

PROMPT COMPLET — APPLICATION WEB DE GESTION DE STOCK, CAISSE ET FACTURATION

Je veux développer une application web professionnelle, moderne, sécurisée et responsive permettant de gérer intégralement le **stock des marchandises, les entrées et sorties de stock, les ventes, la caisse, les factures, les utilisateurs et les rapports**.

L'application doit être conçue pour une entreprise, un magasin, une boutique ou un dépôt commercial.

1. OBJECTIF DE L'APPLICATION

Créer une plateforme centralisée permettant de :

- gérer les produits et marchandises ;
- gérer les catégories de produits ;
- enregistrer les entrées de marchandises ;
- enregistrer les sorties de marchandises ;
- suivre automatiquement les quantités disponibles ;
- gérer les ventes ;
- générer les factures ;
- gérer la caisse ;
- enregistrer les encaissements et décaissements ;
- suivre les mouvements financiers ;
- consulter l'état de la caisse ;
- consulter l'état du stock ;
- produire des rapports journaliers, mensuels et annuels ;
- gérer les utilisateurs et leurs permissions ;
- conserver l'historique de toutes les opérations.

L'application doit avoir une interface professionnelle, simple à utiliser et adaptée aux ordinateurs, tablettes et smartphones.

2. AUTHENTIFICATION

Créer un système sécurisé de connexion avec :

- nom d'utilisateur ou adresse e-mail ;
- mot de passe ;
- récupération du mot de passe ;
- changement du mot de passe ;
- déconnexion ;

- gestion des sessions ;
- protection des pages selon les permissions de l'utilisateur.

Après connexion, l'utilisateur doit être redirigé vers son tableau de bord selon son rôle.

3. GESTION DES UTILISATEURS

Créer un module **Utilisateurs** permettant à l'administrateur de :

- créer un utilisateur ;
- modifier un utilisateur ;
- désactiver/activer un utilisateur ;
- supprimer un utilisateur ;
- réinitialiser son mot de passe ;
- attribuer un rôle ;
- définir ses permissions ;
- consulter son historique d'activité.

Chaque opération importante doit enregistrer :

- utilisateur ;
- date ;
- heure ;
- opération effectuée ;
- module concerné ;
- ancienne valeur si nécessaire ;
- nouvelle valeur si nécessaire.

Rôles

Prévoir au minimum :

Administrateur système

Accès complet à l'application :

- utilisateurs ;
- rôles et permissions ;
- produits ;
- stock ;
- ventes ;
- caisse ;
- factures ;
- rapports ;
- paramètres ;
- journal d'audit.

Gestionnaire de stock

Accès principalement à :

- produits ;
- catégories ;
- fournisseurs ;
- entrées ;
- sorties ;
- inventaires ;
- mouvements de stock ;
- rapports de stock.

Caissier

Accès à :

- ventes ;
- encaissements ;
- caisse ;
- factures ;
- clôture de caisse ;
- rapports autorisés.

Vendeur

Accès à :

- clients ;
- ventes ;
- factures ;
- consultation des produits et stocks.

Comptable / Responsable financier

Accès à :

- caisse ;
- recettes ;
- dépenses ;
- encaissements ;
- décaissements ;
- rapports financiers ;
- factures.

Les rôles et permissions doivent être configurables par l'administrateur.

4. TABLEAU DE BORD

Créer un tableau de bord moderne affichant en temps réel :

Stock

- nombre de produits ;
- valeur totale du stock ;
- produits en rupture ;
- produits bientôt en rupture ;
- entrées du jour ;
- sorties du jour.

Ventes

- ventes du jour ;
- ventes du mois ;
- nombre de factures ;
- chiffre d'affaires.

Caisse

- solde actuel ;
- total des entrées ;
- total des sorties ;
- recettes du jour ;
- dépenses du jour.

Alertes

- produits en rupture ;
- stock faible ;
- factures impayées ;
- caisse à clôturer ;
- anomalies éventuelles.

Afficher également des graphiques :

- évolution des ventes ;
- évolution du stock ;
- entrées/sorties de marchandises ;
- évolution de la caisse.

5. GESTION DES PRODUITS

Créer un module de gestion des produits.

Chaque produit doit contenir :

- code produit ;
- référence/SKU ;

- nom du produit ;
- description ;
- catégorie ;
- unité de mesure ;
- prix d'achat ;
- prix de vente ;
- quantité initiale ;
- stock minimum ;
- stock maximum ;
- fournisseur ;
- emplacement ;
- date de création ;
- statut actif/inactif.

Prévoir également :

- recherche ;
- filtres ;
- tri ;
- pagination ;
- import Excel/CSV ;
- export Excel/CSV/PDF ;
- impression.

Le système doit empêcher les doublons de code produit.

6. CATÉGORIES

Créer un module permettant de :

- créer une catégorie ;
- modifier une catégorie ;
- supprimer une catégorie ;
- activer/désactiver une catégorie ;
- consulter le nombre de produits par catégorie.

7. FOURNISSEURS

Créer un module fournisseurs.

Informations :

- nom du fournisseur ;
- entreprise ;

- téléphone ;
- e-mail ;
- adresse ;
- personne de contact ;
- conditions de paiement ;
- notes.

Permettre de consulter l'historique des achats auprès de chaque fournisseur.

8. ENTRÉES DE STOCK

Créer un module **Entrées de marchandises**.

Une entrée doit contenir :

- numéro d'entrée ;
- date ;
- fournisseur ;
- référence du document ;
- produit ;
- quantité ;
- prix d'achat ;
- montant ;
- utilisateur ayant enregistré l'opération ;
- observation.

Lorsqu'une entrée est validée :

Stock disponible = Stock précédent + Quantité entrée

Le système doit automatiquement mettre à jour le stock.

Prévoir plusieurs motifs :

- achat fournisseur ;
- retour client ;
- transfert entrant ;
- correction de stock ;
- inventaire ;
- autre.

9. SORTIES DE STOCK

Créer un module **Sorties de marchandises**.

Une sortie doit contenir :

- numéro de sortie ;

- date ;
- produit ;
- quantité ;
- motif ;
- destination ;
- responsable ;
- observation.

Motifs possibles :

- vente ;
- perte ;
- casse ;
- consommation interne ;
- transfert ;
- retour fournisseur ;
- correction ;
- autre.

Lors de la validation :

Stock disponible = Stock précédent - Quantité sortie

Le système doit empêcher une sortie supérieure au stock disponible, sauf permission spéciale de l'administrateur.

10. MOUVEMENTS DE STOCK

Créer une page permettant de consulter tous les mouvements.

Afficher :

Date Produit Type Entrée Sortie Stock restant Utilisateur

Types :

- entrée ;
- sortie ;
- vente ;
- retour ;
- correction ;
- transfert ;
- inventaire.

Ajouter des filtres :

- période ;
- produit ;

- catégorie ;
- type de mouvement ;
- utilisateur.

11. INVENTAIRE

Créer un module d'inventaire permettant de :

- lancer un inventaire ;
- sélectionner les produits ;
- afficher le stock théorique ;
- saisir le stock physique ;
- calculer automatiquement les écarts ;
- justifier les écarts ;
- valider l'inventaire.

Exemple :

Stock théorique : 100

Stock physique : 96

Écart : -4

Après validation, le système doit créer automatiquement le mouvement de correction correspondant.

12. GESTION DES CLIENTS

Créer un module clients :

- nom ;
- prénom ;
- entreprise ;
- téléphone ;
- e-mail ;
- adresse ;
- numéro d'identification si nécessaire ;
- historique des achats ;
- total acheté ;
- solde éventuel.

13. GESTION DES VENTES

Créer une interface de vente rapide de type caisse/POS.

Le vendeur doit pouvoir :

1. rechercher un produit ;
2. sélectionner le produit ;
3. saisir la quantité ;
4. voir le prix ;
5. appliquer éventuellement une remise autorisée ;
6. ajouter plusieurs produits ;
7. sélectionner le client ;
8. choisir le mode de paiement ;
9. valider la vente ;
10. générer automatiquement la facture.

Le système doit automatiquement :

- diminuer le stock ;
- enregistrer la vente ;
- enregistrer l'encaissement ;
- mettre à jour la caisse ;
- générer la facture ;
- enregistrer l'opération dans l'historique.

14. FACTURATION

Créer un module de facturation.

Chaque facture doit avoir :

- numéro unique ;
- date ;
- client ;
- liste des produits ;
- quantités ;
- prix unitaires ;
- remises ;
- sous-total ;
- taxes si applicables ;
- total ;
- montant payé ;
- reste à payer ;
- mode de paiement ;
- utilisateur ;
- statut.

Statuts :

- brouillon ;
- validée ;
- payée ;
- partiellement payée ;
- impayée ;
- annulée.

Prévoir :

- aperçu de la facture ;
- impression ;
- téléchargement PDF ;
- partage/impression ;
- recherche ;
- annulation selon permissions.

15. MODES DE PAIEMENT

Prévoir :

- espèces ;
- mobile money ;
- virement bancaire ;
- carte bancaire ;
- chèque ;
- paiement à crédit ;
- autre.

Le système doit permettre de configurer les modes de paiement.

16. GESTION DE LA CAISSE

Créer un module complet de gestion de caisse.

Ouverture de caisse

Lors de l'ouverture :

- utilisateur ;
- date ;
- heure ;
- montant initial ;
- caisse concernée.

Entrées de caisse

Enregistrer :

- vente ;
- paiement client ;
- remboursement reçu ;
- autre recette.

Sorties de caisse

Enregistrer :

- dépense ;
- achat ;
- transport ;
- salaire ;
- remboursement ;
- retrait ;
- autre dépense.

Chaque mouvement doit contenir :

- numéro ;
- date ;
- heure ;
- type ;
- motif ;
- montant ;
- mode de paiement ;
- utilisateur ;
- observation ;
- pièce justificative si nécessaire.

17. ÉTAT DE LA CAISSE

Créer une page **État de caisse**.

Afficher :

Solde initial

- Total des entrées
- Total des sorties
- = **Solde théorique**

Afficher également :

- montant réellement compté ;

- différence ;
- justification de l'écart ;
- utilisateur ;
- date de clôture.

Exemple :

Solde initial : 100 USD

Entrées : 500 USD

Sorties : 150 USD

Solde théorique : 450 USD

Solde physique : 445 USD

Écart : -5 USD

18. CLÔTURE DE CAISSE

Permettre au caissier de clôturer sa caisse.

À la clôture :

- calculer automatiquement le solde théorique ;
- saisir le montant physique ;
- calculer l'écart ;
- demander une justification si écart ;
- enregistrer la clôture ;
- empêcher les modifications non autorisées après clôture.

L'administrateur doit pouvoir consulter les clôtures historiques.

19. RAPPORTS DE CAISSE

Créer des rapports :

Journalier

Afficher :

- solde initial ;
- total ventes ;
- total entrées ;
- total sorties ;
- solde final ;
- écarts ;
- nombre de transactions.

Mensuel

Afficher :

- total recettes ;

- total dépenses ;
- chiffre d'affaires ;
- solde ;
- évolution quotidienne ;
- répartition par mode de paiement.

Annuel

Afficher :

- recettes mensuelles ;
- dépenses mensuelles ;
- chiffre d'affaires mensuel ;
- solde mensuel ;
- total annuel.

Permettre l'export :

- PDF ;
- Excel ;
- CSV ;
- impression.

20. RAPPORTS DE STOCK

Créer les rapports suivants :

État actuel du stock

Produit Catégorie Stock Stock minimum Prix achat Valeur stock

Rapport journalier

- entrées ;
- sorties ;
- ventes ;
- produits concernés ;
- quantités ;
- valeur.

Rapport mensuel

- total des entrées ;
- total des sorties ;
- produits les plus vendus ;
- produits les moins vendus ;
- valeur du stock.

Rapport annuel

- évolution du stock ;
- achats ;
- ventes ;
- sorties ;
- valeur du stock.

21. RAPPORTS DES VENTES

Créer :

- ventes du jour ;
- ventes du mois ;
- ventes de l'année ;
- ventes par produit ;
- ventes par catégorie ;
- ventes par vendeur ;
- ventes par client ;
- ventes par mode de paiement ;
- produits les plus vendus ;
- chiffre d'affaires.

Permettre de sélectionner une période personnalisée.

22. JOURNAL D'AUDIT

Créer un journal complet des activités.

Enregistrer :

- connexion ;
- déconnexion ;
- création ;
- modification ;
- suppression ;
- validation ;
- annulation ;
- entrée de stock ;
- sortie de stock ;
- vente ;
- paiement ;
- mouvement de caisse ;
- clôture de caisse ;

- modification des paramètres.

Afficher :

- utilisateur ;
- action ;
- module ;
- date ;
- heure ;
- adresse IP si disponible ;
- détails de l'opération.

L'historique ne doit pas être modifiable par les utilisateurs ordinaires.

23. PARAMÈTRES DE L'APPLICATION

Créer une section **Paramètres** permettant à l'administrateur de configurer :

Entreprise

- nom ;
- logo ;
- adresse ;
- téléphone ;
- e-mail ;
- site web ;
- devise ;
- informations fiscales si nécessaires.

Stock

- stock minimum ;
- gestion des unités ;
- autorisation de stock négatif ;
- règles d'inventaire.

Vente

- remise maximale ;
- taxes ;
- numérotation des factures ;
- conditions de paiement.

Caisse

- devise ;
- modes de paiement ;
- règles d'ouverture ;

- règles de clôture.

Utilisateurs

- rôles ;
- permissions ;
- sécurité ;
- durée des sessions.

24. NOTIFICATIONS

Créer un système de notifications pour :

- stock faible ;
- rupture de stock ;
- nouvelle vente ;
- facture impayée ;
- caisse non clôturée ;
- écart de caisse ;
- inventaire terminé.

Afficher les notifications dans l'application.

25. RECHERCHE GLOBALE

Ajouter une recherche globale permettant de rechercher rapidement :

- produit ;
- facture ;
- client ;
- fournisseur ;
- vente ;
- mouvement de stock ;
- mouvement de caisse ;
- utilisateur.

26. INTERFACE UTILISATEUR

Créer une interface moderne et professionnelle.

Structure :

Sidebar

- Tableau de bord
- Produits

- Catégories
- Fournisseurs
- Clients
- Entrées stock
- Sorties stock
- Inventaire
- Mouvements stock
- Ventes
- Factures
- Caisse
- Rapports
- Utilisateurs
- Journal d'audit
- Paramètres

Barre supérieure

- recherche ;
- notifications ;
- utilisateur connecté ;
- profil ;
- déconnexion.

Utiliser des tableaux modernes avec :

- recherche ;
- filtres ;
- pagination ;
- tri ;
- export ;
- impression ;
- actions rapides.

L'application doit être responsive.

27. RÈGLES MÉTIER IMPORTANTES

Le système doit garantir la cohérence des données.

Stock

Une vente validée doit automatiquement créer une sortie de stock.

Une entrée validée doit augmenter le stock.

Une sortie validée doit diminuer le stock.

Une annulation doit générer une opération inverse plutôt que supprimer silencieusement l'historique.

Caisse

Une vente payée en espèces doit automatiquement générer une entrée de caisse.

Une dépense validée doit générer une sortie de caisse.

Le solde doit être calculé automatiquement.

Factures

Chaque facture doit avoir un numéro unique.

Une facture validée ne doit pas être supprimée définitivement.

Les annulations doivent être historisées.

Sécurité

Les utilisateurs ne doivent accéder qu'aux fonctionnalités autorisées par leurs permissions.

Les opérations sensibles doivent être protégées par des permissions supplémentaires.

28. BASE DE DONNÉES

Concevoir une base de données relationnelle propre.

Prévoir notamment les tables :

- users
- roles
- permissions
- role_permissions
- products
- categories
- suppliers
- customers
- stock_entries
- stock_entry_items
- stock_exits
- stock_exit_items
- stock_movements
- inventories
- inventory_items
- sales
- sale_items
- invoices
- invoice_items

- payments
- cash_registers
- cash_sessions
- cash_movements
- cash_closures
- audit_logs
- settings
- notifications

Utiliser des relations cohérentes, des clés primaires, des clés étrangères, des index et des contraintes d'intégrité.

29. SÉCURITÉ

Implémenter :

- authentification sécurisée ;
- mots de passe hashés ;
- contrôle d'accès basé sur les rôles ;
- permissions granulaires ;
- validation des données côté serveur ;
- protection contre les injections SQL ;
- protection CSRF si nécessaire ;
- protection XSS ;
- gestion sécurisée des sessions ;
- journalisation des actions sensibles.

30. PERFORMANCES

L'application doit être conçue pour supporter un grand nombre :

- de produits ;
- de ventes ;
- de factures ;
- de mouvements de stock ;
- de mouvements de caisse ;
- d'utilisateurs.

Utiliser :

- pagination ;
- indexation ;
- requêtes optimisées ;

- chargement différé lorsque nécessaire ;
- cache lorsque pertinent.

31. EXPORTS ET IMPRESSIONS

Tous les principaux rapports doivent pouvoir être :

- consultés à l'écran ;
- imprimés ;
- exportés en PDF ;
- exportés en Excel ;
- exportés en CSV.

Les factures PDF doivent avoir une présentation professionnelle avec :

- logo ;
- informations de l'entreprise ;
- informations du client ;
- numéro de facture ;
- date ;
- détails des produits ;
- sous-total ;
- taxes ;
- remise ;
- total ;
- montant payé ;
- reste à payer ;
- mode de paiement ;
- signature/cachet si nécessaire.

32. TECHNOLOGIES

Utiliser une architecture moderne.

Frontend :

- React / Next.js ;
- TypeScript ;
- Tailwind CSS ;
- composants UI modernes ;
- tableaux interactifs ;
- graphiques.

Backend :

- Node.js / NestJS ou autre framework moderne ;
- API REST ou architecture équivalente ;
- authentification sécurisée.

Base de données :

- PostgreSQL ou MySQL.

Utiliser une architecture propre séparant :

- présentation ;
- logique métier ;
- accès aux données ;
- authentification ;
- permissions.

33. TABLEAU DE BORD ADMINISTRATEUR

L'administrateur système doit avoir une vue globale :

Aujourd'hui

- chiffre d'affaires ;
- recettes ;
- dépenses ;
- solde caisse ;
- nombre de ventes ;
- nombre de factures ;
- entrées stock ;
- sorties stock ;
- produits en rupture.

Ce mois

- chiffre d'affaires ;
- dépenses ;
- bénéfice estimatif si les données de coût sont disponibles ;
- évolution des ventes ;
- évolution du stock.

Cette année

- chiffre d'affaires ;
- dépenses ;
- évolution mensuelle ;
- valeur du stock ;

- produits les plus vendus.

34. PRINCIPES D'ERGONOMIE

L'application doit être :

- simple ;
- rapide ;
- intuitive ;
- professionnelle ;
- responsive ;
- accessible aux utilisateurs non techniques.

Utiliser des boutons d'action clairement identifiables :

- Ajouter
- Modifier
- Voir
- Valider
- Annuler
- Imprimer
- Exporter
- Supprimer
- Rechercher
- Filtrer
- Clôturer

Afficher des messages de confirmation avant les opérations sensibles.

35. LIVRABLE FINAL

Je veux obtenir une application web fonctionnelle et prête à être déployée.

Elle doit comprendre :

1. Authentification
2. Gestion des utilisateurs
3. Rôles et permissions
4. Tableau de bord
5. Gestion des produits
6. Gestion des catégories
7. Gestion des fournisseurs
8. Gestion des clients
9. Entrées de stock

10. Sorties de stock
11. Mouvements de stock
12. Inventaire
13. Ventes
14. Facturation
15. Gestion de caisse
16. Ouverture de caisse
17. Clôture de caisse
18. Rapports de caisse
19. Rapports de stock
20. Rapports de ventes
21. Rapports journaliers
22. Rapports mensuels
23. Rapports annuels
24. Journal d'audit
25. Notifications
26. Paramètres
27. Export PDF/Excel/CSV
28. Impression des factures et rapports.

IMPORTANT

Ne créer pas uniquement des maquettes.

Je veux une **application réellement fonctionnelle**, avec :

- frontend fonctionnel ;
- backend fonctionnel ;
- base de données ;
- API ;
- authentification ;
- autorisation ;
- logique de stock ;
- logique de caisse ;
- logique de facturation ;
- rapports ;
- validation des données ;
- gestion des erreurs ;
- données de démonstration ;
- tests des principales fonctionnalités.

Commencer par définir l'architecture complète du projet, le schéma de base de données et les relations entre les modules, puis développer progressivement chaque module sans casser les fonctionnalités précédentes.

Toutes les opérations financières et de stock doivent être **traçables, cohérentes et historisées**.