

République Algérienne Démocratique et Populaire

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique



Université A. Mira de Béjaïa

Faculté des Sciences Exactes

Département d'Informatique

Mémoire de Fin de cycle

En vue de l'obtention du diplôme de Master en Informatique

Génie Logiciel

Thème

**Développement d'une application web service pour
l'organisation d'événements pour l'entreprise DECIBEL
EVENTS**

Présenté par :

Manal BENDJELILI & Walid BOUMEGUER

Devant le jury composé de :

Président :	Loutfi ZERARGA . MCB	U.A/Mira Béjaïa
Examineur :	Louiza BELKHIRI . MAA	U.A/Mira Béjaïa
Encadrant :	Souhila.AIT HACENE. MAA	U.A/Mira Béjaïa

Année Universitaire : 2022/2023

Remerciements

*Tout d'abord, nous tenons à remercier **Dieu** le tout puissant, de nous avoir donné la santé, la volonté et la patience pour terminer notre formation de master et pouvoir réaliser ce modeste travail.*

*Nous tenons à remercier notre encadrante **S.AIT HACENE** pour son expertise, ses conseils éclairés et son accompagnement bienveillant tout au long de ce parcours.*

*Nous tenons à remercier l'équipe de **DECIBEL EVENTS**, en particulier **Mr KESSASSI Amine Mehdi** et **DRIES Amel***

Nous tenons à exprimer nos sincères remerciements aux les membres du jury qui ont accepté de consacrer leur temps et leur expertise pour évaluer notre travail.

Nous tenons à remercier tous les enseignant que nous avons eu le plaisir de côtoyer pendant la période de notre formation à l'Université de Bjaia.

Un grande merci à nos familles pour leurs soutiens aussi bien moral que financier et pour leurs sacrifices.

Dédicace

C'est avec une grande modestie et un immense plaisir que je dédie ce modeste travail :

À ma merveilleuse mère, qui a été ma source d'amour inconditionnel, de soutien et de sagesse tout au long de ma vie.

À mon cher père, que son âme repose en paix, pour avoir été mon modèle de courage, de détermination et de générosité. Tu restes à jamais dans mon cœur et tes enseignements continuent de guider mes pas. Cette dédicace est un hommage à ton héritage et à ton amour éternel.

*À mon frère et sœurs bien-aimés, **Souhila, Mounir et Lina.***

*À mon binome **Walid** pour sa collaboration et son engagement tout au long de notre projet.*

À mes chers amis, en témoignage de l'amitié qui nous unie et des souvenirs de tous les moments que nous avons passé ensemble, je vous dédie ce travail et je vous souhaite une vie pleine de réussite et de bonheur.

Un grand merci à toutes les personnes ayant contribué à l'élaboration de ce mémoire.

Manal

Dédicace

Je dédie ce modeste travail :

À mes chers parents qui nous ont soutenus et encouragés durant toute notre scolarité. Votre confiance en mes capacités a été ma plus grande force et je vous en suis infiniment reconnaissant

À la mémoire de mes chers grands-parents, votre sagesse et à l'héritage précieux que vous avez laissé derrière vous. Vous avez été des figures centrales dans ma vie, m'offrant un soutien inconditionnel et des valeurs qui continuent de guider ma route (paix à vos âmes).

*À ma sœur **Cylia** tu as toujours été là pour moi, me soutenant et me motivant à aller de l'avant. Notre lien fraternel est une bénédiction qui m'inspire et me donne la force de surmonter tous les obstacles.*

*À ma binome **Manal** ton engagement et ta persévérance ont été des atouts précieux qui ont enrichi notre projet.*

À mes chers amis

Walid

Table des matières

Table des matières	i
Liste des tableaux	v
Liste des figures	vi
Liste des abréviations	ix
Introduction générale	1
1 Présentation de l'organisme d'accueil et méthodologie de développement	3
1.1 Introduction	3
1.2 Organisme d'accueil	3
1.2.1 Historique	3
1.2.2 Organigramme général	3
1.2.3 Etude de l'existant	5
1.3 Problématique	5
1.4 Solution proposée	6
1.5 Méthodologie de développement à adopter	6
1.5.1 Définition de la méthode SCRUM	6
1.5.2 Concepts de base de la méthode SCRUM	6
1.5.3 Cycle de vie de SCRUM	9
1.6 Langages de modélisation UML	10
1.6.1 Définition d'UML	10
1.6.2 Les diagrammes d'UML	10
1.7 Architecture de l'application	12

1.8	Environnement et langage de programmation	13
1.8.1	Environnement de développement	13
1.8.2	Les langages de programmation	14
1.9	Conclusion	16
2	Les services Web	17
2.1	Introduction	17
2.2	Définition	17
2.3	Architecture des services Web	18
2.3.1	Architecture de référence	18
2.3.2	Architecture étendu	19
2.4	Technologies des services Web	20
2.4.1	Le langage XML	20
2.4.2	Le protocole SOAP	20
2.4.3	Le langage WSDL	21
2.4.4	L'annuaire UDDI	21
2.5	Avantages et inconvénients des services Web	21
2.5.1	Avantages des services Web	21
2.5.2	Inconvénients des services Web	21
2.6	Conclusion	22
3	Préparation de projet	23
3.1	Introduction	23
3.2	Spring 0	23
3.3	Spécification des besoins	23
3.3.1	Identification des acteurs du système	24
3.3.2	Diagramme de cas d'utilisation	26
3.4	Product Backlog	28
3.4.1	Planification des releases	30
3.5	Conclusion	30

4	Conception et réalisation	31
4.1	Introduction	31
4.2	Développement du Sprint 01 « Authentification, inscription » :	31
4.2.1	SCRUM Board (task board)	31
4.2.2	User story du sprint 01	32
4.2.3	Backlog du sprint 01	32
4.2.4	Analyse	33
4.2.5	Conception	34
4.2.6	Réalisation	37
4.3	Développement du Sprint 02 « Gérer le profil, Gérer les comptes » :	41
4.3.1	SCRUM Board (task board)	41
4.3.2	User story du sprint 02	41
4.3.3	Backlog du sprint 02	42
4.3.4	Analyse	42
4.3.5	Conception	44
4.3.6	Réalisation	45
4.4	Développement du Sprint 03 « Gérer les propositions, Gérer les événements, Gérer les services » :	47
4.4.1	SCRUM Board (task board)	47
4.4.2	User story du sprint 03	47
4.4.3	Backlog du sprint 03	48
4.4.4	Analyse	49
4.4.5	Conception	51
4.4.6	Réalisation	56
4.5	Développement du Sprint 04 « Consulter des proposition, poster un commentaire » :	61
4.5.1	SCRUM Board (task board)	61
4.5.2	User story du sprint 04	61
4.5.3	Backlog du sprint 04	61
4.5.4	Analyse	62
4.5.5	Conception	64
4.5.6	Réalisation	65

4.6	Dictionnaire de donnée	67
4.7	Diagramme de classe	68
4.8	Schéma relationnel	70
4.8.1	Définition	70
4.8.2	Règles de passage du diagramme de classe au modèle relationnel	70
4.8.3	Modèle relationnel de donnée	71
4.9	Conclusion	71

Conclusion générale **72**

Liste des tableaux

1.1	Les logiciels utilisés par DECIBEL EVENTS pour la gestion d'événements	5
3.1	Les rôles de SCRUM	23
3.2	Product Backlog	29
4.1	User story du sprint 01	32
4.2	Backlog du sprint 01	32
4.4	Description textuelle « Inscription »	34
4.5	User story du sprint 02	41
4.6	Backlog du sprint 02	42
4.7	Description textuelle du cas « Gérer profil »	43
4.8	Description textuelle du cas « Gérer les comptes »	43
4.9	User story du sprint 03	48
4.10	Backlog du sprint 03	48
4.11	Description textuelle du cas « Gérer les événements »	50
4.12	User story du sprint 04	61
4.13	Backlog du sprint 04	62
4.14	Description textuelle du cas« Consulter proposition »	63
4.15	Description textuelle du cas« Poster un commentaire »	63
4.16	Glossaire de données	68

Table des figures

1.1	Organigramme de DECIBEL EVENTS	4
1.2	Les rôles du projet SCRUM [2]	7
1.3	Les cérémonies de la méthode SCRUM [3]	8
1.4	Les artéfacts de la méthode SCRUM [3]	9
1.5	Le processus SCRUM [4]	10
1.6	Diagrammes statiques d'UML	11
1.7	Diagrammes dynamiques d'UML	11
1.8	L'architecture MVC	12
2.1	Les langages de programmation dans les services Web [6]	18
2.2	Architecture de référence de service Web [7]	19
2.3	Architecture en pile [8]	20
3.1	Diagramme de contexte	26
3.2	Diagramme de cas d'utilisation global	27
3.3	Schéma de Release	30
4.1	SCRUM board du sprint 01	31
4.2	Diagramme de cas d'utilisation du sprint 01	33
4.3	Diagramme de séquence « Authentification »	35
4.4	Diagramme de séquence « Inscription »	36
4.5	Interface « Authentification »	37
4.6	Interface « Mot de passe oublié oublié »	38
4.7	Interface « Réinitialiser »	38
4.8	Interface « Inscription »	39

4.9	Interface « Reçois d'un mail d'activation du compte »	40
4.10	Interface « Le mail d'activation »	40
4.11	SCRUM board du sprint 02	41
4.12	Diagramme de cas d'utilisation du sprint 02	42
4.13	Diagramme de séquence « Gérer profil »	44
4.14	Diagramme de séquence « Gérer les comptes »	45
4.15	Interface « Gérer profil »	45
4.16	Interface « Gérer profil(changer le mot de passe) »	46
4.17	Interface « Gérer les comptes »	46
4.18	SCRUM board du sprint 03	47
4.19	Diagramme de cas d'utilisation du sprint 03	49
4.20	Diagramme de séquence « Gérer événement »	51
4.21	Diagramme de séquence « Ajouter un événement »	52
4.22	Diagramme de séquence « Supprimer un événement »	53
4.23	Diagramme de séquence « Rechercher un événement »	54
4.24	Diagramme de séquence « Modifier un événement »	55
4.25	Interface « Gérer les événements »	56
4.26	Interface « Ajouter un événement lors les champs sont vide »	57
4.27	Interface « Ajouter un événement »	57
4.28	Interface « Modifier un événemnt »	58
4.29	Interface « Supprimer un événement »	58
4.30	Interface « Gérer les services »	59
4.31	Interface « La liste des services imprimé »	59
4.32	Interface « Gérer les propositions »	60
4.33	SCRUM board du sprint 04	61
4.34	Diagramme de cas d'utilisation du sprint 04	62
4.35	Diagramme de séquence « Consulter proposition »	64
4.36	Interface « Consulter proposition »	65
4.37	Interface « Voir les détails d'une proposition »	66
4.38	Interface « La liste des favoris des propositions »	66
4.39	Interface « Poster un commentaire »	67

4.40	Diagramme de classe	69
4.41	Interface « L'accueil de notre site »	76
4.42	Interface « À-propos »	77
4.43	Interface « Contact »	78
4.44	Interface « Galerie »	79
4.45	Interface « Accueil admin »	80
4.46	Interface « Matériel de sonorisation »	80
4.47	Interface « Message d'erreur sur l'absence d'un matériel ou table vide »	81
4.48	Interface « Marquer un évènement sur le calendrier »	81
4.49	Interface « Déconnexion »	82

Liste des abréviations

COM : Component Object Model

CORBA : Common Object Request Broker Architecture

DCOM : Distributed Component Object Model

EDI : Echange de Données Informatisées

EAI : Entreprise Application Integration

HTML : Hyper Text Markup Language

HTTP : Hyper Text Transfer Protocol

IBM : International Business Machines Corporation

OASIS : Organization for the Advancement of Structured Information Standards

POP : Post Office Protocol

REST : Representational State Transfer

RMI : Remote Method Invocation

RPC : Appel procédure lointaine

SGML : Standard Generalized Markup Language

SOAP : Simple Object Access Protocol

SOA : Service Oriented Architecture

SW : Service Web

UDDI : Universal Description Discovery and Integration

URI : Uniform Resource Identifier

W3C : World Wide Web Consortium

WSDL : Web Services Description Language

XML : eXtensible Markup Language

Introduction générale

De nos jours, les technologies de l'information et de la communication ont connu une prolifération sans précédent, transformant radicalement la façon dont les entreprises et les organisations opèrent. Les applications web service ont joué un rôle essentiel dans cette révolution numérique en offrant des solutions innovantes pour faciliter les opérations quotidiennes et améliorer la productivité. L'un des domaines qui bénéficie grandement de ces avancées technologiques est celui de l'organisation des événements.

L'organisation d'événements, qu'ils soient professionnels ou particuliers (Séminaires de formation, Conférences et conventions d'entreprise, mariages, anniversaires...etc) nécessite une coordination minutieuse de diverses tâches telles que la planification, la gestion des participants, la réservation de lieux, la communication et bien d'autres tâches encore. L'entreprise DECIBEL EVENTS qui est une société de service qui prend en charge les préparatifs et l'organisation matérielle et logistique d'événement, se retrouve confronté au problème de gestion des opérations qui est souvent effectuées manuellement, ce qui peut entraîner des retards, des erreurs et une gestion inefficace.

Dans ce contexte, la création d'une application web service dédiée à l'organisation des événements présente une opportunité majeure pour simplifier et rationaliser ces services. Une telle application permettrait aux organisateurs d'événements de gérer tous les aspects liés à leur événement de manière centralisée, efficace et sécurisée. Elle offrirait également la possibilité aux participants de s'inscrire, de consulter les détails de l'événement, d'interagir avec d'autres participants et de recevoir des informations en temps réel.

Ce mémoire de fin d'études vise à apporter une contribution significative dans le domaine de l'organisation des événements en proposant une solution moderne et efficace grâce à une application web service couvrant le processus d'organisation des événements. Le but de notre application est de permettre aux entreprises, aux organisations et les organisateurs d'événements à améliorer leur productivité, d'offrir une meilleure expérience aux participants d'événements, de relever les défis de l'organisation d'événements à l'ère numérique.

Le mémoire organisé autour de quatre chapitres :

- Le chapitre 1 intitulé "Présentation de l'organisme d'accueil et méthodologie de développement", présente des informations générales sur l'organisme d'accueil du projet, ainsi que la méthodologie de conception adoptée.
- Le chapitre 2 intitulé "Les services web", aborde les concepts fondamentaux des services web ainsi que les différentes technologies associées à leur mise en œuvre
- Le chapitre 3 intitulé "Sprint zéro :Pilotage avec SCRUM", définit les principaux acteurs impliqués dans notre application, ainsi que les besoins fonctionnels et non fonctionnels et le pilotage de méthode SCRUM
- Le chapitre 4 intitulé "Conception et réalisation" fournit une description détaillée de notre utilisation de la méthode de SCRUM, en mettant en évidence les différentes étapes de chaque sprint

Présentation de l'organisme d'accueil et méthodologie de développement

1.1 Introduction

Dans ce chapitre, nous allons présenter les différents services de l'entreprise DECIBEL EVENTS et leur historique. Ensuite, nous allons exposer la problématique et l'objectif de notre application. Après, nous passons à la présentation de la méthodologie de développement SCRUM et les langages de modélisation ainsi que l'architecture de notre application. Nous terminons ce chapitre par la présentation de l'environnement de développement et langages de programmation.

1.2 Organisme d'accueil

DECIBEL EVENTS est une société de service qui prend en charge les préparations et l'organisation matérielle et logistique d'événementiels particuliers ou professionnels tels que : mariage, réception, séminaire, inauguration, diffusions directes des interventions chirurgicales...etc.

1.2.1 Historique

DECIBEL EVENTS était initialement spécialisée dans la sonorisation, et elle a par la suite élargi son champ d'activité pour inclure d'autres services tels que le DJ, la photographie, la vidéo et l'éclairage. DECIBEL EVENTS propose désormais une gamme complète de services allant de la planification à l'organisation et la mise en action d'événements pour une clientèle diversifiée, incluant les entreprises, les associations, les collectivités, les conférences, la diffusion d'interventions chirurgicales et les masters classes. DECIBEL ne se limite pas aux événements professionnels et propose également ses services aux particuliers pour l'organisation d'événements tels que les mariages, les fiançailles et les circoncisions avec un effectif de 20 à 30 personnes.

1.2.2 Organigramme général

DECIBEL est composé des cinq services suivant :

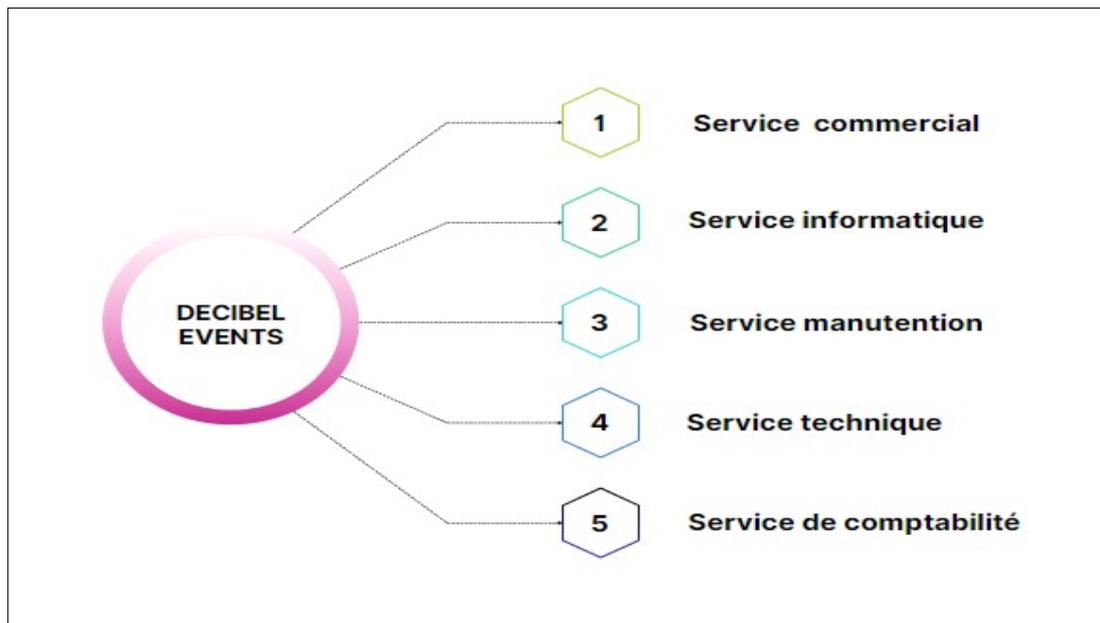


FIGURE 1.1 – Organigramme de DECIBEL EVENTS

Service commercial :

Ce service est chargé de planifier, d'organiser et de mettre en œuvre toutes les actions nécessaires pour assurer une gestion optimale de l'entreprise. Ses principales tâches comprennent :

- La gestion des ressources humaines.
- La gestion des moyens (achat, vente, location, réparations ...).
- La négociation avec le client (premier contact avec le client, prise de rendez-vous, prés-réservation, étude du projet ...).

Service informatique :

Ce service planifie des plans de maintenance et définit les procédures de qualité et de sécurité des systèmes d'information pour un résultat plus fiable et sécurisant.

Service manutention :

Ce service gère la logistique et l'approvisionnement en matériel. Cela comprend les opérations suivantes :

- Réparation des articles endommagés.
- La surveillance des stocks.
- La livraison des marchandises et la réception de nouveaux articles.

Service technique :

Ce service s'occupe du développement des logiciels et des tests ainsi que la réparation et la manutention des équipements.

Service de comptabilité :

Ce service prend en charge les opérations suivantes : la tenue de la comptabilité, la gestion de la trésorerie (dépenses, recettes et placements) ainsi que la tenue des inventaires et le contrôle de gestion (comptabilité et contrôle budgétaire et chiffre d'affaire).

1.2.3 Etude de l'existant

Les équipements informatique utilisés les informaticiens de DECIBEL EVENTS sont les suivants :

1. Matériel

- Des ordinateurs caractérisés par :

- Processeur Apple M1 3,2 GHZ.
- Capacité mémoire 8Go de RAM
- Système d'exploitation MacOS.

- Des imprimantes de type :

- Canon Pixma TS6350a
- HP Deskjet 3760

2. Logiciel

Domaine	Logiciel
Production vidéo	OBS
Visio-conférence + streaming	VMIX
Création vidéo	Adobe after effect
Montage vidéo	Filmoora
Infographie	Adobe Photoshop
Retouche photo	Adobe lightroom

TABLE 1.1 – Les logiciels utilisés par DECIBEL EVENTS pour la gestion d'événements

1.3 Problématique

Lors de notre séjour à l'entreprise, nous avons remarqué que l'organisation des évènements se fait principalement de façon manuelle et cela peut engendrer une perte d'informations. Nous avons aussi remarqué que l'entreprise ne dispose pas d'un site web ou d'application permettant la gestion de l'organisation des différents évènements tel que : mariage, séminaire, conférence,...etc. La présence physique des clients était essentielle pour qu'ils puissent avoir un accès direct aux informations, interagir avec l'organisateur d'évènement et prendre des décisions éclairées.

1.4 Solution proposée

Au vu des problèmes cités suite à l'analyse de l'environnement de travail et la variété des matériels utilisés, nous avons décidé de développer une application web service qui est utilisable sur n'importe quel système d'exploitation, notre application est une compositions de plusieurs services web que nous avons dû développer nous même tel que Userservice , Mailservice, Cartservice et d'autre service. Notre solution vise d'atteindre les objectifs suivants :

- Améliorer la communication avec les clients dans le but de mieux capturer ses besoins et les satisfaire.
- Afficher tous les services de l'entreprise en un seul écran.
- Garantir une diffusion plus large d'information.
- Intégrer un module d'organisation en ligne pour plus de flexibilité.

1.5 Méthodologie de développement à adopter

Dans le but de produire une application en respectant les délais qui nous sont imposés tout en satisfaisant les exigences de notre client,nous avons opté pour les méthodes agile qui sont des méthodologies essentiellement dédiés à la gestion de projet informatique reposant sur le cycle de développement itératif et incrémental mettent en avant la collaboration entre des équipes auto-organisées et pluri-disciplinaires et leurs clients [17].Nous avons choisi la méthode SCRUM car son processus permet d'améliorer la fluidité des livraisons et d'assurer une plus grande souplesse et réactivité lors du développement de notre application.

1.5.1 Définition de la méthode SCRUM

SCRUM est un cadre de travail « Framework » de gestion de projet.Le terme est inspiré du terme SCRUM en rugby qui désigne une mêlée. C'est une technique qui remet une équipe sur de bonne rails par un effort collectif, dans le but d'améliorer sa productivité. Cette méthodologie se caractérise par trois éléments principaux : la transparence, l'inspection et l'adaptation [1].

1.5.2 Concepts de base de la méthode SCRUM

Ci-dessous se trouvent les éléments essentiels pour mettre en pratique la méthode SCRUM

a. Roles

Les roles de méthodes SCRUM sont les suivants : [2] :

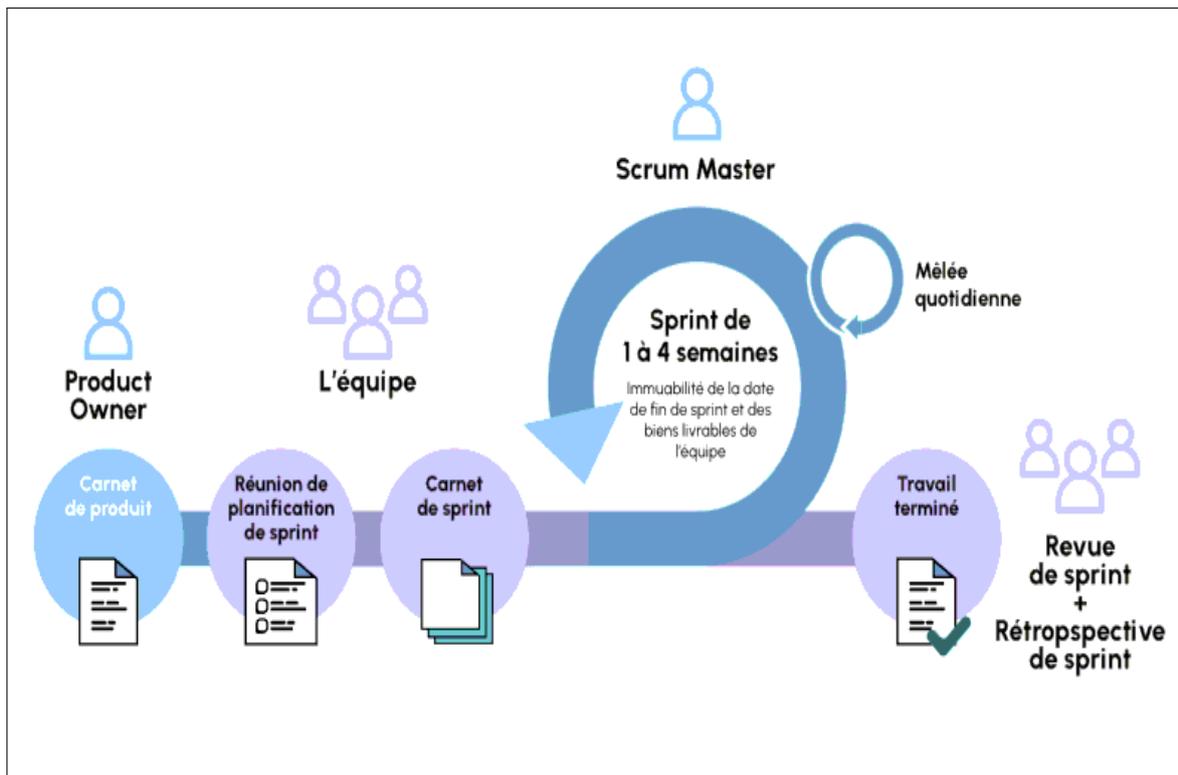


FIGURE 1.2 – Les rôles du projet SCRUM [2]

Propriétaire du produit (Product OWNER)

Le Product Owner est responsable de la communication avec l'équipe de développement et représente les clients. Il détermine les fonctionnalités prioritaires à développer ou à corriger, valide les fonctionnalités terminées et gère le Product Backlog [2].

Le maître de mêlée (Le SCRUM master)

Le SCRUM Master est responsable de la mise en œuvre de la méthode SCRUM. Il s'assure que l'équipe est opérationnelle et productive et facilite la collaboration entre tous les rôles et les fonctions [2].

L'équipe de développement (Développment team)

Les membres de l'équipe de développement créent l'incrément, ils sont composés de développeurs, testeurs, architectes, designers et de toute autre personne nécessaire à la réalisation du projet [2].

b. Evénements

SCRUM propose différentes cérémonies « réunion » pour rythmer l'ensemble des sprints comme présenté ci-dessous [3] :

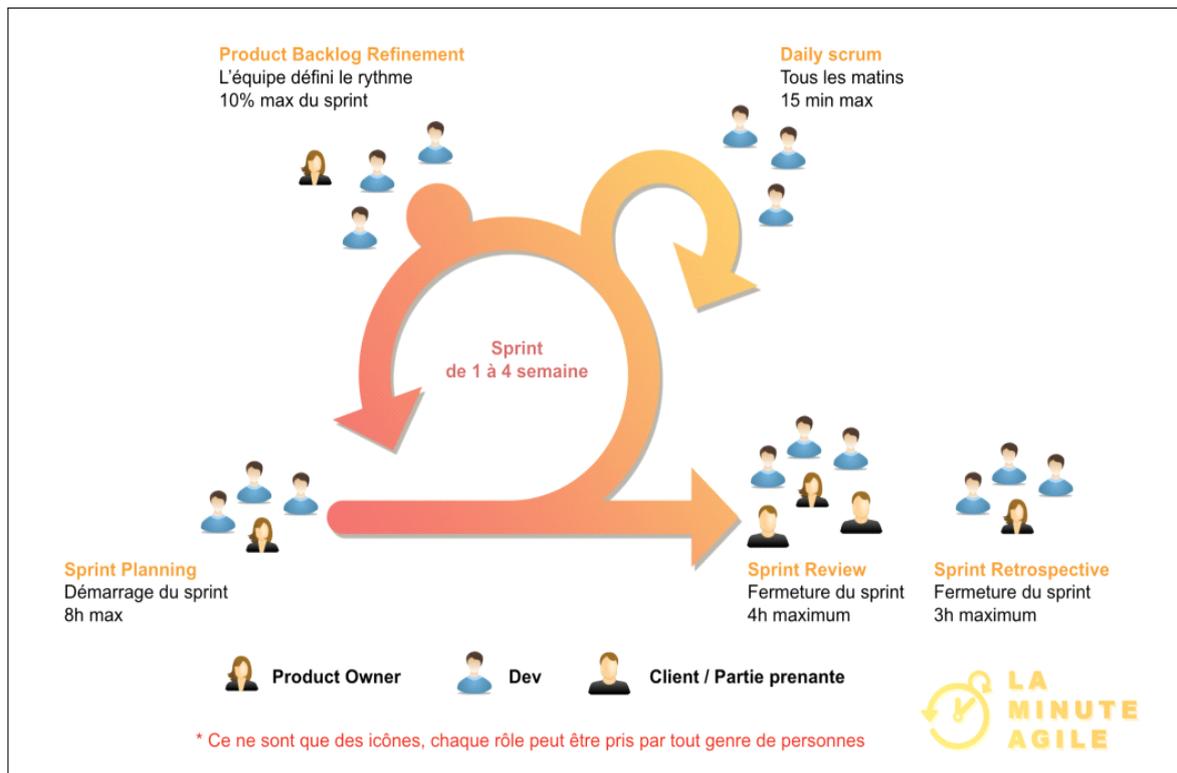


FIGURE 1.3 – Les cérémonies de la méthode SCRUM [3]

Sprint :

Un sprint est une période de temps dont la durée maximum est d'un mois à la fin de chaque sprint une version terminée, utilisable et potentiellement livrable du logiciel est créé [3]. Les sprints sont constituée de :

- Sprint planning (Planification de sprint)
- Daily SCRUM (Mêlée quotidienne)
- Sprint review (Revue du sprint)
- Sprint Retrospective (Rétrospective de sprint)

c. Artéfacts

Les artéfacts désignent les éléments matériels et visibles qui sont employés dans le cadre de la méthodologie SCRUM. Leur rôle est de favoriser la transparence, la collaboration et la gestion efficace du travail au sein de l'équipe de développement, composée de [3] :

- Carnet du produit (Product backlog)
- Carnet de sprint (Sprint backlog).
- Incrément de produit (Product increment).

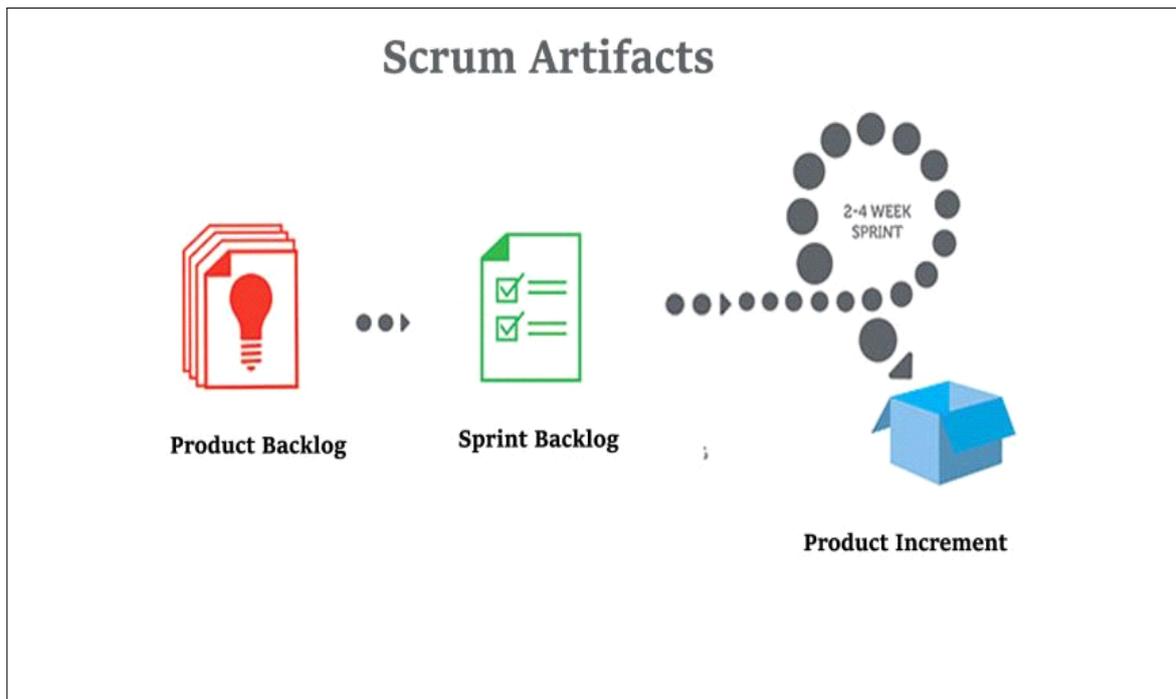


FIGURE 1.4 – Les artéfacts de la méthode SCRUM [3]

d. L'histoire de l'utilisateur (User stories)

Les « users stories » sont les descriptions des besoins de l'utilisateur en termes de fonctionnalités. C'est le Product Owner qui est responsable de la collecte des besoins des utilisateurs et de leur traduction en « user story » compréhensible par l'équipe [18].

1.5.3 Cycle de vie de SCRUM

Le principe de la méthode agile SCRUM est de concentrer l'équipe de développement sur une série de fonctionnalités à réaliser itérativement. Ces itérations peuvent s'étaler sur une durée de deux à quatre semaines, appelées des Sprints. Chaque Sprint doit aboutir à la livraison d'un produit partiel (une partie du logiciel) [4].

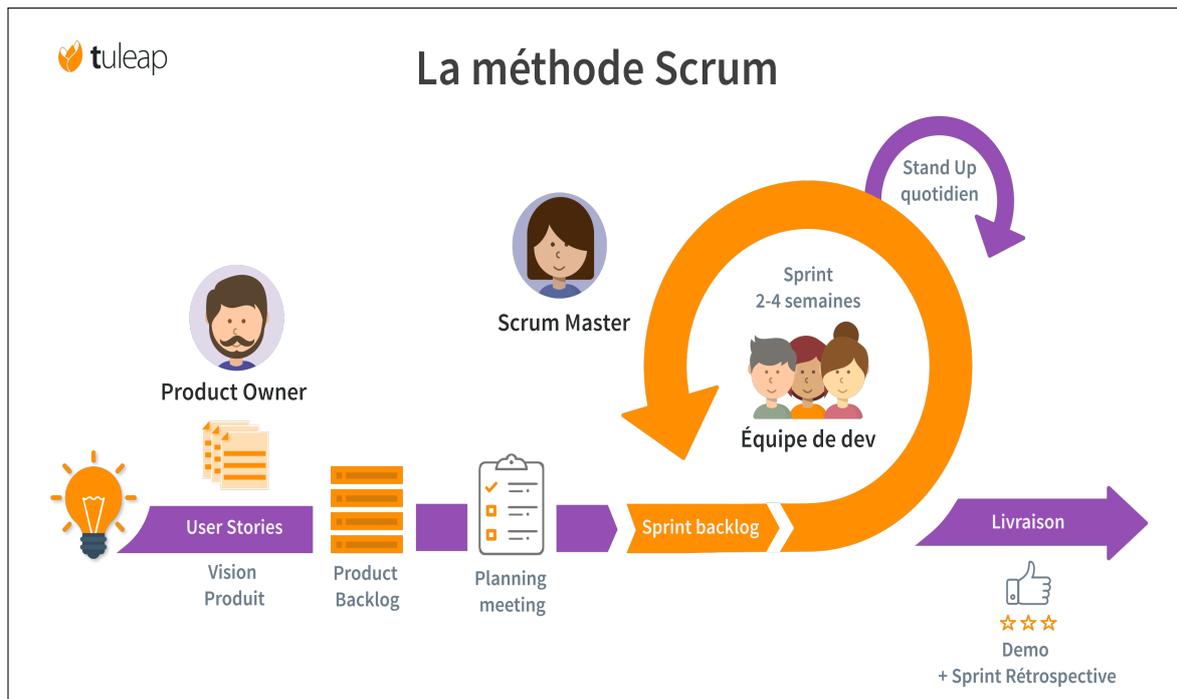


FIGURE 1.5 – Le processus SCRUM [4]

Comme indiqué dans la figure ci-dessus, pour la mise en place de la méthode SCRUM, le nombre maximal de fonctionnalités à réaliser doit d'abord être identifié afin de créer le Backlog du produit. Deuxièmement, il faudra définir les priorités relatives aux fonctionnalités et choisir celles qui seront réalisées à chaque itération.

Ensuite, l'idée serait de focaliser l'équipe sur l'ensemble des fonctionnalités à accomplir par le biais de Sprints, qui sont des itérations

1.6 Langages de modélisation UML

La méthode SCRUM repose sur l'utilisation de UML en tant que langage de modélisation, donc dans cette section nous allons définir le langage de modélisation UML ainsi que leurs différents diagrammes.

1.6.1 Définition d'UML

UML est un langage de modélisation graphique et textuelle destiné à définir des besoins, documenter des systèmes, concevoir des solutions et communiquer des points de vue. Il permet de faire une modélisation des aspects structurels et comportementaux d'un système [5].

1.6.2 Les diagrammes d'UML

La version UML 2.0 est basée sur 13 diagrammes, divisés en deux catégories de diagrammes structurels et comportementaux qui sont définies dans les figures qui suivent :

Diagrammes structurels (vue statique)d'UML

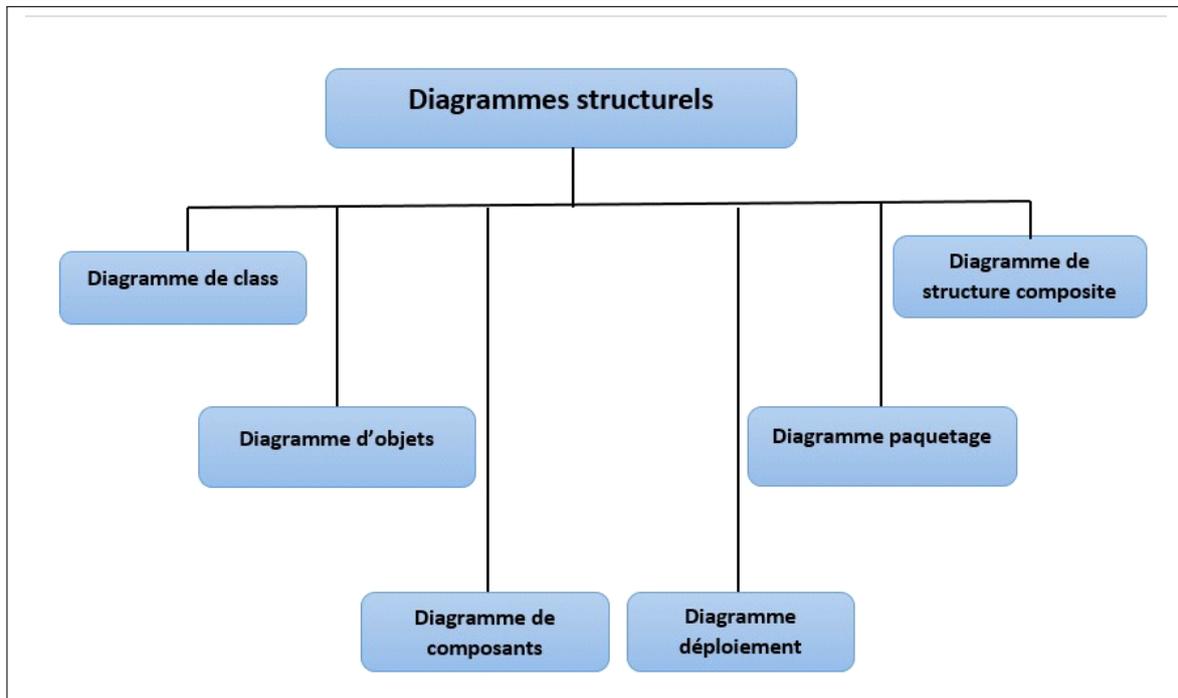


FIGURE 1.6 – Diagrammes statiques d'UML

Diagrammes comportementaux (vue dynamique)d'UML

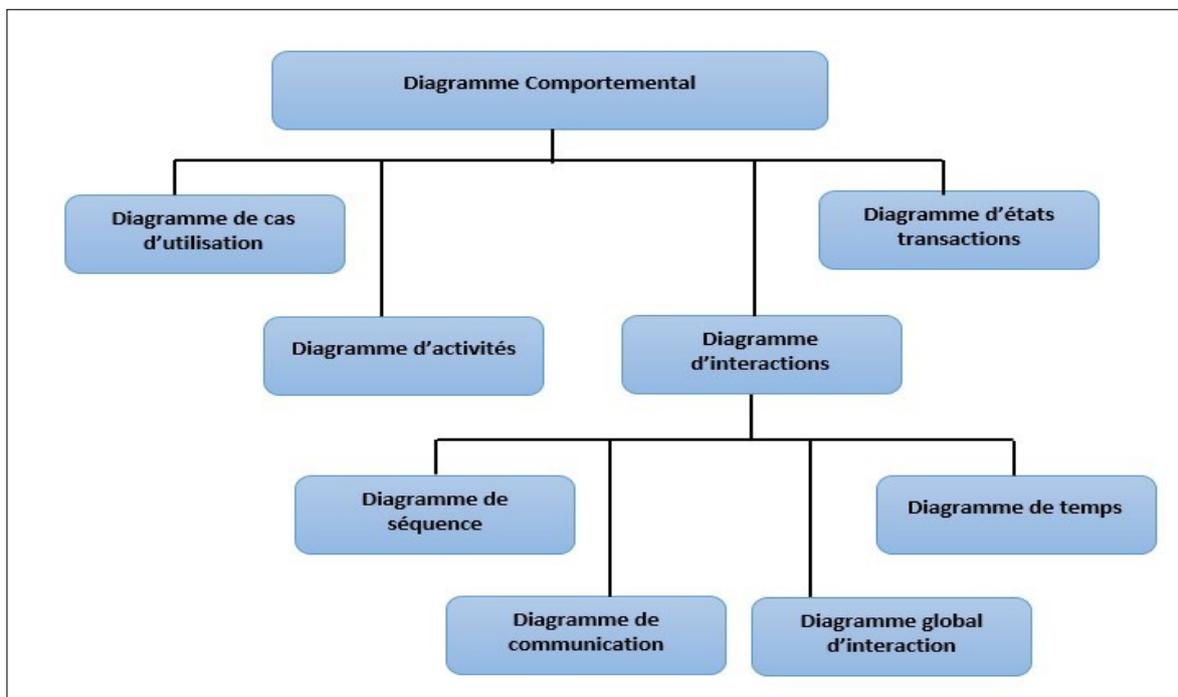


FIGURE 1.7 – Diagrammes dynamiques d'UML

Dans notre projet, nous avons utilisé quatre digrammes qui sont les plus utilisés lors de la modélisation d'un logiciel à suivre (diagramme de cas d'utilisation, de séquence, de classes, de contexte).

1.7 Architecture de l'application

Le modèle MVC (Modèle-Vue-Contrôleur) est un pattern de conception couramment utilisé pour développer des applications Web. Il permet de séparer les différentes responsabilités d'une application en trois composants distincts : le modèle, la vue et le contrôleur.

Lorsqu'il est combiné avec Spring Boot, qui est un framework Java pour le développement rapide d'applications, le modèle MVC devient une approche populaire pour construire des applications Web robustes et évolutives. [19].

On va opter pour l'architecture MVC lors du développement de notre application car elle offre une séparation claire des responsabilités et elle nous permet de favoriser la réutilisabilité du code, à permettre l'évolutivité du système, à faciliter les tests et la collaboration entre les développeurs. Ces atouts ont indéniablement amélioré la qualité, la maintenabilité et la flexibilité de mon application.

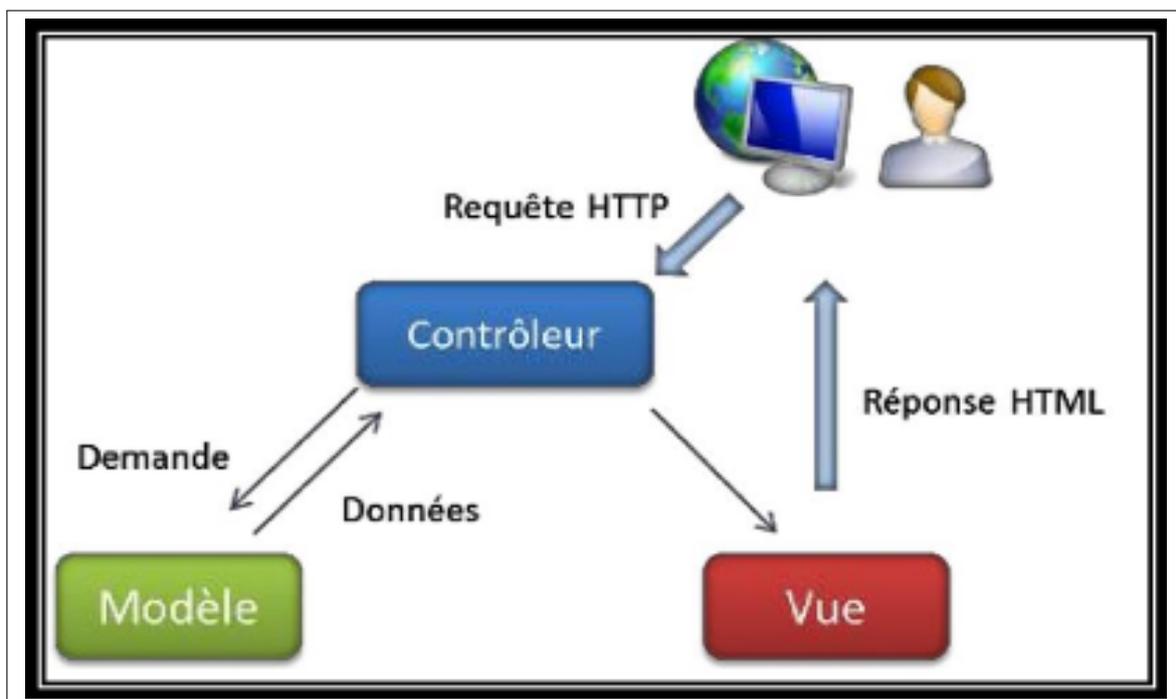


FIGURE 1.8 – L'architecture MVC

Le modèle MVC comprend trois parties, qui sont :

- **Le modèle (model)** : correspond à la partie de l'application qui gère le logique métier. Il récupère, traite et manipule les données selon les concepts propres à l'application.
- **La vue (view)** : La vue est responsable de l'affichage des données à l'utilisateur. Elle peut être implémentée à l'aide de technologies telles que Thymeleaf, JSP (JavaServer Pages) ou des

frameworks frontaux comme Angular ou React. La vue récupère les données nécessaires du contrôleur et les présente à l'utilisateur.

- **Le contrôleur (controller) :** Le contrôleur est responsable de la gestion des requêtes HTTP et de la coordination entre le modèle et la vue.

1.8 Environnement et langage de programmation

Dans ce qui suit, nous définirons l'environnement de développement (Spring Tool Suite, SCRUMblr, Wamp...etc), ainsi que les langages de programmations utilisés lors du développement de notre application.

1.8.1 Environnement de développement

Spring Tool Suite (STS)



STS est gratuit, open source et optimisé par VMware. Il offre une prise en charge de classe mondiale pour le développement d'applications d'entreprise basées sur Spring [20].

Php my admin

PhpMyAdmin est un outil logiciel gratuit écrit en PHP , destiné à gérer l'administration de MySQL sur le Web. phpMyAdmin prend en charge un large éventail d'opérations sur MySQL et MariaDB, traduit en 72 langues et prend en charge les langues LTR et RTL [21].



MYSQL



MySQL est un système de gestion de bases de données relationnelles (SGBDR) open source développé par MySQL AB (maintenant Oracle Corporation) en 1995 [22].

Wampserver

WampServer est une plate-forme de développement Web sous Windows pour des applications Web dynamiques à l'aide du serveur Apache2, du langage de scripts PHP et d'une base de données MySQL [23].



Eclipse Papyrus



Eclipse Papyrus est un logiciel de modélisation UML et tout type de modèle EMF (Eclipse Modeling Framework), développé par la fondation Eclipse [24].

Adobe Illustrator

Un outil de conception et d'édition graphique vectorielle utilisée, entre autres, pour les logos et la composition [25].



1.8.2 Les langages de programmation

HTML (HyperText Markup Language)



Un langage de balisage qui permet d'écrire de l'hypertexte. « Hypertexte » fait référence aux liens qui relient les pages Web les unes aux autres, soit au sein d'un même site Web, soit entre des sites Web [27].

CSS (Cascading Style Sheets)

Est un langage de feuille de style utilisé pour décrire la présentation d'un document écrit en HTML ou XML. CSS décrit comment les éléments doivent être rendus à l'écran, sur papier, dans la parole ou sur d'autres supports [28].



JavaScript



JavaScript est un langage de script léger, orienté objet, principalement connu comme le langage de script des pages web, permet de créer du contenu mis à jour de façon dynamique, de contrôler le contenu multimédia, d'animer des images...etc [29].

Java



Java est un langage de programmation de haut niveau orienté objet créé par James Gosling et Patrick Naughton, employés de Sun Microsystems, avec le soutien de Bill Joy [31].

JPA (Java Persistence API)

JPA est une interface de programmation Java permettant aux développeurs d'organiser des données relationnelles dans des applications utilisant la plateforme Java [32].



Les frameworks :

Springboot



Springboot est un Framework open-source (gratuit) développé par Pivotal destiné aux applications java (J2EE), il facilite le développement et les tests [33].

Spring Security

Spring Security est un framework Java/Java EE qui propose des fonctionnalités de sécurité pour les applications d'entreprise. Il inclut des fonctionnalités telles que l'authentification et l'autorisation. Initié par Ben Alex en 2003 sous le nom de "Acegi Security", le projet a été publié en mars 2004 sous la licence Apache [33].



Bootstrap



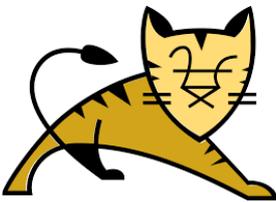
Bootstrap est un framework de développement frontal gratuit et open source utilisé pour la création de sites Web et d'applications Web [34].

Thymleaf

Thymeleaf est un moteur de Template, sous licence Apache 2.0, écrit en Java pouvant générer du XML/XHTML/HTML5, utilisé dans les applications web basées sur le modèle MVC [35].



Tomcat



Tomcat est un serveur d'applications. Tomcat est une implémentation open source des spécifications Jakarta Servlet, qui permet d'écouter toutes les demandes des clients et charger la classe de servlet correspondante et il utilise le mappage de servlet pour gérer les demandes des clients [36].

Service REST (representational state transfer)

REST est un style d'architecture logicielle qui définit un ensemble de contraintes pour la création de services Web [37].



1.9 Conclusion

Ce premier chapitre fournit une description globale de l'organisme d'accueil. Nous avons défini la problématique et la solution proposée pour le résoudre à savoir une application composée de plusieurs services Web.

Nous avons passés par la suite à la présentation de la méthodologie de développement SCRUM suivi lors de la réalisation de notre application.

Nous passons après à la présentation du langage de modélisation UML ainsi que l'architecteur de l'application et nous concluons enfin ce chapitre par la présentation des outils et langages de programmations utilisés lors de la réalisation de notre application.

Le chapitre suivant abordera les concepts fondamentaux en relation avec les services Web.

Les services Web

2.1 Introduction

Les services Web permettent de façon innovante de créer des applications qui répondent aux besoins d'utilisateurs et garantissent une interopérabilité optimale. Ce chapitre aborde les concepts fondamentaux des services Web, en commençant par une définition de cette technologie, suivait d'une présentation des différentes technologies associées à leur mise en œuvre. Enfin, nous citons les avantages et les inconvénients de l'utilisation de cette technologie.

2.2 Définition

Un service Web est un système logiciel qui repose sur l'architecture orientée services (SOA) et qui est identifié par une URI spécifique. Ses interfaces publiques et ses « bindings » sont définis et décrits en utilisant le langage de balisage XML. En d'autres termes, un service Web est une application accessible sur le Web, qui peut être programmée dans différents langages tels que C, PHP, Java et qui peut être exécutée sur différentes plateformes telles que Linux, Windows, MacOS. Les services Web utilisent les protocoles Internet pour la communication en plus d'un langage standard pour décrire son interface [6].



FIGURE 2.1 – Les langages de programmation dans les services Web [6]

2.3 Architecture des services Web

On peut distinguer deux types d'architectures pour les services Web : une qui se concentre sur les références, et l'autre qui se focalise sur l'extension des fonctionnalités [7].

2.3.1 Architecture de référence

Le modèle de services Web implique trois rôles interagissant entre eux : le fournisseur de service, l'annuaire de service et le client.

- **Les fournisseurs de service** : crée le service Web et publie toutes ces caractéristiques.
- **L'annuaire de service** : l'annuaire de services Web fournit les interfaces d'accès au service, ainsi que les informations sur le contrat et l'architecture utilisés.
- **Le consommateur (ou client)** : Le client accède à l'annuaire pour rechercher les services Web nécessaires et pour déterminer les normes à respecter lors de l'interaction avec ces services.

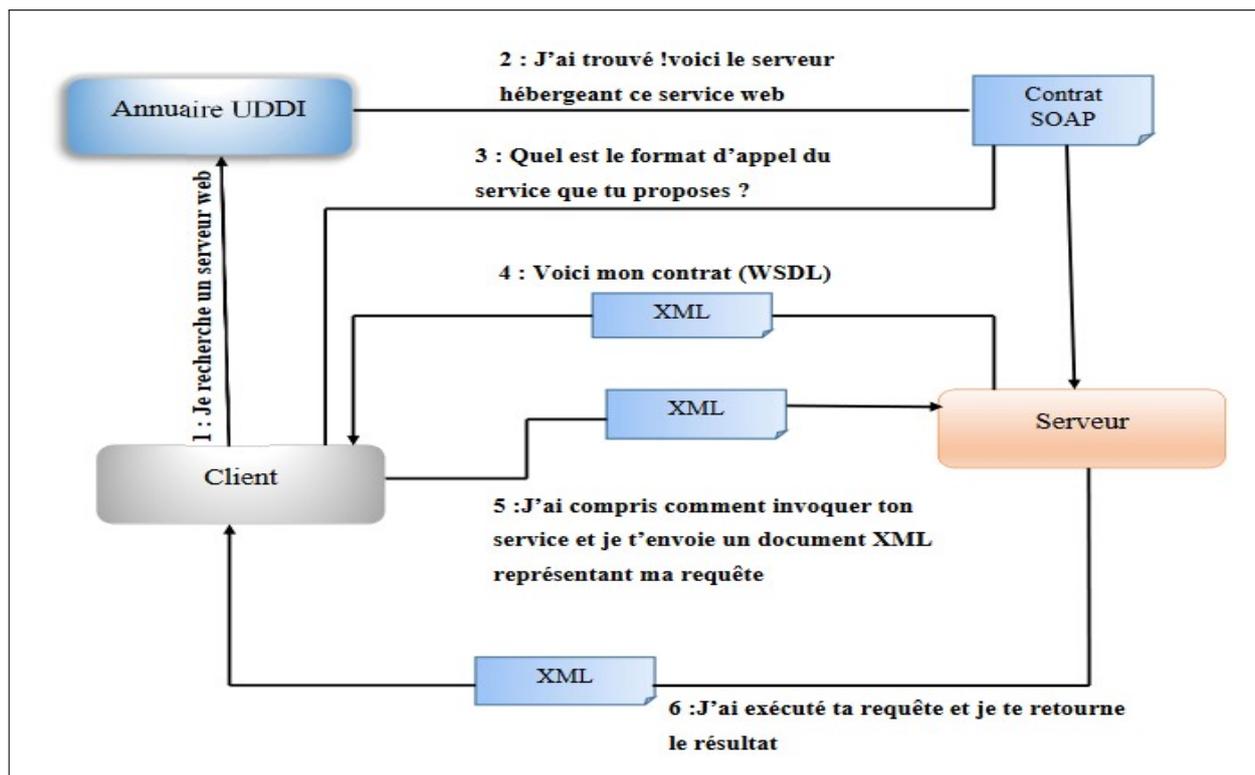


FIGURE 2.2 – Architecture de référence de service Web [7]

2.3.2 Architecture étendu

On peut décomposer l'architecture étendue en trois types de couches distinctes : [7] [8]

- **L'infrastructure de base** : elle est constituée de trois couches, qui s'appuient sur les standards émergent, tel que SOAP ou REST, pour l'échange de messages, WSDL pour la description des services Web et UDDI pour la publication. Ainsi, elle établit les bases techniques fondamentales de l'architecture de référence.
- **La couche Business Processus** : L'architecture étendue facilite l'intégration et l'utilisation efficace des services Web dans le domaine du commerce électronique.
- **La couche transversale** : (sécurité, transaction, Administration et QOS1) cette couche rendent l'utilisation des services Web viable dans le domaine industriel.

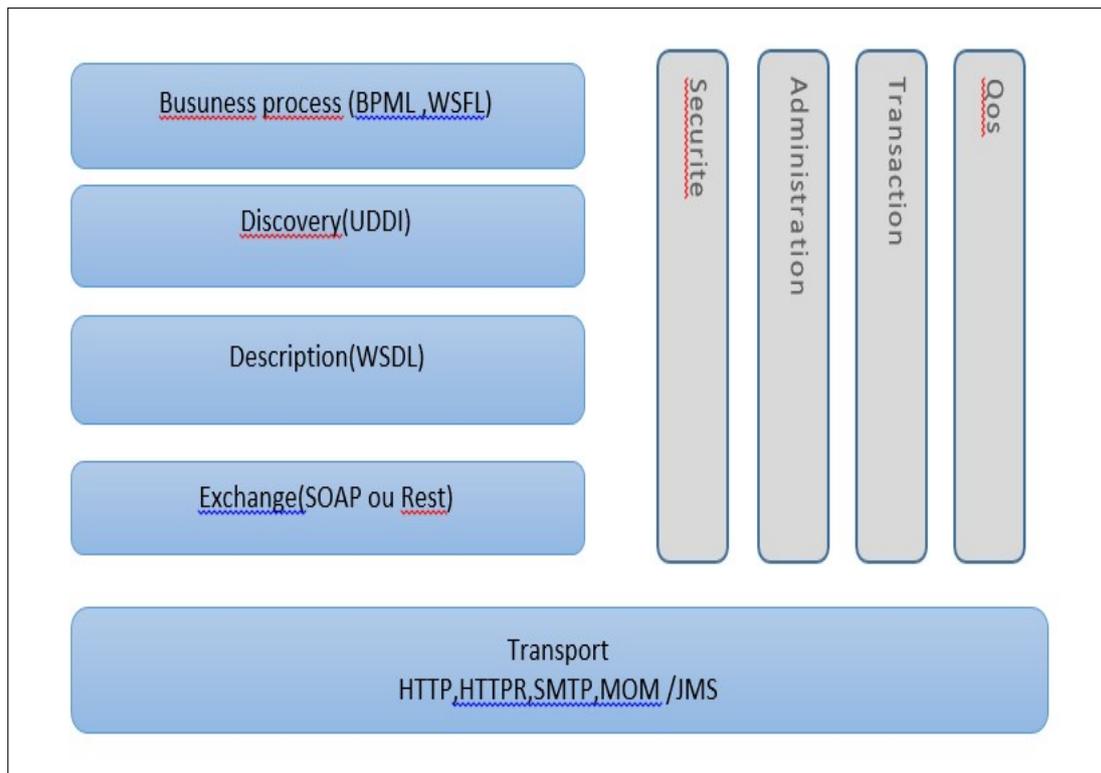


FIGURE 2.3 – Architecture en pile [8]

2.4 Technologies des services Web

Les services Web communiquent via un ensemble de technologies fondamentales qui partagent une architecture commune dont XML, SOAP, WSDL et UDDI. Les services Web se basent sur les concepts suivants :

2.4.1 Le langage XML

XML est un langage informatique de balisage recommandé par le W3C, permettant de définir et de stocker des données de manière à pouvoir les partager et créer ses propres balises selon le besoin [9].

2.4.2 Le protocole SOAP

SOAP est un protocole standard de communication entre application permettant l'invocation de méthodes définies dans des services distants. Il utilise des technologies XML pour définir un cadre de messagerie extensible avec les protocoles HTTP, SMTP et POP sous forme de texte structuré [10].

2.4.3 Le langage WSDL

WSDL est un langage standard de description des services Web, écrit en XML, qui fournit une description précise des détails liés au service Web, tels que les protocoles utilisés, les ports, les opérations disponibles, les formats de messages d'entrée et de sortie, ainsi que les exceptions possibles qui peuvent être levées [11].

2.4.4 L'annuaire UDDI

UDDI est une norme de l'OASIS qui facilite la publication et la recherche d'informations relatives aux services Web. Il s'agit d'un standard de description et de découverte des services Web [12]

2.5 Avantages et inconvénients des services Web

Comme toutes les technologies, les services Web possèdent des points forts mais aussi certains points faibles.

2.5.1 Avantages des services Web

- Les services Web utilisent des standards et protocoles ouverts ce qui permet d'interagir de manière transparente avec d'autres systèmes, .
- Les services Web sont indépendants des plateformes et des langages grâce à l'utilisation de standards ouverts, tels que XML.
- La plupart des services Web utilisent le protocole HTTP pour échanger des messages entre les clients et les serveurs.
- Ils autorisent un couplage faible entre le client et le serveur.
- Les services Web offrent des mécanismes de découverte facilitant la recherche et l'utilisation de services disponibles [13].

2.5.2 Inconvénients des services Web

- Les services Web ont de faibles performances par rapport à d'autres approches de l'informatique répartie telles que le RMI, CORBA, ou DCOM.
- L'application qui utilise un service Web doit être connectée à Internet.
- Si un service Web est supprimé, cela peut entraîner des erreurs dans l'application qui l'utilise.
- Les services Web utilisant le protocole HTTP peuvent contourner les mesures de sécurité mises en place via des pare-feu [13].

2.6 Conclusion

Les services Web sont actuellement une technologie révolutionnaire qui mérite une attention particulière. Ils fournissent un cadre permettant de décrire et exécuter des applications à travers le réseau internet indépendamment de tout langage de programmation et de toute plate-forme d'exécution. Avec ces technologies telles que SOAP, XML, WSDL et UDDI qui permettent à des applications distantes de dialoguer entre eux et d'échanger des informations. Dans ce qui suit, nous passons à la conception et la réalisation de notre application.

Préparation de projet

3.1 Introduction

Dans ce chapitre, nous allons tout d'abord présenter le cycle de développement de la méthode SCRUM, par la suite nous allons définir les principaux acteurs impliqués dans notre application, ainsi que les besoins fonctionnels et non fonctionnels de cette dernière.

3.2 Spring 0

La première étape selon la méthode SCRUM est la phase de planification connue sous le nom de Sprint 0. Cette phase est considérée comme cruciale dans le cycle de développement SCRUM. Dans le Sprint 0, nous allons réaliser les actions suivantes :

- Constitution de l'équipe SCRUM (Product Owner, SCRUM Master, Équipe de développement).
- Écriture des users stories.
- Écriture et Priorisation du Product Backlog.
- Estimation et planification des Sprints et éventuellement des Releases.
- Mise en place de tout élément nécessaire au commencement du Sprint 1.

Role SCRUM	Role SCRUM
Product Owner	Mr KESSASSI
SCRUM master	Mme S.AIT HACENE epse OUHADDA
Team	BOUMEGUER Walid et BENDJELILI Manal

TABLE 3.1 – Les rôles de SCRUM

3.3 Spécification des besoins

Dans cette section, nous nous concentrons sur les exigences des utilisateurs que nous traduisant sous forme de spécifications fonctionnelles et non fonctionnelles. L'objectif est de créer une application

de haute qualité qui répondra aux besoins du client, afin de finaliser notre backlog de produit.

3.3.1 Identification des acteurs du système

Un acteur est un composant qui interagit avec le système et n'est pas nécessairement un être humain. Il est considéré comme une entité qui participe au fonctionnement du système [14].

Les acteurs principaux de notre application sont :

- **Organisateur** : cette personne est chargée d'organiser un événement de A à Z pour un client, qu'il s'agisse d'un particulier ou professionnel.
- **Client** : les clients sont représentés par le grand public qui consulte notre application pour en savoir plus sur les différents services et voir les propositions des organisateurs.
- **Directeur** : est un acteur actif, qui vérifie toutes les actions effectuées dans l'application. C'est lui qui va gérer le contenu du futur système.

a. Les besoins fonctionnels

Un besoin fonctionnel est une exigence directe qui exprime une réponse à une action, une demande ou une information du système. Les besoins fonctionnels de notre système se décrivent comme suit :

Coté Directeur :

Les actions effectuées par le directeur sont :

- Gestion des comptes : la possibilité de supprimer ou rechercher ou valider un compte.
- Gestion de son profil : modifier le profil ou l'afficher.
- Gestion des événements : l'ajout, la suppression, la recherche, la modification ou consulter la liste des événements
- Gestion des services : l'ajout, la suppression, la recherche, la modification ou consulter la liste des services

Coté organisateur :

Les actions effectuées par l'organisateur sont :

- Gestion des propositions : l'ajout, la suppression, la recherche, la modification ou consulter la liste des propositions.
- Gestion de son profil : Modifier le profil ou l'afficher.

Coté client :

Notre application permet aux clients de :

- La consultation des propositions.
- La gestion de son profile :modifier le profil ou l'afficher.
- La possibilité de poster un commentaire.

b. Les besoins non fonctionnels :

Les besoins non fonctionnels sont des exigences qui décrivent les caractéristiques et les contraintes du système, plutôt que ses fonctionnalités spécifiques. Ils sont souvent liés à des aspects tels que :

- **L'ergonomie** : C'est un besoin non fonctionnel de l'application, qui implique que les interfaces proposées doivent être bien structurées, faciles à utiliser et à manipuler. Cela inclut des choix esthétiques tels que les couleurs et les styles utilisés.
- **La disponibilité** : le système doit être disponible ou accessible en temps réel par l'administrateur et les utilisateurs pour que chacun puisse réaliser la tâche souhaitée ou demandée.
- **La maintenance** : les différentes pages Web doivent être faciles à maintenir, le code doit être lisible et bien structuré.
- **La fiabilité** : définissent les exigences en matière de stabilité du système, de prévention des erreurs, de récupération ou la réctification après un échec, de sauvegarde des données et de gestion des erreurs.
- **La sécurité** : vise à garantir la protection contre les risques et les menaces, afin de préserver l'intégrité, la confidentialité et la disponibilité des biens.

c. Diagramme de contexte dynamique

Le diagramme de contexte fournit une représentation simplifiée du système et de son environnement, mettant en évidence les interactions et les échanges d'informations essentiels pour comprendre le fonctionnement global du système [15].

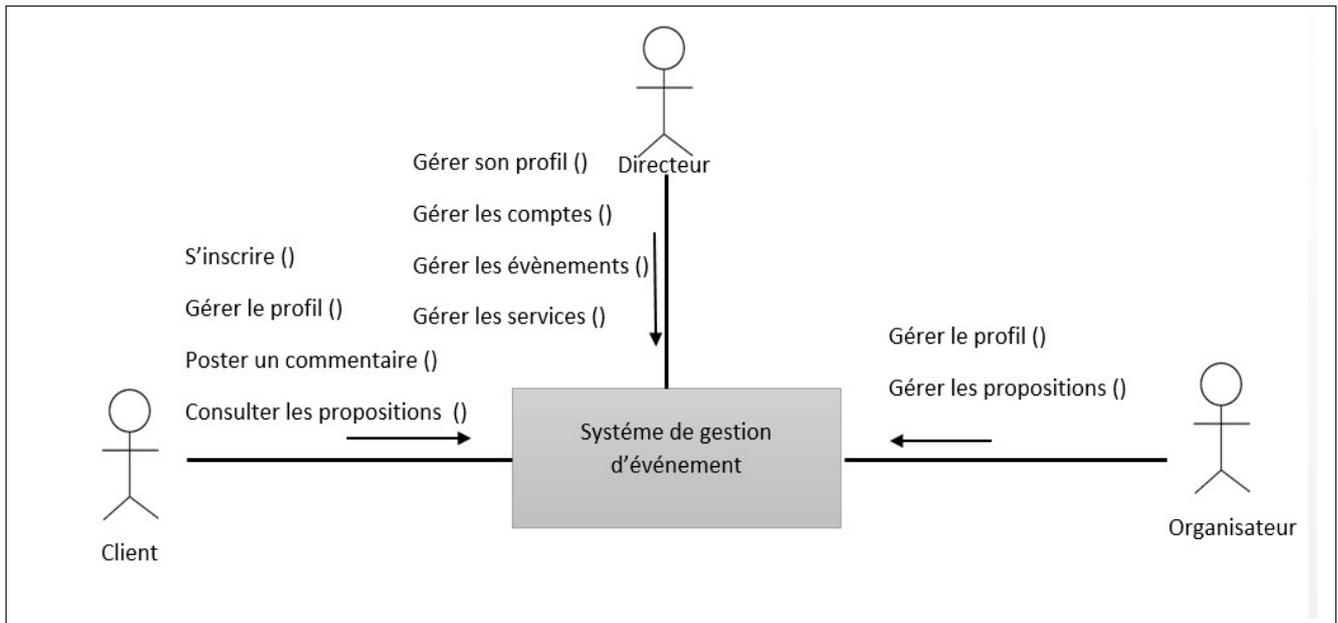


FIGURE 3.1 – Diagramme de contexte

3.3.2 Diagramme de cas d'utilisation

Le diagramme de cas d'utilisation est un outil de modélisation UML qui offre une représentation visuelle des besoins du système en décrivant les interactions entre les utilisateurs et le système. Il permet ainsi de définir la manière dont les utilisateurs interagissent avec le système [38].

Diagramme de cas d'utilisation global

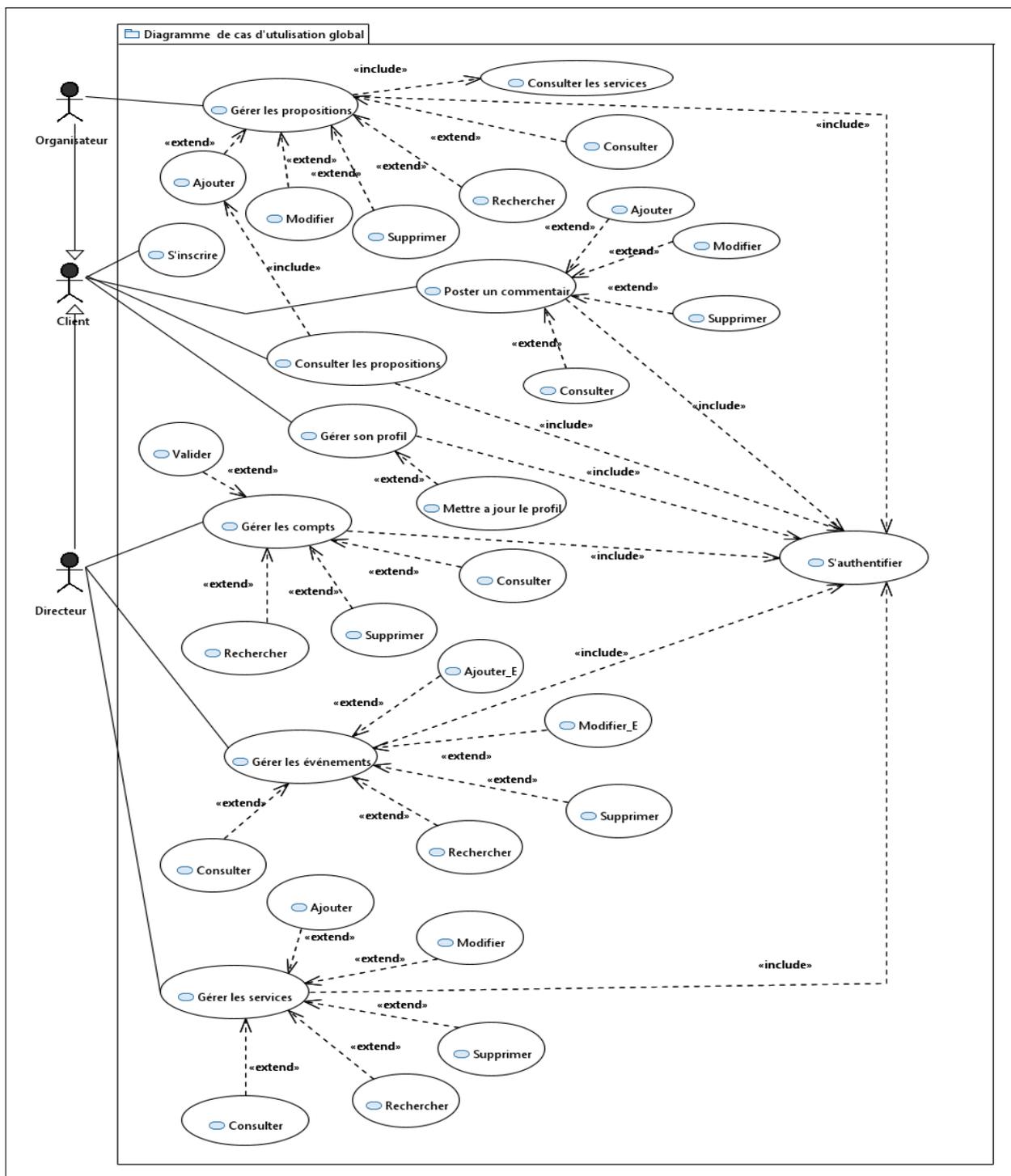


FIGURE 3.2 – Diagramme de cas d'utilisation global

3.4 Product Backlog

La seconde étape c'est la définition du product backlog qu'est composé d'un ensemble de user story et chaque user story est caractérisée par une priorité dénie par le « Product Owner ».

Le product backlog de notre futur système est présenté dans le tableau suivant :

Backlog					
N°	ID	User story	Cas d'utilisation	Priorité	Sprint
01	1.1	En tant que client, je peux créer mon propre compte.	Inscription	01	01
02		En tant qu'utilisateur ,je peux :	Authentification	01	01
	2.1	Annuler où valider ce que j'ai saisie.			
	2.2	Saisir le nom utilisateur et mon mot de passe.			
	2.3	Récupérer mon mot de passe.			
03		En tant qu'utilisateur,je peux :	Gérer le profil	02	02
	3.1	Modifier mon profil.			
	3.2	Afficher mon profil.			
04		En tant que directeur, je peux :	Gérer les comptes	03	02
	4.1	Valider un compte.			
	4.2	Supprimer un compte.			
	4.3	Rechercher un compte.			
	4.4	Consulter la liste des comptes.			
05		En tant que directeur, je peux :	Gérer les événements	04	03
	5.1	Ajouter un événement.			
	5.2	Supprimer un événement.			
	5.3	Modifier un événement.			
	5.4	Rechercher un événement.			
	5.6	Consulter la liste des événements.			
06		En tant que directeur, je peux :	Gérer les services	05	03

	6.1	Ajouter un service.			
	6.2	Supprimer un service.			
	6.3	Modifier un service.			
	6.4	Rechercher un service.			
	6.5	Consulter la liste des services.			
07		En tant qu'organisateur, je peux :	Gérer les propositions	06	03
	7.1	Ajouter une proposition.			
	7.2	Supprimer une ou plusieurs propositions.			
	7.3	Modifier une ou plusieurs propositions.			
	7.4	Rechercher une ou plusieurs propositions.			
	7.5	Consulter la liste des propositions.			
08	8.1	En tant que client, je peux consulter les propositions.	Consulter une proposition	07	04
09	9.1	En tant que client, je peux poster un commentaire .	Poster un commentaire	08	04

TABLE 3.2: Product Backlog

3.4.1 Planification des releases

Après avoir terminé la deuxième étape qui consiste à établir le Product backlog, l'étape suivante dans le processus SCRUM est la planification des releases. Dans le cadre de notre projet, la planification des quatre sprints est représentée par le schéma suivant :

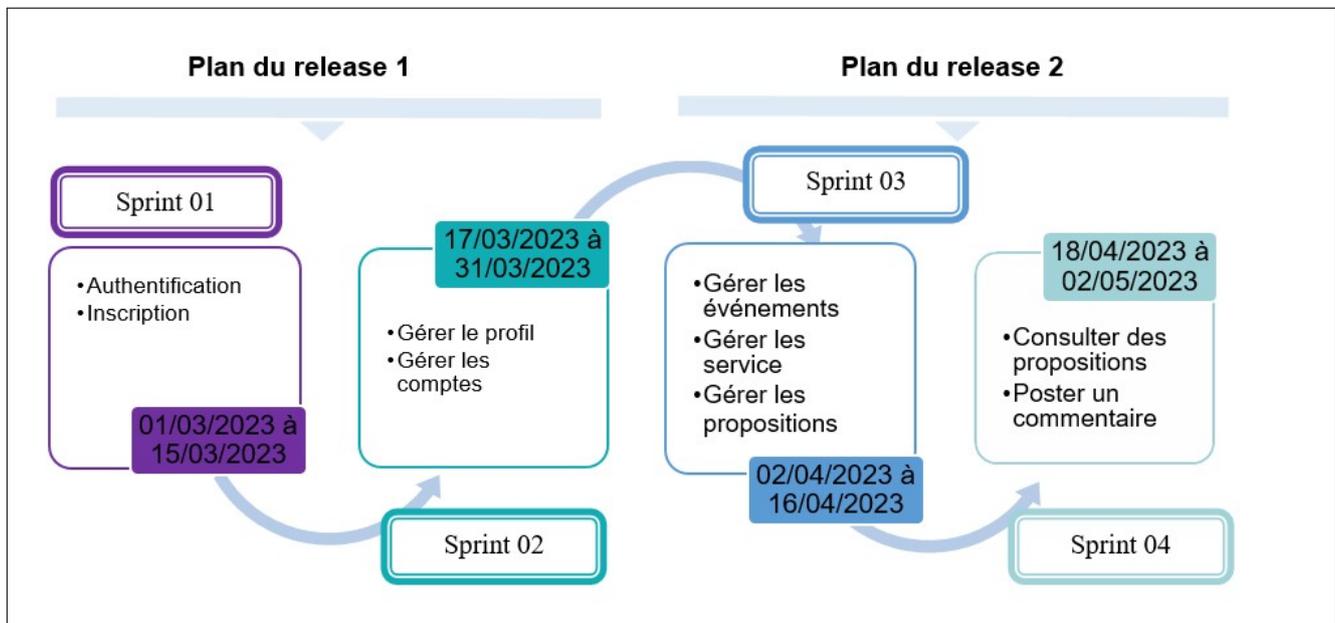


FIGURE 3.3 – Schéma de Release

3.5 Conclusion

Dans ce chapitre, nous avons tout d'abord présenté le pilotage de la méthode SCRUM ,puis nous avons défini les acteurs principaux de notre application ainsi que les besoins fonctionnels et non fonctionnels,et nous avons spécifié les différents cas d'utilisation à travers des diagrammes. Dans la suite de ce rapport, nous examinons la problématique et en proposons une solution et en détaillons la conception du projet.

Conception et réalisation

4.1 Introduction

Dans ce chapitre, nous allons fournir une description détaillée de notre utilisation de la méthode SCRUM, en mettant en évidence les différentes étapes de chaque sprint, telles que l'analyse, la conception et la réalisation.

4.2 Développement du Sprint 01 « Authentification, inscription » :

4.2.1 SCRUM Board (task board)

Le SCRUM Board est un outil permettant de visualiser les éléments du Sprint Backlog pour aider les équipes. Cet outil est mis à jour par l'équipe et affiche tous les éléments qui doivent être complétés pour le sprint en cours. Le SCRUM Board du sprint 01 est le suivant :

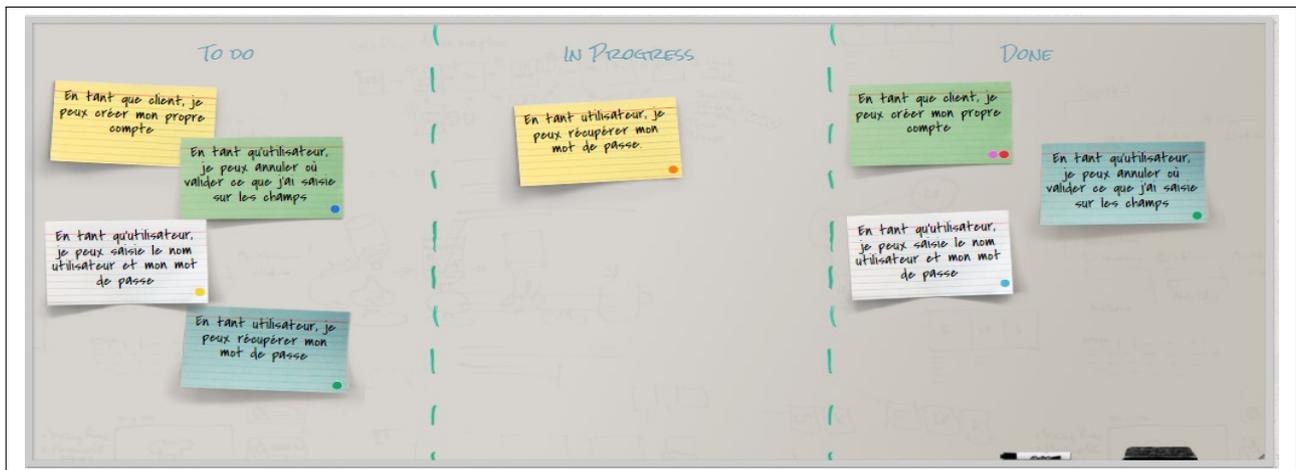


FIGURE 4.1 – SCRUM board du sprint 01

4.2.2 User story du sprint 01

ID	User Story
01	En tant que client, je peux créer mon propre compte.
2.1	En tant qu'utilisateur, je peux annuler ou valider ce que j'ai saisi sur les champs.
2.2	En tant qu'utilisateur, je peux saisir mon nom d'utilisateur et mon mot de passe.
2.3	En tant qu'utilisateur, je peux récupérer mon mot de passe.

TABLE 4.1 – User story du sprint 01

4.2.3 Backlog du sprint 01

Item	User Story	Priorité
Inscription	Inscrire	01
Authentification	Authentifier	01

TABLE 4.2: Backlog du sprint 01

4.2.4 Analyse

Diagramme de cas d'utilisation du sprint 01

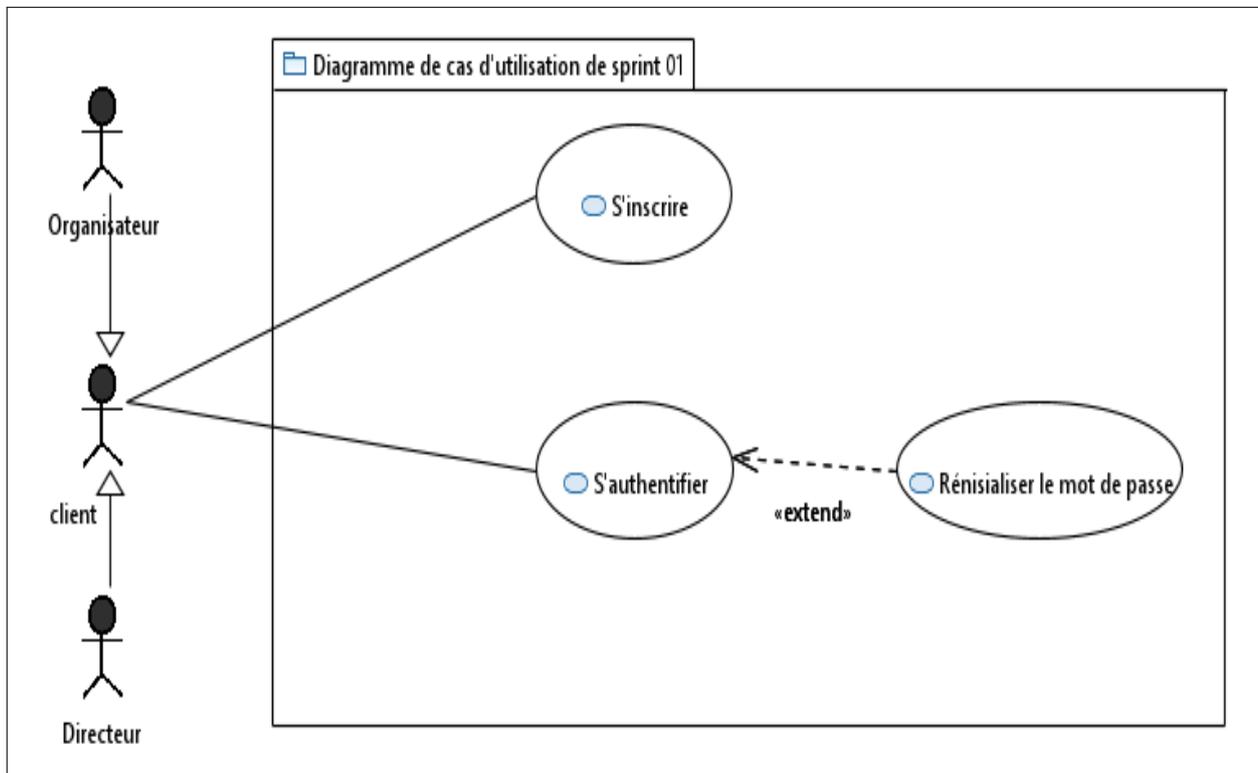


FIGURE 4.2 – Diagramme de cas d'utilisation du sprint 01

Description textuelle des cas d'utilisation

Il est important d'avoir une description textuelle du cas d'utilisation afin de comprendre la séquence d'actions qui y sont exécutées.

Description textuelle « Authentification »

Cas d'utilisation	S'authentifier
Acteur	Directeur, Client, Organisateur
Objectif	Accès aux fonctionnalités réservées à l'utilisateur
Scénario nominal	<ol style="list-style-type: none"> 1. L'utilisateur accède à l'application. 2. Le système affiche l'interface d'authentification. 3. L'utilisateur saisit son nom d'utilisateur et son mot de passe. 4. Le système vérifie l'existence du compte. 5. Le système affiche l'espace approprié pour chaque utilisateur.
Scénario alternatif	<p>Cas 01 : Login et mot de passe incorrects</p> <p>Un retour vers la page d'authentification sera effectué avec un message d'erreur.</p>

Description textuelle « Inscription »

Cas d'utilisation	Inscription
Acteur	Client
Objectif	Pouvoir s'inscrire
Scénario nominal	<ol style="list-style-type: none"> 1. Le client remplit le formulaire d'inscription. 2. Le client valide son inscription. 3. Le système enregistre la demande d'inscription et affiche un message qui indique que l'inscription a été faite avec succès. 4. Le système met à jour les informations dans la base de données.
Scénario alternatif	<p>Cas 01 : Champs obligatoires non valides ou vides.</p> <p>Le système affiche un message d'erreur.</p> <p>Cas 02 : Client existant.</p> <p>Le système affiche un message d'erreur.</p>

TABLE 4.4: Description textuelle « Inscription »

4.2.5 Conception

Le diagramme de séquence est une catégorie de diagrammes comportementaux (dynamiques) dans UML qui illustre les interactions entre les éléments du système ainsi que les acteurs, en fonction du temps. [01]

Diagramme de séquence « Authentification »

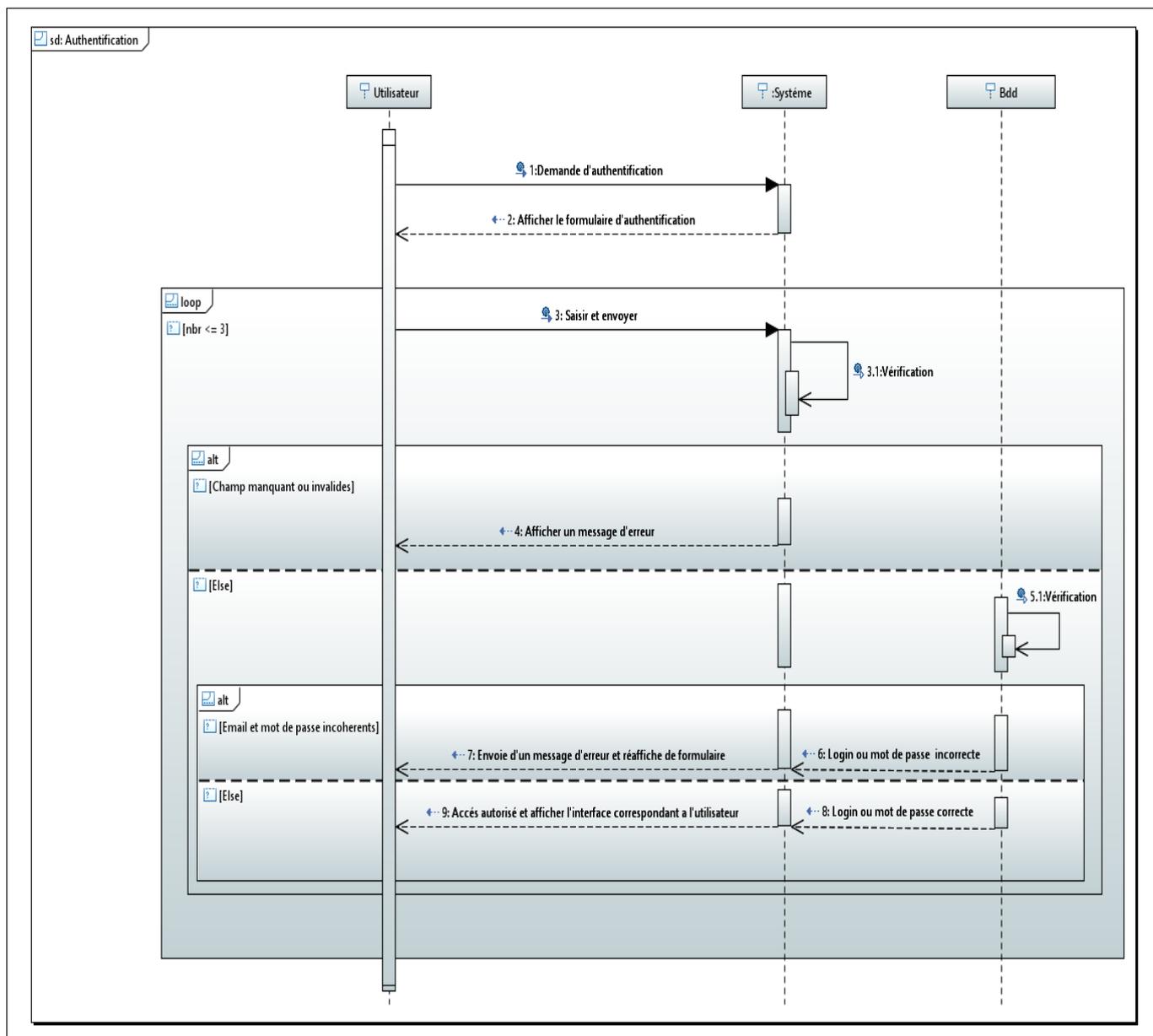


FIGURE 4.3 – Diagramme de séquence « Authentification »

Diagramme de séquence « Inscription »

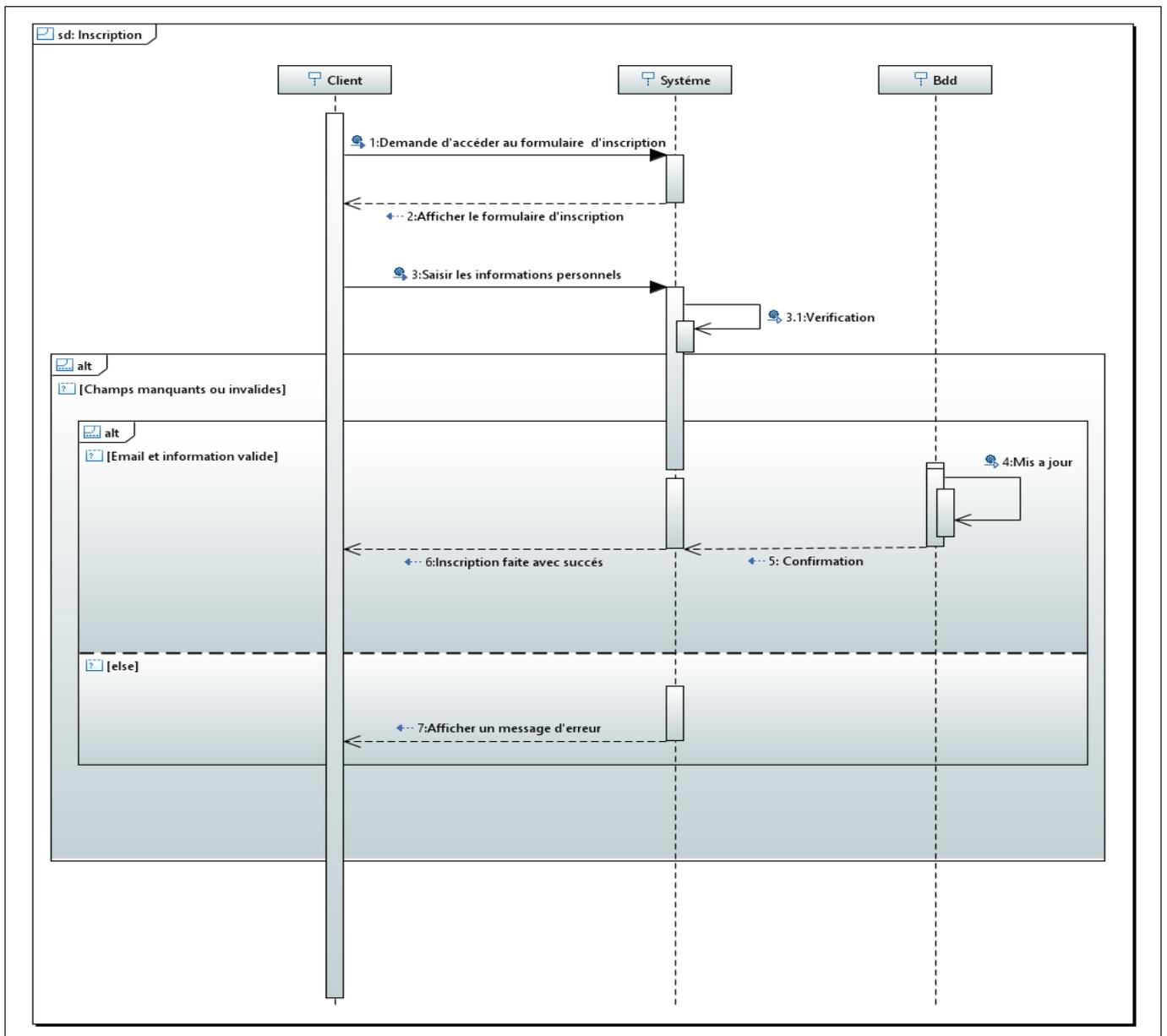


FIGURE 4.4 – Diagramme de séquence « Inscription »

4.2.6 Réalisation

Interface « Authentification »

Cette interface d'authentification permet à divers utilisateurs d'accéder au contenu de l'application, en fonction de leur rôle

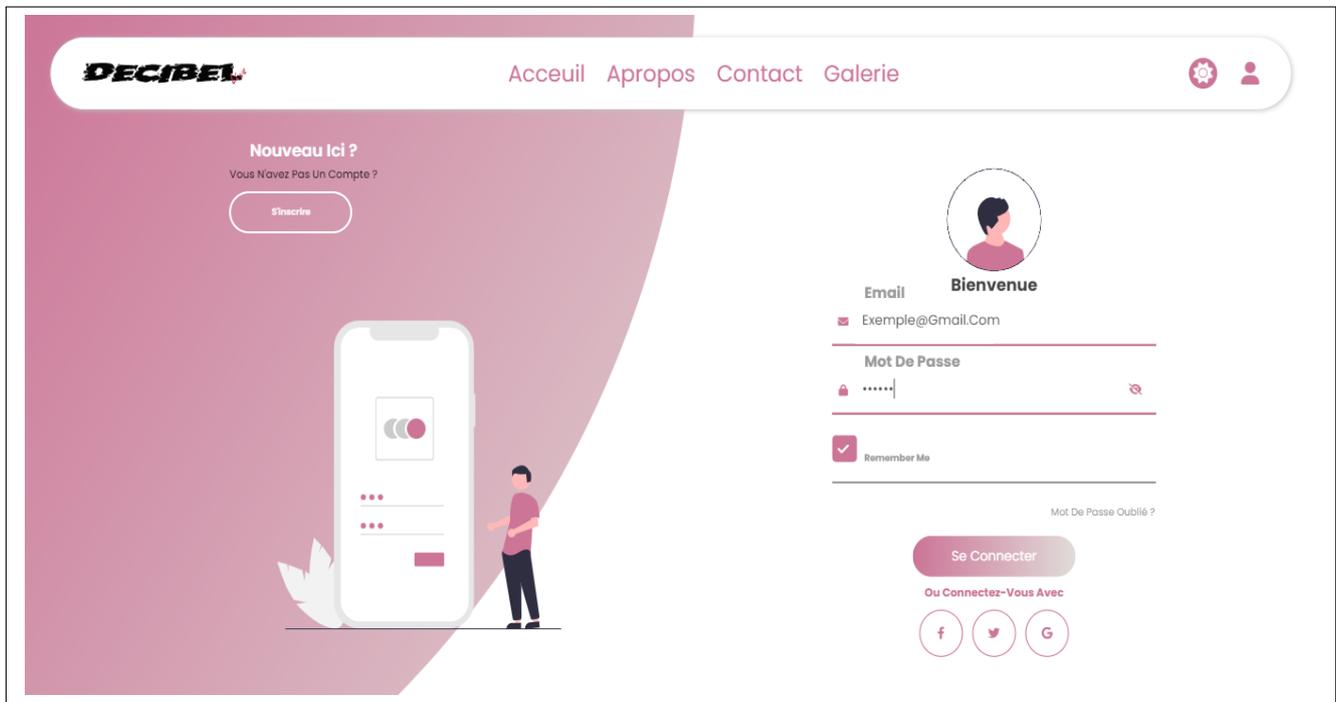


FIGURE 4.5 – Interface « Authentification »

Pour récupérer le mot de passe il suffit d'abord de saisir l'email et le numéro de téléphone (interface mot de passe oublié) après la validation des champs saisie l'utilisateur peut réinitialiser son mot de passe (interface réinitialiser).

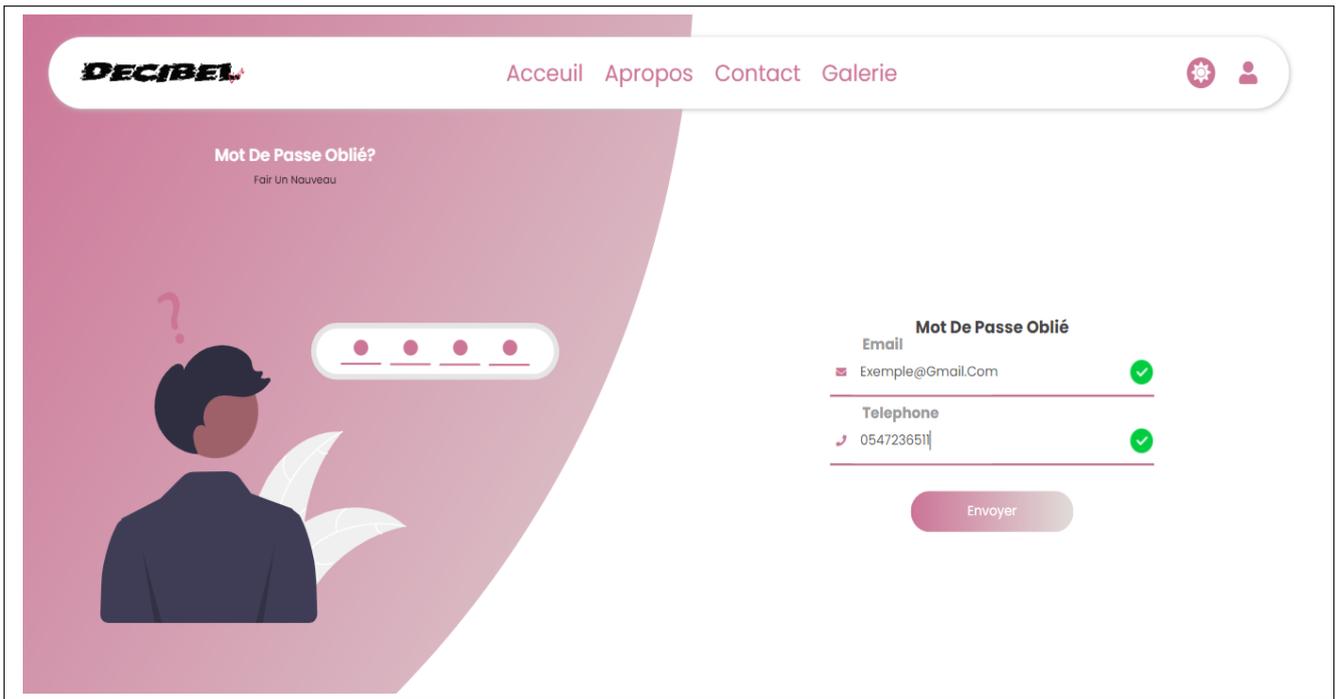


FIGURE 4.6 – Interface « Mot de passe oublié oublié »

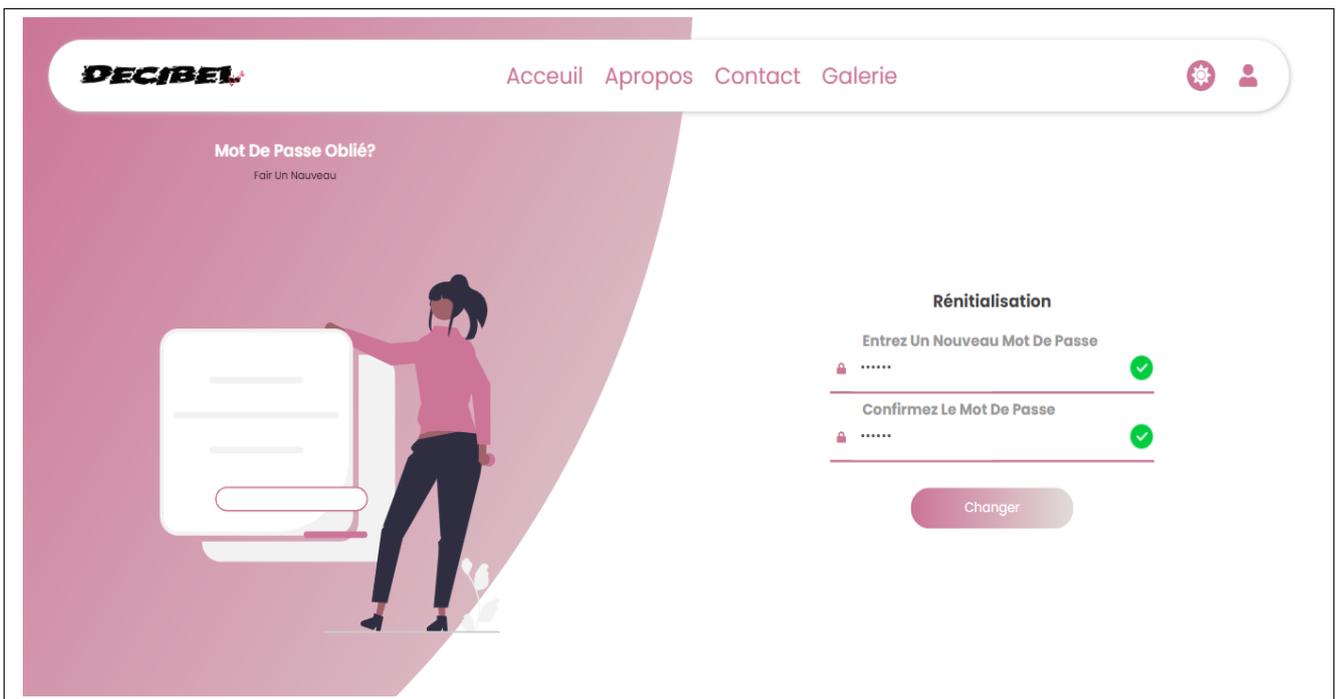


FIGURE 4.7 – Interface « Réinitialiser »

Interface « Inscription »

Cette interface permet au client de créer un compte afin d'accéder au contenu de l'application ,après la validation des champs

The screenshot shows the registration interface for the DECIBEL application. The top navigation bar includes the logo 'DECIBEL' and links for 'Accueil', 'Apropos', 'Contact', and 'Galerie'. On the right side of the navigation bar, there are icons for settings and a user profile.

The registration form on the left is titled 'Inscription' and contains the following fields, each with a green checkmark indicating successful validation:

- Nom**: Input field with value 'Nom'.
- Email**: Input field with value 'Exemple@Gmail.Com'.
- Téléphone**: Input field with value '0547236511'.
- Adresse**: Input field with value 'Bejaia'.
- Mot De Passe**: Password field with masked characters '.....'.
- Confirmer Le Mot De Passe**: Confirmation password field with masked characters '.....'.

Below the form is a reCAPTCHA widget with the text 'Je ne suis pas un robot' and a 'S'inscrire' button.

The right section of the interface is titled 'Un De Nous ?' and contains the text 'Vous Avez Déjà Un Compte ?' and a 'Se Connecter' button. Below this, there are several empty input fields and an illustration of a person standing.

FIGURE 4.8 – Interface « Inscription »

Après la création d'un compte un mail sera envoyé pour activer le compte comme indiqué dans les figure qui suit.

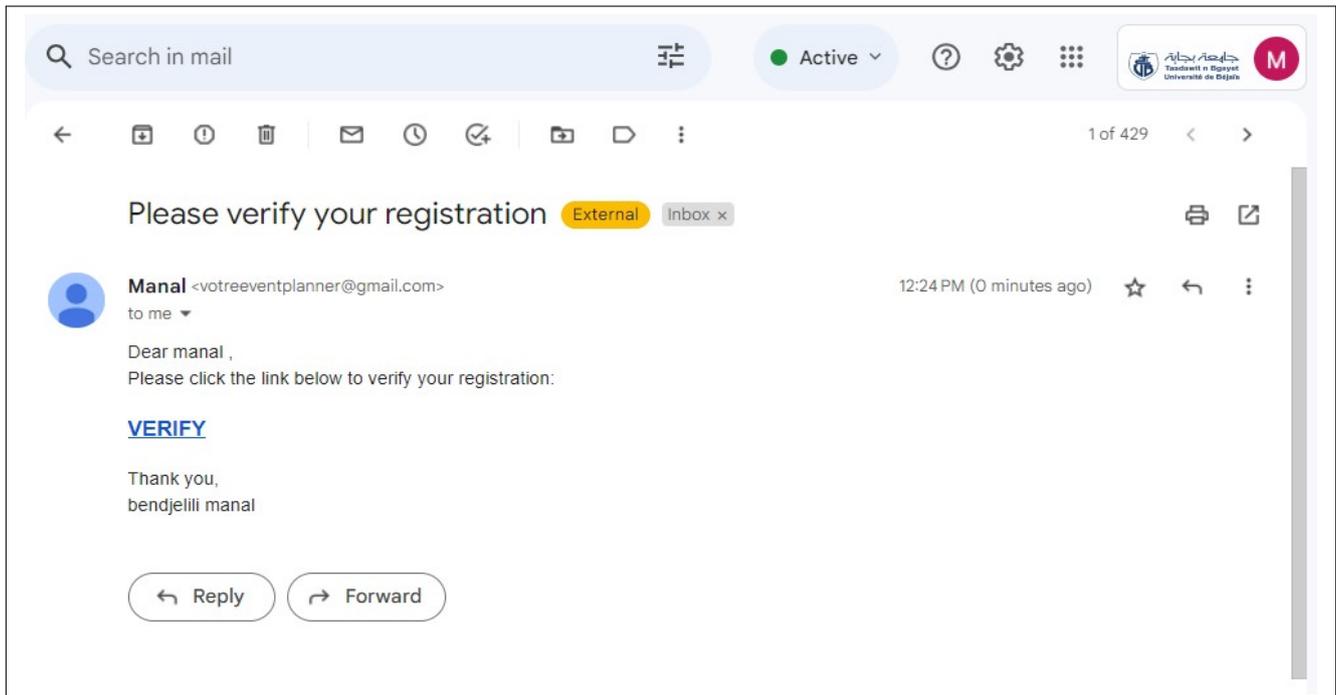


FIGURE 4.9 – Interface « Reçois d'un mail d'activation du compte »



FIGURE 4.10 – Interface « Le mail d'activation »

4.3 Développement du Sprint 02 « Gérer le profil, Gérer les comptes » :

4.3.1 SCRUM Board (task board)

Le SCRUM Board du sprint 02 est le suivant :

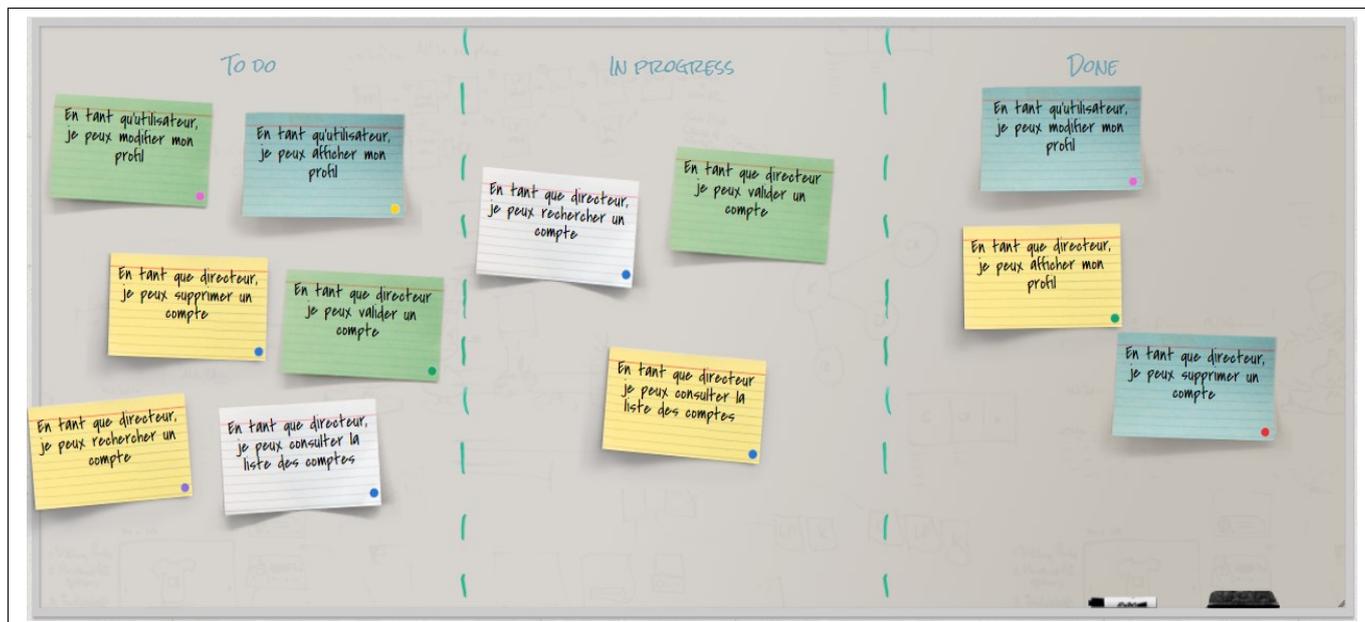


FIGURE 4.11 – SCRUM board du sprint 02

4.3.2 User story du sprint 02

ID	User Story
3.1	En tant qu'utilisateur, je peux modifier mon profil.
3.2	En tant qu'utilisateur, je peux afficher mon profil.
4.1	En tant que directeur, je peux valider un compte.
4.2	En tant que directeur, je peux supprimer un compte.
4.3	En tant que directeur, je peux rechercher un compte.
4.4	En tant que directeur, je peux consulter la liste des comptes.

TABLE 4.5 – User story du sprint 02

4.3.3 Backlog du sprint 02

Item	User Story	Priorité
Gérer le profil	Modifier le profil	02
	Afficher le profil	
Gérer les comptes	Valider un compte	03
	Supprimer un compte	
	Rechercher un compte	
	Consulter la liste des comptes	

TABLE 4.6: Backlog du sprint 02

4.3.4 Analyse

Diagramme de cas d'utilisation du sprint 02

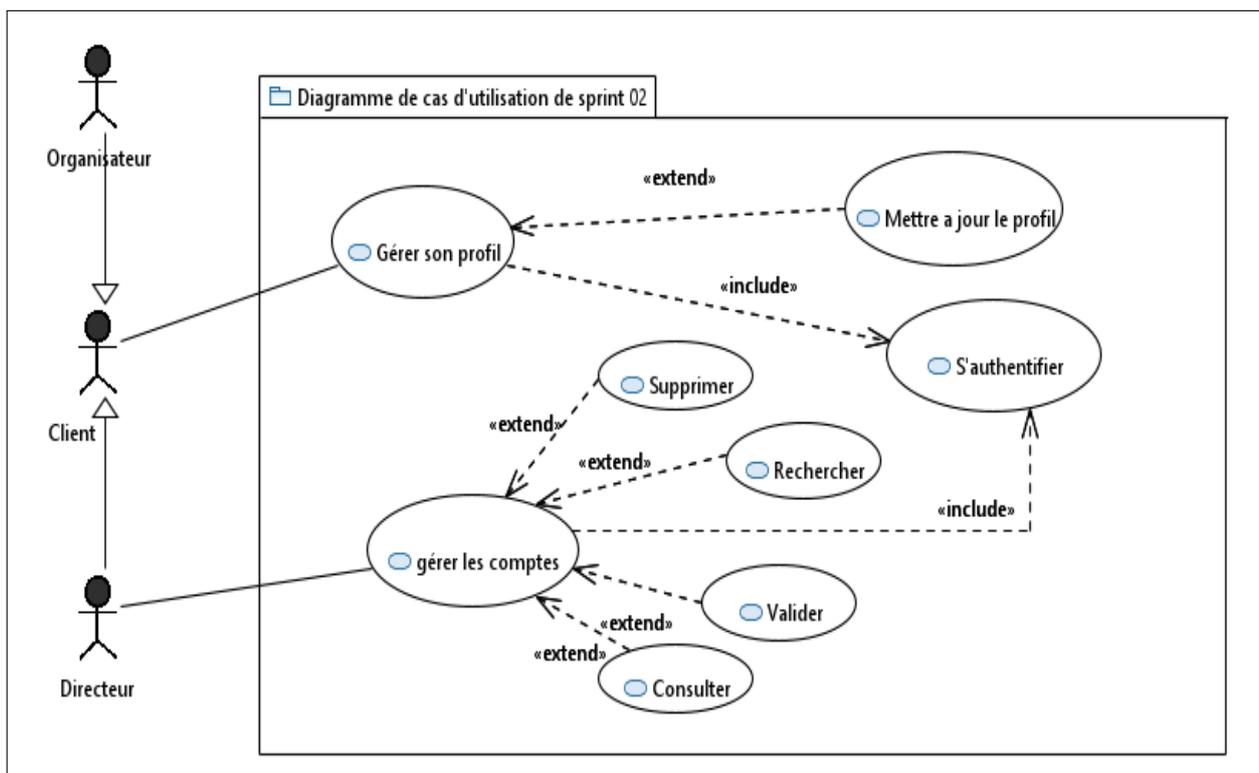


FIGURE 4.12 – Diagramme de cas d'utilisation du sprint 02

Description textuelle des cas d'utilisation

Description textuelle du cas « Gérer profil »

Cas d'utilisation	Gérer profil
Acteur	Directeur,organisateur,client
Objectif	Mise à jour du profil
Scénario nominal	<ol style="list-style-type: none"> 1. L'utilisateur accède à son profil. 2. Le système affiche le formulaire de ses informations. 3. L'utilisateur peut se modifié les champs voulus. 4. Le système met à jour les informations dans la base de données.
Exception	<p>Cas 01 : Champs obligatoires non valide / ou vide Le système affiche un message d'erreur.</p>

TABLE 4.7: Description textuelle du cas « Gérer profil »

Description textuelle du cas « Gérer les comptes »

Cas d'utilisation	Gérer les comptes
Acteur	Directeur
Objectif	La gestion des comptes (valider, supprimer ou rechercher un compte)
Scénario nominal	<ol style="list-style-type: none"> 1. Le directeur accède à la fonctionnalité de gestion des comptes. 2. Le système affiche la liste des comptes d'utilisateurs existants. 3.Le directeur peut effectuer les actions suivantes : <ul style="list-style-type: none"> . Valider un compte. . Supprimer un compte . Rechercher un compte 4. Le système met à jour les informations des comptes d'utilisateurs selon les actions effectuées par le directeur général. 5. Le directeur peut visualiser la liste mise à jour des comptes d'utilisateurs.
Exceptions	<p>Cas 01 : Si le compte à valider n'est pas valide Le système affiche un message d'erreur et empêche la validation.</p> <p>Cas 02 : Si le compte à supprimer n'existe pas Le système affiche un message d'erreur.</p> <p>Cas 03 : Si la recherche ne renvoie aucun compte correspondant aux critères Le système affiche un message indiquant l'absence de résultats.</p>

TABLE 4.8: Description textuelle du cas « Gérer les comptes »

4.3.5 Conception

Diagramme de séquence « Gérer profil »

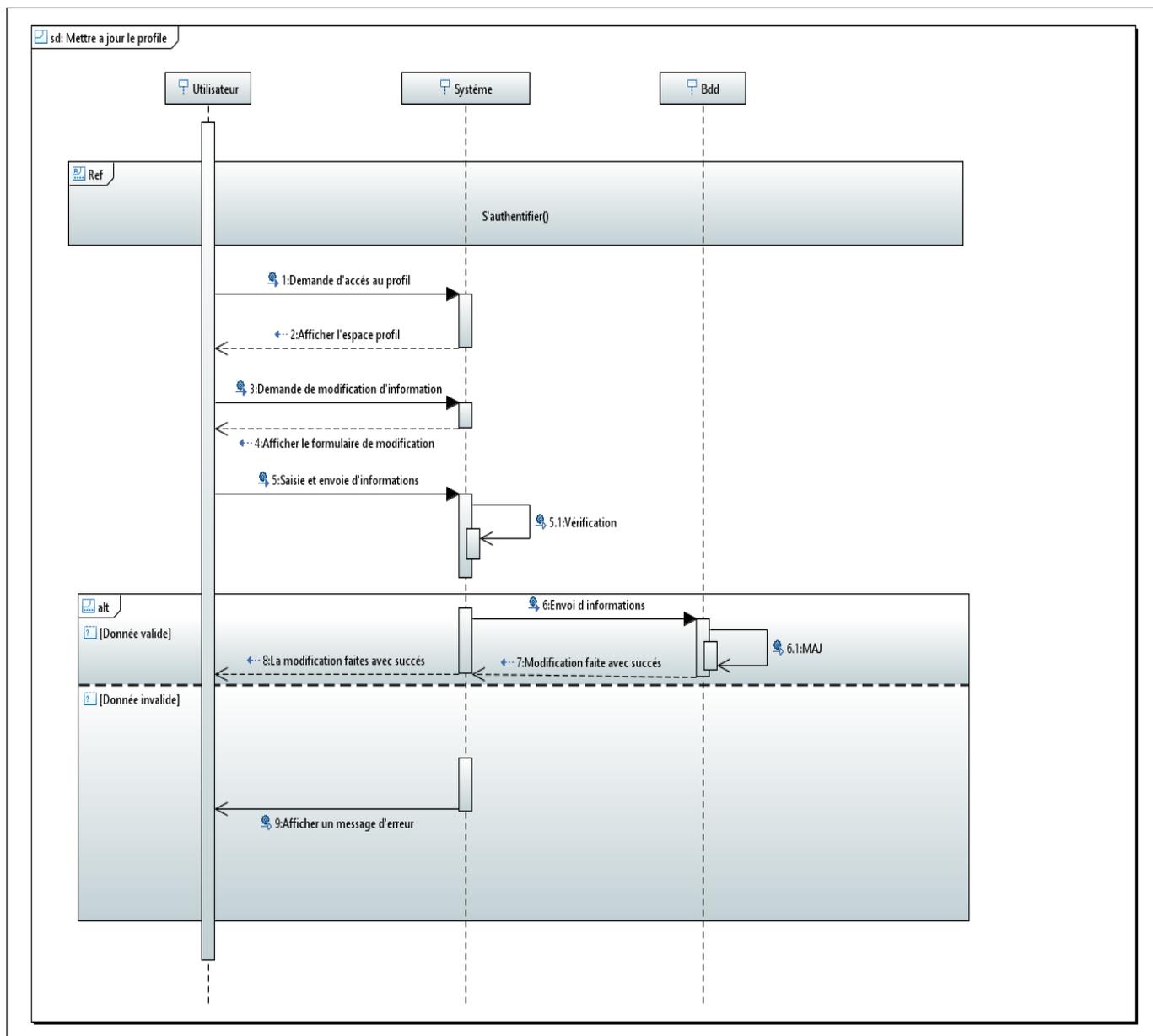


FIGURE 4.13 – Diagramme de séquence « Gérer profil »

Diagramme de séquence « Gérer les comptes »

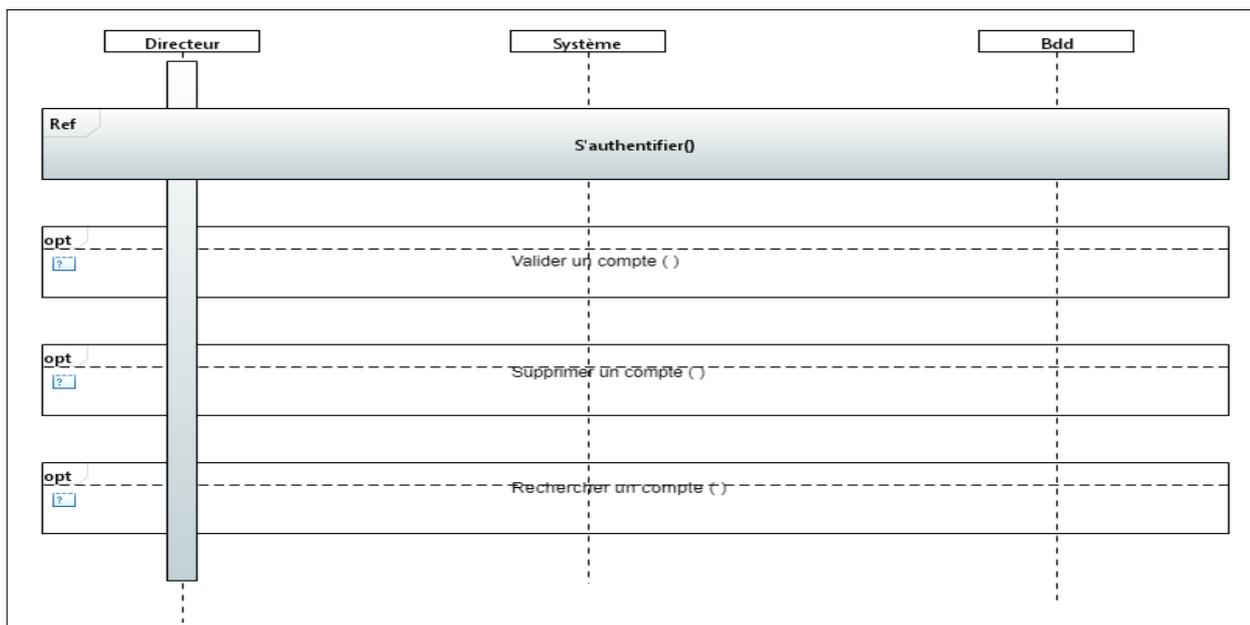


FIGURE 4.14 – Diagramme de séquence « Gérer les comptes »

4.3.6 Réalisation

Interface « Gérer profil »

Cette interface permet de gérer le profil, donnant ainsi la possibilité aux utilisateurs de modifier leurs informations.

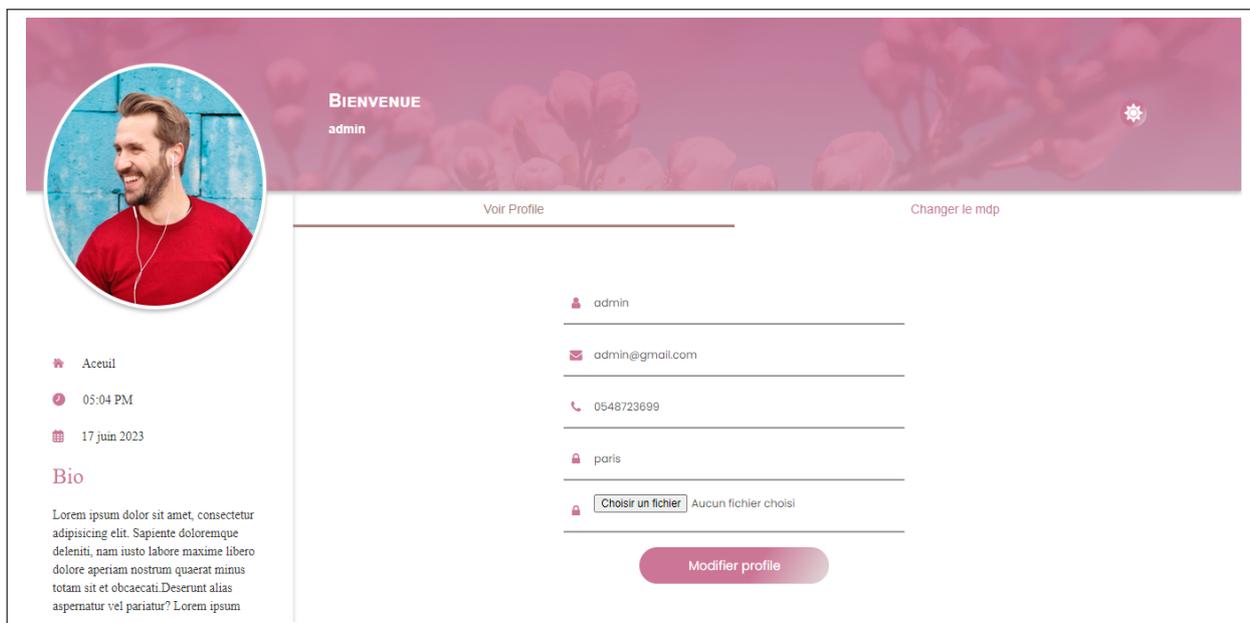


FIGURE 4.15 – Interface « Gérer profil »

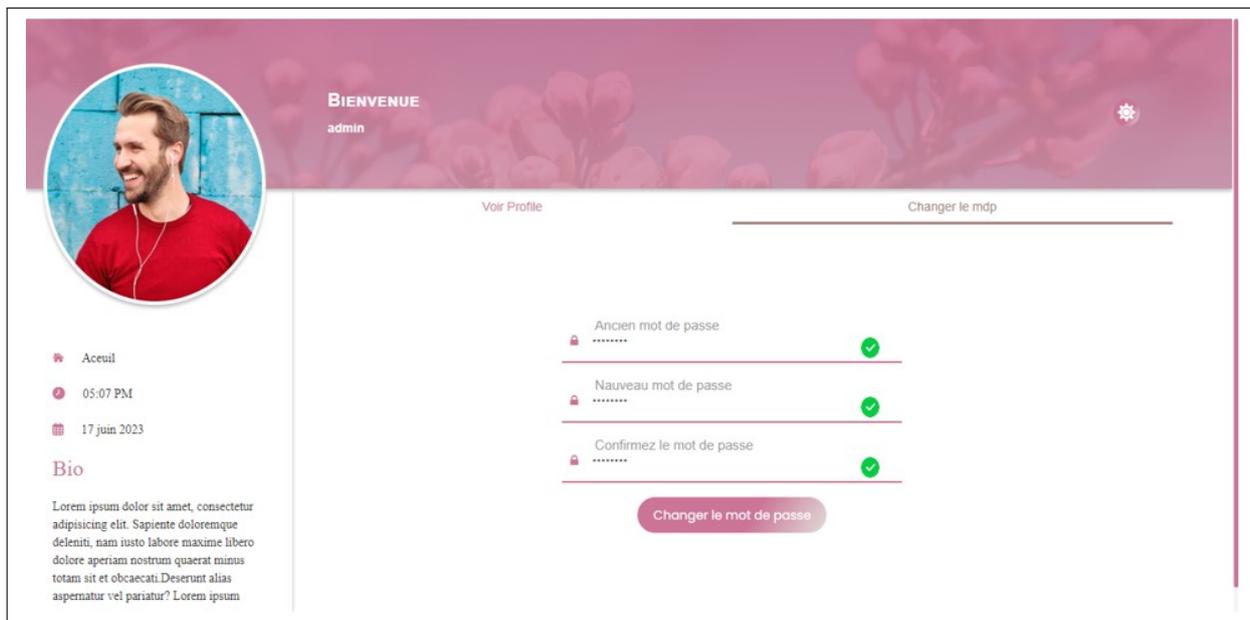


FIGURE 4.16 – Interface « Gérer profil(changer le mot de passe) »

Interface « Gérer les comptes »

Cette interface permet de gérer les comptes, offrant la possibilité de valider, supprimer ou rechercher un compte.

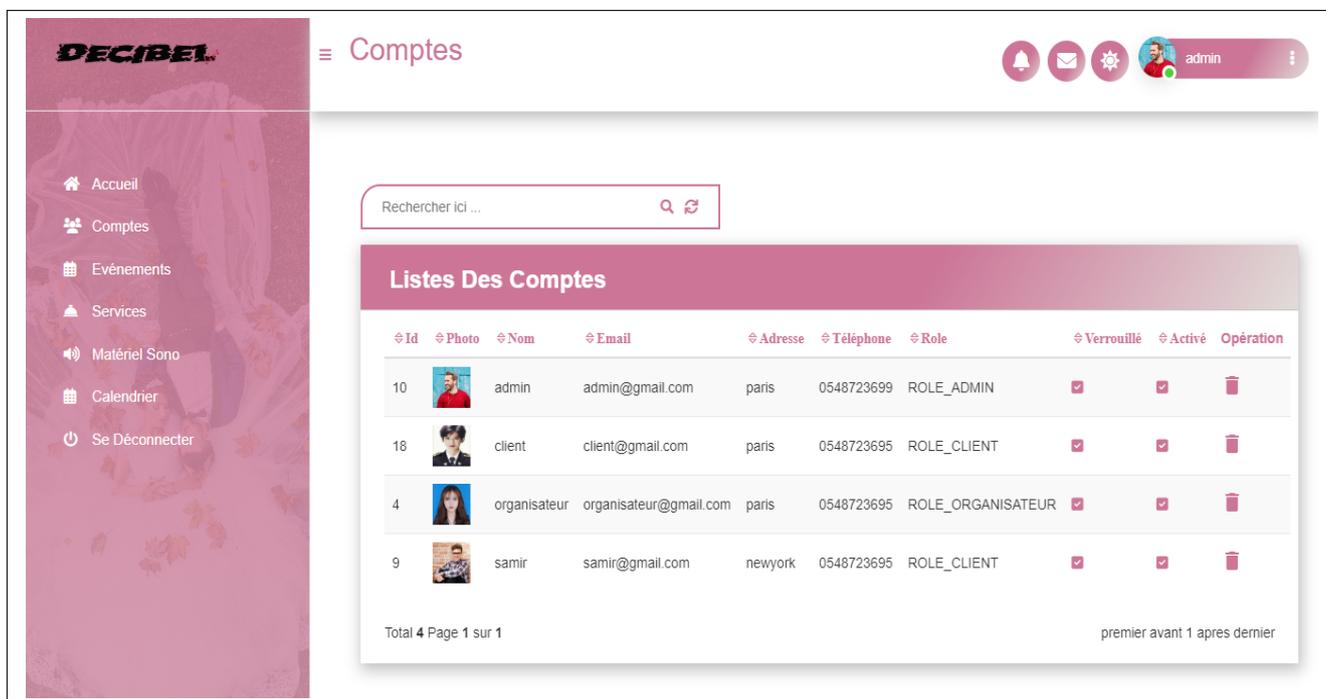


FIGURE 4.17 – Interface « Gérer les comptes »

4.4 Développement du Sprint 03 « Gérer les propositions, Gérer les événements, Gérer les services » :

4.4.1 SCRUM Board (task board)

Le SCRUM Board du sprint 03 est le suivant :

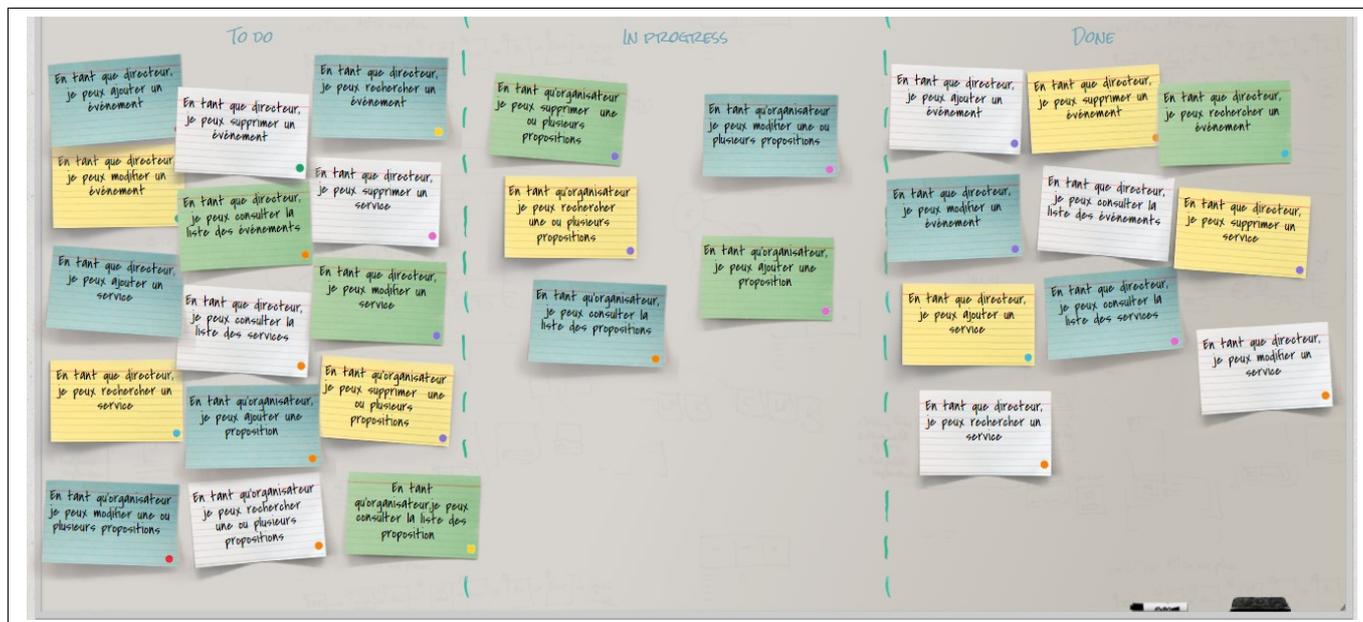


FIGURE 4.18 – SCRUM board du sprint 03

4.4.2 User story du sprint 03

ID	User Story
5.1	En tant que directeur, je peux ajouter un événement.
5.2	En tant que directeur, je peux supprimer un événement.
5.3	En tant que directeur, je peux modifier un événement.
5.4	En tant que directeur, je peux rechercher un événement.
5.5	En tant que directeur, je peux consulter la liste des événements.
6.1	En tant que directeur, je peux ajouter un service.
6.2	En tant que directeur, je peux supprimer un service.
6.3	En tant que directeur, je peux modifier un service.
6.4	En tant que directeur, je peux rechercher un service.
6.5	En tant que directeur, je peux consulter un service.
7.1	En tant qu'organisateur, je peux ajouter une proposition.

7.2	En tant qu'organisateur, je peux supprimer une ou plusieurs propositions.
7.3	En tant qu'organisateur, je peux modifier une ou plusieurs propositions.
7.4	En tant qu'organisateur, je peux rechercher une ou plusieurs propositions.
7.5	En tant qu'organisateur, je peux consulter la liste des propositions.
7.6	En tant qu'organisateur, je peux consulter les services.

TABLE 4.9: User story du sprint 03

4.4.3 Backlog du sprint 03

Item	User Story	Priorité
Gérer les événements	Ajouter un événement Supprimer un événement Modifier un événement Rechercher un événement Consulter la liste des événements	04
Gérer les services	Ajouter un service Supprimer un service Modifier un service Rechercher un service Consulter la liste des services	05
Gérer les propositions	Ajouter une proposition Supprimer une proposition Modifier une proposition Rechercher une proposition Consulter la liste des propositions	06

TABLE 4.10: Backlog du sprint 03

4.4.4 Analyse

Diagramme de cas d'utilisation du sprint 03

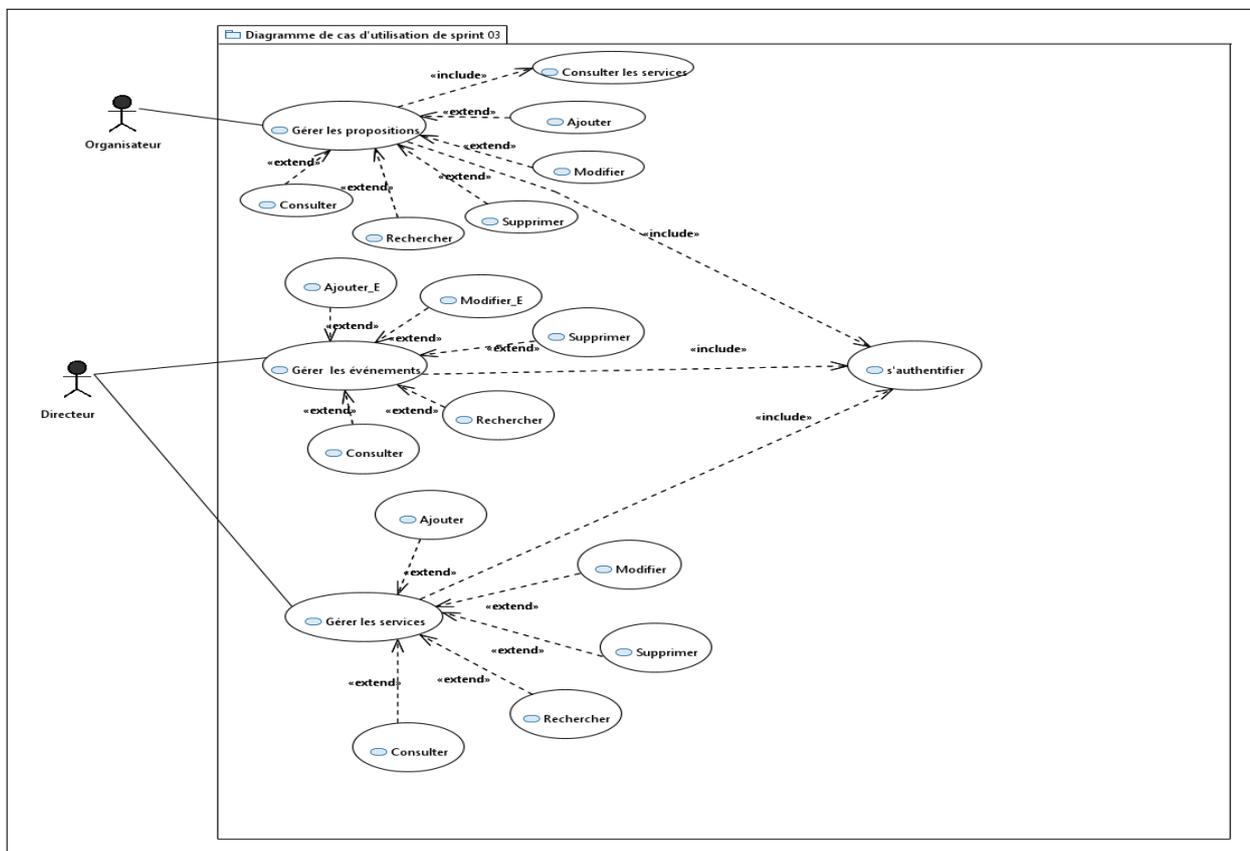


FIGURE 4.19 – Diagramme de cas d’utilisation du sprint 03

Description textuelle des cas d’utilisation

Description textuelle du class « Gérer les événements »

Cas d'utilisation	Gérer les événements
Acteur	Directeur
Précondition	Le directeur doit s'authentifier
Objectif	La possibilité d'ajouter, modifier, supprimer ou rechercher un événement
Scénario nominal	<p>L'ajout d'un événement</p> <ol style="list-style-type: none"> 1. Le directeur choisit d'ajouter un évènement 2. Le système affiche le formulaire d'ajout 3. Le directeur renseigne et valide le formulaire 4. Le système mit à jour les informations à la base de données 5. Le système actualise la liste des événements et l'affiche

	<p>La modification d'un événement</p> <ol style="list-style-type: none"> 1. Le directeur choisit de modifier un événement 2. Le système affiche le formulaire de modification 3. Le directeur effectue les modifications 4. Le directeur valide ou annule la modification 5. Le système met à jour les informations dans la base de données 6. Le système confirme la modification <p>La suppression d'un événement</p> <ol style="list-style-type: none"> 1. Le directeur choisit un événement à supprimer 2. Le système demande une confirmation 3. Le directeur confirme ou annule la suppression 4. Le système supprime l'événement de la base de données 5. Le système actualise la liste des événements et l'affiche <p>La recherche d'un événement</p> <ol style="list-style-type: none"> 1. Le directeur choisit de rechercher un événement 2. Le système affiche la zone de recherche 3. Le directeur remplit et valide la zone de recherche 4. Le système recherche l'événement dans la base de données 5. Le système affiche l'événement correspondante
Exception	<p>Cas 01 : événement non existante ou champs non conformes / formulaire vide</p> <p>Le système affiche un message d'erreur</p>

TABLE 4.11: Description textuelle du cas « Gérer les événements »

La description textuelle des cas d'utilisation « gérer les service, gérer les propositions » se déroule de la même façon que cas « gérer événement ».

4.4.5 Conception

Diagramme de séquence « Gérer les événements »

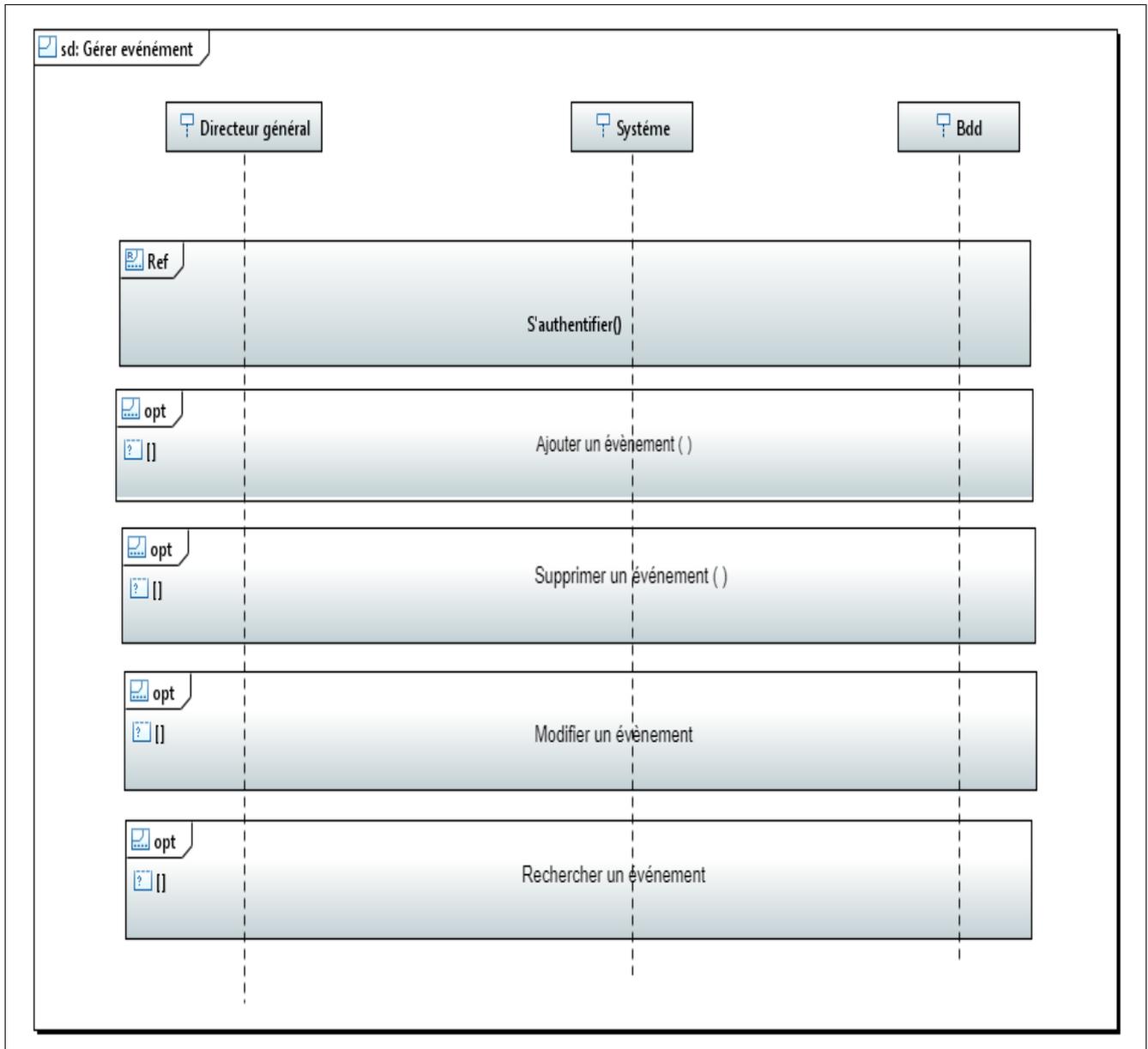


FIGURE 4.20 – Diagramme de séquence « Gérer événement »

Diagramme de séquence « ajouter un événement »

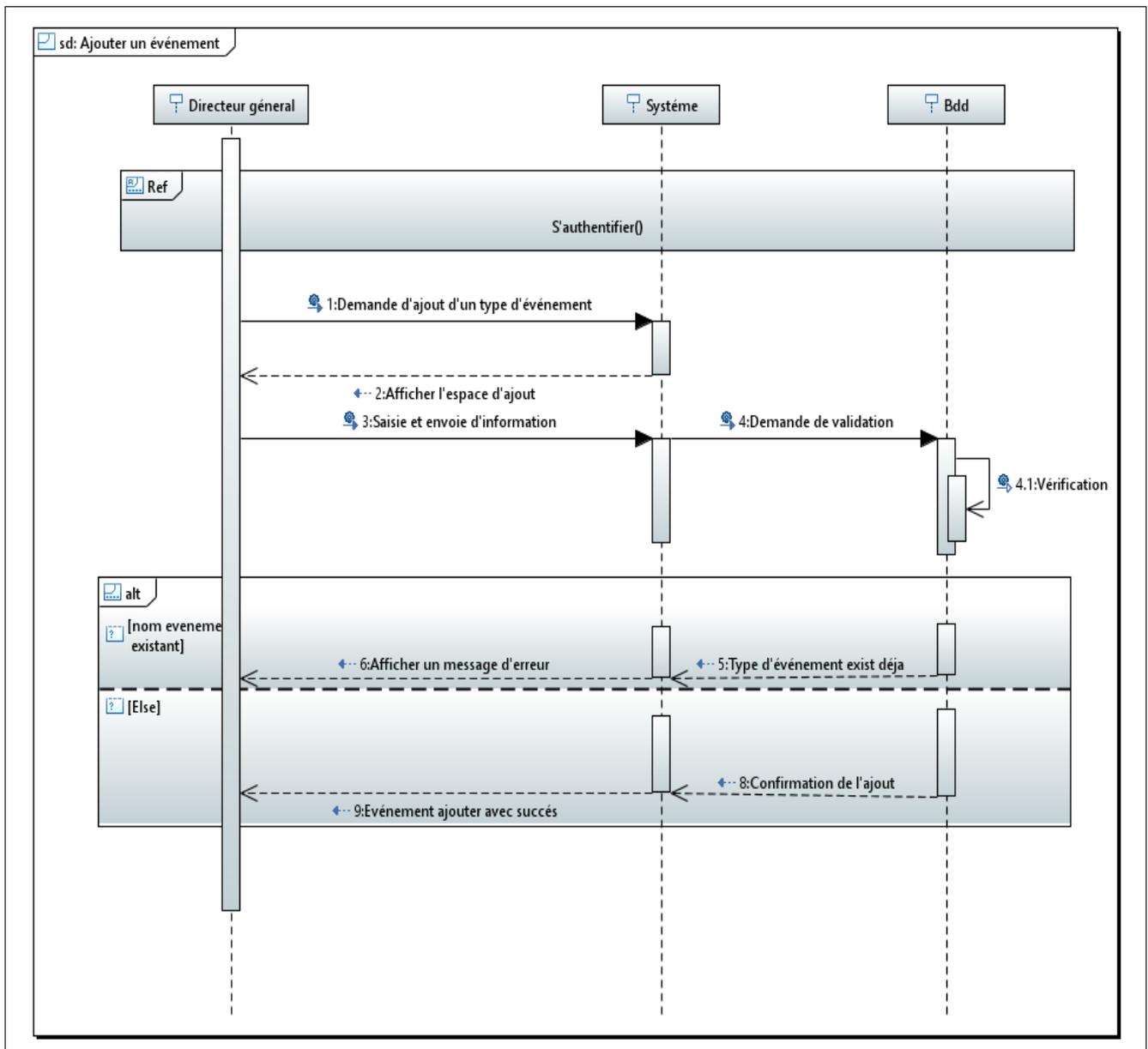


FIGURE 4.21 – Diagramme de séquence « Ajouter un événement »

Diagramme de séquence « Supprimer un événement »

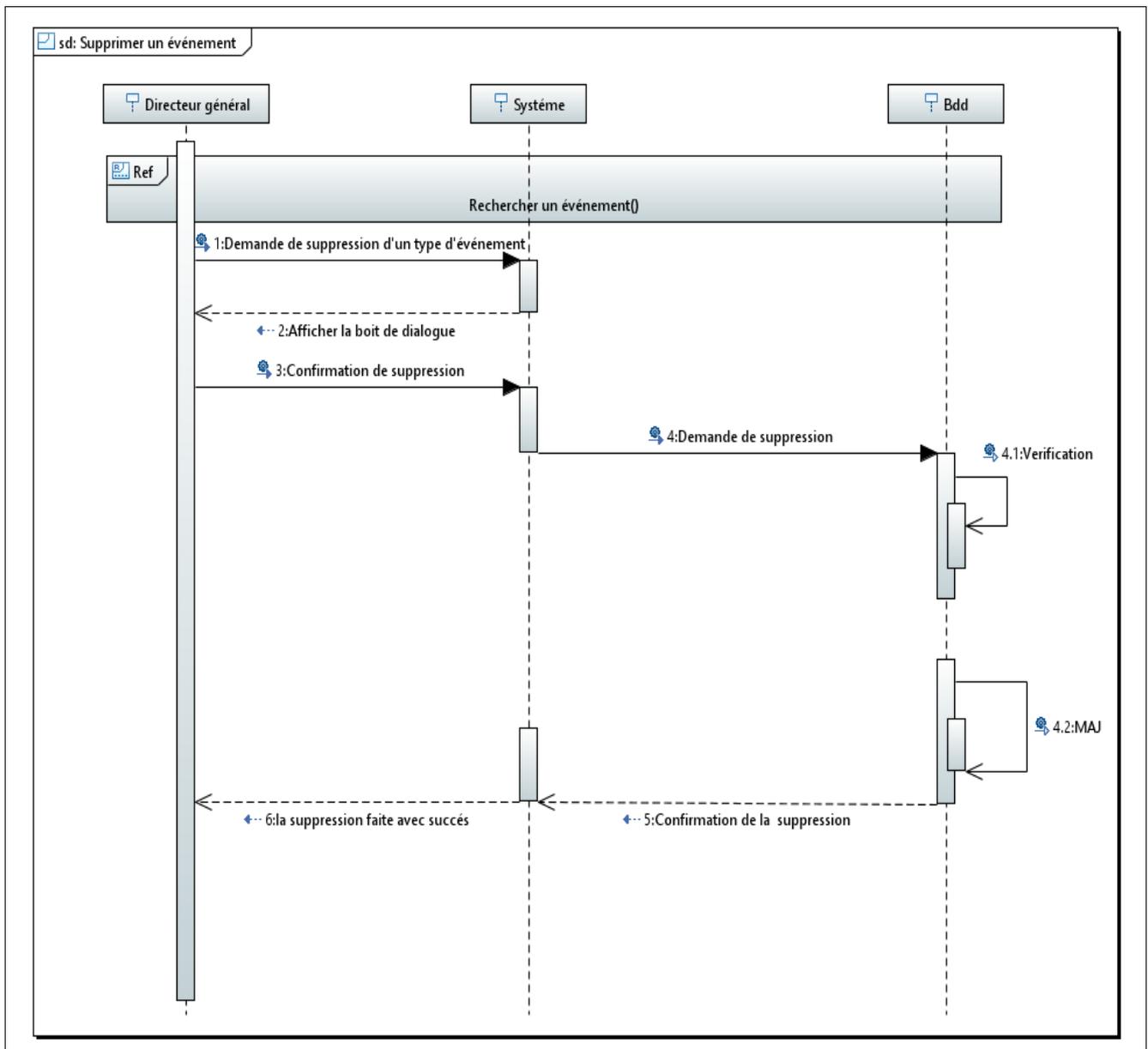


FIGURE 4.22 – Diagramme de séquence « Supprimer un événement »

Diagramme de séquence « Rechercher un événement »

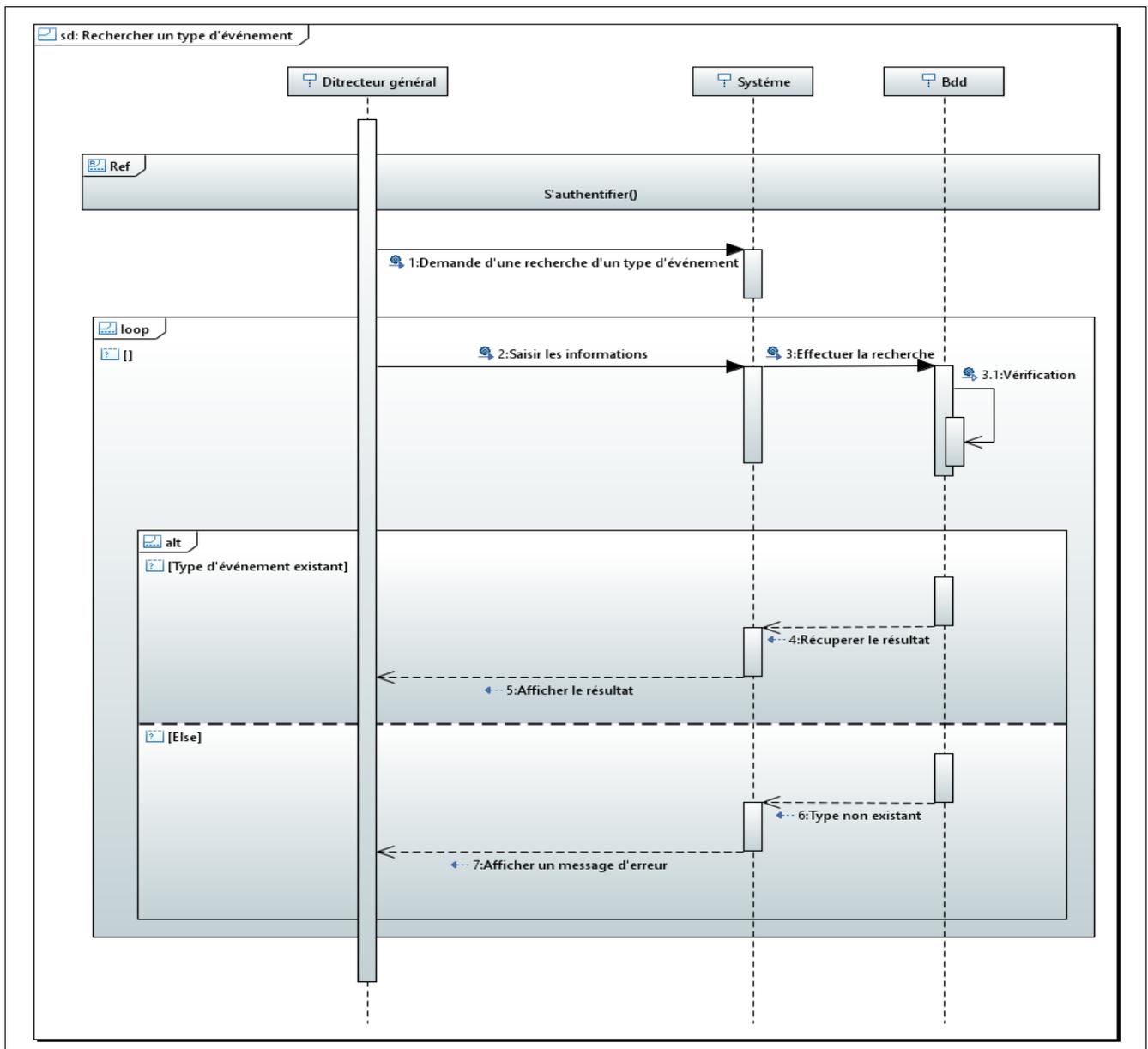


FIGURE 4.23 – Diagramme de séquence « Rechercher un événement »

Diagramme de séquence « Modifier un événement »

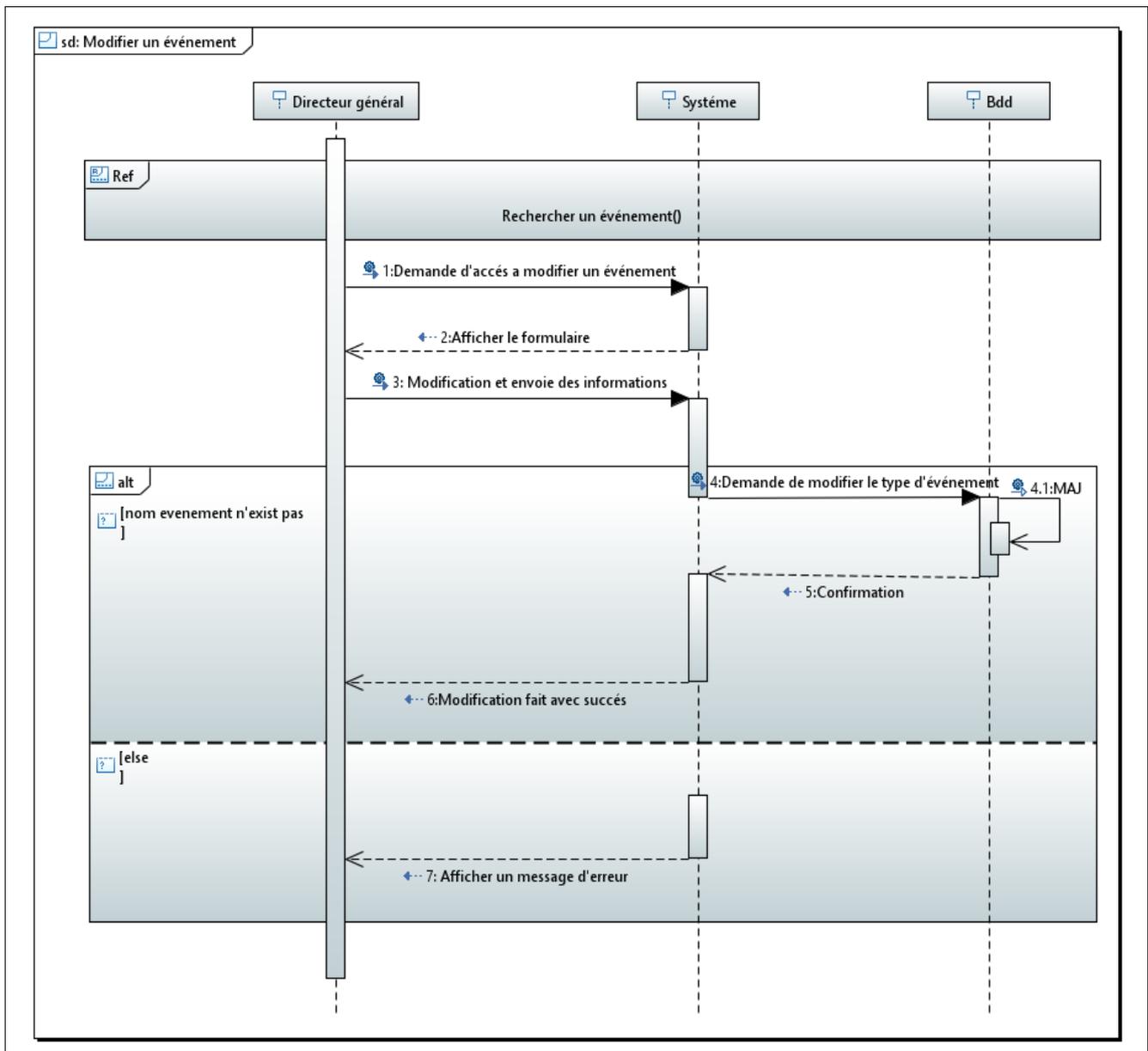


FIGURE 4.24 – Diagramme de séquence « Modifier un événement »

Les diagrammes de séquence gérer les services et gérer les propositions se déroule exactement comme le diagramme de séquence gérer événement

4.4.6 Réalisation

Interface « Gérer les événements »

Cette interface permet d'ajouter un événement ou modifier, rechercher, supprimer ou consulter la liste des événements

The screenshot shows the 'DECIBEL' interface for managing events. The sidebar on the left contains the following menu items: Accueil, Comptes, Evénements, Services, Matériel Sono, Calendrier, and Se Déconnecter. The main content area is titled 'Evénements' and features a search bar with the placeholder text 'Rechercher ici ...'. Below the search bar is a table titled 'Listes Des Evénements' with an 'Ajouter' button. The table has four columns: Id, Nom, Type, and Opération. It displays three rows of event data:

Id	Nom	Type	Opération
1	soutenance	particulier	[Edit] [Delete]
2	mariage	particulier	[Edit] [Delete]
6	bac	particulier	[Edit] [Delete]

At the bottom of the table, it indicates 'Total 4 Page 1 sur 2' and 'premier avant 1 2 apres dernier'.

FIGURE 4.25 – Interface « Gérer les événements »

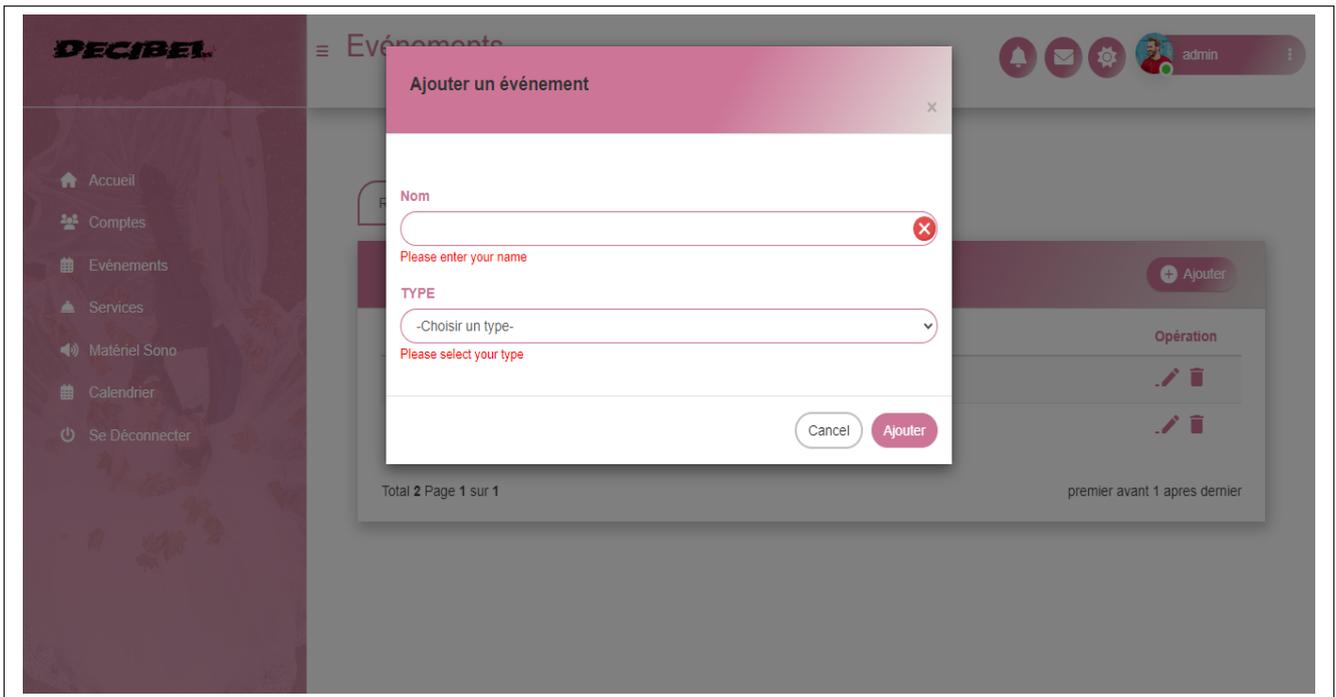


FIGURE 4.26 – Interface « Ajouter un événement lors les champs sont vide »

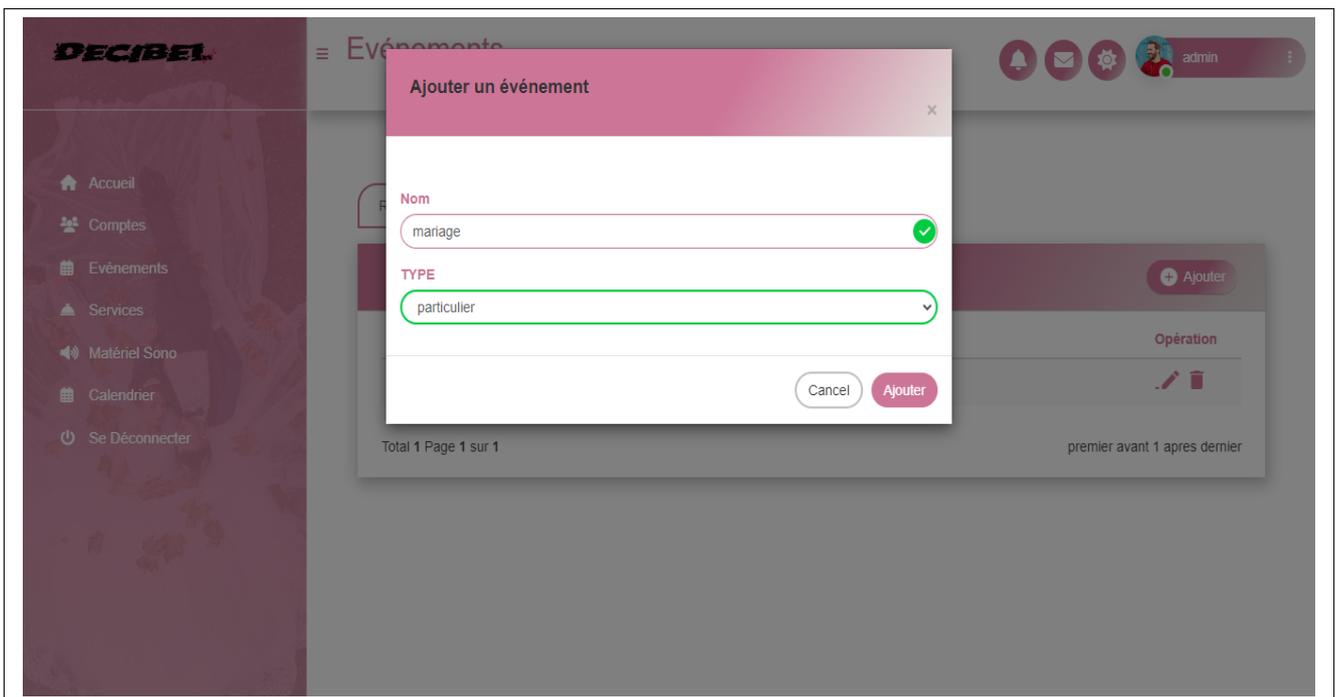


FIGURE 4.27 – Interface « Ajouter un événement »

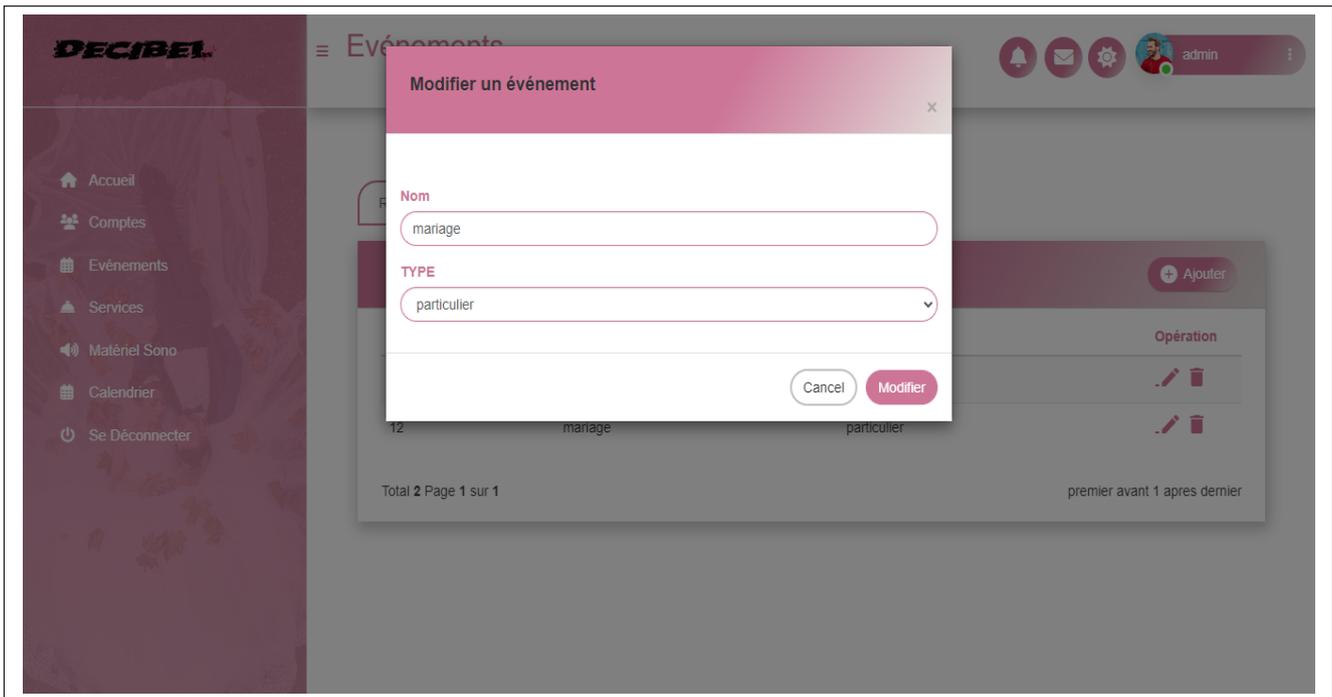


FIGURE 4.28 – Interface « Modifier un événement »

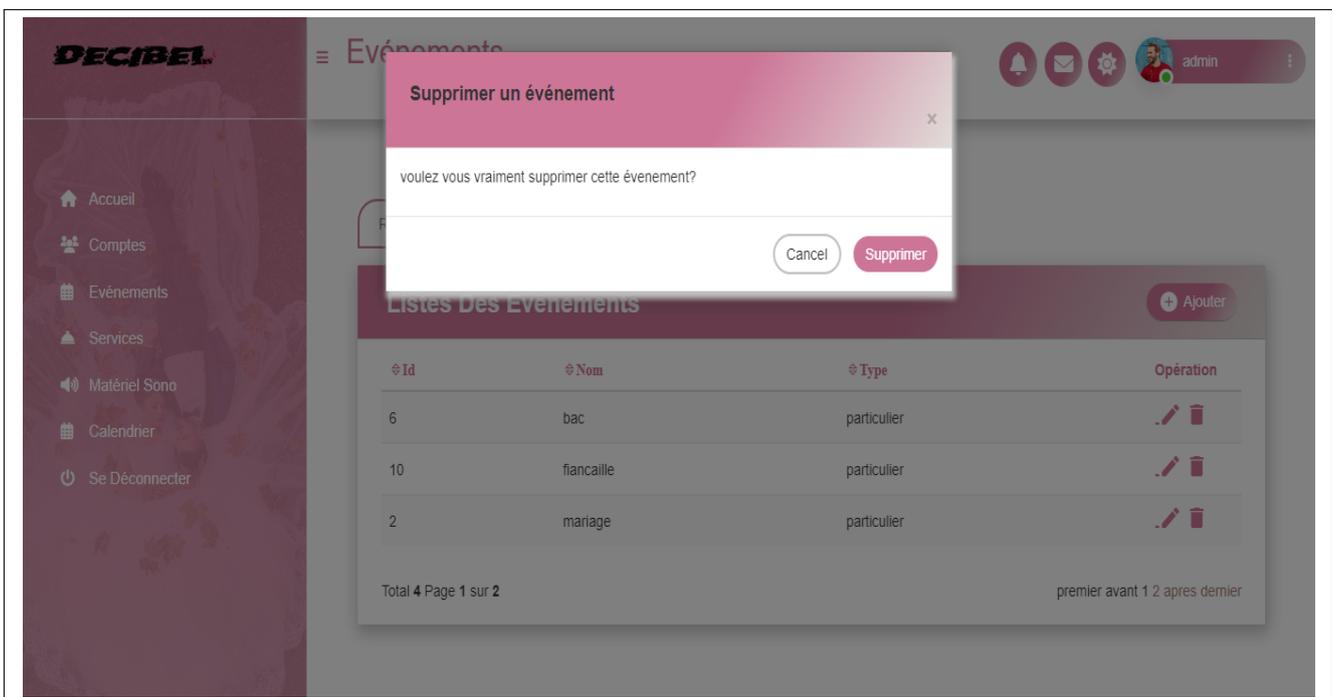


FIGURE 4.29 – Interface « Supprimer un événement »

Interface « Gérer les services »

Cette interface permet d'ajouter un service ou modifier, rechercher, supprimer ou consulter, imprimer la liste des services

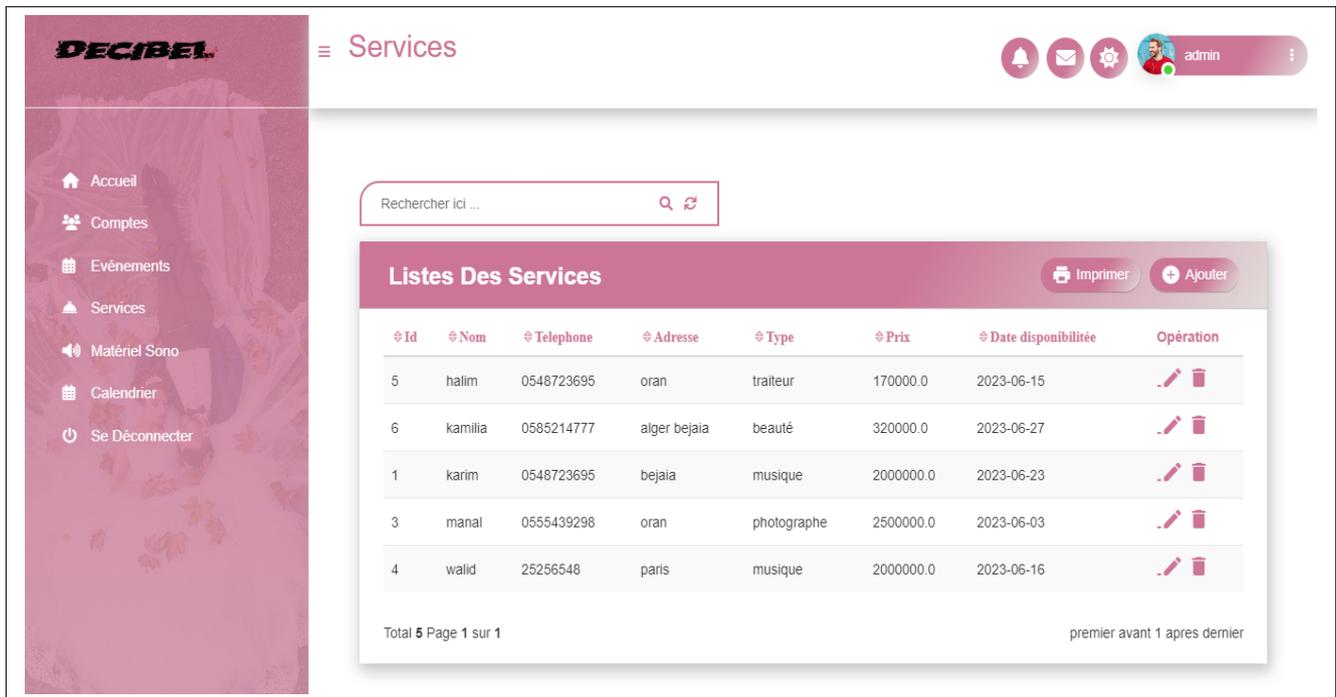


FIGURE 4.30 – Interface « Gérer les services »

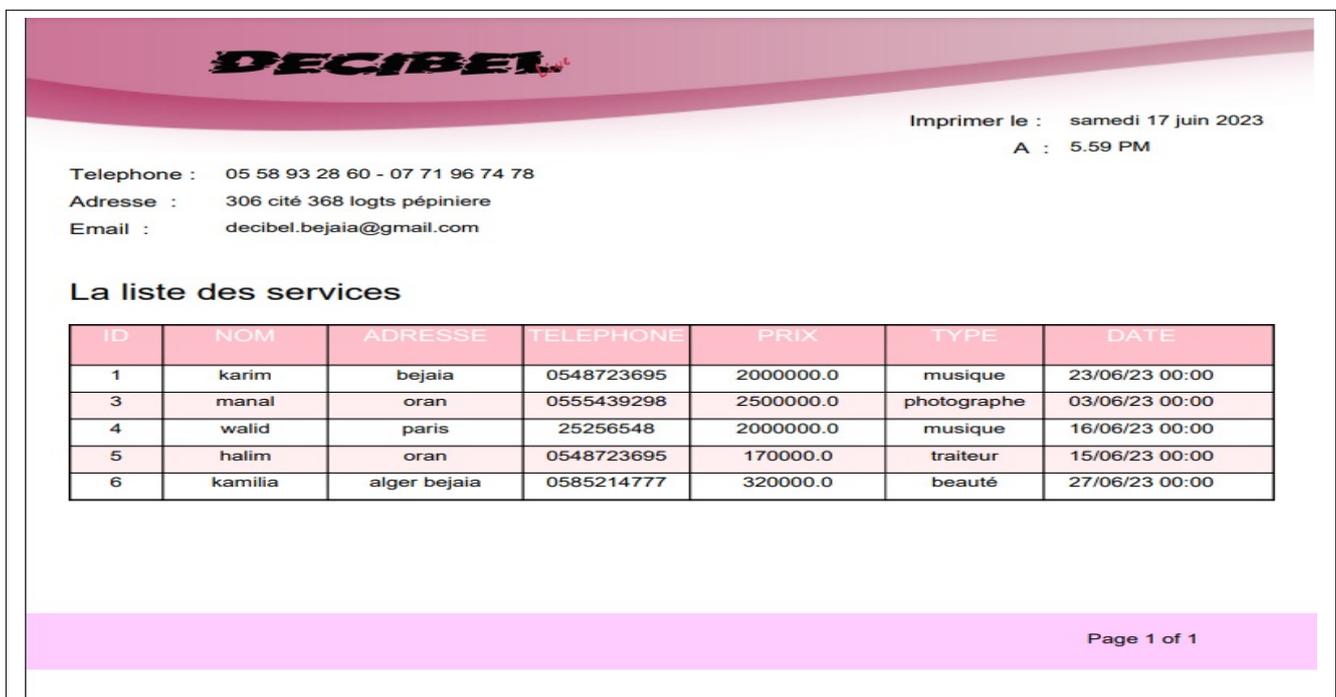


FIGURE 4.31 – Interface « La liste des services imprimé »

Interface « Gérer les propositions »

Cette interface permet d'ajouter une proposition ou modifier, rechercher, supprimer ou consulter la liste des propositions

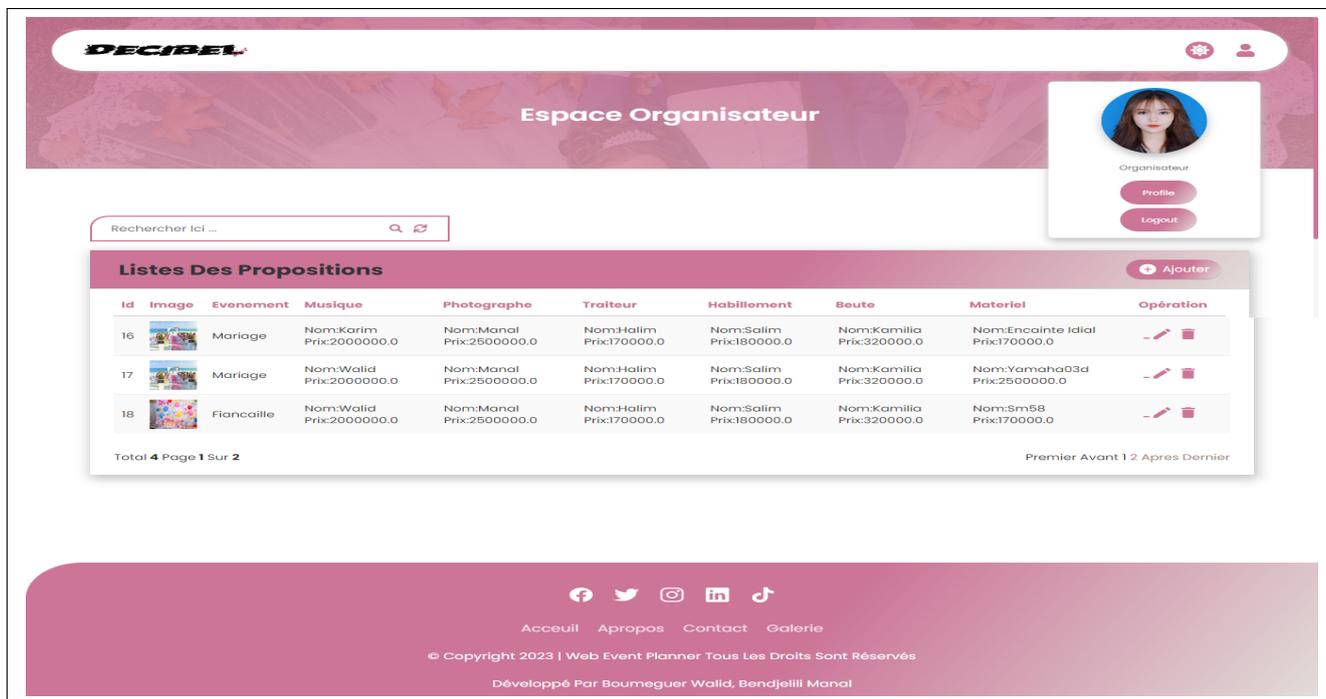


FIGURE 4.32 – Interface « Gérer les propositions »

4.5 Développement du Sprint 04 « Consulter des proposition, poster un commentaire » :

4.5.1 SCRUM Board (task board)

Le SCRUM Board du sprint 04 est le suivant :

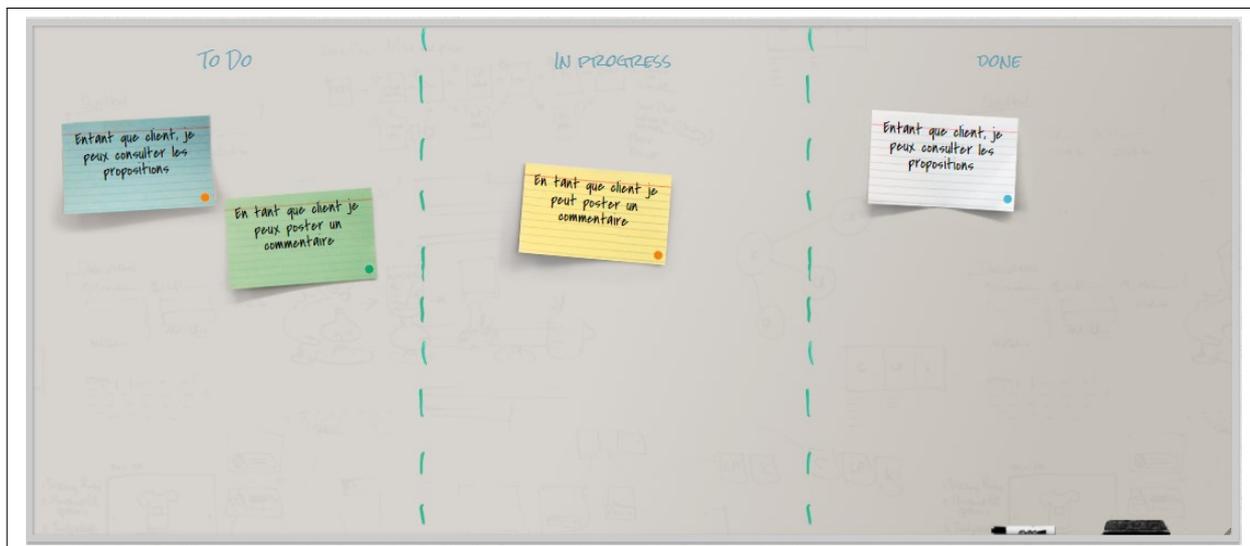


FIGURE 4.33 – SCRUM board du sprint 04

4.5.2 User story du sprint 04

ID	User Story
8.1	En tant que client, je peux consulter les propositions
9.1	En tant que client, je peux poster un commentaire

TABLE 4.12 – User story du sprint 04

4.5.3 Backlog du sprint 04

Item	User Story	Priorité
Consulter une proposition	Consulter une proposition	07
Poster un commentaire	Poster un commentaire	08

TABLE 4.13: Backlog du sprint 04

4.5.4 Analyse

Diagramme de cas d'utilisation du sprint 04

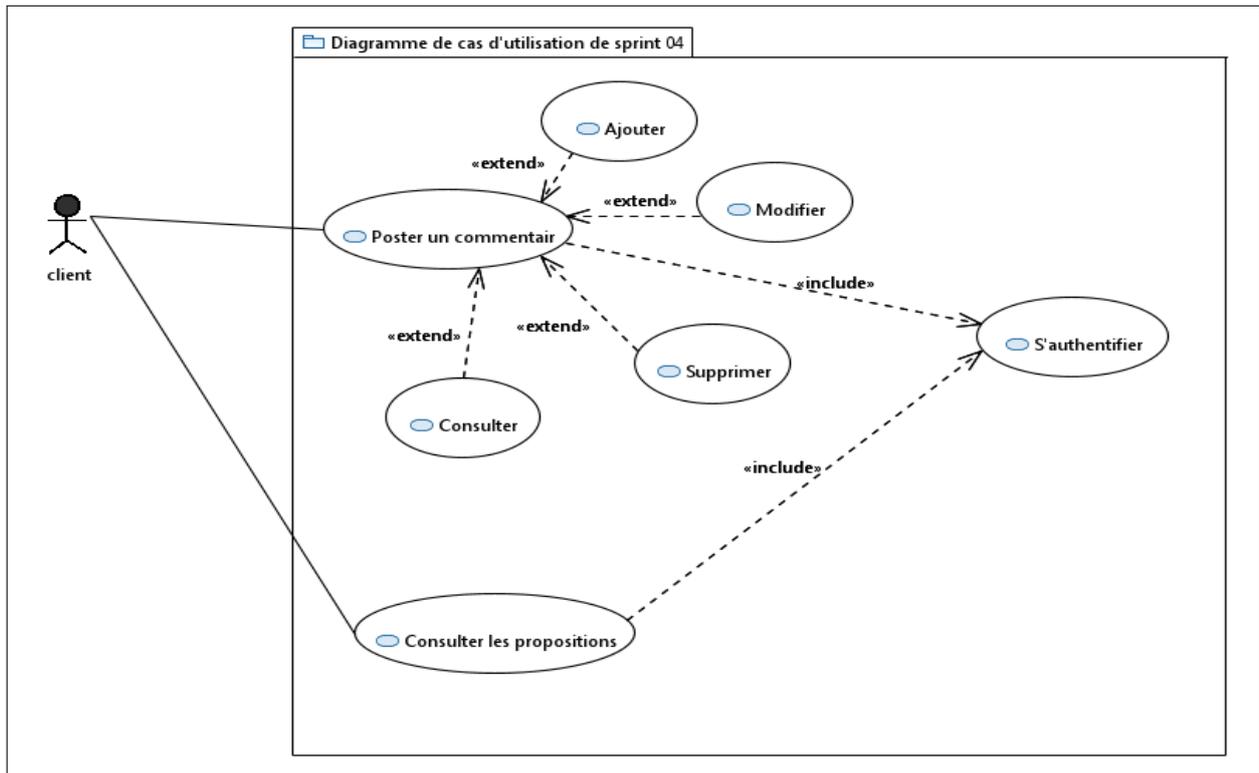


FIGURE 4.34 – Diagramme de cas d'utilisation du sprint 04

Description textuelle des cas d'utilisation

Description textuelle du cas « Consulter proposition »

Cas d'utilisation	Consulter proposition
Acteur	Client
Objectif	Consulter proposition et avoir plus d'information
Scénario nominal	<ol style="list-style-type: none"> 1. Le client accède à son espace 2. Le système affiche une liste des propositions disponibles. 3. L'utilisateur clique sur la proposition qu'il souhaite consulter. 4. Le système affiche les informations détaillées de la proposition, telles que son titre et toute autre information pertinente. 5. L'utilisateur peut voir les détails de la proposition, prendre des notes, ou ajouter la proposition a sa liste des favoris. 6. Une fois la consultation terminée, l'utilisateur peut revenir à la liste des propositions ou effectuer d'autres actions dans l'application
Exception	<p>Cas 01 : lors de l'ajout d'une proposition qui existe dans la liste de ses favoris.</p> <p>le système affiche un message d'erreur.</p>

TABLE 4.14: Description textuelle du cas« Consulter proposition »

Description textuelle « Poster un commentaire »

Cas d'utilisation	Poster un commentaire
Acteur	Client
Précondition	Le client doit s'authentifier/ Accès à l'internet
Objectif	Le pouvoir de poster, modifier ou supprimer un commentaire
Scénario nominal	<p>Poster un commentaire</p> <ol style="list-style-type: none"> 1. Le client accède à son espace 2. le client choisi de poster un commentaire 3. Le client poste son commentaire 4. Le système affiche commentaire poster avec succès
Scénario alternatif	<p>Cas 01 : Champs non conformes</p> <p>Le système affiche un message d'erreur</p>

TABLE 4.15: Description textuelle du cas« Poster un commentaire »

4.5.5 Conception

Diagramme de séquence « Consulter proposition »

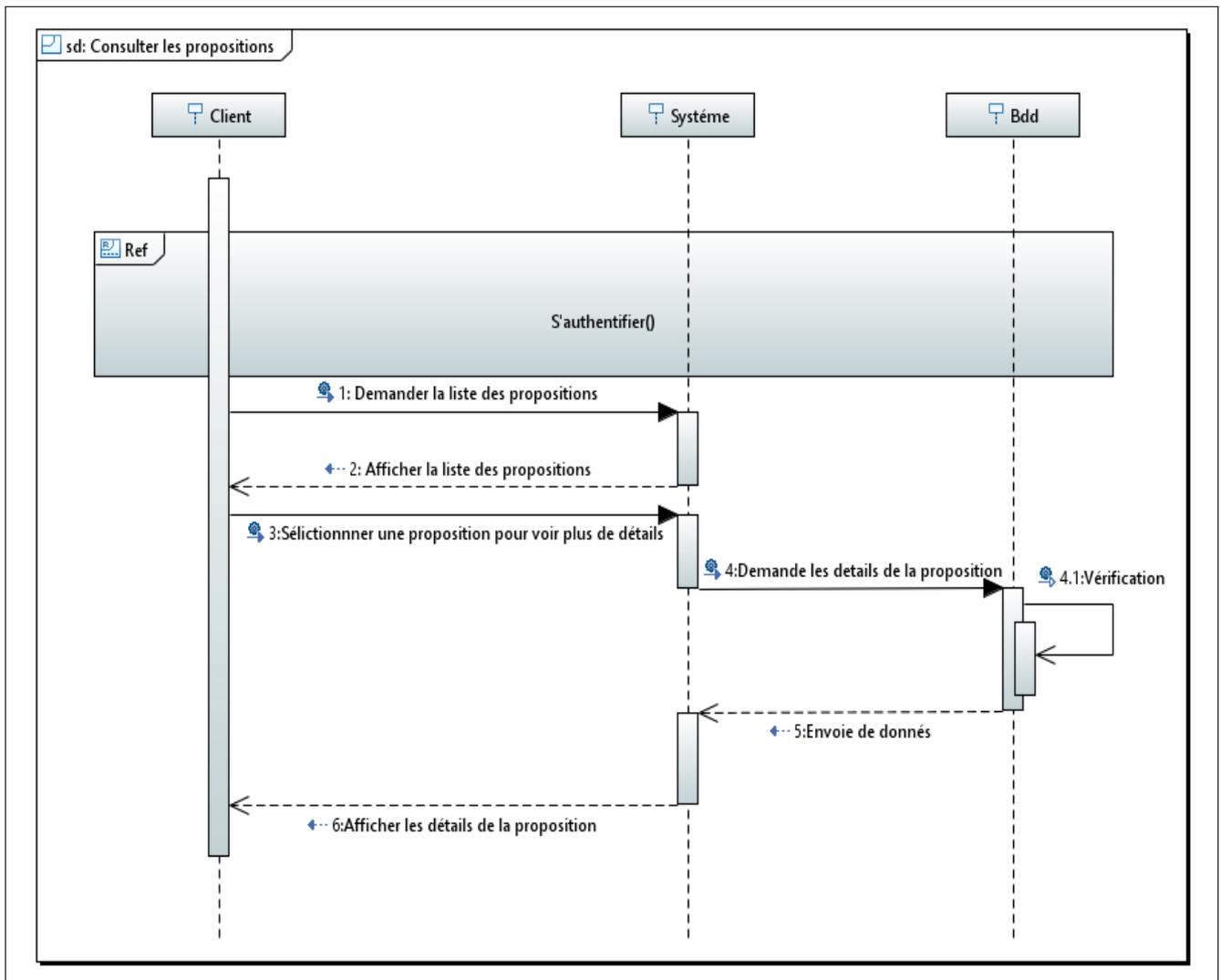


FIGURE 4.35 – Diagramme de séquence « Consulter proposition »

Le diagramme de séquence de poster un commentaire suit les mêmes étapes que le diagramme de séquence gérer événement

4.5.6 Réalisation

Interface « Consulter proposition »

Cette interface permet d'afficher les propositions saoulent l'événement en pleine d'écran pour le client. Le client peut voir plus de détail d'une proposition ou l'ajouter à sa liste des favoris.

Le client peut supprimer une proposition ou plusieurs à la fois ,imprimer sa liste des favoris.

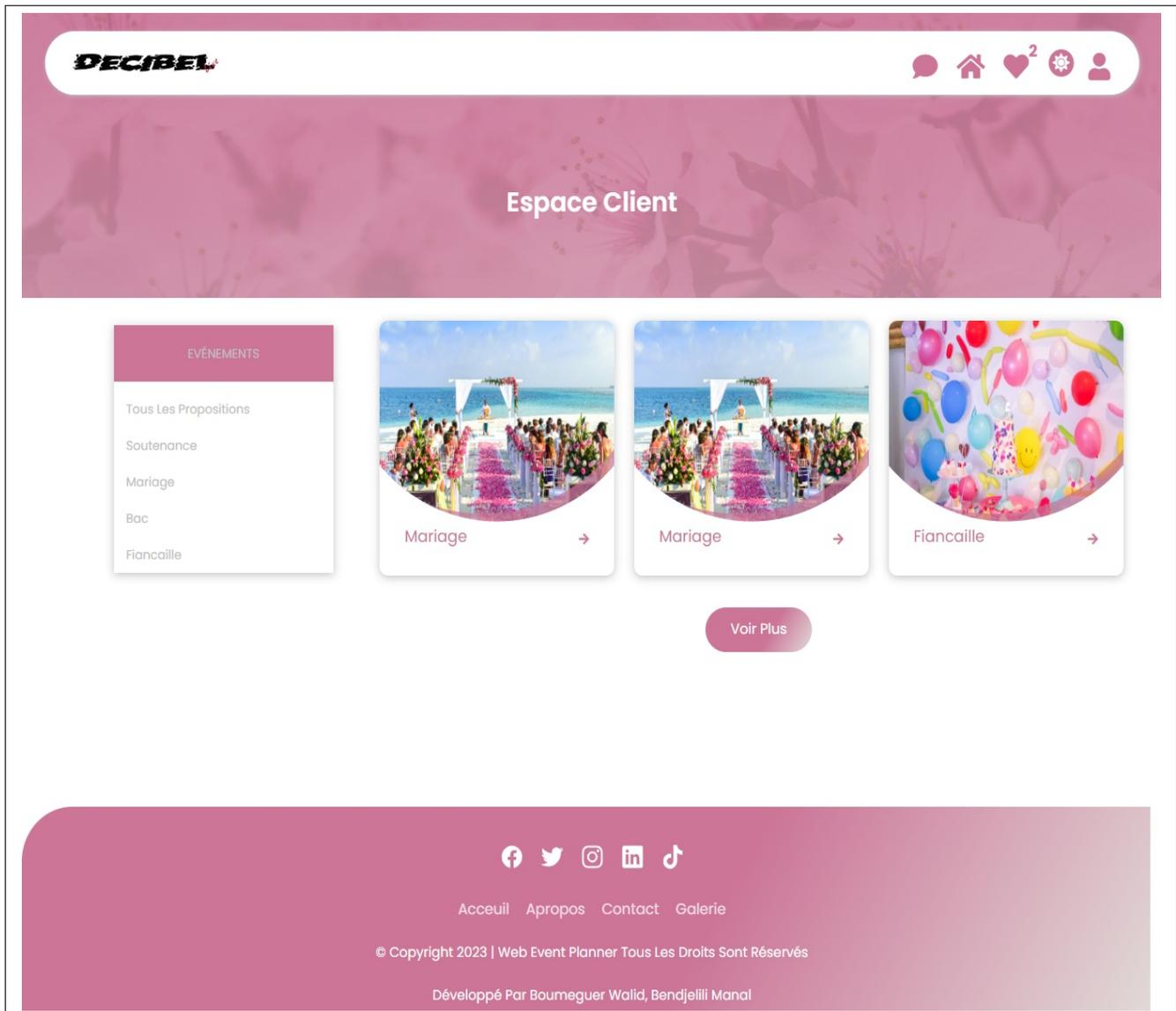


FIGURE 4.36 – Interface « Consulter proposition »

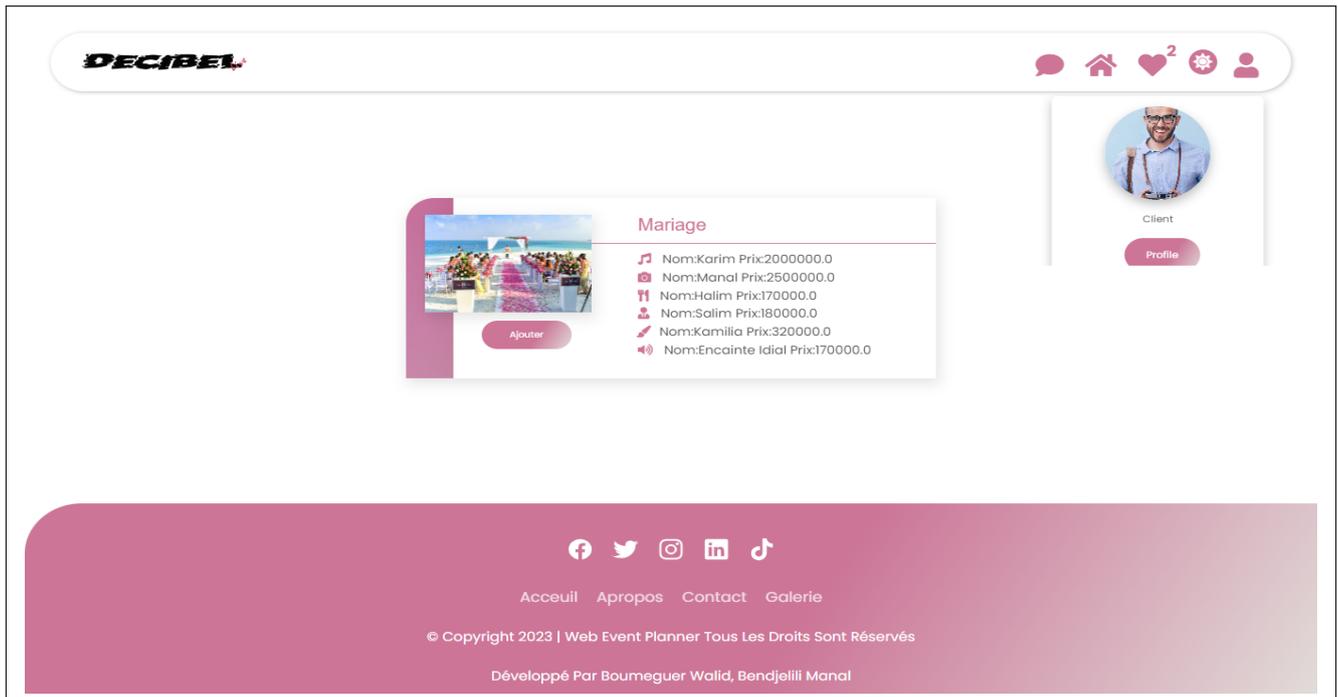


FIGURE 4.37 – Interface « Voir les détails d’une proposition »

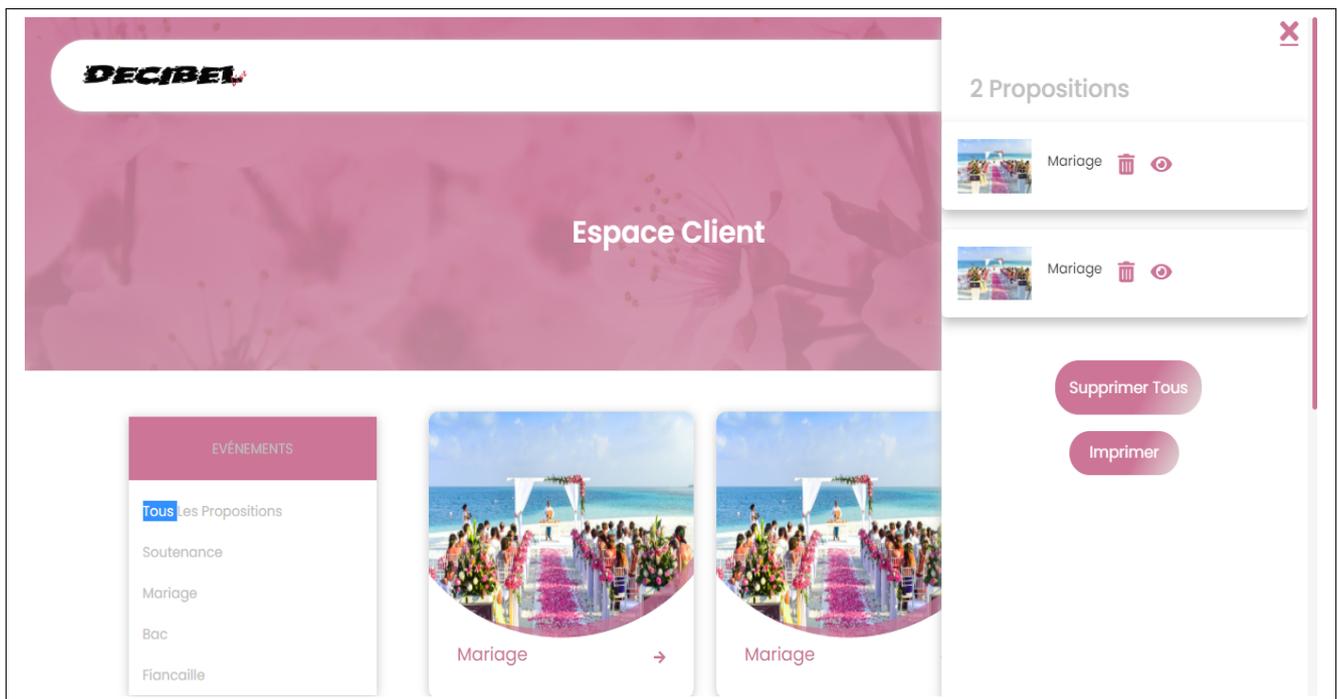


FIGURE 4.38 – Interface « La liste des favoris des propositions »

Interface « Poster un commentaire »

Cette interface permet de poster un commentaire

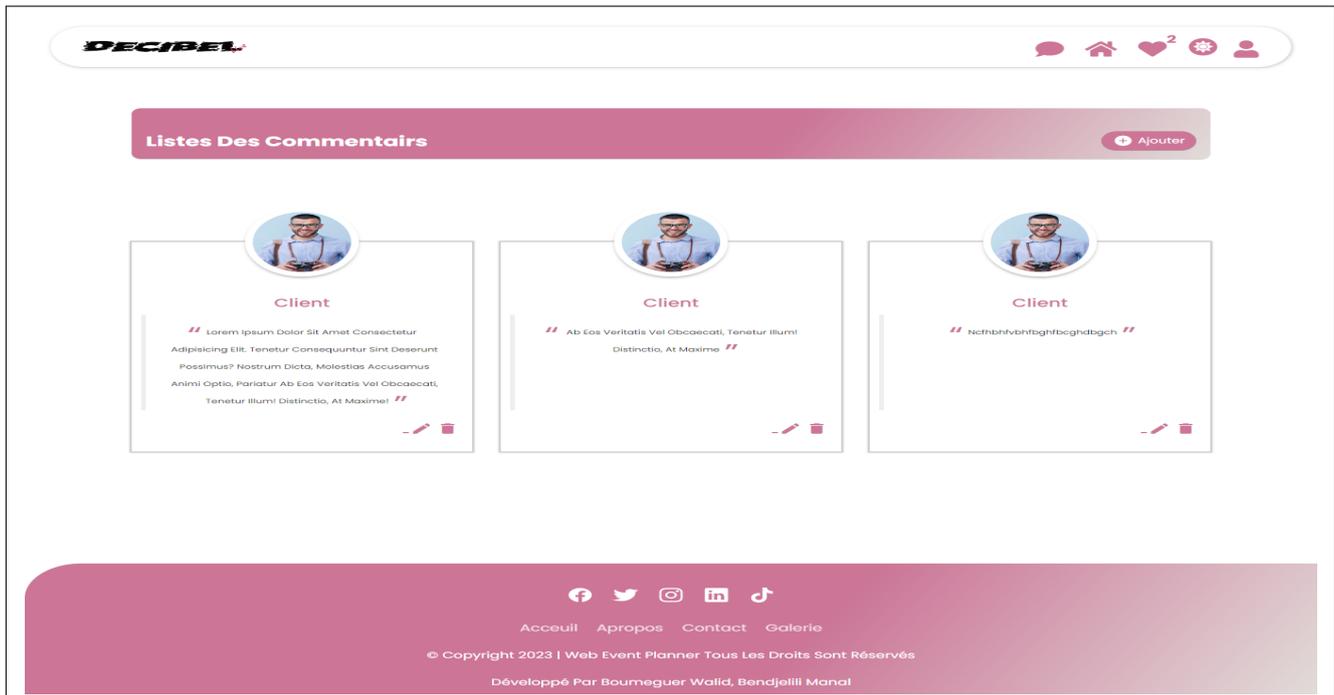


FIGURE 4.39 – Interface « Poster un commentaire »

4.6 Dictionnaire de donnée

Le dictionnaire de données répertorie tous les termes et leurs définitions.

Class	Attributs	Signification	Types	Taille	Méthodes
Utilisateurs	Id_U	Identifiant d'utilisateur	Long	30	Supprimer()
	Nom_U	Nom d'utilisateur	String	30	Valider()
	Adresse_U	Adresse d'utilisateur	String	30	Rechercher()
	Num_tel_U	Numéro de téléphone	String	30	Consulter()
	Email_U	L'email d'utilisateur	String	30	
	Mdp_U	Mot de passe d'utilisateur	String	30	
	Role_U	Role d'utilisateur	String	30	
Événements	Image_U	Image d'utilisateur	String	30	
	Id_E	Identifiant de l'événement	Long	30	Ajouter() Rechercher() Consulter()
	Nom_E	Nom d'événement	String	30	Modifier()
	Type_E	Type d'évènement	String	30	Supprimer()

Matériel-sonorisation	Id_M	Identifiant de matériel	Long	30	Ajouter()
	Nom_M	Nom de matériel	String	30	Modifier()
	Prix_M	Prix de matériel	Float	30	Supprimer()
	Image_M	Image de matériel	String	30	Consulter() Rechercher()
Services	Id_S	Identifiant de service	Long	30	Ajouter()
	Type_S	Type de service	String	30	Modifier()
	Nom	Nom de service	String	30	Supprimer()
	Tel	Numéro _{tel} de service	String	30	Rechercher()
	Adresse	Adresse de service	String	30	Consulter()
	Prix	Prix de service	Float	30	
	Date_disp_S	Date de disponibilité du service	String	30	
Proposition	Id_P	Identifiant de proposition	Long	30	Ajouter()
	evenement_P	Nom d'événement	String	30	Modifier()
	Image_P	Photo de proposition	String	30	Supprimer()
	Photographe	Nom de Photographe	String	30	Rechercher()
	Musique	Nom de musicien	String	30	Consulter()
	Traiteur	Nom de traiteur	String	30	
	Habillement	Nom d'habillement	String	30	
	Beauté	Nom de coiffeur	String	30	
	Material_sono	Nom de matériel de sonorisation	String	30	
Commentaire	Id_c	Identifiant de commentaire	Long	30	Ajouter()
	Commentaire	le commentaire de client	String	30	Modifier()
	Nom_C	Nom de client	String	30	Supprimer()
	Image_C	Image de profile de client	String	30	Modifier()
Louer	Date_dispM	Date de disponibilité	Long	30	

TABLE 4.16: Glossaire de données

4.7 Diagramme de classe

Le diagramme de classe est une représentation visuelle utilisée en informatique pour décrire les classes, les interfaces et les relations entre celles-ci dans un système. Cette représentation est une partie du diagramme statique UML [14].

La figure qui suit représente le diagramme de classe de notre application :

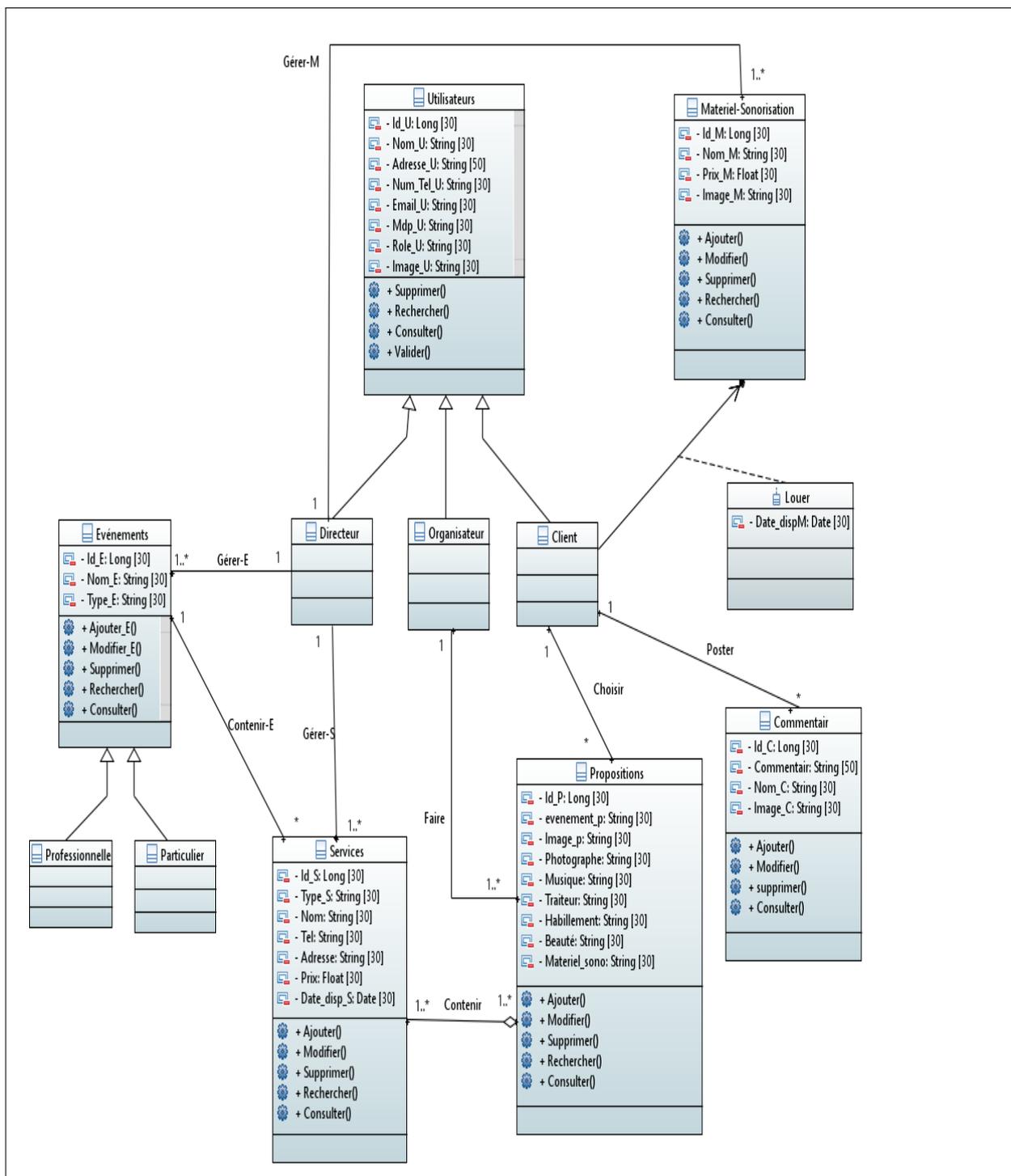


FIGURE 4.40 – Diagramme de classe

4.8 Schéma relationnel

4.8.1 Définition

Le modèle relationnel est un modèle logique de donnée qui décrit le schéma d'une base de données relationnelle. Ce schéma comprend des tables, des champs pour chaque table avec leurs propriétés, des clés primaires pour les tables, des clés étrangères permettant de lier les tables entre elles, ainsi que des contraintes d'intégrité portant sur ces liaisons [16].

4.8.2 Règles de passage du diagramme de classe au modèle relationnel

Les règles de passage du diagramme de classe au modèle relationnel :

- **Règle 1 :** Toute entité devient une relation ayant pour clé primaire son identifiant. Chaque propriété se transforme en attribut.
- **Règle 2 :** Toute association hiérarchique de type $[1, n]$ se traduit par une clé étrangère. La clé primaire correspondant à l'entité père (côté n) migre comme clé étrangère dans la relation correspondant à l'entité fils (côté 1).
- **Règle 3 :** Toute association non hiérarchique de type $[n, n]$ devient une relation. La clé primaire est formée par la concaténation de l'ensemble des identifiants des entités reliées. Toutes les propriétés éventuelles deviennent des attributs qui ne peuvent pas faire partie de la clé.
- **Règle 4 :** La clé primaire des relations déduites des classes composantes doit contenir l'identifiant de la classe composite (quel que soit les multiplicités).
- **Règle 5 :** Trois décompositions sont possibles pour traduire une association d'héritage en fonction des contraintes existantes :
 - Décomposition descente (push-down) : s'il existe une totalité ou de partition sur l'association d'héritage, il est possible de ne pas traduire la relation issue de la surclasse. Il faut alors faire migrer tous ses attributs dans la relation issue de la sous-classe.
 - Décomposition ascendante (push-up) : il faut supprimer la relation issue de la sous-classe et faire migrer les attributs dans la relation issue de la surclasse.
 - Décomposition par distinction : Il faut transformer chaque sous-classe à une relation. La clé primaire de la surclasse migre dans la relation issue de la sous-classe et devient à la fois clé primaire et la clé étrangère.

4.8.3 Modèle relationnel de donnée

En appliquant les règles de passages pour au diagramme de classes, nous aboutissons au schéma relationnel suivant :

1-Utilisateurs (Id_U,Nom_U,Adresse_U,Num_Tel_U,Email_U,Mdp_U,Role_U,Image_U)

2-Événements (Id_E, Nom_E, Type_E, #Id_U)

3-Services (Id_S,Type_S,Nom,Tel,Adresse,Prix,Date_disp_S,#Id_U,#Id_E)

4-Matériel-Sonorisation(Id_M,Nom_M,Image_M,Prix_M,#Id_U)

5-Commentaire(Id_C,Commentaire,Nom_C,Image_C,#Id_U)

5-Propositions (Id_P,evenement,Image_p,Photographen,Musique,Traiteur,Beauté,Materiel_sono, Habillement,#Id_U,#Id_U)

6-Contenir (#Id_S,#Id_P)

7-Louer (Date_disp,#Id_U,#Id_M)

4.9 Conclusion

Au terme de ce chapitre, nous avons atteint notre objectif en fournissant une version améliorée à travers de la création d'une application web service répondant aux exigences du client. Nous avons également fourni une description de l'application de la méthode SCRUM sur notre projet avec une présentation de quelques exemples d'interfaces. Enfin, nous terminons ce rapport avec une conclusion générale.

Conclusion générale

Ce mémoire de fin d'études a montré que la création d'une application Web service pour l'organisation des événements offre de nombreux avantages tant pour les organisateurs que pour les participants. Cette solution permet d'optimiser la gestion des ressources, de favoriser la communication et de créer une expérience utilisateur plus satisfaisante.

Il est important de souligner que ce domaine est en constante évolution, avec de nouvelles technologies et des attentes changeantes des utilisateurs. Il convient donc de continuer à suivre les tendances et à s'adapter aux besoins du marché pour maintenir l'application à jour et compétitive.

Finalement, nous espérons que ce mémoire a apporté une contribution significative à la compréhension et à l'amélioration de l'organisation des événements grâce à une application Web service. Nous encourageons les entreprises, les organisations et les futurs chercheurs à explorer davantage ce domaine passionnant et à développer des solutions innovantes pour répondre aux défis de l'organisation des événements à l'ère numérique.

Les perspectives

- L'inscription avec google.
- La visioconférence

Bibliographie

- [1] Subra, J.-P. (13 septembre 2017). *Méthode agile pour vos projets*. Editions ENI . ISBN 2409009662.
- [2] Sutherland, K. S. (2017, Novembre Jeu). Développé et maintenu par les créateurs de SCRUM,Le Guide de Référence de SCRUM.
- [3] Kurzawa, G. (2017, JUIN). *AGILITE EN PRATIQUE :SCRUM,Un cadre propice à la transparence et à l'amélioration continue* .
- [4] F. Lothon. (2010). *Guide de démarrage SCRUM*. L'Agiliste. ISBN 978-2-1234-5678-9.
- [5] DEBRAUWER, N. K.-F.-L. (2009). *UML2 Maîtrisez le langage de modélisation*. Editions ENI .
- [6] LOPEZ-VELASCO, C. *Sélection et thèse :omposition de services Web pour la génération d'applications adaptées au contexte d'utilisation*. Thèse de doctorat, UNIVERSITE JOSEPH FOURIER, novembre 2018.
- [7] BOUZERA Toufik, H. I.LOPEZ-VELASCO, C. *Conception et réalisation d'un parseur de service web* . Mémoire de fin d'étude université de Bejaia 2011.
- [8] NEWCOMER, E. *Understanding Web services- XML, WSDL, SOAP and UDDI*.wesley édition
- [9] MELITI, T. *Interopérabilité des services web complexes. Application système multiagents*. Mémoire de magister,Université de paris IX Dauphine, Decembre 2004.
- [10] Maesano.L, C. B. *Services web avec j2ee et net Conception et implémentations*. Editions Eyrolles .ISBN 2-212.
- [11] Leblanc, M. *Les Web services : Définition, technologies, acteurs, impacts sur les entreprises et problèmes* . Rapport de projet ,novembre 2002.
- [12] Graham .S, D. *Building Web Services with Java Making sense of XML,SAOP, WSDL,AND UDDI*. Sams Publishing,2005.
- [13] Benouaret, K.*Benouaret, K. (2012, 10 09). Thèse :Advanced Techniques for Web Service Query Optimization*. Thèse de doctorat,Université Claude Bernard Lyon 1, octobre 2012.

-
- [14] Pascal R. *UML 2 par la pratique : études de cas et exercices corrigés*. Éditeur Eyrolles, Paris, 2011.
- [15] Bertrand M. *Introduction to the Theory of Programming Languages*. New York, États-Unis : Springer, 2009
- [16] Begg, T. C. *Database Systems : A Practical Approach to Design, Implementation, and Management*. Éditeur Pearson, Upper Saddle River, NJ (États-Unis), 2014. ISBN 0132943263

Webographie

- [17] <https://www.nutcache.com/fr/blog/les-methodes-agiles/>. *les-methodes-agiles*. Consulté le 10/02/2023.
- [18] <https://lyontesting.fr/fr/du-bon-usage-des-user-story-et-epic/>. Consulté le 09/02/2023.
- [19] <https://developer.mozilla.org/fr/docs/Glossary/MVC>. Consulté le 10/02/2023.
- [20] <https://spring.io/tools>. Consulté le 09/02/2023.
- [21] <https://www.phpmyadmin.net>. Consulté le 09/02/2023.
- [22] <https://www.mysql.com/fr/>. Consulté le 09/02/2023.
- [23] <https://www.wampserver.com/>. Consulté le 09/02/2023.
- [24] <https://www.eclipse.org/papyrus/>. Consulté le 09/02/2023.
- [25] <https://www.udemy.com/fr/topic/adobe-illustrator/>. Consulté le 14/02/2023.
- [26] <https://www.latex-project.org/>. Consulté le 14/02/2023.
- [27] <https://developer.mozilla.org/en-US/docs/Web/HTML>. Consulté le 14/02/2023.
- [28] <https://www.w3schools.com/css>. Consulté le 09/02/2023.
- [29] <https://www.w3schools.com/js/>. Consulté le 13/02/2023.
- [30] <https://sql.sh/>. Consulté le 09/02/2023.
- [31] <https://www.java.com/en/>. Consulté le 14/02/2023.
- [32] <https://blog.paumard.org/cours/jpa/>. Consulté le 14/02/2023.
- [33] https://www.tutorialspoint.com/spring_boot. Consulté le 16/02/2023.
- [34] <https://getbootstrap.com>. Consulté le 16/02/2023.
- [35] <https://www.thymeleaf.org/>. Consulté le 16/02/2023.
- [36] <https://tomcat.apache.org/>. Consulté le 16/02/2023.
- [37] <https://www.redhat.com/fr/topics/api/what-is-a-rest-api>. Consulté le 16/02/2023.
- [38] <https://www.uml.org/>. *UML - Unified Modeling Language*. Consulté le 11/02/2023.

Annexe Interface « Accueil »

Cette interface présente l'accueil de notre site



FIGURE 4.41 – Interface « L'accueil de notre site »

Interface « À-propos »

Cette interface permet d'afficher plus d'informations sur l'entreprise

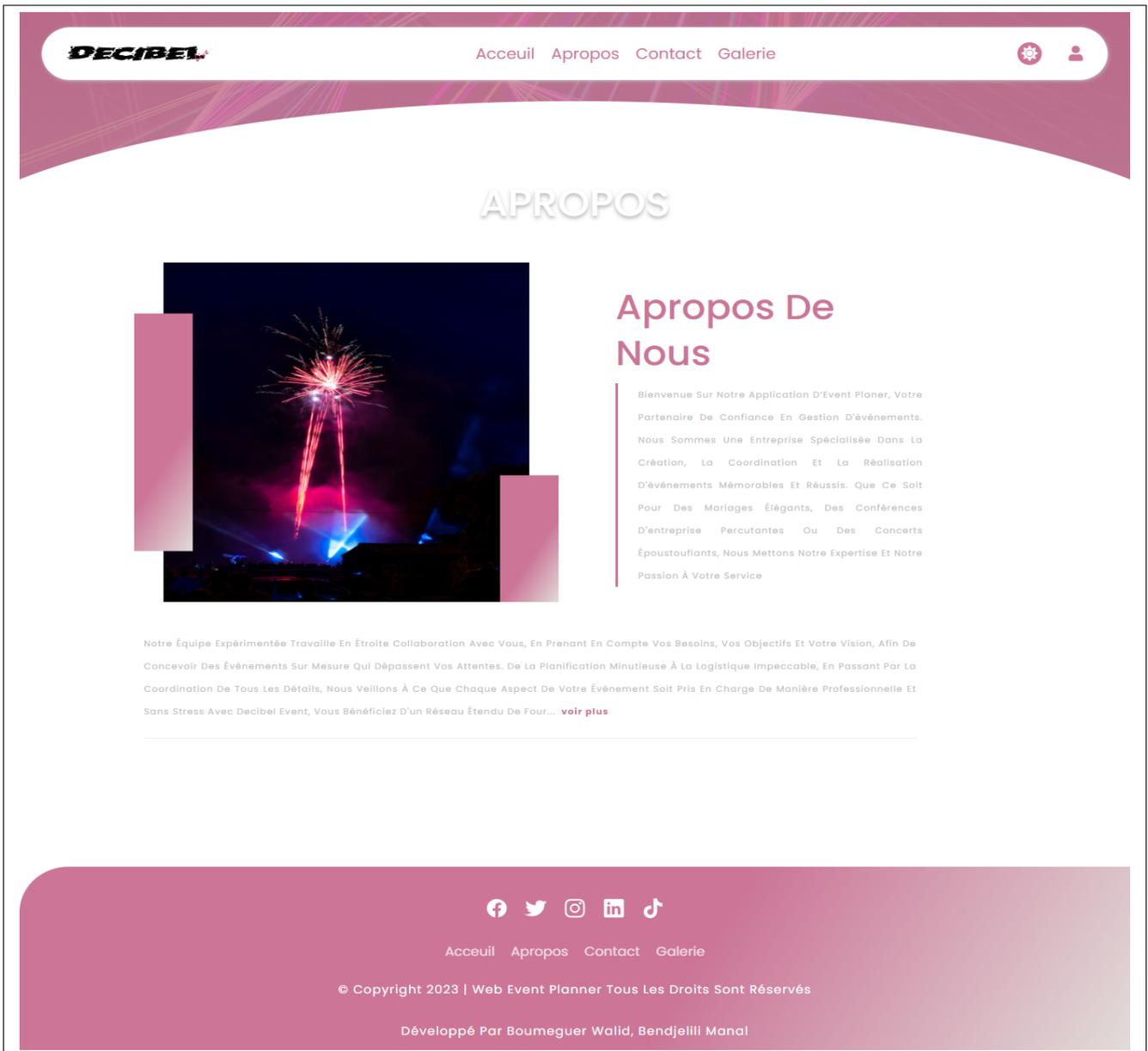


FIGURE 4.42 – Interface « À-propos »

Interface « Contact »

Cette interface permet aux utilisateurs d'envoyer un mail ou de contacter l'entreprise soit avec téléphone ou trouver leur adresse

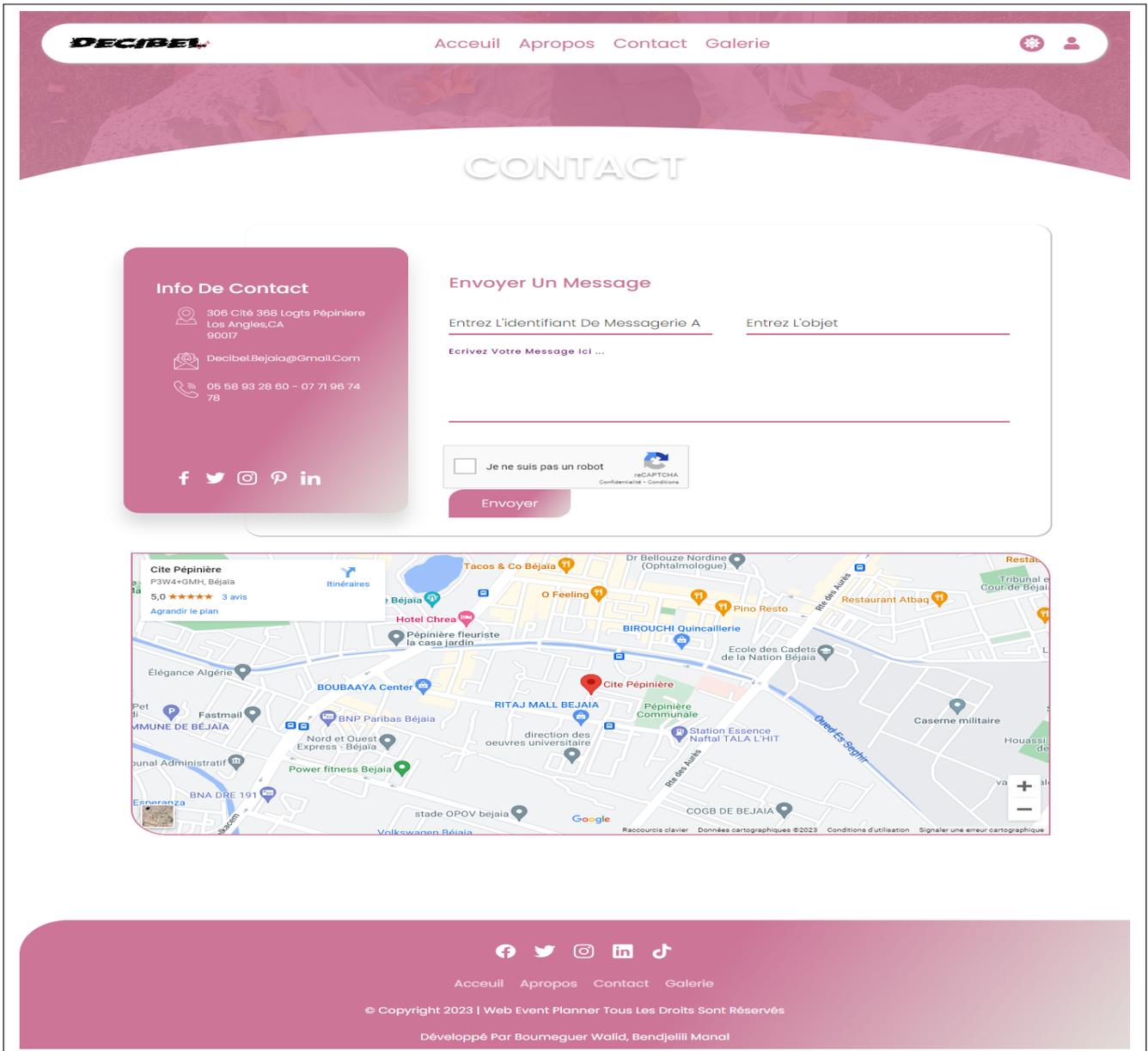


FIGURE 4.43 – Interface « Contact »

Interface « Galerie »

Cette interface permet d'afficher plusieurs image de différent proposition

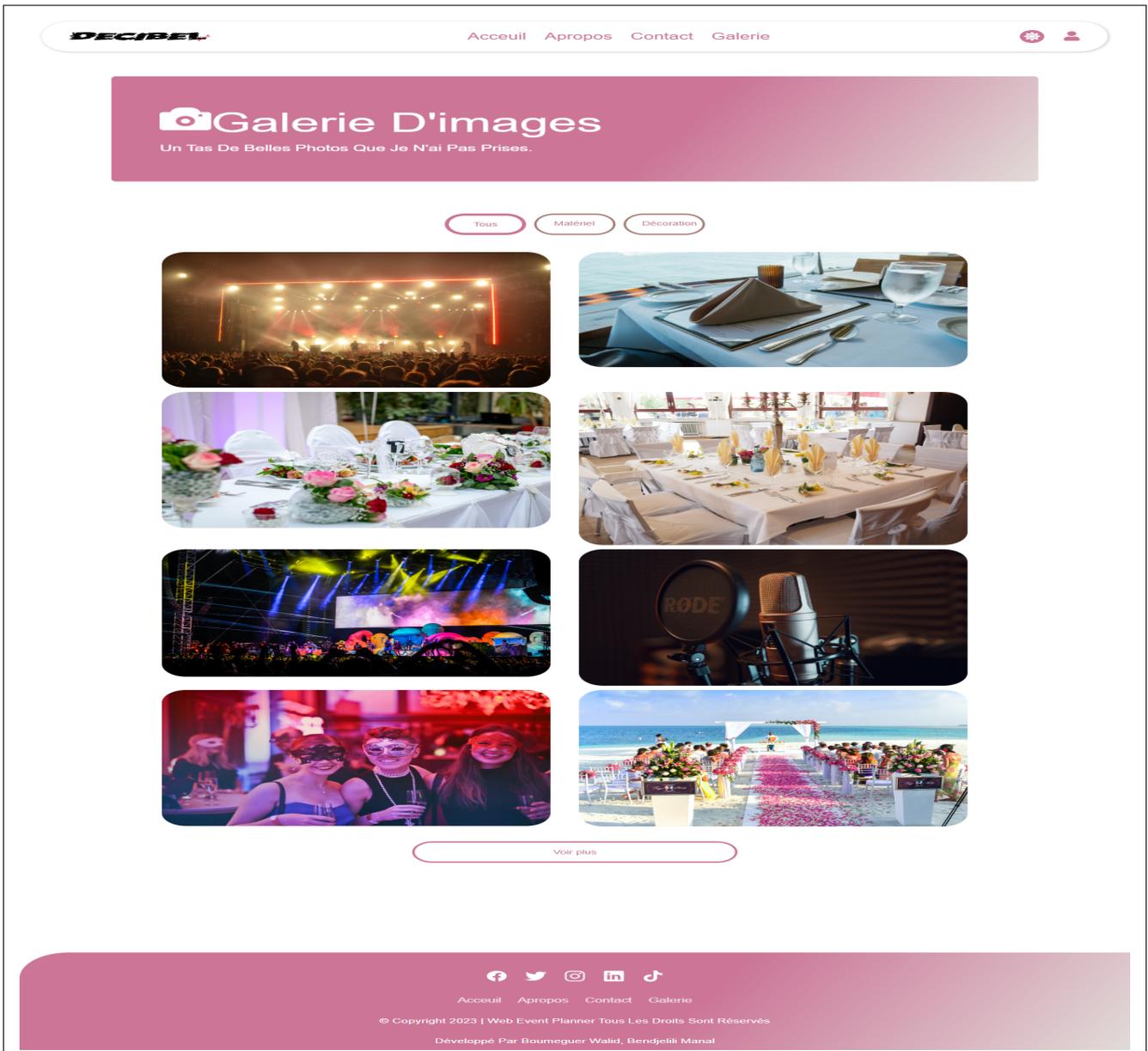


FIGURE 4.44 – Interface « Galerie »

Interface « Accueil admin »



FIGURE 4.45 – Interface « Accueil admin »

Interface « Matériel de sonorisation »

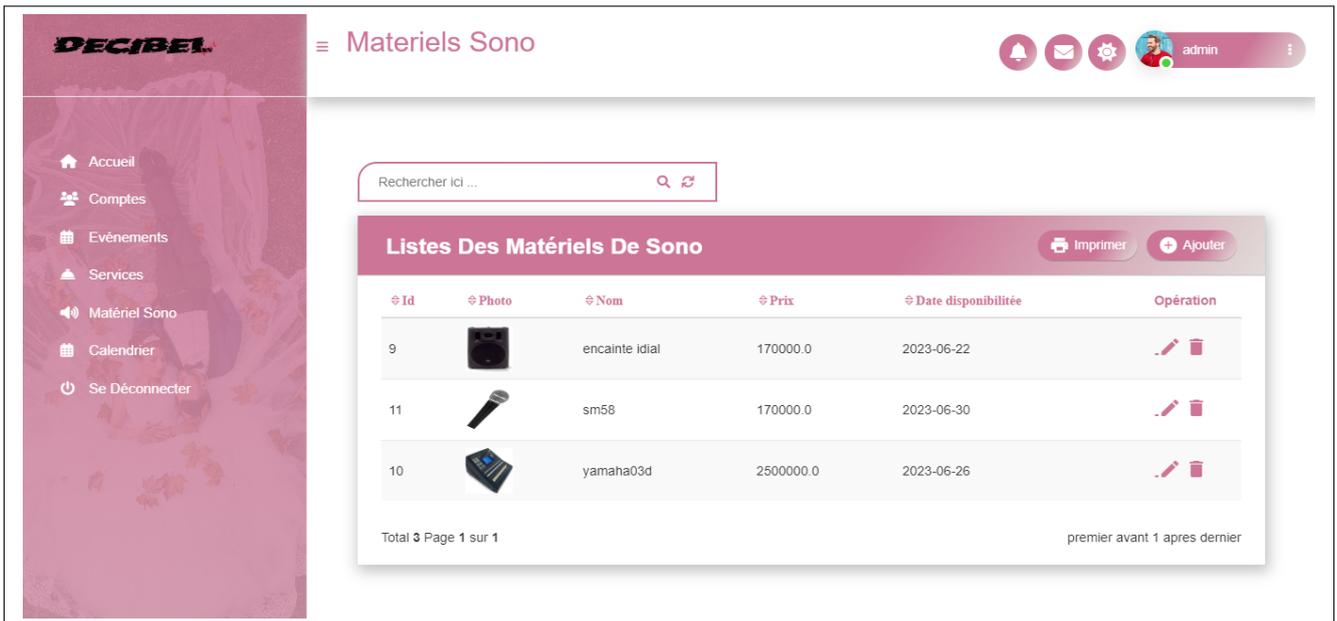


FIGURE 4.46 – Interface « Matériel de sonorisation »

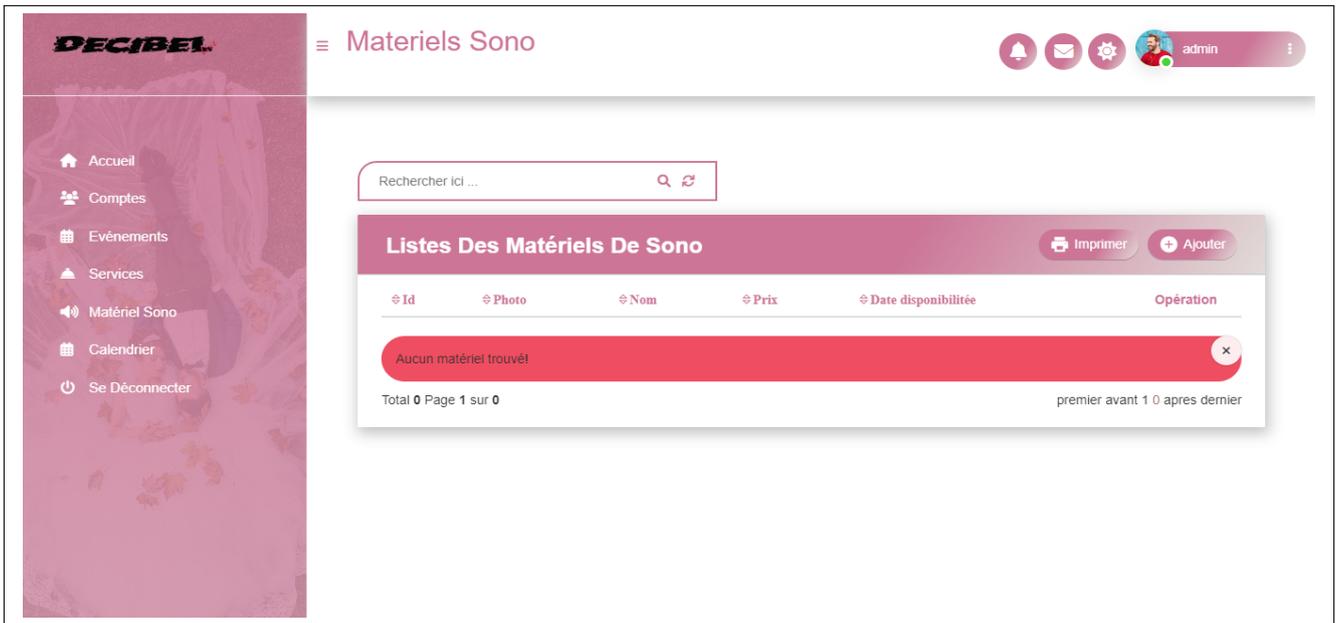


FIGURE 4.47 – Interface « Message d’erreur sur l’absence d’un matériel ou table vide »

Interface « Marquer un évènement sur le calendrier »

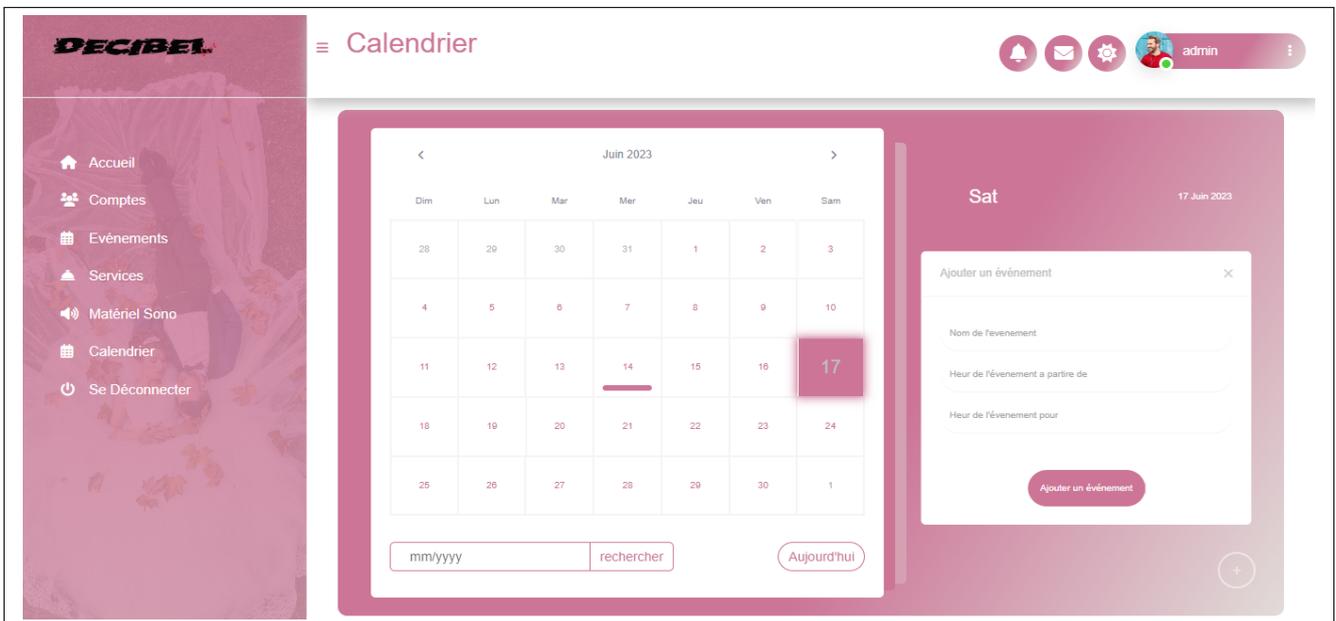


FIGURE 4.48 – Interface « Marquer un évènement sur le calendrier »

Interface « Déconnexion »

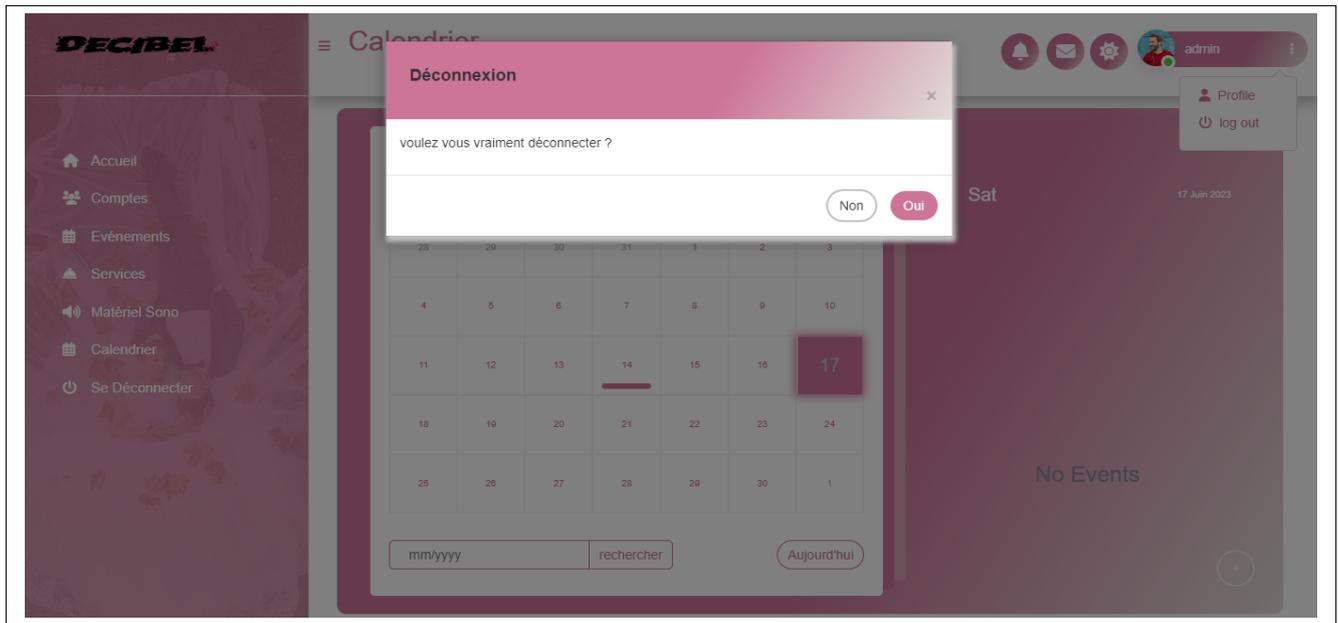


FIGURE 4.49 – Interface « Déconnexion »

Résumé

L'objectif principal de ce projet est de réaliser une application web service dédié à l'organisation des événements en composant des services web qui offre de nombreux avantages en terme d'efficacité dans le but d'améliorer la gestion des événements et rest compétitif dans un environnement en évolution constante.

Pour mettre en œuvre notre solution, nous avons utilisé une méthode agile appelé SCRUM qui se base sur UML comme langage de modélisation et le pattern de conception MVC.

Cette conception est mise en œuvre sous l'environnement de développement intégré (STS),Springboot,bootstrap comme framework et tomcat comme serveur et un système de gestion de la base données PhpMyAdmin .

Mots clés :service Web,DECIBEL EVENTS

Abstract

The main objective of this project is to develop a web service application dedicated to event organization,by composing web services that offer numerous advantages in terms of efficiency, aiming to improve event management and stay competitive in an ever-evolving environment.

To implement our solution, we adopted an agile method called SCRUM, which is based on UML as a modeling language and the MVC design pattern.

This design is implemented using the integrated development environment (STS), Springboot, and Bootstrap as the framework, with Tomcat as the server, and a PhpMyAdmin database management system.

Key words :Web services,DECIBEL EVENTS