

Rapport de stage

REALISATION D’UN SITE WEB DU FAMILY CALENDAR

Effectué à fst Marrakech

**Encadré par :**

**M.Bencharef Omar**

**Réalisé** **par :**

Kaabouch Mohamed

Ait lhaj abdelilah

**Année Universitaire 2023-2022**

**Dédicace**

Je dédie cet acte humble et notre profonde gratitude à mes parents pour l'éducation qu'ils m'ont donnée par tous les moyens et au prix de tous les sacrifices qu'ils ont consentis pour nous, pour le sens du devoir qu'ils nous ont inculqué dès l'enfance.

Sans oublier nos cher(e)s ami(e)s, et professeurs que sans eux en n’aurait pas trouvé le sens de l'orientation pour arriver jusqu’ici, merci encore pour leur aide, leur temps, leurs encouragements, leur assistance et soutien.

Une dévotion particulière également à notre chère famille, pour leur dévouement, leur patience et leurs prières.

Nous dédions aussi ce travail à :

Nos encadrants pour leur patience, leur encouragement, et leurs conseils précieux. Nos professeurs qui doivent voir dans ce travail la fierté d'un savoir bien acquis.

# REMERCIEMENTS

Avant tout développement sur cette expérience professionnelle, il apparaît opportun de commencer ce rapport de stage par des remerciements, à ceux qui m’ont beaucoup appris au cours de ce stage, et à ceux qui ont eu la gentillesse de faire de ce stage un moment très profitable

Au début, nous tenons à adresser nos remerciements les plus sincères à tout le corps professionnel et administratif de la Faculté des Sciences et Techniques de Marrakech .

Nous remercions Dieu Tout-Puissant de m'avoir donné la force courage et patience pour développer cet humble travail aussi Nous remercions dans un premier temps, toute l’équipe pédagogique de la faculté des sciences et techniques et les intervenants professionnels responsables de la filier IRSI qui ont contribué à nous donner une formation solide tout au long de mon étude.

Nous remercions tout particulièrement mon professeur et notre encadrant Pr. **Bencharef Omar**, pour tous ses conseils, le soutien dont il nous a fait bénéficier, de ses encouragements, les aides précieuses qu’il n'a pas cessés de je apporte tout au long de la période de ce projet et pour ses efforts afin d’assurer le bon déroulement de project .

Enfin Nous remercions toutes les personnes qui ont contribuées de près ou de loin à la réalisation de ce travail, ainsi qu’au bon déroulement du project, et dont les noms ne figurent pas dans ce document

Table des matières

[REMERCIEMENTS 3](#_Toc142928711)

[1 INTRODUCTION GENERAL 7](#_Toc142928712)

[CONTEXTE GENERALE DU PROJET 8](#_Toc142928713)

[2 CHAPITRE I :CONTEXTE GENERALE DU PROJET 9](#_Toc142928714)

[*2.1* *PRESENTATION DU LIEU DE STAGE* 9](#_Toc142928715)

[*2.1.1* *Introduction :* 9](#_Toc142928716)

[*2.1.2* *Organigramme de fst* 9](#_Toc142928717)

[2.2 Cadre et objectif du projet 9](#_Toc142928718)

[2.2.1 CRITIQUE DE L’EXISTANT 9](#_Toc142928719)

[2.2.2 SOLUTION PROPOSEE 9](#_Toc142928720)

[2.2.3 DIAGRAMME DE GANTT 10](#_Toc142928721)

[2.3 Analyse de l’existant 10](#_Toc142928722)

[2.4 Cycle de développement 11](#_Toc142928723)

[2.4.1 Modèle en V : 11](#_Toc142928724)

[3 CHAPITRE II :ANALYSE ET CONCEPTION 12](#_Toc142928725)

[3.1 Analyse des besoins FONCTIONNELS 12](#_Toc142928726)

[3.1.1 LES ACTEURS DU SYSTEME 12](#_Toc142928727)

[3.1.2 DESCRIPTION DES ROLES DE VISITEUR : 12](#_Toc142928728)

[3.1.3 DESCRIPTION DES ROLES DE UTILISATEUR 13](#_Toc142928729)

[3.1.4 DESCRIPTION DES ROLES DE L’ADMINISTRATEUR 13](#_Toc142928730)

[4 Modélisation ET CONCEPTION 13](#_Toc142928731)

[4.1 DIAGRAMME DE CLASSES 13](#_Toc142928732)

[4.2 MODELE LOGIQUE DE DONNEES (MLD) 13](#_Toc142928733)

[4.3 IDENTIFICATION DES CAS D’UTILISATION 14](#_Toc142928734)

[4.3.1 Partie de cas d’utilisation de visiteur(client) : 14](#_Toc142928735)

[4.4 LES DIAGRAMME DE SEQUENCES 14](#_Toc142928736)

[4.4.1 Diagramme de séquences : Authentification 14](#_Toc142928737)

[4.4.2 Diagrammes de séquences : Ajouter 14](#_Toc142928738)

[4.4.3 Diagrammes de séquences : Modifier 14](#_Toc142928739)

[4.4.4 Diagrammes de séquences : gestion infraction : 14](#_Toc142928740)

[5 Réalisations 14](#_Toc142928741)

[5.1 Outils de développements 14](#_Toc142928742)

[5.1.1 Langage et outils : 14](#_Toc142928743)

[5.1.2 Php 14](#_Toc142928744)

[5.1.3 Xampp 14](#_Toc142928745)

[5.2 Présentation de l’application 14](#_Toc142928746)

[5.2.1 Architecture application 14](#_Toc142928747)

[5.2.2 Captures de l’application : 14](#_Toc142928748)

[6 CONCLUSION 14](#_Toc142928749)

# INTRODUCTION GENERAL

La gestion efficace du temps et la coordination des événements au sein des familles constituent des défis constants dans nos vies modernes. Dans cette ère de dynamisme et d'agendas chargés, la nécessité de maintenir un équilibre entre les horaires individuels et les activités communes est devenue plus cruciale que jamais. C'est dans ce contexte que le projet "Family Calendar" a vu le jour.

Le projet "Family Calendar" incarne une réponse innovante à ces défis. Il vise à simplifier et à faciliter la planification et la gestion des événements familiaux en créant un espace interactif où chaque membre de la famille peut participer activement. En permettant aux membres de partager leurs événements individuels tout en ayant une vue d'ensemble des activités de la famille, ce projet vise à renforcer la communication interne et à renouveler la manière dont les familles organisent leurs vies.

Cette introduction offre un aperçu de la motivation sous-jacente à la création du projet "Family Calendar". Les sections suivantes de ce rapport exploreront en détail les fonctionnalités, la mise en œuvre, les résultats et les avantages apportés par cette initiative, tout en mettant en lumière les contributions uniques qu'elle apporte à la gestion harmonieuse du temps au sein des familles modernes.

En somme, ce rapport explore comment "Family Calendar" propose une solution pratique pour transformer la manière dont les familles abordent la coordination des événements, en favorisant la collaboration, la communication et la création de souvenirs partagés.

**CHAPITRE I :**

# CONTEXTE GENERALE DU PROJET

# CHAPITRE I :CONTEXTE GENERALE DU PROJET

## PRESENTATION DU LIEU DE STAGE

### Introduction :

### Organigramme de fst

## Cadre et objectif du projet

### CRITIQUE DE L’EXISTANT

La planification et la coordination des événements au sein des familles ont longtemps été une tâche complexe, souvent associée à des défis qui découlent de la diversité des horaires individuels et des activités collectives. Avant l'avènement des solutions technologiques dédiées, les familles dépendaient principalement des calendriers physiques, des listes de tâches manuscrites et des rappels oraux pour gérer leurs emplois du temps. Cependant, cette approche comportait plusieurs lacunes, dont les suivantes :

Fragmentation de l'Information : Les membres de la famille disposaient de leurs propres agendas, ce qui rendait difficile la consolidation et la communication des événements à tous les membres. Les risques d'oubli ou de confusion étaient élevés.

Manque de Communication en Temps Réel : Les changements de dernière minute ou les ajouts d'événements n'étaient pas communiqués en temps réel à tous les membres de la famille, entraînant des conflits d'horaires et des désorganisations.

Difficulté à Coordonner les Activités Collectives : La coordination d'événements impliquant plusieurs membres nécessitait souvent des échanges répétés d'appels téléphoniques ou de messages, créant une charge de travail supplémentaire.

Gestion Limitée des Rappels : La gestion des rappels était tributaire de la mémoire individuelle ou de notes manuscrites, pouvant conduire à des oublis et à des rendez-vous manqués.

### SOLUTION PROPOSEE

Le projet "Family Calendar" offre une solution complète et innovante pour améliorer la planification et la coordination des événements au sein des familles. En adoptant une approche technologique centrée sur l'utilisateur, cette solution vise à simplifier la gestion des horaires familiaux et à favoriser une communication transparente. Voici un aperçu des caractéristiques clés de la solution :

Calendrier Partagé et Interactif : Le cœur de la solution réside dans la création d'un calendrier partagé en ligne, accessible à tous les membres de la famille. Chaque membre peut ajouter des événements, visualiser ceux des autres et même les catégoriser en fonction de leur nature (travail, école, loisirs, etc.). Cette visualisation commune garantit que tous les membres ont une vue d'ensemble des activités de la famille.

Notifications et Rappels : La solution intègre des fonctionnalités de rappel automatisé pour les événements à venir. Les utilisateurs peuvent choisir d'être notifiés par e-mail ou notifications push pour éviter les oublis et garantir une participation maximale aux événements.

Gestion des Membres de la Famille : La solution permet également d'ajouter et de gérer les membres de la famille. Chaque membre a son propre profil, et l'accès au calendrier peut être personnalisé en fonction des préférences individuelles.

Facilité d'Utilisation : L'interface conviviale de la plateforme garantit une expérience utilisateur intuitive. Les événements peuvent être ajoutés en quelques clics, et les options de modification et de suppression sont facilement accessibles.

Partage Sécurisé : La solution priorise la sécurité et la confidentialité des données familiales. Les événements ne sont accessibles qu'aux membres de la famille autorisés, créant ainsi un espace sécurisé pour partager des informations sensibles.

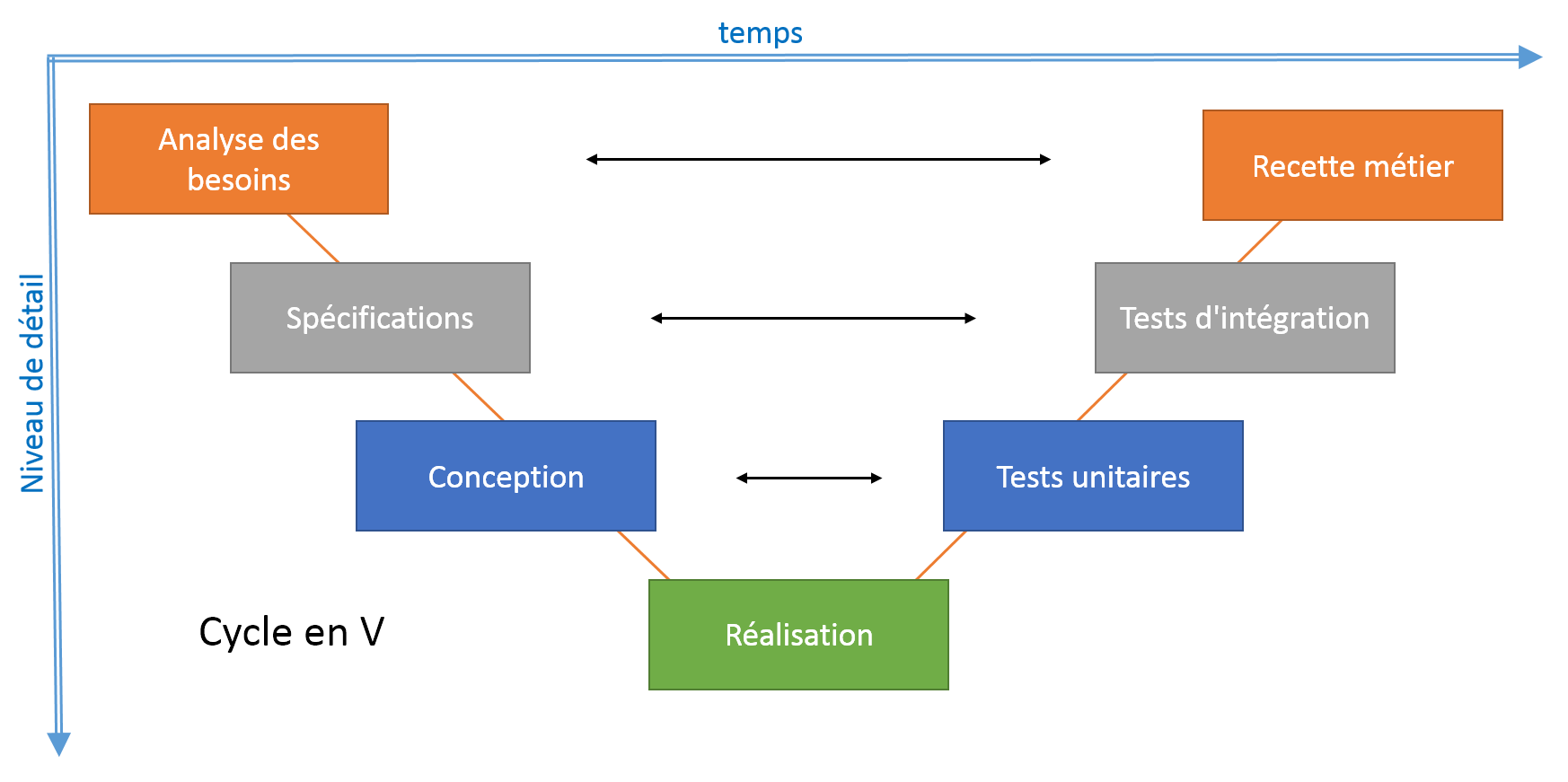
### DIAGRAMME DE GANTT

## Analyse de l’existant

Avant l'avènement du projet "Family Calendar", la gestion des événements au sein des familles était souvent entravée par des méthodes traditionnelles de coordination. Les calendriers en papier et les listes de tâches manuscrites étaient les principaux outils utilisés pour planifier les activités familiales. Cependant, ces approches présentaient des lacunes majeures, telles que la fragmentation des informations, les risques d'oubli et la difficulté de communication en temps réel. Les membres de la famille étaient souvent déconnectés des horaires des autres, ce qui rendait la coordination des activités complexes et sujette à des erreurs. L'analyse de cette situation a mis en évidence la nécessité d'une solution plus efficace. C'est ainsi que le projet "Family Calendar" a été conçu pour offrir une plateforme numérique interactive qui centralise les événements familiaux, favorisant une meilleure communication et une organisation harmonieuse.

## Cycle de développement

### Modèle en V :

Le modèle du cycle en V est un modèle conceptuel de gestion de projet imaginé suite au problème de réactivité du modèle en cascade. Il permet en cas d'anomalie, de limiter un retour aux étapes précédentes. Les phases de la partie montante doivent renvoyer de l'information sur les phases en vis-à-vis lorsque des défauts sont détectés, afin d'améliorer le logiciel.

C’est le modèle avec lequel nous allons travailler, il corrige les problèmes qui peuvent résulter des méthodes classiques, et se compose de deux branches :

• ***La branche de gauche :*** correspond à une cascade ; chaque phase prépare les jeux de tests pour vérifier le produit livrable de cette phase. Il s'agit de ne pas proposer des éléments impossibles à vérifier une fois le logiciel réalisé :

o L'architecture développée, lors de la conception préliminaire, doit être accompagnée d’une description des tests d'intégration (plan et Jeu de tests).

o Les composants développés, lors du codage, doivent être accompagnés de la description des tests unitaires (plan et jeu de tests).

• ***La branche de droite*** : correspond à la réalisation effective des tests; chaque phase de tests s'appuie sur le résultat de la phase correspondante dans la branche de gauche.

**CHAPITRE II :**

# ANALYSE ET CONCEPTION

# CHAPITRE II :ANALYSE ET CONCEPTION

## Analyse des besoins FONCTIONNELS

### LES ACTEURS DU SYSTEME

Un acteur est une construction qui représente un rôle joué par un utilisateur humain ou un autre système qui interagit directement avec le système étudié. Un acteur participe à au moins un cas d’utilisation

Nous avons identifié deux acteurs principaux, le tableau ci-dessous présente les acteurs et leurs rôles :

|  |  |
| --- | --- |
| ***Acteurs*** | ***Rôles*** |
| Visiteur | Cree un message |
| User (non responsable de famile) | Gestion des évènements  Voir votre membre de famile  Accede a les mombres de famile |
| User (responsable de famile) | Gestion de ton famile  Gestion des évènements  Accede a les mombres de famile |
| Administrateur | Gestion des évènements  Gestion des familes .  Gestion des contacts.  Gestion des categorie .  Gestion des informations de site. |

### DESCRIPTION DES ROLES DE VISITEUR :

* **La saisie** : saisir les informations son nom, téléphone, email, sujet et le message puis valider le tout pour envoyer message.

### DESCRIPTION DES ROLES DE UTILISATEUR

✓ **S’authentifier :** utilisateur doit s’authentifier pour réaliser les différentes tâches.

✓ **La saisie** : saisir les informations évènements sur son famile si il a le droit de faire le retrait puis valider le tout.

✓ **La recherche** : chercher les informations sur la évènements par clé name

✓ **Modification** : modifier les informations de la évènements soit les attributs ou les autre information de famile si il est le responsable de famile .

✓ **Consultation** : consulter les informations du évènements, informations sur les famile, informations sur categorie .

✓ **suppression** : supprimerles données de les évènements et famile .

### DESCRIPTION DES ROLES DE L’ADMINISTRATEUR

✓ **S’authentifier :** l’admin doit s’authentifier pour réaliser les différentes tâches.

✓ **La saisie** : saisir les informations sur la catégorie puis valider le tout.

✓ **La recherche** : chercher les informations sur la évènements par clé name , ou name de famile dans la partie famile .

✓ **Modification** : modifier les informations de la évènements soit les attributs ou les autre information de familes et aussi catégorie .

✓ **Consultation** : consulter les informations du évènements, informations sur les catégorie, informations sur familes , et les informations sur contacts

✓ **suppression** : supprimerles données de les évènements a savoir famile ou les cetegorie et aussi contact .

**CHAPITRE 3 :**

**MODELISATION ET CONCEPTION**

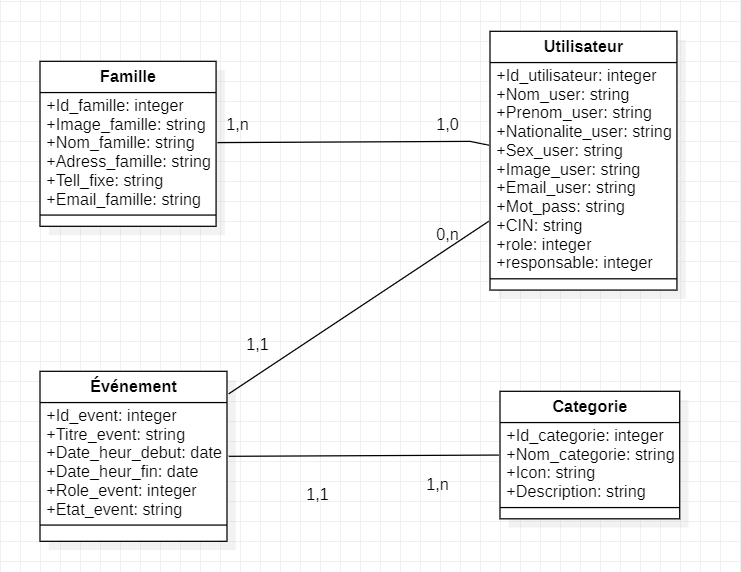
# Modélisation ET CONCEPTION

## DIAGRAMME DE CLASSES

Le diagramme de classes est le point central dans le développement orienté objet. Un diagramme de classes représente la ***structure statique*** du système sous forme de classes et de relations entre classes.

Les classes constituent la base pour la ***génération de code*** et pour la ***génération des schémas de bases de données****.*

Le diagramme de classes de notre application web est présenté comme suit :



## MODELE LOGIQUE DE DONNEES (MLD)

Famile( Id\_famile, image\_famile,nom\_famile, Adress\_famile, Tell\_fixe , Email\_famile);

Utilisateur( Id\_utilisateur, Nom\_user, Prenom\_user , Nationalite\_user, sex\_user , image\_user, Email\_user , mot\_pass , CIN ,role , responsable , #Id\_famile);

Evenement(ID\_event , Titre\_event , Date\_heur\_event , Date\_heur\_fin , role\_event , Etat\_event , #Id\_utilisateur , #Id\_categorie) ;

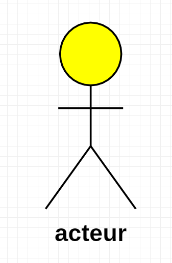
Categorie( Id\_categorie , nom\_categorie , Icon , description ) ;

## IDENTIFICATION DES CAS D’UTILISATION

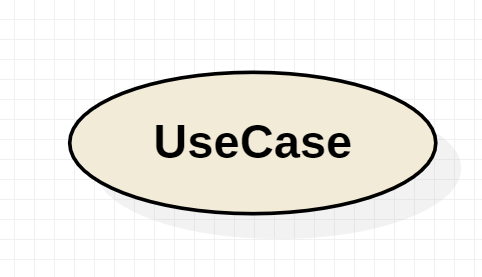
Définition :

les diagrammes de cas d'utilisation modélisent le comportement d'un système et permettent de capturer les exigences du système. Les diagrammes de cas d'utilisation décrivent les fonctions générales et la portée d'un système.

Le diagramme des cas d’utilisation se compose de trois éléments principaux suivants:



**Un Acteur :** c’est l’idéalisation d’un rôle joué par une personne externe, un processus ou une chose qui interagit avec un système. Il se représente par un petit bonhomme avec son nom inscrit dessous



**Un cas d’utilisation :** c’est une unité cohérente représentant une fonctionnalité visible de l’extérieur. Il réalise un service de bout en bout avec, un déclenchement, un déroulement et une fin pour l’acteur qui l’initie.

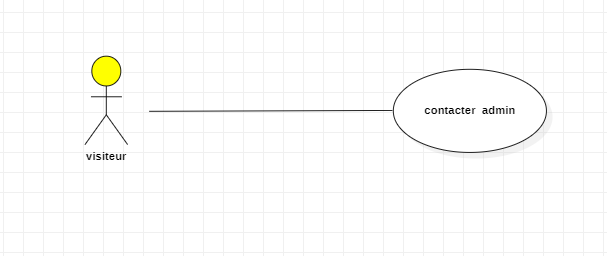
Un cas d’utilisation modélise donc un service rendu par le système, sans imposer le mode de réalisation de ce service. Il est représenté par une ellipse contenant le nom du cas (un verbe à l’infinitif), et optionnellement, au-dessus du nom, un stéréotype

**Les relations :** Les relations de cas d'utilisation en UML sont essentielles pour modéliser les interactions entre acteurs et cas d'utilisation. Les trois types courants sont **association**, **inclusion et extension**. On ajoute également la généralisation, l'utilisation, et la dépendance. Ces relations permettent de définir les dépendances, les scénarios d'utilisation et les héritages, aidant ainsi à communiquer les exigences fonctionnelles et à planifier le développement du système.

* **Association** (relation simple) : Acteur interagit avec le cas d'utilisation.
* **Inclusion** (include) : Cas d'utilisation principal inclut un autre cas d'utilisation plus petit.
* **Extension** (extend) : Cas d'utilisation source peut étendre le comportement d'un autre cas d'utilisation en fonction de certaines conditions.

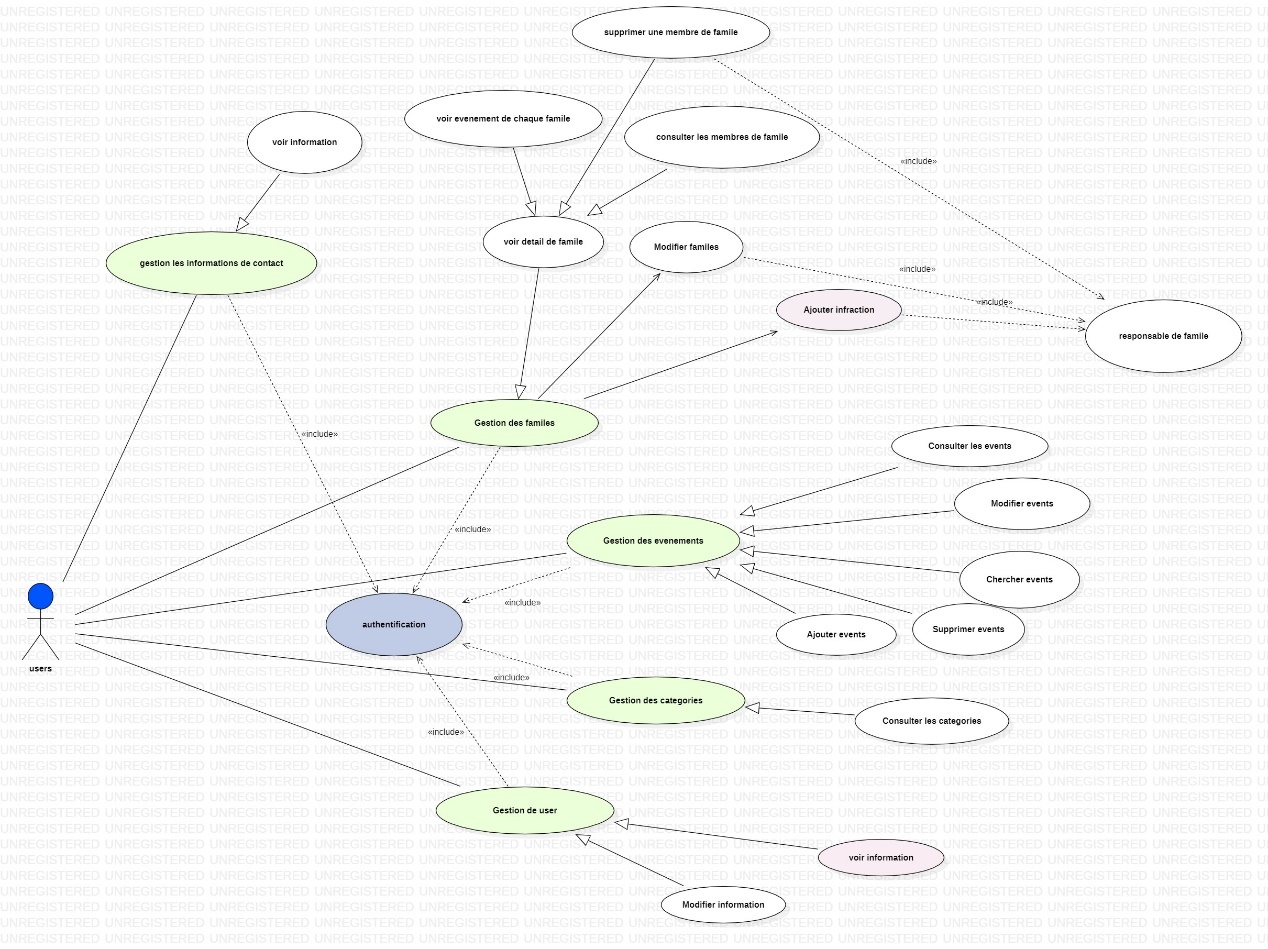
### Partie de cas d’utilisation de visiteur(client) :

Dans le cadre d'une utilisation typique par un visiteur de site, la fonctionnalité de messagerie aux administrateurs offre une voie de communication essentielle. Les visiteurs ont la possibilité de contacter les administrateurs du site en fournissant leur nom, leur adresse e-mail, le sujet de leur message et le contenu de leur message. Cette fonctionnalité assure une interaction directe entre les utilisateurs et les administrateurs, permettant aux visiteurs de soumettre leurs préoccupations, questions ou commentaires de manière efficace. Grâce à cette fonctionnalité, les administrateurs peuvent rapidement recevoir les messages des visiteurs, prendre des mesures appropriées et fournir une réponse personnalisée, renforçant ainsi l'engagement et la satisfaction des utilisateurs.



### Partie de cas d’utilisation d’utilisateur :

Les utilisateurs de la plateforme "Family Calendar" bénéficient d'un ensemble de fonctionnalités conçues pour simplifier la gestion de leurs activités familiales. Ils peuvent créer, modifier et supprimer leurs propres événements, contribuant ainsi à la planification harmonieuse au sein de la famille. Si l'utilisateur détient des privilèges d'administration, il peut également créer des familles, inviter des membres et gérer les demandes d'adhésion. La possibilité de consulter les événements des autres membres de la famille favorise une coordination sans faille, tandis que la fonction de notification garantit qu'aucune demande d'adhésion ne passe inaperçue. Grâce à une interface conviviale, les utilisateurs peuvent personnaliser leur expérience en fonction de leurs besoins, facilitant ainsi une communication transparente et une organisation efficace des événements familiaux.



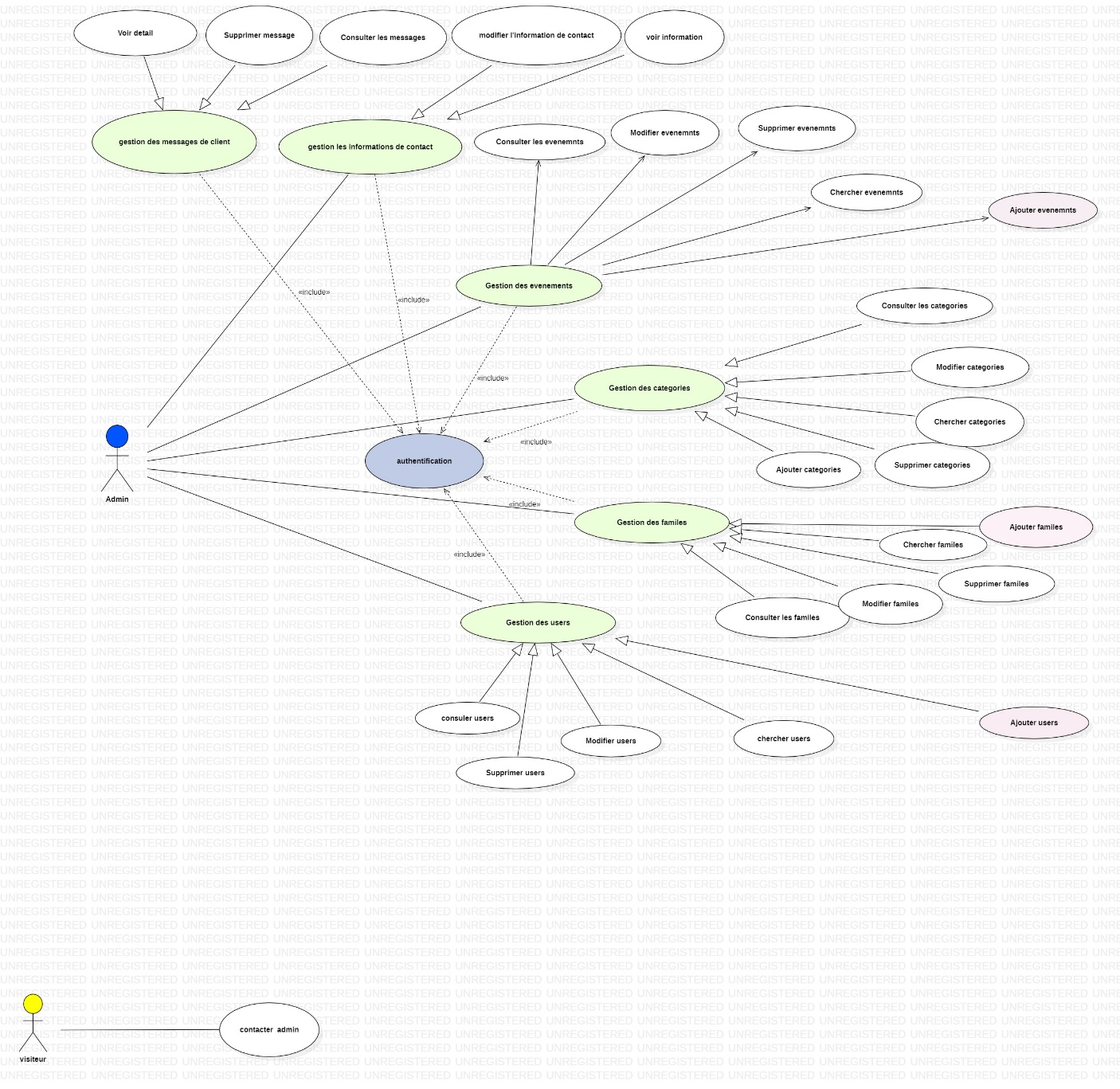
### Cas d’utilisation d’administrateur

Au cœur de la gestion opérationnelle de la plateforme "Family Calendar", l'administrateur joue un rôle essentiel dans la supervision et l'optimisation de divers aspects. Responsable de la gestion des informations du site, il veille à ce que le contenu soit à jour et en accord avec l'identité visuelle de la plateforme. La gestion des utilisateurs est également entre ses mains, impliquant la création, la modification et la désactivation de comptes pour maintenir une base de données utilisateur fiable.

La gestion des catégories d'événements permet à l'administrateur de créer une structure cohérente, tandis que la supervision des familles assure un environnement sûr et approprié pour les utilisateurs. En ayant un accès global aux événements et aux données, l'administrateur peut résoudre les conflits d'horaires et garantir la qualité des informations affichées.

En tant que point de contact principal pour le support utilisateur, l'administrateur répond aux questions et résout les problèmes techniques, assurant une expérience sans heurts. La sécurité et la maintenance du site sont également de sa responsabilité, avec une vigilance constante pour prévenir les problèmes potentiels et garantir la disponibilité continue de la plateforme.

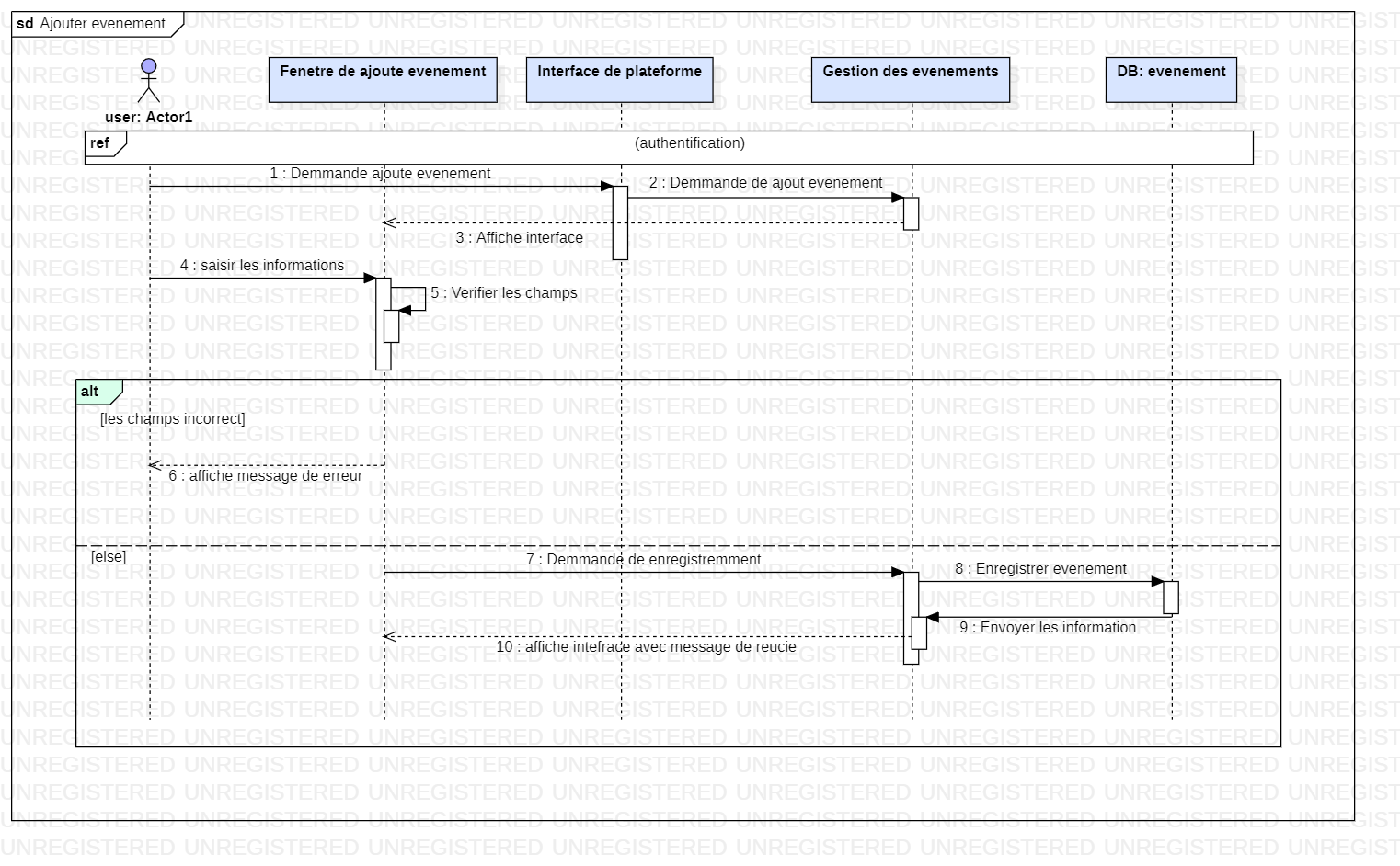
Dans l'ensemble, l'administrateur exerce une influence substantielle sur la fluidité des opérations et la satisfaction des utilisateurs, jouant un rôle pivot dans la réussite globale de la plateforme "Family Caleendar ‘’



## LES DIAGRAMME DE SEQUENCES

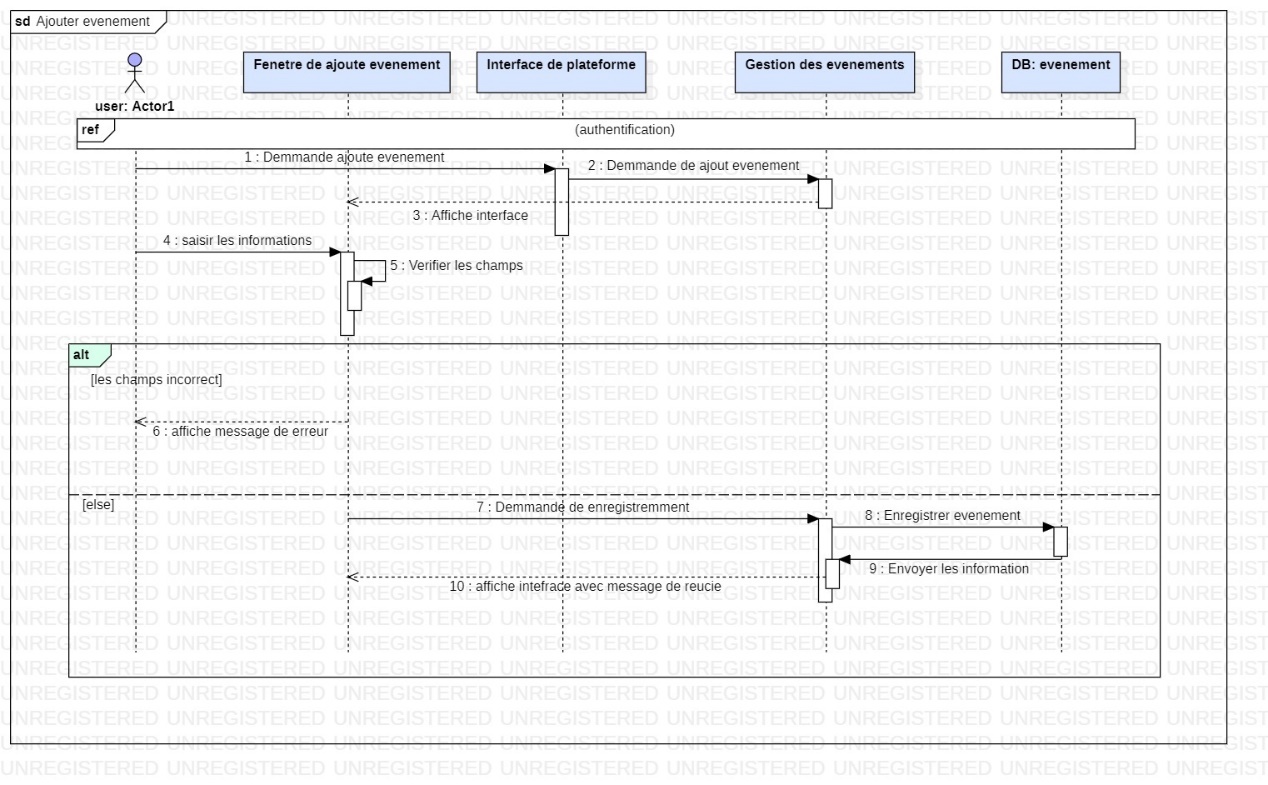
### Diagramme de séquences : Authentification

Le diagramme de séquence pour l'authentification est un outil graphique de modélisation utilisé dans le cadre du langage de modélisation UML (Unified Modeling Language). Il permet de représenter de manière séquentielle les interactions entre les acteurs et les objets lors du processus d'authentification d'un utilisateur sur un système.

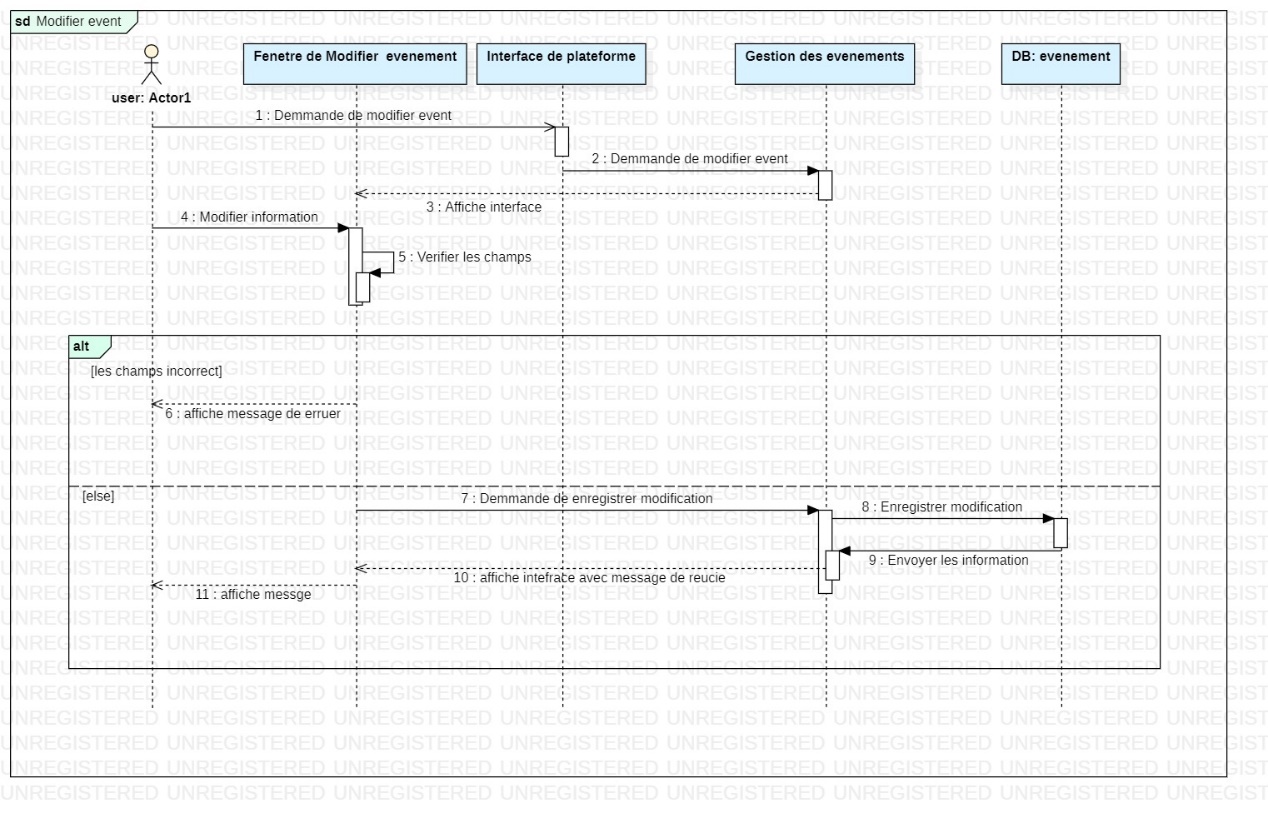


### Diagrammes de séquences : Ajouter

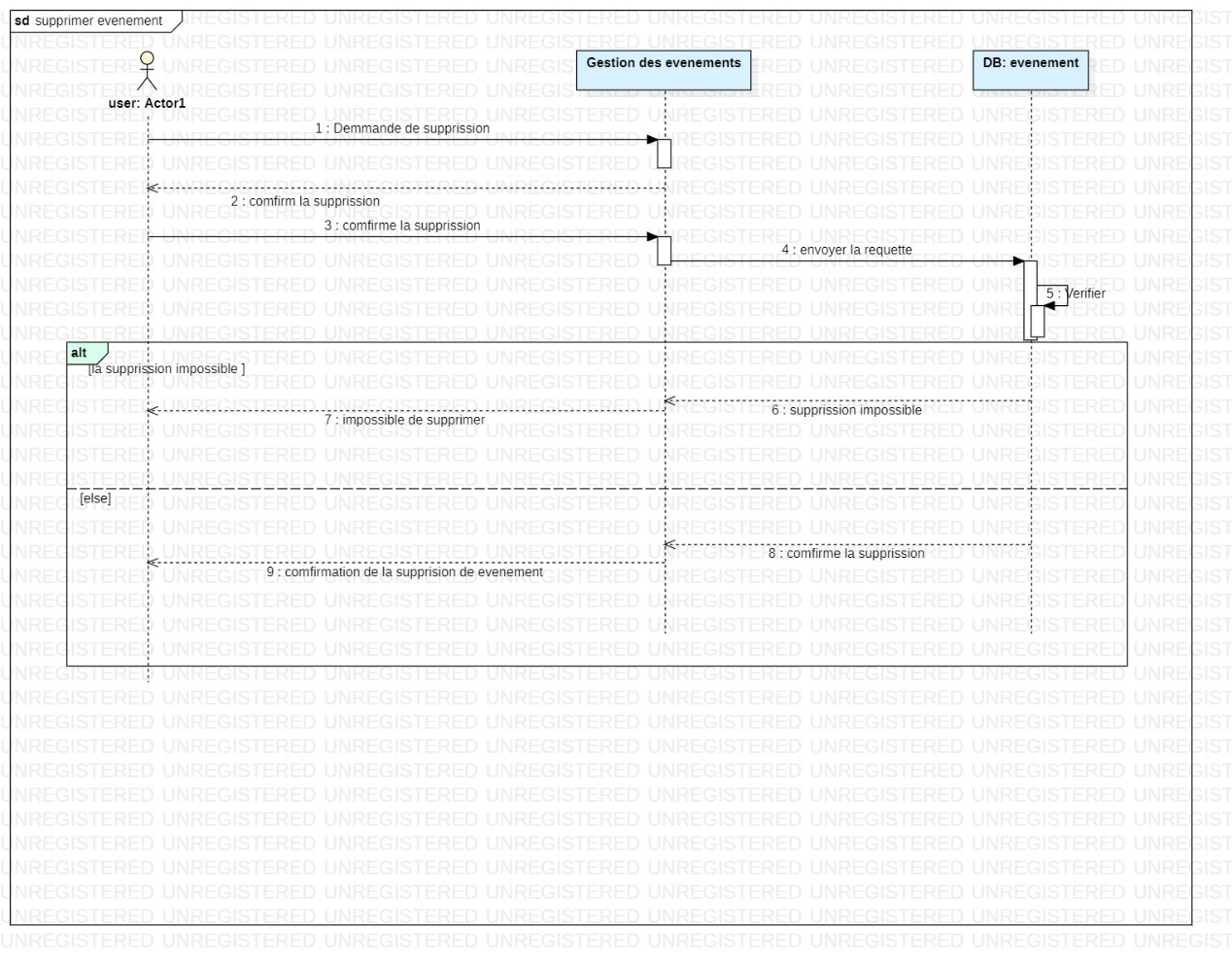
Lorsqu'il choisit d'ajouter une nouvelle évènement, le système affiche un formulaire pour collecter les détails pertinents. Une fois que l'utilisateur remplit le formulaire, le système enregistre les informations de la nouvelle évènement dans la base de données des évènements, associant ainsi la évènement à l'utilisateur. Le système informe ensuite l'utilisateur que l'ajout de évènement s'est déroulé avec succès.



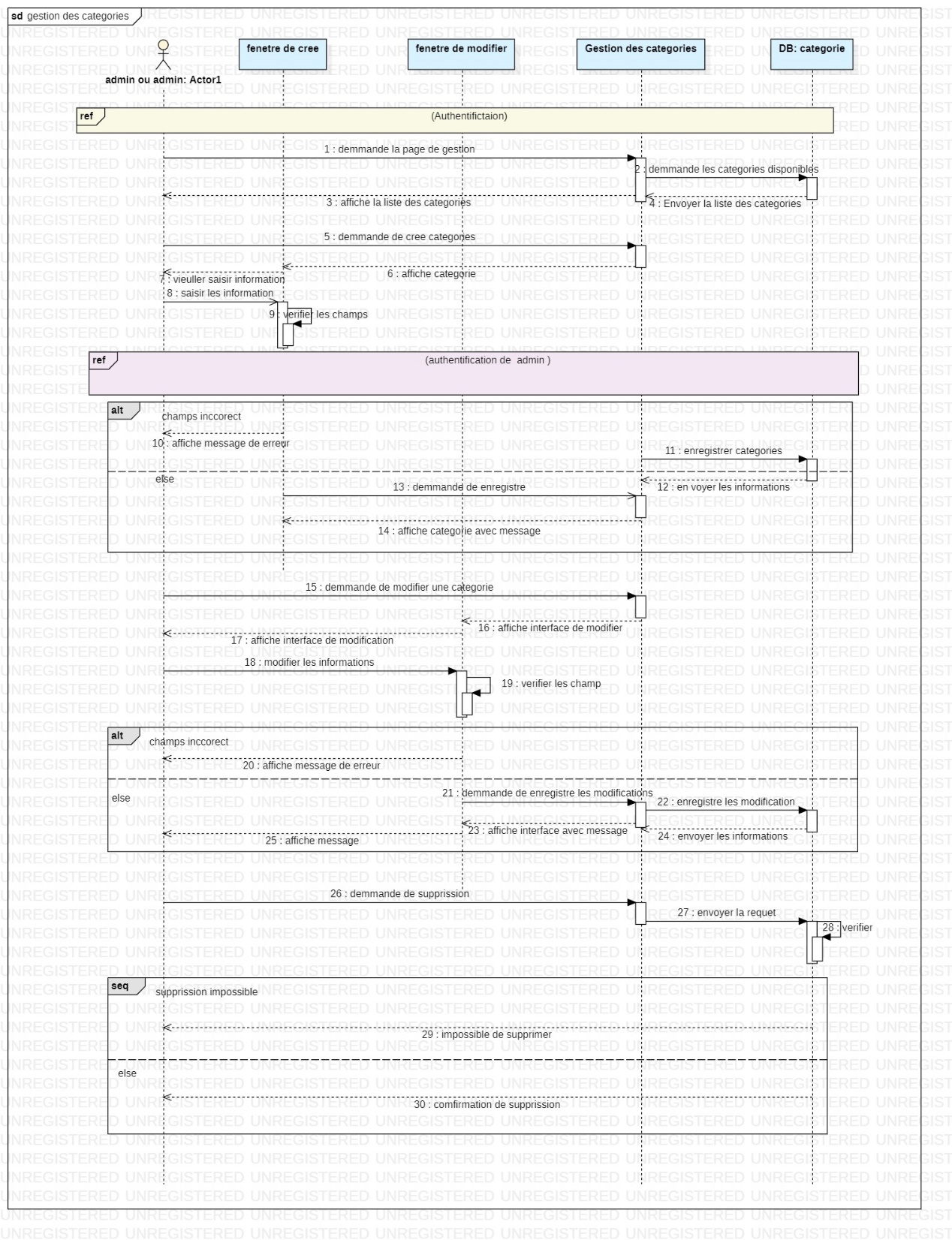
### Diagrammes de séquences : Modifier



### Diagrammes de séquences : suppression:



### Diagrammes de séquences :gestion (catégorie par exemple )



**CHAPITRE 4 :**

# Réalisations

# Réalisations

## Outils de développements

### Langage et outils :

✓ **UML : Unified Modeling Language**

Un langage de modélisation graphique à base de pictogrammes. Il est apparu dans le monde du génie logiciel, dans le cadre de la « conception orientée objet ». Couramment utilisé dans les projets logiciels, il peut être appliqué à toutes sortes de systèmes ne se limitant pas au domaine informatique. C’est l'accomplissement de la fusion de précédents langages de modélisation objet : Booch, OMT, OOSE. Principalement issu des travaux de Grady Booch, James Rumbaugh et Ivar Jacobson, UML est à présent un standard défini par l'Object Management Group (OMG)



✔ **HTML 5 : HyperText Markup Language**.

C’est un langage de balisage qui permet d’écrire de l’hypertexte, d’où son nom. HTML permet également de structurer sémantiquement et de mettre en forme le contenu des pages, d’inclure des ressources multimédias dont des images, des formulaires de saisie, et des éléments programmables tels que des applets. Il permet de créer des documents interopérables avec des équipements très variés de manière conforme aux exigences de l’accessibilité du web. Il est souvent utilisé conjointement avec des langages de programmation (JavaScript) et des formats de présentation (feuilles de style en cascade). HTML est initialement dérivé du Standard Generalized Markup Language (SGML).



HTML 5 : Est La dernière version, encore assez peu répandue, elle fait beaucoup parler d'elle car elle apporte de nombreuses améliorations .

✓**CSS 3 : Cascading Style Sheets.**

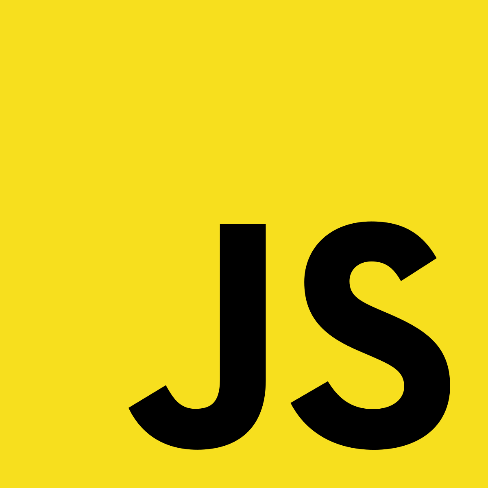
Un langage informatique qui sert à décrire la présentation des documents HTML et XML. Les standards définissant CSS sont publiés par le World Wide Web Consortium (W3C). Introduit au milieu des années 1990, CSS devient couramment utilisé dans la conception sites web et bien pris en charge par les navigateurs web dans les années 2000



**CSS 3** : Est la dernière version, qui apporte des fonctionnalités particulièrement attendues comme les bordures arrondies, les dégradés, les ombres… etc.

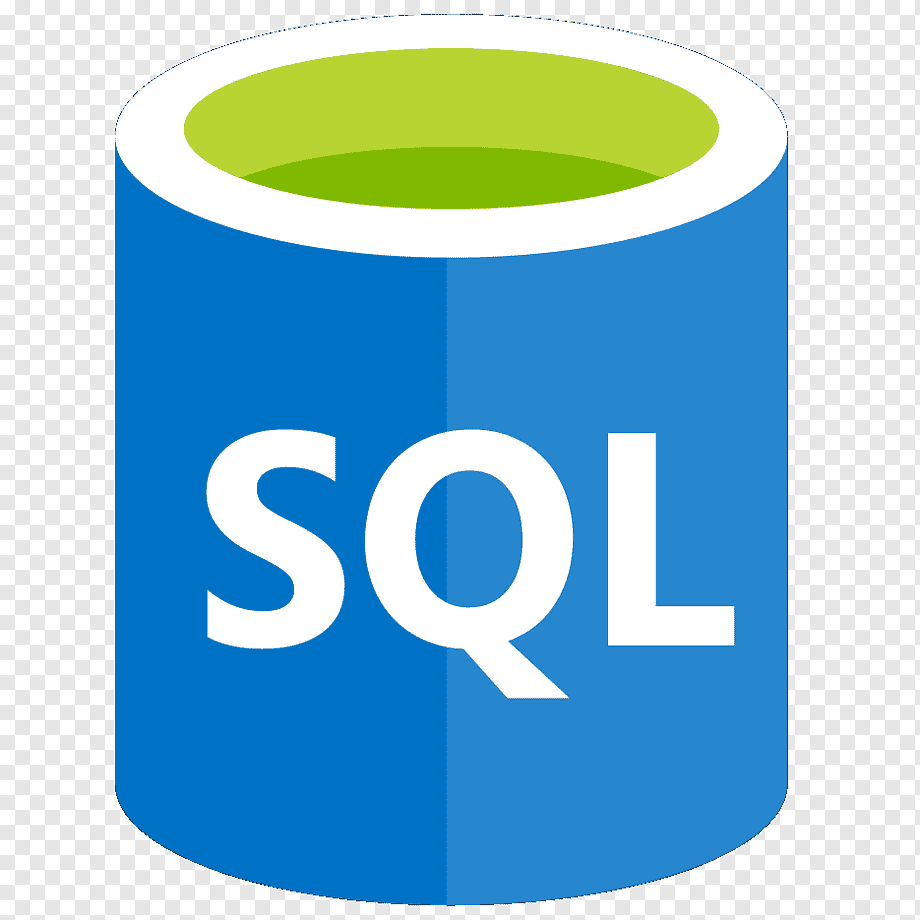
* **JS :JavaScript**

Créé à l'origine par Netscape, ce langage de programmation est conçu pour traiter localement des événements provoqués par le lecteur (par exemple, lorsque le lecteur fait glisser la souris sur une zone de texte, ce dernier change de couleur). C'est un langage interprété, c'est-à-dire que le texte contenant le programme est analysé au fur et à mesure par l'interprète, partie intégrante du browser, qui va exécuter les instructions. Ce langage a fait l'objet d'une normalisation



* **SQL :**

« Structured Query Language » : langage de requête structurée, SQL est un langage informatique normalisé servant à exploiter des bases de données relationnelles. La partie langage de manipulation des données de SQL permet de rechercher, d'ajouter, de modifier ou de supprimer des données dans les bases de données relationnelles.



* **SGBD choisie : MySQL**

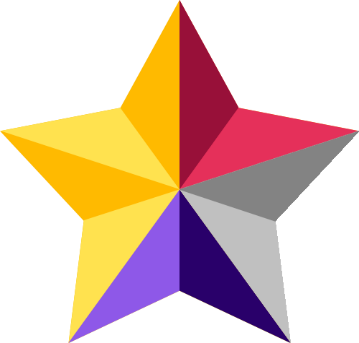
MySQL est un système de gestion de base de données (SGBD). Selon le type d'application, sa licence est libre ou propriétaire. Il fait partie des logiciels de gestion de base de données les plus utilisés au monde, autant par le grand public (applications web principalement) que par des professionnels, en concurrence avec Oracle et Microsoft SQL Server



* **Outil pour la modélisation avec UML : staruml**

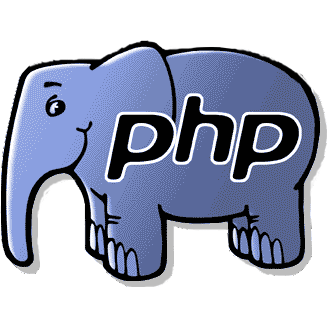
***StarUML*** est un logiciel de modélisation UML (Unified Modeling Language) open source qui peut remplacer dans bien des situations des logiciels commerciaux et coûteux comme Rational Rose1 ou Together2. Étant simple d’utilisation, nécessitant peu de ressources système, supportant UML 2, ce logiciel constitue une excellente option pour une familiarisation à la modélisation. Cependant, seule une version Windows est disponible.

Site web du logiciel : [http://staruml.sourceforge.net/en/](http://staruml.sourceforge.net/en/ )



* **PHP :**  Hypertext Preprocessor

Un langage de scripts libre4 principalement utilisé pour produire des pages Web dynamiques via un serveur HTTP3, mais pouvant également fonctionner comme n'importe quel langage interprété de façon locale, en exécutant les programmes en ligne de commande. PHP est un langage impératif disposant depuis la version 5 de fonctionnalités de modèle objet complètes. En raison de la richesse de sa bibliothèque, on désigne parfois PHP comme une plate-forme plus qu'un simple langage



### Laravel

**Laravel est un framework de développement web open-source basé sur le langage de programmation PHP. Il a été créé par Taylor Otwell et est devenu l'un des frameworks PHP les plus populaires et les plus utilisés.**

#### Introduction

**Laravel est un framework de développement web open-source basé sur le langage de programmation PHP. Il a été créé par Taylor Otwell et est devenu l'un des frameworks PHP les plus populaires et les plus utilisés.**



#### Xampp

**XAMPP** est un logiciel puissant et polyvalent qui facilite la création d'un environnement de développement web complet sur un ordinateur local. Cette suite logicielle, distribuée gratuitement en open-source, rassemble plus de composants essentiels pour le développement du web, notamment le serveur web Apache, le système de gestion de bases de données MySQL, le programme de démation de langage PHP et le serveur de messagerie électronique Mercury.

Site web du logiciel : <https://www.apachefriends.org/fr/download.html>



#### Point fort de laravel

* **Syntaxe expressive** : Laravel propose une syntaxe claire et expressive qui facilite le développement d'applications web
* **Structure MVC (Modèle-Vue-Contrôleur)** : Laravel suit le modèle MVC qui sépare clairement les composants de l'application (logique métier, présentation et gestion des requêtes) pour une meilleure organisation et maintenabilité du code.
* **ORM (Object-Relational Mapping) :** Laravel intègre un ORM appelé Eloquent qui simplifie la gestion des bases de données
* **Système de routage puissant :** Laravel offre un système de routage flexible qui permet de définir facilement les URL de l'application
* **Gestion des migrations de base de données**
* **Système de templates Blade :** Laravel intègre le moteur de templates Blade, qui offre une syntaxe simple et expressive pour la création de vues.
* **Gestion des sessions et de l'authentification**.
* **Large écosystème :** Laravel bénéficie d'une communauté active et d'un vaste écosystème de packages supplémentaires

#### Instalation de laravel

**Pour installer Laravel, vous pouvez suivre les étapes suivantes :**

**Configuration requise : Vérifiez que votre environnement de développement satisfait aux exigences minimales de Laravel. Cela inclut PHP (version 7.4 ou supérieure) avec les extensions nécessaires, Composer (un gestionnaire de dépendances PHP) et un serveur de bases de données compatible.**

**Installation de Composer : Si vous n'avez pas encore Composer installé, téléchargez-le à partir du site officiel de Composer (https://getcomposer.org/) et suivez les instructions d'installation propres à votre système d'exploitation.**

**Création d'un projet Laravel : Ouvrez une fenêtre de terminal ou de ligne de commande et naviguez vers le répertoire où vous souhaitez créer votre projet Laravel. Ensuite, exécutez la commande suivante pour créer un nouveau projet Laravel :**

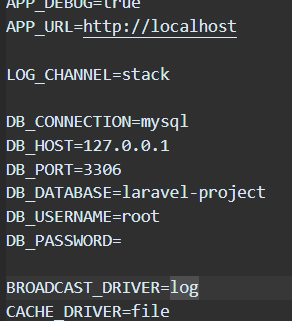
**composer create-project laravel/laravel nom-du-projet**

puis faire ouvrir en Visual Studio Code

#### config data base

**La configuration de la base de données dans Laravel est facile grâce à la modification simple des paramètres de connexion dans le fichier ‘.env’, ce qui permet de spécifier le type de base de données, l'hôte, le port, le nom de la base de données, le nom d'utilisateur et le mot de passe**

**Par exemple :**



#### MVC

##### Model :

**Dans Laravel, un modèle (model en anglais) est une classe qui représente une table de la base de données. Il joue un rôle central dans le modèle MVC (Modèle-Vue-Contrôleur) de Laravel et facilite la manipulation des données**

**Peut être crée un model par la commande suivant :**

**Php artisan make :model Nom\_model**

##### View  :

**Dans Laravel, la partie "View" (vue) fait référence à la couche de présentation de votre application. Elle est responsable de l'affichage des données et de l'interaction avec l'utilisateur. Les vues dans Laravel utilisent un moteur de template puissant appelé Blade qui facilite la création de pages Web dynamiques.**

**Les vues dans Laravel sont généralement stockées dans le répertoire resources/views. Vous pouvez organiser vos vues en sous-répertoires en fonction de la structure de votre application. Chaque vue est un fichier ‘ .blade.php ‘ qui utilise la syntaxe Blade pour incorporer du code PHP et des directives spéciales.**

##### Controller :

**un contrôleur (controller) est une classe qui gère la logique de traitement des requêtes HTTP. Il s'agit d'un élément essentiel de l'architecture MVC (Modèle-Vue-Contrôleur) utilisée par Laravel .**

**Création d'un contrôleur : Vous pouvez créer un contrôleur en utilisant la commande :.**

**php artisan make:controller Name\_controller**

si en cree un controller . il ya plusieurs methodes en controller cree a savoir :

create , store , update , edit , show et destroy .

voici une shema presente le fonctionemment de MVC :



▪ **Justification de choix du modèle MVC**

On a choisi le MVC comme méthodologie d’analyse pour les raisons suivantes :

• Une conception claire et efficace grâce à la séparation des données de la vue et du contrôleur.

• Un gain de temps de maintenance et d’évolution du site.

• Une plus grande souplesse pour organiser le développement du site entre différents développeurs (indépendance des données, de l’affichage et des actions)

## Présentation de l’application

### Architecture application

Cree une message

Aux administration

**visiteur**

**Information de sit web**

**administrateur**

**Gestions**

évènements

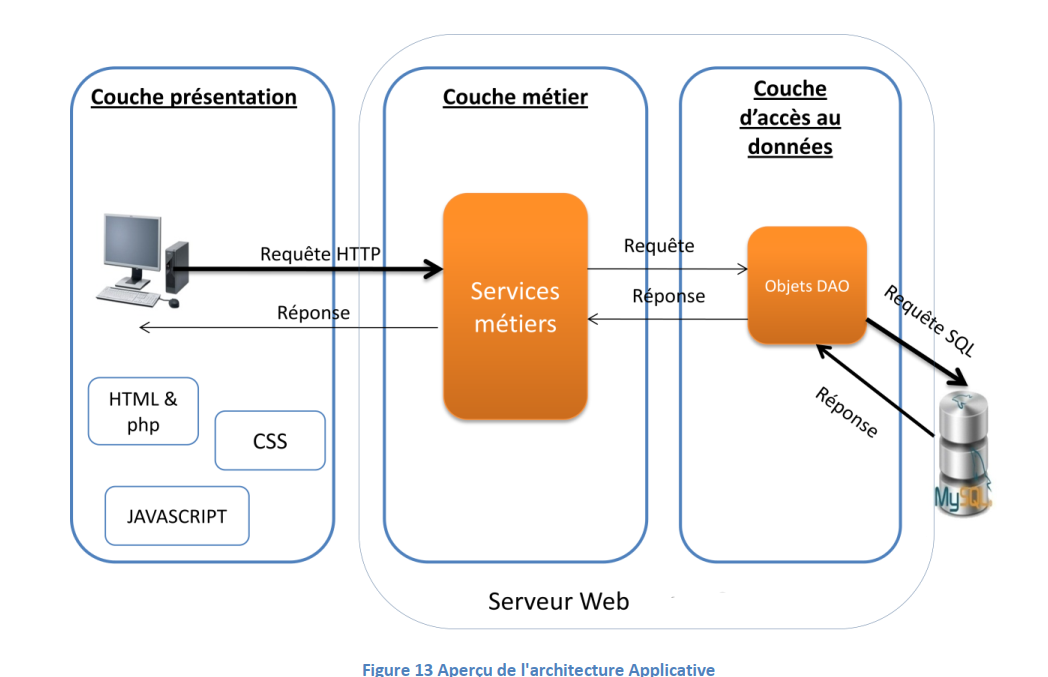
**Gestions des contact**

**Gestions des families**

**Gestions des categories**

**L’architecture applicative** du projet contient trois couches : couche présentation, couche métier et couche d’accès aux données.

La couche présentation contient les pages web ,

l’utilisateur envoie des requêtes http vers la couche métier, cette dernière contient un ensemble de fonctionnalités principales de l’application qui interagit avec la couche d’accès aux données qui gère le stockage et la manipulation des données l’aide des requêtes SQL vers la base de donnée MYSQL

### Captures de l’application :

**CHAPITRE 5 :**

# CONCLUSION

**CONCLUSION**

Le travail effectué s'est avéré très riche pour mon expérience professionnelle, tant sur le plan technique que sur le plan humain. En effet, durant ces 5 semaines de formation chez FST Marrakech , vous m'avez fait découvrir le fonctionnement de laravel.

J'ai pu apporter mes connaissances théoriques et approfondir certains domaines que je ne connaissais pas encore ; J'ai pu découvrir un tas d'outils utilisés dans la programmation web.

Une des réalisations les plus satisfaisantes de ce stage a été la mise en place d'un système de requêtes SQL préparées avec mysql, garantissant la sécurité des données et protégeant la plateforme contre les attaques par injection SQL. De plus, l'intégration de StarUML pour la modélisation UML et de XAMPP pour le développement en laravel a grandement facilité le processus de développement.

Bien que le projet "Family Calendar" ait déjà franchi ses premières étapes avec succès, il n'en est qu'à ses débuts. Le potentiel d'amélioration et d'expansion est considérable. Dans les prochaines phases, envisager l'ajout de fonctionnalités avancées comme la génération de rapports pour mieux comprendre les habitudes familiales. L'intégration de systèmes de paiement pourrait faciliter la gestion des dépenses collectives. L'idée passionnante d'une application mobile pourrait également prendre forme, offrant une expérience encore plus pratique et en phase avec la mobilité moderne. Ces évolutions, parmi d'autres, promettent de renforcer l'utilité et la pertinence du "Family Calendar" pour les familles en quête d'organisation et de communication efficace.