

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 MASTER
Option : Génie Logiciel

Thème

Conception et réalisation d'une plateforme de
gestion du marché du BTP en Algérie basée sur
l'architecture REST

Réalisé par:

M. *CHETTOUT* Abdeslam
M. *HAMMOUCHE* Mohand Mouloud

Devant le jury composé de :

Examineur 1	Mme. KESSIRA Dalila
Examineur 2	Mme. HOUHA Amel
Encadrant	M. AKILAL Karim

2021/2022

Résumé

Les applications web se présentent de plus en plus comme une solution majeure dans la résolution de plusieurs problèmes, dans plusieurs domaines de la vie réel (gestion de ressources, commerce, publicité; services, divertissement... etc). Notre problématique se pose sur le manque de plateformes qui propose des services et produits liés au domaine du BTP, ainsi que la fiabilité et la facilité d'utilisations de ces derniers.

Binaa est une plateforme de gestion du marché du BTP en Algérie basée sur l'architecture REST visant à enlever l'ambiguïté sur le marché du BTP et à proposer une expérience accessible fiable et efficace, afin de permettre au client d'avoir des **services** et des **produits** plus compétitifs, et qui répondent plus facilement à leur exigence.

Ce mémoire décrit les étapes de la création de cette plateforme. nous commençant par l'énonciation de la problématique par la suite à proposer des solutions à cette problématique et finalement la réalisation de la plateforme.

Mots clés : *BTP, REST*

Abstract

Web applications are increasingly presenting themselves as a major solution in solving several problems, in several areas of real life (resource management, commerce, advertising; services, entertainment). Our problem arises on the lack of platforms that offers services and products related to the field of construction, as well as the reliability and ease of use of these.

Binaa is a construction market management platform in Algeria based on the REST architecture aimed at removing ambiguity in the construction market and offering a reliable and efficient accessible experience, in order to allow the customer to have **services** and more competitive **products**, which meet their requirements more easily.

This thesis describes the process of the creation of this platform. starting with the statement of the problem then proposing solutions to this problem and finally the realization of the platform.

Keywords : *BTP, REST*

Remerciements

Nos remerciements vont en premier lieu à nos parents qui ont su nous donner courage, volonté et patience afin d'achever ce projet ainsi que tous les projets de la vie.

*Nous tenons à remercier également **Mr. AKILAL Karim** notre promoteur, pour son aide, ses conseils, ses encouragements et ses orientations qui nous ont guidés jusqu'à l'aboutissement de ce travail.*

Merci aux membres du jury d'avoir accepté de juger notre travail et de nous faire part de leurs remarques pertinentes.

Nous exprimons notre plus grande gratitude à tous ceux qui ont contribué de près ou de loin à la réalisation de ce travail.

Dédicaces

Je dédie ce modeste travail :

À ma mère qui a œuvré pour la réussite, de par son amour, son soutien, ces sacrifices consentis et ses précieux conseils.

À mon père qui à toujours été disponible pour nous, et prêt à nous aider, je lui confirme mon attachement et mon profond respect.

À mon très cher frère.

CHETTOUT Abdeslam

Dédicaces

Je dédie ce modeste travail :

*À mes parent qui mon soutenue et encourager toutes ces années d'études.
Qu'ils trouvent ici le témoignage de ma profonde reconnaissance.*

*À mes frère et soeur
Pour leurs soutiens moral et leurs précieux conseils tout au long de mes études.*

*À toute ma famille, et mes amis, à mon binôme Abdeslam. et à tous ceux qui ont
contribué de près ou de loin pour que ce projet soit possible, je vous dis merci.*

HAMMOUCHE Mohand Mouloud

Table des matières

Table des matières	iv
Liste des figures	v
Liste des tableaux	vi
Liste des abréviations	vii
Introduction générale	1
1 Généralité sur le système d'information et le développement web	3
1.1 Introduction	3
1.2 Généralité sur le système d'information	3
1.2.1 Définition d'un système d'information	3
1.2.2 Composants des systèmes d'information	3
1.2.2.1 Matériel	4
1.2.2.2 Logiciel	4
1.2.2.3 Télécommunications	4
1.2.2.4 Données	4
1.2.2.5 Ressources humaines	4
1.2.3 Fonctions et fonctionnement des systèmes d'informations	5
1.2.3.1 Collecter	5
1.2.3.2 Stocker	5
1.2.3.3 Traiter	5
1.2.3.4 Diffuser l'information	5
1.2.4 Type de système d'information	5
1.3 Généralité sur le développement web	6
1.3.1 Réseaux informatique	6

1.3.1.1	Web	6
1.3.1.2	Protocole HTTP et HTTPs	6
1.3.1.3	Architecture Client-Serveur	7
1.3.1.4	Architecture RestAPI	7
1.3.2	Développement web	8
1.3.2.1	Application web	8
1.3.2.2	Type de site Web	9
1.4	Conclusion	10
2	Étude préalable	11
2.1	Introduction	11
2.2	Présentation du sujet	11
2.3	Étude de l'existant	12
2.3.1	E-commerce	12
2.3.1.1	Type d'e-commerce	12
2.4	Étude des cas existantes	12
2.4.1	Ouedkniss	12
2.4.2	Sook BTP	13
2.4.3	Wadifny	14
2.4.4	Facebook Marketplace	15
2.5	Tableau comparatif	16
2.6	Conclusion	17
3	Conception	18
3.1	Introduction	18
3.2	Spécification des besoins	18
3.2.1	Besoins fonctionnels et non fonctionnels	18
3.2.1.1	Besoins fonctionnels	18
3.2.1.2	Besoins non fonctionnels	19
3.2.2	L'approche UP (Unified Process)	19
3.2.2.1	Processus unifié	20
3.2.2.2	Le processus 2TUP	20
3.2.2.3	Le langage UML	20
3.2.2.4	Les différents types diagrammes d'UML	21
3.3	Identification des acteurs	21
3.4	Les Diagrammes des cas d'utilisation	22

3.4.1	Diagramme global des cas d'utilisations	23
3.4.2	Les différents cas d'utilisation	24
3.4.3	Cas d'utilisation Gestion d'Annonces	25
3.4.3.1	Diagramme de cas d'utilisation Gestion d'Annonces	25
3.4.4	Cas d'utilisation validation d'annonces	26
3.4.4.1	Diagramme de cas d'utilisation validation d'annonces	26
3.4.5	Cas d'utilisation Evaluer Annonce	27
3.4.5.1	Diagramme de cas d'utilisation Consultation et d'évaluation d'annonce	27
3.4.6	Cas d'utilisation Signalement	28
3.4.7	Cas d'utilisation recherche annonce	28
3.4.7.1	Diagramme de cas d'utilisation recherche annonce	29
3.5	Diagramme de séquence	30
3.5.1	Diagramme de séquence Ajout d'annonce	30
3.5.2	Diagramme de séquence d'évaluation d'annonces	31
3.5.3	Diagramme de séquence de validation d'annonces	32
3.6	Diagramme de classe	33
3.7	Schéma relationnel	35
3.8	Conclusion	36
4	Réalisation	37
4.1	Introduction	37
4.2	Choix technique	37
4.2.1	Technologies du backend	37
4.2.1.1	NodeJs	38
4.2.1.2	NPM	38
4.2.1.3	ExpressJs	38
4.2.1.4	Sequelize	39
4.2.1.5	GraphQL	40
4.2.1.6	Mysql	40
4.2.2	Technologies du frontend	41
4.2.2.1	VueJs	41
4.2.2.2	Vuetify	41
4.2.2.3	Axios	42
4.3	Outils de développement	42
4.3.1	Vs code	42

4.3.2	VueCLI	42
4.3.3	Git et Github	42
4.3.3.1	Git	42
4.3.3.2	GitHub	43
4.4	Quelques captures d'écran de l'application	43
4.4.1	Page de connexion	43
4.4.2	Page de création d'un nouveau compte	44
4.4.3	Page de dashboard	44
4.4.4	L'interface d'une annonce	45
4.4.5	Page de création d'une nouvelle annonce	45
4.4.6	page d'accueil	46
4.4.7	Évaluation d'une annonce	46
4.4.8	Catégorie	47
4.5	Conclusion	47
 Conclusion générale		 48
 Bibliographie		 50
 A Règles de passage du diagramme de classes au modèle logique		 53
A.1	Transformation des entités/classes	53
A.2	Transformation des associations	53
A.2.1	Associations un-à-plusieurs	54
A.2.2	Associations plusieurs-à-plusieurs	54
A.2.3	Associations un-à-un	54

Liste des figures

1.1	Architecture Rest API[1]	8
2.1	Capture page accueil OuedKniss.com	13
2.2	Capture page accueil sookbtp.com	14
2.3	Capture page accueil wadifny.com	15
2.4	Capture page accueil facebook.com/marketplace	16
3.1	La Méthode 2TUP [2]	20
3.2	Types de Diagrammes UML [3]	21
3.3	Diagramme de cas d'utilisation globale	23
3.4	Diagramme du cas d'utilisation Gestion d'Annonces	25
3.5	Diagramme du cas d'utilisation validation d'annonce.	26
3.6	Diagramme du cas d'utilisation Consultation et d'évaluation d'annonce.	27
3.7	Diagramme du cas d'utilisation recherche annonce.	29
3.8	Diagramme de séquence Ajout d'annonce.	30
3.9	Diagramme de séquence d'évaluation d'annonce.	31
3.10	Diagramme de séquence validation des annonces.	32
3.11	Diagramme de classe.	33
4.1	Page de connexion	43
4.2	Page d'inscription	44
4.3	dashboard	44
4.4	Exemple d'une annonce	45
4.5	Page de création d'une nouvelle annonce	45
4.6	Page d'accueil	46
4.7	Évaluer une annonce	46
4.8	Liste de Catégorie	47

Liste des tableaux

2.1	Comparaison entre les trois architectures	16
3.1	Le formalisme de description des cas d'utilisation.	24
3.2	Description des cas d'utilisation Gestion d'Annonces.	25
3.3	Description des cas d'utilisation validation d'annonces.	26
3.4	Description des cas d'utilisation Evaluer annonce.	27
3.5	Description des cas d'utilisation Signalement.	28
3.6	Description des cas d'utilisation recherche annonces.	28
3.7	Dictionnaire de données.	34

Liste des abréviations

BTP : Bâtiment et travaux publics.

UML : Unified Modeling Language.

HTML : HyperText Markup Language

HTTP : Hypertext Transfer Protocol

HTTPS : Hypertext Transfer Protocol Secure

API : Application Programming Interface

REST : Representational State Transfer

URL : Uniform Resource Locator

UP : Unified Process

2TUP : two track unified process

WWW : world wide web

Introduction générale

Le domaine du Bâtiment et travaux publics, ici en Algérie sur Internet sa présence est très négligeable. Ils existent quelques moyens tel que les réseaux sociaux et Ouedkniss qui permet aux particuliers et aux entreprises de mettre en avant des services et des produits liés au domaine du BTP. Réseaux sociaux et Ouedkniss se présentent d'une manière anarchique et sans possibilité de laisser des avis sur la qualité des services et des produits mis en avant. Avec une pauvre catégorisation qui ne pourrait certainement pas permettre de grandement faciliter les recherches des utilisateurs.

Que ce soit pour les grands projets qui nécessitent des entreprises pour les réaliser, ou pour les petits projets qui nécessitent un seul travailleur ou plusieurs, le client rencontre souvent voir toujours des problèmes lors de la recherche de la bonne personne à recruter, car il n'existe pas de plateforme ou d'organisme qui permettent de mettre en avant ce genre de services toutes en mettant en avant les plus compétents grâce à des systèmes d'évaluations et de recommandation. Un autre problème qu'on rencontre souvent aussi ici en Algérie, est celui lié aux prix. Si vous cherchez les prix de n'importe quel produit de BTP en ligne, vous ne trouverez pas grand chose. Et dans les boutiques locales et régionales les prix fluctuent grandement et souvent sans raisons valable. Des personnes pas liées au domaine du BTP, et qui ne connaissent rien au prix des produits, ils peuvent acheter des produits à des prix trop cher qu'il ne le devrait être, et des fois même d'une qualité moindre.

Notre projet consiste à concevoir une plateforme qui permet de résoudre la problématique soulevée ci-dessus, d'un côté mettre en avant les services et produits les mieux évalués par les utilisateurs, ainsi qu'à des prix raisonnables. En outre, proposer une meilleur catégorisation des différents services et produits, tout en proposant une recherche plus ciblée.

Notre mémoire est structuré en quatre chapitres comme suit :

- **Chapitre 1 "Définition et généralité sur les systèmes d'information et le développement Web"** : a pour objectif de présenter les différents composants et types des systèmes d'information, ensuite on va parler d'une manière générale sur le développement web en commençant par définir le web et l'Internet, ensuite on va passer à la définition du protocole HTTP et pour finir on va définir l'application web et on va introduire les deux termes back-end et front-end.
- **Chapitre 2 "Étude préalable"** : a pour but de présenter les applications e-commerce et ses différentes utilités, en démontrant que c'est la solution idéale pour la problématique posée dans l'introduction en introduisant la problématique d'une manière plus approfondie, et on va finaliser par une étude de cas de site similaire en Algérie, OuedKniss, Wadifny, SookBTP, .
- **Chapitre 3 " Conception "** : est consacré au cadre du projet, à l'analyse et la conception de notre plateforme à développer. Nous présentons ainsi les besoins fonctionnels et non fonctionnels et recensons les différents cas d'utilisation pour chaque acteur interagissant avec notre plateforme. Nous complétons cette modélisation par des descriptions textuelles et des diagrammes de séquences. Enfin, nous présentons le diagramme de classes de notre plateforme et le modèle relationnel qui lui est associé.
- **Chapitre 4 " Réalisation "** : dans ce dernier chapitre nous définissons les technologies utilisées et leurs avantages dans notre projet, ainsi que les outils utilisés pendant le processus de développement.

Enfin, nous clôturons ce mémoire par une conclusion dans laquelle nous résumons notre mémoire et nous exposons quelques perspectives futures.

Généralité sur le système d'information et le développement web

1.1 Introduction

Dans ce chapitre nous allons découvrir les systèmes d'information et leur différents aspects qui nous permettront de construire notre système d'information par rapport à notre application et des différents aspects autour des technologies web que nous allons utiliser durant le développement de notre application.

1.2 Généralité sur le système d'information

Dans ce qui suit, nous présentons quelque généralité sur le système d'information.

1.2.1 Définition d'un système d'information

Les systèmes d'information sont des composants interdépendants qui travaillent ensemble pour collecter, traiter, stocker et diffuser des informations pour soutenir la prise de décision, la coordination, le contrôle, l'analyse et la visualisation dans une organisation. [4].

1.2.2 Composants des systèmes d'information

Chaque système d'information est constitué de cinq composants clés : le matériel, les logiciels, les télécommunications, les personnes et les données. Le matériel fait référence aux éléments physiques du système d'information ; le logiciel est la programmation qui contrôle

le système d'information; la télécommunication transmet des informations à travers le système; les humains gèrent et interagissent avec le système d'information; et les données sont des informations stockées dans et traitées par le système.[5]

1.2.2.1 Matériel

Le composant matériel d'un système d'information comprend les éléments physiques du système, qui peuvent être touchés et sentis. Ces mécanismes, équipements et câblages permettent à des systèmes tels que les ordinateurs, les smart-phones, et les tablettes de fonctionner.

Les périphériques d'entrée et de sortie tels que les souris, claviers, imprimantes et moniteurs sont des éléments technologiques essentiels qui permettent aux humains d'interagir avec les ordinateurs et d'autres systèmes d'information. et d'autres composants tels que les microprocesseurs, les disques durs, les blocs d'alimentation, permettent également aux ordinateurs de stocker et de traiter des données.[5]

1.2.2.2 Logiciel

Ce sont les programmes utilisés pour organiser, traiter et analyser les données.[5]

1.2.2.3 Télécommunications

Différents éléments doivent être connectés les uns aux autres, permettent et facilitent la circulation de l'information dans l'organisation.[6]

1.2.2.4 Données

Données sont des faits et des chiffres bruts qui ne sont pas organisés et qui sont ensuite traités pour générer des informations.[6]

1.2.2.5 Ressources humaines

Le personnel est un élément des plus important à tout système d'information, le personnel regroupe tous les utilisateurs que ce soit les utilisateurs finaux ou les gestionnaires de système d'information, l'un sollicite ces services et l'autre assure son bon fonctionnement.[6]

1.2.3 Fonctions et fonctionnement des systèmes d'informations

Il existe quatre fonctions principales d'un systèmes d'information : collecter, stocker, traiter, et diffuser l'information.[7]

1.2.3.1 Collecter

Pour qu'un système fonctionne, il doit être alimenté. Les informations collectées proviennent de sources internes ou externes, et son traitées et diffusées plus tard. Les informations peuvent être stockées sous forme numérique ou sur un support physique.

1.2.3.2 Stocker

Une fois l'information collectée, il faut garantir un stockage durable et fiable et il peut être sous différente forme tel que les base de données ou fichiers.[7]

1.2.3.3 Traiter

Pour être exploitable, l'information subit des traitements. et cela consiste à produire de nouvelles informations à partir d'informations existantes grâce à des programmes informatiques ou des opérations manuelles(c'est de moins en moins souvent le cas). Le traitement de l'information peut prendre différentes formes, comme consultez juste l'information de base sans faire aucune modification, organiser l'information en fonction des critères spécifiques, modifier l'information ou supprimer les informations non pertinentes.[7]

1.2.3.4 Diffuser l'information

Pour qu'une information aie de la valeur, elle doit être transmis à la bonne personne, organisme dans les meilleurs délais.

1.2.4 Type de système d'information

Il existe une multitude de système d'informations en fonctions de leur utilité et parmi eux :

- **Systèmes de traitement des transactions** : qui a but de traiter les données résultant des occurrences de transactions **commerciales**. Leurs objectifs sont de fournir des transactions afin de mettre à jour les enregistrements et de générer des rapports (ex : système de facturation, système de paiement, système de gestion des stocks).

- **Systèmes d'information de gestion** : a but de gérer un ensemble de données disponible. Les informations sont traitées et converties sous une forme résumée pour l'utilisateur, généralement sous forme de rapport (ex : systèmes de gestion des achats et ventes, système de gestion des ressources humaines).
- **Systèmes d'aide à la décision** : Le système d'aide à la décision est un système d'information interactif qui fournit des informations, des modèles et des outils de manipulation de données pour aider à la prise de décision dans une situation semi-structurée et non structurée.
- **Système d'experts** : Les systèmes experts incluent l'expertise afin d'aider les gestionnaires à diagnostiquer des problèmes ou à résoudre des problèmes [8]

1.3 Généralité sur le développement web

1.3.1 Réseaux informatique

Un réseau informatique peut être décrit comme étant un ensemble d'appareils électroniques qui échangent des informations entre eux. L'internet conçu dans les années soixante est un réseaux informatique qui permettent à de nombreux ordinateurs de se connecter et de communiquer entre eux à l'échelle mondiale.[9]

1.3.1.1 Web

Le world wide web (WWW) communément appelé le web, créé dans les années quatre-vingt, est un système d'information mondial basé sur Internet. Le Web est un système de documents hypertextes inter-connectés, qui permet d'échanger des information sur Internet en communiquant via le protocole HTTP.[9]

1.3.1.2 Protocole HTTP et HTTPS

L'Hypertext Transfer Protocol, généralement abrégé HTTP est un protocole de communication client-serveur, il est la base de tout échange de données sur le web. HTTP est un protocole de la couche application conçu pour transférer des informations entre des appareils connectés en réseau.

HTTPS ou HyperText Transfer Protocol Secure est une version plus avancée et sécurisée de l'HTTP, ce qui signifie que les données échangées entre le client et le serveur sont chiffrées et ne peuvent en aucun cas être espionnées ou modifiées.

HTTP fonctionne comme un protocole de requête-réponse entre un client et un serveur, et pour cela HTTP dispose de plusieurs méthodes, et les plus utilisées d'entre elles sont GET, POST, PUT, DELETE.

- **GET** : est utilisée pour lire (ou récupérer) une représentation d'une ressource depuis le serveur.
- **POST** : est le plus souvent utilisé pour créer une nouvelle ressource sur le serveur.
- **PUT** : est utilisé pour modifier (mettre à jour) une donnée sur le serveur.
- **DELETE** : est utilisé pour supprimer une donnée sur le serveur.

1.3.1.3 Architecture Client-Serveur

Les applications web fonctionnent selon l'architecture client/serveur, qui représente l'environnement dans lequel des applications de machines type clientes communiquent avec des applications de type serveurs.[9]

- **Client** : est celui qui demande un ou des services.
- **Serveur** : est celui qui fournit les services demandés.

1.3.1.4 Architecture RestAPI

REST ou REpresentational State Transfer est un style architectural qui définit un ensemble de contraintes à utiliser pour créer des services Web. L'API (Application Program Interface) REST est un moyen d'accéder aux services Web de manière simple et flexible sans aucun traitement.

REST fonctionne au-dessus du protocole HTTP. Il tire parti des capacités natives de HTTP, telles que GET, PUT, POST et DELETE. Lorsqu'une requête est envoyée à une API RESTful, la réponse (la "représentation" de la "ressource" d'information recherchée) est renvoyée sous format JSON, XML ou HTML. Une API RESTful est définie par une adresse Web, ou URI (Uniform Resource Identifier), qui suit généralement une convention de dénomination.[10]

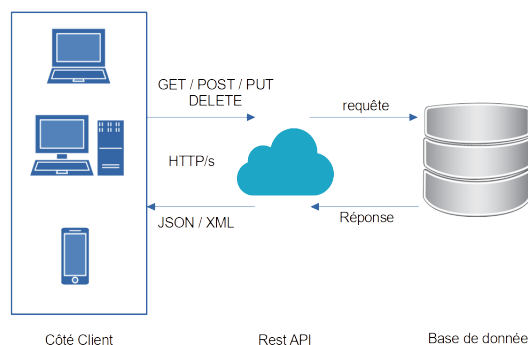


FIGURE 1.1 – Architecture Rest API[1] .

1.3.2 Développement web

Le développement de sites web, est le travail impliqué dans la création, la construction et à la maintenance de sites et d'applications web hébergés sur intranet ou Internet. Mais cela peut inclure aussi la conception, la publication, la programmation, et la gestion de base de données.

1.3.2.1 Application web

Une application web est une interface web applicative disponible uniquement sur le web et accessible via un navigateur internet. C'est une application qui peut être hébergée en cloud ou sur des serveurs dédiés. Toutes les données sont stockées sur un serveur web.[11]

Une application web se compose de deux parties comme on a précisé en haut, une partie front-end et une partie back-end.

- **Back-end :**

Back-end c'est la notation qui englobe tout ce qui s'exécute en côté serveur. Le coté serveur se compose de matériels (hardware), et de logiciels qui s'exécute sur ce dernier. Le serveur physique (hardware) qui est un ordinateur défini par un ensemble de ressources (RAM, processeurs, espace de stockage) et qui exécute un système d'exploitation tel que Windows et Linux, et permet de déployer un serveur (software) tel que Apache et Nginx, qui expose un port ou plusieurs ports sur Internet. Grâce à un URL (Uniform Resource Locator), les fichiers tel que les fichiers HTML et autres tel que des fichiers JavaScript, CSS, Images, vidéo, audio..etc exposées par le serveur et sont accessibles par le côté front-end (client).

D'autre part, le backend se compose généralement de trois parties : un serveur, une application et une base de données [12], parfois la partie serveur et application se fusionne en une partie, et c'est l'avantage d'utiliser quelque technologies et langages tel que NodeJs (le fameux framework ExpressJs), et dans d'autres cas le serveur et l'application sont séparés, c'est le cas pour l'utilisation du langage PHP qui exige d'avoir un serveur comme Apache à part qui permet d'exécuter les script PHP et les renvoyer le résultat de la requête au client. Dans les deux cas, l'application a besoin de stocker les information qui peuvent être représentées sous différents format (texte, images, binaire...), et pour cela on utilise des base de données et des SGBD (systèmes de gestion de base de données) tel que Mysql, MariaDB..etc.

- **Front-end :**

Dans le contexte Web, le front-end est la partie que l'utilisateur voit et interagit avec. Lors de la conception et le développement d'une telle interface web frontale, certains outils et technologies doivent être utilisées, qui sont généralement une combinaison de HTML, CSS et JavaScript étant tous contrôlés par le navigateur.[12]

Il existe plusieurs navigateurs Web qui permettent de surfer sur le net comme Google Chrome et Mozilla FireFox. Tous les navigateurs existants sont basés sur les trois technologies suivantes : HTML, CSS, JavaScript.

HTML (Hypertext Markup Language), comme défini dans le dictionnaire Larousse, est un langage de description de documents servant à présenter des pages Web et à préciser à l'aide de balises les liens hypertextes avec d'autres documents. HTML représente la squelette de la page web contenant le contenu sous format texte. D'autre part CSS (Cascading Style Sheets) ou feuilles de style en cascade, est utilisé pour décrire la présentation d'un document HTML ou XML, autrement dit CSS décrit comment les éléments doivent être présentés visuellement (la couleur, la taille... etc).[13]

D'après le dictionnaire Larousse, Javascript est un langage de programmation utilisé pour écrire des appelettes incorporables à des documents au format HTML. En gros, Javascript décrit le comportement de la page web.

1.3.2.2 Type de site Web

Il existe deux types de sites web :

- **Site web statique** : est une application Web qui peut être livrée directement au navigateur d'un utilisateur final sans aucune modification côté serveur du contenu HTML, CSS ou JavaScript.
- **Site web dynamique** : sont plus complexes sur le plan technique. Ils utilisent des bases de données pour charger des informations et ces informations sont mises à jour chaque fois que l'utilisateur effectue une action qui provoque une mise à jour de la base de données.

1.4 Conclusion

Il existe une grande diversité d'aspect et de technologie Web, conçu pour différents buts. Dans ce chapitre nous avons découvert quelque aspect essentiel de web et comment un site web fonctionne. On a commencé par la définition des systèmes d'information, en passant par ses différentes composants et fonctionnalités, ensuite on a introduit quelques notions tel que les réseaux informatiques, le Web, HTTP...etc, et on a terminé par la définition de frontend et backend en expliquant la relation entre ces deux derniers.

Étude préalable

2.1 Introduction

Dans ce chapitre nous allons définir le périmètre du projet , En étudiant le concept des sites e-commerce qui forment une partie importante de la solution, et on va faire une étude des applications web e-commerce existantes qui offrent des services ou des produits dans le domaine du BTP, tel que OuedKniss, et d'autres sites Web.

2.2 Présentation du sujet

Il existe plusieurs plateformes en Algérie qui offrent des services dans le domaine du BTP, tels que Ouedkniss, Sook BTP ,Wadifny.

Le sujet de notre mémoire consiste en la réalisation d'une application web de type e-commerce dans le but de résoudre la problématique exposée dans l'introduction générale qui se résume dans les points suivants :

- Manque de mécanisme d'évaluation des services et produits BTP en Algérie.
- Ambiguïté du marché du BTP comme les prix et la qualité.

Dans cette étude de l'existant nous allons présenter quelques plateformes et quelques une de leurs fonctionnalités qui nous intéressent en spécifiant leurs avantages et désavantages par rapport à la résolution de notre problématique.

2.3 Étude de l'existant

2.3.1 E-commerce

Le e-commerce ou commerce électronique désigne le fait de vendre et d'acheter des biens et des services ou la transmission de fonds ou de données par l'intermédiaire des réseaux informatiques, notamment Internet. Ces transactions commerciales se produisent soit en tant qu'entreprise à entreprise (B2B), entreprise à consommateur (B2C), consommateur à consommateur (C2C) ou consommateur à l'entreprise (C2B).[14]

2.3.1.1 Type d'e-commerce

- **Entreprise à entreprise (B2B)** : fait référence à une transaction entre entreprise que ce soit la vente ou l'achat de biens, de services ou d'informations.
- **Entreprise à consommateur (B2C)** : fait référence à la vente de biens et de services au grand public par les entreprises.
- **Consommateur à consommateur (C2C)** : fait référence aux systèmes destinés à supporter les échanges de consommateur à consommateur. En vendant ou achetant des produits créés ou fabriqués par eux même ou des produits d'occasion qu'il souhaitent revendre.
- **Consommateur à l'entreprise (C2B)** : fait référence à des consommateurs vendant leur service ou leurs biens à des entreprises.[14]

2.4 Étude des cas existantes

2.4.1 Ouedkniss

Ouedkniss est une plateforme algérienne qui offre beaucoup de services tels que les publicités payantes et les annonces de produits de différentes catégories tels que les véhicules, l'immobilier, les offres d'emplois, des services (transport, bâtiment, réparation... etc), Organisation de voyages, création des boutiques en ligne et qui cible divers types d'utilisateurs (B2B et B2C). Ouedkniss est le site le plus populaire en Algérie.

À cause de la diversité des services et catégories que Ouedkniss offre, les fonctionnalités qu'il offre sont sommaires. Tous les services du BTP sont sous une catégorie générale qui s'appelle Service, et c'est une catégorie qui regroupe plusieurs domaines de services tels que les services de transport, de location, de réparation... etc. Dans ce cas, Ouedkniss ne

peut pas modéliser toutes les caractéristiques d'un service tel que les services du domaine du BTP car c'est différent par rapport au autres domaines. Ceci rend Ouedkniss pauvre dans la spécification des caractéristiques d'une catégorie spécifique.

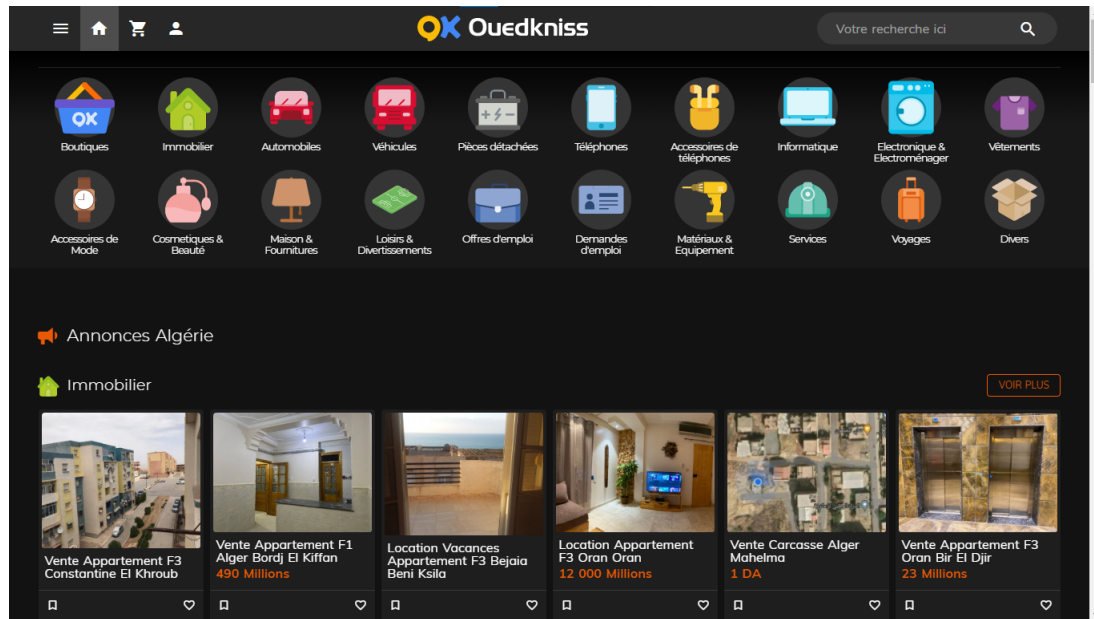


FIGURE 2.1 – Capture page accueil OuedKniss.com

2.4.2 Sook BTP

Sook BTP est le premier web Market dédié au domaine du BTP en Algérie[15] , Sook BTP est une application web qui met en relation des internautes en recherche d'achat et de vente de produits, matériels, matériaux, services, et offres d'emplois relatifs au secteur du BTP et ce dans le respect de la réglementation en vigueur.[16]

Contrairement à OuedKniss, Sook BTP se spécialise et se catégorise en BTP. Avec moins de popularité, depuis sa création en 2018 Sook BTP a eu environ 720 annonces. L'avantage de ce site est qu'il offre la possibilité d'ajouter des annonces de type "Demande" (pour demander un service ou produit) et de type "Offert" (pour offrir un service ou produit), par ailleurs on remarque que les catégories ne sont pas bien claires et ne sont pas bien organisés ce qui pose un problème pour certain utilisateurs lors de la recherche et même lors de l'insertion d'une nouvelle annonce. L'autre problème qui nous concerne le plus est qu'il n'existe pas un système d'évaluation pour les annonces.

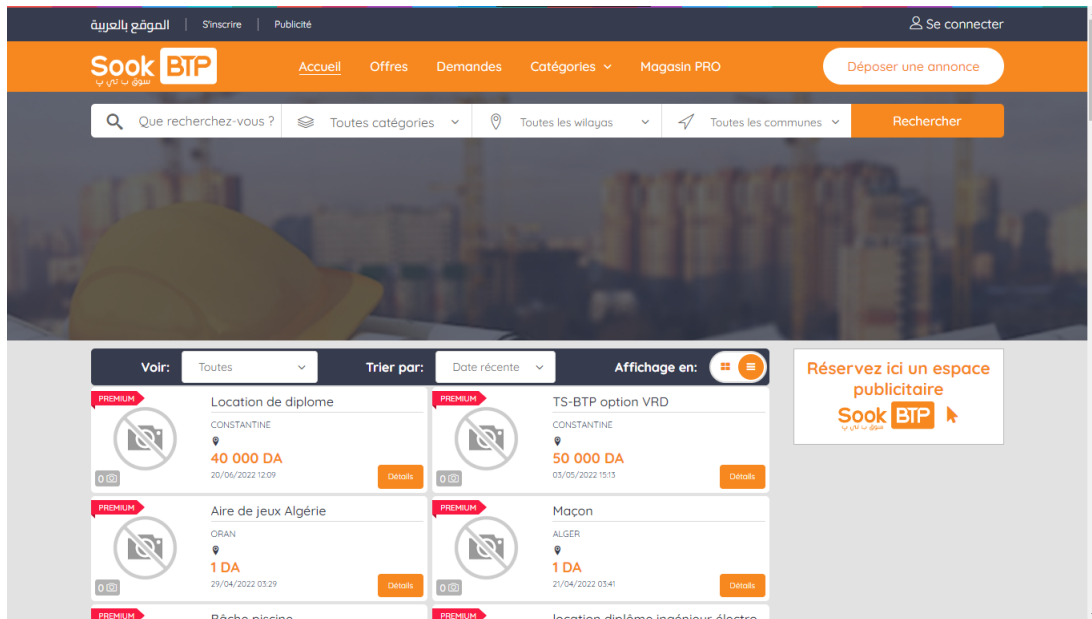


FIGURE 2.2 – Capture page accueil sookbtp.com

2.4.3 Wadifny

Wadifny est un nouveau site Web créé en 2021, et comme défini dans la page d'accueil, Wadifny est une plateforme numérique qui se propose comme intermédiaire entre le client et le prestataire, ce qui permet d'obtenir des offres compétitives pour le service demandé, et de choisir l'offre appropriée.[17]

Ce site web connaît encore quelques anomalies car il est toujours en cours de développement, par conséquent, nous nous focalisons sur les principes de fondation du site et ce qui est visible le jour de la consultation.

Wadifny se spécialise en tout type de service tel que celle du domaine du BTP, transport, organisation d'événement... etc, les services de BTP sont organisés en deux catégories principales : "service de rénovation décoration et construction" et "installation électricité". tout en proposant des sous-catégories bien plus détaillées liées à ce domaine.

Mais d'un autre côté, la recherche liée à ces catégories ainsi qu'aux autres manques d'options de recherche plus ciblée ou de triage.

Wadifny dispose aussi d'un local où vous pouvez vous inscrire ou solliciter un de leur service directement, ce qui ajoute plus de crédibilité aux services qu'ils proposent.

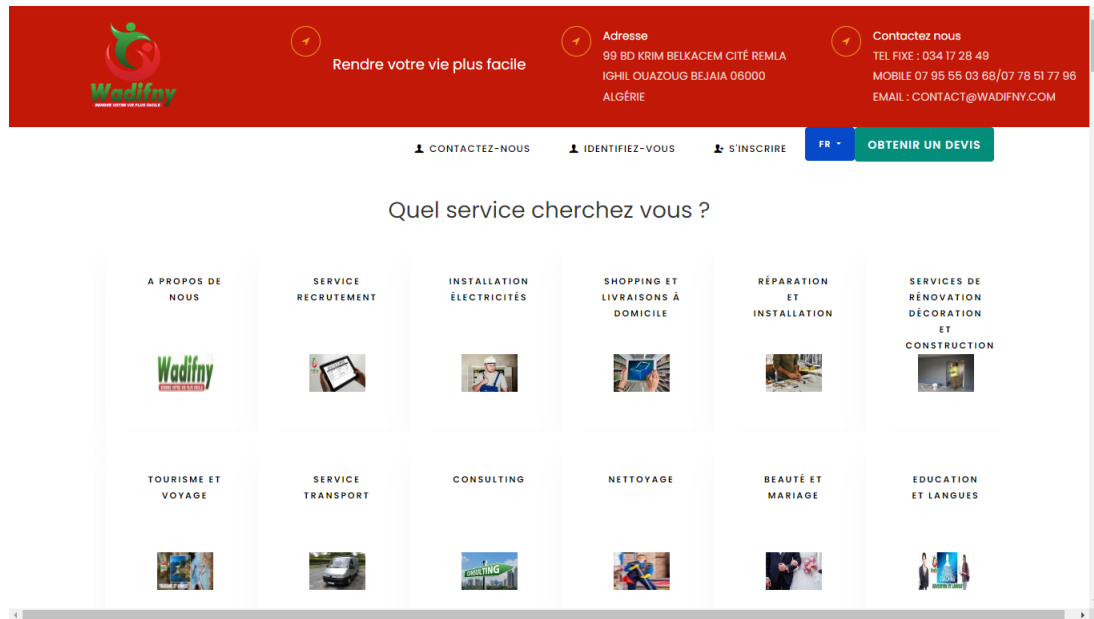


FIGURE 2.3 – Capture page accueil wadifny.com

2.4.4 Facebook Marketplace

Facebook Marketplace est une page web fournie par le fameux réseau social Facebook qui appartient à Meta. Facebook Marketplace est devenu un outil de promotion indispensable pour plusieurs entreprises. Il est utilisé par toutes les tranches d'âge de la communauté algérienne soit pour vendre ou acheter. Tout peut être trouvé, du plus banal des accessoires au plus cher et rare tel que les véhicules et l'immobilier.

Facebook Marketplace est beaucoup moins complexe que tout les autre sites Web qu'on a introduit dans ce chapitre, et offre une recherche très efficace basée sur l'emplacement de la personne qui cherche, avec la possibilité de spécifier un rayon sur la carte graphique en kilomètres pour spécifier la zone de recherche, ou même changer carrément l'emplacement en spécifiant l'emplacement voulu et propose aussi des filtres et des tris.

Facebook marketplace ne supporte que les annonces de vente des biens de divers catégories, mais ceci n'a pas empêché les gens de publier leurs services. La recherche reste toujours efficace et fonctionnelle.

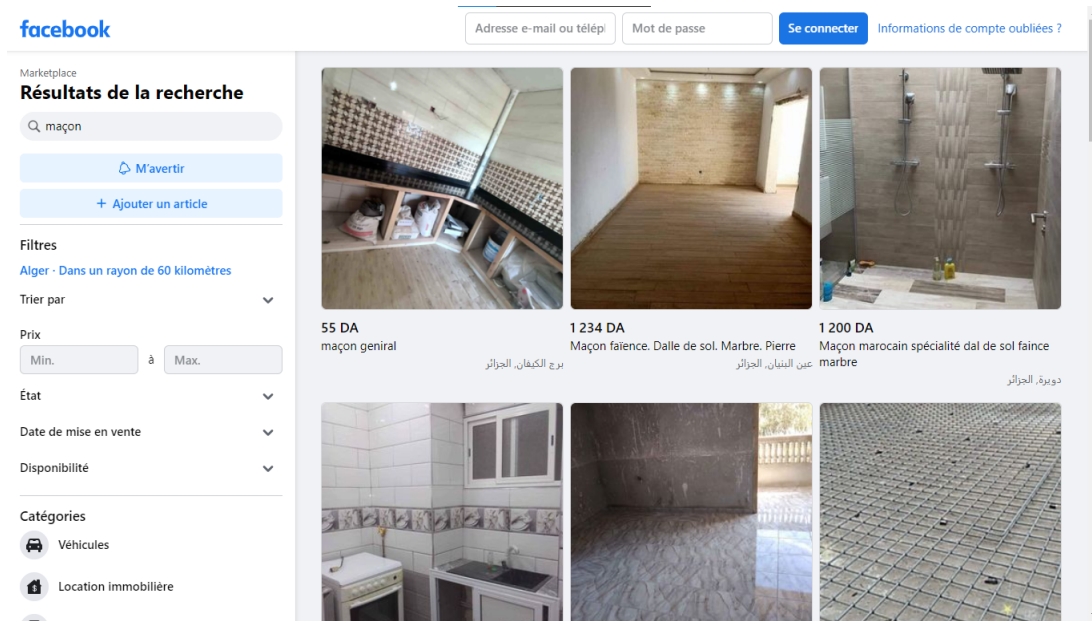


FIGURE 2.4 – Capture page accueil facebook.com/marketplace

2.5 Tableau comparatif

	OuedKniss	SookBTP	Wadifny	Facebook Marketplace
Efficacité de recherche	++	+	-	+++
Facilité d'utilisation	++	+	-	++
Disponibilité des service et des produits	+++	-	--	++
Crédibilité des services et des produits	-	-	+	-
Categorisation BTP	-	+	-	--

TABLE 2.1 – Comparaison entre les trois architectures

-- : médiocre, - : moyen, + : bon, ++ : très bon, +++ : excellent

2.6 Conclusion

Dans ce chapitre, nous avons étudié les différentes plateformes algériennes liées au domaine du BTP, et nous avons pu constater leurs avantages ainsi que leur inconvénient. Cette étude nous a permis de connaître les manques des solutions existantes et de mieux comprendre les besoins des futures utilisateurs de la plateforme dédiée au BTP, que nous comptons concevoir réaliser et dont la conception est détaillée dans le chapitre suivant.

Conception

3.1 Introduction

Dans ce chapitre, nous définissons certaines fonctionnalités que notre système possède, tout en tenant compte des notions théoriques. Nous commençant par une énumérations des besoins, suivi d'une description de l'approche adoptée lors de la conception et des différents diagrammes conceptuels.

3.2 Spécification des besoins

3.2.1 Besoins fonctionnels et non fonctionnels

Les besoins fonctionnels expriment d'une manière directe les besoins de l'utilisateur de l'application et qui conduisent par la suite à l'élaboration des modèles de cas d'utilisation. Par contre, les besoins non fonctionnels expriment des besoins techniques qui garantissent le bon fonctionnement de l'application et qui ne seront probablement pas visibles pour l'utilisateur.

3.2.1.1 Besoins fonctionnels

Les besoins fonctionnels de notre application sont les suivants :

- Inscription ou création d'un nouveau compte sur la plateforme.
- Connexion/Authentification, s'authentifier a un compte déjà crée au préalable.
- Gestion du compte utilisateur, tel la modification d'informations lié à l'utilisateur.

- Gestion des portfolios. Chaque utilisateur peut avoir plusieurs portfolios et chacun d'eux est lié à une seule catégorie, dans ce portfolio l'utilisateur montre ces travaux liés à une catégorie pour donner envie à de potentiels clients de solliciter ces services.
- Gestion des annonces, tel que la modification ou suppression d'annonces.
- Effectuer des recherches sur des annonces avec possibilité de filtrer et trier les résultats.
- Déconnexion, cette option est disponible qu'aux utilisateurs connectés à la plateforme.
- Donner une évaluation sur une annonce, donner une note ainsi qu'un avis/critique sur l'annonce.
- Signaler un abus dans une Annonce, portfolio ou une évaluation.
- Administrer les annonces (valider ou rejeter les annonces avant de les publier/supprimer).
- Administrer les signalements (valider ou rejeter les signalements).
- Administrer les catégories (Ajout, modification, suppression).
- Administrer les utilisateurs (Ajout, modification, suppression, activer, désactiver).

Remarque : La gestion d'une entité inclut la création, modification et suppression des instances de l'entité.

3.2.1.2 Besoins non fonctionnels

Les besoins non fonctionnels de notre application sont les suivants :

- Garantir l'**intégrité**, **cohérence** et **confidentialité** des données.
- **La facilité d'utilisation** (utilisabilité).
- **La maintenabilité** : facile à corriger en cas de problème et facile à mettre à jour (code source compréhensible et commenté).
- **La portabilité** : facile à utiliser de n'importe où et de n'importe quel appareil.
- **Le rendement et l'efficacité**

3.2.2 L'approche UP (Unified Process)

Pour la réalisation de ce projet, nous devons suivre une approche convenable pour le bon déroulement de conception et de réalisation de notre application, pour cela nous avons choisi l'approche UP (Processus unifié).

3.2.2.1 Processus unifié

processus unifié est un processus de développement logiciel construit sur UML ; il est itératif et incrémental, centré sur l'architecture, conduit par les cas d'utilisation et piloté par les risques.[18]

3.2.2.2 Le processus 2TUP

2TUP (2 track unified process) est un processus de développement logiciel qui implémente le Processus Unifié. Le 2TUP propose un cycle de développement qui sépare les aspects techniques des aspects fonctionnels en formant un cycle de développement en Y.[18]

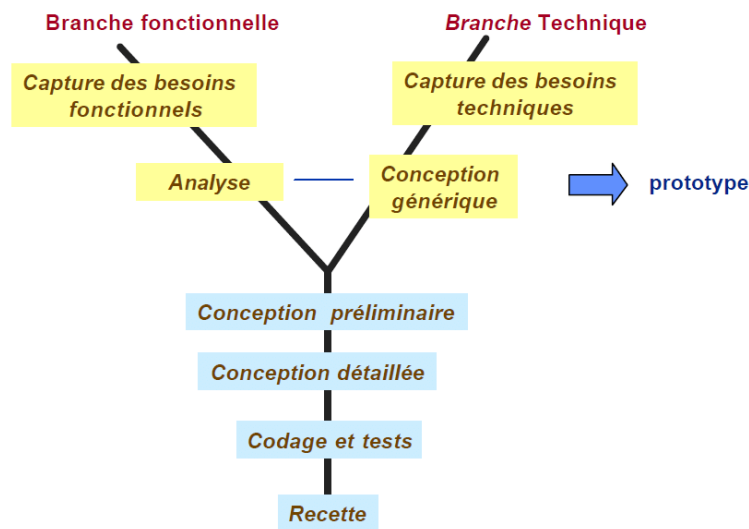


FIGURE 3.1 – La Méthode 2TUP [2]

3.2.2.3 Le langage UML

UML «UNIFIED MODELING LANGUAGE» se définit comme un langage de modélisation graphique et textuel destiné à comprendre et à décrire des besoins, spécifier et documenter des systèmes, esquisser des architectures logicielles, concevoir des solutions et communiquer des points de vue. UML unifie à la fois les notations et les concepts orientés objet. Il ne s'agit pas d'une simple notation, mais des concepts transmis par un diagramme en une sémantique précise et sont porteurs de sens au même titre que les mots d'un langage.[18]

les avantage de UML sont :

- UML est un langage formel et normalisé : il permet un gain de précision et de stabilité.
- UML est un support de communication performant : il Permet grâce à sa représentation graphique d'exprimer visuellement une solution objet, de faciliter la comparaison et l'évolution de solution.
- Son caractère polyvalent et sa souplesse en font un langage universel.[19]

3.2.2.4 Les différents types diagrammes d'UML

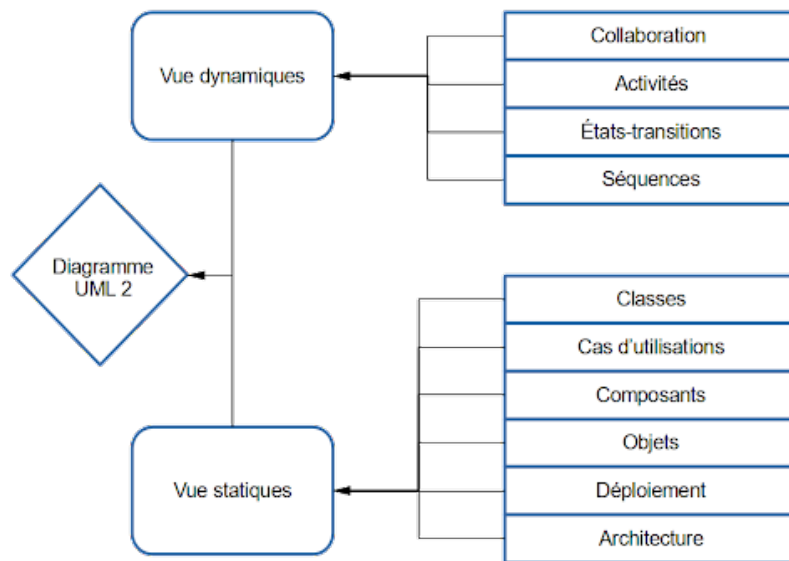


FIGURE 3.2 – Types de Diagrammes UML [3]

3.3 Identification des acteurs

Un acteur représente un rôle d'un utilisateur qui interagit avec le système étudié. L'utilisateur peut être un utilisateur humain, une organisation, une machine ou un autre système externe.[20]

Dans notre cas, nous avons quatre utilisateurs :

- Un **client** (authentifié) est l'utilisateur principal et peut être classifié en deux types : un **particulier** ou une **entreprise**.
- Un **visiteur** est un client non authentifié qui n'a pas accès à certaines fonctionnalités.
- Un **Modérateur** administre les annonces et les signalisations.

- Un **administrateur** administre en plus des annonces et des signalisations les catégories et les utilisateurs.

3.4 Les Diagrammes des cas d'utilisation

Dans le langage UML, les diagrammes de cas d'utilisation modélisent le fonctionnement du système et aident à capturer les exigences du système. Les diagrammes de cas d'utilisation décrivent la fonctionnalité générale et la portée d'un système. Ces schémas définissent également les interactions entre le système et ses acteurs. Les cas d'utilisation et les acteurs du diagramme de cas d'utilisation décrivent ce que fait le système et comment les acteurs l'utilisent, mais ne montrent pas comment le système fonctionne en interne.[20]

3.4.1 Diagramme global des cas d'utilisations

la Figure 3.3 représente le diagramme de cas d'utilisation générale.

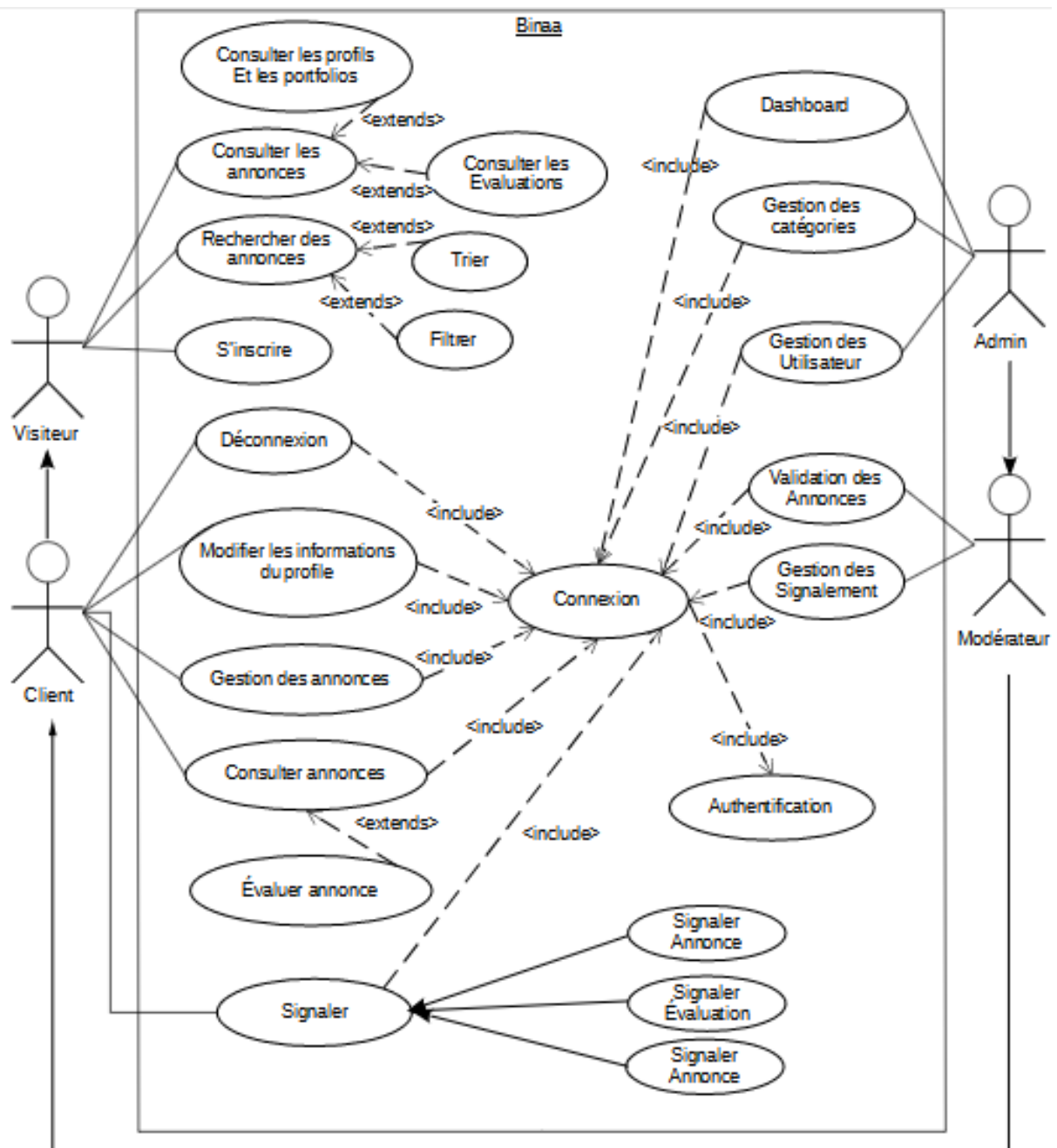


FIGURE 3.3 – Diagramme de cas d'utilisation globale

3.4.2 Les différents 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 décrit 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.[20]

On va associer à chaque cas d'utilisation une description textuelle des interactions entre l'acteur et le système et les actions que le système doit réaliser afin d'atteindre les résultats voulu par les acteurs. Pour exprimer les cas d'utilisation de notre système, on va utiliser les formalisation suivante :

Numéro de cas d'utilisation	Nom de cas d'utilisation.
But	Le but de cas d'utilisation.
Acteur	Acteur principal de cas d'utilisation.
Préconditions	Condition qui doit être remplie avant le début de cas d'utilisation.
Scénario nominal	Séquence d'action normale associée au cas d'utilisation.
Alternative	Séquence d'action alternative pouvant conduire également au succès.
Exception	Séquence d'action conduisant à un échec.

TABLE 3.1 – Le formalisme de description des cas d'utilisation.

3.4.3 Cas d'utilisation Gestion d'Annonces

Cas d'utilisation N°1	Gestion d'Annonces.
But	Gestion des annonces de client (ajout, modification, suppression).
Acteur	Client.
Préconditions	Authentifié.
Scénario nominal	<ul style="list-style-type: none"> - Accéder au tableaux de bord. - En cas de : <ol style="list-style-type: none"> a. Ajout d'une nouvelle annonce, l'utilisateur remplit le formulaire de la nouvelle annonce et valide. b. Sélectionner une annonce existante pour la modifier ou la supprimer carrément. - Le système valide la requête (soit ajout ou modification ou suppression).
Alternative	oublié de remplir un champ lors de la modification, le système demanderait de bien remplir les champs manquants avant de re-valider la modification.

TABLE 3.2 – Description des cas d'utilisation Gestion d'Annonces.

3.4.3.1 Diagramme de cas d'utilisation Gestion d'Annonces

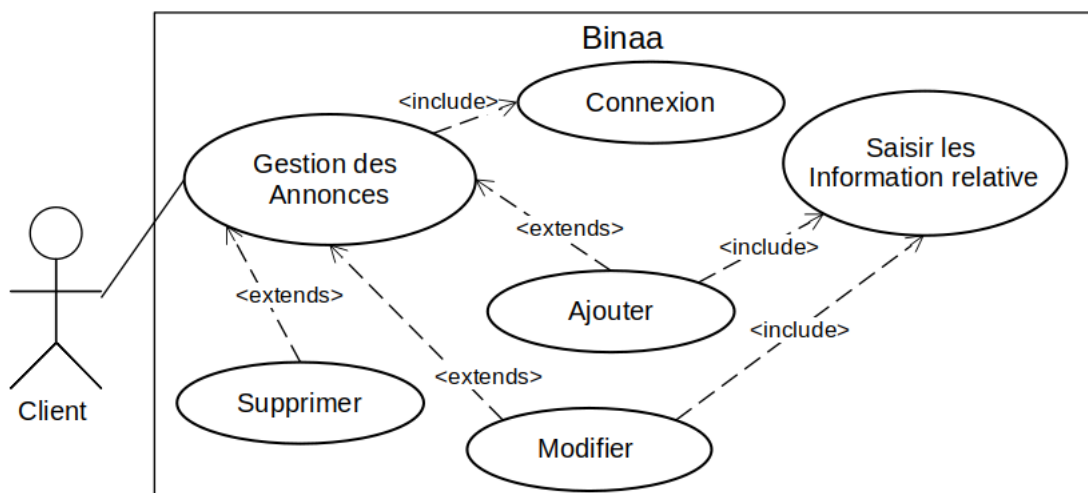


FIGURE 3.4 – Diagramme du cas d'utilisation Gestion d'Annonces

Remarque :

- 1 - La modification d'une annonce nécessite la validation d'un administrateur.
- 2 - L'ajout d'une nouvelle annonce nécessite la validation d'un administrateur.

3.4.4 Cas d'utilisation validation d'annonces

Cas d'utilisation N°2	Validation d'annonces.
But	Valider ou rejeter une annonce.
Acteur	Administrateur, Modérateur.
Préconditions	Authentifié.
Scénario nominal	<ul style="list-style-type: none"> - Accéder à l'espace administrateur. - Accéder à l'espace de validation d'annonce. - Valider les annonces qui répondent aux normes et rejeter celle qui ne répondent pas. - Le système valide les modification

TABLE 3.3 – Description des cas d'utilisation validation d'annonces.

3.4.4.1 Diagramme de cas d'utilisation validation d'annonces

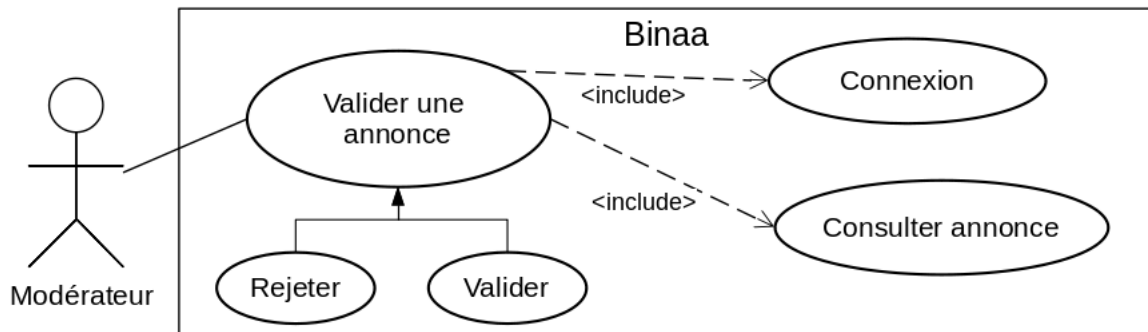


FIGURE 3.5 – Diagramme du cas d'utilisation validation d'annonce.

3.4.5 Cas d'utilisation Evaluer Annonce

Cas d'utilisation N°3	Evaluer Annonce.
But	Donner son avis sur un produit et/ou un service.
Acteur	Client.
Préconditions	Authentifié.
Scénario nominal	<ul style="list-style-type: none"> - Consulter l'annonce. - Naviguer vers la section d'évaluation. - Donner une note sur 5 - Ecrire un commentaire dans le formulaire de commentaire (facultatif) - Valider le formulaire - Le système valide l'évaluation
Alternative	oublié de remplir un champ lors de la validation d'une l'évaluation, le système demanderait de bien remplir les champ manquant avant de re-validé.

TABLE 3.4 – Description des cas d'utilisation Evaluer annonce.

3.4.5.1 Diagramme de cas d'utilisation Consultation et d'évaluation d'annonce

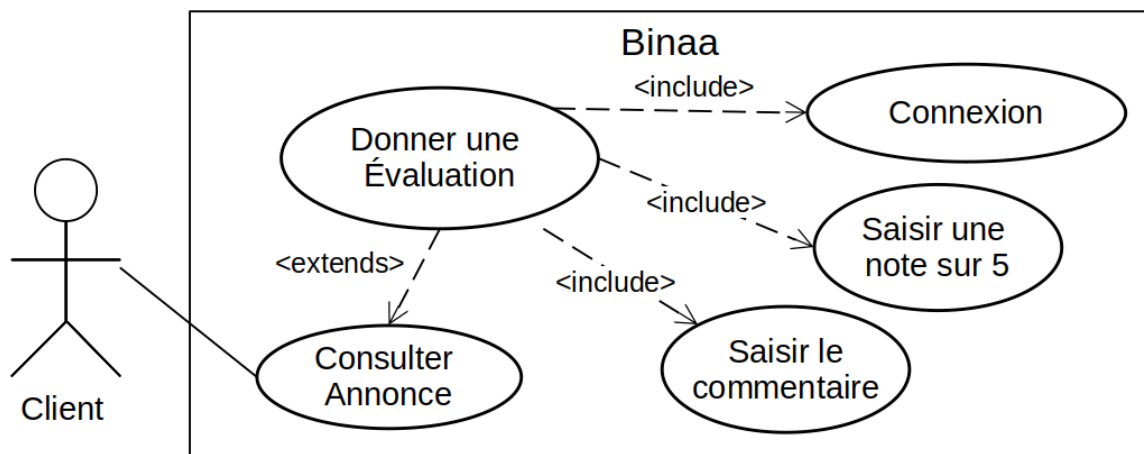


FIGURE 3.6 – Diagramme du cas d'utilisation Consultation et d'évaluation d'annonce.

3.4.6 Cas d'utilisation Signalement

Cas d'utilisation N°4	Signalement.
But	Signaler des annonces, commentaires ou portfolio qui ne répondait pas aux normes.
Acteur	Client, Modérateur, administrateur.
Préconditions	Authentifié.
Scénario nominal	<ul style="list-style-type: none"> - Accéder à l'annonce, commentaire ou portfolio qui ne répondait pas aux normes (contenu vulgaire . . . etc) . - Cliquer sur Signaler. - Valider le signalement.
Alternative	Signaler une entité supprimée, le système indiquera qu'elle n'existe plus.

TABLE 3.5 – Description des cas d'utilisation Signalement.

3.4.7 Cas d'utilisation recherche annonce

Cas d'utilisation N°5	recherche annonce.
But	Recherche annonce avec option de filtrer et trier les résultats.
Acteur	tout le monde.
Préconditions	aucune.
Scénario nominal	<ul style="list-style-type: none"> - Accéder à la page d'accueil. - remplir le champ de recherche avec les mot clé de la recherche souhaitée. -ajouter un filtre (date, prix, livraison. . .). - valider la recherche - Le système affiche la page de résultat avec la possibilité de spécifier à nouveau un filtre et propose des tris (par date, par prix)

TABLE 3.6 – Description des cas d'utilisation recherche annonces.

3.4.7.1 Diagramme de cas d'utilisation recherche annonce

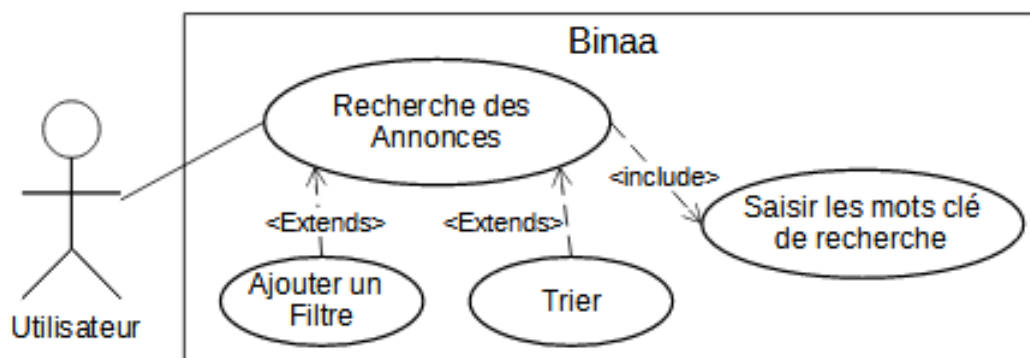


FIGURE 3.7 – Diagramme du cas d'utilisation recherche annonce.

3.5 Diagramme de séquence

Les diagrammes de séquence décrivent les interactions entre un groupe d'objets en montrant, dans l'ordre, l'envoi de messages qui se produit entre les objets. Le diagrammes de séquence peut également montrer les transferts de données échangés lors de l'envoi du message.[21]

3.5.1 Diagramme de séquence Ajout d'annonce

Ce diagramme de séquence montre les étapes et les interactions entre le client (Utilisateur) et la machine (vue) lors de l'ajout d'une nouvelle annonce et dans un second temps les vérifications de formulaire passé lors la première étape et la création de la nouvelle annonce via l'API et la BDD.

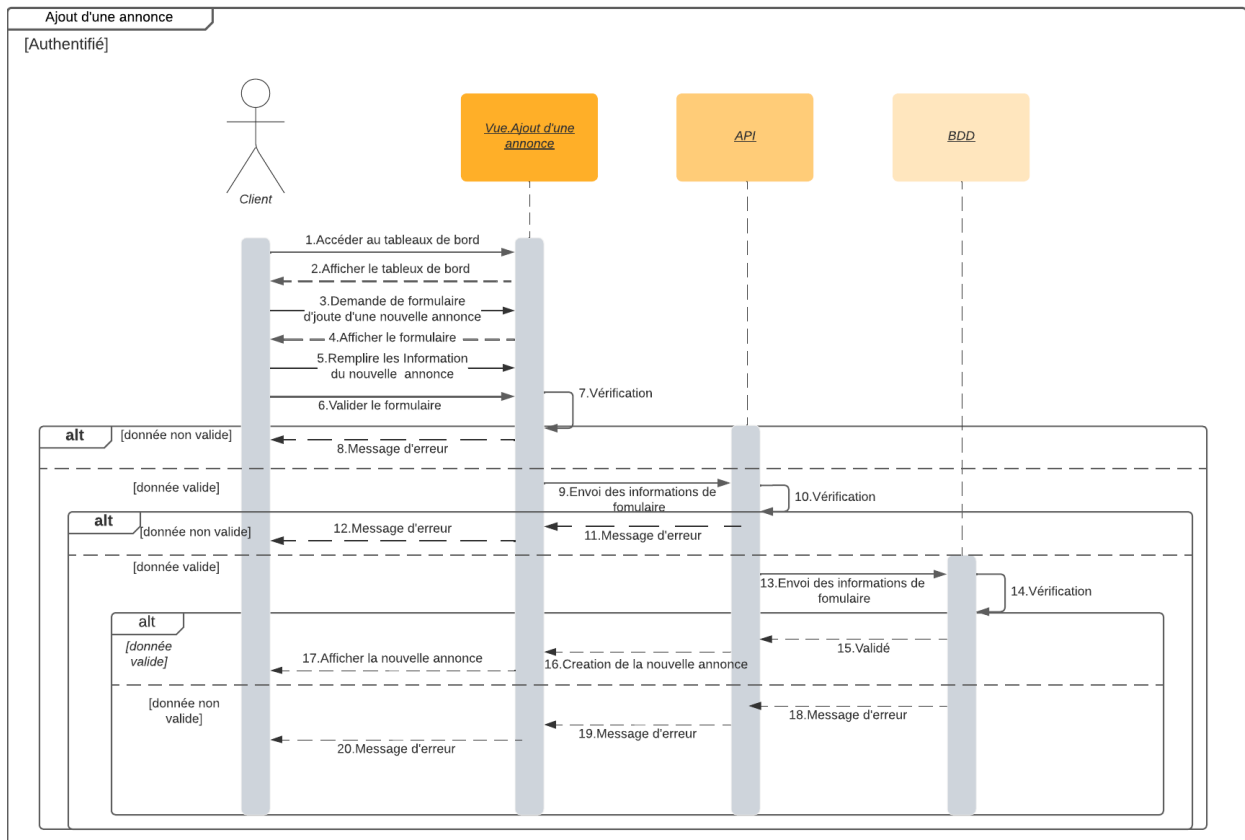


FIGURE 3.8 – Diagramme de séquence Ajout d'annonce.

3.5.2 Diagramme de séquence d'évaluation d'annonces

Ce diagramme de séquence démontre les étapes et les interactions entre le client (Utilisateur) et la machine (vue) lors de l'évaluation d'une annonce et dans un seconde temps la sauvegarde de l'évaluation faite lors de la première étape dans le serveur (l'API et la BDD).

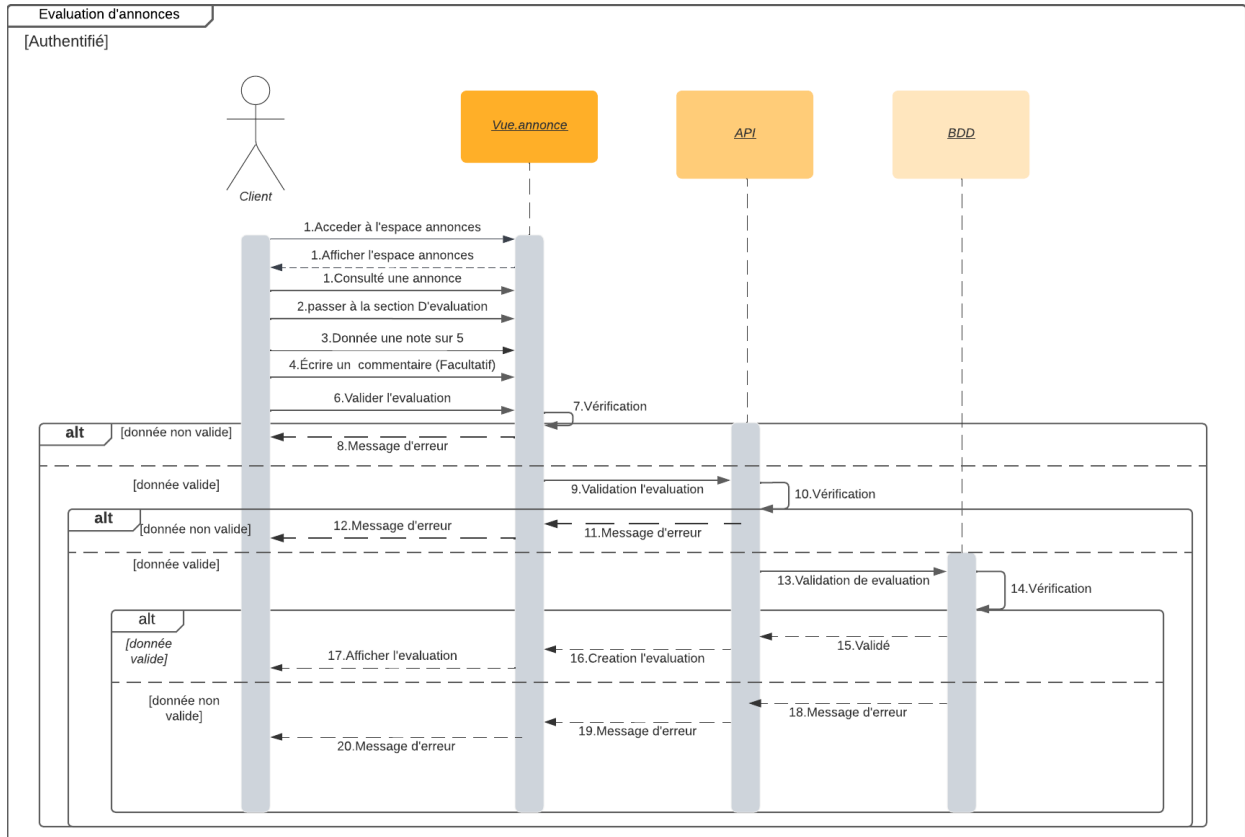


FIGURE 3.9 – Diagramme de séquence d'évaluation d'annonce.

3.5.3 Diagramme de séquence de validation d'annonces

Dans ce diagramme de séquence on présente les interactions l'administrateur et la machine dans un premier temps, il vérifié les annonces et valide ceux qui réponde aux norme et désapprouve les annonces qui ne répondait pas aux normes puis dans un seconde temps la vérification et la création de l'annonce côté serveur (l'API et la BDD).

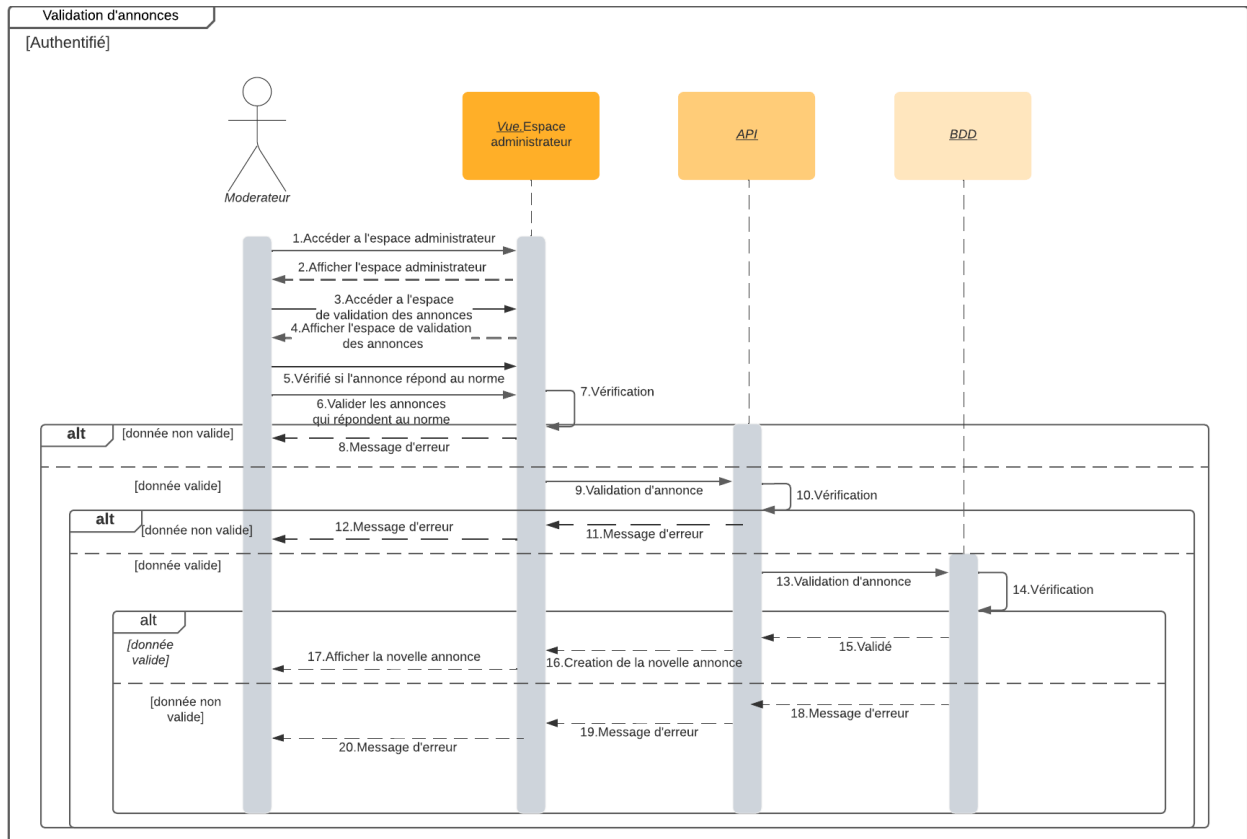


FIGURE 3.10 – Diagramme de séquence validation des annonces.

3.6 Diagramme de classe

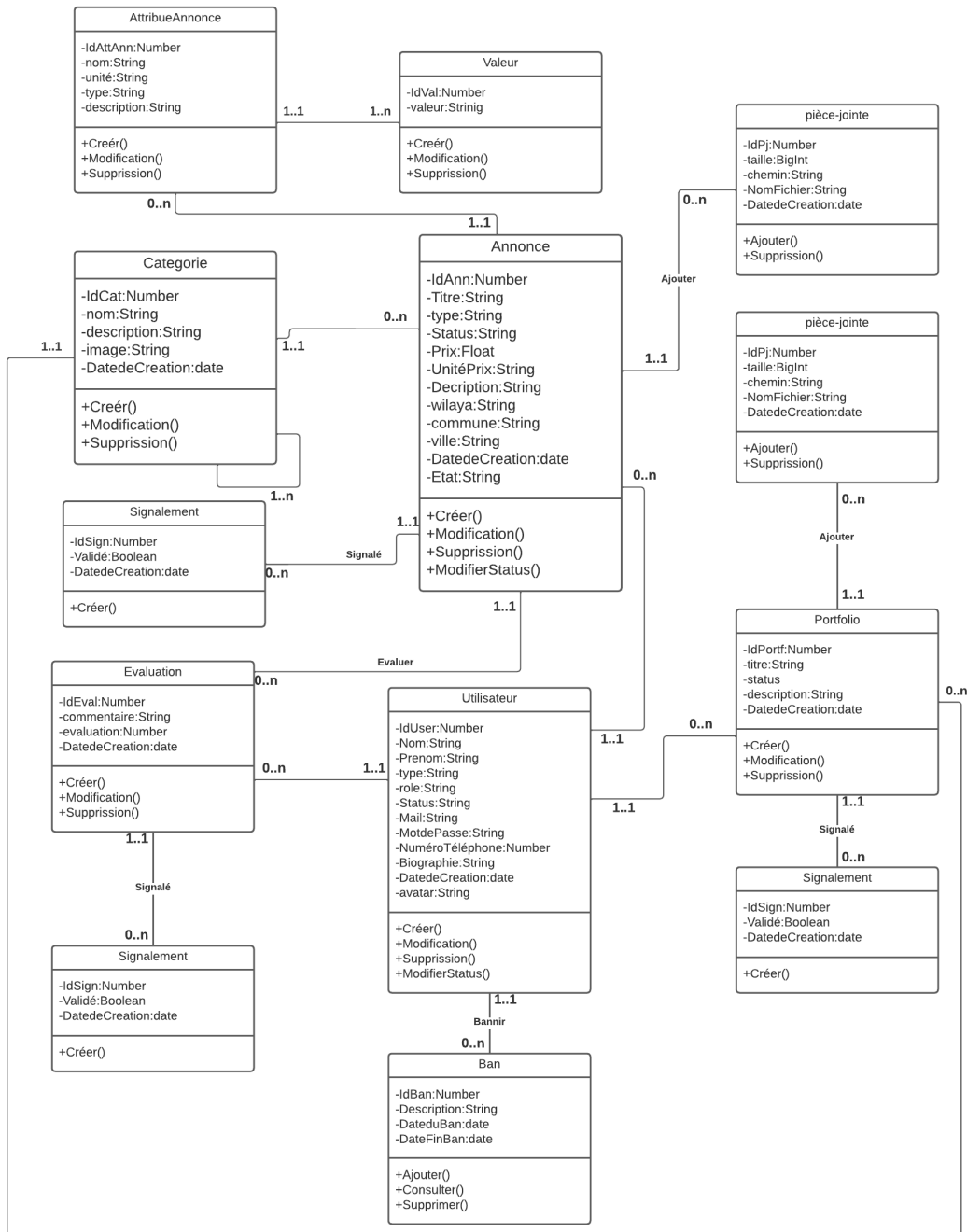


FIGURE 3.11 – Diagramme de classe.

Attribut	Description
id	Identificateur d'élément.
titre	le titre d'une annonce .
type : Annonce	le type d'annonce (produit ou service)
type : AttribueAnnonce	le type de valeur affecter à l'attribut. (chaîne de caractère ou chiffre).
type : Utilisateur	le type de utilisateur : particulier ou entreprise.
nom, prénom	le nom et prénom d'une personne (utilisateur).
Statut	un statut d'un élément (soit active,inactive ou autre).
Prix	le prix d'un service ou d'un produit.
Description	une petite définition de produit ou de service.
rôle	un rôle pour un utilisateur (administrateur, modérateur ou client).
MotDePass	le mot de passe d'un utilisateur .
avatar	le chemin vers l'image de profil d'utilisateur.
Etat	état d'une annonce type produit (eg : neuf, utilisé).
livrable	si l'annonce de type produit est proposé avec un service de livraisons .
mail	une adresse électronique tel que gmail, yahoo. . .
taille	la taille du fichier en MégaOctet .
chemin	le chemin du fichier.
NomFichier	le Nom du fichier.
dateDeCreation	la date de création d'entité.
dateDuBan	la date de début de blocage du compte d'utilisateur.
dateFinBan	la date de fin de blocage du compte d'utilisateur.
validé	Si le signalement a étaié évaluer par un modérateur/administrateur et valider (si valide vrais sinon faux).
Commentaire	un avis sur un service ou un produit.
evaluation	une notation sur 5 d'un produit ou service .
Quantité	quantité de produits commander.
Unité	unité de vente (ex : sac, pièce, mètre carré ...).
adresse	L'adresse ou il habite.

TABLE 3.7 – Dictionnaire de données.

3.7 Schéma relationnel

Le schéma relationnel repose sur des règles de passage qui mène à la transformation d'un modèle conceptuel (diagramme de classe) vers un modèle relationnel, ses règles repose ainsi sur les cardinalités présentes dans le modèle conceptuel.

En appliquant les règles de dérivation du modèle relationnel à partir d'un modèle de classe (Annexe A), nous obtenons les relations suivantes :

Annonce (idAnn, titre, type, status, prix, unitéprix, description, wilaya, commune, ville, datecreation, état, livrable, commander, idCat#, idUser#)

Catégorie (idCat, nom, description, image, datecreation, idCat#)

Evaluation (idEval, commentaire, evaluation, datecreation, idUser#, idAnn#)

Utilisateur (idUser, nom, prenom, type, isadmin, role, status, mail, motdepasse, numéro-téléphone, biographie, datedecreation, avatar)

SignalerEvaluation (idSign, validé, datedecreation, idEval#, idUser#)

SignalerAnnonce (idSign, validé, datedecreation, idAnn#, idUser#)

SignalerPortefolio (idSign, validé, datedecreation, idPortf#, idUser#)

Ban (idBan, description, dateDuBan, dateFinBan, idUser#)

Portefolio (idPortf, titre, status, description, datecreation, idUser#, idCat#)

Pièce-JointeAnnonce (idPj, taille, chemin, nomFichier, datecreation, idAnn#)

Pièce-JointePortefolio (idPj, taille, chemin, nomFichier, datecreation, idPortf#)

Attribue-annonce (idAttAnn, nom, unité, type, description, idAnn#)

Valeur (idVal, valeur, idAttAnn#, idComm#)

Commande (idComm, quantité, description, status, livraison, adresse, datecreation, idAnn#)

3.8 Conclusion

Au cours de ce chapitre, nous avons présenté l'approche sur lequel nous nous sommes basé pour mener une conception détaillée du travail réalisé. Et dans un second temps nous avons présenté quelques-unes des fonctionnalités de notre système ainsi que leur fonctionnement.

Réalisation

4.1 Introduction

Dans ce chapitre nous allons commencer par parler du choix des technologies utilisées tel que l'environnement de développement, les logiciels, et des langages de programmation. Nous terminons par une démonstration de certaines interfaces de notre plateforme.

4.2 Choix technique

4.2.1 Technologies du backend

Le backend comme bien défini dans les chapitres précédents, c'est le côté serveur où l'instance de l'application s'exécute, le choix des technologie du backend comprend le choix du langage de programmation et les outils et bibliothèques et framework construit autour de ce langage et comprend aussi le choix d'un SGBD.

Dans notre cas, nous avons choisi NodeJs comme langage de serveur, et ExpressJs comme framework, Sequelize comme ORM (Object-relational mapping).

4.2.1.1 NodeJs



Node.js est un environnement libre d'exécution JavaScript asynchrone et orienté événement, Node.js est conçu pour générer des applications extensibles [22]. Node.js est implémenté en langage C et C++ par Ryan Dahl en 2009. Node.js est basé sur les performances et la réduction de consommation de mémoire c'est ce qui rend Node.js plus adapté pour les processus serveur de longue durée. Contrairement à la plupart des autres environnements modernes, les processus Node.js ne supportent pas le multithreading mais il est basé sur un modèle d'événement d'E/S (entré/sortie) asynchrone[23].

4.2.1.2 NPM



NPM (Node Package Manager) est un outils de développement qui aide à la gestion des modules de NodeJs (installation, désinstallation, mise à jour... etc), NPM est l'ensemble de la ligne de command du client NPM et le Registre qui joue le rôle d'un serveur.[24]

4.2.1.3 ExpressJs



Express.js est un framework d'application Web fait avec Javascript et qui s'exécute seulement en côté serveur (Node.js), Express.js utilise le module http pour créer un serveur ou plusieurs pour but de servir des fichiers html ou d'autres fichiers que sa soit statique comme les images, vidéo, document ... etc ou des fichiers dynamiques tel que des fichiers HTML combiné avec un langage de templating qui permet d'injecter des données dynamiques venant d'une BDD ou d'un lien externe.

Express.js offre la possibilité d'utiliser un système de routage qui permet de dynamiser les pages web et aussi et permet d'utiliser des middlewares qui est une fonction qui prend en paramètre trois variables : celle qui porte les informations de la requête, celle qui porte les informations de la réponse et un callback (fonction passée en paramètre) pour passer au prochain middleware, les middlewares sont conçus pour un meilleur contrôle sur les requêtes et réponses et pour une meilleure gestion de code, par exemple au lieu de répéter la procédure de vérification d'authentification à chaque fois, on a qu'à assigner un middleware qui va être exécuté d'une manière automatique à chaque déclenchement d'une requête.

4.2.1.4 Sequelize



Sequelize est d'abord un module écrit en TypeScript et qui est disponible sur NPM, il est l'un des ORM les plus populaires dans la communauté Node.js, car il offre beaucoup de fonctionnalités avancées.

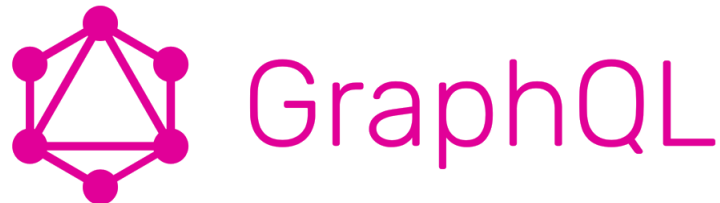
Sequelize permet de faire des requêtes SQL vers différents SGBD existants. Grâce aux migrations nous n'avons plus besoin d'écrire du code SQL, Les migrations représentent le code javascript qui permettra de générer les tables nécessaires pour notre application.

Sequelize permet aussi de créer des Modèles (une classe représentant une table dans

SQL), un modèle n'est qu'une classe qui vas recevoir la même signature et nom qu'une table, et une instance de cette classe représente une ligne de la table cible.

Sequelize met en disposition pour le développeur une interface de programmation qui lui permet d'interagir avec la BDD en utilisant les modèles déjà créés et permet de récupérer les résultats de la requête SQL en format JSON.[25]

4.2.1.5 GraphQL



GraphQL est un langage de requête pour les API. Les API classique faite avec l'architecture REST, fournit pour chaque ressource un URI (Uniform Resource Identifier) unique qui est généralement représenté par une route ce qui implique plusieurs routes (endpoints), mais dans l'architecture de GraphQL, on a qu'une seule route (endpoint), qui reçoit une chaîne de caractère qui représente une requête écrite en langage de GraphQL avec la méthode "POST".[26]

4.2.1.6 Mysql



MySQL est une base de données relationnelle libre très utilisés 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. Les tables sont reliées entre elles par des relations.[27]

4.2.2 Technologies du frontend

4.2.2.1 VueJs



est un framework JavaScript pour la création d'interfaces utilisateur. Il s'appuie sur les standards HTML, CSS et JavaScript et fournit un modèle de programmation déclaratif et basé sur des composants qui vous aide à développer efficacement des interfaces utilisateur, qu'elles soient simples ou complexes.[28]

4.2.2.2 Vuetify



Vuetify est un framework de composants d'interface utilisateur construit au-dessus de Vue.js. Basé sur la spécification Material Design, il fournit aux développeurs les outils dont ils ont besoin pour créer des interfaces utilisateur riches et attrayantes.[29]

4.2.2.3 Axios

Axios est un client HTTP basé sur les promesses compatibles avec node.js et les navigateurs. Il est isomorphe (c'est-à-dire qu'il peut opérer dans le navigateur et dans node.js avec le même code). Côté serveur, il utilise le module natif http de node.js, et côté client (navigateur) il utilise les XMLHttpRequests.[30]

4.3 Outils de développement

Les outils de développement comprennent l'éditeur de texte et l'environnement de développement et tout logiciel facilitant le processus de développement.

4.3.1 Vs code

Visual Studio Code communément appelé VsCode 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).[31]

4.3.2 VueCLI

Vue CLI vise à être la référence d'outillage standard pour l'écosystème Vue. Il garantit que les différents outils de construction fonctionnent correctement avec des valeurs par défaut raisonnables afin que vous puissiez vous concentrer sur l'écriture de votre application au lieu de passer des journées à vous débattre avec des configurations. Dans le même temps, il offre toujours la possibilité de modifier la configuration de chaque outil.[32]

4.3.3 Git et Github

4.3.3.1 Git

Git est un logiciel de gestion de versions conçu pour gérer les projets de petite à très grandes envergures. Il permet de stocker un ensemble de fichiers en conservant la chronologie de toutes les modifications qui ont été effectuées dessus.

C'est un logiciel libre créé par Linus Torvalds qui a souhaité remplacer l'outil propriétaire qui était utilisé pour le développement du noyau Linux.

Un logiciel de gestion de versions conserve toutes les versions des fichiers, ainsi que les différences entre les fichiers.

4.3.3.2 GitHub

Est un service web d'hébergement et de gestion de développement de logiciels, utilisant le logiciel de gestion de versions Git. développée par Chris Wanstrath, PJ Hyett et Tom Preston-Werner.

4.4 Quelques captures d'écran de l'application

Présentation de notre plateforme par quelque photos qui présentes certaines de ces fonctionnalités.

4.4.1 Page de connexion

Cette photo présente la page d'authentification.

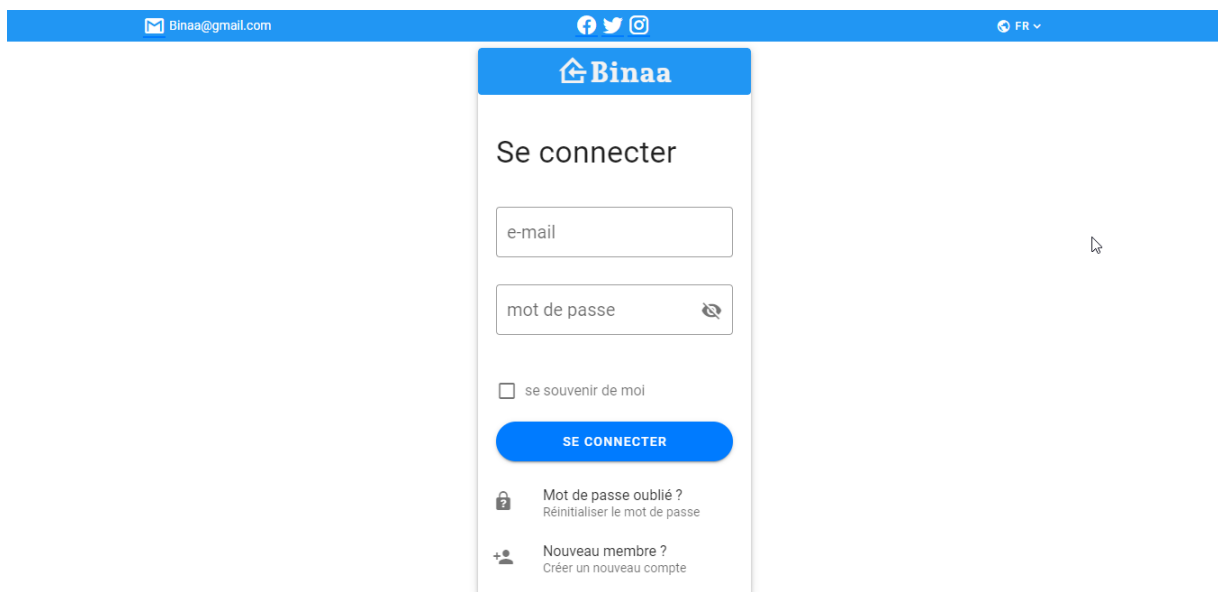


FIGURE 4.1 – Page de connexion

4.4.2 Page de création d'un nouveau compte

Page d'inscription pour les nouveaux utilisateurs intéressés de rejoindre la plateforme et d'avoir un compte personnel.

FIGURE 4.2 – Page d'inscription

4.4.3 Page de dashboard

Cette photo présente la page de dashboard accessible aux utilisateurs inscrits, et affiche toutes les annonces faites par cette utilisateur.

#Id	Image	Type	titre	prix	unité	Actions
158		SERVICE	cripsage exterieure	3900	jour	
159		SERVICE	Revetement exterieure en bois	400	mètre²	
160		SERVICE	lle de sol	4000	jour	
161		SERVICE	cripsage interieure	3000	jour	

FIGURE 4.3 – dashboard

4.4.4 L'interface d'une annonce

Cette photo montre comment une annonce est présentée sur la plateforme.

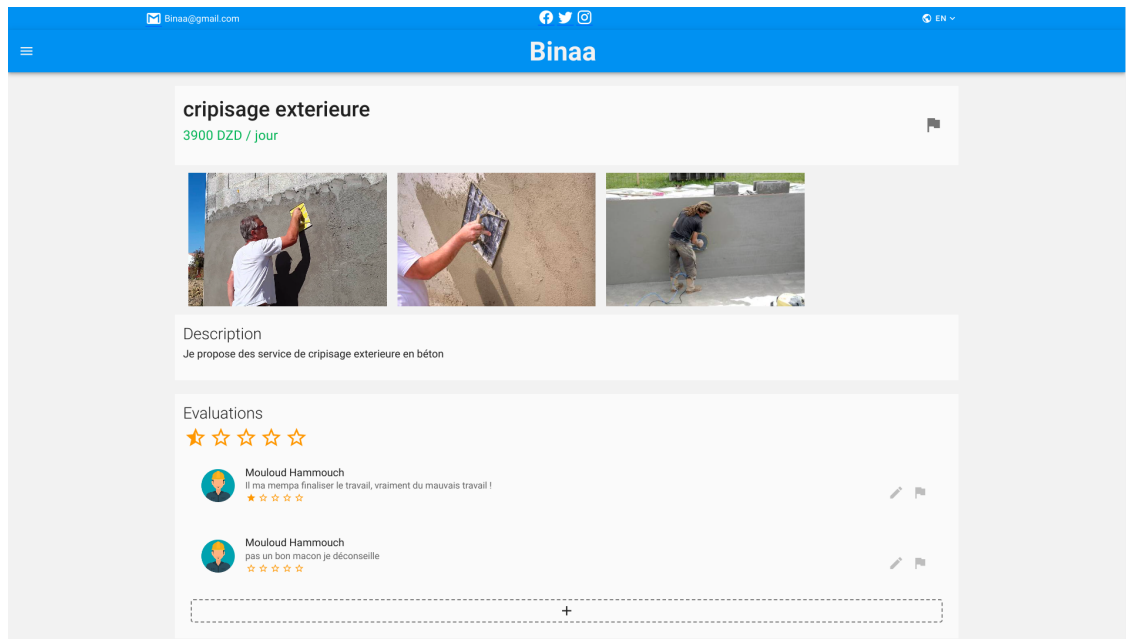


FIGURE 4.4 – Exemple d'une annonce

4.4.5 Page de création d'une nouvelle annonce

Page accessible aux utilisateurs inscrit souhaitant faire une nouvelle annonce.

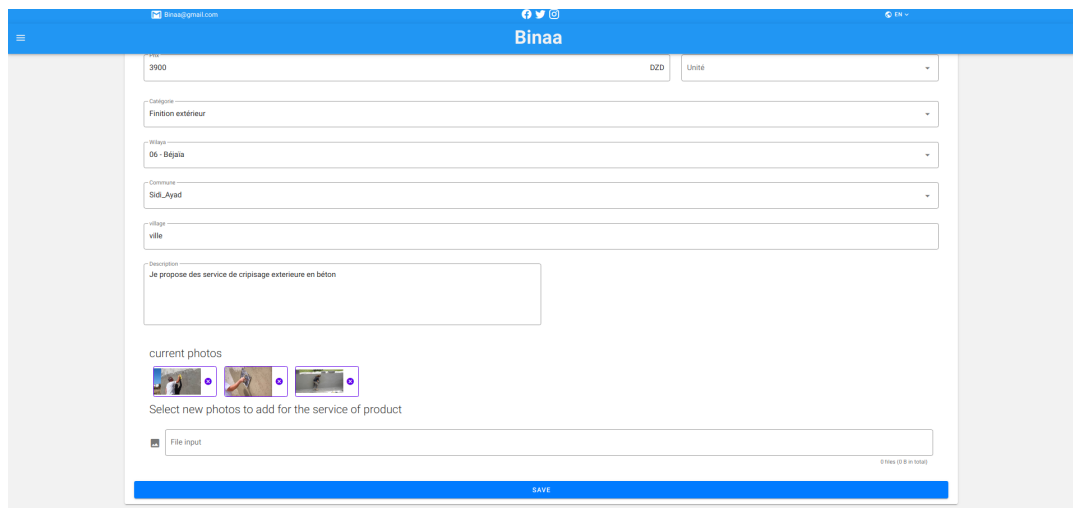


FIGURE 4.5 – Page de création d'une nouvelle annonce

4.4.6 page d'accueil

Cette photo présente la page d'accueil de notre plateforme.

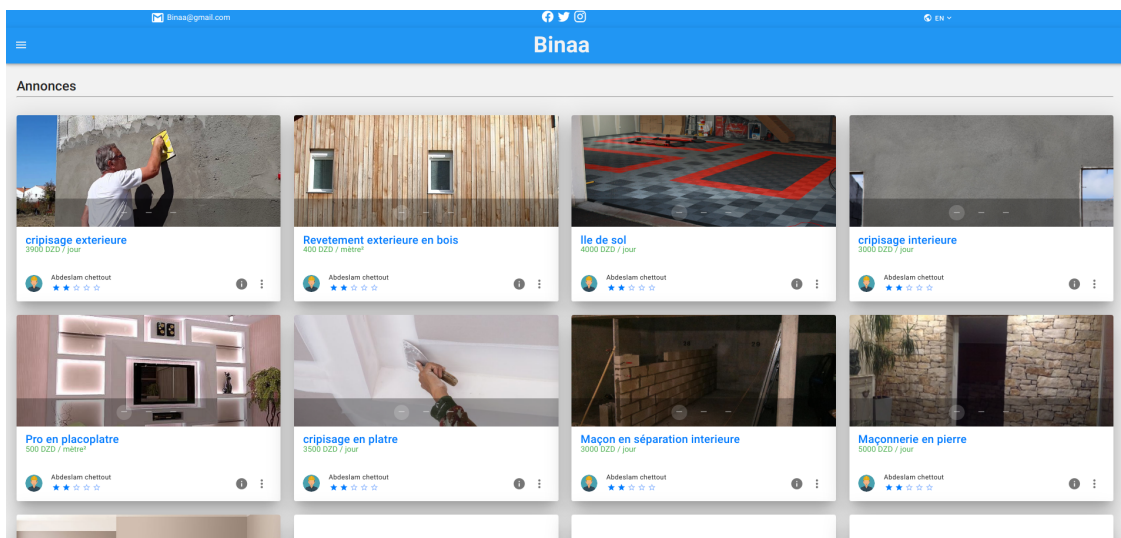


FIGURE 4.6 – Page d'accueil

4.4.7 Évaluation d'une annonce

Cette photo présente la fonctionnalité d'évaluation d'une annonce.

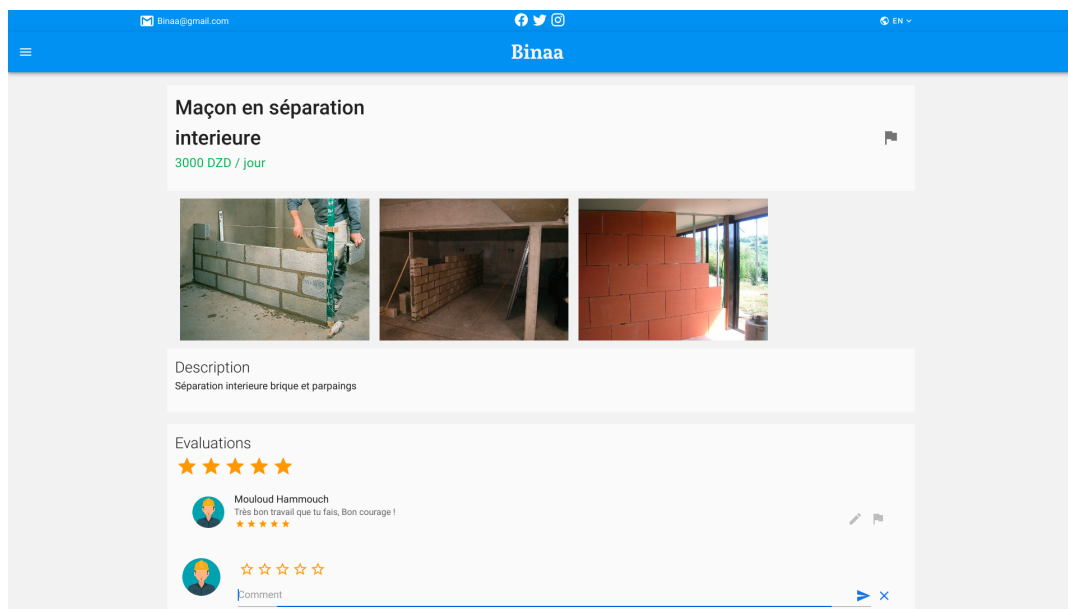


FIGURE 4.7 – Évaluer une annonce

4.4.8 Catégorie

Cette photo montre les catégories présente sur la plateforme.

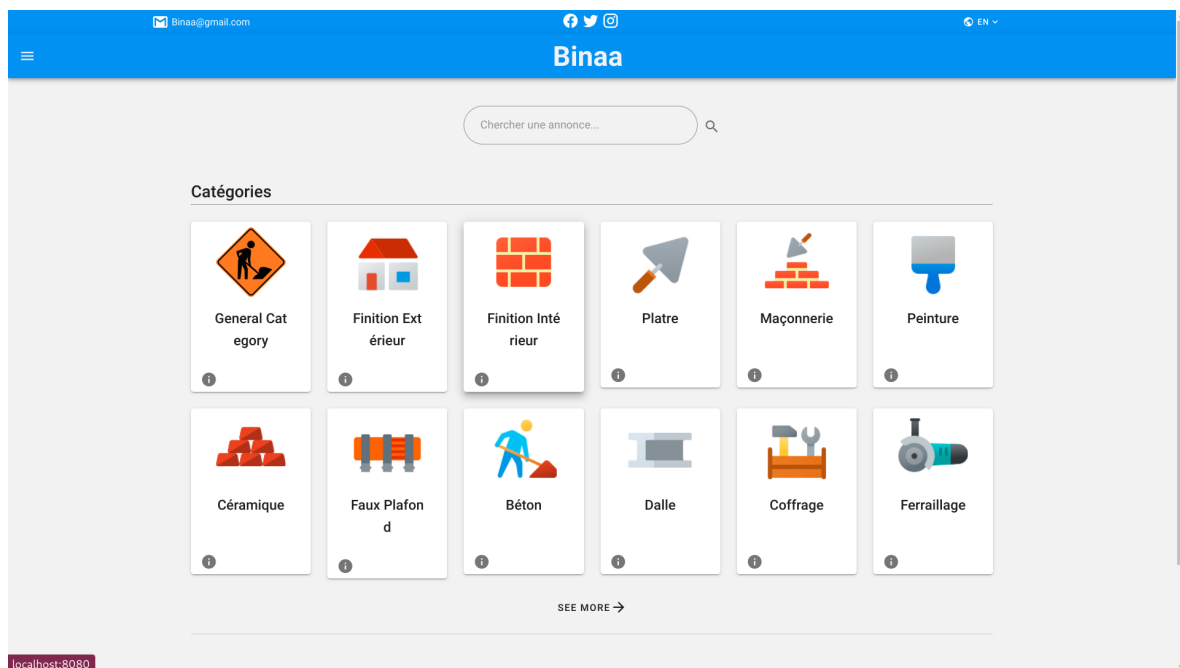


FIGURE 4.8 – Liste de Catégorie

4.5 Conclusion

Dans ce chapitre, dans un premier temps, nous avons présenté les outils et technologies utilisés pour mener à bien notre projet. Par la suite nous avons fait la démonstration de certaines interfaces de notre plateforme.

Conclusion générale

Dans le cadre de notre projet de fin de cycle, nous avons conçu et réalisé une plateforme de gestion du marché du BTP en Algérie basée sur l'architecture REST, cette solution est proposée pour répondre à la problématique de manque de plateformes spécialisées dans le domaine du BTP, et qui s'assure de mettre des évaluations de la qualité de services et produits mis en avant sur leur plateforme. ainsi qu'une meilleure présentation de leurs annonces (services et produits). Quand à l'utilisation de l'API REST c'est pour son côté évolutive, car Binaa est une future application multiplateforme qui peut fournir des applications Android et des applications de bureau pour de multiples utilisations comme la gestion des produits et services et plus de fonctionnalités avancées comme l'impression et la gestion des factures... etc.

Notre projet a pour but d'essayer de proposer une plateforme qui se spécialise dans le domaine du BTP et qui dispose de plus d'outils, afin d'assurer une meilleure fiabilité des annonces, ainsi qu'une meilleure efficacité et facilité d'utilisation.

Notre projet s'est déroulé en trois grandes parties, nous avons d'abord procédé à l'identification des manque d'options et de fonctionnalité sur les plateformes déjà existantes.

Ensuite, nous avons procédé à l'identification des besoins de notre plateforme pour répondre aux mieux aux exigences d'une plateforme spécialisée dans le domaine du BTP, tout en essayant au maximum de remédier aux manquements existant sur d'autres plateformes.

Finalement, nous sommes passés à la réalisation de notre plateforme.

Ce projet a fait l'objet d'une expérience très intéressante, car il nous a permis de nous familiariser avec de nouvelles notions d'une part, et d'améliorer nos connaissances et nos compétences dans le domaine de la programmation, et d'autre part de renforcer notre sens de responsabilité dans la gestion des projets qui permet l'insertion dans le domaine professionnel.

La réalisation de notre projet a été limitée par le manque de temps et d'expérience. Par ailleurs, nous avons envie d'étoffer notre projet vers les perspectives suivantes :

- L'ajout de plus d'options et d'offrir plus de services. tel que la création d'un espace de gestion de vente et d'achats pour des boutiques et d'entreprises.
- La création d'une version mobile.
- Éventuellement étendre ce projet et l'exploiter dans d'autres domaines.

Bibliographie

- [1] <https://ahmetozlu.medium.com/mastering-rest-architecture-rest-architecture-details-e47ec659f6bc>. en ligne ; visité :30/09/2022.
- [2] https://www.researchgate.net/figure/La-methode-2TUP-6_fig6_303497462. en ligne ; visité :20/09/2022.
- [3] Boubker Sbihi. <https://edutice.archives-ouvertes.fr/edutice-00277818/file/a0509b.htm>. en ligne ; visité :04/03/2022.
- [4] Kenneth Laudon and Jane Laudon. Managing information systems, 2009.
- [5] <https://www.geeksforgeeks.org/components-of-information-system/>. en ligne ; visité :18/06/2022.
- [6] <https://www.britannica.com/list/5-components-of-information-systems>. en ligne ; visité : 18/06/2022.
- [7] Francis B. <https://openclassrooms.com/fr/courses/2100086-decouvrez-le-monde-des-systemes-dinformation/5195891-identifiez-les-4-fonctions-du-systeme-d-information>. en ligne ; visité : 18/06/2022.
- [8] <https://www.geeksforgeeks.org/types-of-information-system/>. en ligne ; visité : 18/06/2022.
- [9] Romain Willmann. Apprendre à créer des application web avec javascript node.js, mongodb et le cloud à portée de main, 2017.
- [10] <https://www.akana.com/blog/what-is-rest-api>. en ligne ; visité :18/06/2022.
- [11] <https://www.syloe.com/glossaire/application-web/>. en ligne ; visité : 18/06/2022.

- [12] Hanin M. Abdullah and Ahmed M. Zeki. Frontend and backend web technologies in social networking sites : Facebook as an example, 2014.
- [13] <https://developer.mozilla.org/fr/docs/Web/CSS>. en ligne ; visité : 18/06/2022.
- [14] Éric Brousseau. Commerce électronique : ce que disent les chiffres et ce qu'il faudrait savoir. *Economie et Statistique*, 339-340 :147–170, 2000.
- [15] https://www.sookbtp.com/page_std.php?id=1. en ligne ; visité : 25/06/2022.
- [16] https://www.sookbtp.com/page_std.php?id=7. en ligne ; visité : 25/06/2022.
- [17] <https://www.wadifny.com/>. en ligne ; visité : 25/06/2022.
- [18] Pascal Roques and Franck Vallée. Uml 2 en action de l'analyse des besoins à la conception, 2007.
- [19] <http://uml.free.fr/index-me.html>. en ligne ; visité :02/08/2022.
- [20] ibm. <https://www.ibm.com/docs/fr/rational-soft-arch/9.5?topic=diagrams-use-case>. en ligne ; visité :04/03/2022.
- [21] Laurent DEBRAUWER and Fien VAN DER HEYDE. Uml 2.5 initiation, exemples et exercices corrigés, 2016.
- [22] OpenJS Foundation. <https://nodejs.org/fr/about/>. en ligne ; visité : 25/06/2022.
- [23] Stefan Tilkov and Steve Vinoski. Node.js : Using javascript to build high-performance network programs, 2010.
- [24] <https://docs.npmjs.com/about-npm>. en ligne ; visité :24/06/2022.
- [25] Sequelize. <https://sequelize.org/docs/v6/>. en ligne ; visité :02/08/2022.
- [26] Facebook (meta). <https://graphql.org/learn/>. en ligne ; visité :02/08/2022.
- [27] Sun Microsystems et Oracle MySQL AB. <https://docs.oracle.com/en-us/iaas/mysql-database/index.html>. en ligne ; visité :02/08/2022.
- [28] Evan You (). <https://vuejs.org/guide/introduction.html>. en ligne ; visité :02/08/2022.
- [29] Vuetify. <https://vuetifyjs.com/en/introduction/why-vuetify/>. en ligne ; visité :02/08/2022.

[30] axios. <https://axios-http.com/fr/docs/intro>. en ligne ; visité :02/08/2022.

[31] Microsoft. <https://code.visualstudio.com/docs>. en ligne ; visité :02/08/2022.

[32] <https://cli.vuejs.org/guide/>. en ligne ; visité :02/08/2022.

Règles de passage du diagramme de classes au modèle logique

Cette section présente les règles permettant de décrire un schéma logique dans les modèles relationnel et objet-relationnel à partir d'un diagramme de classes UML.

A.1 Transformation des entités/classes

Chaque entité devient une relation. L'identifiant de l'entité devient la clé primaire de la relation. Chaque classe du diagramme UML devient une relation. Il faut choisir un attribut de la classe pouvant servir d'identifiant. Si aucun attribut ne convient en tant qu'identifiant, un attribut doit être ajouté afin que la relation ait une clé primaire (outils qui permettent l'ajout de ces attributs).

A.2 Transformation des associations

Les règles de transformation que nous allons voir dépendent des cardinalités/multiplicités maximales des associations. Nous distinguons trois familles d'associations :

- un-à-plusieurs.
- plusieurs-à-plusieurs ou classes-associations, et n-aires.
- un-à-un.

A.2.1 Associations un-à-plusieurs

L'attribut de type clé étrangère doit être ajouté dans la relation fils de l'association. L'attribut qui porte le nom de la clé primaire de la relation parent de l'association.

A.2.2 Associations plusieurs-à-plusieurs

L'association devient une relation dont la clé primaire est composée par la concaténation des identifiants des entités (classes) connectés à l'association. Chaque attribut devient clé étrangère. Les attributs de l'association (classe-association) doivent être ajoutés à la nouvelle relation. Ces attributs ne sont ni clé primaire, ni clé étrangère.

A.2.3 Associations un-à-un

Il faut ajouter un attribut clé étrangère dans la relation dérivée de l'entité ayant la cardinalité minimale égale à zéro. Dans le cas de UML, il faut ajouter un attribut clé étrangère dans la relation dérivée de la classe ayant la multiplicité minimale égale à un. L'attribut porte le nom de la clé primaire de la relation dérivée de l'entité (classe) connectée à l'association. Si les deux cardinalités (multiplicités) minimales sont à zéro, le choix est donné entre les deux relations dérivées de la règle R1. Si les deux cardinalités minimales sont à un, il est sans doute préférable de fusionner les deux entités (classes) en une seule.