

ETUDE ET MISE EN PLACE  
D’UNE APPLICATION DE  
DE GESTION DE STOCK  
Gestion de Stock  
Soufiane Hammagi & Abdelilah Lahdili

Conception BD | 17/02/2022

INTRODUCTION:

Chaque entreprise quel que soit son domaine d’activité, ne peut pas avoir des produits ou/et des services comme flux de sortie de son processus de gestion sans avoir un système d’information qui est l'ensemble des actions coordonnées de recherche, de traitement, de distribution et protection des informations utiles. A la base de toutes les décisions, il met le système informatique au service du contenu informationnel. Ce système informatique présente un ensemble de moyens informatiques et de télécommunications ayant pour finalité d'élaborer, traiter, stocker, acheminer, présenter ou détruire des données.

A l'heure actuelle, l'informatique a pris une place prépondérante dans la gestion des organisations. L'équipement informatique des entreprises assurant la gestion des informations, Notre société dédié à la conception d'ouvrages structurels et constructions des bâtiments et la fourniture, et à la mise à disposition de matériaux de construction à un groupe d'autres entreprises dans le même domaine d’une manière traditionnelle. En plus un service de gestion des produits et commande des clients Pour continuer à travailler sur leurs projets.

Le système d’information actuel de plusieurs entrepôt est dite traditionnel et manuel, ça veut dire que les employés de réception travaillent avec les documents et les formulaires papier, l’information circule oralement…des mauvaises pratiques (et Autres), cette mal gestion participe à cumuler plusieurs problèmes au niveau managérial, production de services, déroulement des diagnostics et des réparations , absence des indicateurs de performances/traçabilité ainsi des problèmes rencontrés en qualité des services …

Dans ce cadre, il nous a été proposé la création et la réalisation de l’application Bureau « UNIStock » qui permet la gestion de tous les services de société : service de gestion des clients, les produits disponibles dans l’entrepôt de société et l’organisation des catégories des produits, cela permet aux clients d'effectuer des Commandes qui serait ensuite traitées par le responsable de réception (administrateur de l'application).

Cadrage du projet:

Description du projet:

La fonction première de notre application c’est la gestion des produits, les clients et aussi les commandes effectuées par ces clients ; soit commandes terminées ou bien commandes entrain de révision.

Cependant, pour mettre éventuellement cette application au service de la société et pour la rendre beaucoup plus intégrée, complète, on doit ajouter d’autres fonctions (gestion des catégories de produit).

***Pour l’administrateur :***

Il a le droit de modifier la base de données. En effet, il peut ajouter, supprimer ou modifier un client, un produit, une commande, et les catégories des produits.

Objectifs:

Les clients sont la raison d'être et le capital de toute société, c'est pour cela pour en conquérir de nouveau et surtout conserver les anciens, il faut mettre en place un système informatisé plus précisément un module de gestion qui devra résoudre les problèmes rencontrés dans la gestion manuelle des réceptions et prendre en compte les perspectives d'évolution et les besoins des clients. Pour ce faire, notre travail consistera à mettre en place un système dont les fonctionnalités offriront :

**-**Répondre aux attentes des employés, par la création d’une application bureau officiel qui facilite le travail sur eux.

**-**Satisfaction des clients de la société et les employés de la réception, c’est-à-dire qu’il aura plus de perte de temps pour effectuer des Commandes au niveau de la société.

Fonctionnalités du système:

Les fonctionnalités du système représentent les actions que le système doit exécuter, il ne devient opérationnel que s’il les satisfait.

L’application offre plusieurs fonctionnalités ou bien services pour leurs utilisateurs autant qu’administrateur, ici on présente l’ensemble de ces fonctionnalités.

***Les besoins fonctionnels*** ***:***

Notre application doit couvrir principalement les besoins fonctionnels suivants :

* Permettre à l’administrateur de gérer toutes les catégories d’entrepôt

(Ajouter, supprimer ou modifier une commande d’un client, catégorie de produit et aussi la possibilité d’ajouter, supprimer ou modifier un client).

***Les besoins non fonctionnels*** ***:***

Ce sont des exigences qui ne concernent pas spécifiquement le comportement du système mais plutôt identifient des contraintes internes et externes du système.

Les principaux besoins non fonctionnels de notre application se résument dans les points suivants :

* Le code doit être clair pour permettre des futures évolutions ou améliorations.
* L’ergonomie : L’application offre une interface conviviale et facile à utiliser.
* Garantir l’intégrité et la cohérence des données à chaque mise à jour et à chaque insertion.

Identification des acteurs :

Après les captures des fonctionnalités système, nous avons recensé la liste des acteurs suivants présentés dans le tableau ci-dessus qui résument les différents rôles logiques du système :

|  |  |
| --- | --- |
| Rôle | Description |
| Administrateur | Responsable principale du système, prend en charge la gestion des clients, produits, catégories des produits et des Commandes. |

Tableau 2 : Description des acteurs

La planification opérationnelle :

Définition:

La planification opérationnelle est l’art de planifier un projet pour le rendre pilotable.  
La planification opérationnelle guide le chef de projet dans le choix :

* De la structure du plan
* Du niveau de détail
* Des modes de délégation
* Du management des coûts
* Des modes de pilotages possibles
* Des cycles de pilotage
* De la communication

Elle prend en compte les 3 dimensions du projet :

* La durée
* Les ressources et moyens (humains et matériels)
* Le périmètre

La planification opérationnelle définit clairement la mise en œuvre du plan stratégique pour atteindre certains objectifs spécifiques.

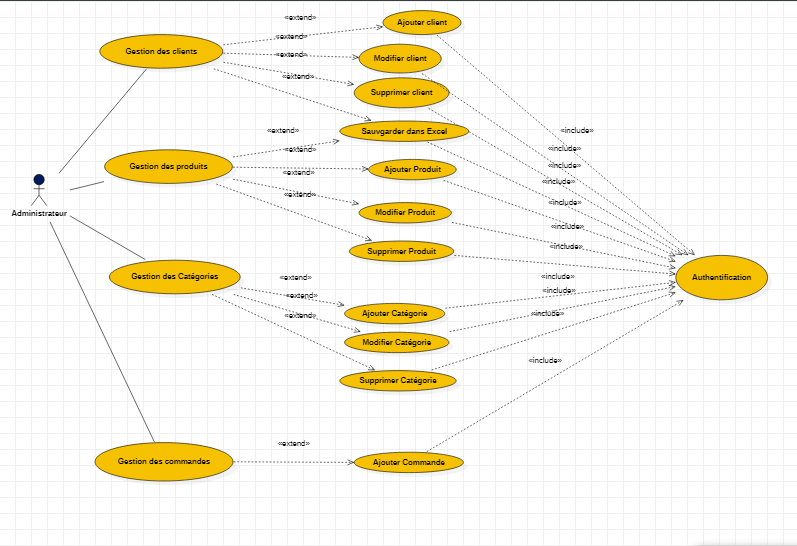
Modélisation d’application:

Introduction :

Après avoir traité les besoins que le projet doit accomplir, nous dédions ce chapitre à la phase de conception. Nous commençons tout d’abord par définir le langage utilisé pour la conception de la solution, ensuite nous présentons les différents diagrammes UML (cas d’utilisation, séquence, et de classe).

# Diagramme de cas d’utilisation :

## Le diagramme de cas d’utilisation à l’aide de StarUML :

*Figure 5 : Diagramme de cas d’utilisation*

Explication :

* Use case : Gestion des clients:
* **Titre :** Gérer les clients
* **Acteur :** Administrateur
* **Description :** L’administrateur a le droit d’ajouter, modifier, et supprimer les clients de l’entreprise.
* **Pré conditions :** -L’acteur est connecté.

- L’acteur choisit l’opération clients.

* Use case : Gestion des produits :
* **Titre :** Gérer les produits
* **Acteur :** Administrateur
* **Description :** L’administrateur a le droit d’ajouter, modifier, et supprimer les produits de l’entreprise.
* **Pré conditions :** -L’acteur est connecté.

-L’acteur choisit la catégorie du produit.

* Use case : Gestion des catégories :
* **Titre :** Gérer les catégories
* **Acteur :** Administrateur
* **Description :** L’administrateur a le droit d’ajouter, modifier, et supprimer les catégories des produits de l’entreprise.
* **Pré conditions :** -L’acteur est connecté.

-L’acteur choisit la catégorie du produit.

* Use case : Gestion des commandes :
* **Titre :** Consulter le Voitures
* **Acteur :** Visiteur
* **Description :** L’administrateur a le droit d’ajouter les commandes réalisées par un client de l’entreprise.
* **Pré conditions :**  -L’acteur est connecté.

-L’acteur choisit les produits et le client.

* Use case : Se connecter :
* **Titre :** Se connecter
* **Acteur :** Administrateur
* **Description :** L’authentification consiste à vérifier les droits d’un utilisateur d’accéder à l’application.
* **Pré conditions :** Inscription.

# Diagramme de séquence :

## Le diagramme de séquence : (Connexion)

Figure 6 : Diagramme de séquence(Connexion)

*Explication :*

Ce diagramme représente le scénario nominal de connexion. Cette étape permet aux administrateurs d’accéder à leurs espaces. Les informations saisies pour remplir le formulaire de la connexion sont : « Nom » et « password ».

* **Action déclencheur** : L’utilisateur de système souhaite se connecter.
* **Scénario nominal :**

1. Le système demande l’identification de l’acteur.

2. L’acteur saisie et valide son Nom et son mot de passe.

3. Le système vérifie l’inscription de l’acteur.

4. Le système vérifie la validité de Nom et du mot de passe.

5. Le système offre la connexion à l’acteur.

* **Action de fin :** L’acteur accède à son compte.
* **Post-conditions :** L’acteur peut accéder aux fonctionnalités actives pour lui.
* **Exception :**
  1. Le nom ou/et le mot de passe sont erronés (Le système indique que l’identification est erronée).
  2. L’utilisateur n’a pas un compte (Le système indique que l’identification est erronée).

## Le diagramme de séquence : (Ajouter un client)

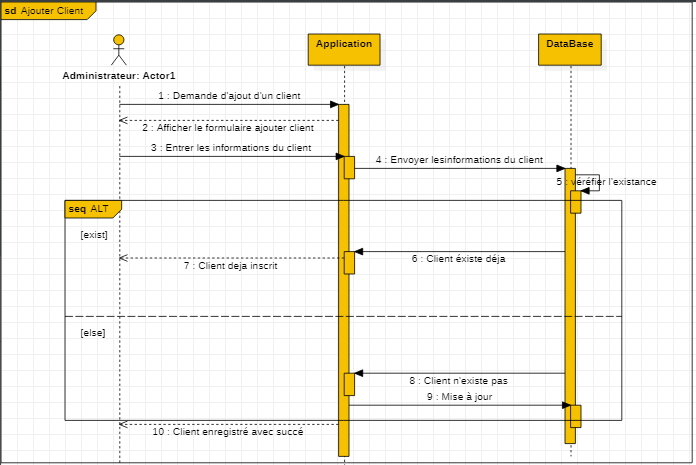


Figure 7 : Diagramme de séquence (Ajouter client)

**Explication :**

Ce diagramme représente le processus d’effectuer un ajout de client :

* **Scénario nominal :**

1. Le système demande l’identification d’acteur.

2 L’acteur saisie et valide son Nom et son mot de passe.

3. Le système offre la connexion au client.

4. L’acteur accède à l’interface d’ajout du client.

5. L’acteur passe les informations du client .

6. Le système vérifie si ce client existe déjà avant de créer un nouveau client .

* **Action de fin :** L’acteur effectue les opérations CRUD possibles sur un client.

## Le diagramme de séquence : (Ajouter Produit)

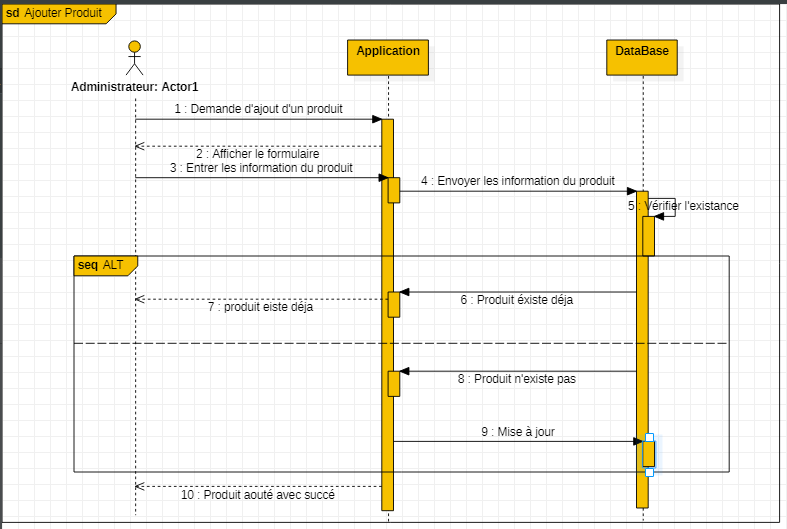


Figure 8 : Diagramme de séquence (Ajouter produit)

Explication :

Ce diagramme représente le processus d’effectuer un ajout d’un produit :

* **Scénario nominal :**

1. Le système demande l’identification d’acteur
2. L’acteur saisie et valide son Nom et son mot de passe.
3. Le système offre la connexion au client.
4. L’acteur accède à l’interface d’ajout du produit.
5. L’acteur passe les informations du produit .
6. Le système vérifie si ce produit existe déjà avant de créer un nouveau produit .

* **Action de fin :** L’acteur effectue les opérations CRUD possibles sur un produit.

## Le diagramme de séquence : (Ajouter Produit)

Figure 9 : Diagramme de séquence (Ajouter commande)

Explication :

Ce diagramme représente le processus d’ajout d’une commande

* **Scénario nominal :**

1. Le système demande l’identification d’acteur.

2 L’acteur saisie et valide son Nom et son mot de passe.

3. Le système offre la connexion au client.

4. L’acteur accède à l’interface d’ajout de Commande.

5. L’acteur passe les détails de commande .

6. Le système vérifie si cette commande existe déjà avant de créer une nouvelle commande .

* **Action de fin :** L’acteur effectue les opérations CRUD possibles sur une Commande.

# Diagramme de classe :

## Le diagramme de classe

Figure 10 : Diagramme de classe

*Explication :*

Pour la gestion des différents processus mis en œuvre pour l’exécution des tâches traités, nous avons défini 6 qui permettront de manipuler toutes les données requises dans le cycle de fonctionnement de l’application :

* La classe « Utilisateur » contient des attributs propres au client de notre Application qui sont « Nom » et « mot de passe ».
* La classe « Client » c’est une classe qui contient les informations propres du Client (id, Nom, Prénom, Adresse, Téléphone, Pays, Ville, Email).
* La classe « Commande » c’est une classe qui contient les informations propres à la Commande réalisée par un client (id, date, id client, total HT, Tva, Total TTC).
* La classe « Détail Commande » qui contient toutes les Détail d’une commande réalise par les clients.
* La classe « Produit » c’est une classe qui contient les informations propres du produit.
* La classe « Catégorie » qui contient toutes les Catégories des produits qui se trouvent dans le stock.

Les règles de gestions :

* Un client peut réaliser plusieurs Commande.
* Une Commande est passée par un seul client.
* Une catégorie peut contenir plusieurs produits.
* Un produit appartient à une seule catégorie.
* Une Commande peut contenir plusieurs produits.
* Un produit appartient à plusieurs commandes.
* Un détail commande appartient à une seule commande.

**Conclusion :**

Dans cette phase, toutes les questions concernant la manière de réaliser le système à développer ont été clarifiées. Le produit obtenu est un modèle graphique (ensemble de diagrammes) prêt à être codé. Dans le chapitre suivant nous allons étudier en détails les outils et les langages utilisés durant la phase de construction.

Implémentation

Les outils de développement :

Microsoft Vidual Studio :

**** **Microsoft Visual Studio** est une suite de logiciels de développement pour Windows et mac OS conçue par Microsoft. La dernière version s'appelle Visual Studio 2019.Visual Studio est un ensemble complet d'outils de développement permettant de générer des applications web ASP.NET, des services web XML, des applications bureautiques et des applications mobiles. Visual Basic, Visual C++, Visual C# utilisent tous le même environnement de développement intégré (IDE), qui leur permet de partager des outils et facilite la création de solutions faisant appel à plusieurs langages. Par ailleurs, ces langages permettent de mieux tirer parti des fonctionnalités du Framework .NET, qui fournit un accès à des technologies clés simplifiant le développement d'applications web ASP et de services web XML grâce à Visual Web Développer.

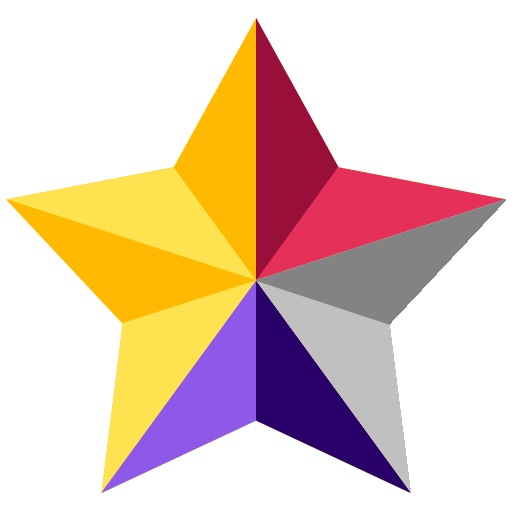
[https://fr.wikipedia.org/wiki/Microsoft\_Visual\_Studio](%20https:/fr.wikipedia.org/wiki/Microsoft_Visual_Studio)

Microsoft SQL Server Management Studio :

**Microsoft SQL Server** est un système de gestion de base de données relationnelle développé par Microsoft. En tant que serveur de base de données, il s'agit d'un produit logiciel dont la fonction principale est de stocker et de récupérer des données comme demandé par d'autres applications logicielles, qui peuvent fonctionner soit sur le même ordinateur, soit sur un autre ordinateur à travers un réseau (y compris Internet). Microsoft commercialise au moins une douzaine d'éditions différentes de Microsoft SQL Server, destinées à différents publics et pour des charges de travail allant des petites applications mono-machine aux grandes applications Internet avec de nombreux utilisateurs simultanés.

<https://en.wikipedia.org/wiki/Microsoft_SQL_Server>

StarUML :

**StarUML** est un logiciel de modélisation UML (Unified Modeling Language) open source qui peut remplacer dans bien des situations des logiciels commerciaux et coûteux comme Rational Rose1 ou Together2. Étant simple d’utilisation, nécessitant peu de ressources système, supportant UML 2, ce logiciel constitue une excellente option pour une familiarisation à la modélisation. Cependant, seule une version Windows est disponible.

<https://fr.wikipedia.org/wiki/StarUML>

GANT Project :

**Gantt Project** est un logiciel [libre](https://fr.wikipedia.org/wiki/Logiciel_libre) de [gestion de projet](https://fr.wikipedia.org/wiki/Gestion_de_projet) écrit en [Java](https://fr.wikipedia.org/wiki/Java_(langage)), ce qui permet de l'utiliser sur de nombreux OS tel que [Windows](https://fr.wikipedia.org/wiki/Windows), [Linux](https://fr.wikipedia.org/wiki/Linux), [MacOs](https://fr.wikipedia.org/wiki/MacOS). Ce projet a été lancé par un étudiant de l’[université de Marne La Vallée](https://fr.wikipedia.org/wiki/Universit%C3%A9_Paris-Est_Marne-la-Vall%C3%A9e) en janvier 2003 et est maintenant proposé sous [licence libre](https://fr.wikipedia.org/wiki/Licence_libre) ([GNU GPL](https://fr.wikipedia.org/wiki/Licence_publique_g%C3%A9n%C3%A9rale_GNU)). Le chef de projet initial est Alexandre Thomas, relayé aujourd'hui par Dmitry Barashev. Gantt Project permet la planification d'un projet à travers la réalisation d'un [diagramme de Gantt](https://fr.wikipedia.org/wiki/Diagramme_de_Gantt). L'outil permet de créer des diagrammes de Gantt, des diagrammes de ressources et des [réseaux PERT](https://fr.wikipedia.org/wiki/R%C3%A9seau_PERT). Il convient de noter qu'en l'état actuel le logiciel ne permet pas de concevoir des tâches durant moins d'une journée. Pour réaliser la planification de mon projet, j’ai utilisé la version 2.8.11

<https://fr.wikipedia.org/wiki/GanttProject>

Conclusion :

L’intégration au sein d’un organisme qui s’ouvre aux nouvelles technologies, nous a permis de mettre en œuvre de nouveaux services qui offrent une grande valeur ajoutée par rapport à certaines méthodes traditionnelles.

L’objectif de notre Stage de fin d’études était de concevoir et implémenter une application bureau de gestion de stock d’une société (UNIRSONE).

Le point de départ de la réalisation de ce projet était une récolte des informations nécessaires pour dresser un état de l’existant, présenter un aperçu sur la problématique

.

Par la suite, nous nous sommes intéressées à l’analyse et la spécification des besoins qui nous a permis de distinguer les différents acteurs interagissant avec l’application visée. L’objectif de la partie suivante était la conception détaillée, dans laquelle nous avons fixé la structure globale de l’application.

Le dernier volet de notre projet était la partie réalisation qui a été consacrée à la présentation des outils du travail et les interfaces les plus significatives de notre application.

Pour toute application, on peut toujours ajouter des améliorations, mais on se trouve souvent face à la contrainte du temps et de difficultés telle que le manque d’expérience en ce genre d’application a nécessité une période intense en autoformation.

L’apport de ce travail a été d’une importance très considérable, en effet, il nous a permis de suivre une méthodologie de travail bien étudié, d’approfondir nos connaissances dans le monde de développement des applications bureau.

La réalisation d’un tel projet, nous a permis d’apprendre et de toucher du doigt une partie de divers aspects du métier de développeur et de celui du concepteur.