

SOMMAIRE

INTRODUCTION	3
I. CONTEXTE ET OBJECTIFS.....	3
1. Contexte général du projet.....	3
2. Objectifs du projet.....	4
3. Besoins à satisfaire.....	4
4. Client concerné.....	5
II. OBJECTIFS DÉTAILLÉS.....	5
1. Fonctions principales à réaliser	5
2. Cas d'utilisation ou scénarios utilisateur	6
III. DESCRIPTION TECHNIQUE.....	6
1. Technologies envisagées	6
2. Architecture du système.....	8
IV. PERIMETRE DU PROJET	8
1. Ce qui est inclus dans le projet	8
2. Ce qui est explicitement exclu.....	9
3. Limites fonctionnelles.....	9
V. ORGANISATION DU TRAVAIL.....	9
1. Répartition des tâches entre les membres du groupe.....	9
2. Répartition des tâches collectives	10
3. Méthodologie de travail	11
VI. FONCTIONNALITÉS DE L'APPLICATION	11
1. Interface vendeur	11
2. Interface client	13
VII. CONTRAINTES	14
1. Contraintes de temps	14
2. Contraintes techniques	14
3. Contraintes budgétaires.....	14

4.	Tests et validation	15
5.	Maintenance	15
VIII.	LIVRABLES.....	15
1.	Livrables prévus en L2	15
2.	LIVRABLES PREVUS EN L3	16
IX.	DEROULEMENT DU DEVELOPPEMENT	16
1.	Méthodologie appliquée.....	17
2.	Phases clés du projet	17
	CONCLUSION.....	18
X.	BIBLIOGRAPHIE ET WEBOGRAPHIE	19
1.	Sources internes.....	19
2.	Webographie	19
3.	Outils d'intelligence artificielle et d'assistance numérique.....	20

INTRODUCTION

Dans le cadre de notre formation en Licence 2 à l'Université de Technologie et de Télécommunication de Loko (UTT Loko), nous réalisons un projet tutoré ayant pour thème : **« Réalisation d'une application web de gestion de stocks et de commandes »**.

Ce projet s'inscrit dans une démarche d'apprentissage pratique des concepts étudiés en cours, notamment en développement web, gestion de bases de données, modélisation UML, et conduite de projet. L'objectif est de mettre en œuvre les compétences acquises pour concevoir et développer une solution informatique complète répondant à un besoin concret de gestion.

L'application visée permettra à un vendeur de gérer efficacement ses produits, commandes et clients, tout en offrant à ces derniers une interface fluide pour consulter les articles, passer commande et suivre leurs achats. Ce projet représente donc une opportunité de développement personnel, de travail en équipe, et d'approfondissement des technologies modernes du web.

I. CONTEXTE ET OBJECTIFS

1. Contexte général du projet

Ce projet est réalisé dans le cadre d'un **projet tutoré universitaire** en deuxième année de Licence à l'Université de Technologie et de Télécommunication de Loko (UTT Loko). Il s'inscrit dans une dynamique pédagogique visant à faire appliquer de manière concrète les compétences acquises en développement web, modélisation, gestion de bases de données et conduite de projet.

Le thème retenu est la **réalisation d'une application web de gestion de stocks et de commandes**, répondant à un besoin réel d'optimisation des opérations de

ventes en ligne. Ce projet simule un environnement professionnel où un vendeur gère une boutique en ligne et interagit avec des clients via une interface intuitive.

2. Objectifs du projet

Le projet a pour objectif principal de concevoir une plateforme complète, moderne et sécurisée, destinée à deux types d'utilisateurs : les vendeurs et les clients.

➤ Objectifs côté vendeur :

- Gérer les produits (ajout, modification, suppression)
- Suivre et traiter les commandes (acceptation, expédition, annulation)
- Consulter des statistiques sur les ventes
- Être alerté en cas de stock faible
- Communiquer avec les clients via une messagerie intégrée

➤ Objectifs côté client :

- Consulter les produits disponibles
- Passer des commandes et choisir un mode de paiement (en ligne ou à la livraison)
- Suivre le statut de leurs commandes
- Contacter les vendeurs via la messagerie intégrée

3. Besoins à satisfaire

L'application devra répondre aux besoins suivants :

- **Centraliser la gestion des stocks et des commandes** à travers une interface simple et fluide.
- **Assurer une communication fluide** entre les vendeurs et les clients grâce à un système de messagerie intégré.

- **Permettre un suivi en temps réel des commandes**, de l'enregistrement à la livraison.
- **Générer des statistiques de vente** pour aider le vendeur à prendre des décisions.

4. Client concerné

Le projet étant réalisé à des fins pédagogiques, **le client est interne**. Il s'agit de l'équipe enseignante, en particulier **M Camara**, qui joue le rôle de commanditaire et d'encadrant. Aucune entreprise réelle n'est impliquée. Le projet est fictif mais s'appuie sur une problématique réaliste du e-commerce.

II. OBJECTIFS DÉTAILLÉS

1. Fonctions principales à réaliser

L'application web à réaliser doit offrir les fonctionnalités essentielles suivantes, réparties selon les deux profils d'utilisateurs (vendeur et client) :

► Côté vendeur :

- Authentification sécurisée
- Gestion des produits (ajout, modification, suppression)
- Gestion des commandes (suivi, traitement, annulation)
- Alertes automatiques en cas de stock faible
- Consultation de statistiques de ventes
- Système de messagerie avec les clients
- Mise à jour du profil vendeur

► Côté client :

- Inscription et connexion
- Consultation du catalogue de produits

- Filtrage et recherche de produits
- Ajout au panier et passage de commande
- Suivi de l'état de commande
- Paiement en ligne (mobile money) ou à la livraison
- Messagerie pour contacter le vendeur

2. Cas d'utilisation ou scénarios utilisateur

Voici quelques scénarios types qui illustrent les interactions prévues dans l'application :

- **Scénario 1** : Un client visite le site, consulte les produits disponibles, ajoute des articles à son panier, passe commande, choisit le paiement à la livraison, puis suit l'état de sa commande.
- **Scénario 2** : Un vendeur se connecte à son tableau de bord, consulte les commandes récentes, accepte une commande, expédie le colis, et échange un message avec le client concerné.
- **Scénario 3** : Un client contacte un vendeur via la messagerie pour poser une question sur un produit, et reçoit une réponse dans l'interface prévue à cet effet.

III. DESCRIPTION TECHNIQUE

1. Technologies envisagées

Le projet repose sur un ensemble d'outils et de technologies web modernes, choisis pour leur fiabilité, leur simplicité de mise en œuvre et leur compatibilité avec les exigences pédagogiques du projet.

► Frontend (interface utilisateur)

- **HTML5** : structure des pages
- **CSS3** : mise en forme
- **JavaScript** : interactivité

- **Tailwind CSS** : framework CSS utilitaire pour une mise en page responsive, moderne et rapide

► Backend (serveur)

- **PHP** : langage de programmation serveur
- **Laravel** : framework PHP robuste, basé sur le modèle MVC, offrant des fonctionnalités intégrées (routage, sécurité, ORM, etc.)

► Base de données

- **MySQL** : système de gestion de base de données relationnelle, utilisé pour stocker les produits, commandes, utilisateurs, messages...

➤ Sécurité

Pour garantir la sécurité des données échangées et stockées :

- **HTTPS / SSL** : sécurisation des échanges client-serveur.
- **Cryptage des mots de passe** : utilisation d'algorithmes de hachage comme bcrypt.
- **Vérification d'identité du vendeur** : demande de CNI lors de l'inscription, vérifiée manuellement.

➤ Messagerie

- **Socket.io** (ou technologie équivalente) sera utilisé pour gérer la messagerie en temps réel entre les vendeurs et les clients.

► Outils complémentaires

- **VS Code** : éditeur de code
- **XAMPP / Laragon** : environnement local de développement (Apache, PHP, MySQL)

- **PhpMyAdmin** : interface graphique pour gérer la base de données
- **Git & GitHub** : gestion de version et travail collaboratif

2. Architecture du système

Le projet adopte une **architecture client-serveur web** :

- Le **client (navigateur)** accède à une interface web responsive conçue avec HTML, Tailwind CSS et JavaScript.
- Le **serveur** gère les requêtes, le traitement des données, l'authentification, les transactions et l'envoi de réponses via PHP/Laravel.
- La **base de données MySQL** est hébergée côté serveur et gère de manière centralisée toutes les informations liées aux produits, utilisateurs, commandes et messages.

Le système est accessible via un navigateur moderne, sans besoin d'installation côté client.

IV. PERIMETRE DU PROJET

1. Ce qui est inclus dans le projet

Le projet couvre les éléments suivants :

- Le développement d'une **application web complète** avec deux interfaces : une pour les vendeurs, une pour les clients.
- La **gestion complète des produits** : ajout, modification, suppression, affichage.
- La **gestion des commandes** : enregistrement, traitement, suivi.
- Un **système de messagerie intégré** entre vendeurs et clients.
- La possibilité pour le client de choisir un **mode de paiement** : mobile money, carte bancaire ou paiement à la livraison (simulation).
- Un **tableau de bord** pour le vendeur avec des **statistiques** (ventes, produits les plus vendus...).

- Une **base de données relationnelle** gérée avec MySQL.
- Une **documentation complète** (technique, utilisateur, installation).

2. Ce qui est explicitement exclu

Certains éléments sont hors du périmètre de ce projet, notamment :

- Le développement d'une **application mobile native** (Android ou iOS).
- L'intégration de **services de livraison externes** (Colisivoire, DHL, etc.).
- L'utilisation d'un **système de paiement bancaire réel sécurisé** (ex : Stripe, PayPal officiel).
- La gestion **multi-vendeurs** ou multi-boutiques.

3. Limites fonctionnelles

- **Un seul vendeur** peut gérer la plateforme (structure mono-boutique).
- L'interface reste **accessible uniquement en français**.
- Le paiement mobile et à la livraison est **simulé** (pas de transaction réelle).
- Le système ne prévoit pas encore de **gestion des retours** ou remboursements automatisés.

V. ORGANISATION DU TRAVAIL

La réalisation de ce projet repose sur une collaboration étroite entre les membres du groupe, chacun ayant un rôle bien défini selon ses compétences. Cette répartition vise à optimiser l'efficacité de l'équipe tout en favorisant l'apprentissage collectif.

1. Répartition des tâches entre les membres du groupe

Le projet est réalisé par une équipe de quatre étudiants. Chacun assume un rôle précis selon ses compétences, tout en participant aux activités transversales comme les tests et la documentation.

- **Koffi Mougoh Christ** (Chef de groupe et développeur full stack) :
Il coordonne l'ensemble du projet, organise les réunions d'équipe, s'assure du respect du planning et participe au développement complet de l'application, tant côté client (frontend) que serveur (backend). Il est également garant de la cohérence globale du projet.
- **Kodjo Loïc** (Développeur frontend)
Il est chargé de concevoir et de développer l'interface utilisateur. Il s'occupe de l'intégration des pages web et veille à l'ergonomie ainsi qu'à la compatibilité responsive de l'application.
- **Kone Zakaria** (Responsable base de données)
Il conçoit le modèle de données, crée et gère la base de données (MySQL). Il met en place les relations entre les entités, assure la cohérence des données et veille aux performances des requêtes utilisées dans l'application.
- **Konkobo Christ** (Développeur backend)
Il s'occupe de la logique métier, du traitement des requêtes côté serveur et de la mise en œuvre des API. Il gère l'intégration avec la base de données et assure la sécurité des communications avec le frontend.

2. Répartition des tâches collectives

- **Modélisation UML** : Réalisée en collaboration, sous la supervision du chef de groupe.
- **Tests** : Tous les membres participeront aux tests fonctionnels, de sécurité et de validation. Chaque fonctionnalité développée sera testée de manière croisée par un autre membre pour garantir la qualité du produit.
- **Documentation** : La rédaction des différents livrables (cahier des charges, manuel d'utilisation, etc.) sera répartie entre les membres, selon les parties qu'ils ont traitées dans le développement.

3. Méthodologie de travail

Le groupe adopte une méthodologie inspirée du **cycle en V**, bien adaptée au cadre pédagogique :

- **Analyse** : compréhension du besoin, identification des utilisateurs et des cas d'usage
- **Conception** : modélisation UML, création de la base de données, maquettes
- **Développement** : implémentation côté client et serveur, tests continus
- **Tests** : vérification fonctionnelle et validation
- **Livraison** : finalisation, documentation et présentation orale

Cette organisation favorise la rigueur, la séparation des responsabilités et l'avancement progressif du projet.

VI. FONCTIONNALITÉS DE L'APPLICATION

L'application web de gestion de stocks et de commandes propose deux interfaces distinctes : une pour les vendeurs, et une pour les clients. Chaque interface comprend des fonctionnalités spécifiques adaptées aux besoins de ses utilisateurs.

1. Interface vendeur

L'espace vendeur permet à l'utilisateur de gérer ses activités commerciales. Les principales fonctionnalités disponibles sont :

- **Authentification** :
 - Connexion sécurisée avec email et mot de passe
 - Option "Mot de passe oublié"
 - Page d'inscription avec saisie des informations personnelles et **téléversement de la photo de la CNI** pour vérification
- **Tableau de bord** :

- Vue synthétique : total des ventes, produits les plus vendus, commandes en attente, alertes de stock
- Accès rapide aux fonctions clés : ajout de produit, gestion de commandes, messagerie
- **Gestion des produits :**
 - Ajouter, modifier ou supprimer un produit
 - Informations : nom, description, prix, quantité en stock, image
- **Gestion des commandes :**
 - Affichage des commandes passées avec leur statut (en attente, expédiée, annulée, etc.)
 - Détails : produits commandés, informations client, adresse de livraison
- **Statistiques de ventes :**
 - Graphiques de performance par mois, trimestre ou année
 - Classement des produits les plus/moins vendus
- **Messagerie intégrée :**
 - Communication avec les clients via une interface de messagerie
 - Réponse aux demandes concernant les produits ou les commandes
- **Profil vendeur :**
 - Affichage et modification des informations personnelles
 - Changement de mot de passe, mise à jour de la photo de profil
- **Alertes de stock faible :**
 - Notification lorsque la quantité d'un produit passe en dessous d'un seuil critique

- Option de réapprovisionnement ou désactivation temporaire du produit

2. Interface client

L'espace client permet à l'utilisateur final de naviguer, acheter et suivre ses commandes de manière fluide :

- **Authentification :**
 - Connexion avec email et mot de passe
 - Option "Mot de passe oublié"
- **Catalogues produits :**
 - Navigation avec filtres (catégorie, prix, popularité, etc.)
 - Fiches produits détaillées : images, descriptions, prix, disponibilité
- **Commande :**
 - Ajout au panier, validation de la commande
 - Choix du mode de paiement :
 - Carte bancaire
 - Mobile money (ex: Wave, Moov Money, MTN Mobile Money)
 - Paiement à la livraison
- **Suivi de commande :**
 - Consultation du statut (en traitement, expédiée, livrée)
 - Historique des commandes passées
- **Messagerie avec le vendeur :**
 - Poser des questions ou signaler un problème
- **Notifications :**

- Alertes par email ou plateforme pour confirmation de commande, mise à jour du statut, réponse du vendeur
- **Sécurité :**
 - Utilisation du protocole HTTPS
 - Cryptage des données sensibles (ex : mots de passe)
 - Validation rigoureuse côté serveur

VII. CONTRAINTES

1. Contraintes de temps

Le projet s'inscrit dans le cadre d'un projet tutoré universitaire et doit être réalisé dans un délai limité. La durée estimée de développement est d'environ **une année**, répartis entre les phases de conception, développement, tests et finalisation. Un respect rigoureux du planning est essentiel pour garantir la livraison des livrables à temps.

2. Contraintes techniques

Le projet devra prendre en compte :

- L'utilisation d'une architecture moderne et évolutive (Laravel + MySQL)
- Une bonne compatibilité entre les navigateurs web
- L'adaptabilité de l'interface aux différents supports (desktop, mobile)
- La sécurisation des échanges et des données

3. Contraintes budgétaires

Étant donné qu'il s'agit d'un projet tutoré universitaire réalisé dans un cadre pédagogique, **aucune contrainte financière n'est imposée**. Les outils et ressources utilisés sont majoritairement libres, open-source ou déjà mis à disposition des étudiants.

4. Tests et validation

L'application fera l'objet de **tests rigoureux** tout au long de son développement. Ces tests permettront de s'assurer du bon fonctionnement des fonctionnalités attendues, de la fiabilité des traitements, et de la sécurité des données.

Les tests seront réalisés de manière collaborative par tous les membres de l'équipe. Ils incluront :

- Des tests de fonctionnement général
- Des vérifications de sécurité (validation des saisies, gestion des accès)
- Des simulations d'utilisation client / vendeur

5. Maintenance

Une fois l'application livrée, une phase de maintenance est prévue pour :

- Corriger d'éventuels bugs ou incohérences
- Apporter des améliorations fonctionnelles
- Mettre à jour les composants techniques si nécessaire

VIII. LIVRABLES

La réalisation du projet est structurée en deux grandes phases : la première en Licence 2 (L2) correspondant à la phase d'analyse et de planification, et la seconde en Licence 3 (L3), orientée vers le développement complet et la livraison finale.

1. Livrables prévus en L2

Durant cette phase préparatoire, les documents suivants seront produits :

- **Cahier des charges** : document de base décrivant le besoin, les objectifs, les acteurs, les fonctionnalités attendues, les technologies, les contraintes, et le cadre général du projet.

- **Plan de gestion du projet** : contient la répartition des tâches, le diagramme de Gantt et l'organisation du travail en équipe.
- **Cahier de recette** : spécifie les critères de validation des fonctionnalités, les scénarios d'utilisation et les conditions d'acceptation de l'application.

2. LIVRABLES PREVUS EN L3

À l'issue de la phase de développement, plusieurs livrables seront remis pour finaliser le projet :

- **Application fonctionnelle** : version complète, stable et testée de l'application web.
- **Code source documenté** : ensemble du code produit pendant le développement, organisé et commenté.
- **Manuel d'installation** : guide détaillant les étapes pour installer et configurer l'application sur un serveur ou en local.
- **Manuel d'utilisation** : destiné aux utilisateurs finaux (clients et vendeurs), il explique comment utiliser les principales fonctionnalités de l'application.
- **Manuel de conception** : regroupe tous les modèles de conception (y compris les diagrammes UML si mis à jour), les choix techniques et les justifications.
- **Rapport final** : document de synthèse du projet retraçant toutes les étapes (du contexte à la mise en production).
- **Présentation finale** : support visuel pour la soutenance orale.

IX. DEROULEMENT DU DEVELOPPEMENT

Le développement du projet se fera selon une approche **progressive, collaborative et structurée**, suivant les bonnes pratiques du développement logiciel. L'équipe a choisi d'adopter une **méthodologie inspirée du cycle en V**, adaptée au contexte

pédagogique, afin de bien séparer les étapes d'analyse, de conception, de réalisation et de validation.

1. Méthodologie appliquée

La méthode retenue repose sur les principes suivants :

- Une phase initiale d'**analyse des besoins** et de modélisation (réalisée en L2)
- Une phase de **développement technique** (prévue en L3)
- Des **tests continus** au fur et à mesure de l'avancement
- Une **révision collaborative** des livrables
- Un **retour régulier au professeur encadrant** pour validation intermédiaire

Cette organisation permet d'assurer la qualité du code, la traçabilité des décisions, et la conformité avec les attentes pédagogiques.

2. Phases clés du projet

Le projet est structuré en plusieurs étapes principales :

1. Analyse et planification

- Étude du besoin
- Rédaction du cahier des charges
- Répartition des tâches
- Élaboration du diagramme de Gantt

2. Conception

- Réalisation des diagrammes UML
- Conception de la base de données
- Conception des interfaces (maquettes)

3. Développement

- Développement frontend (pages vendeur et client)
- Développement backend (API, logique métier, sécurité)
- Intégration de la base de données et messagerie

4. Tests et validation

- Tests fonctionnels sur chaque module
- Vérification des performances
- Relecture croisée entre membres du groupe

5. Documentation et préparation finale

- Rédaction du manuel d'installation et d'utilisation
- Assemblage du rapport final
- Préparation de la soutenance orale

CONCLUSION

Le projet de développement d'une application web de gestion de stocks et de commandes s'inscrit pleinement dans notre parcours de formation à l'UTT Loko. Il nous offre l'opportunité de mettre en pratique les connaissances théoriques acquises en développement web, modélisation UML, gestion de base de données et travail en équipe.

Ce **cahier des charges** constitue un **document de référence essentiel**. Il encadre le projet du début à la fin, en définissant les objectifs à atteindre, les fonctionnalités attendues, les outils à utiliser, les contraintes à respecter et les étapes à suivre. Il permet également de garantir une compréhension commune entre tous les membres du groupe et l'encadrant.

Grâce à une planification rigoureuse, une répartition claire des tâches et une méthode de développement bien structurée, l'équipe est engagée à livrer une application

fonctionnelle, intuitive et conforme aux exigences spécifiées. Ce document guidera toutes les phases du projet, depuis l'analyse jusqu'à la présentation finale.

X. BIBLIOGRAPHIE ET WEBOGRAPHIE

1. Sources internes

Première version du cahier des charges – Projet tutoré L2, avril 2025.

Document de travail initial rédigé par les membres du groupe et validé en partie par l'encadrant M Camara.

Utilisé comme base pour la structuration et la reformulation du cahier final.

2. Webographie

1. Laravel – Documentation officielle

URL : <https://laravel.com/docs>

Consulté le **04 juin 2025 à 14h27**

2. Tailwind CSS – Documentation

URL : <https://tailwindcss.com/docs>

Consulté le **04 juin 2025 à 14h33**

3. MDN Web Docs – Documentation HTML, CSS, JS

URL : <https://developer.mozilla.org/fr/>

Consulté le **04 juin 2025 à 14h45**

4. MySQL – Documentation officielle

URL : <https://dev.mysql.com/doc/>

Consulté le **04 juin 2025 à 14h53**

5. Socket.io – Messagerie temps réel

URL : <https://socket.io/docs/>

Consulté le **04 juin 2025 à 15h02**

6. Draw.io – Création de diagrammes UML

URL : <https://app.diagrams.net>

Consulté le **04 juin 2025 à 15h08**

7. Google Search – Moteur de recherche

URL : <https://www.google.com>

Utilisé pour trouver des ressources techniques et des exemples de mise en page

Consulté entre **04 juin 2025, 14h00 et 15h30**

3. Outils d'intelligence artificielle et d'assistance numérique

1. ChatGPT (OpenAI)

URL : <https://chat.openai.com>

Utilisé pour la reformulation des parties, la structuration du sommaire, la rédaction des sections (livrables, fonctionnalités, contraintes, etc.) et la vérification orthographique.

Consulté le **04 juin 2025 entre 13h40 et 16h00**

2. Grammarly (version gratuite)

URL : <https://www.grammarly.com>

Utilisé ponctuellement pour la correction automatique d'orthographe et de style.

Consulté le **04 juin 2025 à 15h10**

3. Google Docs & Microsoft Word

Utilisés pour la rédaction collaborative, la mise en page, l'insertion du sommaire et des styles (polices, titres, interlignes).