logo

Master 1ère année, promo 2024 – 2025, semestre 1

République De Côte d’Ivoire

Union - Discipline - Travail

Une image contenant clipart, dessin, dessin humoristique, croquis

Le contenu généré par l’IA peut être incorrect.

**Etudiant :**

Yao konan franck schalom

**Enseignant :**

Dr Mamadou BAKOUAN

**Sujet : Réalisez une étude de cas complète pour la conception et l'implémentation d'un système de gestion des stocks.**

**Cas d’une brasserie**

**TABLE DES MATIERES.**

[INTRODUCTION GENERALE. 4](#_Toc192974600)

[ARCHITECTURE METIER 4](#_Toc192974601)

[1. Gestion des entrées et sorties de stock 4](#_Toc192974602)

[2. Suivi des commandes 5](#_Toc192974603)

[3. Gestion des fournisseurs et des clients 5](#_Toc192974604)

[4. Génération de rapports analytiques 5](#_Toc192974605)

[5. Tableau des Cas d’Utilisation 5](#_Toc192974606)

[ARCHITECTURE APPLICATIVE 7](#_Toc192974607)

[1. Module de gestion des stocks 7](#_Toc192974608)

[2. Module de gestion des commandes 7](#_Toc192974609)

[3. Module de gestion des fournisseurs et clients 7](#_Toc192974610)

[4. Module de reporting et d’analyse 7](#_Toc192974611)

[ARCHITECTURE DE DONNEES 8](#_Toc192974612)

[I. Classes Principales de la base de données 8](#_Toc192974613)

[II. Modèle Relationnel pour la Base de Données 9](#_Toc192974614)

[1. Table PRODUIT 9](#_Toc192974615)

[2. Table FOURNISSEUR 9](#_Toc192974616)

[3. Table COMMANDE\_FOURNISSEUR 10](#_Toc192974617)

[4. Table COMMANDE\_CLIENT 10](#_Toc192974618)

[5. Sécurité des données 10](#_Toc192974619)

[III. Schéma relationnel 11](#_Toc192974620)

[ARCHITECTURE TECHNIQUE 11](#_Toc192974621)

[I. Infrastructure matérielle 11](#_Toc192974622)

[II. Réseau et Communication 12](#_Toc192974623)

[III. Outils Logiciels et Technologies 13](#_Toc192974624)

[ARCHITECTURE DE SÉCURITÉ 13](#_Toc192974625)

[I. Gestion des accès et authentification 13](#_Toc192974626)

[II. Protection des données 14](#_Toc192974627)

[III. Sécurité réseau et prévention des attaques 14](#_Toc192974628)

[IV. Surveillance et réaction aux incidents 15](#_Toc192974629)

[CONCLUSION. 15](#_Toc192974630)

[BIBLIOGRAPHIE 16](#_Toc192974631)

# INTRODUCTION GENERALE.

La gestion des stocks est un élément clé pour assurer l’efficacité opérationnelle d’une brasserie. Dans un secteur où la demande peut fluctuer et où la gestion des matières premières est cruciale, un suivi rigoureux des stocks devient impératif.

Ce projet vise ainsi à concevoir et à développer un système de gestion des stocks dédié à une brasserie. L’objectif principal est de mettre en place une solution permettant de suivre en temps réel les niveaux de stock des boissons, d’optimiser les approvisionnements et de fluidifier les processus liés à la distribution.

Grâce à ce système, la brasserie pourra anticiper les ruptures de stock, éviter le surstockage et rationaliser la gestion des commandes.

Les fonctionnalités incluront la traçabilité des mouvements de stock, la gestion des fournisseurs et clients, ainsi que la production de rapports analytiques pour faciliter la prise de décisions stratégiques.

Ce projet repose donc sur une approche intégrée combinant une modélisation des processus métiers et une structuration des données.

# ARCHITECTURE METIER

Afin de garantir la pertinence du système, une analyse détaillée des besoins a été réalisée. Cette étude a permis d’identifier les principales fonctionnalités devant être intégrées dans le système afin de répondre aux défis liés à la gestion des stocks dans une brasserie.

Parmi ces fonctionnalités essentielles, on retrouve :

### Gestion des entrées et sorties de stock

Chaque entrée (réception de matières premières) et sortie (vente de produits finis, utilisation pour la production) doit être enregistrée en temps réel afin d’assurer une traçabilité optimale.

### Suivi des commandes

L’automatisation du suivi des commandes clients et fournisseurs permet d’éviter les erreurs et d’optimiser les délais de livraison et d’approvisionnement.

### Gestion des fournisseurs et des clients

Le système intègre une base de données permettant de centraliser les informations sur les fournisseurs et clients, facilitant ainsi la communication et la gestion des transactions commerciales.

### Génération de rapports analytiques

Pour une meilleure prise de décision, des rapports détaillés sur les niveaux de stock, la fréquence des commandes et les performances des ventes seront générés, offrant ainsi une vue d’ensemble sur les tendances et les besoins de réapprovisionnement.

La mise en place de ces fonctionnalités garantira une gestion centralisée et optimisée des ressources de la brasserie, en assurant un équilibre entre l’offre et la demande tout en réduisant les pertes et en maximisant l’efficacité des opérations.

### Tableau des Cas d’Utilisation

|  |  |  |  |
| --- | --- | --- | --- |
| **Cas d’Utilisation** | **Acteurs** | **Pré-conditions** | **Post-conditions** |
| Gérer les matières premières | Gestionnaire de stock, Responsable des achats | Le produit doit être référencé dans la base | Produit mis à jour ou supprimé |
| Passer une commande fournisseur | Responsable des achats, Fournisseur | Le stock est inférieur au seuil défini | Commande créée et en attente |
| Réception des commandes fournisseurs | Magasinier, Gestionnaire de stock | Une commande fournisseur existe | Stock mis à jour après réception |
| Gérer les produits finis | Gestionnaire de stock | Produit disponible en quantité suffisante | Stock mis à jour après fabrication |
| Enregistrer une vente | Responsable des ventes, Client | Produit en stock et commande validée | Stock mis à jour et facture générée |
| Suivre les mouvements de stock | Magasinier, Gestionnaire de stock | Une action d’entrée ou sortie a eu lieu | Historique des transactions mis à jour |
| Générer des rapports | Gestionnaire de stock | Données de stock et de vente disponibles | Rapport généré et affiché |

# ARCHITECTURE APPLICATIVE

Afin d’assurer une gestion efficace des stocks au sein de la brasserie, une architecture applicative bien structurée a été mise en place. Elle repose sur différents modules interconnectés qui assurent le bon fonctionnement du système en intégrant les services métiers essentiels.

Parmi ces composants applicatifs clés, on retrouve :

### Module de gestion des stocks

Ce module est au cœur du système et permet de suivre en temps réel les mouvements des stocks. Il assure l’enregistrement des entrées et sorties de stock pour les matières premières et les produits finis, la mise à jour automatique des quantités disponibles après chaque transaction, l’activation d’alertes en cas de stock insuffisant garantissant une réactivité optimale dans le réapprovisionnement.

### Module de gestion des commandes

Ce module facilite la gestion des commandes fournisseurs et clients, réduisant les risques d’erreurs et optimisant le traitement des transactions. Il inclut la création et validation des commandes fournisseurs et clients, le suivi des statuts de commande (En attente, Expédiée, Livrée, Annulée), l’intégration avec le module de gestion des stocks garantissant une mise à jour automatique des quantités disponibles après chaque commande.

### Module de gestion des fournisseurs et clients

Pour assurer une traçabilité complète des échanges commerciaux, ce module regroupe toutes les informations relatives aux fournisseurs et clients de la brasserie. Ses fonctionnalités incluent l’enregistrement et la mise à jour des informations de chaque fournisseur et client, l’historique des transactions avec chaque partenaire commercial, l’optimisation des délais d’approvisionnement et de livraison en fonction des habitudes de commande.

### Module de reporting et d’analyse

Un système performant doit permettre une prise de décision rapide et éclairée. C’est pourquoi un module de reporting a été intégré pour générer des rapports détaillés sur l’état des stocks, les performances de vente et l’évolution des commandes. Il permet de consulter des tableaux de bord dynamiques avec des indicateurs clés, d’analyser les tendances des ventes et des consommations de matières premières, d’exporter des rapports personnalisés facilitant la planification stratégique.

# ARCHITECTURE DE DONNEES

L’architecture des données concerne la structuration, le stockage et la gestion des informations dans la base de données du système.

Le modèle relationnel sera conçu de manière à garantir la cohérence, la sécurité et l’accessibilité des données. Les données principales incluent les produits (boissons, emballages), les mouvements de stock (entrées, sorties, transferts), les fournisseurs et les clients (coordonnées, historique des transactions), les commandes (statut, historique, facturation).

L’objectif est de structurer ces données de façon efficace et optimisée, en minimisant les redondances et en assurant une récupération rapide des informations critiques.

## Classes Principales de la base de données

|  |  |  |
| --- | --- | --- |
| **Classe** | **Attributs** | **Relations** |
| Produit | id, nom, type, stock, seuil\_min | Lié à CommandeFournisseur, CommandeClient |
| CommandeFournisseur | id, date, statut, fournisseur\_id | Liée à Fournisseur, Produit |
| CommandeClient | id, date, statut, client\_id | Liée à Client, Produit |
| Fournisseur | id, nom, contact, adresse | Lié à CommandeFournisseur |
| Client | id, nom, contact, adresse | Lié à CommandeClient |
| MouvementStock | id, type, quantité, date, produit\_id | Lié à Produit |

## Modèle Relationnel pour la Base de Données

La base de données repose sur un modèle relationnel structuré autour de plusieurs tables interconnectées.

### Table PRODUIT

|  |  |  |
| --- | --- | --- |
| **Champ** | **Type** | **Contraintes** |
| id | INT | PRIMARY KEY, AUTO\_INCREMENT |
| nom | VARCHAR(255) | NOT NULL |
| type | ENUM('Matière Première', 'Produit Fini') | NOT NULL |
| stock | INT | NOT NULL |
| seuil\_min | INT | NOT NULL |

### Table FOURNISSEUR

|  |  |  |
| --- | --- | --- |
| **Champ** | **Type** | **Contraintes** |
| id | INT | PRIMARY KEY, AUTO\_INCREMENT |
| nom | VARCHAR(255) | NOT NULL |
| contact | VARCHAR(100) | NOT NULL |

### Table COMMANDE\_FOURNISSEUR

|  |  |  |
| --- | --- | --- |
| **Champ** | **Type** | **Contraintes** |
| id | INT | PRIMARY KEY, AUTO\_INCREMENT |
| date | DATE | NOT NULL |
| statut | ENUM('En attente', 'Livrée') | NOT NULL |
| fournisseur\_id | INT | FOREIGN KEY REFERENCES FOURNISSEUR(id) |

### Table COMMANDE\_CLIENT

|  |  |  |
| --- | --- | --- |
| **Champ** | **Type** | **Contraintes** |
| id | INT | PRIMARY KEY, AUTO\_INCREMENT |
| date | DATE | NOT NULL |
| statut | ENUM('En attente', 'Expédiée') | NOT NULL |
| client\_id | INT | FOREIGN KEY REFERENCES CLIENT(id) |

### Sécurité des données

Le système applique des restrictions d’accès basées sur les rôles, empêchant tout utilisateur non autorisé d’accéder ou de modifier certaines données sensibles. Il assure le chiffrement des données sensibles, notamment les mots de passe et transactions critiques, garantit l’intégrité des informations grâce aux contraintes de clé primaire et étrangère, met en place des sauvegardes automatiques pour assurer la récupération des données en cas d’incident.

## Schéma relationnel

Une image contenant texte, capture d’écran, diagramme, nombre

Le contenu généré par l’IA peut être incorrect.

# ARCHITECTURE TECHNIQUE

L’architecture technique définit l’infrastructure matérielle et logicielle utilisée pour assurer le bon fonctionnement du système de gestion des stocks. Elle inclut les serveurs, les réseaux et les technologies utilisées pour garantir la disponibilité, la rapidité et la sécurité du système. L’objectif est de mettre en place une infrastructure robuste, évolutive et sécurisée, permettant une gestion fluide des stocks et des transactions.

## ****Infrastructure matérielle****

|  |  |  |
| --- | --- | --- |
| **Composant** | **Spécifications** | **Rôle** |
| **Serveur de base de données** | MySQL/PostgreSQL sur un serveur Linux dédié ou cloud | Stockage et gestion des données du système |
| **Serveur d’application** | Apache/Nginx avec PHP (Laravel), Django (Python) ou Node.js | Hébergement des services applicatifs |
| **Serveur de sauvegarde** | Serveur distant avec backups automatisés | Stockage sécurisé des copies de la base de données |
| **Infrastructure réseau** | VPN, Pare-feu, Détection des intrusions | Protection contre les cyberattaques et contrôle des accès |
| **Stations de travail** | Ordinateurs connectés en réseau local (LAN) | Accès aux fonctionnalités du système pour les employés |

L’infrastructure matérielle repose sur des serveurs performants et des équipements sécurisés afin de garantir une disponibilité optimale du système et une gestion efficace des transactions en temps réel.

## ****Réseau et Communication****

Le réseau est conçu pour assurer une communication fluide entre les utilisateurs et les serveurs, tout en garantissant une connexion sécurisée. Il repose sur plusieurs technologies et protocoles visant à protéger les échanges de données.

|  |  |  |
| --- | --- | --- |
| **Technologie** | **Description** | **Utilisation** |
| **Réseau local (LAN/Wi-Fi sécurisé)** | Connexion filaire et sans fil avec chiffrement WPA3 | Accès au système dans les bureaux et entrepôts |
| **VPN (Virtual Private Network)** | Tunnel sécurisé pour les connexions distantes | Permet aux employés d’accéder au système en toute sécurité |
| **Pare-feu et filtrage IP** | Protection contre les intrusions | Sécurise les accès aux services applicatifs |
| **Chiffrement SSL/TLS** | Sécurisation des échanges de données | Protège les requêtes entre le client et le serveur |
| **Détection des intrusions (IDS/IPS)** | Surveillance des tentatives de cyberattaques | Protège l’intégrité du système et empêche les accès non autorisés |

L’architecture réseau garantit une connectivité fiable et sécurisée, réduisant les risques de cyberattaques et assurant une transmission fluide des informations essentielles au fonctionnement du système.

## ****Outils Logiciels et Technologies****

L’architecture technique repose sur des outils modernes et des frameworks performants permettant une **interopérabilité fluide et une gestion efficace des stocks.**

|  |  |  |
| --- | --- | --- |
| **Composant** | **Technologie utilisée** | **Rôle** |
| **Backend** | Laravel (PHP), Django (Python), Express.js (Node.js) | Gestion des transactions et logique métier |
| **Frontend** | Vue.js, React.js, Angular.js | Interface utilisateur et affichage des données |
| **Base de données** | MySQL, PostgreSQL | Stockage et gestion des informations critiques |
| **Système de gestion des logs** | Logstash, Kibana | Surveillance et analyse des événements système |
| **Gestion des versions** | GitHub, GitLab | Suivi des évolutions du code et collaboration |

Grâce à ces outils, l’architecture technique assure **une modularité optimale**, facilitant les mises à jour et garantissant une évolutivité sans interruption du service.

# ****ARCHITECTURE DE SÉCURITÉ****

L’architecture de sécurité définit les mécanismes mis en place pour protéger les données, les accès et les transactions au sein du système de gestion des stocks. Elle garantit la confidentialité, l’intégrité et la disponibilité des informations en mettant en œuvre des protocoles de sécurité avancés, des politiques d’accès strictes et des outils de surveillance.

## ****Gestion des accès et authentification****

|  |  |  |
| --- | --- | --- |
| **Mécanisme** | **Description** | **Objectif** |
| **Authentification multi-facteurs (MFA)** | Validation par mot de passe + code SMS/email | Sécuriser l’accès au système |
| **Rôles et permissions** | Définition des niveaux d’accès pour chaque utilisateur | Restreindre l’accès aux fonctionnalités selon le profil |
| **Déconnexion automatique** | Fermeture de session après une période d’inactivité | Empêcher les accès non autorisés en cas d’oubli de déconnexion |
| **Chiffrement des mots de passe** | Hashage avec bcrypt ou Argon2 | Empêcher le vol d’identifiants en cas de fuite de données |
| **Journalisation des connexions** | Enregistrement des tentatives de connexion et détections d’anomalies | Suivi des accès et prévention des intrusions |

La mise en place de ces contrôles d’accès **garantit une authentification sécurisée et une gestion stricte des permissions**, empêchant toute tentative d’intrusion ou d’accès non autorisé.

## ****Protection des données****

|  |  |  |
| --- | --- | --- |
| **Mécanisme** | **Description** | **Objectif** |
| **Chiffrement AES-256** | Sécurisation des données sensibles (transactions, informations clients) | Empêcher l’interception et la manipulation des données |
| **Sauvegardes automatiques** | Stockage des backups sur serveur distant sécurisé | Assurer la récupération des données en cas d’incident |
| **Contrôle d’intégrité des données** | Vérification des modifications et des transactions suspectes | Empêcher toute altération frauduleuse des informations |
| **Droits d’accès basés sur les rôles (RBAC)** | Définition de permissions spécifiques selon les utilisateurs | Empêcher toute modification non autorisée des données critiques |
| **Protection contre les pertes de données (DLP)** | Mise en place de stratégies pour empêcher l’exportation non autorisée des informations | Prévenir les fuites de données confidentielles |

Ces mesures assurent une **protection robuste des informations sensibles**, garantissant leur **confidentialité et leur intégrité** face aux menaces internes et externes.

## ****Sécurité réseau et prévention des attaques****

|  |  |  |
| --- | --- | --- |
| **Mécanisme** | **Description** | **Objectif** |
| **Pare-feu et filtrage IP** | Blocage des adresses suspectes et limitation des accès selon la localisation | Empêcher les attaques réseau et tentatives d’intrusion |
| **Protection contre les injections SQL** | Validation stricte des entrées utilisateur et utilisation d’ORM sécurisé | Éviter l’exploitation de failles dans la base de données |
| **Prévention des attaques XSS et CSRF** | Activation des Content Security Policies (CSP) et jetons CSRF | Protéger l’interface web contre les manipulations malveillantes |
| **Chiffrement SSL/TLS** | Protection des échanges entre le client et le serveur | Empêcher l’interception des données lors des communications |
| **Surveillance et détection des menaces** | Analyse en temps réel des comportements anormaux | Identifier et bloquer les activités suspectes avant qu’elles n’affectent le système |

Ces mécanismes assurent **une protection avancée du réseau et des services**, minimisant les risques liés aux cyberattaques et garantissant la **disponibilité permanente du système**.

## ****Surveillance et réaction aux incidents****

|  |  |  |
| --- | --- | --- |
| **Mécanisme** | **Description** | **Objectif** |
| **Système d’alerte et de notifications** | Envoi d’alertes en cas de connexion suspecte ou d’anomalie détectée | Réagir rapidement aux menaces potentielles |
| **Tests de sécurité réguliers** | Audit de vulnérabilité, pentests et analyses OWASP | Identifier et corriger les failles avant qu’elles ne soient exploitées |
| **Journal d’audit** | Enregistrement des actions critiques des utilisateurs | Assurer une traçabilité des opérations sensibles |
| **Mécanisme de récupération après incident (Disaster Recovery Plan - DRP)** | Procédures en cas de panne, cyberattaque ou perte de données | Assurer la continuité des opérations sans interruption |

Avec ces mesures, l’architecture de sécurité garantit **une surveillance proactive, une réponse rapide aux incidents et une résilience du système face aux menaces.**

# CONCLUSION.

Ce projet a démontré l’importance d’une gestion optimisée des stocks dans une brasserie, en réduisant les erreurs humaines, en améliorant la traçabilité et en facilitant la prise de décision. Grâce à l’utilisation d’outils modernes et à une approche méthodologique, ce système offre une solution fiable et évolutive, adaptée aux besoins des entreprises du secteur brassicole. L’implémentation de ce système permettra à la brasserie d’améliorer sa productivité, de minimiser les pertes et d’optimiser la gestion de ses ressources.

# BIBLIOGRAPHIE

SAATSA, TEZO JORIAN. *SOUTENANCE DE PROJET DE FIN D’ANNEE*. INSTITUTE HIGHER AZIMUT, 24 juillet 2024, <https://www.slideshare.net/slideshow/conception-et-developpement-d-une-application-de-gestion-de-stock-saatsa-tezo-jorian-pdf/270479452>.

https://fr.wikipedia.org/w/index.php?title=Structured\_Query\_Language&action=history. « Structured Query Language ». *Wikipedia*, <https://fr.wikipedia.org/w/index.php?title=Structured_Query_Language&action=history>