

RAPPORT DE TRAVAIL

Système de gestion de stock d'un magasin de vêtements

Conception et développement d'un système de gestion (Stock, Ventes, Caisse, Clients, RH)

Nom : Rania Mhedhbi

Établissement : ISITCOM Sousse

Entreprise : Capwaiv(MAKEWEBSITE.IO)

Projet : Système de gestion de stock d'un magasin de vêtements

1. Résumé du travail réalisé

Ce rapport présente le travail réalisé dans le cadre du projet de gestion de stock pour un magasin de vêtements, développé au sein de l'entreprise Capwaiv (makewebsite.io). L'objectif était de concevoir un système de gestion d'entreprise (ERP) complet, permettant de gérer l'ensemble des opérations du magasin : articles, stock, ventes, services, caisse, dépenses, clients, fournisseurs et ressources humaines.

Le système développé est l'équivalent d'un logiciel comme Sage ou QuickBooks, mais adapté à une entreprise tunisienne, avec une interface entièrement en français et une monnaie affichée en Dinar Tunisien (DT). Il est organisé autour de 10 modules principaux, décrits en détail dans les sections suivantes.

2. Présentation de l'entreprise et du contexte

Ce projet a été réalisé au sein de l'entreprise Capwaiv (makewebsite.io), dans le cadre d'un stage effectué par Rania Mhedhbi, étudiante à l'ISITCOM Sousse. Le besoin exprimé par l'entreprise était de fournir aux magasins de vêtements un outil simple et complet pour gérer leur stock, leurs ventes et leur trésorerie au quotidien, sans dépendre de solutions étrangères coûteuses et mal adaptées au contexte local.

3. Description générale du système

Le système est un ERP (Enterprise Resource Planning) complet, conçu pour gérer toutes les opérations d'un commerce ou d'une petite/moyenne entreprise. Il couvre la gestion du stock, des ventes, des services, de la trésorerie, des dépenses, des clients, des fournisseurs et des employés, tout en calculant automatiquement les bénéfices, les salaires et les alertes en temps réel.

4. Module 1 — Articles & Stock

Ce module permet la gestion complète des articles et du suivi du stock du magasin.

- Créer des articles avec : nom, famille, catégorie, fournisseur, couleur, photo, prix d'achat, prix de vente au détail, prix de vente en gros, prix de location min/max
- Suivre le stock en temps réel
- Voir l'historique de chaque mouvement (entrée/sortie)
- Alertes automatiques quand le stock est bas (≤ 5)

5. Module 2 — Ventes

Ce module gère l'ensemble du processus de vente, du détail au gros.

- Créer des ventes en détail ou en gros
- Ajouter plusieurs articles par vente avec quantité, prix unitaire, remise
- Génération de facture PDF avec TVA 19%
- Le stock se décrémente automatiquement à chaque vente
- Modifier ou annuler une vente (le stock est remis automatiquement)

6. Module 3 — Services

Ce module permet de gérer les prestations complémentaires proposées par le magasin.

- Gérer 4 types de services : Location, Réparation, Livraison, Assistance
- Pour les locations : définir les dates de début/fin par article
- Paiement en 1 fois ou en 2 tranches
- Gestion des cautions (dépôts de garantie)
- Évaluation de l'état du produit à la restitution (Bon / Acceptable / Endommagé + pénalité)
- Assignation d'un chauffeur pour les livraisons

7. Module 4 — Caisse du Jour

Ce module assure le suivi quotidien de la trésorerie du magasin.

- Journal quotidien de toutes les transactions
- Calcul automatique : Revenus – Dépenses = Bénéfice
- Graphique de tendance des 7 derniers jours
- Objectif quotidien personnalisable avec barre de progression
- Ajout de mouvements manuels (entrée/sortie)
- Alertes pour les cautions et retours de location du lendemain
- Impression du journal journalier

8. Module 5 — Dépenses

Ce module permet d'enregistrer et de suivre l'ensemble des dépenses du magasin.

- Enregistrer toutes les dépenses avec : catégorie, libellé, montant, mode de paiement (espèces, chèque, virement, carte)
- Justificatif photo (reçu/facture)
- Statut de paiement (Payée / En attente / Partiellement payée)
- Lier à un fournisseur ou un employé
- Date d'échéance pour les dépenses à payer
- Impression de reçu de dépense

9. Module 6 — Clients

Ce module centralise les informations relatives aux clients du magasin.

- Fiche complète : nom, téléphone, CIN, adresse, email, âge
- Import en masse de clients
- Recherche rapide

10. Module 7 — Fournisseurs

Ce module gère les informations relatives aux fournisseurs du magasin.

- Fiche fournisseur : nom, immatricule, contact, téléphone, email, adresse
- Utilisé dans les modules Articles et Dépenses

11. Module 8 — Ressources Humaines (RH)

Ce module permet la gestion complète du personnel du magasin.

- Fiche complète par employé : poste, département, contrat, salaire, banque, RIB, CNSS
- Calendrier de présence mensuel : Présent / Absent / Retard / Congé Payé / Congé Non-Payé
- Calcul automatique du salaire net : Base – absences (100%) – retards (50%) – avances + heures supplémentaires (+25%)
- Gestion des avances sur salaire (débitées automatiquement de la caisse)
- Historique des paiements avec reçu de salaire imprimable
- Fiche ouvrier PDF exportable

12. Module 9 — Dashboard (Tableau de bord)

Ce module offre une vue d'ensemble de l'activité du magasin.

- Vue globale : revenus totaux, dépenses, bénéfice net, valeur du stock
- Filtres par période : Semaine / Mois / Trimestre / Année / Tout
- Graphique en barres (6 derniers jours)
- Alertes : stock bas, locations en retard, cautions actives
- Masse salariale automatique

13. Module 10 — Historique Articles

Ce module donne une analyse détaillée de la performance de chaque article.

- Tableau de performance détaillée par article
- Pour chaque article : ventes détail + gros + locations, chiffre d'affaires total, coût estimé, bénéfice net, marge %
- Totaux globaux affichés en haut du tableau

14. Authentification et sécurité

- Inscription avec nom d'entreprise et logo
- Connexion sécurisée (mot de passe hashé)
- Session persistante (survit aux redémarrages de MySQL)

15. Double interface

- Mode Web : navigateur standard, accessible à l'adresse <http://localhost/>
- Mode Desktop : application Windows (Electron) avec multi-fenêtres style Windows 11, barre de tâches et raccourcis clavier (F1, F2, F3, F12)

16. Fonctions d'impression

- Facture client (TTC / HT)

- Journal de caisse du jour
- Fiche de paie employé
- Reçu de dépense
- État du stock

17. Automatisation du démarrage (fichier .bat)

Afin de simplifier le lancement quotidien de l'application pour l'utilisateur final, un fichier de démarrage automatique (.bat) a été mis en place. Ce script, exécutable en un seul double-clic, effectue les opérations suivantes :

- Démarrage du serveur Apache (et de MySQL si nécessaire)
- Attente de quelques secondes le temps que les services soient pleinement opérationnels
- Ouverture automatique du site en local (localhost) dans le navigateur par défaut

Cette automatisation évite à l'utilisateur de devoir démarrer manuellement chaque service depuis le panneau de contrôle (par exemple XAMPP), réduisant ainsi les erreurs de manipulation et le temps de mise en route de l'application. Le fichier correspondant (demarrer_site.bat) est fourni séparément et peut être adapté selon le chemin d'installation du serveur local et l'adresse exacte du site.

1. Résumé du travail réalisé

l'ensemble des défauts identifiés dans le projet ainsi que les solutions apportées à chacun d'eux. Les points sont regroupés par thème afin de faciliter la lecture : sécurité, architecture, logique métier, gestion des articles, interface utilisateur, partie technique (frontend) et qualité générale des données.

Au total, 37 points distincts ont été traités, couvrant à la fois des corrections indispensables au bon fonctionnement du système et des améliorations de confort d'utilisation.

2. Sécurité

Ces points concernaient la protection des données et l'accès à l'application. Ils étaient prioritaires car ils exposaient le système à un risque réel.

Point	Problème constaté	Solution apportée
Injection SQL	Les informations saisies par l'utilisateur étaient insérées directement dans les requêtes SQL, ce qui permettait à une personne mal intentionnée de modifier ou de détourner ces requêtes.	Sécurisation des requêtes à l'aide de requêtes préparées et de paramètres liés, empêchant toute donnée saisie d'altérer la requête.
Absence d'authentification	Les API pouvaient être appelées par n'importe qui, sans vérifier qu'un utilisateur était bien connecté.	Mise en place d'une vérification systématique (jeton de session) avant d'autoriser l'accès aux données.
Accès ouvert à tous (CORS)	N'importe quel site web pouvait interroger l'API du projet.	L'accès a été limité à l'application elle-même, les autres origines étant désormais refusées.

Point	Problème constaté	Solution apportée
Erreurs techniques visibles	En cas d'erreur, le serveur affichait des détails techniques bruts dans les réponses, ce qui pouvait exposer des informations sensibles.	L'affichage des erreurs techniques a été désactivé côté serveur et remplacé par des réponses propres.

3. Architecture et performance

Ces corrections visent à rendre l'application plus rapide, plus stable et plus facile à maintenir dans la durée.

Point	Problème constaté	Solution apportée
Lenteur des API	Certaines modifications de la base de données étaient relancées à chaque appel de l'API, ce qui ralentissait toute l'application.	Ces modifications sont désormais exécutées une seule fois, lors de l'installation, et non plus à chaque requête.
Adresse du serveur figée	L'adresse du serveur (« localhost ») était recopiée dans de nombreux fichiers, rendant toute modification future difficile et source d'erreurs.	L'adresse est désormais centralisée dans un seul fichier de configuration et réutilisée partout dans le projet.
Port de démarrage instable	Le port de l'application changeait à chaque redémarrage, obligeant à retrouver la bonne adresse à chaque fois.	Le port a été fixé définitivement, garantissant une adresse stable.
Perte de connexion après redémarrage	Lorsque la base de données redémarrait, les utilisateurs étaient déconnectés sans explication.	Un mécanisme de sauvegarde de session a été ajouté afin de restaurer automatiquement la connexion.

4. Logique métier (Ventes, Caisse, Ressources Humaines)

Ces points touchent directement aux calculs et aux règles de gestion de l'entreprise : stock, bénéfice, cautions et gestion du personnel.

Point	Problème constaté	Solution apportée
Stock non remis à jour	La suppression d'une vente n'entraînait pas la remise en stock des articles concernés.	Le stock est désormais recredité automatiquement dès qu'une vente est supprimée, avec conservation d'un historique.
Caution comptée comme un bénéfice	Une caution, pourtant remboursable, était comptabilisée comme un revenu de l'entreprise.	Les cautions sont désormais isolées et exclues du calcul du bénéfice tant qu'elles ne sont pas définitivement acquises.
Deux calculs de bénéfice différents	La Caisse et le Tableau de bord n'utilisaient pas la même méthode de calcul, ce qui donnait des résultats différents.	Un calcul unique et cohérent a été appliqué partout dans l'application.
Données orphelines après suppression d'un employé	La suppression d'un employé laissait dans la base ses présences, avances et paiements, sans lien avec personne.	La suppression d'un employé entraîne désormais la suppression propre de toutes les données qui lui sont liées.

Point	Problème constaté	Solution apportée
Erreur sur les prestataires sans chauffeur	L'absence de chauffeur assigné à un service provoquait une erreur technique.	Ce cas est désormais correctement pris en charge, sans générer d'erreur.

5. Gestion des articles

Le module Articles présentait plusieurs difficultés qui empêchaient une utilisation normale ; il a fait l'objet d'une attention particulière.

Point	Problème constaté	Solution apportée
Enregistrement impossible	La création d'un article ne fonctionnait pas correctement : les données n'étaient pas enregistrées dans la base.	Un circuit d'enregistrement fiable et direct a été mis en place pour garantir la sauvegarde des articles.
Article invisible après création	Même lorsqu'un article était bien enregistré, il n'apparaissait pas dans la liste sans rechargement manuel.	La liste des articles se met désormais à jour automatiquement après chaque création.
Informations mal transmises	Certains champs (comme le prix d'achat) portaient des noms différents entre l'interface et la base de données, provoquant des pertes d'information.	La correspondance entre les deux formats est désormais assurée automatiquement.
Catégories non filtrées	Le formulaire affichait toutes les catégories, même celles qui ne correspondaient pas à la famille de produit choisie.	Seules les catégories correspondant à la famille sélectionnée s'affichent désormais.
Fournisseurs absents du formulaire	La liste des fournisseurs ne s'affichait pas lors de la création d'un article.	La liste des fournisseurs est désormais chargée automatiquement à l'ouverture du formulaire.

6. Interface utilisateur

Ces points concernent l'affichage et le confort d'utilisation pour la personne qui utilise l'application au quotidien.

Point	Problème constaté	Solution apportée
Colonnes manquantes	Les pages Articles et Stock n'affichaient pas toutes les informations disponibles (référence, fournisseur, prix, dates, etc.).	L'ensemble des colonnes utiles a été ajouté, avec en complément des alertes de stock bas et la valeur totale du stock.
Calendrier figé sur une date fixe	Le calendrier des ressources humaines affichait toujours le même mois, quelle que soit la date réelle.	Le calendrier affiche désormais automatiquement le mois en cours.
Bouton inutilisable	Un bouton (« Bot ») est présent dans l'interface mais ne déclenche aucune action.	Fonctionnalité identifiée comme non prioritaire ; une décision reste à prendre (l'activer ou le retirer).
Texte de débogage visible	Un message technique destiné aux développeurs restait visible dans le formulaire article.	Ce message a été remplacé par un texte clair destiné à l'utilisateur final.

Point	Problème constaté	Solution apportée
Défaut d'affichage mineur	Une erreur d'écriture dans le style d'une fenêtre (RH) empêchait un alignement correct de s'appliquer.	Le style a été corrigé pour un affichage conforme.

7. Partie technique (frontend et gestion des données)

Ces corrections concernent le fonctionnement interne de l'application, invisible pour l'utilisateur mais essentiel à sa fiabilité.

Point	Problème constaté	Solution apportée
Contrôle qualité désactivé	Un contrôle interne, censé repérer les erreurs de manipulation des données en cours de développement, était désactivé.	Ce contrôle est désormais actif en développement et automatiquement désactivé en production, sans impact sur la performance finale.
Rafraîchissement des données mal placé	Le rafraîchissement automatique de la liste d'articles après une action ne fonctionnait pas de façon fiable.	La mise à jour est désormais déclenchée au bon endroit, garantissant un affichage toujours à jour.
Fenêtre qui se ferme trop tôt	La fenêtre de création d'article se fermait avant même d'avoir la confirmation que l'enregistrement avait réussi.	La fenêtre attend désormais la confirmation du serveur avant de se fermer.
Trop de données transmises à la connexion	Lors de la connexion, le serveur renvoyait l'ensemble des informations personnelles de l'utilisateur.	Seules les informations strictement nécessaires (identifiant, nom, rôle) sont désormais transmises.
Mauvaise mise à jour après connexion	Les informations de connexion et le jeton de session n'étaient pas toujours enregistrés correctement dans l'application.	L'enregistrement de la session a été corrigé afin de garantir une connexion stable.

8. Qualité des données et du code

Ces points, moins visibles au quotidien, contribuent à la solidité et à la pérennité du projet sur le long terme.

Point	Problème constaté	Solution apportée
Montants enregistrés en texte	Les prix et montants étaient stockés comme du simple texte plutôt que comme des nombres, ce qui expose à des erreurs de calcul.	Utilisation d'un type de donnée numérique adapté aux montants, garantissant des calculs fiables.
Formats de date incohérents	Les dates n'étaient pas toujours écrites de la même façon selon les parties du projet, ce qui compliquait les comparaisons et les tris.	Un format de date unique a été adopté sur l'ensemble du projet.
Absence de vérification des données	Il était possible d'enregistrer des champs vides ou des valeurs incohérentes (comme des montants négatifs).	Une vérification systématique des données est désormais effectuée avant tout enregistrement.

Point	Problème constaté	Solution apportée
Images stockées dans la base	Les images étaient enregistrées directement dans la base de données, ce qui la rendait très volumineuse et plus lente.	Les images sont désormais stockées dans un dossier dédié ; seule leur référence est conservée en base.
Absence de message en cas de panne	Lorsque le serveur était indisponible, l'application restait bloquée sans aucune explication pour l'utilisateur.	Un message d'erreur clair s'affiche désormais en cas d'indisponibilité du serveur.
Recherches lentes	Les recherches devenaient de plus en plus lentes à mesure que la base de données grandissait.	Des index ont été ajoutés sur les colonnes les plus consultées, accélérant nettement les recherches.
Éléments inutilisés	Une table de la base de données n'était jamais utilisée par l'application, et certains éléments de code n'étaient plus nécessaires.	Ces éléments ont été supprimés afin d'alléger et de clarifier le projet.

18. Conclusion

Le système développé répond aux besoins de gestion d'un magasin de vêtements de façon complète et adaptée au contexte tunisien : gestion du stock, des ventes, des services, de la trésorerie, des dépenses, des clients, des fournisseurs et des employés, avec calcul automatique des bénéfices et des salaires. L'ajout du script de démarrage automatique améliore par ailleurs le confort d'utilisation au quotidien.