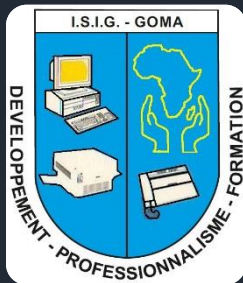


MEMOIRE



INSTITUT SUPERIEUR
D'INFORMATIQUE ET DE GESTION
BP : 841 GOMA
www.isig.ac.cd

E-mail: info@isig.ac.cd

REPUBLIQUE DEMOCRATIQUE DU CONGO
ENSEIGNEMENT SUPERIEUR ET UNIVERSITAIRE

E.S.U

***DEVELOPPEMENT D'UNE API D'ECHANGE
DES DONNEES ENTRE LES SITES WEB
MARCHANDS PARTENAIRES D'UNE
ENTREPRISE COMMERCIALE,
CAS DE MALEKANI SARLU***

Par KAVIRA MALEKANI SHEKINAH

Directeur
CT AMANI HAGUMA Joseph
Encadreur
CT SERGE KIKOBYA

Mémoire présenté et défendu en vue de
l'obtention de Diplôme de Licence en
Informatique
Option : Conception des systèmes
d'information

Année Académique

2021 - 2022

SOMMAIRE

SOMMAIRE	i
EPIGRAPHE	iii
DEDICACE.....	iv
REMERCIEMENTS	v
SIGLES ET ABREVIATIONS	vi
LISTE DES TABLEAUX	vii
LISTE DES FIGURES	viii
RESUME ET MOTS CLES	ix
ABSTRACT AND KEYWOR.....	x
INTRODUCTION GENERALE.....	1
I.1. ETAT DE LA QUESTION	1
I.2. PROBLEMATIQUE	3
I.3. HYPOTHESES	4
I.4. OBJECTIFS	4
I.5. METHODE ET TECHNIQUE	4
I.6. DELIMITATION DU SUJET.....	Erreur ! Signet non défini.
I.7. CHOIX ET INTERET DU SUJET	6
I.8. SUBDIVISION	7
Chapitre 1 : DEFINITION DE TERMES ET CONCEPTS D’ETUDE.....	8
INTRODUCTION.....	8
SECTION 1 : DEFINITION DE CONCEPTS UTILISES	8
SECTION 2 : CONTEXTE D’ETUDE.....	26
CONCLUSION	30
Chapitre 2. REVUE DE LITTERATURE	31
INTRODUCTION.....	31
SECTION 1 : TRAVAUX SIMILAIRES	32
SECTION 2 : METHODE DES RESOLUTION	40
CONCLUSION	42
Chapitre 3 : METHOLOGIE ET RESULTATS	43
INTRODUCTION.....	43
Section 1. METHODOLOGIE	43
SECTION 2 : RESULTATS ET ANALYSE.....	49
CONCLUSION	60

CHAPITRE IV : DISCUSSION ET RECOMMANDATIONS	61
Introduction	61
SECTION 1 : DISCUSSION DES RESULTATS	61
SECTION 2 : RECOMMANDATION	64
CONCLUSION	64
CONCLUSION GENERALE	65
BIBLIOGRAPHIE	66

EPIGRAPHE

« Tout obstacle renforce la détermination. Celui qui s'est fixé un but n'en change pas »

LEONARD DE VINCI

DEDICACE

A nos parents **Alphonse SIKUMBILI** et **DADA KISIRANI**;

A nos frères **Elie, Bam's ,Jephté , Baron** et **Jean-Maradi MALEKANI**

A nos sœurs **Evodie** et **El-Bénédicté MALEKANI**;

A nos amis, camarades et connaissances.

KAVIRA MALEKANI Shekinah

REMERCIEMENTS

Le présent travail est l'œuvre d'une combinaison de plusieurs apports des différentes personnes qui nous ont aidé pendant son élaboration. C'est pourquoi nous ne pouvons pas mettre une conclusion à ce travail sans pour autant penser à les remercier. C'est ainsi que, nous tenons à remercier de loin ou de près tous ceux-là qui nous ont apporté assistance pour la réalisation de ce travail.

Avant toute chose, nous remercions notre Dieu tout puissant qui nous assiste jours et nuits, pour nous avoir donné l'intelligence de réussir aux épreuves auxquelles nous étions soumis pour tester nos capacités. Nous ne sommes pas bon, plus intelligent que ceux-là qui ne sont pas arrivés à ce niveau mais Dieu nous a fait grâce.

Nos hommages et remerciements aux autorités académiques et au corps professoral et administratif de l'institut supérieur d'informatique et de gestion / ISIG GOMA.

Nos remerciements sont adressés au directeur de ce mémoire le Chef de Travaux **AMANI HAGUMA Joseph** pour avoir accepté de diriger ce mémoire malgré ses multiples fonctions. Ensuite, nous remercions l'encadreur de ce mémoire le chef de travaux **SERGE KIKOBYA** pour nous avoir encadré et surtout pour toutes ses reproches, ses efforts fournis et remarques.

Gratitude à notre famille, les **MALEKANI** et a monsieur **MP** dont le soutien tant moral, spirituel, matériel, que financier a toujours été d'une aide précieuse.

Nous remercions vivement papa **SHADRACK GONA** pour nous avoir pris en charge durant notre cursus académique.

Nous remercions de tout cœur monsieur **ABIO BAMONGOYO Gaétan** pour le soutien qui nous a été d'un plus grand réconfort.

KAVIRA MALEKANI Shekinah

SIGLES ET ABBREVIATIONS

API : interface de programme d'application (Application Programming Interface)

BD: Base des données

B2B: business to business

B2C: business to consumer

CSS: cascading style sheets

HTML: Hyper-Text Markup Language

HTTP: Hyper Text Transfer Protocol

HTTPS: Hyper Text Transfer Protocol security

ISIG : Institut Supérieur d'Informatique et de Gestion

IP : internet Protocol

JS: Java Script

JSON : JavaScript Object Notation

RESTAPI: restful Application Programming Interface

PHP: HyperText Processor

RDC: République Démocratique du Congo

RUP: Rational Unified Process

SARLU: société à responsabilité limitée unipersonnelle

SQL: Structure Query Langage

SGBD : Système de Gestion des Bases des données

SDLC : le cycle de vie du développement d'un logiciel

TCP/IP: Transfer Control Protocol/Internet Protocol

UML: Unified Modeling Language

LISTE DES TABLEAUX

1. TABLEAU : TRAVAUX SIMILAIRES	32
2. TABLEAU : ARTICLES SIMILAIRES	36
3. TABLEAU : IDENTIFICATION ET DEROULEMENT DES TACHES	43
4. TABLEAU : ESTIMATION DE COUTS DE LA CONCEPTION	44

LISTE DES FIGURES

Figure 1 : Architecture	42
Figure 2 : Liste des tâches	46
Figure 3 : Réseau de tâche.....	47
Figure 4 : Diagramme de GANTT	48
Figure 5 : Cas d'utilisation	49
Figure 6 : Diagramme de déploiement	51
Figure 7 : Architecture	52
Figure 8 : affichage de l'api par le client (get_all)	56
Figure 9 : affichage de l'api par le client (get_only)	57
Figure 10 : affichage de dans le navigateur.....	58
Figure 11 : affichage de dans le navigateur.....	59

RESUME ET MOTS CLES

Notre travail de Mémoire s'intéresse à l'étude et développement d'une API d'échange des données entre les sites web marchands partenaires d'une entreprise commerciale, MALEKANI sarlu.

Les transformations digitales représentent un véritable challenge, dans l'ère digitale actuelle entre les différents types de e-commerce et la multiplication des sites. les entreprises sont perdues face à la gestion ou l'échange des données entre deux ou plusieurs sites. La baisse de fréquentation des clients sur les sites, la mauvaise rentabilité, peu de visibilité sont des problèmes communs aux entreprises incompetentes face à la centralisation de données de plusieurs sites partenaires. C'est le cas de MALEKANI sarlu. Cette étude vise à résoudre les problèmes liés à la gestion et l'échange des données entre deux ou plusieurs sites.

Pour le cas de notre étude, nous allons développer une API qui permettra de centraliser les contenues et faciliter l'échange des données entre les sites e-commerces de partenaires digitales de l'entreprise pour une bonne visibilité des produits. Cette étude sera analysée par la méthode RUP du langage de modélisation UML et la méthode GANTT de suivi et évolution de projet.

Mots clés : API, B2B, e-commerce, B2C

ABSTRACT AND KEYWOR

Our Memory work focuses on the study and development of an API for exchanging data between merchant websites that are partners of a commercial company, MALEKANI sarlu.

Digital transformations represent a real challenge, in the current digital era between the different types of e-commerce and the proliferation of sites. companies are lost when it comes to managing or exchanging data between two or more sites. The drop in customer traffic on the sites, poor profitability, little visibility are common problems for incompetent companies when faced with the centralization of data from several partner sites. This is the case of MALEKANI sarlu. This study aims to solve problems related to the management and exchange of data between two or more sites.

For the case of our study, we are going to develop an API that will centralize the content and facilitate the exchange of data between the e-commerce sites of the company's digital partners for good product visibility. This study will be analyzed by the RUP method of the UML modeling language and the GANTT method for project monitoring and development.

Keywords: API, B2B, e-commerce, B2C

INTRODUCTION GENERALE

En informatique, une interface de programmation d'applications ou interface de programmation applicative souvent désignée par le terme API pour dire « Application Programming Interface » est un ensemble normalisé de classes, de méthodes, de fonctions et de constantes qui sert de façade par laquelle un logiciel offre des services à d'autres logiciels. Elle est offerte par une bibliothèque logicielle ou un service web, le plus souvent accompagnée d'une description qui spécifie comment des programmes consommateurs peuvent se servir des fonctionnalités du programme fournisseur.

Les API sont apparues à l'aube de l'informatique, avant même les ordinateurs personnels. À cette époque, elles étaient surtout utilisées en tant que bibliothèques pour les systèmes d'exploitation. Elles résidaient presque toutes en local sur les systèmes sur lesquels elles s'exécutaient, même si elles transféraient parfois des messages entre les mainframes. Presque 30 ans après, les API sont sorties de leurs environnements locaux. Au début des années 2000, elles sont devenues importantes pour l'intégration des données à distance.

On parle d'API à partir du moment où une entité informatique cherche à agir avec ou sur un système tiers, et que cette interaction se fait de manière normalisée en respectant les contraintes d'accès définies par le système tiers. On dit alors que le système tiers « expose une API ».

Pourquoi développer une API d'échange des données entre les sites ? Nous voyons que cela est sans grande surprise, les API sont une tendance qui depuis ses débuts (les années 90) ne cesse de croître. Beaucoup de très grosses entreprises se sont lancées dans la digitalisation. Le e-commerce est depuis devenu une véritable source de profit pour beaucoup des entreprises, qu'elles soient spécialisées dans la vente de produits ou de services. Même s'il est possible de percevoir cette croissance fulgurante, il n'est pas nécessairement évident de comprendre les tenants et aboutissants de cette évolution. Il s'agit d'ailleurs d'une des missions de ce mémoire.

I.1. ETAT DE LA QUESTION

Ce mémoire étant un travail de recherche en sciences informatiques, nous avons souhaité nous munir d'outils y afférant. Mais notre méthodologie n'est pas unilatérale, elle ne se cantonne pas systématiquement à un même point de vue. Finalement, ce que ce mémoire cherche à

démontrer, c'est que pour une entreprise commerciale se trouvant avec deux ou plusieurs site e-commerce leur appartenant à la capacité de non seulement gérer et centraliser leur donnée mais aussi d'améliorer la navigation chez les différents types des clients.

Munis de ces outils, nous allons proposer une solution au problème de mettre en place (développer) une API pour faciliter l'échange des données entre les sites web marchands des partenaires d'une entreprise commerciale. Ce mémoire sera entre de répondre à la demande de l'entreprise MALEKANI sarlu en leur offrant une solution informatique qui leur permet non seulement de centraliser les contenus de leurs sites mais aussi de faciliter l'échange des données entre eux et améliorer la navigation chez les différents types des clients tout en rentabilisant celui-ci.

Une première partie rappellera le contexte d'émergence du concept API, ainsi que le dépassement de cette notion dans le temps. Une seconde partie s'attachera plus particulièrement à nos deux sites e-commerce (B 2B et B2C), mais aussi l'apport des API dans la vie courante et dans la programmation en RCD. Enfin, une troisième partie s'intéressera au développement de l'API d'échange des données entre les sites web (les e-commerce).

C'est dans ce contexte que s'inscrit notre travail intitulé : « développement d'une API d'échange des données entre les sites web marchands des partenaires d'une entreprise commerciale ».

Étant donné que notre sujet n'est pas une innovation à proprement parlé, nous avons trouvé certains travaux en rapport avec notre sujet de recherche, dont en premier lieu le travail réalisé les années passées par nos aînés. Voici les différents travaux que nous avons pu trouver et que nous détaillerons dans la partie concernant la revue de littérature :

- « **Étude et mise place d'un module automatisé d'échange des données entre l'université de Lomé et l'université de para.** » par monsieur KOLANI TAMBA en 2019[1].
- « **Conception et mise en œuvre d'une plate-forme pour la sûreté de fonctionnement des services Web.** » par monsieur Nicolas Salatge, en 2020[2].

Dans ses deux travaux on nous montre qu'il est possible d'échanger des données non seulement entre un ou plusieurs sites mais aussi entre plusieurs partenaires pour une tierce entreprise.

Pour anticiper, nous nous sommes résumés sur « le développement d'une API d'échange des données entre les sites web marchands des partenaires d'une entreprise commerciale, cas de MALEKANI sarlu ». Partant des travaux ci-haut cités, notre sujet sera différent vu le contexte de l'administration des ressources, le partage, la sécurisation des données et le cas d'étude d'une structure non étatique au sein d'une ville voire même une province.

I.2. PROBLEMATIQUE

La problématique est définie comme étant un ensemble de questions qu'une science, une philosophie, qu'un chercheur peut valablement se poser en fonction de ses moyens, de son objet et de ses points de vue. Concrètement, elle représente les points d'interrogations au sujet d'un problème préoccupant un chercheur et dont il désire définir les champs des connaissances théoriques[3].

Il sied de préciser que dans une entreprise commerciale, la visibilité est l'aspect principal qui conduit à une meilleure rentabilité, et c'est pour cela que majorité d'entreprises fait recours à des sites web, mais qui jusque-là ne résolvent pas toutes les difficultés liées à la visibilité et partage de données. C'est le cas de MALEKANI sarlu qui dispose de deux site e-commerce B2B et B2C, mais qui ne communiquent pas. Cela conduit à plusieurs situations négatives dont : la baisse de fréquentation des clients sur les sites, la mauvaise rentabilité, peu de visibilité et les 75% des clients qui doutent de l'authenticité des produits de MALEKANI sarlu.

En plus des problèmes ci-haut évoqués, nous avons remarqué les problèmes ci-après :

- Une baisse des fidélisations des clients ;
- Pas d'échange de donnée entre les sites e-commerce de partenaire digitales qui veulent promouvoir les produits de l'entreprise ;
- Une fréquentation non équilibrée de 15% de clients sur le site de l'entreprise, suite à ça elle a une baisse au niveau financier.

De ce qui précède, nous sommes amenés à centrer notre problématique autour de la question ci-après : « quelle solution applicative pouvons-nous développer pour interconnecter les sites e-commerces de MALEKANI Sarlu et aussi ceux de ses partenaires pour l'échange des données ? ».

I.3. HYPOTHESES

Une hypothèse est une proposition de réponse à la question de départ que l'on s'est probablement posée¹. Elle tend à formuler une relation entre des faits significatifs et permet d'interpréter les faits observés afin de donner une signification qui va constituer un élément de théorie.

Partant de la préoccupation au niveau de notre problématique, nous pensons que, le développement d'une API serait la meilleure de solution qui permettrait à l'entreprise de :

- Faciliter les échanges de données entre leurs sites e-commerces et les partenaires ;
- Rendre accessible les produits de l'entreprise par toutes personnes voulant vendre les produits de MALEKANI Sarlu ;
- Améliorer la navigation chez les différents types des clients.

I.4. OBJECTIFS

Partant de la préoccupation au niveau de notre problématique, nous nous sommes fixés comme objectif ; d'offrir à MALEKANI sarlu une interface de programmation d'application en PHP et MYSQL qui va leur permettre de voir les contenus appartenant à un autre site et effectuer les différentes opérations, les clients peuvent faire :

- Voir la liste de tout le produit ;
- Passer la commande de produit ;
- Faire le paiement de la commande ;
- Voir le profil consommateur d'un utilisateur ;

I.5. METHODE ET TECHNIQUE

I.5.1. Méthode

COHENDET définit une méthode comme étant un instrument devant permettre à l'esprit de s'épanouir ; l'expression de s'éclaircir. L'utilisation de la bonne méthode a pour objectif de mettre en valeur la qualité de réflexion ; aussi comme le dit GRAWITZ, la méthode

¹ <https://fr.wikipedia.org/wiki/Hypoth%C3%A8se>

est l'ensemble des opérations intellectuelles par les quelles une discipline cherche à atteindre une vérité qu'elle poursuit, le démontrer et les vérifier.[4]

Dans le but de mieux appréhender les problèmes posés, d'en comprendre la profondeur et proposer des solutions effectives, une approche méthodique du travail est indispensable. L'élaboration de notre mémoire se penche sur le développement d'une API d'échange des données entre deux sites e-commerce. De ce fait le choix des méthodes a porté sur la méthode de prototypage couplée avec la technique de simulation en Html, connecté à une base des données MySQL via un serveur PHP.

- L'outil de modélisation UML : (« Unified Modeling Language », Langage de Modélisation Unifié), nous a aidé pour une étude des besoins et nous a fait aboutir à la proposition d'une solution concrète répondant aux besoins.
- La méthode RUP : (Rational Unified Process) ou Processus rationnel unifié, découlant de UML nous a aidé à donner un cadre précis au développement de notre API. C'est une méthode incrémentale et itérative qui s'adapte très facilement aux processus et aux besoins de développement.
- Enfin le diagramme GANTT : Ce diagramme est un outil efficace au quel nous allons faire recours pour présenter les activités quotidiennes de notre projet de façon chronologique (succession des activités dans le temps) ; elle nous a permis d'établir un calendrier pour le suivi de l'évolution du projet.

I.5.2. Techniques

La technique est un moyen ou un outil à la disposition de la méthode en vue de se saisir de la réalité en question. Dans le cadre de ce mémoire, nous avons ainsi utilisé les techniques suivantes :

- Technique d'interview libre : Elle a pour objet d'organiser l'apport de communication entre deux personnes afin de permettre à l'enquêteur de recueillir certaines informations de l'enquête sur un sujet précis. Elle nous a permis de récolter des réponses au question sur l'entreprise.
- Technique d'observation : Nous a facilité la compréhension de l'environnement dans lequel le service de l'entreprise fonctionne afin d'avoir un aperçu global de sa gestion.
- Technique d'investigation : Par des contacts avec les experts en matière, des professionnels et des spécialistes en Nouvelle Technologie de l'Information et de la

Communication (NTIC), cette technique nous a permis d'acquérir beaucoup d'information utiles à notre travail.

- Technique documentaire : Cette technique nous a facilité la récolter d'informations par la lecture des travaux d'autres chercheurs (Ouvrage, Mémoire, TFC et autres) utiles à l'enrichissement du nôtre.

I.6. DELIMITATION DU SUJET

Tout travail que se veut scientifique doit être examiné dans le temps tout comme dans l'espace, pour bien cerner le contour du sujet et faciliter la démarche scientifique à adopter. Ainsi, le nôtre le sera dans le temps et dans l'espace.

- Délimitation spatiale : Le choix de notre champ d'investigation a été porté à l'entreprise MALEKANI sarlu de Goma
- Délimitation temporelle : Pour notre délimitation temporelle, notre API pourra fonctionner allant du jour de sa mise en usage jusqu'à ce qu'une autre personne propose une solution plus évoluée que la nôtre.

I.7. CHOIX ET INTERET DU SUJET

I.7.1. Intérêt personnel

Cette étude trouve son intérêt dans le fait qu'elle nous aidera à mieux maîtriser, sur le plan pratique, le développement d'une API ou la structuration dans une entreprise, de l'interconnexion entre deux sites.

Ce travail constitue, pour nous, une entrée dans le monde des chercheurs qui placera notre modeste nom dans le monde scientifique.

I.7.2. Intérêt commun

Nous avons choisi ce sujet pour répondre à la demande de l'entreprise MALEKANI sarlu en leur offrant une solution informatique qui leur permet non seulement de centraliser les contenus de leurs sites mais aussi de faciliter l'échange des données entre les deux sites et améliorer la navigation chez les différents types des clients tout en rentabilisant celui-ci.

I.8. SUBDIVISION

Hormis l'introduction et la conclusion, le présent travail est subdivisé en 4 chapitres dont :

- Le Chapitre I. État des lieux et analyse, ici nous allons nous concentrer sur la définition de certains concepts évoqués dans notre travail.
- Le Chapitre II : Revue de la littérature
 Au sein de ce chapitre, nous nous donnons pour tâche de présenter les travaux déjà réalisés et qui cadrent avec notre sujet de recherche.
- Le Chapitre III : Méthodologie et résultats
 Ici nous expliquons le déroulement de la mise en application de la conception en usant des différentes méthodes adoptées et déjà décrites précédemment.
- Le Chapitre IV : Discussions et Recommandations
 Ce chapitre traite premièrement des discussions des résultats en comparant ce qui a été présenté comme travaux dans la revue de littérature et les contributions et améliorations que nous avons effectuées, mais surtout ce que notre solution apporte à l'entreprise et en second lieu les recommandations.

Chapitre 1 : ÉTAT DES LIEUX ET ANALYSE

INTRODUCTION

Dans ce chapitre, nous allons nous concentrer sur la définition de certains concepts évoqués dans notre travail. En définissant ces concepts, l'objectif visé est de permettre à toute personne qui va nous lire de comprendre le fond de notre travail.

SECTION 1 : DEFINITION DE CONCEPTS UTILISES

1. Définition de concepts du sujet

1.1.1. API (Application Programming Interface)

- Selon **Red hat** dans son article « **une API, qu'est-ce que** » Une API, ou interface de programmation d'application, est un ensemble de définitions et de protocoles qui facilite la création et l'intégration de logiciels d'applications[5] ;
- Selon la **Rédaction JDN** dans son article « **Qu'est-ce que une API** » API est l'acronyme d'**Application Programming Interface**, que l'on traduit en français par **interface de programmation applicative** ou **interface de programmation d'application**. L'API est une solution informatique qui permet à des applications de communiquer entre elles et de s'échanger mutuellement des services ou des données. Les interfaces de programmation d'application proposent en général un ensemble de fonctions qui facilitent, via un langage de programmation permettant de lancer des requêtes, l'accès aux services d'une application[6].
- Selon **JobFonning** Une « interface de programme d'application » (API) est un regroupement de routines, de protocoles et d'outils servant un seul objectif, celui de développer des applications logicielles. De base, une API spécifie la méthode par laquelle les composants logiciels doivent interagir. Les API interviennent ainsi au moment de la programmation des composants de l'interface utilisateur graphique (GUI)[7]
- Selon **Daniel Escardo** dans son article « **What Is an API? (The Simple Definition)** » Une API ne doit pas être confondue avec une IPA (qui est délicieuse et mérite d'être écrite, mais pas ici et maintenant). Bien que les deux puissent être acides et vous laisser un peu étourdi, lorsqu'ils sont bien faits, ils peuvent vraiment améliorer votre expérience. API signifie Application Programming Interface. Maintenant, je sais que cela semble très complexe et technique, mais cela signifie simplement une interface, ou une méthode/moyen, pour que

deux logiciels communiquent. Dans cet article, nous allons nous concentrer sur les logiciels en ligne - les sites Web[8].

- Selon **Techopedia** dans son article « **What does Application Programming Interface mean** » Une interface de programmation d'application (API) est un ensemble de protocoles, de routines, de fonctions et/ou de commandes que les programmeurs utilisent pour faciliter l'interaction entre des services logiciels distincts. Les API permettent à un service logiciel d'accéder aux données d'un autre service logiciel sans que le développeur ait besoin de savoir comment fonctionne l'autre service. Par exemple, les API ouvertes du service postal américain permettent aux développeurs de commerce électronique d'ajouter le suivi des colis à leurs sites Web afin que les clients sachent quand s'attendre à la livraison[9].
- Selon **Alyssa Walicer** dans son article « **What is an API ? full form, meaning, types and exemple** » L'interface de programmation d'application (API) est une interface logicielle qui permet à deux applications d'interagir l'une avec l'autre sans aucune intervention de l'utilisateur. API est un ensemble de fonctions et de procédures logicielles. En termes simples, API signifie un code logiciel accessible ou exécuté. L'API est définie comme un code qui aide deux logiciels différents à communiquer et à échanger des données entre eux[10].

D'après toutes les définitions des différents auteurs, nous pouvons définir une api d'une manière simplifier, Est une **interface applicative de programmation** qui permet d'établir des connexions entre plusieurs logiciels pour échanger des données.

1.1.2. Sites web

- Selon **Gabriel Dabi-Schwebel** dans son article « **site web** » Un site web peut être défini comme étant un ensemble de pages web reliées entre elles par des liens hypertextes, accessibles au travers du réseau Internet à l'aide d'une adresse web (url) à partir d'un même nom de domaine. On utilise parfois le terme site Internet, qui est équivalent (Internet est le réseau qui sert de support au web, mais la nuance n'a pas une importance fondamentale dans ce cas)[11].
- Selon **Fédéric Canevet** dans son article « **Définition d'un site web** » La définition d'un site Web (website en anglais) ou site Internet est un ensemble de pages Web reliées entre elles par des liens internes, visualisable depuis un navigateur Web[12].

- Selon **Seiler Laurent** dans son article « **Qu'est-ce qu'un site web** » Techniquement, un site internet ce n'est qu'un fichier ou un ensemble de fichiers, hébergés sur un serveur (une sorte de gros ordinateur connecté à internet) et accessible via internet. On le voit cette définition est large. Une simple page html, une image ou un formulaire peuvent tout à fait être des sites internet. Ou plutôt : un site internet peut n'être composé que d'une simple page html, d'une image ou d'un formulaire. Mais ce qui compte avec un site internet, ça n'est pas tellement ce que c'est, mais plutôt ce que l'on en fait. Car **un site web, c'est avant tout un outil**. Un outil de communication, de prospection, de vente, bien sûr, mais aussi d'échange de données, de travail collaboratif, de transmission d'informations, de comptabilité, etc. Voilà pourquoi nous avons commencé cette fiche en utilisant la définition la plus large : car un site internet, cela peut être absolument ce que vous voulez. Ce qui le définira en dernier lieu, c'est le fait d'être accessible à distance grâce à une connexion internet et une adresse URL (Uniform Resource Locator)[13].
- Selon **Marie** dans son article « **Tout savoir sur la notion de site internet** » Un site Internet, aussi communément appelé site web, est un groupe de pages web accessible depuis un navigateur Internet. Les pages sont reliées entre elles par des liens. Ces liens permettent de naviguer dans le site et de passer d'une page à l'autre. Aussi, un lien, également appelé adresse web ou URL est généralement associé à un nom de domaine. Celui-ci permet de remplacer l'adresse IP qui donne accès au serveur web, là où l'ensemble des fichiers d'un site Internet sont stockés[14].
- Selon **Corinne Bon** dans son article « **Qu'est-ce qu'un site web ? définition** » C'est un ensemble de pages web regroupées et généralement reliées entre elles de plusieurs façons. La section des sciences est l'équivalent d'un site web[15].
- Selon **Alexandre MAROTEL** dans son article « **Définition website ou site web** » Un site web est un ensemble de pages web ordonnées qui représente généralement l'image d'une personne physique ou morale (entreprise, startup, association, freelancer, etc.) sur Internet. Ces différentes pages sont interconnectées entre elles par des liens encore appelés URL (Uniform Resource Locator) et qui permettent de naviguer sur le site[16].

D'après toutes les définitions des acteurs cités ci-haut, nous définissons un site web étant un ensemble de pages web et de ressources reliées par des hyperliens, défini et accessible par une adresse web.

1.1.3. E-commerce

- Selon **Glossaire** dans son article « **E-commerce** » Dans l'absolu, l'e-commerce, c'est-à-dire le commerce électronique, désigne simplement l'achat et la vente de produits et services sur Internet. Cependant, le terme est souvent utilisé pour décrire toutes les mesures et actions entreprises par un vendeur dans le but de vendre des produits directement aux consommateurs. Le processus démarre dès qu'un client potentiel découvre un produit, l'achète et l'utilise jusqu'à, idéalement, devenir un client fidèle. Les données sont le moteur qui sous-tend toutes les opérations d'e-commerce les plus réussies, en tirant parti de diverses bonnes pratiques telles que l'e-mail marketing ciblé, la segmentation des audiences et l'automatisation marketing. Par exemple, lorsque vous vendez en ligne, effectuer le suivi d'un client ayant ajouté un article à son panier sans terminer la transaction peut augmenter considérablement la probabilité que la vente soit effectuée[17].
- Selon **B. Bathelot** dans son article « **E-commerce** » Le e-commerce ou commerce électronique regroupe l'ensemble des transactions commerciales s'opérant à distance par le biais d'interfaces électroniques et digitales. Le e-commerce englobe donc essentiellement les transactions commerciales s'effectuant sur Internet à partir des différents types de terminaux (ordinateurs, tablettes, smartphones, consoles, TV connectées) sur des sites e-commerce ou applications mobiles marchandes[18].
- Selon **Wesley Drai, Brian Holak and Ben cale** dans son article « **E-commerce** » Le commerce électronique (commerce électronique) est l'achat et la vente de biens et de services, ou la transmission de fonds ou de données, sur un réseau électronique, principalement Internet. Ces transactions commerciales se produisent soit en tant qu'entreprise à entreprise (B2B), entreprise à consommateur (B2C), consommateur à consommateur ou consommateur à entreprise. Les termes e-commerce et e-business sont souvent utilisés de manière interchangeable. Le terme e-tail est également parfois utilisé en référence aux processus transactionnels qui composent le commerce de détail en ligne[19].
- Selon **Andrew Bloomenthal** dans son article « **Electronic commerce** » Le terme commerce électronique (commerce électronique) fait référence à un modèle d'affaires qui permet aux entreprises et aux particuliers d'acheter et de vendre des biens et des services sur Internet. Le commerce électronique opère dans quatre segments de marché principaux et peut être effectué sur des ordinateurs, des tablettes, des smartphones et

d'autres appareils intelligents. Presque tous les produits et services imaginables sont disponibles via des transactions de commerce électronique, y compris les livres, la musique, les billets d'avion et les services financiers tels que l'investissement en actions et les services bancaires en ligne[20].

- Selon **Jonn Vande Zande** dans son article « **Whats is an e-commerce** » Le commerce électronique est l'achat et la vente de biens ou de services via Internet, et le transfert d'argent et de données pour compléter les ventes. Il est également connu sous le nom de commerce électronique ou commerce sur Internet[21].
- Selon **Wikipedia**, le e-commerce, ou commerce électronique, est « **l'échange pécuniaire de biens, de services et d'informations par l'intermédiaire des réseaux informatiques, notamment Internet.** » En d'autres termes, il s'agit d'un commerce qui gère les paiements grâce à des moyens électroniques[22].

1.1.4. Entreprise commerciale

- Selon **JDN Rédaction**, une entreprise commerciale ou une société commerciale exerce ce type d'activité dans un but lucratif ou avec une forme juridique de nature commerciale. Il peut s'agir de transactions de biens immobiliers ou mobiliers, de produits, comme de services. Par définition, la législation en vigueur reconnaît automatiquement le caractère commercial d'une entreprise. Cela comprend les banques, mais aussi les entreprises assurant des prestations de transport ou de location, par exemple. Par opposition, on différencie la société commerciale de la société civile où la responsabilité du dirigeant et des associés demeure illimitée.

La société commerciale est considérée comme une personne morale. S'il existe des cas où elle est une société de personnes ou hybride, elle est le plus souvent une société de capitaux. Ces derniers se composent d'apports numéraires (financiers) et en nature (local professionnel, compétences, matériel...). Le capital est partagé entre les associés ou, le cas échéant, détenu par l'unique dirigeant. Il se traduit alors par des parts sociales ou des actions. L'immatriculation de la société commerciale est obligatoire et s'effectue auprès du RCS (Registre du commerce et des sociétés). La création d'une société commerciale permet de mieux gérer son patrimoine tout en limitant les risques financiers et juridiques[23].

- Selon **Belgiki** Les sociétés commerciales sont des sociétés constituées par un contrat aux termes duquel deux ou plusieurs personnes mettent quelque chose en commun, pour

exercer une ou plusieurs activités déterminées et dans le but de procurer aux associés un bénéfice patrimonial direct ou indirect (art. 1er du Code des Sociétés)[24].

- Selon **le coin des entrepreneurs** dans son article **le choix de création des entreprise**, une entreprise commerciale ou société commerciale est une société qui exerce une **activité commerciale** ou une société qui revêt une **forme juridique commerciale**[25].
- Selon **Mathieu George** dans son article **Qu'est-ce qu'une entreprise commerciale** Une société commerciale est une société qui exerce une activité commerciale ou dispose d'une forme juridique de nature commerciale. C'est une structure dotée d'une personnalité morale et régie par le Code de commerce. Les règles et le fonctionnement d'une société commerciale sont définis dans les statuts. D'une manière générale, elle est formée par au moins deux associés qui se partagent les bénéfices[26].
- Selon **Thomas Lailier** dans son article **Qu'est-ce qu'une société commerciale**, par principe, une société compte au moins deux personnes (physique ou morales), mais par exception la loi peut autoriser la création de société unipersonnelle (EURL, SASU, ...). Le but de toute société est fixé par la loi : dégager et partager des bénéfices entre associés.
 - La forme possible que peut prendre la société commerciale est limitée par la loi, un caractère commercial à certains types de sociétés :
 - Les sociétés par actions : société par actions simplifiée (SAS), société par actions unipersonnelles (SASU) et sociétés anonymes (SA) ;
 - Les sociétés à responsabilité limitée (SARL) et entreprises unipersonnelles à responsabilité limitée (EURL) ;
 - Les sociétés en commandite simple (SCS) ;
 - Les sociétés en nom collectif (SNC).
 - L'objet, c'est-à-dire l'activité entreprise, permet aussi de définir une société commerciale (achat pour revente, transport, construction, banque...). La liste est donnée également par le code de commerce[27]

I.1.5. B2C

- Selon **Moovago** dans son article **c'est quoi le B2C, Définition, exemple, différence**, le B2C est un type de business le plus fréquent et qui est très commun pour toutes les personnes. Les entreprises vendent des produits et proposent des prestations de service qui répondent spécialement aux besoins de la population qui sont des particuliers et non des entreprises[28].
- Selon **Pure illusion** dans son article **Définition de B2C**, Le terme B2C signifie "business to consumer". C'est une formulation que l'on retrouve dans le marketing, car elle regroupe l'ensemble des actions commerciales et marketing qui sont réalisées entre une entreprise et un particulier qui est le consommateur final[29].
- Selon **La rédaction JDN** dans son article **B2C ou B to C : définition simple et traduction de business to consumer**, C'est un jargon professionnel très utilisé. L'abréviation B to C désigne l'ensemble des relations qui unissent les entreprises et les consommateurs finaux. Le B to C peut aussi s'écrire B2C. Autrement dit, ce type de commerce recouvre les marchandises ou les prestations conçues pour le grand public[30].
- Selon **AUDREY TIPS** dans son article **Business to Consumer (B2C)**, Pour faire simple, en B to C, la cible est le consommateur final et non un professionnel ou une entreprise. Les acheteurs sont individuels et n'ont pas de connaissances spécialisées. En outre, ils investissent leur propre argent[31].
- Selon **l'info net** dans son article **B2C**, Le B2C (ou "BtoC", ou "business to consumer") désigne toutes les activités commerciales destinées aux clients finaux. L'acronyme B2C fait référence au marché de la consommation de masse, et donc aux entreprises qui offrent leurs services ou produits directement aux utilisateurs finaux[32].
- Selon **Digital guide** dans son article **B2C : business to consumer** L'acronyme « B2C » signifie « business to consumer », c'est-à-dire les relations commerciales entre les entreprises et les consommateurs. Ceci inclut également toute la communication des entreprises avec des clients privés potentiels ou existants, c'est-à-dire également le marketing orienté vers le consommateur[33].

I.1.6. B2B

- Selon **Ooreka** dans son article **B2B**, Le B to B désigne l'activité commerciale interentreprises. Il génère des volumes d'activité économique supérieurs à ceux de l'activité commerciale directement tournée vers consommateur final[34].

- Selon **Pure illusion** dans son article **B2B**, Le terme B2B signifie “business to business”. C’est une formulation très présente en marketing, car elle regroupe l’ensemble des actions commerciales et marketing qui sont réalisées entre entreprises[35].
- Selon **JOPTIMISEMONSITE** dans son article **Le BtoB, c’est quoi ? Comprendre le B to B en 2min seulement**, Le **BtoB**, qui est une abréviation du terme **anglais business to business**, est un terme qui est utilisé pour désigner l’ensemble des relations commerciales entre deux entreprises. En français, on peut le traduire par « commerce interentreprises »[36].
- Selon **Glossaire** dans son article **B2B**, Ensemble des relations commerciales entre deux entreprises. On le rencontre parfois sous la mention " commerce interentreprises. " Les relations B to B se déroulant entre professionnels, elles ne portent pas sur les mêmes variables et les mêmes facteurs que les relations B to C, même si le prix est devenu un critère essentiel en raison d'une certaine généralisation de la qualité[37].
- Selon **Wikipédia** dans son article **Business to business**, Le marketing B to B représente l’ensemble des relations commerciales entre les entreprises et les professionnels (entreprises, administrations, artisans, professions libérales, associations...). On l’appelle aussi « commerce interentreprises ». Le marketing industriel (échange de biens industriels) et le marketing d’affaires (projet, solution, clés en main) sont des sous-ensembles du marketing B to B[38].
- Selon **La Redaction JDN** dans son article **B2B ou B to B : définition simple et traduction de business to business**, C'est quoi le B to B ? L'abréviation B to B désigne l'ensemble des activités commerciales nouées entre deux entreprises. Le terme anglais complet est business to business. Il peut aussi s'écrire également B2B ou BtoB. De manière générale, le B to B concerne tous les moyens utilisés pour mettre en relation ces sociétés et faciliter les échanges de produits, de services et d'informations entre elles[39].

2. Définition de technologie utilisées

2.1.1 FONT-END

En développement web, la notion de front-end fait référence à l’ensemble des éléments visibles et accessibles directement sur un site web (voire sur une application web ou une application web mobile), le langage utilisé pour le front end est appelle langage de balise. Dans notre travail nous allons utiliser le HTML, le CSS et le Framework Bootstrap.

1. HTM

HTML (hypertexte markup language) n'est pas un langage de programmation mais C'est un langage de balises qui définit la structure de votre contenu. HTML se compose d'une série d'éléments, utilisés pour entourer, ou envelopper, les diverses parties du contenu pour les faire apparaître ou agir d'une certaine façon[40]. Il s'agit plus précisément d'un format de données utilisé dans l'univers d'Internet pour la mise en forme des pages Web. Il permet, entre autres, d'écrire de l'hypertexte, mais aussi d'introduire des ressources multimédias dans un contenu.

L'HTML est ce qui permet à un créateur de sites Web de gérer la manière dont le contenu de ses pages Web va s'afficher sur un écran, via le navigateur. Il repose sur un système de balises permettant de titrer, sous-titrer, mettre en gras, etc., du texte et d'introduire des éléments interactifs comme des images, des liens, des vidéos... L'HTML est plus facilement compris des robots de crawl des moteurs de recherche que le langage JavaScript, aussi utilisé pour rendre les pages plus interactives[41].

Le HTML repose sur une syntaxe simple et rigide, dont l'unité de base est la balise. Les balises vont toujours par deux, une pour ouvrir l'élément, l'autre pour le fermer. Elles se construisent toujours de la même façon :

- Un chevron < pour ouvrir la balise
- Le nom de l'élément contenu dans la balise
- Un attribut, si nécessaire, pour qualifier l'élément.

Le htm en conclusion est une Langage de balisage utilisé pour la création de pages web, permettant notamment de définir des liens hypertextes[42]

2. CSS

CSS (Cascading Style Sheets) est un langage de mise en forme des documents il permet de créer des pages web à l'apparence soignée. Le CSS est l'une des trois technologies qui constituent le noyau du web, avec HTML et JavaScript . Le CSS a été développé par un groupe au sein du W3C, nommé le CSS Working Group (ou « groupe de travail CSS » en français). Ce langage est principalement utilisé pour appliquer un style aux éléments HTML , mais peut aussi être utilisé avec d'autres langages de balisage tels que SVG ou encore XML, les documents en question sont des fichiers texte structurés avec un langage de balises[43].

- Le CSS peut être utilisé pour une mise en forme élémentaire des documents (par exemple, changer la couleur et la taille des titres et des liens).
- Il peut être utilisé pour concevoir une maquette (par exemple, transformer un texte affiché sur une colonne en une composition avec un cadre principal et une barre latérale pour les informations reliées.
- Mais aussi nous devons savoir qu'avec CSS, on peut aussi produire des animations.

Le CSS est un langage basé sur des règles, on définit des règles de styles destinées à des éléments ou des groupes d'éléments particuliers dans la page. Par exemple, « Je veux que le titre principal de ma page s'affiche en rouge en gros caractères. » nous ferons :

```
h1 {
    color: red;
    font-size: 5em;
}
```

Le CSS correspond donc à un langage informatique permettant de mettre en forme des pages web (HTML ou XML). Il est composé des fameuses « feuilles de style en cascade » également appelées fichiers CSS (.css) et contient des éléments de codage[44]. L'idée de base des feuilles de style CSS est de séparer la structure d'un document de mise en forme, les attributs que vous allez pouvoir ajouter à votre feuille de style permettra de rendre votre code plus lisible et plus léger. Cela facilitera la mise à jour et Il vous sera plus facile de définir une couleur d'arrière-plan et de pouvoir la changer en quelques clics sur tous vos pages grâce à ce procédé.

3. BOOTSTRAP

En informatique, le verbe anglais « Bootstrap » signifie amorcer, c'est-à-dire charger un programme (généralement un système d'exploitation) dans un ordinateur en utilisant un programme initial beaucoup plus petit (amorce)[45].

Bootstrap est un Framework de frontend gratuit qui devient de plus en plus populaire parmi les développeurs de frontend. Il est facile à utiliser et permet aux développeurs d'économiser beaucoup de temps en évitant d'avoir à écrire manuellement des syntaxes encore et encore.

Le Framework est également très flexible et peut répondre à presque tous les besoins de développement web en frontend. Ses meilleures capacités comprennent, sans s'y limiter, des fonctionnalités réactives qui permettent aux pages web de fonctionner de manière optimale sur toutes les tailles d'écran[46].

Bootstrap.css : est un Framework CSS qui organise et gère la mise en page d'un site web. Alors que le HTML gère le contenu et la structure d'une page web, le CSS s'occupe de la mise en page du site. Pour cette raison, les deux structures doivent coexister pour effectuer une action particulière. Grâce à ses fonctions, le CSS vous permet de créer un aspect uniforme sur autant de pages web que vous le souhaitez. Dites adieu aux heures d'édition manuelle juste pour changer la largeur d'une bordure.

Avec le CSS, il suffit de renvoyer les pages web au fichier CSS. Toute modification nécessaire peut être effectuée dans ce seul fichier.

Les fonctions du CSS ne se limitent pas aux seuls styles de texte car elles peuvent être utilisées pour formater d'autres aspects de la page web tels que les tableaux et les mises en page d'images.

Bootstrap.js : Ce fichier est la partie centrale de Bootstrap. Il est constitué de fichiers JavaScript qui sont responsables de l'interactivité du site web.

Pour gagner du temps en évitant d'écrire de nombreuses fois des syntaxes JavaScript, les développeurs ont tendance à utiliser jQuery. Il s'agit d'une bibliothèque JavaScript multiplateforme à code source ouvert très répandue qui permet d'ajouter diverses fonctionnalités à un site web.

Voici quelques exemples de ce que jQuery peut faire :

- Effectuer des requêtes Ajax comme la soustraction dynamique de données d'un autre emplacement
- Créer des widgets à l'aide d'une collection de plugins JavaScript
- Créer des animations personnalisées en utilisant les propriétés CSS
- Dynamiser le contenu du site

Alors qu'un Bootstrap avec des propriétés CSS et des éléments HTML peut fonctionner parfaitement, il a besoin de jQuery pour créer un design réactif. Sinon, vous ne pouvez utiliser que les parties nues et statiques du CSS.

En conclusion le Bootstrap est un Framework qui offre les composants d'interface utilisateur, des formulaires et des outils JS, ainsi que l'infrastructure d'implémentation. Ce logiciel est disponible au format précompilé ou comme code source.

2.1.2 BACK-END

1. PHP

Le PHP (Hypertext Preprocessor) désigne un langage informatique, ou un langage de script, utilisé principalement pour la conception de sites web dynamiques[47]. Il s'agit d'un langage de programmation sous licence libre qui peut donc être utilisé par n'importe qui de façon totalement gratuite.

Un langage de script est un langage de programmation interprété qui permet de manipuler les fonctionnalités d'un système informatique configuré pour fournir à l'interpréteur de ce langage un environnement et une interface qui déterminent les possibilités de celui-ci. Le langage de script peut alors s'affranchir des contraintes des commandes de bas niveau — prises en charge par l'intermédiaire de l'interface — et bénéficier d'une syntaxe de haut niveau.

Le langage PHP fut créé en 1994 par Rasmus Lerdorf pour son site web. C'était à l'origine une bibliothèque logicielle écrite d'abord en langage Perl, puis en langage C. Il s'en servait pour conserver une trace des visiteurs qui venaient consulter son CV. PHP signifiait alors Personal Home Page tools. Le code source de PHP fut publié par son auteur en 1996. En 1997, deux étudiants, Andi Gutmans et Zeev Suraski, redéveloppèrent le cœur de PHP puis son moteur interne, aboutissant à ce qu'on appelle le Zend Engine.

Depuis les versions du langage se sont succédées, apportant leur lot d'améliorations et corrigeant les (nombreuses) failles de sécurité apparues en chemin. La version 5 de PHP, sortie en 2004, utilise Zend Engine2 et introduit un véritable modèle objet, une gestion des erreurs fondée sur le modèle des exceptions, ainsi que des fonctionnalités de gestion pour les entreprises.[48]

Le langage PHP est souvent associé au serveur de base de données MySQL et au serveur Apache. Avec le système d'exploitation Linux, il fait partie intégrante de la suite de logiciels libres LAMP.

Sur un plan technique, le PHP s'utilise la plupart du temps côté serveur. Il génère du code HTML, CSS ou encore XHTML, des données (en PNG, JPG, etc.) ou encore des fichiers PDF. Il fait, depuis de nombreuses années, l'objet d'un développement spécifique et jouit aujourd'hui une bonne réputation en matière de fiabilité et de performances.

PHP est un langage impératif orienté objet. Il s'agit d'un langage de script interprété côté serveur. Il est considéré comme une des bases de la création de sites web dits dynamiques mais également des applications web. PHP a permis de créer un grand nombre de sites web célèbres, comme Facebook, Wikipédia, etc.

Fonctionnement du PHP

Même s'il peut être utilisé en ligne de commande, PHP est principalement associé à un serveur Web utilisant le protocole HTTP dans le cadre d'une architecture client/serveur.

Un serveur Web en architecture trois tiers est composé d'un système d'exploitation, un serveur HTTP, un langage serveur et enfin un système de gestion de base de données (SGBD), cela constituant une plate-forme.

Dans le cas de PHP comme langage serveur, les combinaisons les plus courantes sont celles d'une plateforme LAMP (pour Linux Apache MySQL PHP) et WAMP (Windows Apache MySQL PHP). Une plate-forme WAMP s'installe généralement par le biais d'un seul logiciel qui intègre Apache, MySQL et PHP, par exemple EasyPHP et WampServer. Il existe le même type de logiciels pour les plates-formes MAMP (Mac OS Apache MySQL PHP), à l'exemple du logiciel MAMP et XAMPP.

En conclusion le PHP est un langage de scripts généraliste et Open Source, spécialement conçu pour le développement d'applications web et pour servir de langage de script coté serveur, ce qui fait qu'il est capable de réaliser tout ce qu'un script quelconque peut faire, comme collecter des données de formulaire, générer du contenu dynamique etc...

2. JS

JS (JavaScript) est un langage de programmation de scripts principalement employé dans les pages web interactives et à ce titre est une partie essentielle des applications web. Avec les langages HTML et CSS, JavaScript est au cœur des langages utilisés par les développeurs web[49]. Le JavaScript a été créé en 1995 par Brendan Eich et intégré au navigateur web Netgateur (navigator). L'implémentation concurrente de JavaScript par Microsoft dans Internet Explorer jusqu'à sa version 9 se nommait JScript, tandis que celle d'Adobe Systems se nommait ActionScript.

JavaScript a été standardisé sous le nom d'ECMAScript en juin 1997 par Ecma International dans le standard ECMA-262. La version en vigueur de ce standard depuis juin 2022 est la 13^e édition.

C'est un Langage Orienté Objet À Prototype les bases du langage et ses principales interfaces sont fournies par des objets. Cependant, à la différence d'un langage orienté objets, les objets de base ne sont pas des instances de classes. En outre, les fonctions sont des objets de première classe. Le langage supporte le paradigme objet, impératif et fonctionnel[50].

JavaScript est un langage de programmation qui permet de créer du contenu mis à jour de façon dynamique, de contrôler le contenu multimédia, d'animer des images, et tout ce à quoi on peut penser.

- Le JavaScript est un langage de programmation qui permet de créer du contenu mis à jour de façon dynamique, de contrôler le contenu multimédia, d'animer des images, et tout ce à quoi on peut penser. Bon, peut-être pas tout, mais vous pouvez faire bien des choses avec quelques lignes de JavaScript.
- Le JavaScript est appliqué à votre page HTML un peu comme le CSS. Les éléments <link> permettent d'appliquer des feuilles de style externes au HTML alors que les feuilles de style internes utilisent les éléments <style>. Pour ajouter du JavaScript à un document HTML, il n'y a qu'une seule façon : avec l'élément <script>. Voyons cela sur un exemple.

Les avantages d'un développement en JavaScript sont nombreux !

- Gain de temps.
- Fluidité et ergonomie.
- Prise en compte de l'expérience utilisateur.

- Compatible avec tous les supports numériques.
- Les inconvénients :

L'interprétation des scripts dépend du navigateur employé et de sa version. Le résultat d'un script pourra donc être différent de ce que l'on a prévu et c'est le code du script qui s'affichera sur la page HTML si le navigateur ne supporte pas la version de Javascript utilisé.

3. REST-API

➤ API

Une API, ou interface de programmation d'application, est un ensemble de définitions et de protocoles qui facilite la création et l'intégration de logiciels d'applications. Une API est un ensemble de définitions et de protocoles qui facilite la création et l'intégration de logiciels d'applications. Elle est parfois considérée comme un contrat entre un fournisseur d'informations et un utilisateur d'informations, qui permet de définir le contenu demandé au consommateur (l'appel) et le contenu demandé au producteur (la réponse). Par exemple, l'API conçue pour un service de météo peut demander à l'utilisateur de fournir un code postal et au producteur de renvoyer une réponse en deux parties : la première concernant la température maximale et la seconde la température minimale.

Les API permettent à votre produit ou service de communiquer avec d'autres produits et services sans connaître les détails de leur mise en œuvre. Elles simplifient le développement d'applications et vous font ainsi gagner du temps et de l'argent.

Les API constituent un moyen simplifié de connecter votre propre infrastructure au travers du développement d'applications cloud-native. Elles vous permettent également de partager vos données avec vos clients et d'autres utilisateurs externes. Les API publiques offrent une valeur métier unique, car elles peuvent simplifier et développer vos relations avec vos partenaires, et éventuellement monétiser vos données (l'API Google Maps en est un parfait exemple).

En résumé, grâce aux API, vous ouvrez l'accès à vos ressources, sans sacrifier le contrôle et la sécurité. Après, c'est vous qui choisissez les ressources que vous souhaitez partager, et avec qui. La sécurité des API est essentiellement une question de bonne gestion de celles-ci, ce qui implique l'utilisation d'une passerelle d'API. La connexion des API et la

création des applications qui utilisent les données ou les fonctionnalités exposées par les API peuvent se faire par l'intermédiaire d'une plateforme d'intégration distribuée qui connecte tout, y compris les systèmes existants et l'internet des objets (IoT).

Nous avons plusieurs types de API à savoir :

- **API privées** : elle n'est utilisable qu'en interne. Cette approche permet de garder un contrôle total sur l'API.
- **API publiques** : elle est accessible à tous. Cette approche autorise les tiers à développer des applications qui interagissent avec votre API et peut devenir source d'innovations.
- **API partenaires** : elle est partagée avec certains partenaires de l'entreprise. Cette approche peut générer de nouveaux flux de revenus sans compromettre la sécurité.

➤ **REST**

Le « Representational State Transfer », ou REST, est un style d'architecture. Cela signifie qu'il n'existe aucune norme officielle qui le régit. L'architecture REST (Representational State Transfer) a été créée par l'informaticien Roy Fielding et selon la définition proposée par Roy Fielding dans sa thèse « Architectural Styles and the Design of Network-based Software Architectures », les API sont RESTful tant qu'elles respectent les six contraintes de conception d'un système RESTful :

- **Architecture client-serveur** : une architecture REST est composée de clients, de serveurs et de ressources et elle traite les requêtes via le protocole HTTP.
- **Serveur stateless** : le contenu du client n'est jamais stocké sur le serveur entre les requêtes. Les informations sur l'état de la session sont, quant à elles, stockées sur le client.
- **Mémoire cache** : la mise en mémoire cache permet de se passer de certaines interactions entre le client et le serveur.
- **Système à couches** : des couches supplémentaires peuvent assurer la médiation dans les interactions entre le client et le serveur. Ces couches peuvent remplir des fonctions supplémentaires, telles que l'équilibrage de charge, le partage des caches ou la sécurité.
- **Code à la demande (facultatif)** : un serveur peut étendre les fonctionnalités d'un client en lui transférant du code exécutable.

- Interface uniforme : cette contrainte est capitale pour la conception des API RESTful et couvre quatre aspects différents :
 - Identification des ressources dans les requêtes : les ressources sont identifiées dans les requêtes et sont séparées des représentations retournées au client.
 - Manipulation des ressources par des représentations : les clients reçoivent des fichiers qui représentent les ressources. Ces représentations doivent contenir suffisamment d'informations pour être modifiées ou supprimées.
 - Messages autodescriptifs : tous les messages renvoyés au client contiennent assez d'informations pour décrire la manière dont celui-ci doit traiter les informations.
 - Hypermédia comme moteur du changement des états applicatifs : après avoir accédé à une ressource, le client REST doit être en mesure de découvrir toutes les autres actions disponibles par des hyperliens.

En conclusion le REST (Representational State Transfer), est un style d'architecture. Cela signifie qu'il n'existe aucune norme officielle qui régit les API web RESTful.

➤ **REST-API**

Une API REST (également appelée API RESTful) est une interface de programmation d'application (API ou API web) qui respecte les contraintes du style d'architecture REST et permet d'interagir avec les services web RESTful[51].

2.1.3 DATA BASE

➤ **Date base**

Une base de données n'est qu'une collection structurée de données qui est organisée pour en faciliter l'utilisation et la récupération, Elle est encore un ensemble d'informations qui est organisé de manière à être facilement accessible, géré et mis à jour. Elle est utilisée par les organisations comme méthode de stockage, de gestion et de récupération de l'informations[52].

Les données sont organisées en lignes, colonnes et tableaux et sont indexées pour faciliter la recherche d'informations. Les données sont mises à jour, complétées ou encore supprimées au fur et à mesure que de nouvelles informations sont ajoutées. Elles contiennent

généralement des agrégations d'enregistrements ou de fichiers de données, tels que les transactions de vente, les catalogues et inventaires de produits et les profils de clients.

➤ **MYSQL**

MySQL est un système de gestion de bases de données relationnelles SQL open source développé et supporté par Oracle[53].

MySQL permet le stockage de données dans des tables séparées. Ces tables permettent de classer les données selon leurs caractéristiques. Elles sont organisées grâce à des lignes et des colonnes, il a été lancé à l'origine en 1995. Depuis, il a connu quelques changements de propriétaire et de gestion, avant de se retrouver chez Oracle Corporation en 2010. Alors qu'Oracle est en charge maintenant, MySQL est toujours un logiciel open source, ce qui signifie que vous pouvez l'utiliser et le modifier librement.

Pour comprendre le fonctionnement de MySQL, il est important de connaître deux concepts liés :

- Base de données relationnelle
- Modèle client-serveur

Base de données relationnelle

Les relations de base de données sont des associations entre des tables qui sont créées à l'aide d'instruction de jointure pour récupérer des données. Le tableau suivant décrit les relations de base de données. Les deux tables ne peuvent avoir qu'un seul enregistrement de chaque côté de la relation.

Lorsqu'il s'agit de stocker des données dans une base de données, il existe différentes approches que vous pouvez utiliser. MySQL opte pour une approche appelée base de données relationnelle.

Une base de données relationnelle est un type de base de données où les données sont liées à d'autres informations au sein des bases des données, vos données sont divisées en plusieurs zones de stockage séparées – appelées tables – plutôt que de tout regrouper dans une seule grande unité de stockage.

1. Modèle client-serveur

Une architecture client-serveur représente l'environnement dans lequel des applications de machines clientes communiquent avec des applications de machines de type serveurs[54].

Un serveur est un appareil utilisé pour accéder à Internet, consulter ses mails, stocker des fichiers et gérer une base de données par exemple.

L'architecture client/serveur désigne un mode de communication entre plusieurs ordinateurs d'un réseau qui distingue un ou plusieurs postes clients du serveur : chaque logiciel client peut envoyer des requêtes à un serveur. Un serveur peut être spécialisé en serveur d'applications, de fichiers, de terminaux, ou encore de messagerie électronique. L'architecture client / serveur est un modèle informatique dans lequel le serveur héberge, fournit et gère la plupart des ressources et services à consommer par le client. Ce type d'architecture a un ou plusieurs ordinateurs clients connectés à un serveur central via une connexion réseau ou Internet.

SECTION 2 : CONTEXTE D'ETUDE

1.2.1. Description de l'environnement d'étude

Dans cette section, nous présenteront notre milieu d'étude et les détails de son fonctionnement afin de nous permettre de comprendre son système existant, dégager les problèmes et proposer des solutions durables aux problèmes trouvés.

1.2.1.1. Présentation de MALEKANI SARLU

MALEKANI Sarlu, est une société de droit congolais œuvrant dans le secteur de prestation de services dont les activités s'étendent dans plusieurs domaines entre autres toutes les opérations financières, commerciales, industrielles, mobilières et immobilières susceptible d'en faciliter la réalisation, l'extension et le développement dont il faut citer :

- Business (Fourniture, Installation et Maintenance des équipements de l'agro-transformation) ;
- Engineering (Etude, Construction et Réhabilitation des travaux de génie civil, bâtiment et Mini centrales hydroélectriques) ;
- Réhabilitation des routes de dessertes agricoles ;
- Nettoyage et entretien des bureaux, bâtiments et édifices ; assainissement ; aménagement des espaces verts ; évacuation des immondices, ...
- Agriculture générale

- Ventes (Achats et ventes de produit) ;

Depuis des siècles, partout dans le monde, certains pays se sont servis de l'agriculture pour soutenir leur croissance économique, et des nations africaines s'en servent également pour sortir de la spirale de la pauvreté. Dans la province du Nord-Kivu, l'agriculture représente 65 % de l'emploi et 75 % du commerce intérieur.

Ce secteur sera donc un vecteur non négligeable de sa croissance économique dans les années à venir. Les petits exploitants agricoles seront la pierre angulaire de cette évolution. Pour les petits exploitants agricoles, certains marchés, à la fois innovants et évolutifs, constituent une promesse de profits plus conséquents.

La population urbaine augmente rapidement et la nourrir impliquera de bénéficier de plus de produits agricoles de meilleure qualité. Les consommateurs urbains auront également de plus en plus besoin de produits agricoles transformés ; la valorisation de la production des exploitants agricoles sera donc cruciale dans les années à venir.

Cela offrira des perspectives lucratives, non seulement aux femmes et jeunes qui travaillent la terre, mais également à une vaste population de travailleurs ruraux, et plus particulièrement aux nouvelles générations.

Le fait d'exploiter ces opportunités consiste, tout d'abord, et cela est important à reconnaître que les petits exploitants agricoles font partie intégrante de l'agrobusiness, quelle que soit la taille de leur exploitation ou son importance.

Malheureusement, en province du Nord-Kivu, un grand nombre de petites exploitations agricoles ne sont ni productives, ni rentables. Deux facteurs significatifs permettent d'expliquer pourquoi elles ne peuvent sortir du cycle de la simple subsistance. Tout d'abord, comme elles n'ont accès ni à la technologie moderne ni à des actifs de production, leur rendement est trop faible pour pouvoir générer des excédents susceptibles d'être commercialisés.

Ensuite, les agriculteurs ne sont pas en mesure d'amener leur production sur les marchés en raison du manque de routes et de l'absence de liens entre la production rurale et les activités économiques en aval, telles que les réseaux de transformation et de commercialisation.

Au Nord-Kivu, l'agriculture et l'agrobusiness doivent subir certaines transformations pour pouvoir répondre aux demandes inhérentes au XIXe siècle.

C'est après ces analyses que notre entreprise a vu le jour. Grâce à notre expérience dans l'agriculture, nous avons remarqué un besoin élevé dans la demande de tomate purée, entraînant un approvisionnement de ce produit du Rwanda voisin.

Notre entreprise a été conçue dans le but de répondre aux attentes de la clientèle dans l'approvisionnement de la tomate purée, du ketchup et de la sauce de tomate dans la ville de Goma.

1.2.1.2. Le problème social

Voyant la vision d'atteindre un bon nombre des personnes dans la vente de produits dans une entreprise, elles ont choisi d'utiliser le commerce internet appelée E-commerce, car cette technologie leur permet d'élargir leur zone géographique dans le commerce et de pouvoir augmenter leur rentabilité.

Après études sur internet, nous avons constaté que beaucoup d'entreprise ont deux ou trois sites e-commerce pour la vente de leur produit et cela constitue un risque pour l'entreprise surtout sur la fidélisation de client.

Lorsque l'on possède plusieurs sites en ligne pour promouvoir une seule entreprise, il est nécessaire de faire attention à ne pas créer du contenu dupliqué. C'est un inconvénient de taille puisque cela pénalise le référencement de vos sites. Ils seront mis en compétition l'un avec l'autre. Cela réduit donc la visibilité sur Internet, et peut avantager vos concurrents.

D'après les statistiques d'une entreprise qui à deux sites internet e-commerce, nous trouvons :

- Ils constatent une baisse de fréquentation des clients dans la plateforme ;
- Seulement 20% d'entreprise arrive à résister sur leur rentabilité ;
- Une baisse de fidélisation des clients ;
- 75% des clients doutent de l'authenticité des produits

Et ces problèmes cités ci haut, est dû à une confusion que les utilisateurs rencontrent. S'ils visitent les deux sites et qu'ils y voient le même nom d'entreprise, les mêmes coordonnées et le même logo, ils se posent des questions et ils se retrouvent perdus et cela renvoie une mauvaise image de l'entreprise et n'inspire pas confiance, voilà pourquoi il y a une baisse de fidélisations de client et une baisse de fréquentation d'un site par rapport à l'autre.

1.2.1.2. Le problème de l'entreprise

Voyant la vision d'atteindre un bon nombre des personnes dans la vente de produits de l'entreprise, l'entreprise a créé deux sites de commercialisations, mais aussi des partenaires pour la promotion de produit de l'entreprise :

- Un site de type B2B, qui leur permet de vendre les produits agricoles aux entreprises d'approvisionnement, de prendre leur commande de produit dans une période donnée et qui leur permet de faire de stocker selon les demandes auprès des paysans qui leur approvisionne ;
- Un site de type B2C, qui leur permet de vendre leur produit au consommateur direct qui sont les clients détaillant et aussi faire des livraisons à domicile pour chaque commande

Ainsi, les sites mis au point les a permis d'élargir les périphériques géographiques de vente de produit, d'être en activités 24 sur 24 et 7 jours sur 7 ainsi de booste leur chiffre d'affaire dans l'entreprise.

D'après les études mènent au sein de MALEKANI sarlu, nous avons constaté les problèmes qui Suits :

- Une baisse des fidélisations des clients ;
- Pas d'échange de donnée entre les sites e-commerce de partenaire digitales qui veulent promouvoir les produits de l'entreprise ;
- Une fréquentation non équilibrer de 15% de clients sur le site de l'entreprise, suite à ça elle a une baisse au niveau financier.

1.2.1.3. La solution

A l'ère de l'open data et de la digitalisation de la société, les interfaces de programmation sont au cœur du fonctionnement du web. Car elle permet un échange de donnée entre deux sites internet et effectuer des opérations d'un autre site tout en restant dans un autre site.

Pour le cas de notre étude, nous allons développer une api qui centralise les contenues et facilite l'échange de donnée entre les sites e-commerces de partenaires digitales de l'entreprise et qui permet d'améliorer la navigation chez diffèrent les clients.

CONCLUSION

Dans ce chapitre, nous avons tout d'abord défini les différents concepts de notre thème de recherche. En ensuite nous avons présenté de manière assez large l'organisation qui fait l'objet de notre étude, c'est-à-dire MALEKANI sarlu. Nous avons soulevé les problèmes qu'ils font face et proposé une solution qui les aidera à pallier aux problèmes.

Chapitre 2. REVUE DE LITTERATURE

INTRODUCTION

A moins d'être une découverte, aucun sujet de recherche n'existe sans précédent, c'est-à-dire sans études antérieures. Ainsi, au sein de ce chapitre, nous nous donnons pour tâche de présenter les travaux déjà réalisés et qui cadrent avec notre sujet de recherche. Cela nous permettra de relever les points de différences et de signaler les innovations à apporter aux travaux antérieurs.

SECTION 1 : TRAVAUX SIMILAIRES

2.1.1. Travaux

1. TABLEAU : TRAVAUX SIMILAIRES

Thème & auteurs	Objectifs traités	Solution innovant	Résultat
1. Etude et mise en place d'un module automatisé d'échange de données entre l'université de LOME et l'université de KARA par Kolani TAMBA[1]	Partant de son constat, l'université de KARA et celle de LOME sont les deux universités publiques du TOGO, il y a une très grande mobilité des enseignants et des étudiants d'une université à l'autre. Cela implique souvent un partage de données entre les deux universités. Alors l'objectif visé ici est de connecter ces deux universités pour leur permettre l'échange de donnée.	il a développé un module permettant de mettre à disposition les ressources à partager en utilisant l'architecture SOA (architecture orientée service), Il a mis en place une API pour permettre cette échange de donnée	<ul style="list-style-type: none"> - Consulter les informations générales des données d'un étudiant provenant d'une université A à partir d'une université B ; - Consulter le parcours d'un étudiant dans son université d'origine : inscriptions, UE suivies (validées ou non), notes, diplômes ... - Consulter les offres de formations (parcours et UE) - Récupérer les données (étudiants, offres de formations, ...) pour enrichir la base donnée locales ; - Consulter la carrière universitaire d'un enseignant

			<p>(différents grades obtenus et années d'obtention) ;</p> <ul style="list-style-type: none"> - Consulter les différentes UE dispensées par un enseignant ; - Pouvoir mettre à la disposition des systèmes coopérants, les données dans un format standardisé (JSON, XML, HTML, etc).
<p>2. API-AGRO : Création d'une plateforme d'échange de données agricoles par Sine M., Haezebrouck T.P, Emonet E[55]</p>	<p>En phase avec les objectifs du Programme National de Développement Rural et Agricole (PNDAR), le projet API-AGRO a été retenu pour améliorer l'interopérabilité et les échanges de données entre les</p>	<p>Conception d'une API (application programming interface) dans la technologie open DATA et open API qui consiste à mettre à disposition de tous des données dans un format accessible, tant pour les humains que pour les machines, avec une licence permettant à tous</p>	<p>Tous ces travaux ont permis une appropriation très rapide par les partenaires de la plateforme, quelques semaines seulement après son déploiement. Le choix d'une solution « sur étagère » et non d'un développement spécifique a permis de déployer très vite une</p>

	<p>différents acteurs de la recherche et du développement agricole. Par rapport aux projets précédents qui se sont concentrés sur la standardisation des formats d'échange des données agricoles pour en assurer l'échange (GIEA, et GIEA2 notamment), ce projet a proposé une nouvelle approche en se basant sur les APIs (Application Programming Interface) qui permettra à faciliter le partage de données avec des règles de diffusion et d'usage claires entre différents acteurs de l'écosystème agricole.</p>	<p>d'accéder, utiliser, valoriser et partager</p>	<p>première version de plateforme qui a incité les partenaires du projet à diffuser des références, à connecter des API et à les valoriser pour leurs besoins internes ou externes, et ce très facilement grâce aux interfaces de visualisation (graphiques et cartographies). Des fonctionnalités souhaitées et non présentes dans la première version de la solution ont été ajoutées tout au long du projet par OpenDataSoft.</p>
--	---	---	--

<p>3. Conception et mise en oeuvre d'une plate-forme pour la sûreté de fonctionnement des services Web par Nicolas Salatge[2]</p>	<p>Dans ce travail, ils se sont essentiellement concentrés sur des fautes provenant des prestataires afin de pouvoir vérifier l'efficacité de leurs mécanismes de recouvrement.</p>	<p>Ils ont conçu un service web qui permet la communication efficace entre le client et le prestataire tout en assurant l'échanger de donnée</p>	<p>La plate-forme qu'ils présentent dans ce mémoire, nommée IWSD (Infrastructure for Web Service Dependability), est une solution non-intrusive qui s'appuie totalement sur les concepts et les protocoles de base qui forment la pierre angulaire des Services Web. C'est une infrastructure générique pouvant s'exécuter sur un nœud distant différent et indépendamment du dispositif mis en place par le client ou le prestataire. Sur cette plate-forme, un client ou un prestataire peut déployer un connecteur spécifique conçu pour un contexte d'utilisation particulier.</p>
--	---	--	--

2.1.2 Articles

2. TABLEAU : ARTICLES SIMILAIRES

Thème & auteurs	Objectifs traités	Solution innovant	Résultat
1. L'interface de programmation d'application (API) des matériaux : une API simple, flexible et efficace pour les données des matériaux basée sur les principes REST (REpresentational State Transfer).[56]	<p>Dans cet article, ils décrivent l'interface de programmation d'application (API) des matériaux, une interface simple, flexible et efficace permettant d'interroger et d'interagir par programmation avec la base de données Materials Project basée sur le modèle REST (Representational State Transfer) pour le Web.</p>	<p>Ils ont utilisés le Rest Api pour interagir et interroger par programme la base de donnée du projet matériaux pour leur plateforme web</p>	<p>Ils ont mis en évidence certaines de ces analyses rendues possibles par l'API Matériaux, en particulier celles nécessitant la consolidation de données sur un grand nombre de matériaux, telles que l'exploration de données sur les tendances structurelles et immobilières et la génération de diagrammes de phase.</p> <p>Ils ont terminé par une discussion sur le rôle de l'API dans la création d'une communauté qui développe de nouvelles applications et analyses basées sur les données du projet Matériaux.</p>

<p>2. Documentation de l'interface de programmation d'applications : Que veulent les développeurs de logiciels ?[57]</p>	<p>Cet article présente les résultats d'une série d'entretiens semi-structurés et d'un questionnaire de suivi mené pour explorer les objectifs d'apprentissage et les stratégies d'apprentissage des développeurs de logiciels, les ressources d'information vers lesquelles ils se tournent et les critères de qualité qu'ils appliquent à la documentation des API.</p>	<p>Création d'un document qui permet au développeur d'apprendre les objectif d'une API</p>	<p>Leurs résultats montrent que les développeurs essaient d'abord de former une compréhension globale concernant l'objectif global et les principales fonctionnalités d'une API, mais adoptent ensuite une stratégie d'apprentissage orientée concepts ou orientée code que la documentation de l'API doit toutes deux aborder.</p> <p>Leurs résultats montrent également que les critères de qualité généraux tels que l'exhaustivité et la clarté sont également pertinents pour la documentation des API. Le développement et la maintenance de la documentation API doivent donc impliquer l'expertise des</p>
---	---	--	--

			professionnels de la communication.
3. Réinventer l'échange de données de santé : une feuille de route compatible avec l'interface de programmation d'applications pour l'Inde[58]	<p>Ce programme national de protection de la santé devrait donner cet élan à une adoption généralisée. Cependant, les données de santé en Inde ne sont généralement pas numérisées. Dans les rares cas où elles le sont, les données ne sont pas normalisées, ne sont pas interopérables et ne sont pas facilement accessibles aux cliniciens, aux chercheurs ou aux décideurs. Bien que de tels obstacles à l'échange facile d'informations sur la santé ne soient pas uniques à l'Inde, la nature vierge de l'infrastructure de santé numérique de l'Inde offre</p>	<p>Ils proposent ici un écosystème d'informations sur la santé fédéré, centré sur le patient et doté d'une interface de programmation d'applications (API)</p>	<p>Il s'appuie sur les meilleures pratiques mondiales et favorise l'adoption de principes de conception centrés sur l'humain, la minimisation des données et les API standard ouvertes. Les recommandations sont le résultat de 18 mois de délibérations avec de multiples parties prenantes en Inde et aux États-Unis, y compris du milieu universitaire, de l'industrie et du gouvernement</p>

	une excellente occasion d'éviter les pièges des systèmes de santé numériques complexes et restrictifs qui ont évolué ailleurs.		
--	--	--	--

SECTION 2 : METHODE DES RESOLUTION

Nous avons consacré cette partie à la présentation de la méthode et à l'élaboration du diagramme de résolution pour pouvoir arborer la structure d'enchaînement et la description des étapes suivies dans la réalisation de notre système.

D'une façon brève, une méthode est une démarche reproductible que l'on suit pour atteindre des résultats considérés comme fiable.[59] Ces différentes méthodes et techniques utilisées doivent mener à atteindre les différents objectifs que nous nous sommes fixés pour ce travail.

2.2.1 Cycle de développement Agile

Les méthodes agiles s'appuient sur l'utilisation d'un cadre méthodologique léger mais suffisant, centré sur l'humain et la communication. Elles préconisent une planification adaptative, un développement évolutif, une livraison précoce et une amélioration continue. Elles encouragent donc des réponses flexibles au changement.[60]

Notons que le cycle de vie du développement d'un logiciel dit SDLC, aide au développement d'un produit d'une manière systématique, bien définie, et rentable. Les diverses raisons pour lesquelles nous nous devons d'utiliser un modèle de cycle de développement sont les suivantes :

- Pouvoir comprendre l'ensemble du processus aboutissant à la réalisation ;
- Se garantir d'une approche structurée pour le développement de la plateforme ;
- Faire la planification des ressources à l'avance.

Le SDLC peut être divisé en 5 voire 9 phases ; cela veut dire qu'un minimum de cinq phases ou un maximum de neuf phases est requis lors du développement. De notre côté nous avons opté pour un SDLC en 5 phases, que nous détaillons ci-dessous :

1. **L'étude préliminaire** : c'est la première étape de tout projet. Elle consiste à faire la reconnaissance du besoin et permet de définir le problème. A ce stade nous devons identifier la nécessité de changement dans le système existant et détecter ses lacunes, consulter le cahier des charges soumis, consulter les documents en rapport avec la demande, se ressourcer sur des sujets similaires.
2. **La conception du système** : est la phase la plus créative et stimulante du SDLC. Elle décrit le système final et le processus par lequel il est développé. Le processus de

conception se traduit par des exigences en une représentation du logiciel qui peut être évaluée pour la qualité avant que le codage ne commence.

3. **Le développement** : il consiste à traduire la conception du système dans le code d'un langage de programmation donnée. Dans cette phase, l'objectif est de mettre en œuvre la conception de la meilleure manière possible. Pour le développement de notre API, nous avons opté pour le langage PHP qui permettra l'échange entre les deux sites. Nous présenterons dans les annexes un extrait du code source de l'API.
4. **Les tests** : constituent la principale mesure de contrôle de qualité utilisée au cours du développement de logiciels. Sa fonction essentielle est de détecter les erreurs dans le logiciel. Le but du test est de découvrir les exigences, la conception et les erreurs de codage dans le programme. Cette phase est une activité extrêmement critique. Elle exige une bonne planification de l'ensemble du processus de test. Lors de l'essai de l'unité, les cas de test spécifiés sont exécutés et les résultats réels sont comparés avec la sortie attendue.
5. **Le déploiement et/ou phase de mise en œuvre** : une fois que le système a été conçu, il est prêt à être exécuté. La mise en œuvre concerne les tâches menant immédiatement à un système pleinement opérationnel. Pour une interface de programmation d'application telle que la nôtre, cela consiste à la faire connecter pour la rendre accessible sur les deux sites web.

2.2.2 Diagramme de résolution

Suivant les méthodes agiles, nous sommes arrivés à un diagramme tel que se dérouleront les différentes étapes de la mise en place de notre système.

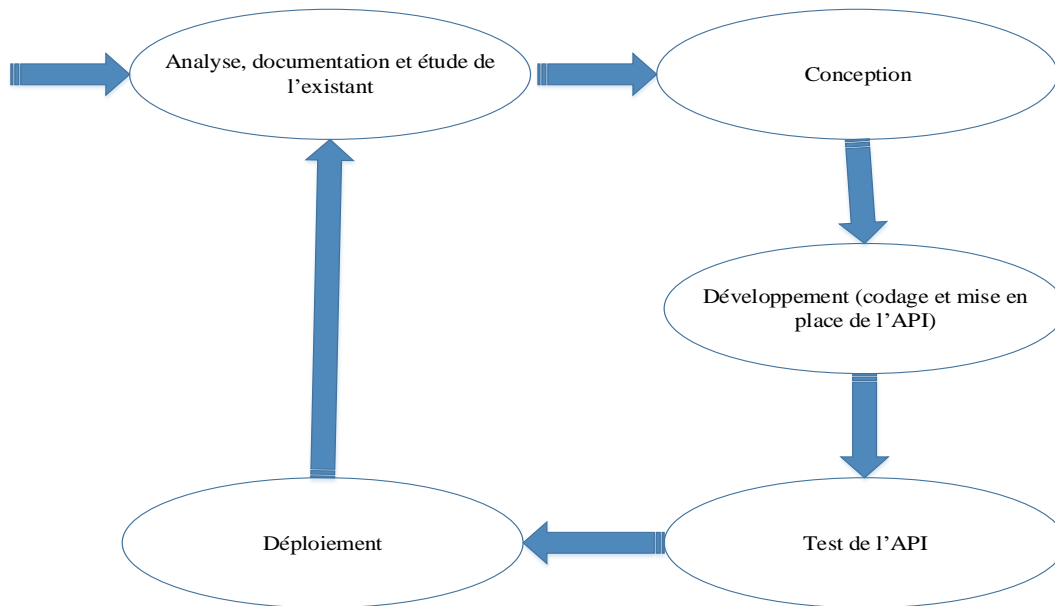


Figure 1 : Architecture

Comme dit précédemment, tous les processus se retrouvent dans une sorte de boucle infinie, car la version finale de la conception n'existe presque pas dans les méthodes agiles ; les mises à jour sont continues et indéterminées, à moins que l'entreprise n'arrive à un moment où elle doit altérer complètement le fond du système, ou que le système ne soit devenu obsolète.

CONCLUSION

Dans ce chapitre nous avons présenté les différents travaux en rapport avec notre sujet d'étude, en relevant les objectifs poursuivis par l'auteur, les points faibles et les points forts de son système pour pouvoir ressortir ce que notre système apporte par rapport aux systèmes précédents. Cette partie a été traitée dans la section première de notre chapitre. La seconde section a concerné la formulation d'une approche méthodologique pouvant nous permettre d'adopter une démarche à suivre dans la résolution de notre problème.

Chapitre 3 : METHOLOGIE ET RESULTATS

INTRODUCTION

Nous entrons dans le chapitre qui concerne la mise en application des différentes théories évoquées dans les parties précédentes du travail. Ici nous expliquons le déroulement de la mise en application de la conception en usant des différentes méthodes adoptées et déjà décrites précédemment. Ainsi donc, partant de problème de MALEKANI sarlu quant à la communication et l'échanges de donnée entre les deux sites internet de l'entreprise.

Nous allons nous focaliser sur le développement de l'API qui envoie et reçoit les informations de différent site de l'entreprise en estimant les coûts du projet et nous présenterons le résultat issu de ce travail c'est-à-dire le système complet et opérationnel répondant au problème soumis.

Section 1. METHODOLOGIE

3.1 Cahier des charges

Les fonctionnements à implémenter dans le site web restent globalement les mêmes que ce dont nous avons trouvé dans l'entreprise. Mais les seules améliorations qui devaient être apportées à ce deux systèmes sont les suivantes :

- L'échange de données entre les deux sites ;
- Visualisations des éléments provenant de l'autre site vers l'autre ;
- Effectuation des opérations entre les deux sites web.

Ces sont ces points qui devaient faire l'objet de notre travail en implémentant une solution.

3.2. Planning

3.2.1. Identification et déroulement des tâches

3. TABLEAU : IDENTIFICATION ET DÉROULEMENT DES TACHES

Taches	Désignations	Jours	Contrainte
1	Prise de contact	2 jours	-
2	Etude du système existant	3 jours	1
3	Récolte de donnée nécessaire	4 jours	2

4	Etudes de technologie	20 jours	3
5	Conception de la maquette	4 jours	4
6	Conception de la structure du backend	6 jours	5
7	Affichage de donnée dans le deux sites	2 jours	6
8	Test de fonctionnalité	6 jours	6 ; 7
TOTAL		47 Jours	

3.2.2. Estimation des coûts de la conception

4. TABLEAU : ESTIMATION DE COUTS DE LA CONCEPTION

Taches	Désignations	Jours	Contrainte	Personne	Coût unitaire	Total(\$)
1	Prise de contact	2 jours	-	1	-	-
2	Etude du système existant	3 jours	1	1	-	-
3	Récolte de donnée nécessaire	4 jours	2	1	24	96
4	Etudes de technologie	20 jours	3	1	24	480
5	Conception de la maquette	4 jours	4	1	24	96
6	Conception de la structure du backend	6 jours	5	1	120	720
7	Affichage de donnée dans le deux sites	2 jours	6	1	24	48
8	Test de fonctionnalité	6 jours	6 ; 7	1	24	144
TOTAL		47 jours				1584\$

4.2.1. Diagramme de Gantt

Ce diagramme permet de faire figurer l'enchaînement du projet dans un calendrier afin de montrer la durée d'exécution de chaque tâche. L'outil en soi est un tableau. Une colonne correspond à une unité de temps (exprimée en années, semaines, jours ou heures) et chaque

ligne correspond à une tâche particulière. Dans celui-ci, la tâche est représentée par une barre horizontale dont la longueur correspond à la durée d'exécution de la tâche.

4.2.1.1. Liste de tâches avec contraintes du projet

[illegible]

Figure 2 : Liste des tâches

4.2.1.2. Réseau des tâches

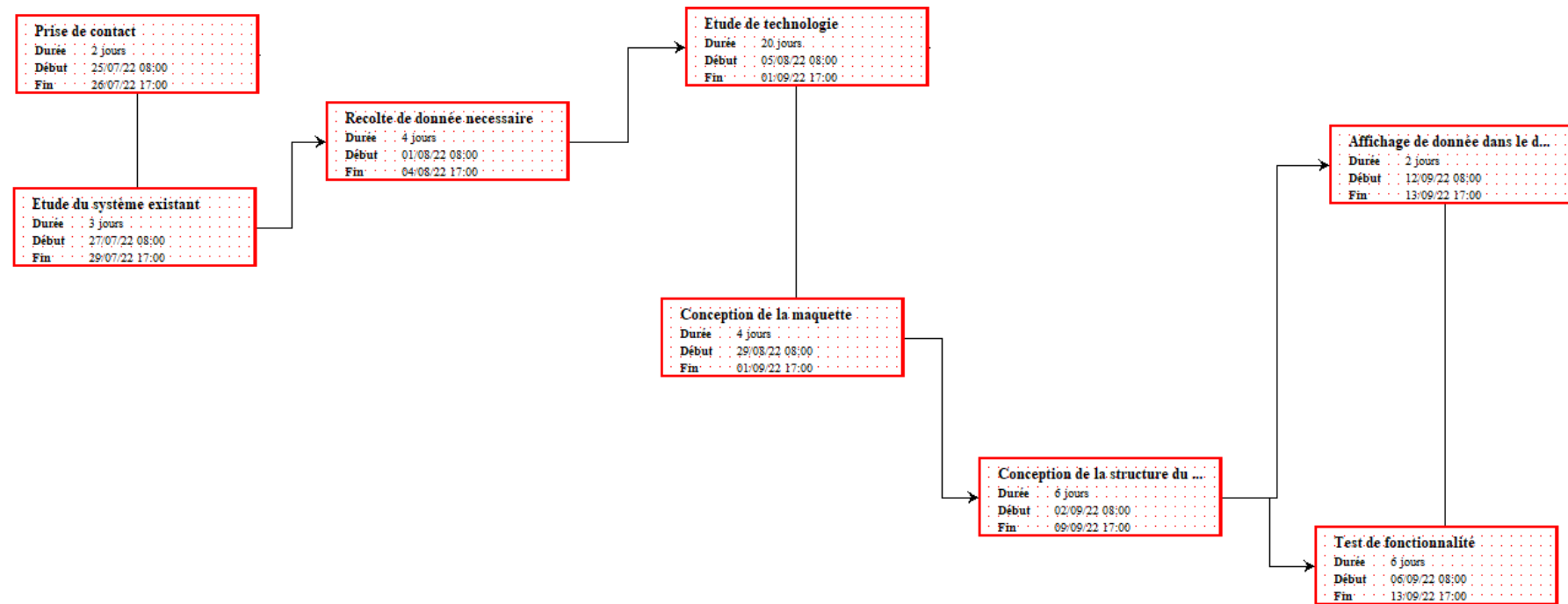


Figure 3 : Réseau de tâche

4.2.1.3. Diagramme de Gantt au plus tôt

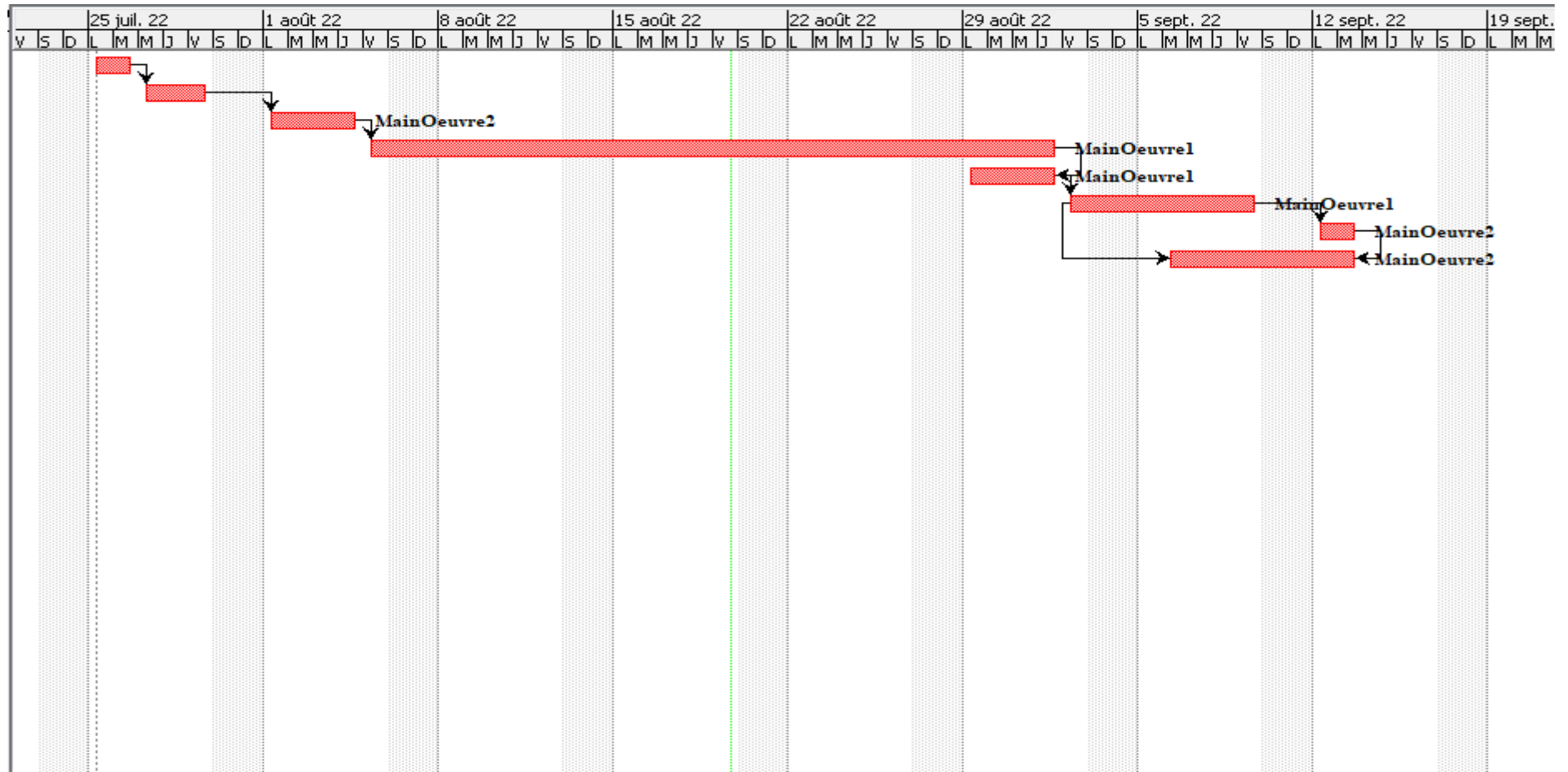


Figure 4 : Diagramme de GANTT

SECTION 2 : RESULTATS ET ANALYSE

3.2.1 Modélisation

a. Diagramme de cas d'utilisation

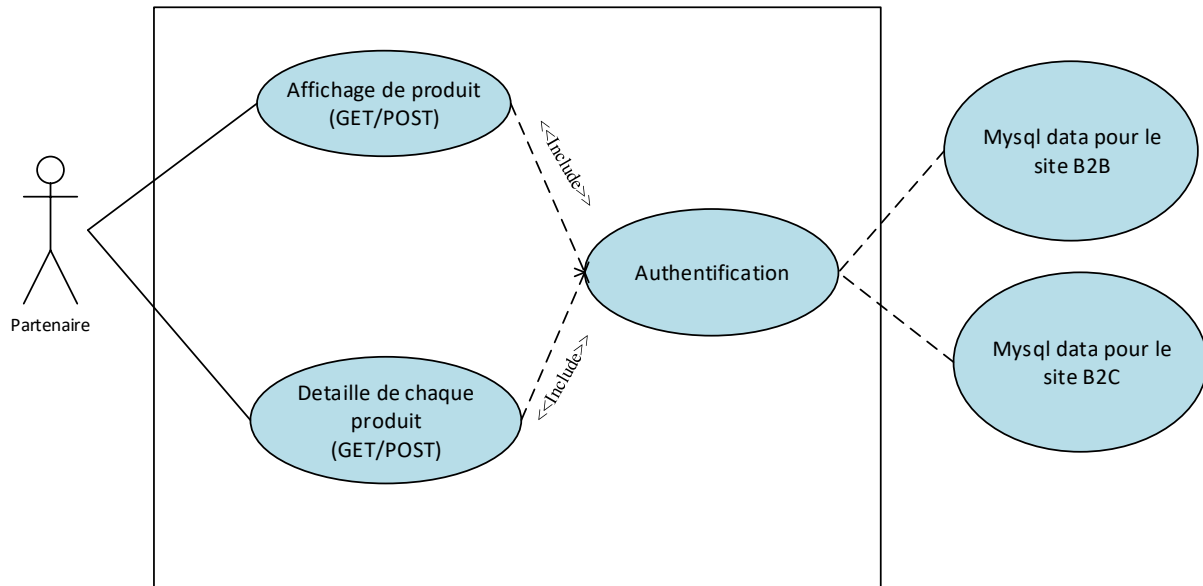


Figure 5 : Diagramme de cas d'utilisation

Description détaillée de diagramme de cas d'utilisation

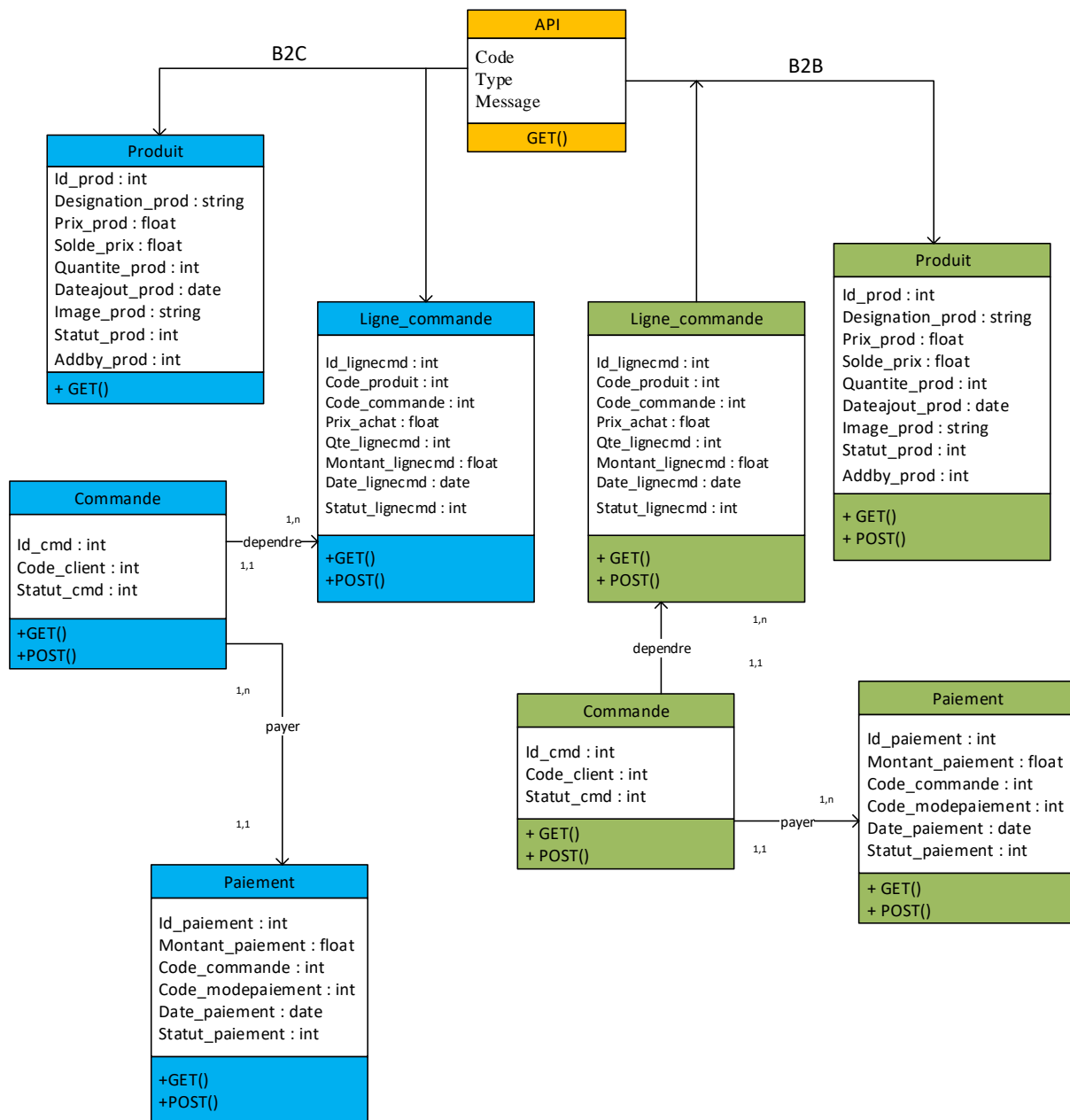
Les présents cas d'utilisation constituent un moyen plus simple et clair de recueillir et de décrire tous les services que le système rend à ses utilisateurs. Il pourra aussi être utilisé par la suite comme moyen d'organisation du développement de l'API, notamment pour la structuration et le déroulement des tests(API).

Quelques manipulations clés :

- **L'authentification** : Avant toute manipulation dans le système, l'utilisateur doit s'authentifier, c'est-à-dire entrer ses informations de login (Qui doivent être valides).
- **L'utilisateur** : en tant que simple utilisateur, il peut visualiser les informations affichées sur la plateforme web qui transitent par une api en provenance des deux sites déjà mis en place mais aussi il peut **les différents produits** de différent site en étant dans l'autre site.

En outre, une persistance de ces informations est rendue possible par l'entremise de deux bases de données.

b. Diagramme de classe



Ce diagramme nous permet de donner la représentation statique du système à développer. Cette représentation est centrée sur les concepts de classe et d'association. Chaque classe se décrit par les données et le traitement dont elle est responsable pour elle-même et vis-à-vis des autres classes.

Description des classes

- **API** : Contient toutes les informations que nous recevons provenant des deux sites e-commerce existant déjà. Le *code* désigne, le *type* montre de quel format les données sont envoyés et le *message* contient les données envoyées par les deux sites.

- **Produit** : contient les produits que l'entreprise met en vente. Cette table contient les détails comme : une *désignation*, un *prix de vente*, un *solde* si le produit est soldé, une *quantité*, une *image* et un *statut* pour voir si le produit existe toujours en stock ;
- **Ligne_commande** : cette table contient les informations de différents produit réservés par le client au moment de la commande, elle contient notamment le *code_produit*, le *code_commande*, le *prix_achat*, la *qte_ligne*
- **Commande** : cette table contient les informations sur le client qui passer de commande
- **Paiement** : cette table contient les informations effectuer pour le paiement de la commande, nous pouvons citer : le *code_commande*, *montant_paye*, *date_paiement* et le *mode de paiement* qui est soit mobil moyen ou via un compte bancaire

c. Diagramme de déploiement

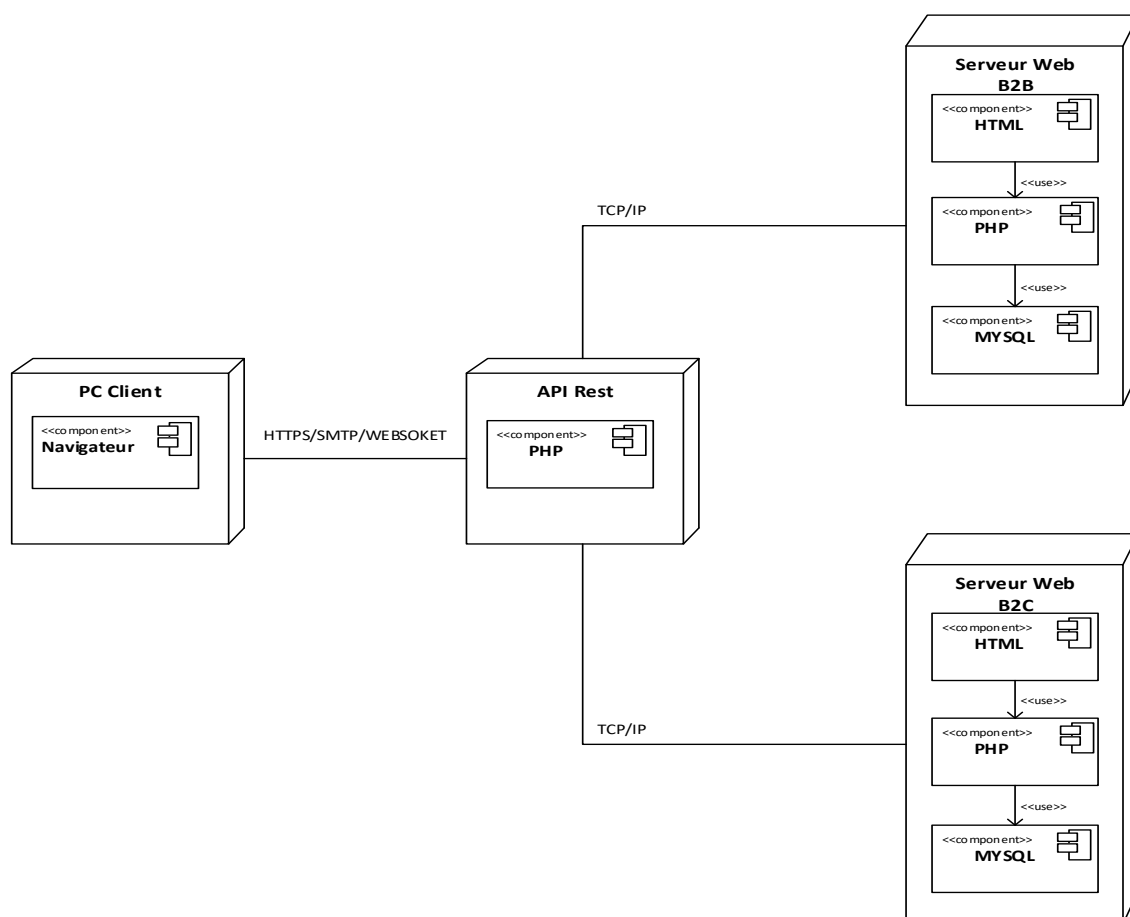


Figure 6 : Diagramme de déploiement

Ce diagramme nous permet de représenter :

- La disposition physique des ressources matériels du système et montre la répartition des différents composants (Élément logiciel) sur différents matériels.
- Ainsi que la nature des connexions de communication entre les différentes ressources matérielles

1. ARCHITECTURE

Une architecture est un modèle conceptuel d'un système qui décrit de manière symbolique et schématique les différents éléments d'un ou plusieurs systèmes informatiques, leurs interrelations et leurs interactions. Elle décrit comment un système doit être conçu de manière à répondre aux spécifications ou exigences. [23]

Nous avons pensé une architecture qui décrit comment l'API interagissent entre les deux sites e-commerce dans notre système.

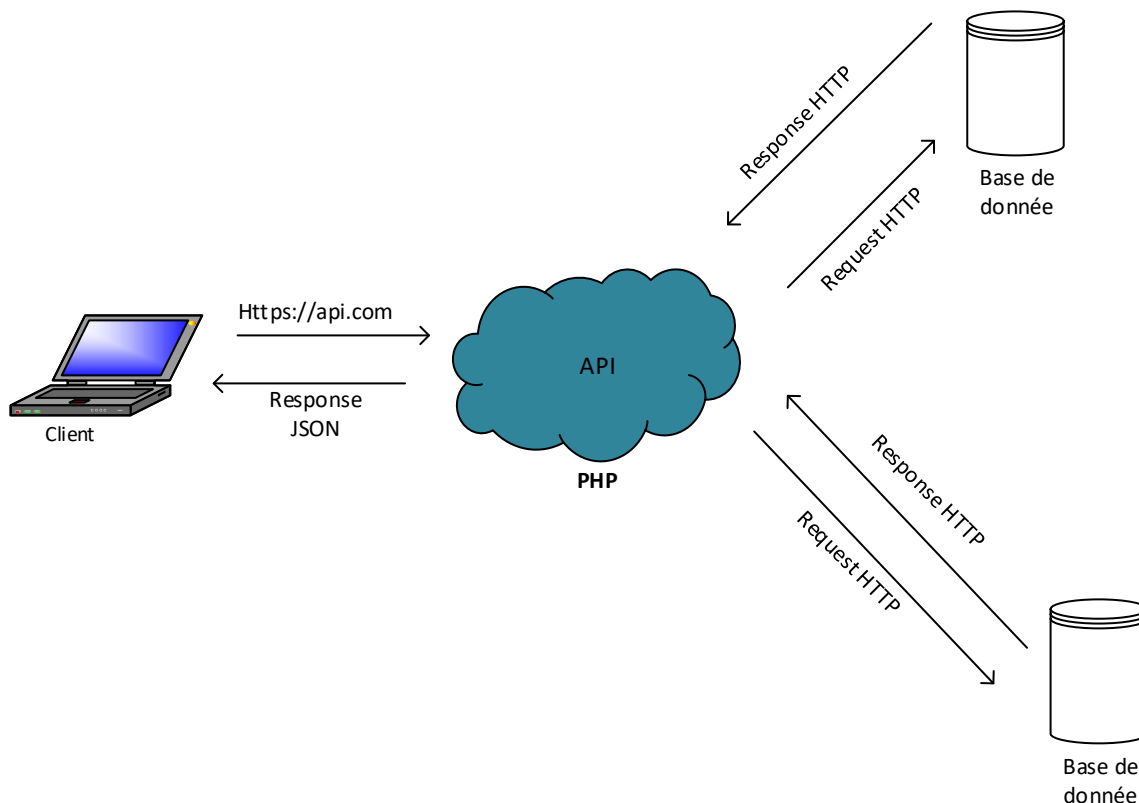


Figure 7 : Architecture

1.1 Description des interactions entre les entités

- Client

A partir d'un site de l'entreprise mis en ligne, B2B ou B2C, le client peut envoyer une requête via notre api qui lui permettra de voir les informations sortant d'un autre site pour pouvoir effectuer des opérations et aussi faire ses paiements.

Les clients peuvent :

- Affichage des produits ainsi que leurs détails ;
- Passer de commande ;
- Paiement de commande.

- API

L'interface de programme d'application ici présent, permet au client de passer une requête https à notre api et ce l'api par rapport au requête reçu, chercher vers quelle base de donnée prendre l'information précise demandé par client et remmener une information sous format JSON ou XML pour permettre au client de l'utilise

- Base de donnée

La base de donnée qui est là, sont les bases de données qui existe déjà et cela nous permettra de passer des requêtes venant de l'api pour pourvoir répondre au besoin de du client qui aimera utiliser les informations venant de l'autre site internet.

Scénario :

L'utilisateur ou client via le navigateur web accède à l'un de site de l'entreprise via l'URL du site, qui est censé l'ouvrir une page qui affiche les produits qui sont mis en vente, cette page qui est le front end (HTML).

Chaque site est géré par un serveur qui traite les informations demandées par un client mais aussi, une page qui permet de voir les produits venant du deuxième site et qui l'api permet de recevoir les requêtes du client et passer une demande à l'autre base de donnée pour pouvoir permettre le client de passer des opérations en restant sur le même site.

2. DEVELOPPEMENT DE L'APPLICATION

2.1 Choix des technologies

➤ MYSQL



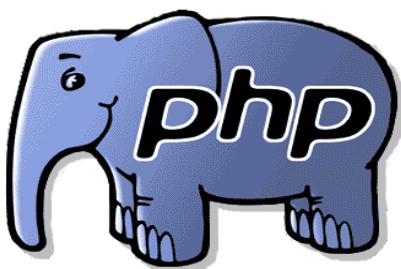
MySQL est un système de gestion de bases de données relationnelles SQL open source développé et supporté par Oracle[53].

MySQL permet le stockage de données dans des tables séparées. Ces tables permettent de classer les données selon leurs caractéristiques. Elles sont organisées grâce à des lignes et des colonnes, il a été lancé à l'origine en 1995. Depuis, il a connu quelques changements de propriétaire et de gestion, avant de se retrouver chez Oracle Corporation en 2010. Alors qu'Oracle est en charge maintenant, MySQL est toujours un logiciel open source, ce qui signifie que vous pouvez l'utiliser et le modifier librement.

Pour comprendre le fonctionnement de MySQL, il est important de connaître deux concepts liés :

- Base de données relationnelle
- Modèle client-serveur

➤ PHP



- Le PHP (Hypertext Preprocessor) désigne un langage informatique, ou un langage de script, utilisé principalement pour la conception de sites web dynamiques[47]. Il s'agit

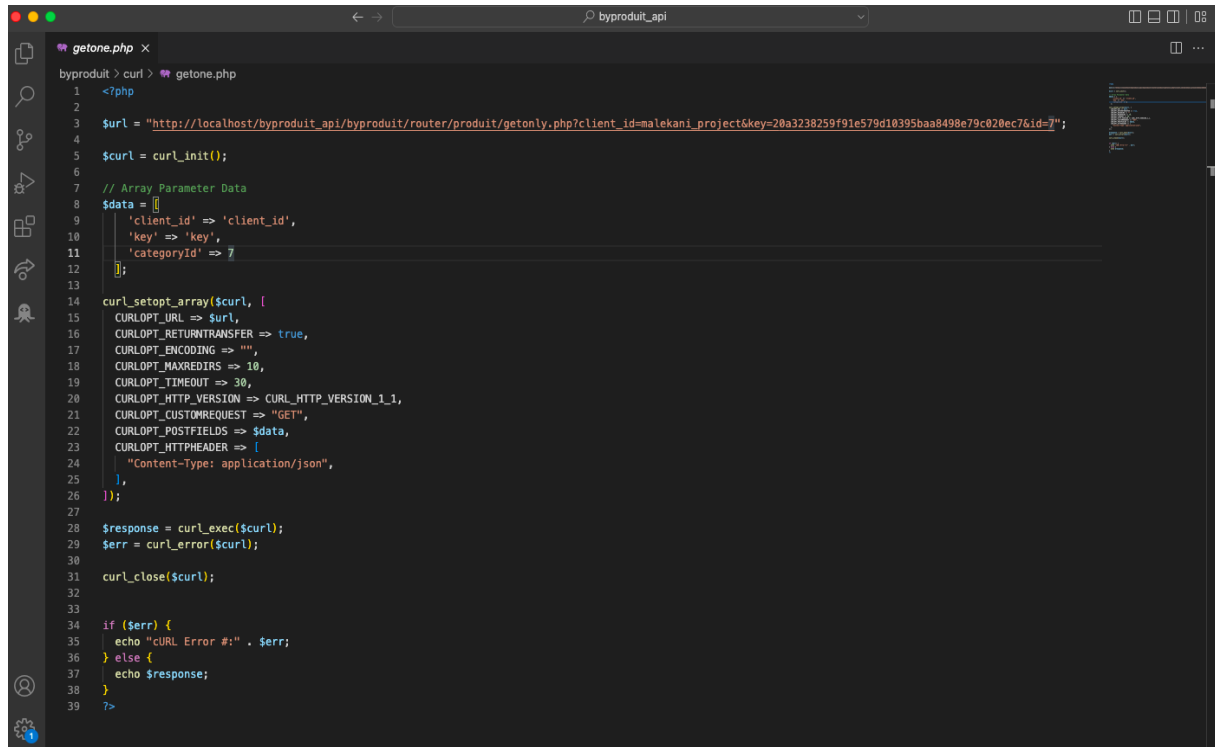
d'un langage de programmation sous licence libre qui peut donc être utilisé par n'importe qui de façon totalement gratuite.

- Un langage de script est un langage de programmation interprété qui permet de manipuler les fonctionnalités d'un système informatique configuré pour fournir à l'interpréteur de ce langage un environnement et une interface qui déterminent les possibilités de celui-ci. Le langage de script peut alors s'affranchir des contraintes des commandes de bas niveau — prises en charge par l'intermédiaire de l'interface — et bénéficier d'une syntaxe de haut niveau.
- Le langage PHP fut créé en 1994 par Rasmus Lerdorf pour son site web. C'était à l'origine une bibliothèque logicielle écrite d'abord en langage Perl, puis en langage C. Il s'en servait pour conserver une trace des visiteurs qui venaient consulter son CV. PHP signifiait alors Personal Home Page tools. Le code source de PHP fut publié par son auteur en 1996. En 1997, deux étudiants, Andi Gutmans et Zeev Suraski, redéveloppèrent le cœur de PHP puis son moteur interne, aboutissant à ce qu'on appelle le Zend Engine.
- Depuis les versions du langage se sont succédées, apportant leur lot d'améliorations et corrigeant les (nombreuses) failles de sécurité apparues en chemin. La version 5 de PHP, sortie en 2004, utilise Zend Engine2 et introduit un véritable modèle objet, une gestion des erreurs fondée sur le modèle des exceptions, ainsi que des fonctionnalités de gestion pour les entreprises.[48]
- Le langage PHP est souvent associé au serveur de base de données MySQL et au serveur Apache. Avec le système d'exploitation Linux, il fait partie intégrante de la suite de logiciels libres LAMP.
- Sur un plan technique, le PHP s'utilise la plupart du temps côté serveur. Il génère du code HTML, CSS ou encore XHTML, des données (en PNG, JPG, etc.) ou encore des fichiers PDF. Il fait, depuis de nombreuses années, l'objet d'un développement spécifique et jouit aujourd'hui une bonne réputation en matière de fiabilité et de performances.
- PHP est un langage impératif orienté objet. Il s'agit d'un langage de script interprété côté serveur. Il est considéré comme une des bases de la création de sites web dits dynamiques mais également des applications web. PHP a permis de créer un grand nombre de sites web célèbres, comme Facebook, Wikipédia, etc.
- **Fonctionnement du PHP**
- Même s'il peut être utilisé en ligne de commande, PHP est principalement associé à un serveur Web utilisant le protocole HTTP dans le cadre d'une architecture client/serveur.

- Un serveur Web en architecture trois tiers est composé d'un système d'exploitation, un serveur HTTP, un langage serveur et enfin un système de gestion de base de données (SGBD), cela constituant une plate-forme.
- Dans le cas de PHP comme langage serveur, les combinaisons les plus courantes sont celles d'une plateforme LAMP (pour Linux Apache MySQL PHP) et WAMP (Windows Apache MySQL PHP). Une plate-forme WAMP s'installe généralement par le biais d'un seul logiciel qui intègre Apache, MySQL et PHP, par exemple EasyPHP et WampServer. Il existe le même type de logiciels pour les plates-formes MAMP (Mac OS Apache MySQL PHP), à l'exemple du logiciel MAMP et XAMPP.
- En conclusion le PHP est un langage de scripts généraliste et Open Source, spécialement conçu pour le développement d'applications web et pour servir de langage de script coté serveur, ce qui fait qu'il est capable de réaliser tout ce qu'un script quelconque peut faire, comme collecter des données de formulaire, générer du contenu dynamique etc...

4. RESULTAT

4.1 PRESENTATION DE L'API



```

1 <?php
2
3 $url = "http://localhost/byproduit_api/byproduit/router/produit/getonly.php?client_id=malekani_project6key=20a3238259f91e579d10395baa8498e79c020ec76id=7";
4
5 $curl = curl_init();
6
7 // Array Parameter Data
8 $data = [
9     'client_id' => 'client_id',
10    'key' => 'key',
11    'categoryId' => 7
12 ];
13
14 curl_setopt_array($curl, [
15     CURLOPT_URL => $url,
16     CURLOPT_RETURNTRANSFER => true,
17     CURLOPT_ENCODING => "",
18     CURLOPT_MAXREDIRS => 10,
19     CURLOPT_TIMEOUT => 30,
20     CURLOPT_HTTP_VERSION => CURL_HTTP_VERSION_1_1,
21     CURLOPT_CUSTOMREQUEST => "GET",
22     CURLOPT_POSTFIELDS => $data,
23     CURLOPT_HTTPHEADER => [
24         "Content-Type: application/json",
25     ],
26 ]);
27
28 $response = curl_exec($curl);
29 $err = curl_error($curl);
30
31 curl_close($curl);
32
33
34 if ($err) {
35     echo "CURL Error #:" . $err;
36 } else {
37     echo $response;
38 }
39 ?>

```

Figure 8 : affichage de l'api par le client (get_all)

Avant d'avoir l'accès à l'api, il faut créer un compte ou connectez-vous en utilisant vos informations d'identification existantes. Si vous n'avez pas de compte, le partenaire doit s'inscrire. Il va copier et coller les codes ci-après pour lister les produits dans son site internet.

```

1 <?php
2
3 $curl = "http://localhost/byproduit_api/byproduit/router/produit/read.php?client_id=malekani_project&key=20a3238259f91e579d10395baa8";
4
5 $data_array = array(
6     'client_id' => 'client_id',
7     'key' => 'key'
8 );
9
10 $data = http_build_query($data_array);
11 $ch = curl_init();
12
13 curl_setopt($ch, CURLOPT_URL, $curl);
14 curl_setopt($ch, CURLOPT_POST, true);
15 curl_setopt($ch, CURLOPT_POSTFIELDS, $data);
16 curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
17
18 $resp = curl_exec($ch);
19
20 if($e = curl_error($ch)){
21     echo $e;
22 }
23 else{
24     $decoded = json_decode($resp, true);
25
26     print_r($decoded);
27 }
28 curl_close($ch);
29

```

Figure 9 : affichage de l'api par le client (get_only)

```

1 const express = require('express')
2 const app = express()
3 const bodyParser = require('body-parser')
4 // const axios = require('axios').create({baseUrl:'http://127.0.0.1:3311/'})
5 const axios = require('axios')
6 const PORT = 3303
7
8 // configuration
9 app.use(bodyParser.json())
10 app.use(bodyParser.urlencoded({ extended: true }))
11
12 // ping test api
13 app.get('/ping', (req, res) => {
14     res.send({
15         status: true,
16         message: "welcome to malekani API"
17     })
18 })
19
20 app.get('/clients', async (req, res) => {
21
22     const data = await axios.get("http://localhost:8080/byproduit_api/byproduit/router/produit/read.php?client_id=malekani_project&key=20a3238259f91e579d10395baa8")
23     .then(response => {
24         res.status(200).json({
25             status: true,
26             total: response.data.data.length,
27             data: response.data.data,
28         })
29     })
30     .catch((err) => {
31         res.status(500).json({ message: err.message })
32     })
33 })
34
35 app.listen(PORT, () => {
36     console.log(`Server start at http://127.0.0.1:${PORT}`)
37 })

```

Figure 10: affichage de l'api par le client qui utilise node js

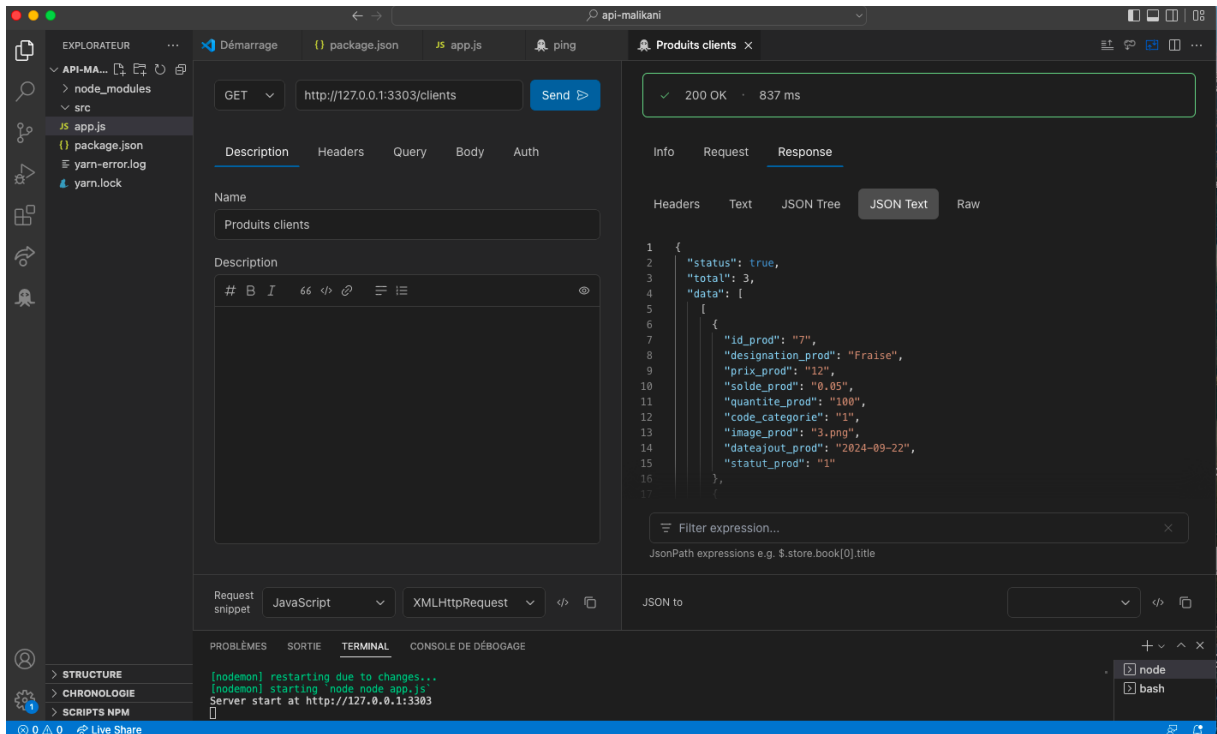


Figure 10: affichage de l'api par le client qui utilise node js

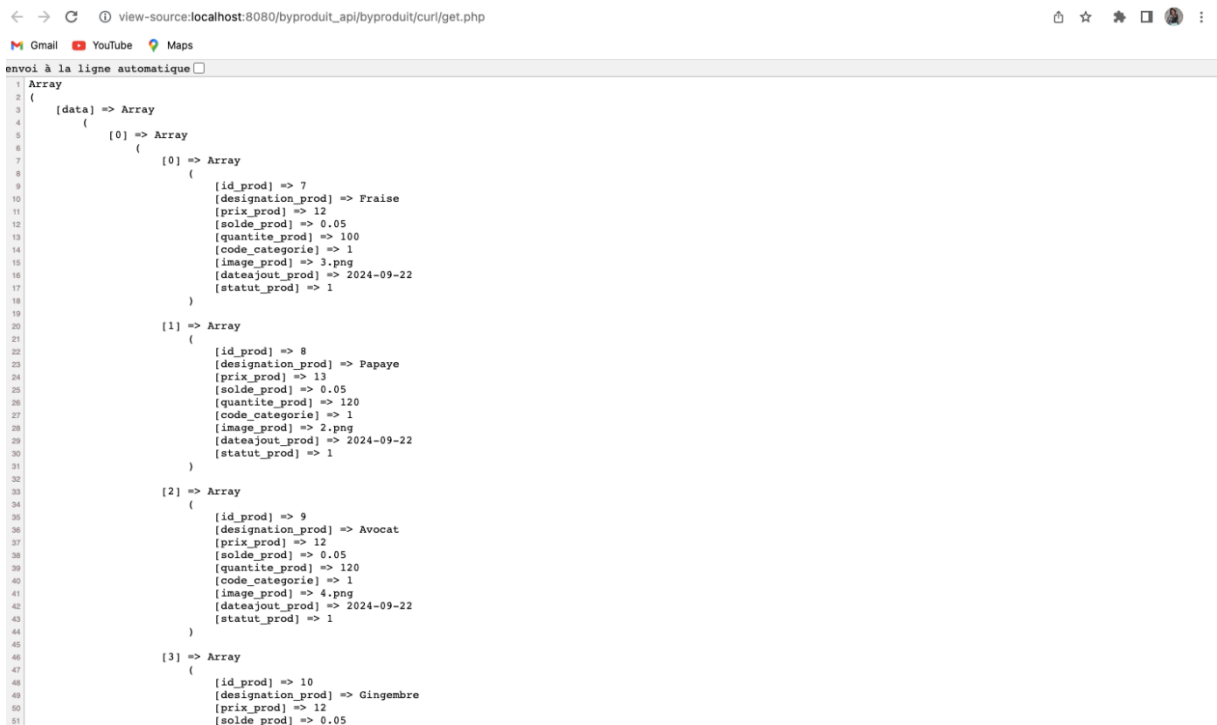


Figure 10 : affichage de dans le navigateur

```

view-source:localhost:8080/byproduit_api/byproduit/curl/get.php
Gmail YouTube Maps
14      [code_categorie] => 1
15      [image_prod] => 3.png
16      [dateajout_prod] => 2024-09-22
17      [statut_prod] => 1
18      )
19
20      [1] => Array
21      (
22          [id_prod] => 8
23          [designation_prod] => Papaye
24          [prix_prod] => 13
25          [solde_prod] => 0.05
26          [quantite_prod] => 120
27          [code_categorie] => 1
28          [image_prod] => 2.png
29          [dateajout_prod] => 2024-09-22
30          [statut_prod] => 1
31      )
32
33      [2] => Array
34      (
35          [id_prod] => 9
36          [designation_prod] => Avocat
37          [prix_prod] => 12
38          [solde_prod] => 0.05
39          [quantite_prod] => 120
40          [code_categorie] => 1
41          [image_prod] => 4.png
42          [dateajout_prod] => 2024-09-22
43          [statut_prod] => 1
44      )
45
46      [3] => Array
47      (
48          [id_prod] => 10
49          [designation_prod] => Gingembre
50          [prix_prod] => 12
51          [solde_prod] => 0.05
52          [quantite_prod] => 0
53          [code_categorie] => 1
54          [image_prod] => 8.png
55          [dateajout_prod] => 2024-09-22
56          [statut_prod] => 1
57      )
58
59      )
60
61      [1] => pour afficher une image utiliser 'http://localhost/byproduit/admin/fichier/'
62      [2] => 1
63      )
64      )
65      )
66

```

Figure 11 : affichage de dans le navigateur

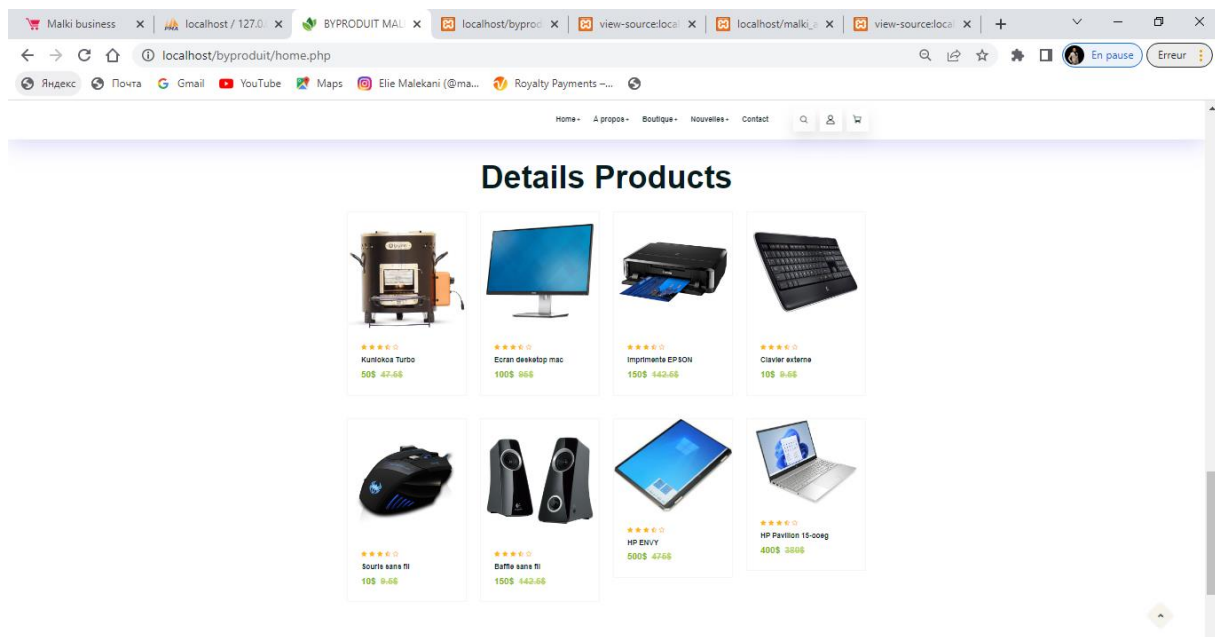


Figure 12 : affichage de produit B2C dans un site partenaire B2B

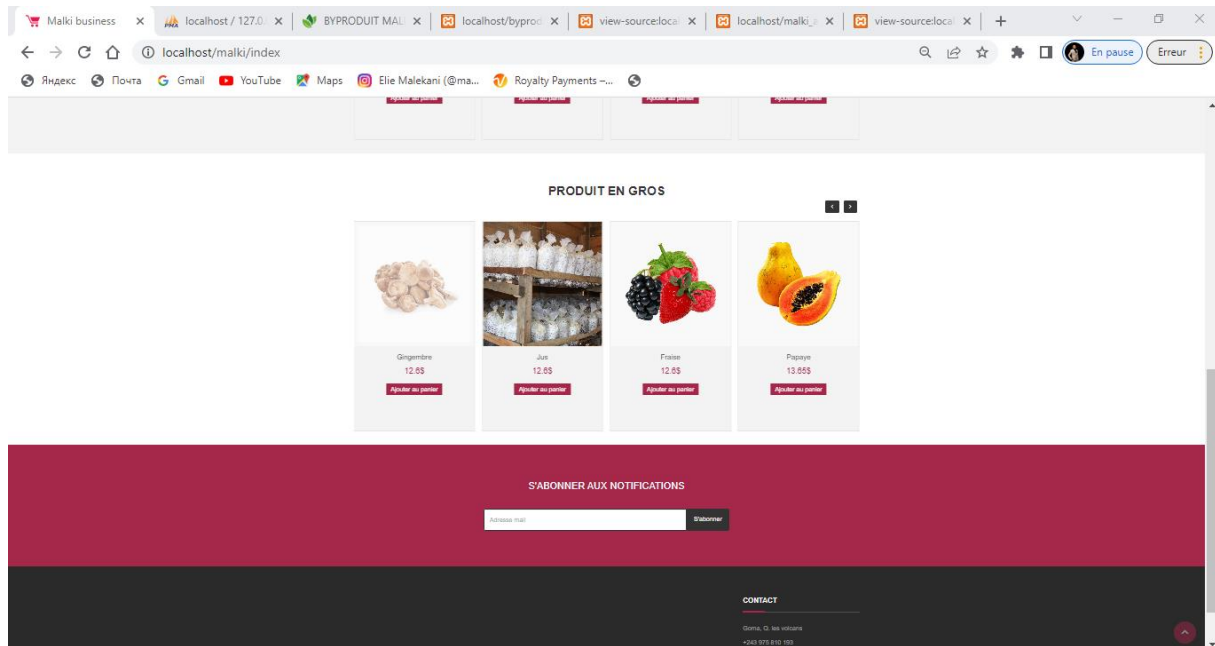


Figure 13 : affichage de produit B2B dans un site partenaire B2C

CONCLUSION

Grâce au cahier de charge du présent projet, nous avons pu faire une subdivision des tâches puis un diagramme de Gantt au plus tôt pour le bon suivi des activités sous forme d'un calendrier. Nous avons développé une interface API intermédiaire qui facilitera la centralisation et l'échange de données des sites web marchand partenaires. Ainsi l'API permettra de voir les contenus appartenant à un autre site et effectuer les différentes opérations que peuvent effectuer les clients.

CHAPITRE IV : DISCUSSION ET RECOMMANDATIONS

Introduction

Ce chapitre traite premièrement les discussions des résultats en comparant ce qui a été présenté comme travaux dans la revue de littérature, les contributions et améliorations que nous avons effectuées, mais surtout ce que notre solution apporte à l'entreprise. Deuxièmement nous avons donné l'évaluation et les coûts de la solution c'est-à-dire l'estimation du cout de la solution, les différentes recommandations pour la mise en place de notre système, les contraintes que peut subir le système dans le long terme et des nouvelles pistes de solution pour garantir son évolutivité.

SECTION 1 : DISCUSSION DES RESULTATS

A. APPORTS DU SYSTEME

Notre apport au sein de MALEKANI Sarlu consiste en une mise en place d'une API d'échange de donnée qui centralise les contenues et facilite l'échange de donnée entre les deux sites et qui permet d'améliorer la navigation chez les différent type de clients. Il s'agit principalement de développer une RestAPI qui sera une passerelle de communication entre deux sites, elle sera aussi entre de compare les éléments de deux bases de données pour éviter la duplication de donnée.

Voici ce qui a été exiger par MALEKANI Sarlu de manière détaillée, combiné à ce que nous avons donné comme solution :

I. La liste de produit ;

Suite à la non échange de données sur le site de MALEKANI Sarlu, la société nous a demandé une solution pouvant leur permettre de résoudre le souci de la visibilité de produit d'un site sur un autre site avec l'objectif de permettre aux clients d'accéder aux données des deux sites avec un seul compte, faciliter l'échange de données entre le deux sites, palier au problème de duplication de données à la partie cliente et améliore en equilibra le taux de visite sur les deux sites.

Notre solution générale étant une RestAPI d'échange de donnée elle leur offre dans ses fonctionnalités une possibilité de :

- Inter changer les données entre les deux sites c'est à dire les donnée B2B seront visible sur le B2C et vice versa.

- Un compte unique c'est-à-dire le client ne sera plus entre de créer un compte sur chacun de leurs deux sites mais avec un compte il aura accès aux données de deux sites.

En effet, avec le compte unique les visites, la visibilité de deux sites sera égale donc équilibrer mais aussi avec l'inter change de donnée nous sommes entre palier au problème non seulement de visibilité de produit mais aussi de duplication de données.

II. Passer la commande de produit ;

Ladite solution qui est une API, le client doit être à mesure de passer des commandes de produit peu importe dans quel site il se trouve. Cela est rendu possible grâce à la solution que

III. Faire le paiement de la commande ;

La société comporte Plusieurs type d'opération (achat, vente, réservation, précommande, commande etc...), et elles toutes à leur finalité nous voyons le paiement. Comment se fera le paiement ? et la question que se pose la société étant donné leur type de clients.

La société nous demande un moyen de paiement qui pourra nous seulement satisfaire à leur clients national, international ou local ou encore B2B ou B2C mais une solution simple, rapide, efficace et facilement accessible.

Nous leur avons proposé comme solution le paiement en ligne par carte bancaire ou par mobile money, un module de paiement qui sera inclus dans celle-ci et qui pourra s'effectuer en étant dans le même site.

B. APPORTS PAR RAPPORT A LA REVUE DE LITTÉRATURE

Les différents travaux que nous avons présentés dans la revue de littérature traitaient en général sur les API ou sur les échanges de données entre deux entités informatiques. Chacun des réalisations a essayé à sa manière de mettre en place une solution aux problèmes rencontrés dans les fonctionnements de tel ou tel autre système informatique.

L'avantage de notre solution est qu'elle peut être adaptée pour plusieurs organisations qui aimeraient vendre les produits de l'entreprise MALEKANI Sarlu. Elle n'est pas bornée uniquement à MALEKANI Sarlu et cette solution est arrivée à :

- Renforcer la vente de produit de l'entreprise ;
- Faciliter la navigation du client dans les sites de l'entreprise.

C. LES RISQUES

Les attaques ciblant les APIs sont l'une des menaces de sécurité les plus graves auxquelles les entreprises sont confrontées, car elles fournissent des accès directs à des données et des fonctionnalités sensibles. Et les attaquants ont pris conscience de la prévalence des APIs et de l'existence de vulnérabilités critiques dans ces interfaces. Le problème est que les applications web restent la cible principale des attaques (90% selon Verizon) et que les APIs représentent désormais 90% de la surface d'attaque des applications web. Ainsi, les APIs sont devenues un des principaux vecteurs d'attaque, avec des conséquences financières désastreuses pour les entreprises qui en font les frais. Les risques que court une api sont :

- Manque de Rate limiting, attaques DoS et attaques brute force sur l'API dont L'objectif est tout simplement de nuire au fonctionnement d'un service web ou à la réputation d'une entreprise offrant ce type de services ;
- Défaut de validation des entrées utilisateurs et attaques par injection.

D. LES MENACES

Voici les menaces potentielles qui peuvent affecter le fonctionnement continu de notre API :

- Vulnérabilités dans la gestion de l'authentification
- Exposition de données sensibles
- Défaut de contrôle d'accès au niveau des rôles utilisateurs

E. STRATÉGIES DES PRÉVENTION DES RISQUES ET MENACES

Pour prévenir toutes les menaces applicables, nous avons élaboré un ensemble des stratégies et mesures de sécurité pour garantir l'authenticité, l'intégrité et la confidentialité des données de l'API en :

- Utilisant un serveur avec un nom de domaine sécurisé : en HTTPS ;
- Implémenter des mécanismes de Rate Limiting pour contrer les attaques DoS et les attaques brute force ;
- Valider les entrées utilisateurs pour déjouer les attaques par injection ;

- Sécurisation de l'authentification et des autorisations dans l'API pour la réception de donnée

SECTION 2 : RECOMMANDATION

1. Éléments pour la mise en place du Système

Pour que MALEKANI Sarlu implémente notre API au sein de sa structure il devra :

- Former les techniciens et personnels sur le fonctionnement général de l'API
- Faire toujours les mises en jours ;

2. Cout du Système

Pour établir un budget sur le cout total, nous avons scindé notre projet sur l'aspect de la conception d'une application interface de programmation.

Dans la partie méthodologie, nous avons fait une estimation des coûts du projet, qui se sont avérés les mêmes jusqu'à l'aboutissement de notre travail au complet. Le coût total du projet est de 1584 \$.

CONCLUSION

Ce chapitre a essayé de relever les points amélioratifs de notre solution par rapport à ce qui a été demandé en évaluant l'impact de notre API et donner ses différents apports, risques et menaces potentielles en cas de mauvaise manipulation. Nous avons donné ensuite une liste détaillée des éléments fondamentaux pour la mise en place de celle-ci.

CONCLUSION GENERALE

Nous voici au terme de notre travail de mémoire qui a porté sur le « développement d'une API d'échange de données entre les sites web marchands partenaires d'une entreprise commerciale, cas de MALEKANI sarlu ».

Pour aborder notre sujet, et partant des problèmes constatés dans le quotidien de MALEKANI sarlu, la question suivante nous a servi de guide : « quelle solution applicative pouvons-nous développer pour interconnecter les sites e-commerces de MALEKANI Sarlu et aussi ceux de ses partenaires pour l'échange des données ? »

Cette dernière nous a amené à penser que, le développement d'une API serait la meilleure de solution qui permettrait à l'entreprise de :

- Faciliter les échanges des données entre leurs sites e-commerces et les partenaires ;
- Rendre accessible les produits de l'entreprise par toutes personnes voulant vendre les produits de MALEKANI Sarlu ;
- Améliorer la navigation chez les différents types de clients.

Pour ce faire, grâce à l'outil de modélisation UML aux méthodes RUP, le diagramme GANTT et les techniques tels que l'interview libre, l'observation, l'investigation, la documentation et en nous appuyant aux outils de développement MySQL et PHP nous avons réussi à concevoir une interface de programmation d'application qui facilite l'échange de données entre les sites web marchands partenaires de MALEKANI Sarlu.

Pour finir, nous pouvons conclure avoir répondu aux exigences du cahier des charges soumis par MALEKANI Sarlu. Mais nous sommes conscients que ce travail peut contenir quelques imperfections cependant, nous invitons les futurs chercheurs qui, d'une manière ou d'une autre, aimerait contribuer à l'amélioration de notre réalisation, de pouvoir aborder les aspects sur lesquels nous avons failli.

BIBLIOGRAPHIE

1. TAMBA, K., *Etude et mise en place d'un module automatisé d'échange de données entre l'université de LOME et l'université de KARA.*
2. Salatge, N., *Conception et mise en oeuvre d'une plate-forme pour la sûreté de fonctionnement des services Web*
3. KAKULE, C.O., *Cours de l'initiation à la recherche scientifique* 2016.
4. GRAWITZ., M., *Méthodes des sciences sociales.* Ed. Dalloz, Paris 1996 **1ère Ed.**
5. hat, R., *Api, qu'est-ce que.* 31 octobre 2020.
6. JDN, *API (interface de programmation) : définition, technos, exemples.* journalunet.fr, 2021.
7. JobFonning, *interface de programme d'application* 2022.
8. Escardo, D., *What Is an API? (The Simple Definition)* 2020.
9. Techopedia, *What does Application Programming Interface mean.* 12 Février 2022.
10. Walicer, A., *What is an API full form, meaning, definition, types and exemple.* 2022.
11. Dabi-Schwebel, G., *site web* 2022.
12. Canevet, F., *Définition d'un site web.* 2022.
13. Laurent, S., *Qu'est-ce qu'un site web* 7 Avril 2021.
14. Marie, *Tout savoir sur la notion de site internet* 27 Juin 2020.
15. Bon, C., *Qu'est-ce qu'un site web ? définition* 20 Novembre 2021.
16. MAROTEL, A., *Définition website ou site web* 2022.
17. Glossaire, *E-COMMERCE.* 2021.
18. Bathelot, B., *E-COMMERCE.* 14 Juillet 2021.
19. Wesley Draï, B.H.a.B.c., *E-COMMERCE.* 2021.
20. Bloomenthal, A., *Electronic commerce* 16 Juin 2021.
21. Zande, J.V., *Whats is an e-commerce* 2021.
22. Wikipedia, *E-COMMERCE.* 2020.
23. Rédaction, J., *Une société commerciale.* 2021.
24. Belgiki, *Société commerciale.* 2021.
25. Verheyen, W. and M.K. Kołacz, *Enhancing safety in B2C delivery chains.* Transport Policy, 2022. **117**: p. 12-22.
26. George, M., *Qu'est-ce qu'une entreprise commerciale* 2021.
27. Lailler, T., *Qu'est-ce qu'une société commerciale.* 2022.
28. Moovago, *C'est quoi le B2C, Définition, exemple, différences.* 2021.
29. Illusion, P., *DÉFINITION DE B2C.* 2021.
30. JDN, L.R., *B2C ou B to C : définition simple et traduction de business to consumer.* 15/01/2019.
31. TIPS, A., *Business to Consumer (B2C).* 26/09/2020.
32. net, I., *B2C.* 12/08/2020.
33. IONOS, D.g., *B2C : business to consumer.* 2019.
34. Ooreka, *B2B.* 2022.
35. illusion, P., *Définition de B2B.* 2021.
36. JOPTIMISEMONSITE, *Le BtoB, c'est quoi ? Comprendre le B to B en 2min seulement.* 2020.
37. Glossaire, *B to B (B2B).* 2020.
38. wikipedia, *Business to business.* 2020.
39. JDN, L.R., *B2B ou B to B : définition simple et traduction de business to business.* 25/12/2019.
40. Linkweb, *Front-End.* 2021.
41. net, J.d., *html-hypertext-markup-langage-definition.* 2021.

42. Robert, D.I., *HTML*. 2020.
43. mozilla, D., *Qu'est-ce que CSS*. 2020.
44. Glossaire, *Css*. 2021.
45. Lemagit, *Bootstrap*. 2020.
46. Hostinger, *C'est quoi bootstrap*. 2021.
47. net, J.d., *Php hypertext preprocessor definition*. 2021.
48. Codingame, *le php les bases du langage*. 2020.
49. wikipedia, *JavaScript*. 2019.
50. Developer, *What is JavaScript*. 2021.
51. Redhat, *What is a rest api*. 2021.
52. Oracle, *Définition de la base de données*. 2020.
53. Kinsta, *Qu'est-ce-que mysql*. 2021.
54. Geonov, *Architecture client serveur*. 2021.
55. Sine, M., E. Emonet, and H. Théo-Paul, *API-AGRO : Création d'une plateforme d'échange de données agricoles fédératrice d'acteurs publics et privés*. 2019. **71**: p. 211-224.
56. Ong, S.P., et al., *The Materials Application Programming Interface (API): A simple, flexible and efficient API for materials data based on REpresentational State Transfer (REST) principles*. Computational Materials Science, 2015. **97**: p. 209-215.
57. Meng, M., S. Steinhardt, and A. Schubert, *Application Programming Interface Documentation: What Do Software Developers Want?* Journal of Technical Writing and Communication, 2017. **48**(3): p. 295-330.
58. Balsari, S., et al., *Reimagining Health Data Exchange: An application programming interface-enabled roadmap for India*. Journal of medical Internet research, 2018. **20**(7): p. e10725.
59. Muller, *Modélisation Objet avec UML*. 2007.
60. Messenger, V., *Gestion des projets agile : avec Scrum, Lean, EXtreme Programming*. 2013: p. p. 42,49.

TABLE DE MATIERE

SOMMAIRE	i
EPIGRAPHE.....	iii
DEDICACE.....	iv
REMERCIEMENTS	v
SIGLES ET ABREVIATIONS	vi
LISTE DES TABLEAUX	vii
LISTE DES FIGURES.....	viii
RESUME ET MOTS CLES	ix
ABSTRACT AND KEYWOR.....	x
INTRODUCTION GENERALE.....	1
I.2. PROBLEMATIQUE	3
I.3. HYPOTHESES	4
I.4. OBJECTIFS	4
I.5. METHODE ET TECHNIQUE.....	4
I.5.1. Méthode.....	4
I.5.2. Techniques	5
I.6. DELIMITATION DU SUJET.....	6
I.7. CHOIX ET INTERET DU SUJET	6
I.7.1. Intérêt personnel.....	6
I.7.2. Intérêt commun	6
I.8. SUBDIVISION	7
Chapitre 1 : ÉTAT DES LIEUX ET ANALYSE.....	8
INTRODUCTION	8
SECTION 1 : DEFINITION DE CONCEPTS UTILISES.....	8
1. Définition de concepts du sujet	8
2. Définition de technologie utilisées	15
SECTION 2 : CONTEXTE D'ETUDE.....	26
1.2.1. Description de l'environnement d'étude	26
1.2.1.1. Présentation de MALEKANI SARLU	26
1.2.1.2. Le problème social.....	28
1.2.1.2. Le problème de l'entreprise	29
CONCLUSION	30
Chapitre 2. REVUE DE LITTERATURE	31
INTRODUCTION	31
SECTION 1 : TRAVAUX SIMILAIRES	32
2.1.1. Travaux.....	32

2.1.2 Articles	36
SECTION 2 : METHODE DES RESOLUTION	40
2.2.1 Cycle de développement Agile.....	40
2.2.2 Diagramme de résolution.....	41
CONCLUSION	42
Chapitre 3 : METHOLOGIE ET RESULTATS	43
INTRODUCTION	43
Section 1. METHODOLOGIE	43
3.1 Cahier des charges	43
3.2. Planning.....	43
SECTION 2 : RESULTATS ET ANALYSE	49
3.2.1 Modélisation	49
CONCLUSION	60
CHAPITRE IV : DISCUSSION ET RECOMMANDATIONS.....	61
Introduction	61
SECTION 1 : DISCUSSION DES RESULTATS	61
SECTION 2 : RECOMMANDATION	64
CONCLUSION	64
CONCLUSION GENERALE	65
BIBLIOGRAPHIE	66
TABLE DE MATIERE	68