# Table des matières

[Table des matières I](#_Toc115725331)

[Liste des tableaux V](#_Toc115725332)

[Introduction générale 1](#_Toc115725333)

[Chapitre I : Etude préalable 2](#_Toc115725334)

[1 Introduction 2](#_Toc115725335)

[2 Présentation générale du laboratoire de recherche « SETIT » 2](#_Toc115725336)

[3 Etude d’existant 3](#_Toc115725337)

[3.1 Problématique 3](#_Toc115725338)

[3.2 Solutions proposées 4](#_Toc115725339)

[3.3 Spécification des besoins 5](#_Toc115725340)

[3.3.1 Les besoins fonctionnels 5](#_Toc115725341)

[3.3.2 Besoins non fonctionnels 6](#_Toc115725342)

[4 Prototypes 6](#_Toc115725343)

[4.1 Organigramme de l’espace administrateur 7](#_Toc115725344)

[4.2. Organigramme de la partie membre 9](#_Toc115725345)

[5 Conclusion 9](#_Toc115725346)

[Chapitre II : Etude conceptuelle 10](#_Toc115725347)

[1. Introduction 10](#_Toc115725348)

[2. Language de modélisation UML 10](#_Toc115725349)

[2.1. Définition 10](#_Toc115725350)

[2.2. Les différents types des diagrammes d’UML 10](#_Toc115725351)

[2.3. Les points forts d’UML 11](#_Toc115725352)

[3. Méthodologie de travail : cycle en V 11](#_Toc115725353)

[3.1. Définition 11](#_Toc115725354)

[3.2. Explication de modèle cycle en V 12](#_Toc115725355)

[4. Diagramme de cas d’utilisation 13](#_Toc115725356)

[4.1. Les acteurs systèmes 14](#_Toc115725357)

[4.2. Le diagramme de cas d’utilisation globale 14](#_Toc115725358)

[4.3. Le diagramme de cas d’utilisations « Gestion des comptes des utilisateurs » 16](#_Toc115725359)

[4.4. Diagramme de d’utilisation « Gestion des productions » 17](#_Toc115725360)

[4.5. Diagramme de cas d’utilisations « Gestion des coopérations » 18](#_Toc115725361)

[4.6. Diagramme de cas d’utilisation « Gestion des évènements » 18](#_Toc115725362)

[5. Diagramme de séquence 19](#_Toc115725363)

[5.1. Diagramme de séquence « Connexion » 19](#_Toc115725364)

[5.2. Diagramme de séquence « Ajouter Actualités » 21](#_Toc115725365)

[5.3. Diagramme de séquence « Modification d’un domaine » 23](#_Toc115725366)

[5.4. Diagramme de séquence « Envoie d’un message » 25](#_Toc115725367)

[6. Diagramme de classe 26](#_Toc115725368)

[7. Conclusion 28](#_Toc115725369)

[Chapitre III : Réalisation 29](#_Toc115725370)

[1. Introduction 29](#_Toc115725371)

[2. Environnement de développement 29](#_Toc115725372)

[2.1. Environnement matériel 29](#_Toc115725373)

[2.2. Environnement logiciel 29](#_Toc115725374)

[2.2.1. HTML 5 29](#_Toc115725375)

[2.2.2. CSS 3 30](#_Toc115725376)

[2.2.3. JavaScript 30](#_Toc115725377)

[2.2.4. JQuery 31](#_Toc115725378)

[2.2.5. Ajax 31](#_Toc115725379)

[2.2.6. Node JS 31](#_Toc115725380)

[2.2.7. Bootstrap5 32](#_Toc115725381)

[2.2.8. Xampp 32](#_Toc115725382)

[2.2.9. PhpMyAdmin 32](#_Toc115725383)

[2.2.10. MySQL 33](#_Toc115725384)

[2.2.11. Laravel 9 33](#_Toc115725385)

[2.2.12. Github 34](#_Toc115725386)

[2.2.13. Visual Studio 34](#_Toc115725387)

[3. Architecture globale 35](#_Toc115725388)

[3.1. Les composants de MVC 35](#_Toc115725389)

[3.1.1. Modèle 35](#_Toc115725390)

[3.1.2. Vue 36](#_Toc115725391)

[3.1.3. Contrôleur 36](#_Toc115725392)

[4. Réalisation 36](#_Toc115725393)

[4.1. Interface « Accueil » 37](#_Toc115725394)

[4.2. Interface «Thème de recherche » 37](#_Toc115725395)

[4.3. Interface 38](#_Toc115725396)

[5. Building 38](#_Toc115725397)

[6. Conclusion 38](#_Toc115725398)

[Conclusion générale 39](#_Toc115725399)

[WEBOGRAPHIE 40](#_Toc115725400)

**Table des figures**

[**Figure 1.** Logo du laboratoire de recherche SETIT 2](#_Toc115725442)

[**Figure 2.** Organigramme l’espace administrateur 8](#_Toc115725443)

[**Figure 3.** Organigramme l'espace membre 9](#_Toc115725444)

[**Figure 4.** Méthode de cycle en V 12](#_Toc115725445)

[**Figure 5.** Les étapes de la méthode de cycle en V 13](#_Toc115725446)

[**Figure 6.** Cas d'utilisation globale 15](#_Toc115725447)

[**Figure 7.** Diagramme de cas d'utilisation de gestion des comptes utilisateurs 17](#_Toc115725448)

[**Figure 8.** Diagramme de cas d'utilisation "Gestion des publications" 18](#_Toc115725449)

[**Figure 9.** Diagramme de cas d'utilisation "Gestion des coopérations" 18](#_Toc115725450)

[**Figure 10.** Diagramme d'utilisation "gestion des évènements" 19](#_Toc115725451)

[**Figure 11.** Diagramme de séquence "Connexion" 20](#_Toc115725452)

[**Figure 12.** Diagramme de séquence « Actualités » 22](#_Toc115725453)

[**Figure 13.** Diagramme de séquence "Modification domaine" 24](#_Toc115725454)

[**Figure 14.** Diagramme de séquence "Envoie des messages" 25](#_Toc115725455)

[**Figure 15.** Diagramme de classe de l'application 27](#_Toc115725456)

[**Figure 16.** Logo HTML5 29](#_Toc115725457)

[**Figure 17.** Logo de CSS3 30](#_Toc115725458)

[**Figure 18.** Logo de JavaScript 30](#_Toc115725459)

[**Figure 19.** Logo de JQuery 31](#_Toc115725460)

[**Figure 20.** Logo d’AJAX 31](#_Toc115725461)

[**Figure 21.** Logo de NodeJS 32](#_Toc115725462)

[**Figure 22.** Logo de Bootstrap5 32](#_Toc115725463)

[**Figure 23.** Logo de Xampp 32](#_Toc115725464)

[**Figure 24.** Logo de PhpMyAdmin 33](#_Toc115725465)

[**Figure 25.** Logo de MySQL 33](#_Toc115725466)

[**Figure 26.** Logo de Laravel 9 34](#_Toc115725467)

[**Figure 27.** Logo de Github 34](#_Toc115725468)

[**Figure 28.** Logo de Visual Studio 35](#_Toc115725469)

[**Figure 29.** Architecture MVC 35](#_Toc115725470)

[**Figure 30.** Modèle 36](#_Toc115725471)

[**Figure 31.**Vue 36](#_Toc115725472)

[**Figure 32.** Contrôleur 36](#_Toc115725473)

[**Figure 33.** Interface « Accueil » 37](#_Toc115725474)

[**Figure 34.** Interface « Thème de recherche » 37](#_Toc115725475)

# Liste des tableaux

[**Tableau 1.** Description des acteurs systèmes 18](#_Toc115545964)

[**Tableau 2.** Les types de services 20](#_Toc115545965)

[**Tableau 3.** Description textuelles de diagramme de séquence "connexion" 24](#_Toc115545966)

[**Tableau 4.** Description textuelle de diagramme de séquence "Actualités" 26](#_Toc115545967)

[**Tableau 5.** Description textuelle de diagramme de séquence "Modification d'un domaine" 28](#_Toc115545968)

[**Tableau 6.** Description textuelle de diagramme de séquence "Envoie des messages" 29](#_Toc115545969)

[**Tableau 7.** Environnement matériel 33](#_Toc115545970)

# Introduction générale

# Chapitre I : Etude préalable

# Introduction

Dans ce chapitre, nous proposons une solution complète d’un portail web qui permet de gérer les activités du laboratoire de recherche « SETIT » et automatiser la gestion et la mise à jour des publications des chercheurs. Ce portail offre aussi d’autres services aux acteurs de cette application qu'on va détailler dans la suite de ce chapitre juste après la présentation du laboratoire de recherche « SETIT » dans laquelle nous effectuons notre stage.

# Présentation générale du laboratoire de recherche « SETIT »

Créé en janvier 2020, le Laboratoire de recherche. « **S**ystèmes intelligents pour l’ingénierie & l'**E**-santé basés sur les **T**echnologies de l'**I**mage & les **T**élécommunications » (Lab. SETIT) n’est autre qu’une évolution de l´Unité de Recherche **S**ciences **E**t **T**echnologies de l’**I**mage et des **T**élécommunications (UR SETIT). Cette dernière. trouve ses origines au sein du laboratoire de recherche LETI de l´école Nationale d´Ingénieur de Sfax ENIS. Depuis sa création en Janvier 2004, l´unité SETIT, devenue Lab SETIT, est rattachée. à l´Institut Supérieur de Biotechnologie de Sfax ISBS.

Le laboratoire de recherche « SETIT » touche plusieurs domaines de recherche tel que l´analyse, le traitement, la sécurisation (cryptage et tatouage) la compression, la transmission des images ainsi que l'interaction Homme Machine. [[1]](#_WEBOGRAPHIE)



**Figure 1.** Logo du laboratoire de recherche SETIT

Les chercheurs. du laboratoire de recherche « SETIT » tentent de répondre à ces thématiques en mêlant une approche mathématique fortement algorithmique et une approche pratique basée sur les outils informatiques. Dans le cadre de l'activité de SETIT, ils tentent de prendre en compte les préoccupations tant médicales qu'industrielles dans le choix des objectifs stratégiques du laboratoire. Leurs travaux sont naturellement regroupés autour de grands thèmes tels que : la segmentation, le tatouage, la compression, la crypto-compression et la transmission.

Par ailleurs, ils notent que le tatouage d'image et la crypto-compression deux domaines dans lesquels ils émergent à l'échelle mondiale grâce au développement de nouvelles approches. Ils sont aussi les premiers à les soigner en Tunisie.

# Etude d’existant

## 3.1 Problématique

Le portail web actuel du laboratoire de recherche SETIT présente plusieurs problèmes et plusieurs failles.

On a reparti les problèmes en quatre parties :

* **Les problèmes du font, on cite :**
* Le couleur sombre de font ;
* Il existe quelques caractères qui sont illisibles ;
* Le titre de tous les pages sont uniques « AdminLTE3.. » ;
* Des éléments sont mal positionnées ;
* Lors de l’édition du n’importe quel formulaire, l’entête est sous le nom de « Création » ;
* Il existe des problèmes de margin et de padding ;
* …
* **Les problèmes de back-end, on cite :**
* Aucune indication dans la base des données aux modifications dues sur les formulaires ;
* Lors de la suppression d’une ligne dans un tableau, elle se détruit de la base des données.
* **Les problèmes liés à l’utilisation de la technologie de programmation**, ce portail a été programmé avec le PHP Native, bien que cette technologie est facile à apprendre et open source, elle présente des inconvénients tel que :
* Problème de connexion ;
* Problèmes de sécurité ; ce portail est facile à pirater..
* **Problèmes de fonctionnement**, il manque des fonctionnalités tel que :
* La messagerie entre les utilisateurs et l’administrateur ;
* Le produit « Master » n’existe pas dans le module de production ;
* Ce portail de la partie administrative ne présente pas les statiques.

3.2 Solutions proposées

Pour éviter ces problèmes le laboratoire de recherche SETIT, propose de refaire le développement de portail web qui permet d’offrir divers services au chercheur membre du laboratoire.

Pour aboutir à réaliser le travail demandé, j’ai été mené à utiliser le Framework PHP Laravel 9, suivre une conception. IHM facile, lisible, garantir la sécurité. d’accès des membres ainsi que la création des comptes membres. La plateforme à développer devrait aussi permettre de définir les rôles de chaque utilisateur (espace membre de l’espace administrateur).

Parmi les fonctionnalités demandées dans ce projet est de développer un portail web qui permet :

* La consultation des manifestations, des productions, des actualités.. de laboratoire de recherche « SETIT » ;
* La gestion du laboratoire de recherche « SETIT ».
* La mise à jour des publications des chercheurs.
* La gestion des différentes permises à un membre :
* Authentification ;
* La gestion de profil ;
* La gestion des publications ;
* La gestion des productions scientifiques ;
* La gestion des coopérations ;
* La gestion des différentes tâches autorisées à l’administrateur :
* Authentification.
* La gestion de profil.
* Visualisation des statiques.
* La gestion des utilisateurs.
* Jouer avec les rôles des membres.
* La gestion des publications.
* La gestion des productions scientifiques.
* La gestion des manifestations.
* La gestion des coopérations.
* La gestion des événements.
* La gestion des messages.

## ***3.3 Spécification des besoins***

Pour garantir la réussite d’une solution, il est nécessaire de spécifier les besoins fonctionnels et les besoins non fonctionnels auxquels doit répondre le système à développer

### 3.3.1 Les besoins fonctionnels

L’objectif de notre projet est de développer un portail web qui permet la gestion de ressources du laboratoire de recherche « SETIT ». Ce projet doit répondre aux besoins fonctionnels qui nous sont présentés dans la suite :

* Les besoins fonctionnels de visiteur.
* Le visiteur peut découvrir le laboratoire ;
* Le visiteur peut consulter les manifestations ;
* Le visiteur peut consulter les domaines de recherches ;
* Le visiteur peut consulter les productions scientifiques ;
* Le visiteur peut consulter les publications ;
* Le visiteur peut consulter les actualités ;
* Le visiteur peut s’inscrire s’il y est un membre dans le laboratoire.
* Les besoins fonctionnels de membre.
* Le membre peut gérer les publications par l’ajout, suppression et modifications ;
* Le membre peut gérer les productions par l’ajout, suppression et modifications ;
* Le membre peut gérer les coopérations par l’ajout, suppression et modifications ;
* Le membre peut envoyer des messages au administrateur ;
* Le membre peut gérer son profil par la modification des données.
* Les besoins fonctionnels de l’administrateur.
* L’administrateur peut gérer les membres par ajout, suppression, modification, définition tant qu’administrateur et retirer tant qu’administrateur.
* L’administrateur peut gérer les publications par ajout, suppression et modification.
* L’administrateur peut gérer les productions scientifiques par ajout, suppression et modification.
* L’administrateur peut gérer les coopérations par ajout, suppression et modifications.
* L’administrateur peut gérer les manifestations par ajout, suppression et modifications.
* L’administrateur peut gérer l’agenda des événements par ajout, suppression et modification.
* L’administrateur peut gérer les messages (lire et/ou supprimer).
* L’administrateur peut visualiser les statistiques des membres, des manifestations, des productions et des publications.
* L’administrateur peut gérer son profil.

### 3.3.2 Besoins non fonctionnels

En plus des besoins fonctionnels cités précédemment, pour répondre correctement aux exigences des utilisateurs de ce site Web, le système doit assurer les besoins non fonctionnels suivants :

* Le site Web doit être efficace, fiable et facile à utiliser.
* Les taches doivent être indépendantes pour ne pas être bloquées à une étape particulière.
* Le site doit être basé sur une conception orientée objet.
* Le site doit être sécurisé et doit recevoir des améliorations.
* Le site doit utiliser des outils de programmations innovantes.
* Le site doit prendre en compte l’esthétique de la conception d’applications web.
* Le site doit être facile à maintenir.

# Prototypes

Dans ce projet, je vais développer un portail web de la partie administrative qui permet de gérer les activités. du laboratoire de recherche SETIT et automatisé la mise. à jour des publications des chercheurs membre. Ce portail permet aussi de facilité la communication entre les membres du laboratoire. et offre des coopérations avec des chercheurs nationaux et étrangers.

## 4.1 Organigramme de l’espace administrateur

Afin de schématiser les liens fonctionnels, organisationnels et hiérarchiques de la partie administrative. On présente les différentes fonctionnalités de l’espace administrateur qu’on va développer par l’organigramme ci-dessous (Figure 2).

**Figure 2.** Organigramme l’espace administrateur

## 4.2. Organigramme de la partie membre

La figure suivante présente une schématisation des liens fonctionnels, organisationnels et hiérarchique de la partie membre.

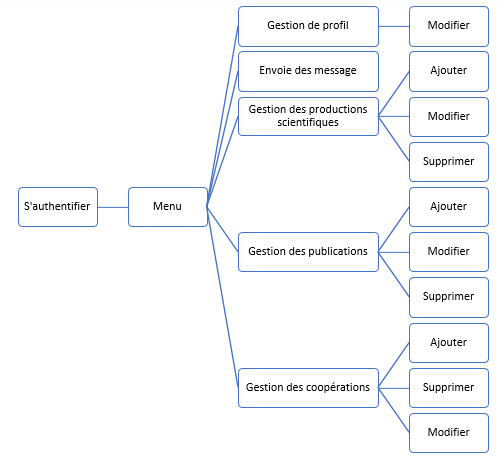


Figure . Organigramme l'espace membre

# Conclusion

Dans ce chapitre, on a présenté les principaux services du portail, tout en identifiant et modélisant les. besoins fonctionnels et non fonctionnels. Dans le chapitre suivant, on détaille la partie conception du projet ainsi que la méthodologie de travail adoptée.

# Chapitre II : Etude conceptuelle

1. Introduction

Dans ce chapitre, on présente la méthodologie de travail adoptée et la partie conception du portail en se basant sur le langage de modélisation UML.

# Language de modélisation UML

## Définition

UML (Unified Modeling Language) se compose de. diagrammes d'intégration utilisés par les développeurs informatiques pour représenter visuellement. des objets, des états et des processus dans des logiciels ou des systèmes [[2].](#_WEBOGRAPHIE)

## Les différents types des diagrammes d’UML

UML définit 14 types de diagrammes, divisés en deux groupes.

Les diagrammes de structure représentent les éléments du système, leurs propriétés et les relations entre eux.

• Diagramme de classes. ;

• Diagramme d’objets. ;

• Diagramme de composants ;

• Diagramme de structure composite ;

• Diagramme d'ensemble ;

• Diagramme de déploiement. ;

• Diagramme de profil.

Les diagrammes de comportement. représentent les processus et les interactions entre les objets.

• Diagramme de cas d'utilisation ;

• Diagramme d'activité ;

• Diagramme. d'état-transition ;

• Diagramme de séquence ;

• Diagramme de communication ;

• Diagramme de temps ;

• Diagramme d'aperçu d'interaction.

## Les points forts d’UML

UML est un langage formel standardisé. qu’il permet de gagner la précision, la stabilité et encourager l'utilisation d'outils. En effet, UML permet de construire un cadre analytique qui aide à comprendre les représentations abstraites et complexes. D’autre part, son polyvalence. et son flexibilité font un langage graphique universel qui est conçu .pour spécifier, construire, visualiser et modéliser tous types de systèmes.

# Méthodologie de travail : cycle en V

## Définition

Afin de mener notre projet dans les meilleures conditions de développement, nous avons adopté le processus de développement de cycle en V. Le modèle en V est une méthodologie de développement de projets informatiques et un modèle conceptuel de gestion de projet depuis les années 80. C’est une méthodologie qui décrit les étapes essentielles du développement d'un logiciel et le cycle de vie du projet. La branche descendante du modèle en V contient toutes les étapes de la conception du projet, et la branche montante toutes les étapes de tests du projet. La pointe du V représente la réalisation concrète du projet qui est le codage [[3].](#_WEBOGRAPHIE)

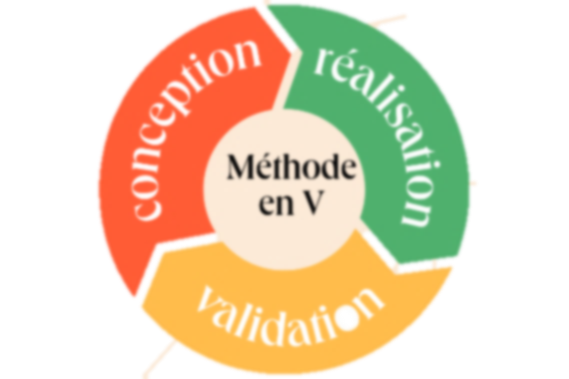


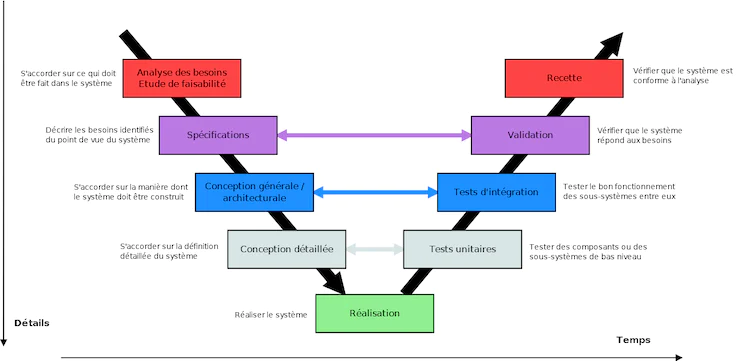
Figure . Méthode de cycle en V

## Explication de modèle cycle en V

Les neuf étapes peuvent être regroupées en trois phases : la conception, la mise en œuvre et la validation.

* **La conception (la partie descendante)**
* L’expression des besoins et l’étude de faisabilité : Recueil des besoins du client (produit souhaité, budget disponible…). Cette étape peut aussi faire l’objet d’une étude de faisabilité.
* La définition des spécifications et du cahier des charges fonctionnel : Rédaction d’un cahier des charges fonctionnel (CDCF) à partir des spécifications fonctionnelles du produit. Il s’agit d’un document résumant les objectifs du projet, les parties prenantes, les contraintes techniques, les attentes… Cette étape peut s’appuyer sur une analyse SWOT pour un diagnostic externe et interne complet de l’entreprise.
* La conception générale/architecturale : Liste des moyens nécessaires à la réalisation du projet. Cette étape permet d’appréhender les besoins techniques et financiers pour la réalisation du projet. Si l’entreprise repère un point de blocage à cette étape, elle peut, en accord avec le client, revoir le cahier des charges.
* La conception détaillée : Détail des composants ou étapes indispensables à la création du produit.
* **La mise en œuvre**
* Réalisation ou intégration système : Création et assemblage de tous les composants pour réaliser le produit final.
* **La validation (la partie ascendante)**
* Les tests unitaires : Pour chaque composant ou fonctionnalité : cette étape fait écho à la conception détaillée et lui fait face sur le schéma du V.
* Test d’intégration : L’intégration et les tests d’intégration, sur le produit fini, cette étape permet de vérifier le fonctionnement du système défini à l’étape de conception générale.
* Tests système (aussi appelés tests de validation) : C’est-à-dire la conformité fonctionnelle du produit ou du logiciel par rapport aux spécifications communiquées par le client, cette étape est la validation du respect des spécifications définies en phase descendante.
* Test d’acceptation (aussi appelé recette fonctionnelle) : cette étape est la validation de la conformité face à l’expression des besoins.

On peut résumer les 9 étapes de cette méthode par la figure ci-dessous :



**Figure 5.** Les étapes de la méthode de cycle en V

# Diagramme de cas d’utilisation

En langage UML, 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. Ces diagrammes identifient également les interactions entre le système et ses acteurs. Les cas d'utilisation et les acteurs dans les diagrammes de cas d'utilisation décrivent ce que le système fait et comment les acteurs l'utilisent, mais ne montrent pas comment le système fonctionne en interne [4].

* Acteur : représente un rôle joué par une personne qui interagit directement avec le système étudié.
* Cas d’utilisation (use case) : représente un ensemble des séquences d’actions qui sont réalisées par le système et qui produisent un résultat observable intéressant pour un acteur particulier.

## 4.1. Les acteurs systèmes

Dans notre système en fait intervenir 3 acteurs compte tenu de déroulement et la complémentarité des opérations décrites dans le tableau suivant.

Tableau . Description des acteurs systèmes

|  |  |
| --- | --- |
| **Acteurs** | **Description** |
| Administrateur | 1. S’authentifier ; 2. Gérer les messages ; 3. Gérer le profil ; 4. Visualiser les statistiques ; 5. Gérer les membres du site : ajouter (), supprimer (), modifier (), définir tant qu’administrateur (), retirer tant qu’administrateur () ; 6. Gérer les actualités : ajouter (), supprimer (), modifier () ; 7. Gérer les évènements : ajouter (), supprimer (), modifier (); 8. Gérer les domaines de recherche : ajouter (), supprimer (), modifier (); 9. Gérer les publications : ajouter (), supprimer (), modifier (); 10. Gérer les productions scientifiques : ajouter (), supprimer (), modifier (); 11. Gérer les coopérations : ajouter (), supprimer (), modifier () ; 12. Se déconnecter. |
| Membre | 1. S’authentifier ; 2. Gérer le profile ; 3. Envoyer des messages ; 4. Gérer les publications : ajouter (), supprimer (), modifier (); 5. Gérer les productions scientifiques : ajouter (), supprimer (), modifier (); 6. Gérer les coopérations : ajouter (), supprimer (), modifier () ; 7. Se déconnecter. |
| Visiteur | 1. Consulter les publications ; 2. Consulter les productions scientifiques ; 3. Consulter les manifestations ; 4. Consulter les domaines de recherche ; 5. Consulter les actualités ; 6. S’inscrire. |

## 4.2. Le diagramme de cas d’utilisation globale

Le diagramme de cas d’utilisation globale représenté par la Figure 6 montre les différentes activités qu’un administrateur, membre ou visiteur peut effectuer.

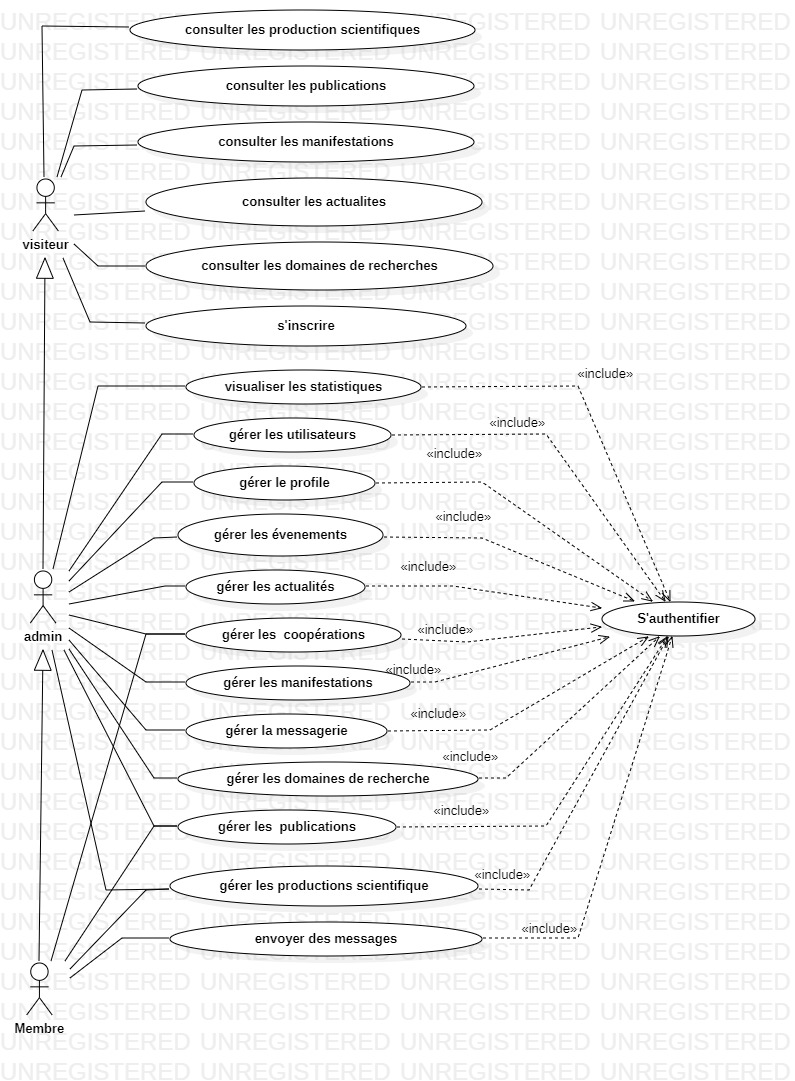


Figure . Cas d'utilisation globale

Le tableau 2 décrit les différents types de publications, productions scientifiques, manifestations et coopérations.

Tableau . Les types de services

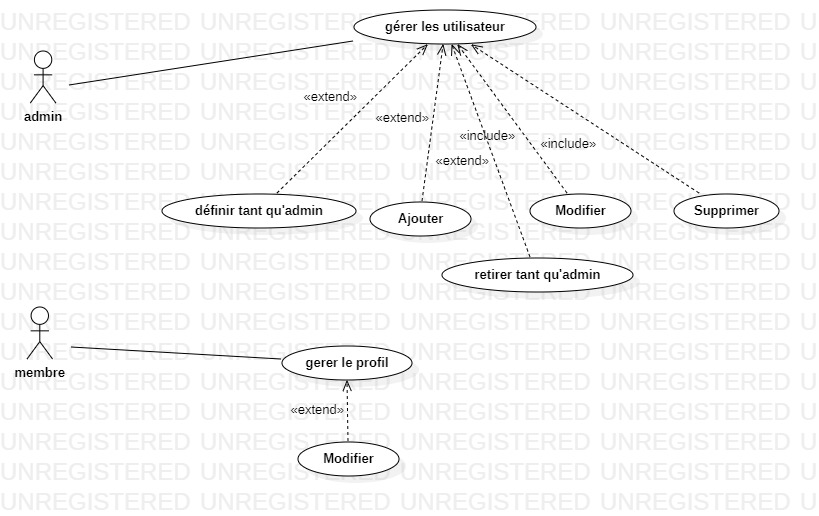
|  |  |
| --- | --- |
| **Services** | **Types** |
| Publications | 1. Article scientifiques ; 2. Ouvrage scientifique ; 3. Chapitre d’ouvrage ; 4. Brevets ; 5. Conférence. |
| Productions | 1. Habilitation ; 2. Thèse ; 3. Master ; 4. PFE. |
| Manifestations | 1. Evènements ; 2. Conférences ; 3. Formations |
| Coopérations | 1. Internationaux ; 2. Nationaux. |

## 4.3. Le diagramme de cas d’utilisations « Gestion des comptes des utilisateurs »

Ce diagramme modélise les différentes étapes de la gestion des comptes des utilisateurs pour les membres et les administrateurs.

L’administrateur peut gérer les comptes utilisateurs en effectuant soit l’ajout, la modification, la suppression, la définition tant qu’administrateur, le retrait de l’administration de site.

Le membre peut gérer son profil par des modifications.

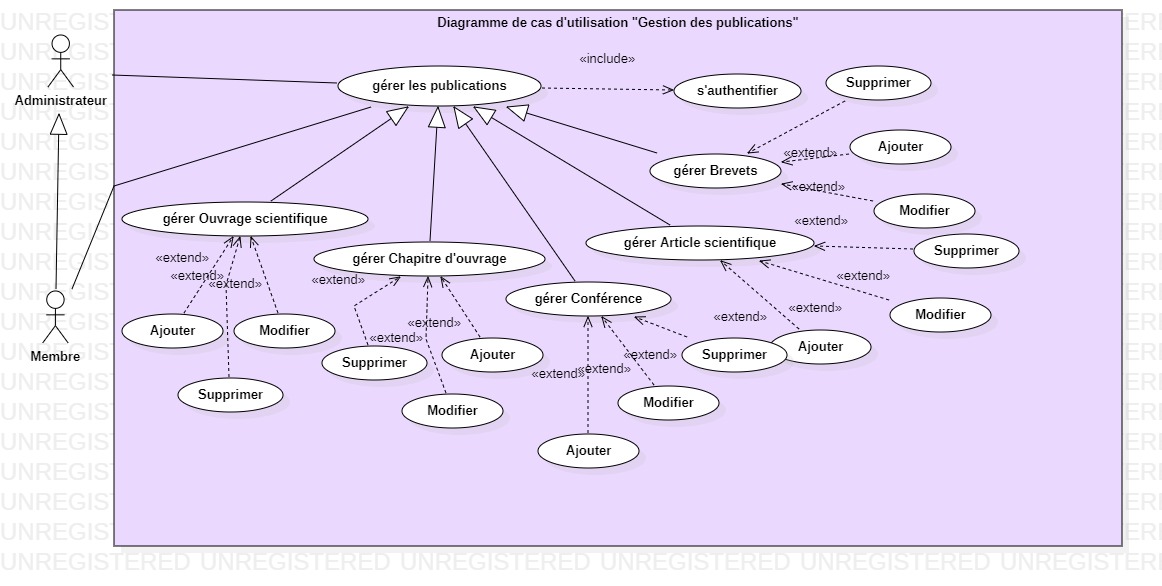


**Figure 7.** Diagramme de cas d'utilisation de gestion des comptes utilisateurs

## Diagramme de d’utilisation « Gestion des productions »

Il existe quelques fonctionnalités qui sont commun entre les administrateurs et les membres du site tels que la gestion des productions et la gestion des publications.

Le diagramme de cas d’utilisation pour la gestion des productions est illustré par la figure 8.

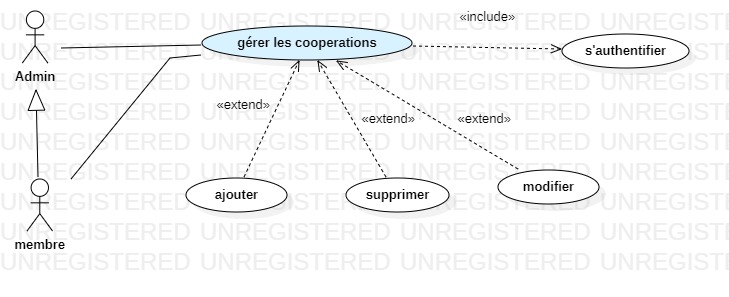


**Figure 8.** Diagramme de cas d'utilisation "Gestion des publications"

## Diagramme de cas d’utilisations « Gestion des coopérations »

Le diagramme ci-dessous présente les différentes étapes de la gestion des coopérations pour les acteurs administrateur et membres.

Les coopérations peuvent être gérer soit par l’administrateur soit par les membres.

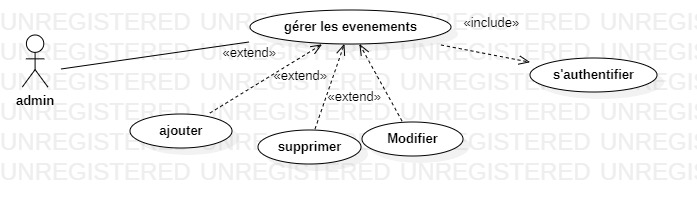


**Figure 9.** Diagramme de cas d'utilisation "Gestion des coopérations"

## Diagramme de cas d’utilisation « Gestion des évènements »

Seul l’administrateur peut gérer les évènements par l’ajout, la modification et la suppression.

Bref, le diagramme de cas d’utilisation pour la gestion des évènements est décrit par la figure 10.



**Figure 10.** Diagramme d'utilisation "gestion des évènements"

Les étapes de la gestion des évènements rassemblent beaucoup aux gestions des actualités, des domaines et des manifestations du coup ils présentent des diagrammes de cas d’utilisations identiques à celle de la gestion des évènements.

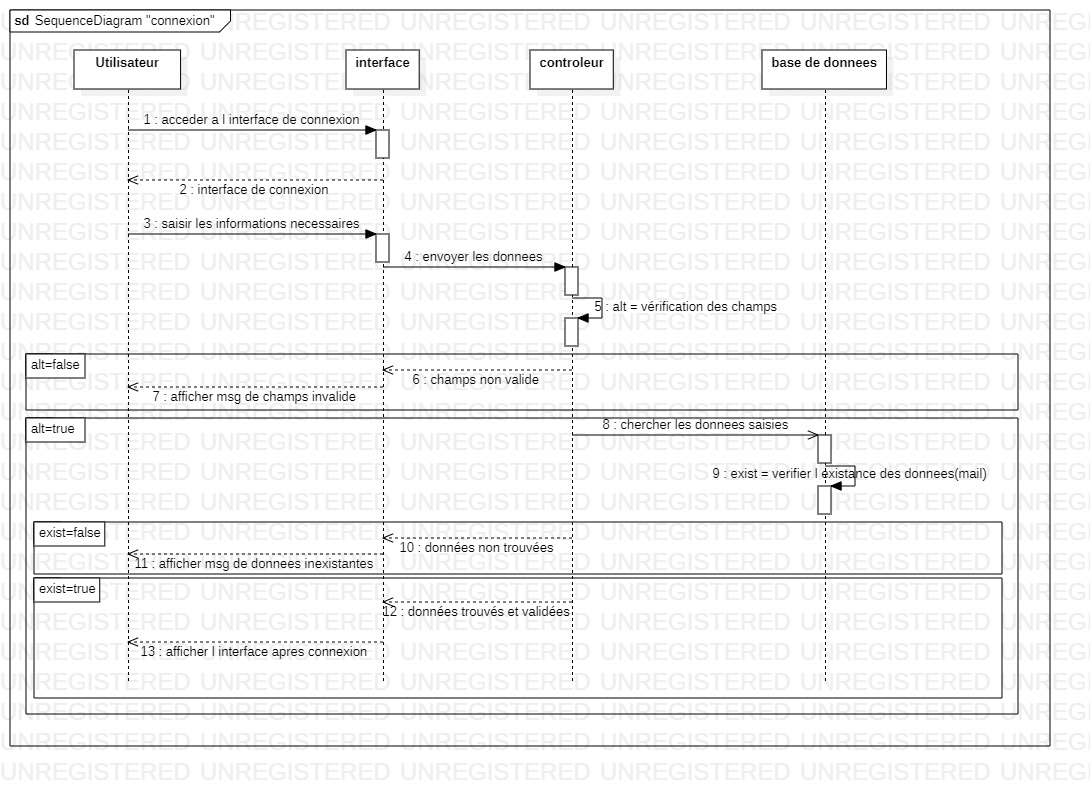
# Diagramme de séquence

Un diagramme de séquence est un diagramme UML qui représente la séquence de messages entre les objets au cours d'une interaction. Un diagramme de séquence comprend un groupe d'objets, représentés par des lignes de vie, et les messages que ces objets échangent lors de l'interaction [[5].](#_WEBOGRAPHIE)

## Diagramme de séquence « Connexion »

En fait, pour profiter des services offerts par notre site, il faut tout d’abord se connecter par une adresse mail et un mot de passe.

Un diagramme de séquence illustre les scénarios possibles qui se passent pour réussir la connexion (figure 10).



**Figure 11.** Diagramme de séquence "Connexion"

Le tableau 3 représente une description textuelle du diagramme de séquence élaboré par la figure 10.

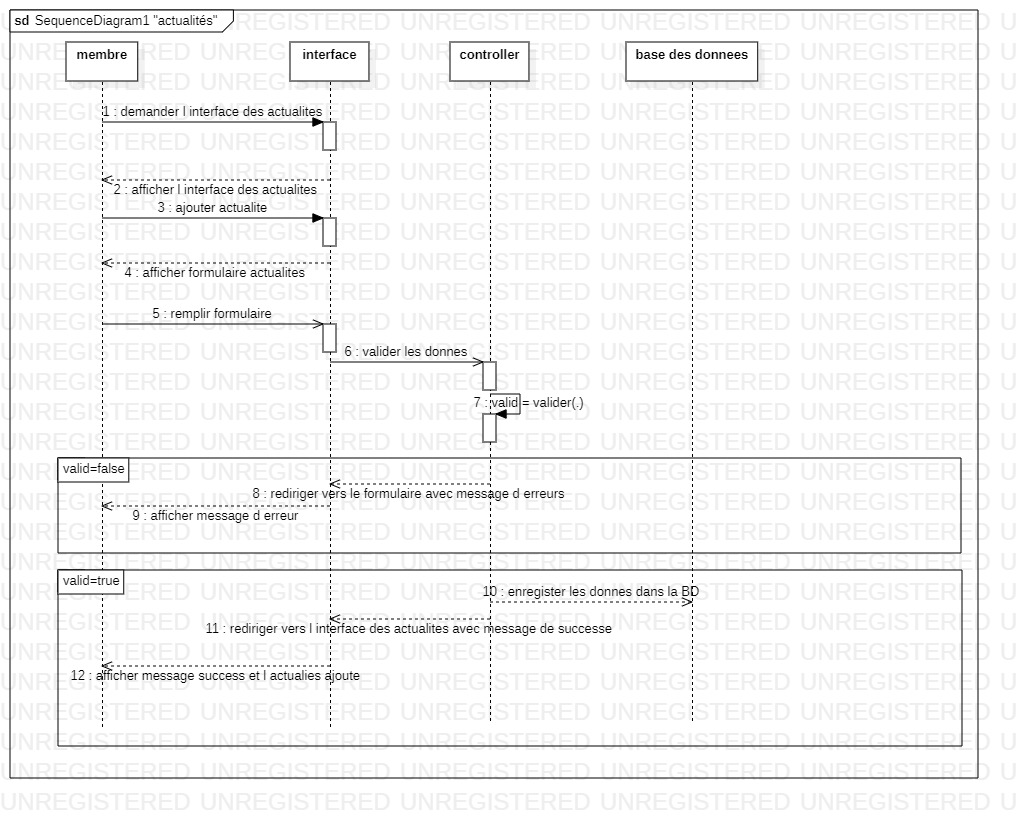
**Tableau 3.** Description textuelles de diagramme de séquence "connexion"

|  |  |
| --- | --- |
| Précondition | L’opération doit être demandée. |
| Scénario nominal | Le membre doit remplir les champs de saisis ;  Le système vérifie les champs non vides ;  Le membre valide par « se connecter » ;  Le système vérifie l’existence des données saisis ;  Si les données existent, l’espace sera disponible sous le nom d’utilisateur. |
| Scénario alternatif | Champs invalide : un message s’affiche pour l’utilisateur de saisir à nouveau les données ;  Données invalides : un message s’affiche pour l’utilisateur de bien vérifié les données saisis. |
| Post condition | Authentification réussite |

## Diagramme de séquence « Ajouter Actualités »

Les actualités affiché sur la vitrine de site est gérer par l’administrateur qui a le droit d’ajouter, modifier ou supprimer une actualité.

Le diagramme de séquence de la Figure 11 détaille le déroulement de cette opération avec des différents scénarios.



**Figure 12.** Diagramme de séquence « Actualités »

Pour bien expliquer le diagramme de séquence « Actualités », on décrit textuellement ce dernier dans le tableau ci-dessous.

**Tableau 4.** Description textuelle de diagramme de séquence "Actualités"

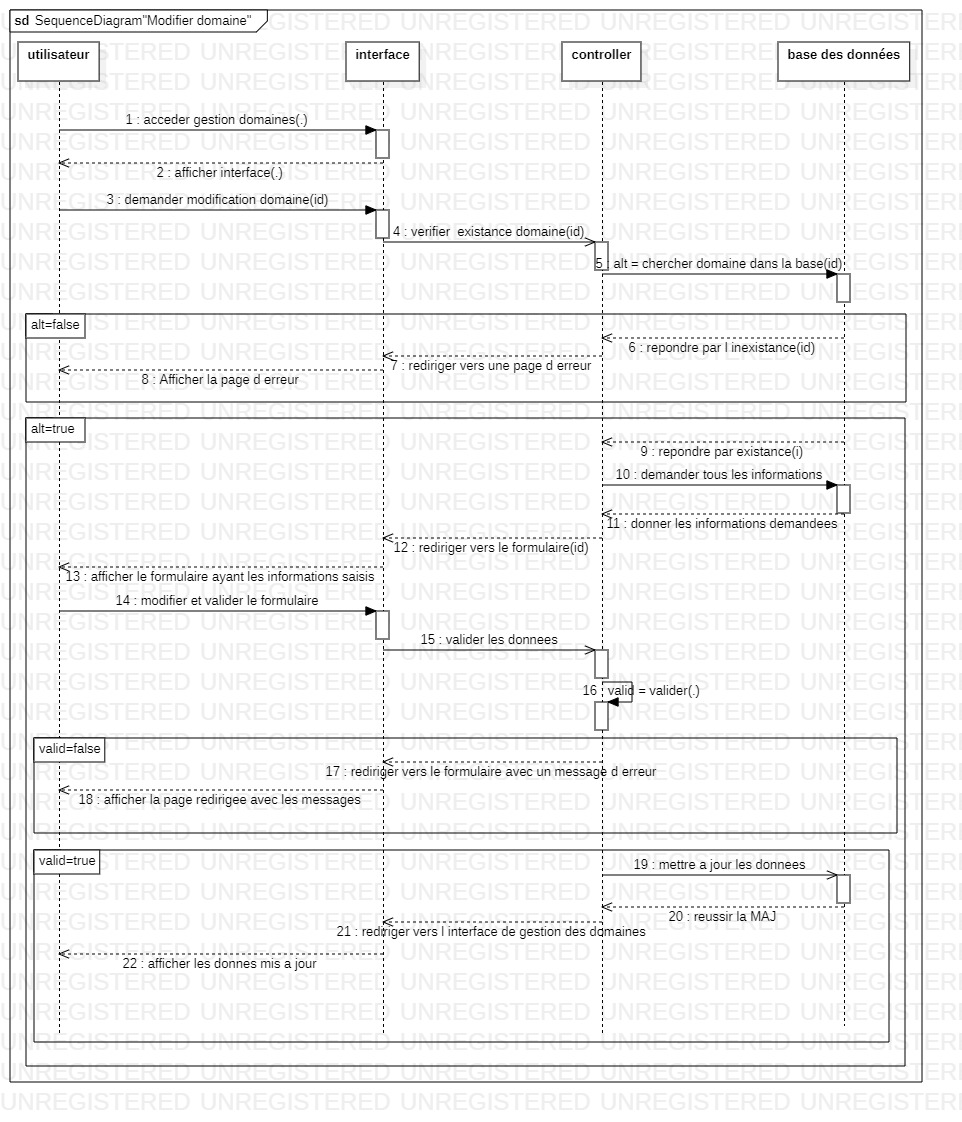
|  |  |
| --- | --- |
| **Précondition** | S’authentifier |
| **Scénario nominal** | L’administrateur accédé à son interface personnelle ;  Il accédé à l’interface de gestion des actualités ;  Il choisit d’ajouter une actualité ;  Il remplit le formulaire ;  Le système vérifie les champs remplis ;  Si les champs sont valides, un message de succès s’affiche à l’admin. |
| **Scénario alternatif** | Si les champs saisis sont invalide : le formulaire s’affiche avec des messages d’erreurs qui localisent où exactement l’erreur persiste. |
| **Post condition** | L’ajout de l’actualité est effectué avec succès. |

Dans notre projet on a opté ces scénarios dans presque tous les formulaires d’ajout. Du coup toutes les demandes d’ajout auront le même diagramme de séquence de celle d’Actualités.

## Diagramme de séquence « Modification d’un domaine »

Dans la partie ci-dessus on a découvert les scénarios qui se déroulent lors d’un ajout de n’importe quels services, toutefois ces scénarios ne rassemblent pas aux demandes de modifications cependant la même interface s’affiche à l’utilisateur avec des données pré saisis.

Bref, on révèle dans le diagramme figuré ci-dessous les scénarios qui peuvent être exposé lors d’une demande de modification (Figure 13).



**Figure 13.** Diagramme de séquence "Modification domaine"

On détaillera par la suite dans un tableau (Tableau 5) les détails de l’action de modification

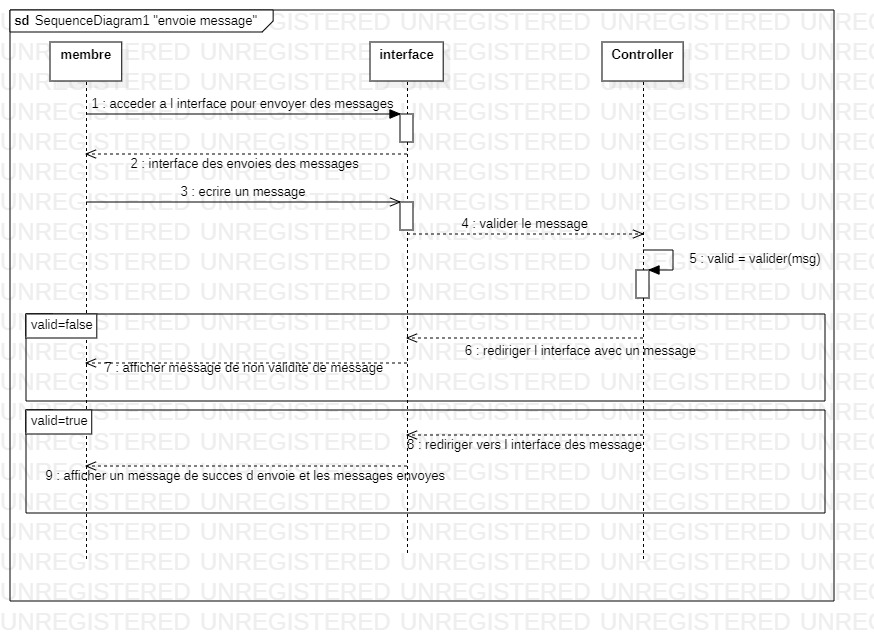
**Tableau 5.** Description textuelle de diagramme de séquence "Modification d'un domaine"

|  |  |
| --- | --- |
| **Précondition** | S’authentifier. |
| **Scénario nominal** | L’administrateur accédé à son interface personnelle ;  Il accédé à l’interface de gestion des domaines ;  Il choisit de modifier un domaine ;  Le système vérifie l’existence des données dans la base de données ;  Il met à jour le formulaire ;  Le système vérifie les champs remplis ;  Si les champs sont valides, un message de succès s’affiche à l’admin. |
| **Scénario alternatif** | Si les données n’existent pas dans la base des données : une page 404 not found s’affiche à l’utilisateur ;  Si les nouveaux données saisis par l’utilisateur sans invalide : l’application se redirige vers le formulaire à nouveau avec un message d’erreur et le type d’erreur. |
| **Post-condition** | Le domaine est mis à jour avec succès. |

## Diagramme de séquence « Envoie d’un message »

Parmi les fonctionnalités du membre, il peut envoyer des messages aux administrateurs de site qui vont la gérer par la suite.

Si on cherche dans les détails et les dimensions temporelles de l’envoie de message, on trouve ce diagramme de séquence (Figure 14).



**Figure 14.** Diagramme de séquence "Envoie des messages"

On résume la figure 14 dans le tableau ci-dessous :

**Tableau 6.** Description textuelle de diagramme de séquence "Envoie des messages"

|  |  |
| --- | --- |
| **Pré condition** | S’authentifier. |
| **Scénario nominal** | Le membre accède à son interface membre ;  Il choisit l’interface pour envoyer un message ;  Il remplit le champ de saisie de message ;  Le système valide le message ;  Si le message est valide ; un message de succès est envoyé au membre. |
| **Scénario alternatif** | Si le message est invalide, l’interface d’envoie des messages s’affiche de nouveau avec un message d’échec. |
| **Post condition** | L’envoie de message a été effectué avec succès. |

Ces scénarios se répètent toute fois l’administrateur ou le membre veut mettre à jour un service.

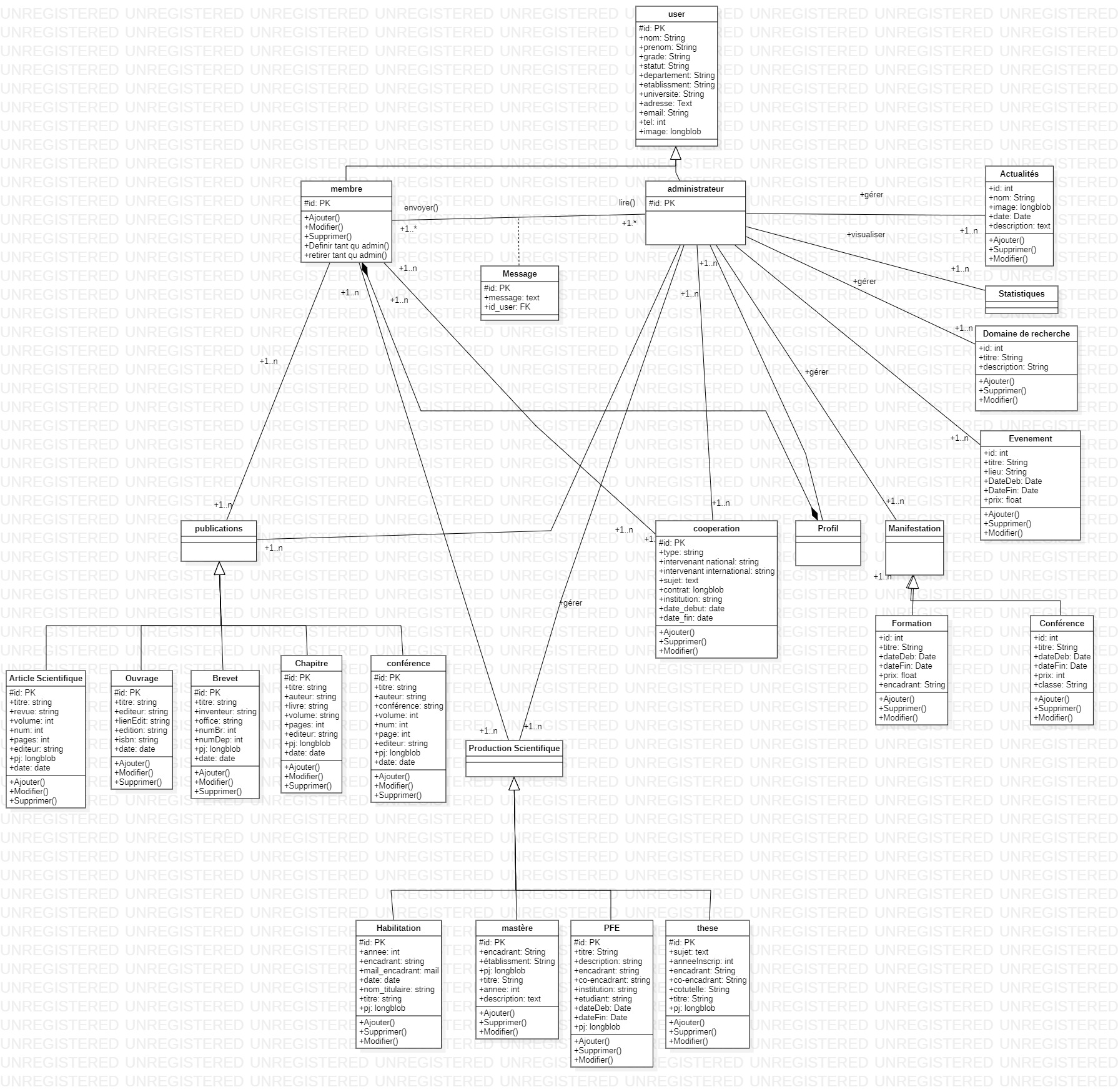
# Diagramme de classe

Un diagramme de classe est un type de diagramme UML qui décrit un système en visualisant les différents types d’objets au sein d’un système et les types de relations statiques qui existent entre eux. Il illustre également les opérations et les attributs des classes.

Ils sont généralement utilisés pour explorer les concepts de domaine, comprendre les exigences logicielles et décrire les conceptions détaillées [[6].](#_WEBOGRAPHIE)

Une classe est un ensemble de fonctions et de données (attributs) qui sont liées ensemble par un champ sémantique. Les classes sont utilisées dans la programmation orientée objet. Elles permettent de modéliser un programme et ainsi de découper une tâche complexe en plusieurs petits travaux simples [[7]](#_WEBOGRAPHIE) .

Après une étude profonde de notre projet, on a arrivé à élaborer ce diagramme de classe présentant les classes et la relation utile pour la réalisation qui constitue notre base de données.



**Figure 15.** Diagramme de classe de l'application

Notre diagramme contient globalement 23 classes, parmi eux, on a quatre (4) super classes notant la classe utilisateur qui a comme attribut un id, nom, prénom, email, mot de passe,.. et possède comme sous classes un administrateur et un membre associé à une classe Message qui a comme attribut l id (clé primaire), un clé secondaire est l’id de l’utilisateur et le contenu de message.

Aussi on a la super classe publication qui a cinq (5) sous classes, article, ouvrage, chapitre d’ouvrage, brevet, et conférence. Comme exemple la sous classe Article ayant les attributs suivants : id comme un clé primaire, titre, édition, éditeur.. et en ce qui concerne les méthodes on a la méthode Ajouter, Supprimer et Modifier. L’autre super classe sous le nom productions spécifique ayant Habilitation, Thèse, Master et PFE comme des sous classes et ont Ajouter , Supprimer et Modifier comme opérations communs et la dernière super classe est la manifestation a deux sous classes Formation qui a pour attributs id comme un clé primaire, titre, date début, date de fin, prix et encadrant et comme opérations Ajouter, Supprimer et Modifier.

Dans notre diagramme on a quatre (4) autres classes ; classe statique qui peut être visualisé par un ou plusieurs administrateur de site de même pour les classes évènement, actualité, domaine de recherche ; un ou plusieurs administrateurs peuvent gérer un ou plusieurs éléments de ces classes.

Encore, un ou plusieurs soit membre ou administrateur peuvent gérer un ou plusieurs soit publication, soit productions scientifique ou coopération.

Pour la gestion de profil, seul un membre peut gère son profil par contre le profil d’un membre peut être gérer par un ou plusieurs administrateur.

# Conclusion

Tout au long de ce chapitre, on a détaillé la conception de notre site web à travers ses différentes fonctionnalités. Ce chapitre a mis en évidence la méthode de travail qu’on a adopté, ainsi, on a exposé la modélisation conceptuelle de la plateforme suivant l’approche UML, ce qui nous a permis de bien approfondir et détailler les solutions par l’élaboration de ces diagrammes (les diagrammes des cas d’utilisation, le diagramme des classes et les diagrammes des séquences).

Le prochain chapitre sera dédié au détail de la réalisation du notre projet

# Chapitre III : Réalisation

# Introduction

# Environnement de développement

## Environnement matériel

Afin de réaliser notre projet, j’ai utilisé un ordinateur « ASUS » :

Tableau . Environnement matériel

|  |  |
| --- | --- |
| **Marque** | **Description** |
| ASUS | Processeur : Intel(R) Core(TM) i5-7200U CPU ;  RAM : 8GB ;  Disque dur : 480 SSD ;  Système d’exploitation : Windows 10 |

## Environnement logiciel

### HTML 5

HTML5 spécifie deux syntaxes d'un modèle abstrait défini en termes de DOM : HTML5 et XHTML5. Le langage comprend également une couche application avec de nombreuses API, ainsi qu'un algorithme afin de pouvoir traiter les documents à la syntaxe non conforme. Le travail a été repris par le W3C en mars 2007 après avoir été initié par le WHATWG. Les deux organisations travaillent en parallèle sur le même document afin de maintenir une version unique de la technologie. [[8]](#_WEBOGRAPHIE)



Figure . Logo HTML5

### CSS 3

Le CSS pour Cascading Style Sheets, est un langage informatique utilisé sur Internet pour la mise en forme de fichiers et de pages HTML. On le traduit en français par feuilles de style en cascade. Apparu dans les années 1990, le CSS se présente comme une alternative à la mise en forme via des balises, notamment HTML. Un peu plus complexe à maîtriser, il permet un gain de temps considérable dans la mise en forme d'une page web par rapport à ces balises. Grâce au CSS, vous pouvez en effet appliquer des règles de mise en forme (titrage, alignement, polices, couleurs, bordures, etc.) à plusieurs documents simultanément. [[9]](#_WEBOGRAPHIE)



Figure . Logo de CSS3

### JavaScript

JavaScript désigne un langage de développement informatique, et plus précisément un langage de script orienté objet. On le retrouve principalement dans les pages Internet. Il permet, entre autres, d'introduire sur une page web ou HTML des petites animations ou des effets. Créé en 1995 par Brendan Eich, en même temps que la technologie Java, le langage JavaScript se distingue des langages serveurs par le fait que l'exécution des tâches est opérée par le navigateur lui-même, sur l'ordinateur de l'utilisateur, et non sur le serveur web. Il s'active donc généralement sur le poste client plutôt que côté serveur. [[10]](#_WEBOGRAPHIE)



Figure . Logo de JavaScript

### JQuery

JQuery est une bibliothèque JavaScript gratuite, libre et multiplateforme. Compatible avec l'ensemble des navigateurs Web (Internet Explorer, Safari, Chrome, Firefox, etc.), elle a été conçue et développée en 2006 pour faciliter l'écriture de scripts. Il s'agit du Framework JavaScript le plus connu et le plus utilisé. Il permet d'agir sur les codes HTML, CSS, JavaScript et AJAX et s'exécute essentiellement côté client.[[11]](#_WEBOGRAPHIE)



Figure . Logo de JQuery

### Ajax

AJAX (Asynchrones JavaScript + XML) n'est pas une technologie en soi, mais un terme désignant une « nouvelle » approche utilisant un ensemble de technologies existantes, dont : HTML ou XHTML, CSS, JavaScript, DOM, XML.. et surtout l'objet XMLHttpRequest. Lorsque ces technologies sont combinées dans le modèle AJAX, les applications web sont capables de réaliser des mises à jour rapides et incrémentales de l'interface utilisateur sans devoir recharger la page entière dans le navigateur. Les applications fonctionnent plus rapidement et sont plus réactives aux actions de l'utilisateur. [[12]](#_WEBOGRAPHIE)



Figure . Logo d’AJAX

### Node JS

NodeJS est un environnement d’exécution permettant d’utiliser le JavaScript côté serveur. Grâce à son fonctionnement non bloquant, il permet de concevoir des applications en réseau performantes, telles qu’un serveur web, une API ou un job CRON. [[13]](#_WEBOGRAPHIE)

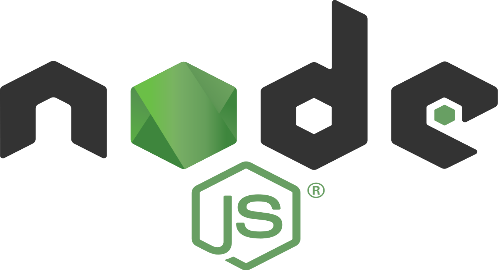


Figure . Logo de NodeJS

### Bootstrap5

Bootstrap est l’un des Frameworks les plus populaires pour le développement de sites Web. L’outil fournit des modèles pour CSS et HTML avec lesquels vous pouvez mettre en place une conception Web moderne de manière relativement simple. La mise en page, les polices, les boutons et les divers éléments de navigation peuvent être créés et positionnés très simplement. [[14]](#_WEBOGRAPHIE)



Figure . Logo de Bootstrap5

### Xampp

XAMPP est un ensemble de logiciels servant à mettre en place aisément un serveur Web, un serveur FTP et un serveur de messagerie électronique. C'est une distribution de logiciels libres (X Apache MySQL Perl PHP) offrant une bonne souplesse d'utilisation, reconnue pour son installation simple et rapide. Ainsi, il est à la portée de la plupart de personnes dans la mesure où il ne requiert pas de connaissances spécifiques et fonctionne, qui plus est, sur les dispositifs d'exploitation les plus communs. [[15]](#_WEBOGRAPHIE)



Figure . Logo de Xampp

### PhpMyAdmin

PhpMyAdmin est une application Web pour gérer les bases de données MySQL et MariaDB à l'aide d'une interface utilisateur graphique (GUI). Il permet aux utilisateurs de créer, modifier et supprimer facilement des tables et des champs ; ajouter, supprimer et modifier des données ; exécuter des instructions SQL et gérer les privilèges des utilisateurs.[[16]](#_WEBOGRAPHIE)



Figure . Logo de PhpMyAdmin

### MySQL

MySQL est une base de données relationnelle libre qui a vu le jour en 1995 et très employée sur le Web, souvent en association avec PHP (langage) et Apache (serveur web). MySQL fonctionne indifféremment sur tous les systèmes d'exploitation (Windows, Linux, Mac OS notamment). Le principe d'une base de données relationnelle est d'enregistrer les informations dans des tables, qui représentent des regroupements de données par sujets.[[17]](#_WEBOGRAPHIE)



Figure . Logo de MySQL

### Laravel 9

Laravel est un Framework PHP multiplateforme permettant de créer des applications web en utilisant un script plutôt qu’un langage PHP strict, il est principalement dédié pour le développement Backend bien qu’il offre des fonctionnalités de Frontend.

En effet, cet outil se distingue par sa supériorité par rapport aux autres frameworks web PHP. Il dispose d'outils de développement avancés et de fonctionnalités qui permettent le développement rapide d'applications Web. C'est l'un de ces frameworks qui possède de vastes fonctionnalités et une grande flexibilité, ce qui en fait une plateforme parfaite pour la création d'applications.

*Pourquoi Laravel ?*

Pour répondre à cette question, on devrait tout d’abord aborder aux avantages de Laravel.

De prime abord Laravel est facile à apprendre si on dispose d’une formation adéquate, après il simplifie le processus de développement, et ça est le raison pour lequel il est conçu, il permet aussi d’intégrer facilement des modules préfabriqués dans une application, à l’aide d’interfaces intuitives et expressives de ligne de commande et de Composer.

Quant à la sécurité, ce framework comprend un certain nombre de fonction de sécurité citons l’authentification des utilisateurs, l’autorisation des rôles des utilisateurs, la vérification des e-mails, le hachage des mots de passe.. De plus il simplifie la validation des données entrantes.

En dernier lieu, Laravel offre d’emblée d’importantes fonctionnalités de test, en fait il prend en charge les tests unitaires qui teste de petites sections isolées du code de l’application, ainsi que les tests de fonctionnalités, qui testent des sections plus importantes du code et des fonctionnalités de plus haut niveau.



Figure . Logo de Laravel 9

### Github

Github est un site web et un service de Cloud qui aide les développeurs à stocker et à gérer leur code, ainsi qu’à suivre et contrôler les modifications qui lui sont apportées. [[18]](#_WEBOGRAPHIE)

En fait il est considéré comme un service d’hébergement de projet en cours de développement en ce quel que soit le langage utilisé.

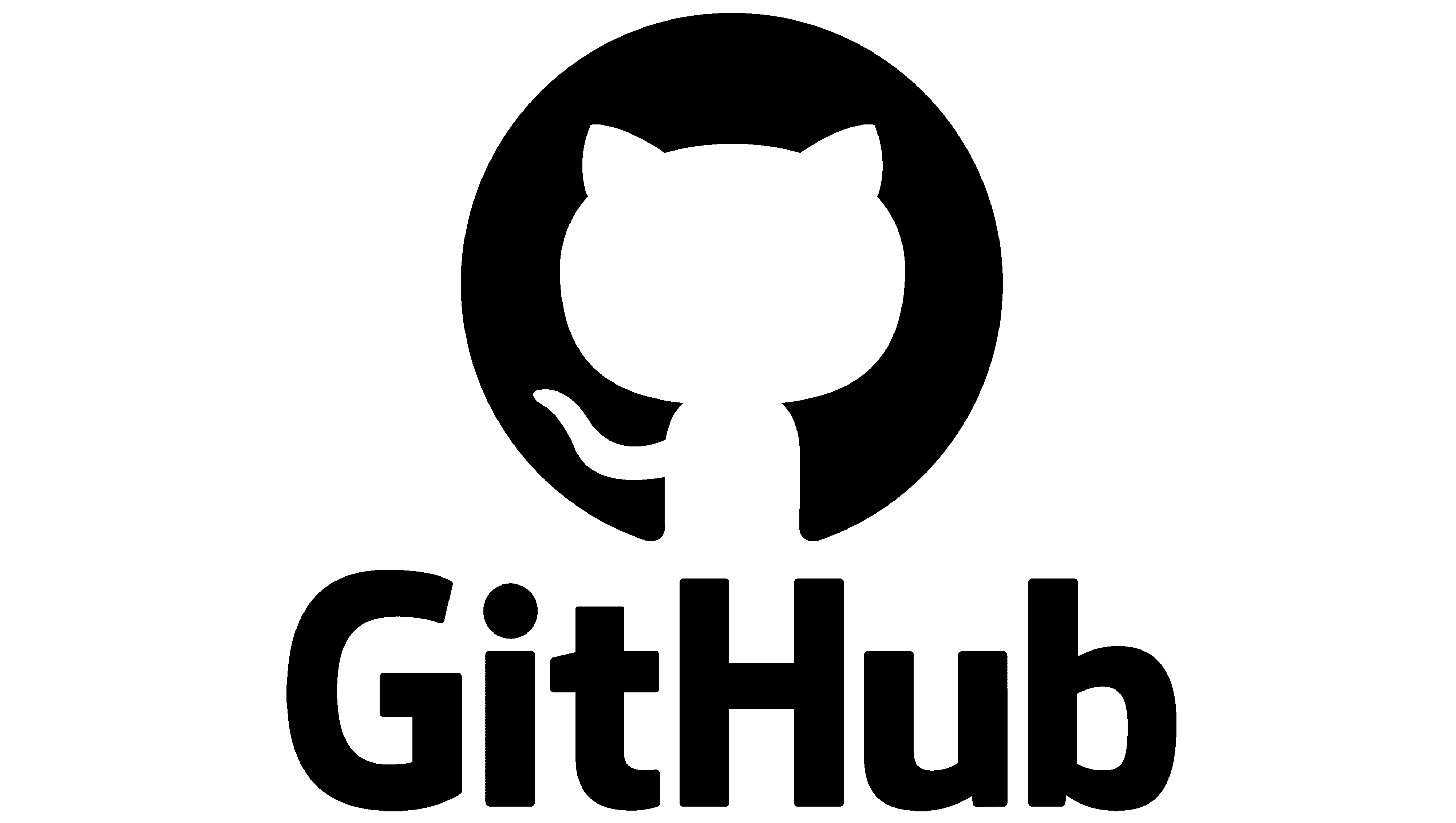


Figure . Logo de Github

### Visual Studio

Visual Studio Code est un éditeur de code source léger mais puissant qui s'exécute sur votre bureau et est disponible pour Windows, macOS et Linux. Il est livré avec un support intégré pour JavaScript, TypeScript et Node.js et dispose d'un riche écosystème d'extensions pour d'autres langages et runtimes (tels que C++, C#, Java, Python, PHP, Go, .NET). [[19]](#_WEBOGRAPHIE)

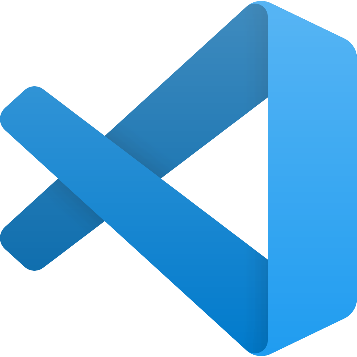


Figure . Logo de Visual Studio

# Architecture globale

Pour notre application web j’ai adopté l’architecture la plus populaire MVC (Model View Controller) grâce à sa robustesse dans ces genres d’applications. Cette architecture a pour objectif globale de séparer les aspects traitement, données et présentation et de définir les interactions entre ces trois aspects.

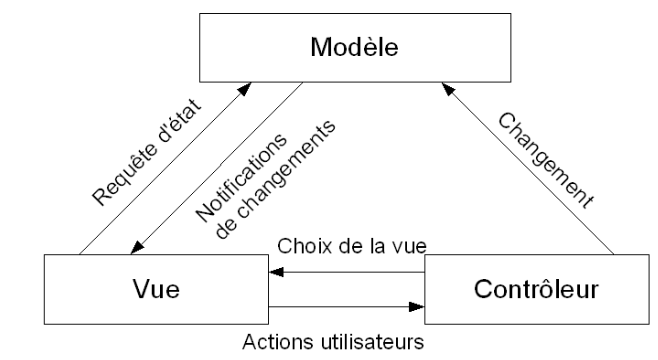


Figure . Architecture MVC

## Les composants de MVC

### Modèle

Un modèle contient les données utilisées par un programme. Il peut s’agir d’une base de données, d’un fichier ou d’un simple objet. Par exemple, un objet Client récupérera les informations de la base de données, les manipulera et mettra à jour ses données dans la base de données.

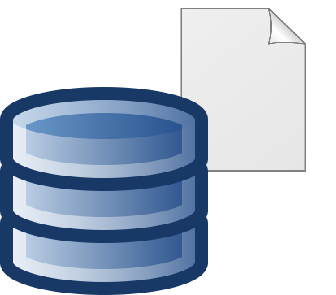


Figure . Modèle

### Vue

Une vue est un moyen d’afficher des objets dans une application. Par exemple, l’affichage d’une fenêtre ou des boutons ou d’un texte dans une fenêtre. Il comprend tout ce que l’utilisateur peut voir.

La vue est l’interface utilisateur. La vue permet à l’utilisateur d’afficher les données à l’aide d’un modèle et lui permet également de modifier les données.

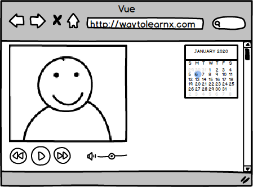


Figure .Vue

### Contrôleur

Les contrôleurs agissent comme une interface entre le modèle et la vue, pour traiter toute le logique métier et les requêtes entrantes, manipuler les données à l’aide du composant Modèle et interagir avec les Vues pour rendre le résultat final.

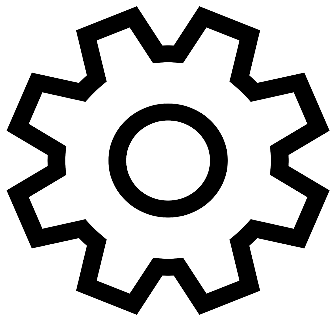


Figure . Contrôleur

# Réalisation

Dans la partie qui suit, on va donner un aperçu des diverses interfaces qui constitue notre portail web.

## Interface « Accueil »

L’interface illustrée par la figure 33 présente la première vue qu’un utilisateur consulte, elle expose une préface le laboratoire de recherche.



Figure . Interface « Accueil »

## Interface «Thème de recherche »

Cette vue représente les thèmes de recherche qui concerne le laboratoire (figure 34).



Figure . Interface « Thème de recherche »

## Interface « Publications »

|  |  |
| --- | --- |
|  |  |

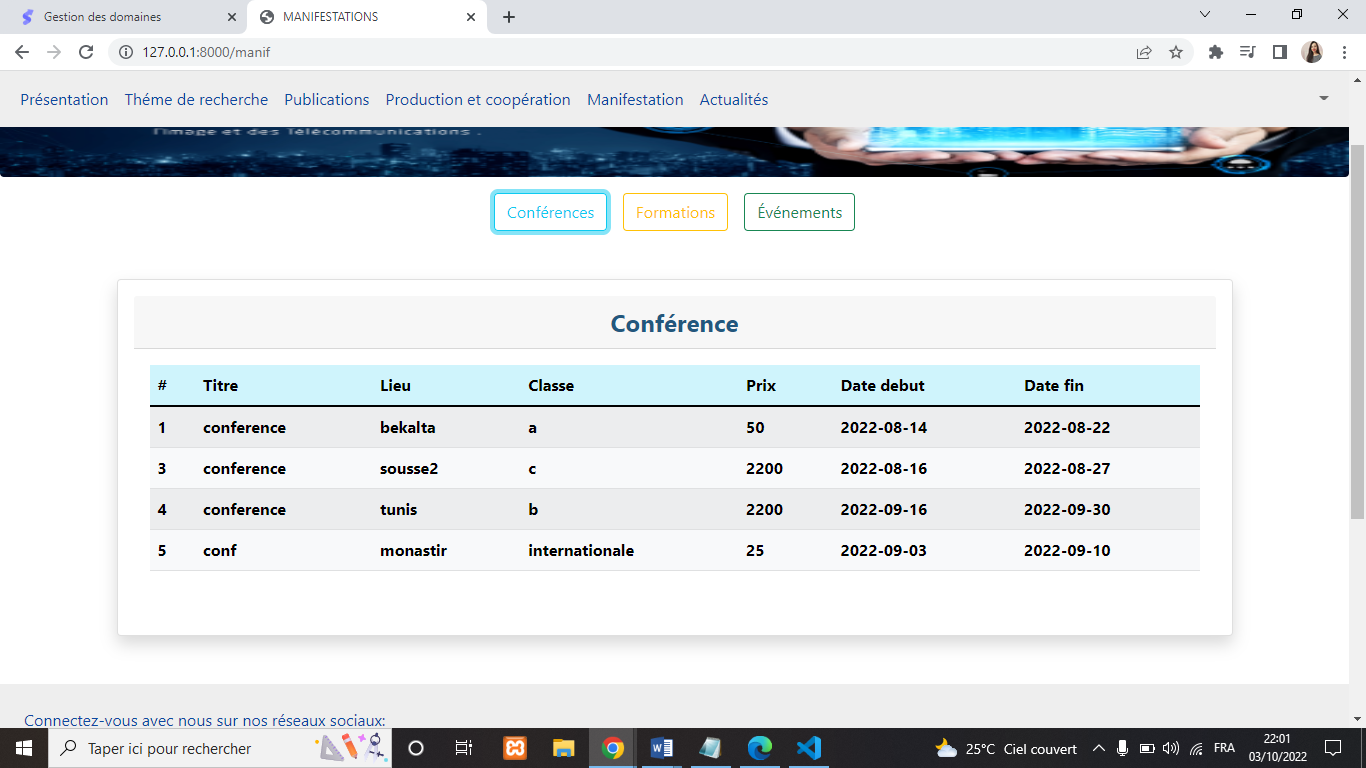
**Figure 35.** Interface « Publications »

## Interface « Production et coopération »

|  |  |
| --- | --- |
|  |  |

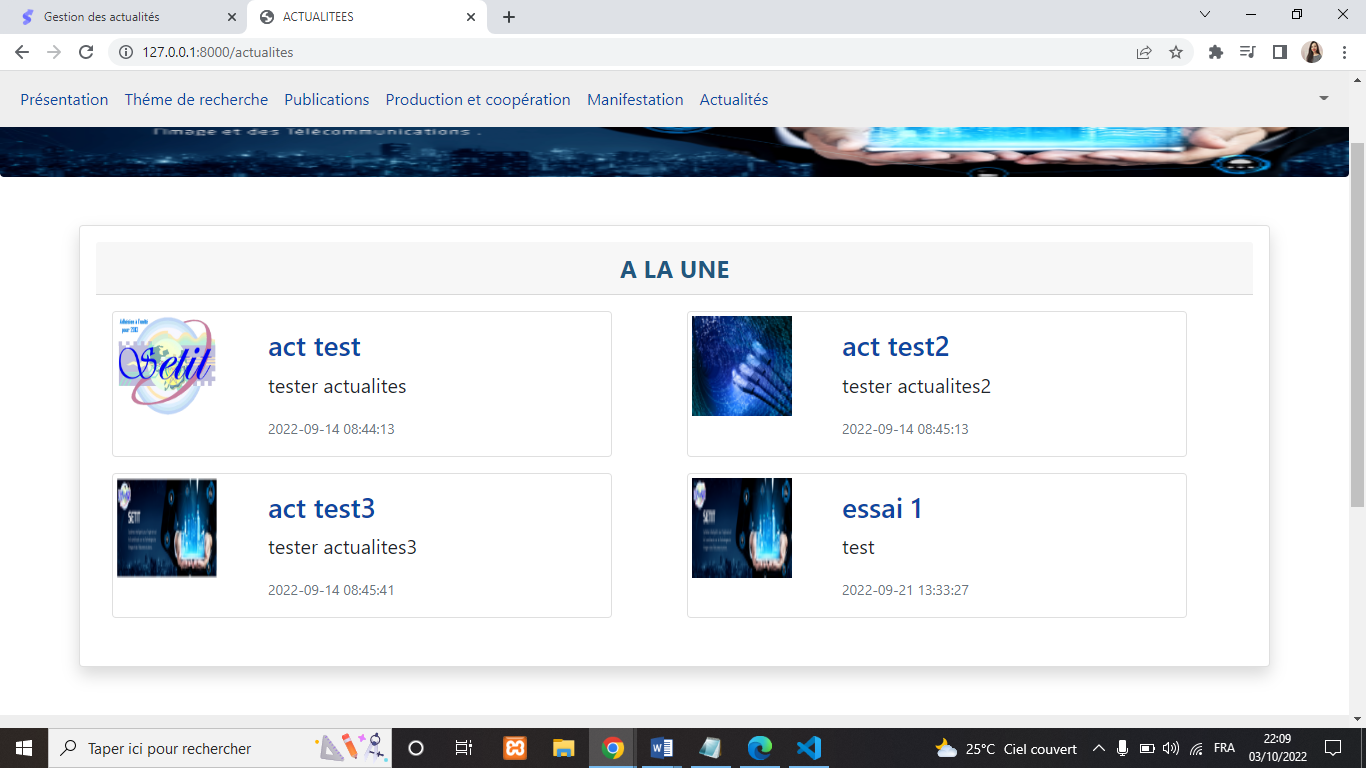
**Figure 36.** Interface « Production et coopérations »

## Interface « Manifestation »



**Figure 37.** Interface « Manifestation »

## Interface « Actualités »



**Figure 38.** Interface « Actualités »

## Interface « Inscription »



**Figure 39.** Interface « Inscription »

## Interface « Connexion »

# Building

# Conclusion

# 

# Conclusion générale

# WEBOGRAPHIE

[1] <http://www.setit.rnu.tn/URSETIT/> « consulté le 03/09/2022

[2] <https://www.futura-sciences.com/tech/definitions/informatique-uml-3979/> « consulté le 20/09/2022 »

[3] [Comprendre le modèle Cycle en V vs Agile en gestion de projet (appvizer.fr)](https://www.appvizer.fr/magazine/operations/gestion-de-projet/cycle-v) « consulté le 20/09/2022 »

[4] <https://www.ibm.com/docs/fr/rational-soft-arch/9.5?topic=diagrams-use-case> « consulté le 25/09/2022 »

[5] <https://www.ibm.com/docs/fr/rsm/7.5.0?topic=uml-sequence-diagrams> « consulté le 27/09/2022 »

[6] <https://creately.com/blog/fr/uncategorized-fr/tutoriel-sur-les-diagrammes-de-classe/> « consulté le 27/09/2022 »

[7] <https://www.ibm.com/docs/fr/rsar/9.5?topic=diagrams-class> « consulté le 28/09/2022 »

[[8]](#_HTML) <https://www.techno-science.net/glossaire-definition/HTML5.html> « consulté le

28/09/2022 »

[[9]](#_CSS) <https://www.journaldunet.fr/web-tech/dictionnaire-du-webmastering/1203277-css-cascading-style-sheets-definition-traduction/> « consulté le 1/10/2022 »

[[10]](#_JavaScript) <https://www.journaldunet.fr/web-tech/dictionnaire-du-webmastering/1203585-javascript/> « consulté le 1/10/2022 »

[[11]](#_JQuery) <https://www.journaldunet.fr/web-tech/dictionnaire-du-webmastering/1203587-jquery-definition/> « consulté le 1/10/2022 »

[[12]](#_Ajax) <https://developer.mozilla.org/fr/docs/Web/Guide/AJAX> « consulté le 1/10/2022 »

[[13]](#_Node_JS) <https://welovedevs.com/fr/articles/nodejs/> « consulté le 1/10/2022 »

[[14]](#_Bootstrap5) <https://www.ionos.fr/digitalguide/sites-internet/developpement-web/bootstrap-5-les-nouveautes/#:~:text=Bootstrap%20est%20l'un%20des,moderne%20de%20mani%C3%A8re%20relativement%20simple>. « Consulté le 1/10/2022 »

[[15]](#_Xampp) <http://www.standard-du-web.com/xampp.php> « Consulté le 1/10/2022 »

[[16]](#_PhpMyAdmin) <https://www.websiterating.com/fr/web-hosting/glossary/what-is-phpmyadmin/> « Consulté le 1/10/2022 »

[[17]](#_MySQL) <http://www.mosaique-info.fr/glossaire-web-referencement-infographie-multimedia-informatique/m-glossaire-informatique-et-multimedia/448-mysql-definition.html> « Consulté le 1/10/2022 »

[[18]](#_Github) <https://kinsta.com/fr/base-de-connaissances/base-de-connaissances-github/> «consulté le 29/08/2022 »

[[19]](#_Visual_Studio) <https://code.visualstudio.com/docs> «consulté le 29/09/2022 »

[[20]](#_Laravel_9)