4.2.2.3FrameworkBack end.....	36
4.2.2.4 Framework Front end .....	37
4.2.3 Outils divers .....	38
4.2 SPRINT 1 : MODULE DES COMPTES UTILISATEURS .....	39
4.3 SPRINT 2 : MODULE DE GESTION DES SESSIONS DE FORMATIONS.....	39
4.4 SPRINT 3 : MODULE DE GESTION DES ACTUALITES .....	40
4.5 SPRINT 4 : MODULE DE GESTION DES EVENEMENTS.....	41
4.6 SPRINT 5 : MODULE DE GESTION DES DEMANDES ADHESIONS .....	41
4.7 SPRINT 6 : MODULE DE GESTION DES DEPENSES .....	42
4.8 SPRINT 7 : MODULE DE GESTION DES DONs .....	42
<b>CHAPITRE 5 : PRESENTATION DE L'APPLICATION.....</b>	<b>44</b>
INTRODUCTION .....	44
5.1 QUELQUES INTERFACES DE L'APPLICATION.....	44
5.1.1 Page d'authentification .....	44
5.1.2 Page de création de compte pour un membre de l'association .....	45
5.2 AUTRES CARACTERISTIQUES PARTICULIERES DE L'APPLICATION.....	47
5.2.1 Persistance des données .....	47
5.2.2 Sécurité.....	49
5.2.3 StarUML : .....	50
5.2.4 Draw.io.....	50
CONCLUSION.....	51
<b>CONCLUSION GENERALE ET PERSPECTIVES .....</b>	<b>52</b>

### Liste des figures

<b>FIGURE 1 ORGANIGRAMME DE L'ENTREPRISE D'ACCUEIL.....</b>	<b>4</b>
<b>FIGURE 2 STRUCTURE DE SCRUM .....</b>	<b>11</b>
<b>FIGURE 3 DIAGRAMME DE SEQUENCE « S’AUTHENTIFIER » .....</b>	<b>22</b>
<b>FIGURE 4 DIAGRAMME DE SEQUENCE «AJOUT ACTUALITES » .....</b>	<b>23</b>
<b>FIGURE 5 DIAGRAMME DE SEQUENCE « AJOUT EVENEMENT» .....</b>	<b>24</b>
<b>FIGURE 6 DIAGRAMME DE SEQUENCE « AJOUT SESSION FORMATION ».....</b>	<b>25</b>
<b>FIGURE 7 DIAGRAMME D’ACTIVITE « AUTHENTIFICATION ».....</b>	<b>27</b>
<b>FIGURE 8 DIAGRAMME D’ACTIVITE « AJOUT D’UNE SESSION DE FORMATION » .....</b>	<b>28</b>
<b>FIGURE 9 DIAGRAMME D’ACTIVITE « AJOUT D’UN EVENEMENT».....</b>	<b>28</b>
<b>.....</b>	<b>29</b>
<b>FIGURE 10 DIAGRAMME DE CAS D’UTILISATION GLOBAL.....</b>	<b>29</b>
<b>FIGURE 11 DIAGRAMME DE CLASSES GLOBAL .....</b>	<b>32</b>
<b>FIGURE 12 : PAGE D'AUTHENTIFICATION.....</b>	<b>45</b>
<b>FIGURE 13 : PAGE D'INSCRIPTION .....</b>	<b>45</b>
<b>FIGURE 14 : PAGE D'ACTUALITES POUR LES UTILISATEURS .....</b>	<b>46</b>

<b>FIGURE 15 : GESTION DES SESSIONS DE FORMATION .....</b>	<b>47</b>
<b>FIGURE 16 : MONGO DB .....</b>	<b>49</b>
<b>FIGURE 17 BCrypt.....</b>	<b>50</b>
<b>FIGURE 18 : STARUML .....</b>	<b>50</b>
<b>FIGURE 19 : DRAW-IO .....</b>	<b>50</b>



## Liste des tableaux

<b>TABLEAU 1 PRODUCT BACKLOG.....</b>	<b>16</b>
<b>TABLEAU 2 SPRINT .....</b>	<b>17</b>
<b>SYMBOLE DE MESSAGES ASYNCHRONES : .....</b>	<b>21</b>
<b>DIAGRAMME DE SEQUENCE « S’AUTHENTIFIER » .....</b>	<b>21</b>
<b>TABLEAU 6 EDITEUR VISUEL STUDIO CODE .....</b>	<b>36</b>
<b>TABLEAU 7 LOGICIELS DIVERS .....</b>	<b>36</b>
<b>TABLEAU 8 FRAMEWORK BACK END  NODE JS.....</b>	<b>37</b>
<b>TABLEAU 9 FRAMEWORK FRONT END REACT JS BOOTSTRAP .....</b>	<b>38</b>
<b>TABLEAU 10 OUTILS DIVERS.....</b>	<b>39</b>

# Introduction Générale

Pour nous, étudiants en Licence Appliquée en Technologies de l'Informatique, parcours Développement des Systèmes d'Information (DSI), le stage de fin d'études revêt une signification particulière. C'est une étape charnière où théorie et pratique se rejoignent, où les connaissances acquises au cours de notre formation prennent forme dans des projets concrets au sein d'une entreprise ou d'une organisation. C'est un moment où nous avons l'opportunité tant attendu de mettre en pratique nos compétences et de découvrir le monde professionnel du développement logiciel.

Au sein de cette entreprise spécialisée dans le développement d'applications web sur mesure, notre mission a été des plus stimulantes. Chargés de concevoir et de développer une application web innovante dédiée à la gestion des formations et des paiements, nous avons été confrontés à des défis techniques et conceptuels captivants.

Ce stage nous a permis de plonger au cœur du processus de développement logiciel, de comprendre les enjeux du marché et les attentes des clients, et surtout, de mettre en œuvre nos compétences pour répondre à ces besoins de manière efficace et novatrice.

Dans ce rapport exhaustif, nous avons pris le temps de détailler chaque aspect de notre projet, depuis sa genèse jusqu'à sa réalisation finale. Nous partagerons avec vous le contexte initial qui a donné naissance à notre projet, les objectifs ambitieux que nous nous sommes fixés, les choix technologiques que nous avons opérés, ainsi que les méthodes de développement que nous avons adoptées pour mener à bien notre mission. Notre objectif est de vous offrir une vision complète de notre parcours, de vous immerger dans notre processus de réflexion et d'action, et de vous transmettre les enseignements précieux que nous avons tirés de cette expérience enrichissante.

Nous espérons que ce rapport saura rendre compte du travail accompli avec rigueur et passion, et qu'il saura également inspirer les générations futures d'étudiants en informatique dans leur propre parcours professionnel.

# Chapitre 1 : Etude Préalable

## Introduction

L'étude préalable constitue une étape préliminaire pour la réalisation d'une application. En effet, elle permet d'analyser, d'évaluer et de critiquer le fonctionnement habituel, tout en élaborant la liste des solutions possibles. Dans ce chapitre, nous allons présenter l'entreprise d'accueil, le cadre du travail, la problématique et les objectifs de notre application.

### 1.1 Cadre du travail

Ce projet s'inscrit dans le cadre d'un stage de fin d'études pour l'obtention d'un diplôme de Licence Appliquée en Technologies de l'Informatique parcours Développement des Systèmes d'Information de l'Institut Supérieur des Etudes Technologiques de Djerba.

Ce stage est effectué au sein de l'Association de Djerba pour les Sciences et Technologie (ADST). Nous proposons la conception et le développement d'une application web pour l'association.

### 1.2 Présentation de l'organisme d'accueil

L'Association Djerba pour la Science et la Technologie est une organisation à but non lucratif dédiée à la promotion de l'éducation, de la recherche scientifique, et de l'innovation technologique sur l'île de Djerba.

Fondée avec une vision ambitieuse, elle s'efforce de jouer un rôle clé dans le développement intellectuel et technologique de la région.

L'association compte sur l'engagement actif de ses membres, composés de professionnels de la science et de la technologie, d'éducateurs, d'étudiants et de passionnés.

Elle établit également des partenariats avec des institutions académiques, des entreprises locales et des organisations internationales partageant des valeurs similaires.

### 1.2.1 Organigramme de l'entreprise d'accueil

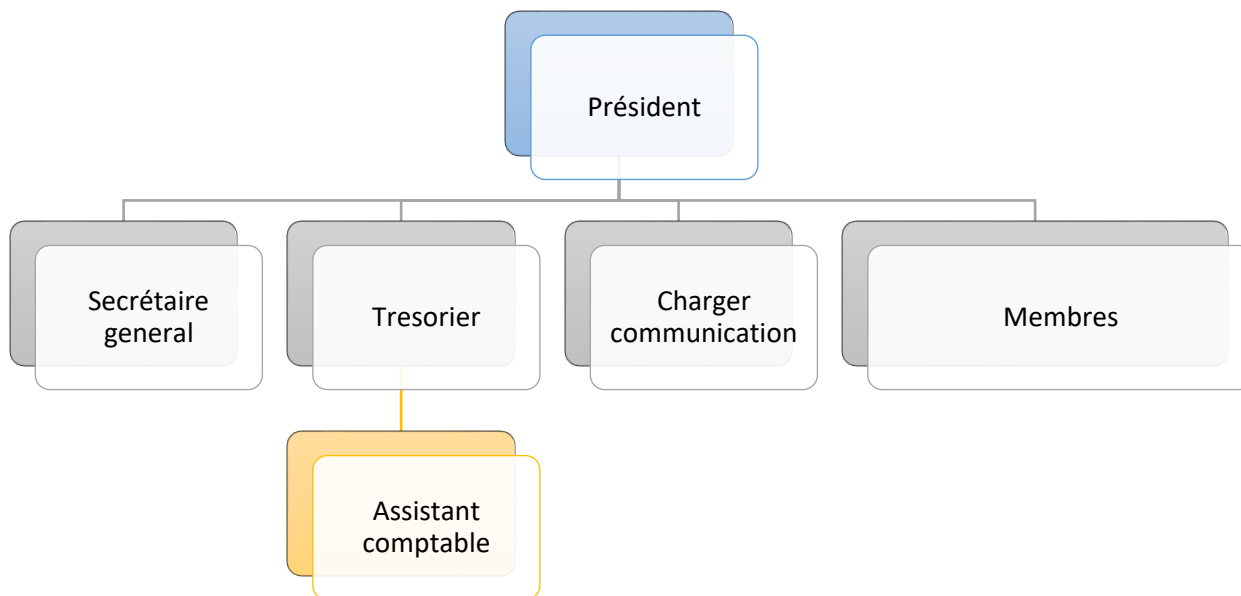


Figure 1 organigramme de l'entreprise d'accueil

Cet organigramme offre une vue détaillée de la structure organisationnelle de l'Association Djerba pour la Science et la Technologie (ADST), en mettant en avant les différents membres et leurs responsabilités au sein de l'organisation.

#### **Conseil d'Administration:**

**Président:** Le président de l'ADST est le chef de l'organisation, chargé de diriger le conseil d'administration, de superviser les activités de l'association et de représenter l'ADST dans les interactions externes.

**Secrétaire Général:** Le secrétaire général assure la gestion administrative de l'ADST, notamment la documentation des réunions, la coordination des communications internes et externes et le suivi des activités administratives.

**Trésorier:** Le trésorier est responsable de la gestion financière de l'ADST, y compris la tenue des comptes, la gestion des budgets, le suivi des dépenses et la préparation des rapports financiers.

#### **Membres:**

Les membres de l'ADST sont des individus ou des représentants d'entreprises qui soutiennent la mission et les objectifs de l'association. Ils contribuent à divers aspects des activités de l'ADST, y compris la participation aux réunions, le soutien aux projets et initiatives, et la représentation de l'ADST dans leurs domaines respectifs.

### **Département de la Communication:**

Chargé de Communication: Le responsable de la communication est chargé de promouvoir les activités et les initiatives de l'ADST auprès du public. Cela inclut la gestion des médias sociaux, la rédaction de communiqués de presse, la création de contenu promotionnel et la coordination des événements de sensibilisation.

### **Département Financier:**

Assistant Comptable: L'assistant comptable soutient le trésorier dans ses responsabilités financières. Cela peut inclure la saisie des données comptables, le suivi des transactions financières et la préparation des documents comptables.

Cet organigramme illustre la structure organisationnelle de l'ADST, mettant en avant les rôles et responsabilités de chaque membre et département. Cette structure permet à l'ADST de fonctionner de manière efficace et de poursuivre sa mission de promotion de la science et de la technologie à Djerba.

## **1.3 Etude de l'existant**

Avant d'entamer la conception et le développement de notre application web dédiée à la gestion des formations et des paiements, il était essentiel de procéder à une analyse approfondie de l'existant. Cette étape nous a permis de comprendre les lacunes et les besoins du système actuel, ainsi que d'identifier les fonctionnalités à améliorer ou à intégrer dans notre solution.

Cette étude de l'existant nous fournira une base solide pour la conception de notre application, en nous permettant de cibler les besoins réels des utilisateurs et d'identifier les axes d'amélioration nécessaires pour optimiser la gestion des formations et des paiements au sein de l'organisation.

Pour cela nous commençons par la description des différents processus de travaux mener à l'ADST.

### **1.3.1 Gestions des adhésions**

La gestion des adhésions concerne le suivi des membres de l'association, de leur statut d'adhérent et de leurs cotisations. Actuellement, cette gestion se fait généralement à l'aide de documents papier ou de feuilles de calcul, ce qui peut entraîner des problèmes de suivi et de gestion. Il est donc essentiel de développer un système de gestion des adhésions qui simplifiera ce processus et assurera un suivi précis du statut des membres et de leurs cotisations.

### **1.3.2 Gestion des bénéficiaires**

Le système actuel de gestion des bénéficiaires repose sur des processus conventionnels, ce qui demande des efforts considérables de la part des administrateurs. Les bénéficiaires sont ajoutés individuellement, et leurs données personnelles telles que les noms, les adresses et les informations de contact sont saisies manuellement dans des feuilles Excel. Cette méthode présente des risques de saisie erronée ou de perte de données, et le suivi des

autorisations d'accès peut devenir complexe à mesure que le nombre de bénéficiaires augmente.

Pour optimiser cette gestion, nous proposons de développer une application web dédiée. Cette application permettra aux administrateurs d'ajouter aisément des bénéficiaires en remplissant des formulaires intuitifs et conviviaux. Les données des bénéficiaires seront stockées de manière sécurisée dans une base de données centralisée, offrant ainsi une meilleure protection des informations sensibles.

De plus, cette application web offrira des fonctionnalités avancées de gestion des autorisations d'accès. Les administrateurs pourront définir et modifier les niveaux d'accès des bénéficiaires en fonction de leurs besoins spécifiques. Par exemple, certains bénéficiaires pourraient avoir accès à des ressources restreintes, tandis que d'autres pourraient avoir des privilèges plus étendus.

Pour éviter la répétition des données lorsqu'un utilisateur souhaite participer à plusieurs sessions de formation, notre application permettra de conserver les informations des bénéficiaires de manière permanente. Ainsi, lorsqu'un bénéficiaire s'inscrit à une nouvelle session, ses informations seront automatiquement récupérées à partir de la base de données, évitant ainsi la saisie répétée des mêmes données. Cela simplifiera le processus d'inscription pour les bénéficiaires et réduira la charge de travail des administrateurs.

En automatisant ces processus de gestion des bénéficiaires, notre solution permettra de réduire les erreurs humaines, d'améliorer l'efficacité opérationnelle et de garantir une meilleure conformité aux normes de sécurité des données. En outre, elle offrira une interface conviviale et intuitive, facilitant ainsi la tâche des administrateurs dans leur gestion quotidienne des bénéficiaires.

### 1.3.3 Gestion des actualités

La diffusion d'informations et d'actualités au sein de l'association se fait actuellement par e-mail ou par d'autres moyens de communication. Pour une communication plus efficace, nous proposons d'intégrer un module de gestion des actualités dans notre application, permettant aux administrateurs de publier et de partager des informations importantes avec les membres de l'association.

### 1.3.4 Gestions des revenus et des dépenses

La gestion des paiements pour les sessions de formation, les adhésions et les dons constitue un aspect crucial de la gestion financière de notre association. Actuellement, ces transactions sont généralement gérées de manière manuelle, ce qui peut entraîner des inefficacités et des erreurs dans le processus.

Pour améliorer cette gestion, il est impératif de mettre en place un système de gestion des paiements spécifiquement dédié à ces différentes sources de revenus. Ce système permettra de faciliter la collecte des frais d'inscription, des adhésions et des dons, tout en assurant une traçabilité et une transparence accrues dans les transactions financières de l'association.

Le système de gestion des paiements devrait inclure les fonctionnalités suivantes :

**Inscription en ligne :** Les participants devraient pouvoir s'inscrire aux sessions de formation en ligne, en remplissant un formulaire d'inscription et en effectuant le paiement des frais d'inscription via une plateforme sécurisée. De même, les nouveaux membres devraient pouvoir adhérer et payer leur cotisation en ligne, et les donateurs devraient pouvoir faire des dons facilement.

**Suivi des paiements :** Le système devrait permettre de suivre et de gérer les paiements des participants enregistrés, des adhérents et des donateurs, en fournissant des informations détaillées sur les paiements effectués, les montants payés, les méthodes de paiement utilisées, etc.

**Notifications de paiement :** Le système devrait envoyer automatiquement des notifications de paiement aux participants, adhérents et donateurs pour confirmer la réception de leur paiement et valider leur inscription, adhésion ou donation.

**Gestion des remboursements :** En cas d'annulation de la participation à une session de formation ou d'adhésion, le système devrait permettre de gérer les demandes de remboursement et de traiter les remboursements de manière efficace et transparente.

**Rapports financiers :** Le système devrait fournir des rapports financiers détaillés sur les revenus générés par les sessions de formation, les adhésions et les dons, les paiements en attente, les remboursements effectués, etc., afin de permettre une analyse approfondie de la situation financière de l'association.

**Gestion des dépenses :** En ce qui concerne les dépenses, l'association doit gérer efficacement les coûts liés à l'organisation des sessions de formation, y compris l'achat de matériel tel que des feutres, des supports pédagogiques, etc. Le système devrait permettre de suivre ces dépenses et de générer des rapports détaillés pour une meilleure gestion budgétaire.

En implémentant ce système, notre association pourra améliorer significativement la gestion de ses revenus et de ses dépenses, garantissant ainsi une plus grande efficacité et une meilleure transparence financière.

### 1.3.5 Gestion des sessions de formations

La gestion des sessions de formation consiste à planifier, organiser et suivre les différentes sessions proposées par

l'association. Les détails de chaque session, tels que la date, le lieu, les intervenants et les participants, doivent être gérés de manière efficace pour assurer le bon déroulement des activités de formation. Il est donc nécessaire de développer un système qui permettra de centraliser et de suivre ces informations de manière organisée.

### 1.3.6 Gestions des événements

Enfin, la gestion des événements concerne l'organisation et la coordination des différents événements organisés par l'association, tels que les conférences, les ateliers, les rencontres, etc. Actuellement, cette gestion se fait principalement manuellement, ce qui peut entraîner des difficultés dans la planification et la coordination des événements.

## 1.4. Evaluation de l'existant

### **Diffusion limitée :**

Actuellement, la diffusion des actualités se fait principalement par le biais de supports traditionnels tels que des papiers, des messages par téléphone ou des courriers électroniques.

Cela limite la portée des informations et rend difficile la diffusion rapide et efficace des actualités à tous les membres de l'association.

### **Manque d'interactivité :**

Les supports traditionnels de diffusion des actualités ne permettent pas aux membres de réagir, de commenter ou d'interagir avec les informations diffusées. Cela limite l'engagement des membres et ne favorise pas les échanges au sein de la communauté.

### **Difficulté de mise à jour :**

Avec les supports traditionnels, la mise à jour et la gestion des actualités peuvent être laborieuses et fastidieuses, nécessitant souvent un effort manuel important de la part des responsables de l'association.

### **Accessibilité limitée :**

Les supports traditionnels de diffusion des actualités peuvent limiter l'accessibilité aux informations, notamment pour les membres qui ne peuvent pas facilement accéder aux documents papier ou aux courriers électroniques.

### **Manque de suivi et d'analyse :**

Les supports traditionnels ne permettent pas de suivre efficacement l'engagement des membres avec les actualités diffusées, ni d'analyser les tendances et les préférences en matière de contenu.



## 1.5 Solution proposée

En tenant compte des évaluations de l'existant, on est amené à proposer une solution qui répond aux objectifs et pallier aux lacunes constatées au niveau du processus existant.

On propose alors de concevoir et développer une application web de gestion pour l'association qui assure la disponibilité des informations en tout temps et avec n'importe quel moyen mais aussi une meilleure sécurité en termes d'accès sans oublier une interface ergonomique offrant une bonne expérience utilisateur.

## 1.6 Choix technologiques

Notre projet de gestion d'une association pour la science et la technologie se compose également de deux parties distinctes : la partie back end pour la création des services et la partie front end pour les interfaces utilisateur. Pour ce faire, nous avons opté pour les technologies suivantes :

Partie Back end :

Node.js et Express.js : Nous avons choisi Node.js pour sa capacité à gérer des opérations non bloquantes et à grande échelle. Express.js est un Framework web minimaliste pour Node.js, offrant des fonctionnalités robustes pour la création d'API RESTful.

Partie Front end :

React.js : React.js est un Framework JavaScript populaire et puissant pour la construction d'interfaces utilisateur interactives. Son approche basée sur les composants facilite le développement, la réutilisation et la maintenance du code front end.

## 1.7 Choix méthodologique

La méthodologie est un aperçu de la façon dont est effectuée l'analyse et la conception. Elle définit les techniques ou les procédures utilisées pour identifier et analyser les informations concernant un sujet de recherche spécifique. Pour ce faire, on part d'un énoncé informel (le besoin tel qu'il est exprimé par l'utilisateur).

### 1.7.1 Unified Modeling Language

UML est un langage qui permet de modéliser tout type d'application grâce aux différents diagrammes qu'il propose. Il permet ainsi de visualiser et de documenter notre système afin de répondre à toutes les exigences. Il unifie également les notations nécessaires aux différentes activités d'un processus de développement.

Les principaux diagrammes que nous allons utiliser tout au long de notre démarche sont :

**Le diagramme de cas d'utilisation** : qui permet de modéliser l'interaction entre les acteurs et le système. Chaque cas représente une fonctionnalité qui est déclenchée par une action utilisateur et qui possède une ou plusieurs exécutions réalisables mais un seul état de terminaison.

**Le diagramme de classes** : qui permet de représenter les éléments du système (Classes et Paquetages) ainsi que les associations qui les relient. Il décrit la structure des entités manipulées, Chaque classe possède des attributs (Nom, Type et Valeur) et des méthodes qui sont préférable qu'elles suivent des règles de nommage.

**Le diagramme de séquences** : qui permet de modéliser les actions de l'utilisateur et la réponse du système en mettant l'accent sur la chronologie des opérations.

**Le diagramme d'activité** : permet de mieux visualiser les schémas de procédés, d'identifier les points à améliorer et de mettre en évidence les domaines performants.

### 1.7.2 Scrum

On a choisi le Scrum, la méthodologie Scrum est une approche de gestion de projet agile qui permet aux équipes, de développer des produits de manières itérative et collaborative. Les équipes Scrum travaillent en Sprints, qui sont des périodes de temps déterminées pour accomplir un certain nombre de tâches et produire un livrable. Scrum est : léger, simple à comprendre et difficile à maîtriser. (1)

#### **Acteurs de Scrum Les acteurs de Scrum sont :**

**Le Product Owner** : il travaille en interaction avec l'équipe de développement, qui suit son management. Il est responsable de la gestion du « Product Backlog », ou « carnet de produit » en français. Mon product Owner qui priorise les fonctionnalités à développer ou à corriger, et qui valide les fonctionnalités terminées de l'application n'est d'autre que M. Mejdî.

**Le scrum Master** : est responsable de la compréhension, de l'adhésion et de l'organisation de la méthodologie de Scrum. Il la maîtrise donc parfaitement. Il veille ainsi à ce que ses principes et ses valeurs soient respectés.

Comme M. Mohamed Ourir Raouf qui, aide à améliorer la communication entre nous et cherche à maximiser sa productivité et son savoir-faire.

**L'équipe de développement** : est chargée de transformer les besoins définis par le Product Owner en fonctionnalités utilisables. Cette équipe n'est d'autre que moi.

D'autres termes sont à connaître pour comprendre la méthode de Scrum :

**Le product Backlog (Carnet du produit)** : ce document contient les exigences initiales dressées puis hiérarchisées avec le client en début du projet. Néanmoins il va évoluer tout au long de la durée du projet, en fonction des divers besoins du client.

**Le sprint Backlog (Carnet de sprint) :** en chaque début de sprint, l'équipe définit un but. Puis lors de la réunion de sprint, l'équipe de développement choisit les éléments du carnet à réaliser. L'ensemble de ces éléments constitue alors le sprint Backlog.

**User story :** ce terme désigne les fonctionnalités décrites par le client.

**Stand Up meeting(Scrum) :** c'est une réunion d'avancement organisée de manière quotidienne durant le sprint.

- Dans notre cas nous allons repartir notre projet en trois (3) sprints :
- Module des comptes utilisateurs.
- Module de gestion des équipements.
- Module des réclamations.

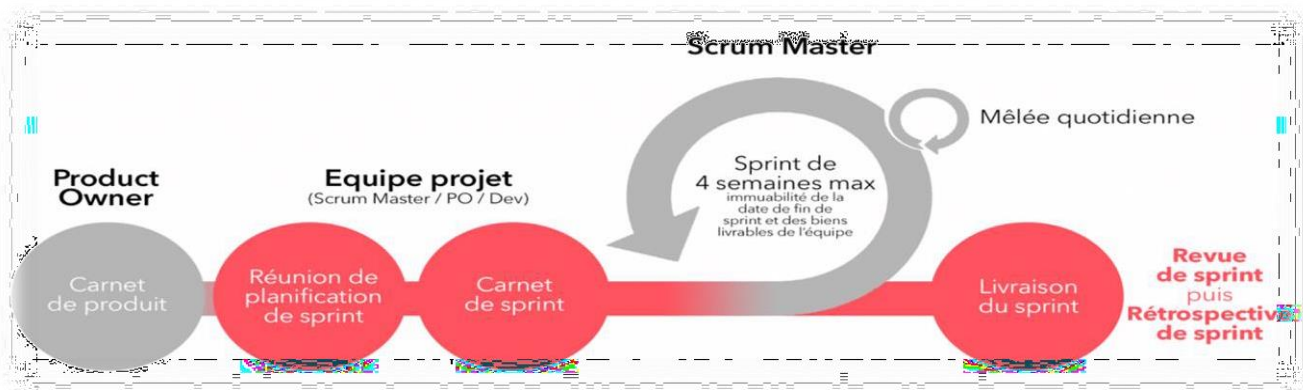


Figure 2 Structure de SCRUM

### Pourquoi utiliser Scrum ?

Scrum est l'une des méthodes Agiles les plus simples à comprendre et à expliquer. Les règles se communiquent facilement. Très structurée, cette approche constitue donc un bon point de départ pour se lancer dans l'utilisation des méthodologies Agile avec une équipe novice.

La méthodologie Scrum implique par ailleurs de constituer une équipe pluridisciplinaire et autonome. Les participants profitent d'une réelle responsabilisation. La plupart gagnent ainsi en motivation et en performance. Scrum place l'humain au centre de la méthodologie.

Les valeurs fondamentales de Scrum

Scrum repose sur 5 valeurs qui sont :

**Focus :** pour suivre le rythme de réalisation, les membres de l'équipe devront être focus pendant le Sprint et à l'accomplissement du Sprint Goal.

**Ouverture :** les membres de l'équipe doivent être ouverts à la collaboration avec d'autres équipes ou personnes

et aux critiques constructifs qui permettent l'amélioration continue.

**Respect :** les membres de l'équipe Scrum se respectent mutuellement et respectent les processus de Scrum.

**Courage :** Devenir autonome n'est pas si facile surtout dans certaines organisations complexes. Il faudra du courage aux équipes scrum pour atteindre cette autonomie.

**Engagement :** les membres de l'équipe Scrum s'engagent à atteindre les objectifs du sprint.

## Conclusion

Tout au long de ce chapitre, nous avons évoqué le contexte du projet, nous avons commencé par une présentation de l'organisme d'accueil qui a été suivi par une étude de l'existant et les critiques, ceci nous a permis de comprendre les besoins et d'envisager la solution aux attentes.

Dans le prochain chapitre, nous allons entamer maintenant la phase de spécifications et de l'analyse des besoins.

# Chapitre 2 : Analyses et spécifications des besoins

## Introduction

On procède dans ce chapitre à la présentation du projet, ses objectifs, ensuite à la spécifications des besoins du cahiers de charges pour mieux comprendre le travail demandé en dégagant les besoins des différents utilisateurs que le système doit accomplir et en fin détailler la planification de celui-ci et sa méthode de développement.

## 2.1 Spécification des besoins

Dans le cadre de notre projet de gestion pour l'association de la science et de la technologie, la spécification des besoins est une étape cruciale pour définir les fonctionnalités nécessaires et les contraintes à respecter. Nous commencerons par présenter les besoins fonctionnels et non fonctionnels de notre application, puis nous identifierons les acteurs impliqués dans son utilisation.

### 2.1.1 Besoins fonctionnels

Les besoins fonctionnels définissent les actions que notre système doit être capable d'effectuer pour répondre aux exigences de l'utilisateur. Notre application comprendra plusieurs modules de gestion permettant des opérations de base telles que la recherche, la création, la lecture, la mise à jour et la suppression (SCRUD). Ces modules incluent notamment :

- ✓ Gestion des membres
- ✓ Gestion des événements
- ✓ Gestion des actualités
- ✓ Gestion des formations
- ✓ Gestion des paiements
- ✓ Gestion des dépenses
- ✓ Gestion des tâches
- ✓ Gestion des paramètres

### 2.1.2. Besoins non fonctionnels

Les besoins non fonctionnels définissent les contraintes et les exigences de performance que notre système doit respecter. Voici les principaux besoins non fonctionnels pour notre application :

**Fiabilité** : L'application doit répondre correctement aux besoins fonctionnels lors de son utilisation.

**Disponibilité** : L'application doit être accessible à tout moment et utilisable par n'importe quel utilisateur.

**Modularité** : Le code de l'application doit être simple, facile à maintenir et à comprendre.

**Extensibilité** : L'application doit être conçue de manière à pouvoir être étendue et évoluer dans le temps.

**Ergonomie de l'interface** : L'interface de l'application doit être conviviale et claire pour les utilisateurs.

**Sécurité** : L'application doit garantir la sécurité des informations personnelles stockées.

## 2.2 Product Backlog

Il s'agit de l'élément clé de tout projet utilisant la méthode Scrum, Le Backlog produit ou Backlog scrum doit être défini et entretenu soigneusement. Que doit-il contenir, comment le construire ? Nous allons essayer d'apporter des réponses à ces questions dans une définition Backlog simple.

### 2.2.1 Identification des acteurs

Cette section a pour objectif de présenter les acteurs et les fonctionnalités auxquelles doit répondre cette application.

Un acteur représente l'abstraction d'un rôle joué par des entités externes (utilisateur, dispositif matériel ou autre système) qui interagissent directement avec le système.

Les acteurs impliqués dans l'utilisation de notre application sont les suivants :

**Administrateur** : Responsable de la gestion globale de l'association.

- ✓ Gestion des membres
- ✓ Gestion des événements
- ✓ Gestion des actualités
- ✓ Gestion des formations
- ✓ Gestion des paiements
- ✓ Gestion des dépenses
- ✓ Gestion des tâches
- ✓ Gestion des paramètres

User : Utilisateur de l'application

Un utilisateur de l'application est une personne qui utilise l'application de gestion de l'association après avoir créé son propre compte.

Les actions qu'il peut effectuer dépendent des autorisations accordées par l'administrateur de l'association.

L'utilisateur peut généralement accéder aux fonctionnalités suivantes :

Créer et gérer son compte utilisateur (profil, informations personnelles, etc.).

Accéder aux fonctionnalités et aux données qui lui sont autorisées en fonction de son rôle ou de ses permissions définis par l'administrateur.

Exécuter les tâches qui lui sont assignées par l'administrateur ou par d'autres membres de l'association.

Collaborer avec d'autres membres de l'association en utilisant les fonctionnalités de communication et de collaboration de l'application.

Consulter les informations relatives aux membres, aux événements, aux actualités, aux formations, aux paiements, aux dépenses, aux tâches et aux paramètres, en fonction de ses autorisations

### 2.2.2 Eléments du Product Backlog

Dans cette partie, nous présenterons chaque « sprint » avec ses différentes « fonctionnalités » en indiquant les « priorités » selon leurs importances et les « users story » chacune avec son « ID ».

<b>Module des comptes utilisateurs</b>	Gestions des comptes utilisateurs	En tant qu'admin, je peux affecter des rôles à l'utilisateur et archiver ou les désarchiver. Voir les détails de chaque utilisateur.	<b>3</b>
		En tant qu'adhérent je peux créer mon compte aussi, je peux consulter mon profil et le modifier.	
<b>Module de gestion des sessions de formation</b>	Gestion des sessions	En tant qu'admin, je peux créer des sessions de formation, Supprimer, modifier, consulter le détail de chaque session.	
		En tant qu'adhérent ou user, je peux voir la liste des sessions disponible.	
<b>Module de gestion des demandes adhésions</b>	Gestion des Adhésions	En tant qu'admin je peux consulter la liste des adhésions pour pouvoir accepter une adhésion.	
		En tant qu'adhérent je peux passer une demande adhésion.	
<b>Module de gestion des inscriptions aux sessions de formation</b>	Gestion des inscriptions aux sessions de formation	En tant qu'admin je peux consulter la liste des adhésions pour pouvoir accepter une adhésion.	<b>3</b>
		En tant qu'adhérent, je peux m'inscrire à une session de formation.	
<b>Module de gestion des actualités</b>	Gestion des actualités	En tant qu'admin je peux créer, publier, modifier et Supprimer des actualités.	<b>3</b>
		En tant qu'adhérent, je peux consulter les actualités.	
<b>Module de gestion des évènements</b>	Gestion des évènements	En tant qu'admin je peux créer, modifier et Supprimer des évènements.	
		En tant qu'adhérent, je peux consulter, commenter des événements.	

**Tableau 1 Product Backlog**



### 2.2.3 Planning des sprints

Nous avons choisi de répartir le projet en 5 sprints, de sorte que chaque sprint sera développé en module d'application. Ce tableau indique comment la division est faite.

Sprints	Module	Durée
<b>Sprint 0</b>	Module des outils de développement	<b>4 semaines.</b>
<b>Sprint 1</b>	Module des comptes utilisateurs	<b>4 semaines.</b>
<b>Sprint 2</b>	Module de gestion des sessions de formations	<b>5 semaines.</b>
<b>Sprint 3</b>	Module de gestion des actualités	<b>3 semaines.</b>
<b>Sprint 4</b>	Module de gestion des évènements	<b>3 semaines.</b>
<b>Sprint 5</b>	Module de gestion des demandes adhésions	<b>3 semaines.</b>
<b>Sprint 6</b>	Module de gestion des dépenses	<b>3 semaines.</b>
<b>Sprint 7</b>	Module de gestion des dons	<b>3 semaines.</b>
<b>Sprint 8</b>	Module de gestion des Bilans	<b>3 semaines.</b>

**Tableau 2 Sprint**

## Conclusion

Dans ce chapitre nous avons pu analyser et spécifier nos besoins fonctionnels et non fonctionnels de notre application ce qui nous amène à aller encore plus loin dans notre projet et nous donner la possibilité de passer à la conception qui fera l'objet du prochain chapitre

# Chapitre 3 : Etude Conceptuelle

## Introduction

Le but de la conception dans une application web est de créer une expérience utilisateur agréable, intuitive et efficace. La conception web vise à offrir une expérience utilisateur agréable, facile à utiliser et efficace tout en respectant les contraintes techniques de l'application. C'est ainsi que dans ce chapitre, nous allons parler du fonctionnement de notre système par une étude générale basée sur la spécification des besoins, puis nous terminons par une étude du projet par plusieurs approches.

## 3.1 Modélisation dynamique

C'est une opération permettant de construire le modèle de l'évolution d'un système ou d'un mécanisme dans le temps ainsi que la séquence de ses opérations.

### 3.1.1 Diagramme de séquences

Un diagramme de séquence est un diagramme d'interaction qui expose en détail la façon dont les opérations sont effectuées : quels messages sont envoyés et quand ils le sont. Les diagrammes de séquence sont organisés en fonction du temps. Le temps s'écoule au fur et à mesure que vous parcourez la page. Les objets impliqués dans l'opération sont répertoriés de gauche à droite en fonction du moment où ils prennent part dans la séquence de messages. Le diagramme de séquence UML affiche les événements par ordre chronologique. C'est pourquoi on l'appelle parfois diagramme d'événement ou scénario d'événement. L'ordre (c'est-à-dire l'ordre exact) est plus important que des dates précises. (3)

#### 3.1.1.1 Composants et symboles élémentaire

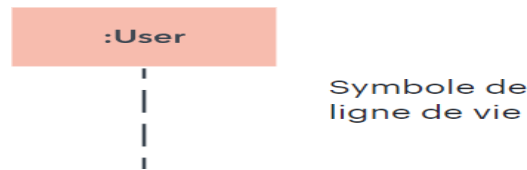
Les diagrammes de séquences sont composés des icônes et des éléments suivants :

- **Symbole d'objet** : montre comment un objet va se comporter dans le contexte du système. Les attributs de classe ne doivent pas être énumérés dans cette forme.

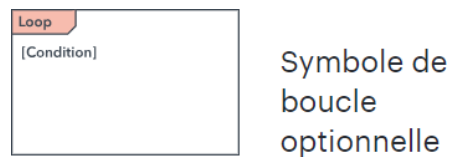


Symbole  
d'objet

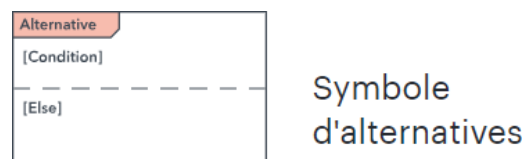
**Symbole de ligne de vie :** les lignes de vie peuvent commencer par une forme rectangulaire avec un intitulé ou par un symbole d'acteur, cette ligne verticale en pointillés montre les évènements séquentiels.



**Symbole de boucle optionnelle :** on utilise ce symbole pour modéliser des scénarios ou une situation qui ne se produira qu'à certaines conditions.



**Symbole d'alternatives :** symbolise des choix (qui en général s'excluent mutuellement) entre deux séquences de messages ou plus. Pour représenter les alternatives, utilisez la forme rectangulaire composant un intitulé et une ligne en pointillés à l'intérieur.

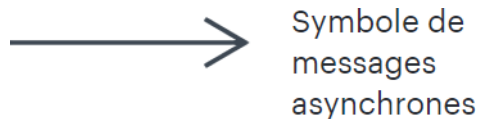


**Symbole de messages synchrones :** représenté par une ligne pleine terminée par une pointe de flèche. On utilise ce symbole lorsqu'un expéditeur doit attendre une réponse à un message avant de continuer. Le diagramme doit montrer à la fois l'appel et la réponse.

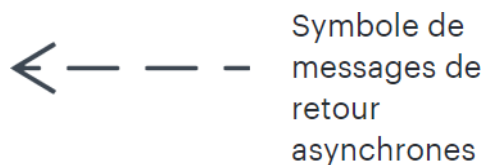


**Symbole de messages asynchrones :**

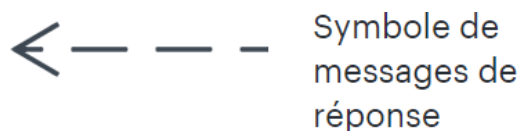
Représentés par une ligne pleine terminée par une pointe de flèche. Les messages asynchrones ne nécessitent pas de réponse avant que l'expéditeur ne continue. Seul l'appel doit être inclus dans le diagramme.



- **Symbole de messages de retour asynchrones :** représentés par une ligne en pointillés terminée par une tête de flèche.



- **Symbole de messages de réponse :** représentés par une ligne en pointillés terminée par une ligne pointe de flèche, ces messages sont des réponses aux appels.



**Diagramme de séquences du sprint (1)**

**Diagramme de séquence « s'authentifier »**

L'utilisateur saisit son nom d'utilisateur et son mot de passe, le système vérifie les données saisies et envoie une requête d'authentification à la base de données. Si les données correspondent à un compte valide, la connexion sera rétablie sinon un message d'erreur sera affiché et ensuite il sera redirigé à la page login.

L'interface d'authentification qui permet de contrôler l'accès à la plateforme. Tout au long de session de l'utilisateur, il a la possibilité de d'accéder qu'aux services dont il est autorisé.

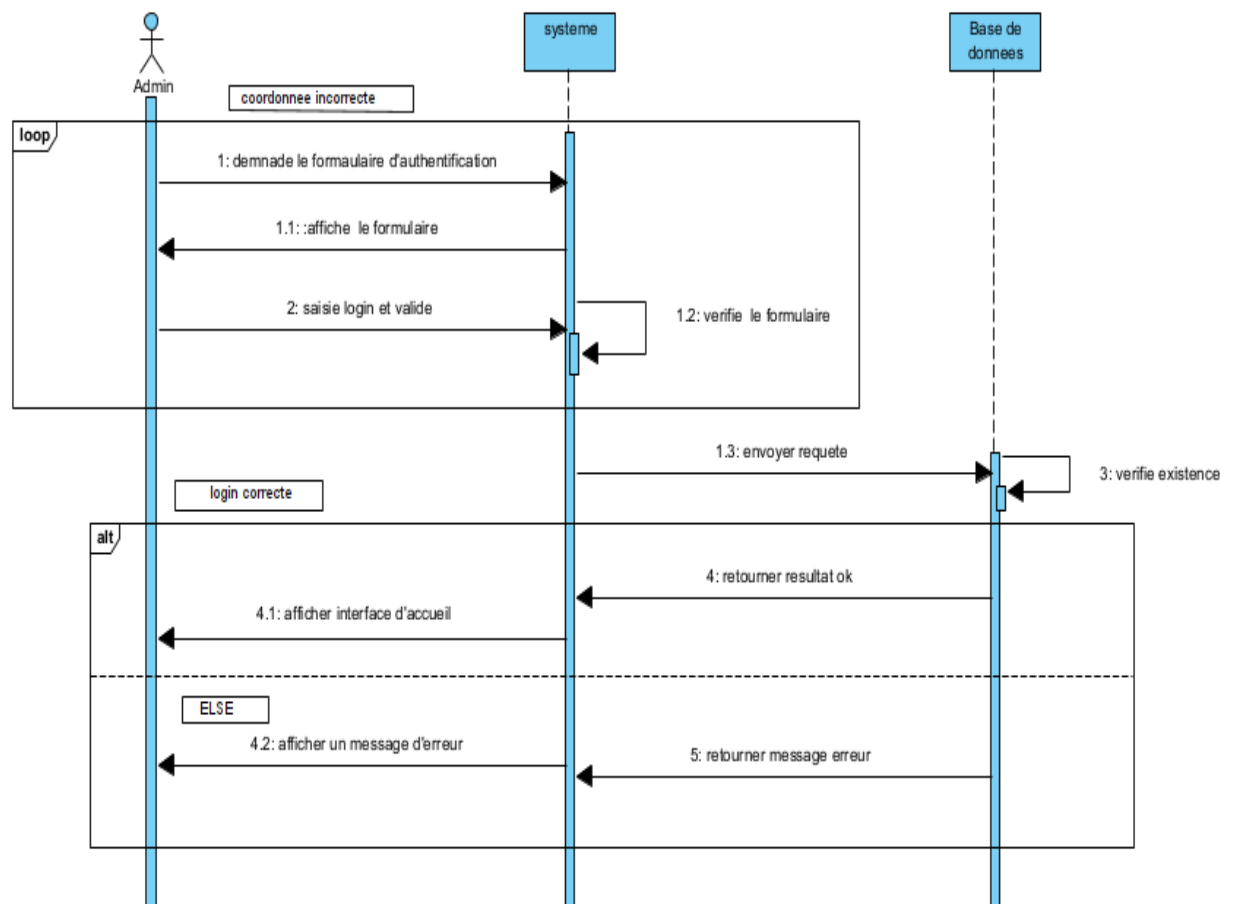


Figure 3 Diagramme de séquence « s'authentifier »

### 3.1.2 Diagramme de séquence « Ajout d'une actualité »

Après authentification, l'administrateur accède à la page d'administration du système puis il choisit le menu actualité sur laquelle il peut : ajouter une actualité.

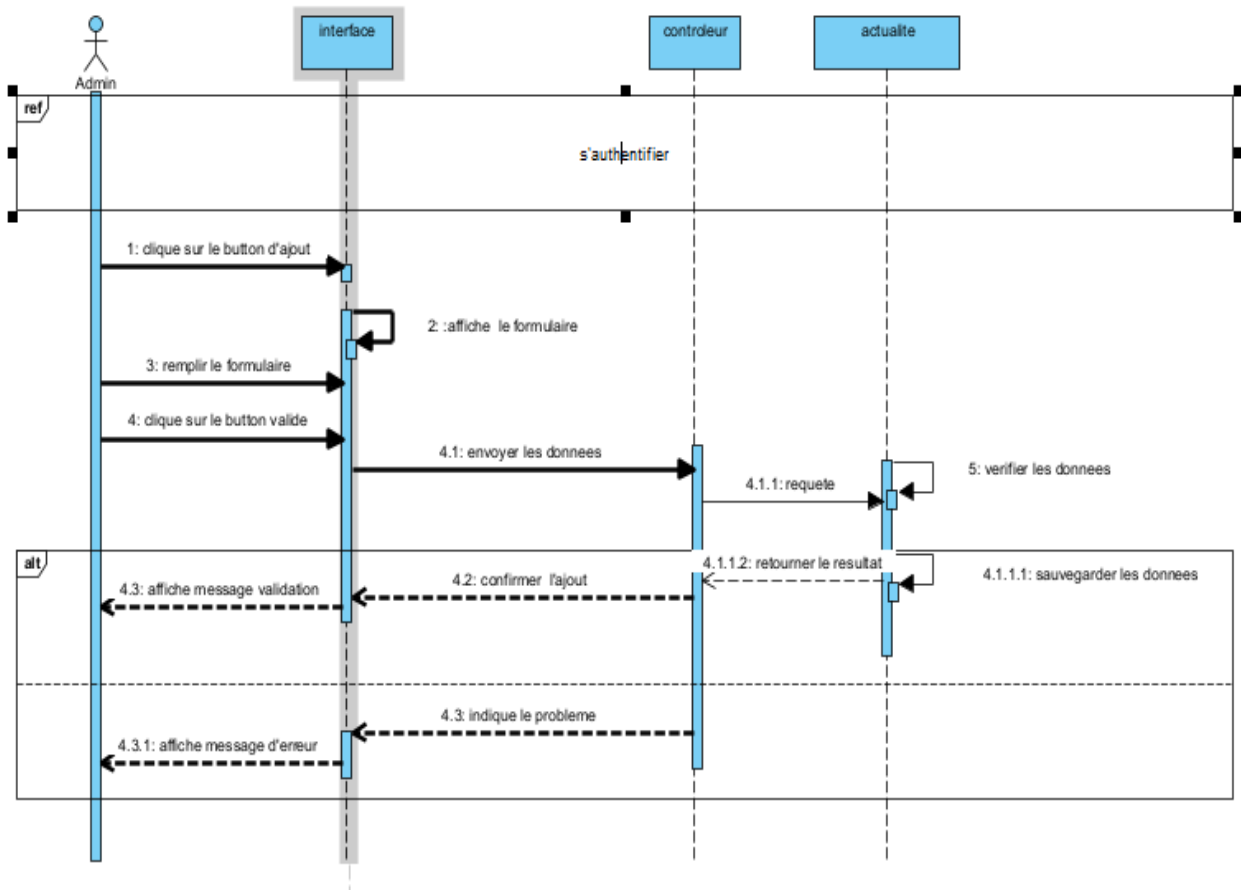


Figure 4 Diagramme de séquence «Ajout actualités »

### 3.1.3 Diagramme de séquence « Ajout d'un événement »

Après authentification, l'administrateur accède à la page d'administration du système puis il choisit le menu événement sur laquelle il peut : ajouter un événement.

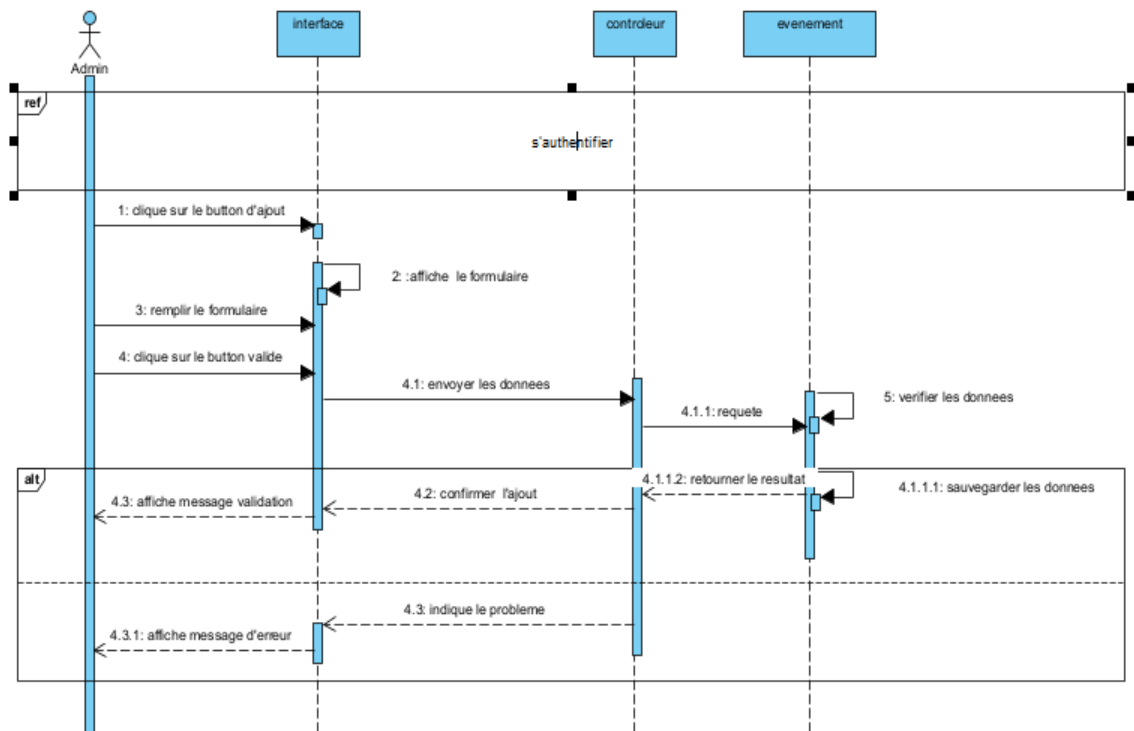


Figure 5 Diagramme de séquence « Ajout évènement »

### 3.1.4 Diagramme de séquence « Ajout d'une session de formation »

Après authentification, l'administrateur accède à la page d'administration du système puis il choisit le menu de session de formation sur laquelle il peut : ajouter une session de formation.



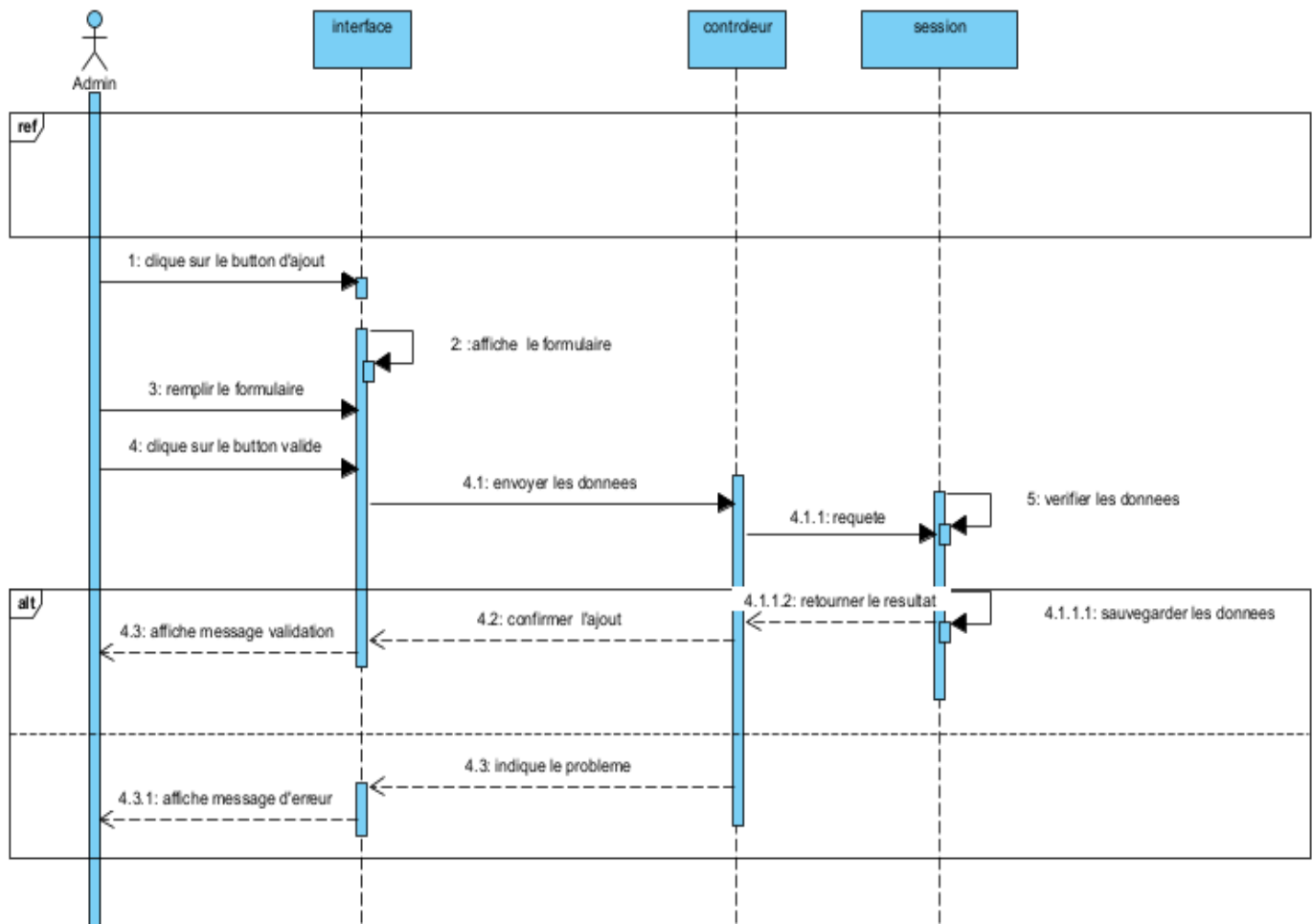


Figure 6 Diagramme de séquence « Ajout session formation »

### 3.1.5 Diagrammes d'activité

Les diagrammes d'activité sont la représentation graphique des interactions entre acteurs et le système selon un ordre logique dans la formulation UML.

On montre ces interactions dans le cadre d'un scénario d'un diagramme de cas d'utilisation. Le but est de décrire comment se déroulent les actions entre les acteurs ou objets. s

Le diagramme est composé de :

- **Action** : étape dans l'activité où les utilisateurs ou le logiciel exécutent une tâche donnée. Les actions sont symbolisées par des rectangles aux bords arrondis.
- **Nœud de décision** : embranchement conditionnel dans le flux, qui est représenté par un losange. Il comporte

une seule entrée et au moins deux sorties.

- **Flux de contrôle** : autre nom donné aux connecteurs qui illustrent le flux entre les étapes du diagramme.
- **Nœud de départ** : élément symbolisant le début de l'activité, que l'on représente par un cercle noir.
- **Nœud de fin** : élément symbolisant l'étape finale de l'activité, que l'on représente par un cercle noir avec un contour.

Et de quelques symboles qu'on a utilisés dans notre diagramme qui sont :

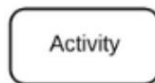
- **Symbole de début** : représente le début d'un processus ou d'un flux de travail dans un diagramme d'activités.

Il peut être utilisé seul ou avec un symbole de note qui explique le point de départ.



Symbole de début

- **Symbole d'activité** : indique les activités qui composent un processus modélisé. Ces symboles, qui comprennent



Symbole d'activité

de brèves descriptions dans la forme, sont les principales composantes d'un diagramme d'activités.

• **Symbole de raccord** : indique le flux directionnel, ou flux de contrôle, de l'activité. Une flèche entrante marque le début d'une étape activité ; une fois l'étape terminée, le flux se poursuit avec la flèche sortante.

- **Symbole de décision** : représente une décision et possède toujours au moins deux embranchements avec le



Symbole de  
raccord

texte de la condition pour permettre aux utilisateurs de voir les options.

Ce symbole représente la ramification ou la fusion et sert de cadre ou de conteneur.



Symbole de  
décision

- **Symbole de fin** : Marque l'état final d'une activité et représente l'achèvement de tous les flux d'un procédé.



Symbole de fin

### - Diagramme d'activité « Authentification »

Le diagramme d'activité d'authentification nous permet de voir les comportements internes du système, lors du démarrage de l'application par l'utilisateur, le système lui affiche le formulaire d'authentification, après que le mot de passe soit saisi le système vérifie sa validité et affiche la page d'accueil sinon il affiche un message d'erreur.

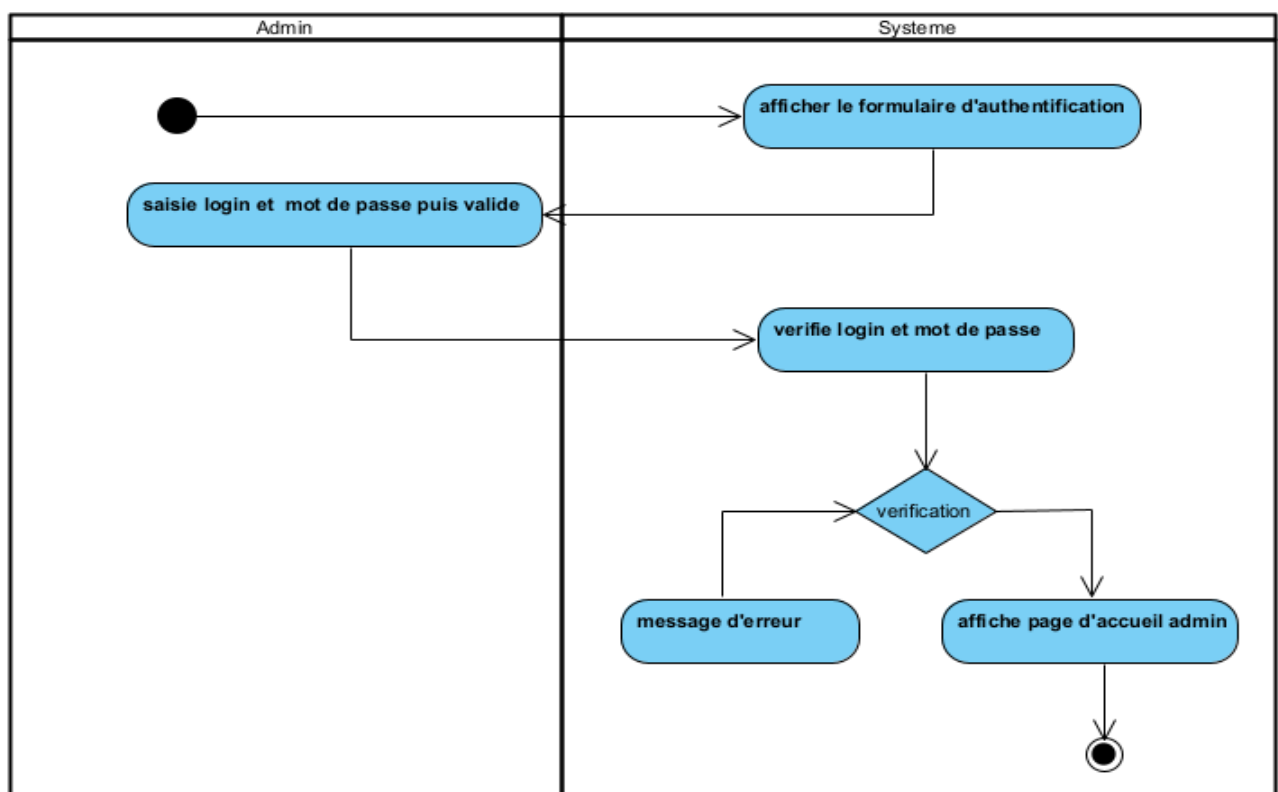


Figure 7 Diagramme d'activité « Authentification »

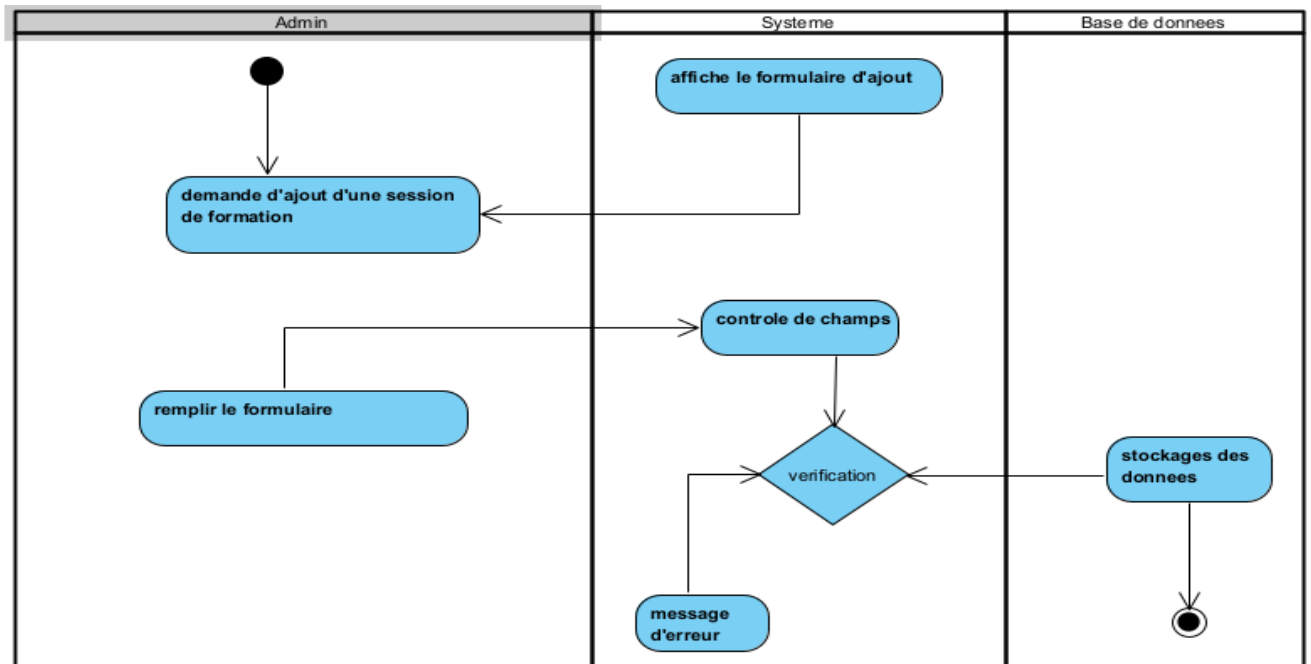


Figure 8 Diagramme d'activité « Ajout d'une session de formation »

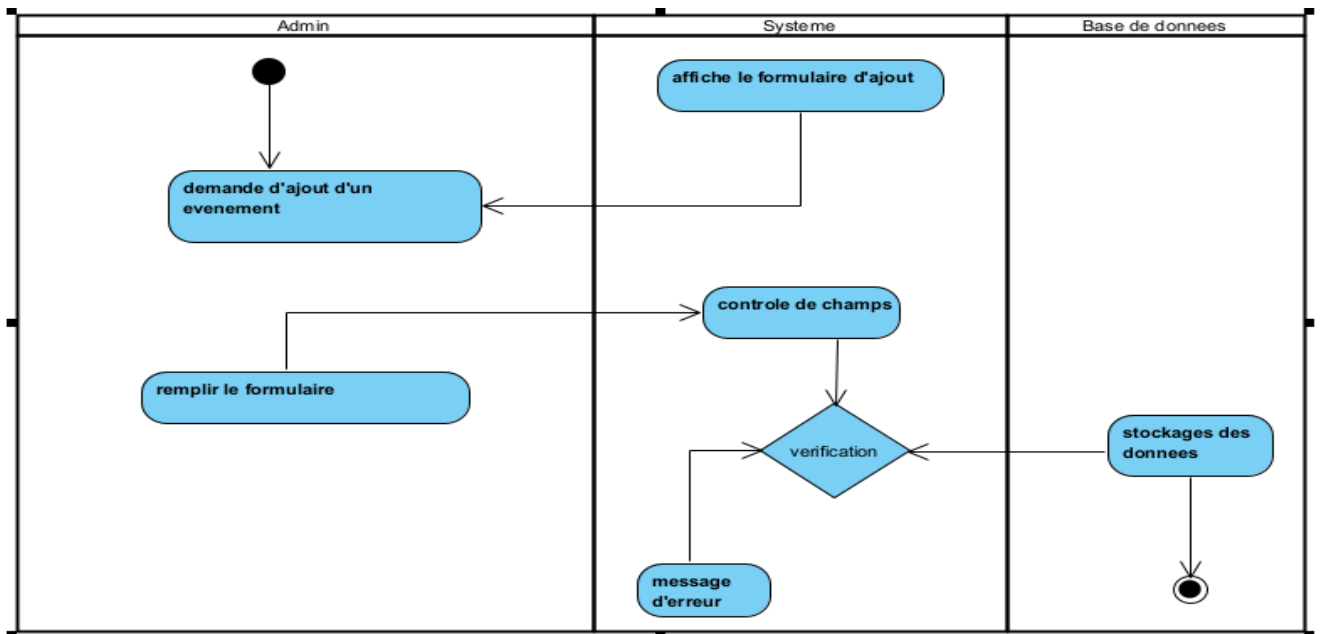


Figure 9 Diagramme d'activité « Ajout d'un évènement»

## 3.2 Modélisation statique

### 3.2.1 Diagramme de cas d'utilisation global

Le diagramme de cas d'utilisation est un diagramme UML utilisé pour donner une vision globale du comportement fonctionnel d'un système ...L'objectif d'un diagramme de cas d'utilisation UML est de représenter les différentes façons dont un utilisateur peut interagir avec un système. Les principaux concepts de ce diagramme sont les acteurs, cas d'utilisation et sujets. Un cas d'utilisation représente une unité discrète d'interaction entre un utilisateur (humain ou machine) et un système. La relation « inclus de » est utile car tous les utilisateurs doivent obligatoirement

La figure ci-dessous illustre la fonctionnalité fournie pour les trois acteurs

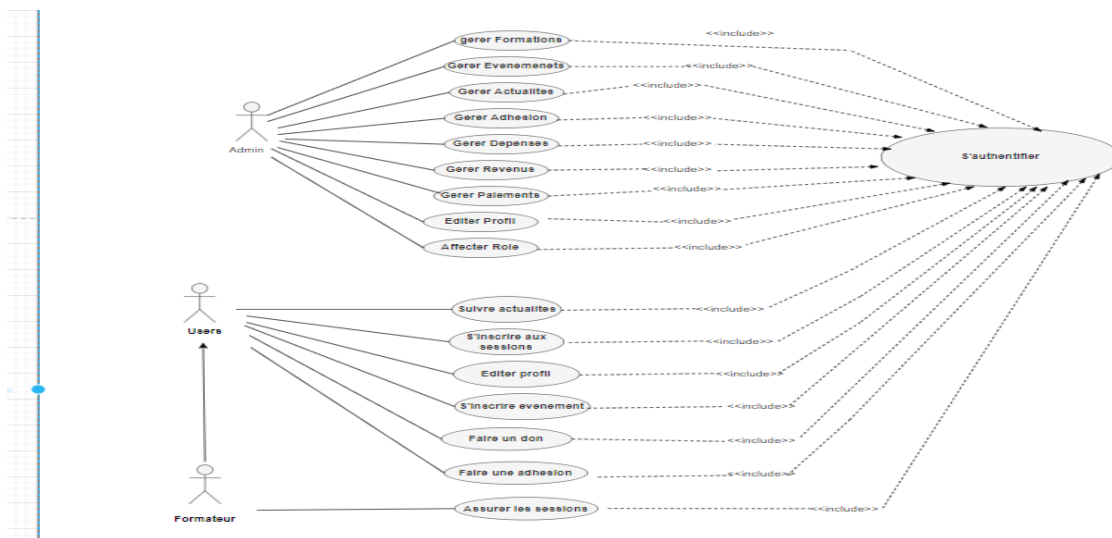


Figure 10 Diagramme de cas d'utilisation global

### 3.2.2 Description détaillée du cas d'utilisation « S'authentifier »

Nom du cas d'utilisation	S'authentifier
Acteurs	administrateur, users, adhérent et formateur
Préconditions	-le système fonctionne -l'utilisateur saisit son email et son mot de passe

Enchaînement nominal	<p>Le système vérifie si les champs ne sont pas vides</p> <p>Il vérifie ensuite si les informations sont valides</p> <p>3- Le système redirige l'acteur vers son espace selon son rôle</p>
Enchaînement alternatif	<p>- En (1) : message d'erreur « un champ est vide »;</p> <p>- En (2) : message d'erreur « email ou mot de passe incorrect »</p>
Exceptions	<p>Le système affiche un message d'erreur dans le cas où le serveur ne fonctionne pas</p>
Post conditions	<p>Ouverture de l'espace de l'acteur</p>

Tableau 3 Description détaillée du cas d'utilisation « S'authentifier »

### 3.2.3 Description détaillée du cas d'utilisation « Affecter rôles »

Cas d'utilisation « Affecter rôle »	
<b>Acteur</b>	<b>Administrateur</b>
<b>Précondition</b>	<b>Administrateur authentifié</b>
<b>Post condition</b>	<b>Administrateur ajouté.</b>
<b>Scenario nominal</b>	<p><b>L'administrateur choisit utilisateur ou adhérent</b></p> <p><b>L'administrateur appuie sur le bouton affecter.</b></p> <p><b>Le système affiche la liste des rôles en check box permettant a l'admin de cocher le rôle.</b></p> <p><b>Le système vérifie si cet utilisateur n'a pas de rôle.</b></p> <p><b>Le système valide affectation et affiche un message de succès.</b></p>
<b>Scenario alternatif</b>	<b>-Message d'erreur : “cet utilisateur a déjà ce rôle ”</b>

<b>Exception</b>	<b>Le système affiche un message d'erreur dans le cas où le serveur n'est pas démarré.</b>
------------------	--

Tableau 4 Description détaillée du cas d'utilisation « Supprimer utilisateur»

### 3.2.4 Description détaillée du cas « Supprimer utilisateur »

Cas d'utilisation « Supprimer utilisateur »	
<b>Acteur</b>	Administrateur
<b>Precondition</b>	Administrateur authentifié
1- Post condition	2- Utilisateur creer.
3- 4- 5- Scenario nominal	6- L'administrateur choisit le menu Utilisateur 7- Le système affiche la liste des utilisateurs. 8- Scenario Supprimer : 9- L'administrateur clique sur le bouton Supprimer. 10- Le système archive utilisateur en affichant un message de succès puis il l'affiche utilisateur dans la liste des utilisateurs archives.
<b>Exception</b>	Le système affiche un message d'erreur dans le cas où le serveur n'est pas démarré.

Tableau 5 Description détaillée du cas d'utilisation « Supprimer utilisateur»

### 3.2.2 Diagramme de classes global

Dans notre diagramme de classe nous observons qu'il Ya des interactions, ces désigne les relations et liens divers qui peuvent exister dans les diagrammes de classes et d'objets. On peut distinguer plusieurs types de relations entre classes : l'association, l'agrégation, la spécialisation/généralisation, la dépendance, dans notre cas nous aurons à parler de :

**Association bidirectionnelle** : relation par défaut entre deux classes. Chacune des deux classes a conscience de l'existence de l'autre et de sa relation avec elle. Cette association est

représentée par une ligne droite entre deux classes.

**Héritage** : également connu sous le nom de généralisation, il s'agit du processus par lequel un enfant ou une sous-classe adopte la fonctionnalité d'un parent ou d'une super-classe. On le symbolise par une ligne de connexion droite avec une pointe de flèche fermée orientée vers la super-classe.

Il y a aussi la multiplicité ou cardinalité qui :

Exactement un : 1 ou 1..1 ;

Plusieurs : \* ou 0..\* ;

Au moins un : 1..\* ;

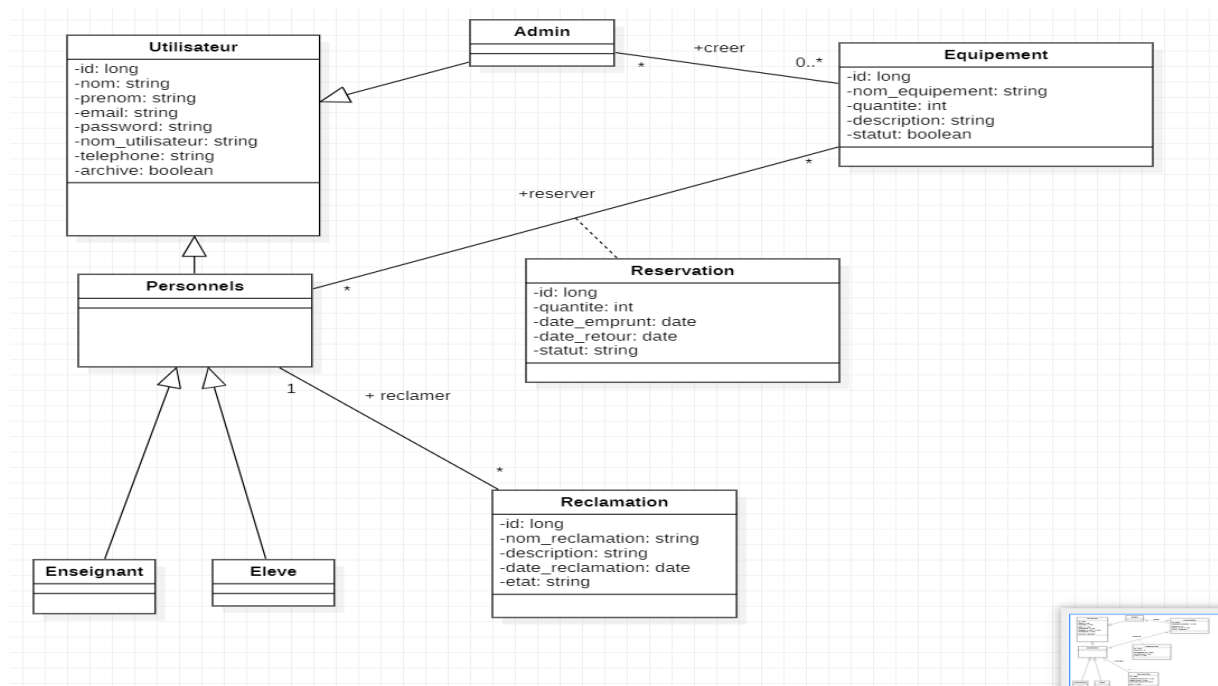


Figure 11 Diagramme de classes global



## Conclusion

Ce chapitre a été consacré à la modélisation UML du système. Nous avons présenté différents diagrammes: les diagrammes de cas d'utilisation, de classes, de séquences et d'activités afin de spécifier de façon détaillée les aspects fonctionnels, dynamiques et statiques du système.

Dans le chapitre suivant, nous présenterons l'environnement de travail et l'architecture de notre application.

# Chapitre 4 : Réalisations

## Introduction

Après avoir étudié l'existant et le critiquer, on a exprimé les besoins fonctionnels et non fonctionnels de la solution, ensuite on a fait une conception de celle-ci. Il ne reste plus qu'à l'implémenter.

Dans ce chapitre, nous décrivons les réalisations effectuées lors de chaque sprint du projet. Chaque sprint a été axé sur le développement de modules spécifiques, avec une durée prédéterminée pour chaque sprint

## 4.1 Sprint 0 : Module des outils de développement

Au cours de ce sprint initial, l'équipe de développement s'est concentrée sur la mise en place de l'environnement de développement et des outils nécessaires pour le projet. Cela comprenait la configuration des IDE, des systèmes de gestionnaire de base de données, des systèmes de contrôle de version, et des outils de collaboration.

**Codage :** L'équipe a travaillé sur la configuration des IDE (Environnement de Développement Intégré) en choisissant les outils adaptés au projet et en les paramétrant selon les besoins spécifiques. De plus, elle a mis en place les systèmes de gestionnaires de bases de données en installant et en configurant les serveurs et les bases de données nécessaires. En parallèle, l'équipe a configuré les systèmes de contrôle de version, comme Git, et a créé les premiers dépôts pour le projet. Enfin, elle a choisi et configuré les outils de collaboration, tels que des plateformes de gestion de projet ou des outils de communication en ligne.

**Test :** Pendant ce sprint, l'équipe a réalisé des tests pour s'assurer que chaque composant de l'environnement de développement fonctionnait correctement. Cela inclut des tests d'intégration pour vérifier la compatibilité des différents outils, des tests de charge pour évaluer les performances des serveurs de bases de données, et des tests de sécurité pour garantir la robustesse de l'environnement contre d'éventuelles vulnérabilités.

### 4.1.1 Environnement de travail

En informatique, un environnement désigne, pour une application, l'ensemble des matériels

---

et des logiciels système, dont le système d'exploitation, sur lesquels sont exécutés les programmes de l'application.

### 4.1.2 Environnement matériel

L'environnement matériel est l'ensemble du matériel, sur lequel le code de l'application est développé et exécuté.

Pour développer ce projet, on a utilisé un ordinateur de la marque "acer" ayant les caractéristiques suivantes :

- ✓ Système d'exploitation : Windows 10 x64.
- ✓ Processeur : Intel Core i5 2410M.
- ✓ RAM : 12Go
- ✓ Disque dur : 450Go SSD 36

### 4.1.3 Environnement logiciel

Dans cette section, on fait la description de tous les logiciels qu'on a utilisés pour la réalisation de l'application.

#### 4.1.3.1 Environnements de développement

Un environnement de développement communément appelé IDE (Integrated

Développent Environment) est un ensemble d'outils qui permet d'augmenter la productivité des programmeurs qui développent des logiciels, des sites web, etc...

Il comporte un éditeur de texte destiné à la programmation, des fonctionnalités

Telles que le débogage et tant d'autres. Certains de ces environnements sont dédiés à un langage de programmation en particulier.

<b>Visual Studio Code (VS Code)</b>	Visual Studio Code est un éditeur de code source développé par Microsoft. Il offre une prise en charge robuste pour JavaScript, Type Script et Node.js, ainsi que de nombreuses fonctionnalités d'édition et de débogage utiles pour le développement d'applications web.
-------------------------------------	---

---

**Tableau 6 éditeur visuel studio code**

**Logiciels divers :**

<b>Postman</b>	Postman reste un outil valable pour tester et développer des API REST, ce qui est pertinent pour votre projet. Vous pouvez l'utiliser pour envoyer des requêtes HTTP (GET, POST, PUT, etc.) à votre API back end développée avec Node.js et Express.js, ainsi que pour vérifier les réponses et les performances.
<b>MongoDB Compass</b>	MongoDB Compass est un outil visuel permettant de gérer et d'interagir avec des bases de données MongoDB. Vous pouvez l'utiliser pour explorer et manipuler les données stockées dans votre base de données MongoDB, qui est probablement utilisée dans le back end de votre application.

**Tableau 7 logiciels divers**

**4.2.2.2 Framework de développement**

Dans cette section, on fait la description de tous les Framework dont on s'est servi pour le back end et pour le front end.

**4.2.2.3FrameworkBack end**

<b>Node.js avec Express.js</b>	
<b>Description</b>	<b>Pourquoi utiliser Node.js avec Express.js ?</b>

Node.js est un environnement d'exécution JavaScript côté serveur, tandis qu'Express.js est un Framework web minimal et flexible pour Node.js. Ensemble, ils permettent de construire des applications web rapides et évolutives en utilisant JavaScript côté serveur. Express.js simplifie la création d'API RESTful et offre une grande flexibilité pour la gestion des routes et des middlewares.	L'utilisation de Node.js avec Express.js offre plusieurs avantages, notamment une grande performance, une scalabilité horizontale facile à mettre en place, une syntaxe JavaScript unifiée sur toute la pile technologique, ainsi qu'une vaste gamme de modules complémentaires et de middleware pour étendre les fonctionnalités de base.
---	--

**Tableau 8 Framework back end node js**

#### 4.2.2.4 Framework Front end

<b>React.js</b>	
<b>Description</b>	<b>Pourquoi utiliser React.js ?</b>
React.js est un Framework JavaScript pour la création d'interfaces utilisateur et d'applications web monopages (SPA). Il offre des fonctionnalités avancées telles que le rendu côté client, la gestion de l'état avec des composants réutilisables, et facilite le développement modulaire grâce à son approche basée sur les composants. React.js est largement utilisé pour sa performance, sa flexibilité et sa communauté active de développeurs.	React.js présente plusieurs avantages, notamment une architecture basée sur des composants qui facilite la réutilisation du code, une gestion efficace de l'état de l'application, un rendu côté client qui améliore les performances et une large communauté de développeurs qui fournit des ressources et des bibliothèques complémentaires pour accélérer le développement.
<b>Bootstrap</b>	
<b>Description</b>	<b>Pourquoi utiliser Bootstrap ?</b>

<p>Bootstrap est un Framework qui contient un ensemble de codes HTML et CSS ainsi que des extensions JavaScript en option fonctionnant ensemble et qu'on va pouvoir utiliser pour créer des designs complexes de manière relativement simple.</p>	<p>Bootstrap permet de gagner énormément de temps. On n'a pas besoin de coder toute la partie CSS. Il suffit de sélectionner les classes qu'on veut en codant la page HTML. Ensuite, grâce à l'enchaînement de classes prédéfinies, on structurera le code et obtiendra plus de cohérence dans celui-ci.</p>
---	--

**Tableau 9 Framework front end React js bootstrap**

### 4.2.3 Outils divers

<b>MongoDB</b>	<p>MongoDB est une base de données NoSQL open source, orientée documents. Contrairement aux bases de données relationnelles traditionnelles, MongoDB stocke les données sous forme de documents JSON flexibles, ce qui offre une grande évolutivité et une plus grande souplesse dans la modélisation des données.</p> <p>MongoDB sera utilisé comme système de gestion de base de données pour stocker et gérer les données de votre application de gestion d'association pour la science et la technologie. Il permettra de gérer les informations sur les membres, les événements, les projets, les publications, etc., offrant une flexibilité et une évolutivité pour répondre aux besoins changeants de l'association.</p>
<b>GitHub</b>	<p>GitHub est une plateforme de développement logiciel basée sur Git, offrant des fonctionnalités de contrôle de version, de collaboration et d'hébergement de code source. Il permet aux développeurs de travailler ensemble sur des projets, de suivre les modifications du code et de gérer les versions du logiciel.</p> <p>GitHub sera utilisé pour héberger le code source de votre application de gestion d'association pour la science et la technologie. Il facilitera la collaboration entre les membres de l'équipe de développement, suivra les modifications du code et gèrera les versions du logiciel, assurant ainsi un développement fluide et bien organisé.</p>

<b>StarUML</b>	<p>StarUML est un logiciel de modélisation UML (Unified Modeling Language) qui permet de concevoir et de visualiser différents types de diagrammes pour représenter la structure et le comportement d'un système logiciel.</p> <p>StarUML sera utilisé pour créer des diagrammes UML décrivant l'architecture, les relations entre les composants et d'autres aspects de votre application de gestion d'association pour la science et la technologie. Ces diagrammes faciliteront la compréhension du système par les membres de l'équipe de développement et aideront à guider le processus de développement de manière plus efficace.</p>

**Tableau 10 outils divers**

## 4.2 Sprint 1 : Module des comptes utilisateurs

Ce sprint était dédié au développement du module de gestion des comptes utilisateurs.

Les fonctionnalités développées comprenaient l'inscription des utilisateurs, la connexion, la gestion des profils utilisateurs, la réinitialisation des mots de passe, et la gestion des autorisations.

**Codage :** L'équipe a développé les fonctionnalités du module de gestion des comptes utilisateurs en implémentant les mécanismes d'inscription, de connexion, de gestion des profils, de réinitialisation des mots de passe et de gestion des autorisations. Elle a également intégré ces fonctionnalités avec l'interface utilisateur et les bases de données configurées lors du sprint précédent.

**Test :** Pendant ce sprint, des tests unitaires ont été réalisés pour chaque fonctionnalité développée afin de vérifier leur bon fonctionnement individuel. Des tests d'acceptation ont également été effectués pour valider l'ensemble du flux utilisateur, depuis l'inscription jusqu'à la gestion des autorisations.

## 4.3 Sprint 2 : Module de gestion des sessions de formations

Durant ce sprint, l'équipe a travaillé sur le développement du module de gestion des sessions

---

de formations. Les fonctionnalités principales comprenaient la création, la modification et la suppression de sessions de formation, la gestion des inscriptions aux sessions, et la génération de rapports sur les participants.

**Codage :** L'équipe a travaillé sur le développement du module de gestion des sessions de formations en mettant en place les fonctionnalités de création, modification et suppression de sessions de formation, ainsi que la gestion des inscriptions et la génération de rapports sur les participants. Elle a assuré l'intégration de ces fonctionnalités avec le système de comptes utilisateurs développé lors du sprint précédent.

**Test :** Des tests de régression ont été effectués pour garantir que l'ajout de nouvelles fonctionnalités n'a pas introduit de régressions dans les fonctionnalités existantes. Des tests de performance ont également été réalisés pour évaluer la réactivité du système face à un grand nombre d'utilisateurs simultanés. Enfin, des tests de génération de rapports ont été effectués pour vérifier l'exactitude et la pertinence des données fournies.

## 4.4 Sprint 3 : Module de gestion des actualités

Ce sprint a été consacré au développement du module de gestion des actualités. Les fonctionnalités développées incluaient la création, l'édition et la suppression d'articles d'actualités, la gestion des commentaires, et la publication automatique sur les réseaux sociaux.

**Codage :** L'équipe a développé les fonctionnalités du module de gestion des actualités en mettant en place les mécanismes de création, d'édition et de suppression d'articles d'actualités. Elle a également intégré la gestion des commentaires associés à chaque article et a mis en œuvre la fonctionnalité de publication automatique sur les réseaux sociaux, si elle était prévue dans les spécifications du projet.

**Test :** Durant ce sprint, des tests unitaires ont été réalisés pour chaque fonctionnalité développée, notamment pour s'assurer que les articles peuvent être créés, édités et supprimés correctement. Des tests d'intégration ont également été effectués pour vérifier que les commentaires sont associés aux articles de manière appropriée. Enfin, des tests de bout en bout ont été réalisés pour vérifier le fonctionnement global du module, y compris la publication automatique sur les réseaux sociaux.



---

## 4.5 Sprint 4 : Module de gestion des événements

Au cours de ce sprint, l'équipe a travaillé sur le développement du module de gestion des événements. Les fonctionnalités principales comprenaient la création, la modification et la suppression d'événements, la gestion des inscriptions aux événements, et la génération de rapports sur les participants.

**Codage :** L'équipe a travaillé sur le développement du module de gestion des événements en mettant en place les fonctionnalités de création, modification et suppression d'événements. Elle a également développé le système de gestion des inscriptions aux événements, permettant aux utilisateurs de s'inscrire et de se désinscrire aux événements disponibles. De plus, elle a mis en place la génération de rapports sur les participants pour chaque événement.

**Test :** Pour ce sprint, des tests unitaires ont été réalisés pour chaque fonctionnalité développée, notamment pour garantir que les événements peuvent être créés, modifiés et supprimés correctement. Des tests d'intégration ont également été effectués pour vérifier que les inscriptions aux événements sont correctement gérées et que les rapports sur les participants sont précis.

## 4.6 Sprint 5 : Module de gestion des demandes adhésions

Le dernier sprint était dédié au développement du module de gestion des demandes d'adhésions. Les fonctionnalités développées incluaient la soumission de demandes d'adhésion par les utilisateurs, la gestion et l'approbation des demandes par les administrateurs, et la communication avec les demandeurs.

**Codage :** L'équipe a concentré ses efforts sur le développement du module de gestion des demandes d'adhésions en mettant en place les fonctionnalités de soumission de demandes d'adhésion par les utilisateurs. Elle a également développé les fonctionnalités de gestion et d'approbation des demandes par les administrateurs, ainsi que les mécanismes de communication avec les demandeurs tout au long du processus.

**Test :** Pour ce sprint, des tests unitaires ont été réalisés pour chaque fonctionnalité développée, notamment pour garantir que les utilisateurs peuvent soumettre leurs demandes d'adhésion correctement et que les administrateurs peuvent les gérer efficacement. Des tests d'intégration ont également été effectués pour s'assurer que les demandes sont correctement

---

traitées dans l'ensemble du système.

## 4.7 Sprint 6 : Module de gestion des dépenses

Ce sprint est dédié au développement du module de gestion des dépenses, qui vise à suivre et à enregistrer les dépenses engagées lors de l'organisation des sessions de formation et d'autres événements. Cela inclut le suivi des dépenses pour les frais de location de salle, les fournitures telles que les feutres, ainsi que d'autres dépenses liées à la logistique et à l'organisation des événements.

**Codage :** Pendant ce sprint, l'équipe travaille sur l'implémentation des fonctionnalités nécessaires pour enregistrer les dépenses. Cela comprend la création d'une interface utilisateur permettant aux utilisateurs autorisés d'ajouter des dépenses, en spécifiant les détails tels que la description de la dépense, le montant, la date et la catégorie. De plus, l'équipe développe les mécanismes de validation pour s'assurer que les données saisies sont cohérentes et correctes. Elle assure également l'intégration avec les autres modules du système, notamment le module de gestion des sessions de formation et le module de gestion des événements, pour permettre une traçabilité complète des dépenses par rapport aux activités organisées.

**Test :** Pour ce sprint, des tests unitaires sont réalisés pour chaque fonctionnalité développée, notamment pour vérifier que les dépenses peuvent être ajoutées correctement et que les données saisies sont valides. Des tests d'intégration sont également effectués pour s'assurer que les dépenses sont correctement liées aux sessions de formation et aux événements correspondants. De plus, des tests de validation sont mis en place pour garantir que seules les personnes autorisées peuvent ajouter ou modifier des dépenses. Enfin, des tests de performance peuvent être effectués pour évaluer la réactivité du système lors de la gestion d'un grand nombre de dépenses.

## 4.8 Sprint 7 : Module de gestion des dons

**Description :** Ce sprint se concentre sur le développement du module de gestion des dons, permettant de suivre et de gérer les dons financiers reçus pour soutenir les activités telles que les sessions de formation et autres événements.

**Codage :** Pendant ce sprint, l'équipe développe les fonctionnalités permettant d'enregistrer les dons financiers dans le système. Cela inclut la création d'une interface utilisateur permettant aux donateurs de soumettre leurs dons en spécifiant le montant, la date, et éventuellement une

---

description ou une note associée au don. L'équipe met également en place des mécanismes de validation pour s'assurer que les données saisies sont correctes et cohérentes. Elle assure également l'intégration avec d'autres modules du système, permettant une traçabilité complète des dons par rapport aux activités qu'ils soutiennent.

**Test :** Des tests unitaires sont réalisés pour chaque fonctionnalité développée, notamment pour vérifier que les dons peuvent être ajoutés correctement et que les données saisies sont valides. Des tests d'intégration sont effectués pour s'assurer que les dons sont correctement liés aux activités correspondantes, comme les sessions de formation ou les événements. Des tests de validation sont mis en place pour garantir que seuls les utilisateurs autorisés peuvent soumettre des dons. Enfin, des tests de performance peuvent être effectués pour évaluer la réactivité du système lors de la gestion d'un grand nombre de dons.

---

# Chapitre 5 : Présentation de l'application

## Introduction

Nous sommes à la phase finale, la partie la plus importante puisqu'elle met en réalité toute la théorie précédente. Nous commencerons par quelques interfaces de notre application ensuite nous terminerons par les caractéristiques de l'application.

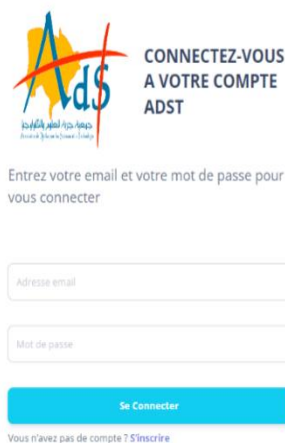
Dans cette section, on va voir quelques interfaces tout en donnant une description de celles-ci.

### 5.1 Quelques interfaces de l'application

#### 5.1.1 Page d'authentification

Cette page est la première de l'application, elle permet aux différents utilisateurs de l'application d'accéder à leur espace.

Pour s'authentifier, l'utilisateur donne son email et son mot de passe et si les deux informations sont correctes alors il accède à son espace si non le système lui demande de saisir les informations correctes.



The login form features the ADST logo on the left, which includes a stylized 'AdS' and the text 'الجمعية التونسية للعلوم والتكنولوجيا' and 'Association Tunisienne des Sciences et Technologies'. To the right of the logo, the text reads 'CONNECTEZ-VOUS A VOTRE COMPTE ADST'. Below this, a prompt says 'Entrez votre email et votre mot de passe pour vous connecter'. There are two input fields: 'Adresse email' and 'Mot de passe'. A blue 'Se Connecter' button is positioned below the fields. At the bottom, a link states 'Vous n'avez pas de compte ? S'inscrire'.



Figure 12 : Page d'authentification

### 5.1.2 Page de création de compte pour un membre de l'association

La page de création de compte permet aux membres potentiels de l'association de créer un compte utilisateur et de rejoindre l'association. Une fois le compte créé et validé, l'utilisateur peut accéder à son espace personnel.

Pour créer un compte, l'utilisateur doit fournir les informations requises. Si les informations sont valides, l'utilisateur peut accéder à son espace personnel. Dans le cas contraire, le système demande à l'utilisateur de corriger les informations fournies.



Figure 13 : Page d'inscription

Une fois authentifié, l'utilisateur peut accéder à la page de gestion des actualités à partir du menu latéral, qui répertorie toutes les actualités publiées par l'association. Sur cette page, l'utilisateur peut effectuer des recherches en fonction de différents critères pertinents, tels que la date de publication, le sujet, l'auteur, etc.

En outre, l'utilisateur peut interagir avec chaque actualité répertoriée en utilisant des options

Cette page offre donc à l'utilisateur une interface conviviale et intuitive pour gérer efficacement les actualités de l'association, en veillant à ce que les membres soient informés des événements et des nouvelles importantes dans le domaine de la science et de la technologie.

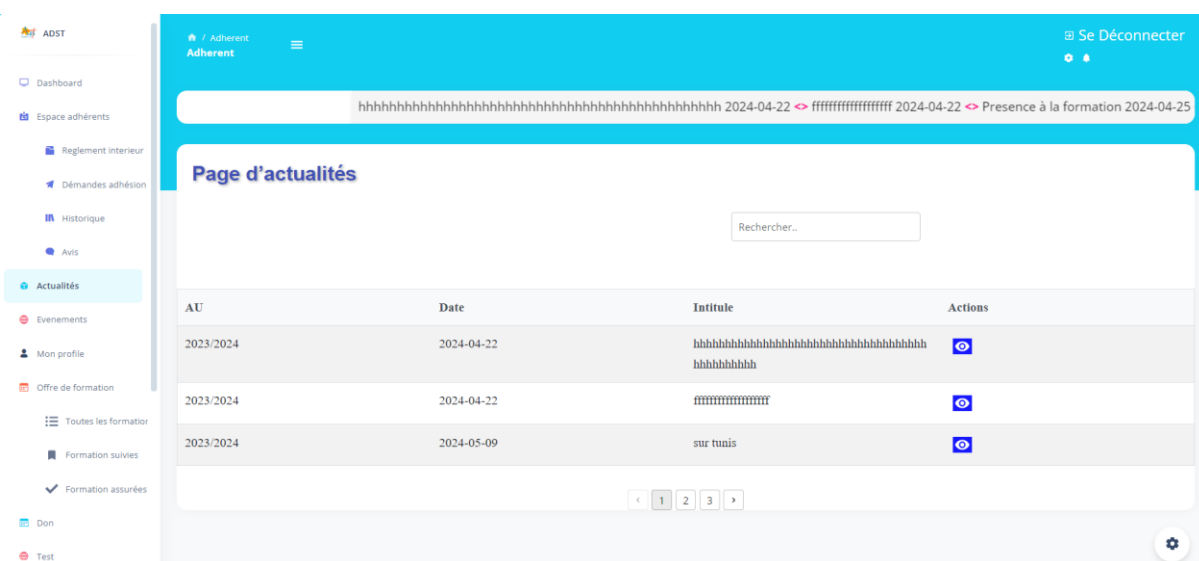


Figure 14 : Page d'actualités pour les utilisateurs

## Gestion des sessions de formation

Une fois authentifié, l'utilisateur peut accéder à la page de gestion des sessions de formation à partir du menu latéral, qui répertorie toutes les sessions de formation organisées par l'association. Sur cette page, l'utilisateur peut effectuer des recherches en fonction de différents critères pertinents, tels que la date, le sujet, le formateur, etc.

En outre, l'utilisateur peut interagir avec chaque session de formation répertoriée en utilisant des options d'action. Par exemple, en cliquant sur une session dans la liste, l'utilisateur peut afficher les détails complets de la session, y compris une description détaillée, le formateur, le lieu, la date et l'heure.

Cette page offre donc à l'utilisateur une interface conviviale et intuitive pour gérer efficacement les sessions de formation organisées par l'association, en veillant à ce que les membres aient accès à des opportunités de développement professionnel pertinentes dans le

domaine de la science et de la technologie.

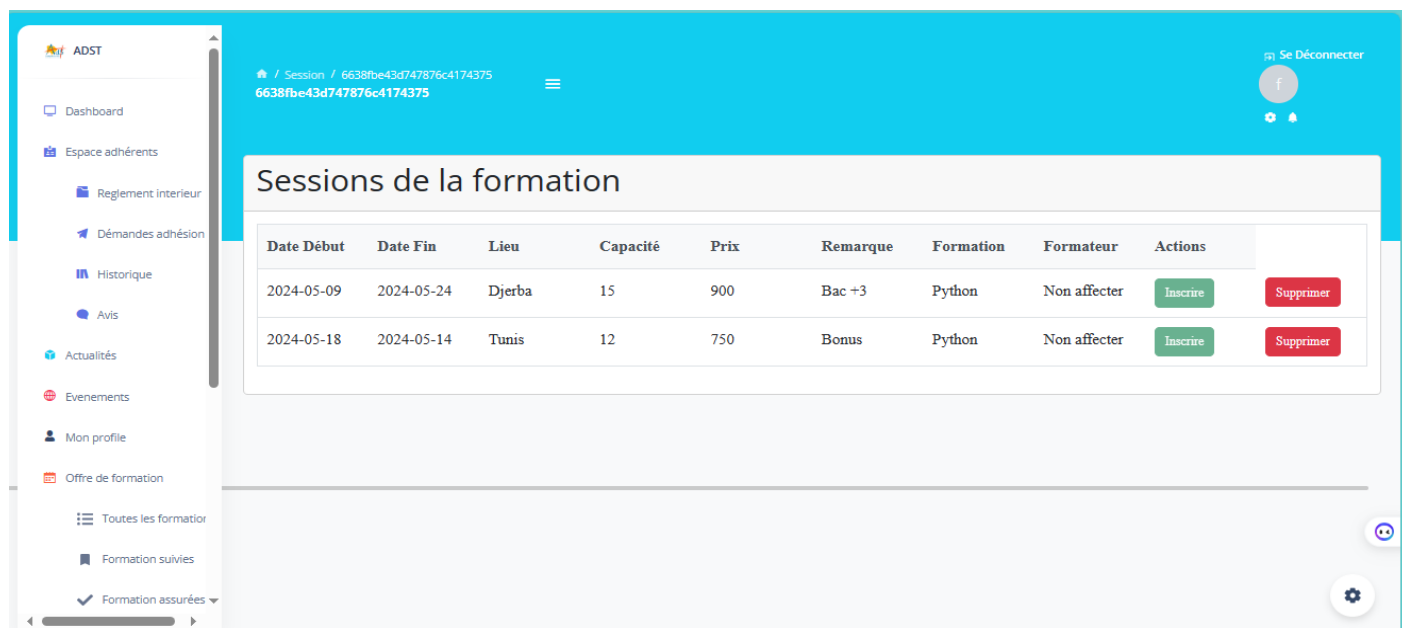


Figure 15 : Gestion des sessions de formation

## 5.2 Autres caractéristiques particulières de l'application

### 5.2.1 Persistance des données

Nous utilisons MongoDB comme système de gestion de base de données pour notre projet de gestion d'association pour la science et la technologie.

MongoDB est une base de données NoSQL orientée documents. Voici comment cela s'applique à notre projet :

Utilisation de MongoDB : MongoDB offre une flexibilité et une évolutivité importantes pour stocker et gérer les données de notre association. Étant une base de données orientée documents, elle permet de stocker les informations de manière souple et intuitive, adaptée à nos besoins variés, tels que les membres, les événements, les ressources, etc.

Mongoose pour Node.js : Pour faciliter l'interaction avec MongoDB dans notre application Node.js, nous utilisons Mongoose. Mongoose nous permet de définir des schémas de données, de réaliser des opérations de CRUD et de valider les données de manière efficace.







Figure 16 : Mongo DB

### 5.2.2 Sécurité

Dans notre projet, la sécurité est une priorité essentielle. Pour assurer l'authentification sécurisée des utilisateurs, nous avons opté pour l'utilisation de bcrypt, une bibliothèque de hachage de mots de passe robuste pour Node.js.

#### **Utilisation de bcrypt :**

Bcrypt est une bibliothèque de hachage de mots de passe qui utilise un algorithme de hachage sécurisé pour stocker les mots de passe de manière sûre dans la base de données. Voici comment cela fonctionne dans notre contexte :

**Hachage sécurisé des mots de passe :** Lorsqu'un utilisateur crée un compte ou modifie son mot de passe, bcrypt est utilisé pour hasher le mot de passe avant de le stocker dans la base de données. Cela garantit que les mots de passe des utilisateurs ne sont jamais stockés en texte brut, offrant ainsi une protection supplémentaire contre les violations de sécurité.

**Comparaison sécurisée des mots de passe :** Lorsqu'un utilisateur tente de se connecter, bcrypt est utilisé pour comparer le mot de passe fourni avec le hachage stocké dans la base de données. Cela garantit une vérification sécurisée du mot de passe sans jamais exposer le mot de passe réel.

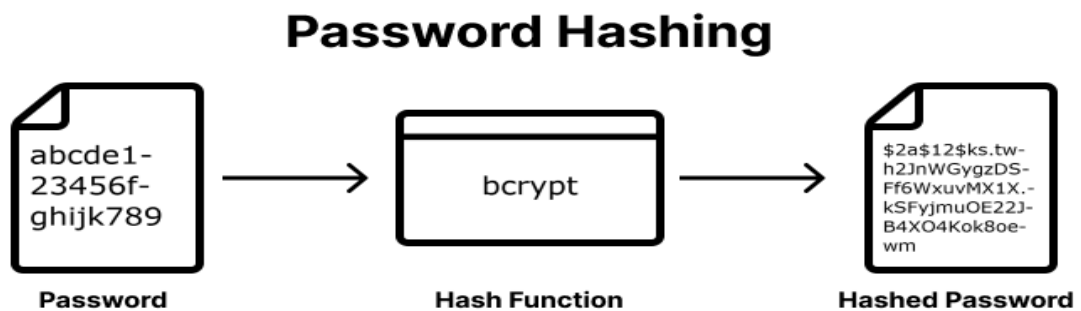


Figure 17 bcrypt

### 5.2.3 StarUML :

StarUML est un outil de modélisation UML (Unified Modeling Language) utilisé pour concevoir et visualiser des diagrammes de classes, de séquence, d'activité, de cas d'utilisation et bien d'autres encore. Avec une interface utilisateur conviviale et des fonctionnalités avancées, StarUML permet aux développeurs et aux concepteurs de créer facilement des modèles visuels pour

représenter la structure, le comportement et les interactions des systèmes logiciels.



Figure 18 : StarUML

**5.2.4 Draw.io** : est une application gratuite en ligne, accessible via son navigateur (protocole https) qui permet de dessiner des diagrammes ou des organigrammes. Cet outil vous propose de concevoir toutes sortes de diagrammes, de dessins vectoriels, de les enregistrer au format XML puis de les exporter. Draw.io est un véritable couteau suisse de la frise chronologique, de la carte mentale et des diagrammes de tout genre.



Figure 19 : Draw-io

## Conclusion

Dans ce chapitre, on a présenté l'environnement de travail (logiciel et matériel) et les langages qui ont permis de réaliser cette application.

Et enfin on a présenté quelques captures d'écran de l'application, une brève explication de leur fonctionnement et présenté quelques particularités de l'application.

## Conclusion Générale et Perspectives

À l'ère du numérique, l'importance de l'informatisation dans divers secteurs, y compris celui de la science et de la technologie, est indéniable. Pour rester compétitives sur le marché en constante évolution, les associations doivent moderniser leurs pratiques et adopter des solutions technologiques innovantes.

Ce projet de gestion d'association pour la science et la technologie, mené au sein de ADST avait pour objectif de fournir une plateforme complète répondant aux besoins fonctionnels et techniques des membres de l'association. Après une analyse approfondie des processus existants, nous avons conçu et développé une application web basée sur les principes de l'Agilité, notamment le Scrum.

La partie back end de l'application consistait à développer des micros services pour gérer efficacement les différentes activités de l'association, telles que la gestion des adhérents, des événements, des projets, etc. La partie front end, quant à elle, offrait des interfaces conviviales permettant une interaction intuitive avec ces micros services.

Notre application a permis d'améliorer la gestion quotidienne des activités de l'association, en offrant une solution simple et efficace pour les membres. Elle leur a également fait gagner du temps et de l'énergie en automatisant certaines tâches répétitives, leur permettant ainsi de se concentrer davantage sur leurs missions principales.

Pour l'avenir, plusieurs perspectives s'offrent à nous. Nous envisageons de développer de nouvelles fonctionnalités telles que la gestion des rôles et des équipes, la récupération de mot de passe, ainsi que la création d'une version mobile de l'application. Ces évolutions permettront à l'association d'offrir encore plus de valeur à ses membres et de rester à la pointe de l'innovation dans son domaine. En résumé, ce projet a permis à l'association de bénéficier d'une solution informatique adaptée à ses besoins spécifiques, tout en ouvrant la voie à de nouvelles possibilités d'amélioration et de croissance pour l'avenir.