|  |
| --- |
| Projet Stock manager |

Raphaël Schneiter

SI-CA2a

Table des matières

[1 Analyse préliminaire 4](#_Toc511999404)

[1.1 Introduction 4](#_Toc511999405)

[1.1 Organisation 5](#_Toc511999406)

[1.2 Objectifs 5](#_Toc511999407)

[1.3 Planification initiale 5](#_Toc511999408)

[2 Analyse / Conception 5](#_Toc511999409)

[2.1 Concept 5](#_Toc511999410)

[2.2 UseCase 6](#_Toc511999411)

[2.3 Scénarii 7](#_Toc511999412)

[2.4 Maquette 14](#_Toc511999413)

[2.5 MCD et MLD 20](#_Toc511999414)

[2.6 Stratégie de test 21](#_Toc511999415)

[2.7 Planification final 21](#_Toc511999416)

[2.8 Dossier de conception 21](#_Toc511999417)

[3 Réalisation 21](#_Toc511999418)

[3.1 Dossier de réalisation 21](#_Toc511999419)

[3.2 Test 22](#_Toc511999420)

[3.2.1 Description des tests effectués 22](#_Toc511999421)

[3.2.2 Test premier démarrage login et mot de passe vide : 22](#_Toc511999422)

[3.2.3 Test premier démarrage login remplis et mot de passe vide : 23](#_Toc511999423)

[3.2.4 Test premier démarrage login remplis et deux mot de passe différent 24](#_Toc511999424)

[3.2.5 Test premier démarrage réussi 25](#_Toc511999425)

[3.2.6 Test login vide et mot de passe vide : 26](#_Toc511999426)

[3.2.7 Test login faux et mot de passe vide : 26](#_Toc511999427)

[3.2.8 Test login valide et mot de passe vide : 27](#_Toc511999428)

[3.2.9 Test login valide et mot de passe non valide : 27](#_Toc511999429)

[3.2.10 Test login réussi mais base de donnée non trouver : 28](#_Toc511999430)

[3.2.11 Modification des options de connexion faux : 29](#_Toc511999431)

[3.2.12 Modification des options de connexion faux : 30](#_Toc511999432)

[3.2.13 Modification des options de connexion vrai : 32](#_Toc511999433)

[3.2.14 Test login réussi : 33](#_Toc511999434)

[3.2.15 Test ajout produit : 34](#_Toc511999435)

[3.2.16 Test Modification d’un produit : 36](#_Toc511999436)

[3.2.17 Test supprimer un produit 39](#_Toc511999437)

[3.2.18 Test ajouter un loueur 42](#_Toc511999438)

[3.2.19 Test modifier un loueur 44](#_Toc511999439)

[3.2.20 Test supprimer un loueur 46](#_Toc511999440)

[3.2.21 Test créée une location 49](#_Toc511999441)

[3.2.22 Test voir l’inventaire 51](#_Toc511999442)

[3.2.23 Test retour d’un produit 53](#_Toc511999443)

[3.2.24 Test de la recherche par mot-clé 55](#_Toc511999444)

[3.2.25 Test historique d’un produit 57](#_Toc511999445)

[3.2.26 Test Hors-délai des locations 59](#_Toc511999446)

[4 Conclusions 60](#_Toc511999447)

[5 Annexes 61](#_Toc511999448)

[5.1 Résumé du rapport du TPI / version succincte de la documentation 61](#_Toc511999449)

[5.2 Sources – Bibliographie 62](#_Toc511999450)

[5.3 Journal de travail 63](#_Toc511999451)

[5.4 Journal de bord 64](#_Toc511999452)

[5.5 Planification initial 65](#_Toc511999453)

[5.6 Planification final 66](#_Toc511999454)

[5.7 Manuel d'Installation 67](#_Toc511999455)

[5.8 Manuel d'Utilisation 68](#_Toc511999456)

# Analyse préliminaire

## Introduction

Le projet consiste à réaliser une application "clé en main" pour la gestion du stock d'une PME spécialisée dans la gestion de matériel de cuisine.

Durant le temps imparti, les tâches décrites ci-dessous devront être réalisées.

Procéder dans un premier temps à l'analyse et à la conception du programme, de la base de donnée avec entre autres l'architecture, la modélisation, les algorithmes, les maquettes avec la navigation. Suivra la réalisation avec l'implémentation de la base de données et les tests.

L'application s'adresse à des personnes qui n'ont pratiquement aucune notion en informatique.

L’application doit au préalable se connecter à une base de donnée pour pouvoir commencer à manipuler les données.

L’utilisation du logiciel nécessite un identifiant et un mot de passe unique.

Chaque instrument de cuisine dans l’inventaire est identifié par un code numérique unique (ID).

Cas d’utilisations :

* Entrée en stock :
  + Scénario 1 : saisie du code ID du matériel déjà connu du système
  + Scénario 2 : saisie d'une nouvelle pièce du stock avec création du code ID et Insertion des informations liées à la pièce (nom et description)
* Sortie de stock :
  + Scénario 1 : location d’une pièce avec saisie de son code ID, de la durée de la Location et du loueur (nom)
* Inventaire
  + Scénario 1 : Visualisation de toutes les pièces « sorties » et dans le stock.
* Recherche :
  + Scénario 1 : Recherche du matériel par mot-clé (sur le nom et la description)
  + Scénario 2 : Consultation de l’historique de location pour un matériel donné
  + Scénario 3 : Consultation du matériel en location hors-délai (matériel dont la date de retour est dépassée)

## Organisation

Elevé : SCHNEITER, Raphaël, [raphael.schneiter@cpnv.ch](mailto:raphael.schneiter@cpnv.ch)

Responsable de projet enseignant : CHEVILLAT, Jérôme, [jerome.chevillat@cpnv.ch](mailto:jerome.chevillat@cpnv.ch)

## Objectifs

1. Modèle de la base de donnée respectant les normes en vigueur
2. Algorithme de recherche par mot clé
3. Ergonomie de l’interface
4. Pertinence des messages affichés à l’utilisateur
5. Documentation du code
6. Qualité des tests effectués
7. Solution proposée pour la connexion unique au logiciel

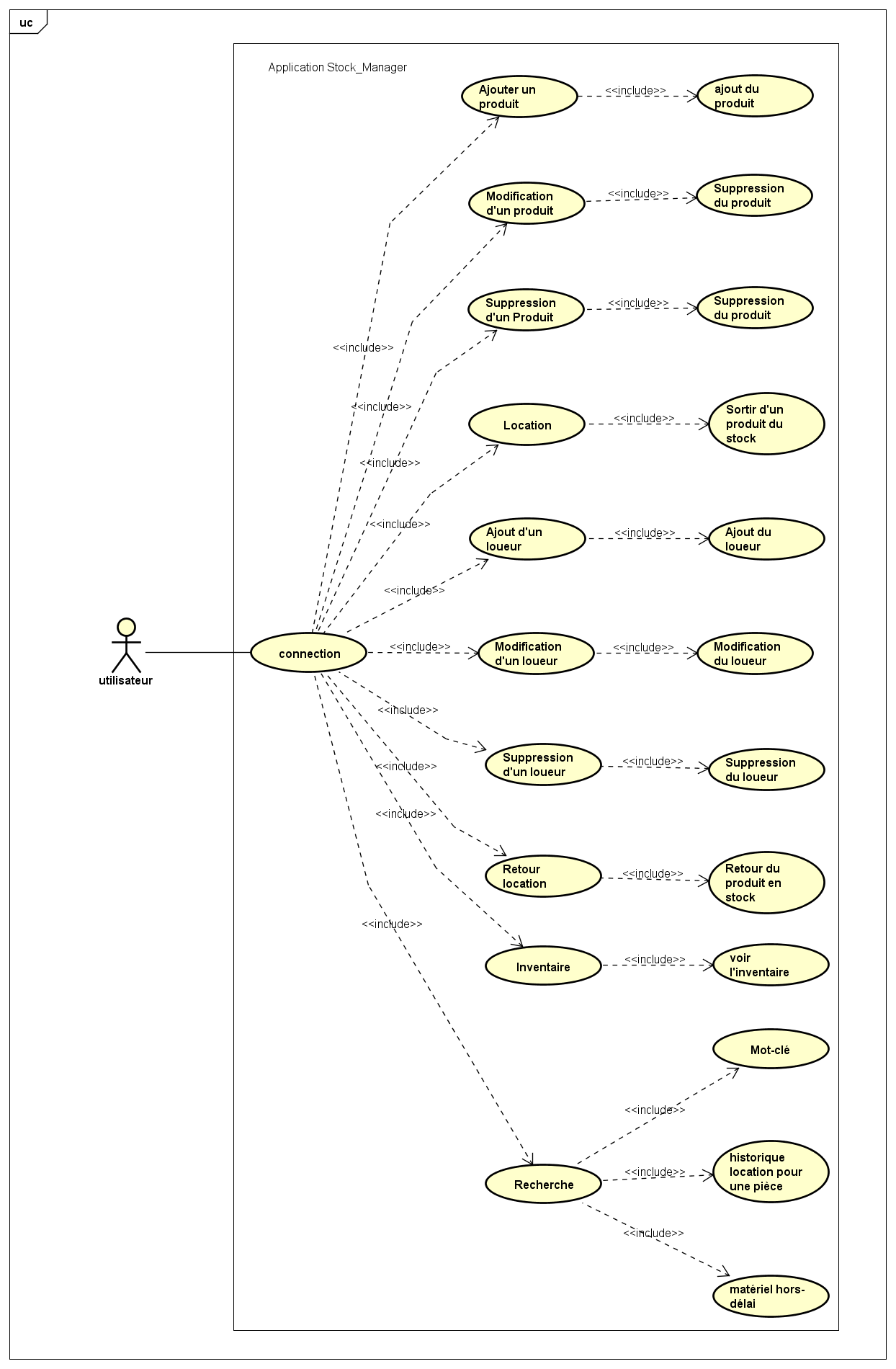
## Planification initiale

Voir annexe le chapitre 5.5 Planification initial

# Analyse / Conception

## Concept

## UseCase



## Scénarii

|  |  |
| --- | --- |
| Identifiant | Stock\_Manager01 |
| En tant que | Utilisateur |
| Je veux | Me connecter |
| Pour | Faire le retour d’un produit |

|  |  |
| --- | --- |
| **Action** | **Réaction** |
| Je clique sur l’application | L’application démarre |
| Ensuite je me connecte | L’application me montre le menu |
| Je clic sur Retour d’un produit | Une fenêtre s’ouvre |
| Dans la fenêtre, je sélection le numéro du produit en retour | L’application indique la réussite |

|  |  |
| --- | --- |
| Identifiant | Stock\_Manager02 |
| En tant que | Utilisateur |
| Je veux | Me connecter |
| Pour | Ajouter un nouveau produit |

|  |  |
| --- | --- |
| **Action** | **Réaction** |
| Je clique sur l’application | L’application démarre |
| Ensuite je me connecte | L’application me montre le menu |
| Je clic sur Ajout/ modification produit | Une fenêtre s’ouvre |
| Dans la fenêtre, je sélection Ajouter un produit | L’application indique les champs supplémentaires |
| Je remplis les champs supplémentaires et je valide | L’application efface tous les champs |

|  |  |
| --- | --- |
| Identifiant | Stock\_Manager03 |
| En tant que | Utilisateur |
| Je veux | Me connecter |
| Pour | Pour faire la sortie d’un produit |

|  |  |
| --- | --- |
| **Action** | **Réaction** |
| Je clique sur l’application | L’application démarre |
| Ensuite je me connecte | L’application me montre le menu |
| Je clic sur Location d’un produit | Une fenêtre s’ouvre |
| Dans la fenêtre, je sélection le numéro du produit et le nom du loueur, puis je valide la location | L’application ajout la location dans la base de donnée |

|  |  |
| --- | --- |
| Identifiant | Stock\_Manager04 |
| En tant que | Utilisateur |
| Je veux | Me connecter |
| Pour | Pour voir l’inventaire de toutes les produit |

|  |  |
| --- | --- |
| **Action** | **Réaction** |
| Je clique sur l’application | L’application démarre |
| Ensuite je me connecte | L’application me montre le menu |
| Je clic sur Voir l’inventaire | Une fenêtre s’ouvre et affiche la liste des produits en stock et les locations en cours |

|  |  |
| --- | --- |
| Identifiant | Stock\_Manager05 |
| En tant que | Utilisateur |
| Je veux | Me connecter |
| Pour | Faire une recherche par mot-clé |

|  |  |
| --- | --- |
| **Action** | **Réaction** |
| Je clique sur l’application | L’application démarre |
| Ensuite je me connecte | L’application me montre le menu |
| Je clic sur Affichez le menu pour la recherche | L’application me montre un menu |
| Je clic sur Mot-clé | L’application affiche une ligne de saisie |
| J’écrite un mot ou un nombre | L’application fait la recherche et l’affiche |

|  |  |
| --- | --- |
| Identifiant | Stock\_Manager06 |
| En tant que | Utilisateur |
| Je veux | Me connecter |
| Pour | Faire une recherche par historique de location pour une produit |

|  |  |
| --- | --- |
| **Action** | **Réaction** |
| Je clique sur l’application | L’application démarre |
| Ensuite je me connecte | L’application me montre le menu |
| Je clic sur Affichez le menu pour la recherche | L’application me montre un menu |
| Je clic sur historique | L’application affiche une ligne de saisie |
| Je sélection l’id du produit puis je valide avec le bouton Chercher | L’application fait la recherche et l’affiche l’historique |

|  |  |
| --- | --- |
| Identifiant | Stock\_Manager07 |
| En tant que | Utilisateur |
| Je veux | Me connecter |
| Pour | Faire une recherche pour voir le matériel qui est hors-délai |

|  |  |
| --- | --- |
| **Action** | **Réaction** |
| Je clique sur l’application | L’application démarre |
| Ensuite je me connecte | L’application me montre le menu |
| Je clic sur Affichez le menu pour la recherche | L’application me montre un menu |
| Je clic sur hors-délai | L’application affiche la liste des produits pas rendu dans les temps |

|  |  |
| --- | --- |
| Identifiant | Stock\_Manager08 |
| En tant que | Utilisateur |
| Je veux | Me connecter |
| Pour | Modifier un produit |

|  |  |
| --- | --- |
| **Action** | **Réaction** |
| Je clique sur l’application | L’application démarre |
| Ensuite je me connecte | L’application me montre le menu |
| Je clic sur Ajout/ modification produit | Une fenêtre s’ouvre |
| Dans la fenêtre, je sélection un numéro | L’application remplis les champs automatiquement |
| Je modifier le nom ou la description du produit et je valide avec le bouton Modification | L’application modifier le produit dans la base de donnée |

|  |  |
| --- | --- |
| Identifiant | Stock\_Manager09 |
| En tant que | Utilisateur |
| Je veux | Me connecter |
| Pour | Supprimer un produit |

|  |  |
| --- | --- |
| **Action** | **Réaction** |
| Je clique sur l’application | L’application démarre |
| Ensuite je me connecte | L’application me montre le menu |
| Je clic sur Ajout/ modification produit | Une fenêtre s’ouvre |
| Dans la fenêtre, je sélection un numéro | L’application remplis les champs automatiquement |
| Je clique sur Supprimer | L’application affiche une demande de confirmation avec le nom du produit sélection |
| 1.a je clique sur non | L’application n’efface pas le produit |
| 1.b. je clique sur oui | L’application efface le produit |

|  |  |
| --- | --- |
| Identifiant | Stock\_Manager10 |
| En tant que | Utilisateur |
| Je veux | Me connecter |
| Pour | Supprimer un loueur |

|  |  |
| --- | --- |
| **Action** | **Réaction** |
| Je clique sur l’application | L’application démarre |
| Ensuite je me connecte | L’application me montre le menu |
| Je clic sur Ajout/ modification loueur | Une fenêtre s’ouvre |
| Dans la fenêtre, je sélection un nom | L’application remplis les champs automatiquement |
| Je clique sur Supprimer | L’application affiche une demande de confirmation avec le nom du loueur sélection |
| 1.a je clique sur non | L’application n’efface pas le loueur |
| 1.b. je clique sur oui | L’application efface le loueur |

|  |  |
| --- | --- |
| Identifiant | Stock\_Manager11 |
| En tant que | Utilisateur |
| Je veux | Me connecter |
| Pour | Modifier un loueur |

|  |  |
| --- | --- |
| **Action** | **Réaction** |
| Je clique sur l’application | L’application démarre |
| Ensuite je me connecte | L’application me montre le menu |
| Je clic sur Ajout/ modification loueur | Une fenêtre s’ouvre |
| Dans la fenêtre, je sélection un nom | L’application remplis les champs automatiquement |
| Je clique sur Modifier | L’application modifie le nom du loueur dans la base de donnée |

|  |  |
| --- | --- |
| Identifiant | Stock\_Manager12 |
| En tant que | Utilisateur |
| Je veux | Me connecter |
| Pour | Ajouter un loueur |

|  |  |
| --- | --- |
| **Action** | **Réaction** |
| Je clique sur l’application | L’application démarre |
| Ensuite je me connecte | L’application me montre le menu |
| Je clic sur Ajout/ modification loueur | Une fenêtre s’ouvre |
| Dans la fenêtre, je sélection Ajouter un loueur | L’application fait rien |
| je remplis le champ et j’appuis sur ajouter | L’application vide le champ |

## Maquette

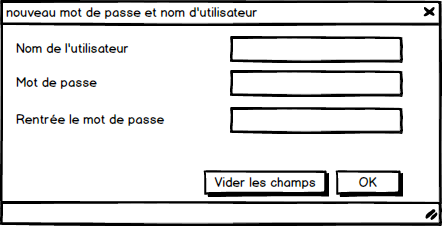


Figure 1 Premier login

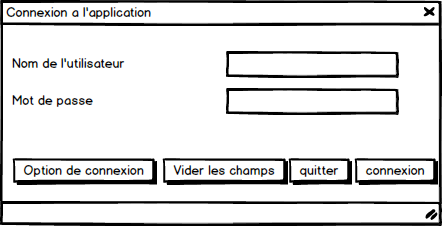


Figure 2 connexion à l'application

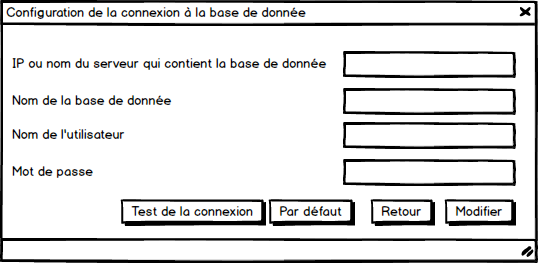


Figure 3 Configuration de la connexion à la base de donnée

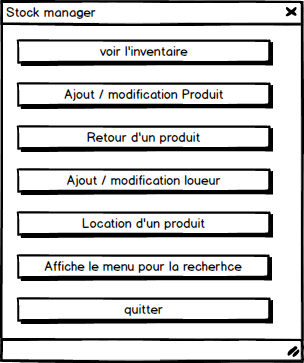


Figure 4 Menu Principal

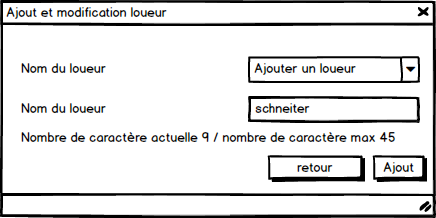


Figure 5 Ajout d'un loueur

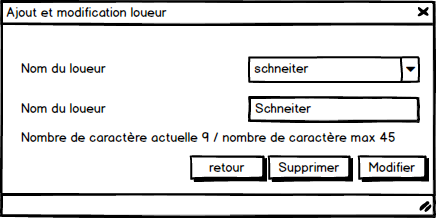


Figure 6 modifier ou supprimer un loueur

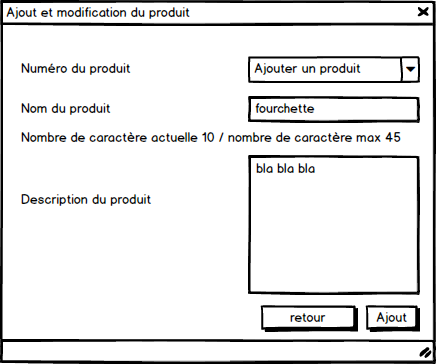


Figure 7 Ajout d'un produit

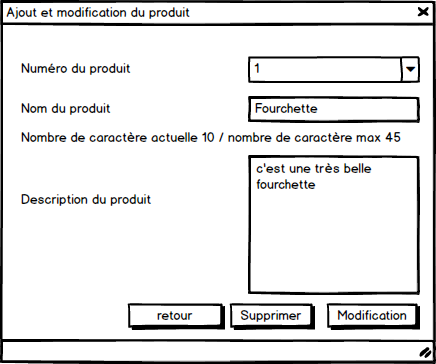


Figure 8 Modifier ou supprimer un produit

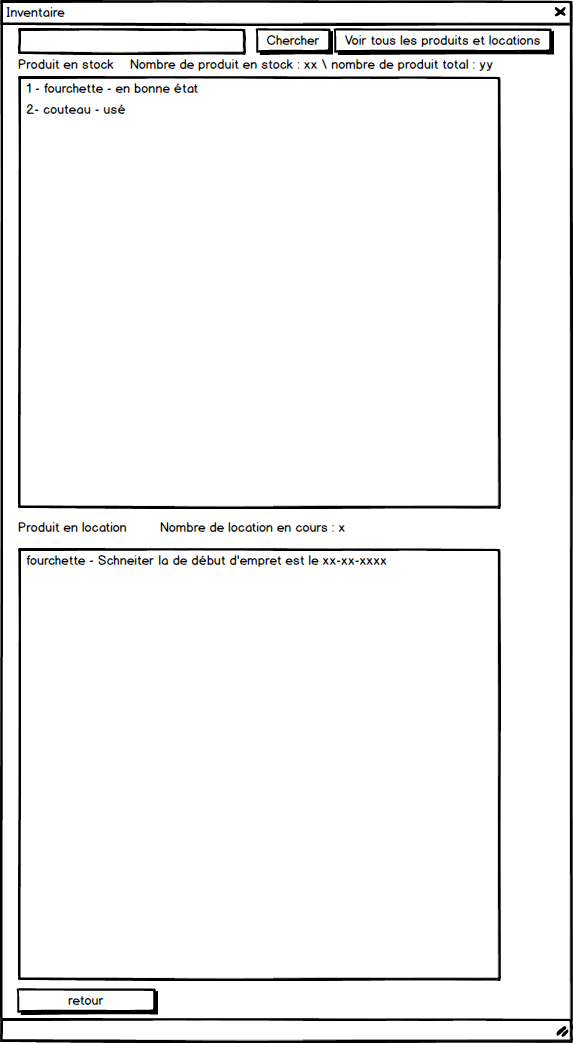


Figure 9 Inventaire

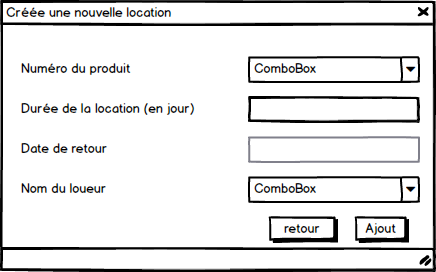


Figure 10 Créée une nouvelle location

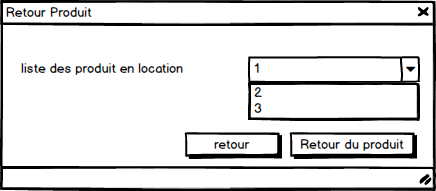


Figure 11 Retour d'un produit d’une location

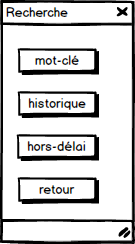


Figure 12 Menu des recherche

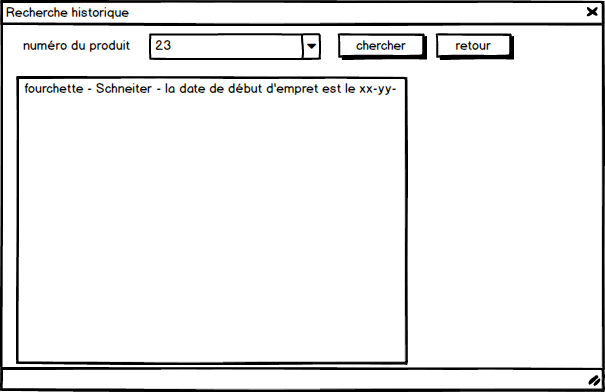


Figure 13 Rechercher de l'historique d'un produit

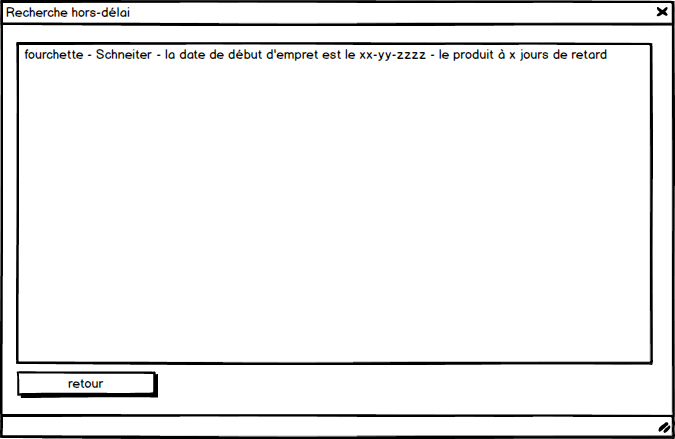


Figure 14 Recherche des produit hors-délai

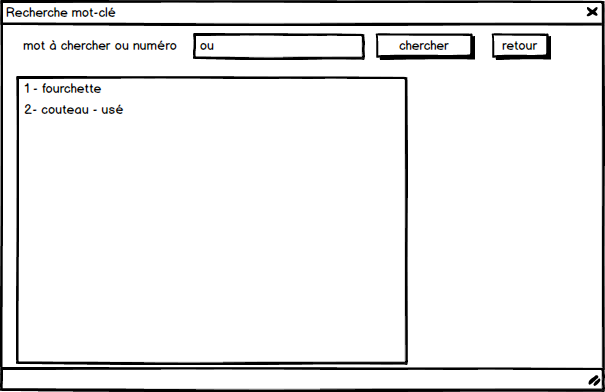


Figure 15 Recherche par mot-clé

## MCD et MLD



Figure 16 MCD

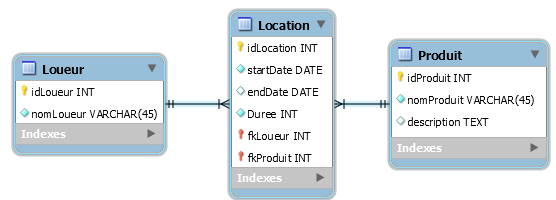


Figure 17 MLD

## Stratégie de test

Décrire la stratégie globale de test :

* Test d’intégration
  + Pour les test d’intégration à chaque nouvelle fonction codé je l’ai test
* Test de performance sur le code et la base de donnée
  + Je regarde comment se comporte l’application avec beaucoup de produit
  + Je regarde comment se comporte la base de donnée avec un « select » qui retourne environ 1000 résultat
* Test d’intégration sur le système OS
  + Je test un utilisateur qui n’a pas de droit administrateur comment se comporte l’application s’il lui faut des droits particuliers
* Couverture des tests (tests exhaustifs ou non, si non, pourquoi ?).
  + Test exhaustifs pour le test d’intégration sur le système OS
  + Je n’approfondis pas les autres tests car ils sont assez simple
* Données de test à prévoir
  + 1000 produits et 1000 clients et 25 location en cours avec environ 10 locations en retard, 10 location rendu

## Planification final

Voir annexe le chapitre 5.6 Planification final

## Dossier de conception

* Utilisation de Microsoft Visual Studio 2015 pour faire le code et les tests.
* Utilisation de WAMP pour avoir une base de donnée en local sur la machine et d’une machine Debian 9 avec une base de donnée MariaDB.
* Utilisation de MySQL Workbench pour la création de la base de donnée et ajouter des données dans la base de donnée.
* Utilisation d’une machine Windows 7 pour le développement et pour les tests

# Réalisation

## Dossier de réalisation

Décrire la réalisation "physique" de votre projet

Par défaut le programme s’installe dans « C:\Program Files (x86)\Stock-Manager »

Utiliser le fichier script.sql sur la machine qui abrite une base de donnée pour créer la base de donnée et l’utilisateur par défaut

Visual studio Entreprise 2015 version 14.0.25431.01 Update 3

Windows 7 Entreprise SP1 (32 Go de RAM, Intel Core i7-4770 CPU @3.40GHz, 64 bit)

## Test

### Description des tests effectués

Avec MySQL Workbench il faut créée une connexion avec la base de donnée pour créer la base de donnée en utilisent le fichier « script.sql »

### Test premier démarrage login et mot de passe vide :

|  |  |  |
| --- | --- | --- |
| Démarche | Résultat attendu | Validé |
| Ouvrir l’application | L’application s’ouvre et une fenêtre pour mettre un login et mot de passe s’ouvre | OK |
| Laissez les champs login et mot de passe vide | L’application affiche un message d’erreur | OK |

### Test premier démarrage login remplis et mot de passe vide :

|  |  |  |
| --- | --- | --- |
| Démarche | Résultat attendu | Validé |
| Ouvrir l’application | L’application s’ouvre et une fenêtre pour mettre un login et mot de passe s’ouvre | OK |
| Mettez comme login « vendeur » et pas de mot de passe | L’application affiche un message d’erreur | Ok |

### Test premier démarrage login remplis et deux mot de passe différent

|  |  |  |
| --- | --- | --- |
| Démarche | Résultat attendu | Validé |
| Ouvrir l’application | L’application s’ouvre et une fenêtre pour mettre un login et mot de passe s’ouvre | OK |
| Mettez comme login « vendeur » et mot de passe « Pa$$w0rd » et rentrée le mot de passe « PA$$w0rd » | L’application affiche un message d’erreur | Ok |

### Test premier démarrage réussi

|  |  |  |
| --- | --- | --- |
| Démarche | Résultat attendu | Validé |
| Ouvrir l’application | L’application s’ouvre et une fenêtre pour mettre un login et mot de passe s’ouvre | OK |
| Mettez comme login « vendeur » et mot de passe « Pa$$w0rd » et rentrée le mot de passe « Pa$$w0rd » | L’application affiche la fenêtre de connexion | OK |

### Test login vide et mot de passe vide :

|  |  |  |
| --- | --- | --- |
| Démarche | Résultat attendu | Validé |
| Ouvrir l’application | L’application s’ouvre | OK |
| Laisser les champs vide | L’application afficher un message d’erreur | Ok |

### Test login faux et mot de passe vide :

|  |  |  |
| --- | --- | --- |
| Démarche | Résultat attendu | Validé |
| Ouvrir l’application | L’application s’ouvre | OK |
| Mettez comme login « Vendeur » et pas de mot de passe | L’application afficher un message d’erreur | Ok |

### Test login valide et mot de passe vide :

|  |  |  |
| --- | --- | --- |
| Démarche | Résultat attendu | Validé |
| Ouvrir l’application | L’application s’ouvre | OK |
| Mettez comme login « vendeur » et pas de mot de passe | L’application afficher un message d’erreur | Ok |

### Test login valide et mot de passe non valide :

|  |  |  |
| --- | --- | --- |
| Démarche | Résultat attendu | Validé |
| Ouvrir l’application | L’application s’ouvre | OK |
| Mettez comme login « vendeur » et comme mot de passe « PA$$w0rd » | L’application afficher un message d’erreur | Ok |

### Test login réussi mais base de donnée non trouver :

|  |  |  |
| --- | --- | --- |
| Démarche | Résultat attendu | Validé |
| Ouvrir l’application | L’application s’ouvre | OK |
| Mettez comme login « vendeur » et comme mot de passe « Pa$$w0rd » | L’application afficher une message d’erreur | Ok |

### Modification des options de connexion faux :

|  |  |  |
| --- | --- | --- |
| Démarche | Résultat attendu | Validé |
| Ouvrir l’application | L’application s’ouvre | OK |
| Cliquer sur « Option de connexion » | L’application affiche le formulaire des options de connexion | OK |
| Enlever le mot de passe de connexion ou l’utilisateur |  | OK |
| Cliquer sur « Test de la connexion » | L’application affiche une message d’erreur | OK |

### Modification des options de connexion faux :

|  |  |  |
| --- | --- | --- |
| Démarche | Résultat attendu | Validé |
| Ouvrir l’application | L’application s’ouvre | OK |
| Cliquer sur « Option de connexion » | L’application affiche le formulaire des options de connexion | OK |
| Donner une fausse adresse IP ou nom de serveur |  | OK |
| Cliquer sur « Test de la connexion » | L’application affiche une message d’erreur | OK |

### Modification des options de connexion vrai :

|  |  |  |
| --- | --- | --- |
| Démarche | Résultat attendu | Validé |
| Ouvrir l’application | L’application s’ouvre | OK |
| Cliquer sur « Option de connexion » | L’application affiche le formulaire des options de connexion | OK |
| Mettre les bonnes valeurs dans les champs |  | OK |
| Cliquer sur « Test de la connexion » | L’application affiche une message d’information | OK |

### Test login réussi :

|  |  |  |
| --- | --- | --- |
| Démarche | Résultat attendu | Validé |
| Ouvrir l’application | L’application s’ouvre | OK |
| Mettez comme login « vendeur » et comme mot de passe « Pa$$w0rd » | L’application affiche le menu | Ok |

### Test ajout produit :

|  |  |  |
| --- | --- | --- |
| Démarche | Résultat attendu | Validé |
| Ouvrir l’application | L’application s’ouvre | OK |
| Mettez comme login « vendeur » et comme mot de passe « Pa$$w0rd » | L’application affiche le menu | Ok |
| Sélectionner « Ajouter un produit » dans la combobox |  | OK |
| Remplisse les différent champs avec des valeurs puis cliquer sur « Ajouter » |  | OK |
| l’application affiche un message de confirmation |  | OK |
| Les différent champs sont vide |  | OK |

### Test Modification d’un produit :

|  |  |  |
| --- | --- | --- |
| Démarche | Résultat attendu | Validé |
| Ouvrir l’application | L’application s’ouvre | OK |
| Mettez comme login « vendeur » et comme mot de passe « Pa$$w0rd » | L’application affiche le menu | Ok |
| Cliquer sur « Ajout / modification produit » |  | OK |
| Sélectionner « le numéro du produit » dans la combobox |  | OK |
| Modifier les différent champs avec des valeurs puis cliquer sur « Modifier » |  | OK |
| l’application affiche un message de confirmation |  | OK |
| Les différent champs sont vide |  | OK |

### Test supprimer un produit

|  |  |  |
| --- | --- | --- |
| Démarche | Résultat attendu | Validé |
| Ouvrir l’application | L’application s’ouvre | OK |
| Mettez comme login « vendeur » et comme mot de passe « Pa$$w0rd » | L’application affiche le menu | Ok |
| Cliquer sur « Ajout / modification produit » |  | OK |
| Sélectionner « le numéro du produit » dans la combobox |  | OK |
| Cliquer sur « Supprimer » |  | OK |
| l’application affiche un message de confirmation pour la suppression |  | OK |
| Si vous cliquer sur non le produit n’est pas effacer |  | OK |
| Si vous cliquer sur oui le produit sera effacer et toutes les locations qui y correspond aussi | Une fenêtre de confirmation apparée | OK |
| Les différent champs sont vide |  | OK |

### Test ajouter un loueur

|  |  |  |
| --- | --- | --- |
| Démarche | Résultat attendu | Validé |
| Ouvrir l’application | L’application s’ouvre | OK |
| Mettez comme login « vendeur » et comme mot de passe « Pa$$w0rd » | L’application affiche le menu | Ok |
| Cliquer sur « Ajout / modification loueur » |  | OK |
| Remplisse le champ avec une valeur puis cliquer sur « Ajouter » |  | OK |
| l’application affiche un message de confirmation |  | OK |

### Test modifier un loueur

|  |  |  |
| --- | --- | --- |
| Démarche | Résultat attendu | Validé |
| Ouvrir l’application | L’application s’ouvre | OK |
| Mettez comme login « vendeur » et comme mot de passe « Pa$$w0rd » | L’application affiche le menu | Ok |
| Cliquer sur « Ajout / modification loueur » |  | OK |
| Sélectionner « nom du loueur » dans la combobox |  | OK |
| Modifier le différent champs avec une nouvelle valeur puis cliquer sur « Modifier » |  | OK |
| l’application affiche un message de confirmation |  | OK |
| Les différent champs sont vide |  | OK |

### Test supprimer un loueur

|  |  |  |
| --- | --- | --- |
| Démarche | Résultat attendu | Validé |
| Ouvrir l’application | L’application s’ouvre | OK |
| Mettez comme login « vendeur » et comme mot de passe « Pa$$w0rd » | L’application affiche le menu | Ok |
| Cliquer sur « Ajout / modification loueur » |  | OK |
| Sélectionner « nom du loueur » dans la combobox |  | OK |
| Cliquer sur « Supprimer » |  | OK |
| l’application affiche un message de confirmation pour la suppression |  | OK |
| Si vous cliquer sur non le produit n’est pas effacer |  | OK |
| Si vous cliquer sur oui le produit sera effacer et toutes les locations qui y correspond aussi | Une fenêtre de confirmation apparée | OK |
| Les différent champs sont vide |  | OK |

### Test créée une location

|  |  |  |
| --- | --- | --- |
| Démarche | Résultat attendu | Validé |
| Ouvrir l’application | L’application s’ouvre | OK |
| Mettez comme login « vendeur » et comme mot de passe « Pa$$w0rd » | L’application affiche le menu | Ok |
| Cliquer sur « Location d’un produit » |  | OK |
| Sélectionner un produit dans la combobox « Numéro du produit », indiquer la durée de la location et sélectionner dans la combobox le nom du loueur, puis cliquer sur « Ajouter » |  | OK |
|  | Une fenêtre de confirmation apparée | OK |

### Test voir l’inventaire

|  |  |  |
| --- | --- | --- |
| Démarche | Résultat attendu | Validé |
| Ouvrir l’application | L’application s’ouvre | OK |
| Mettez comme login « vendeur » et comme mot de passe « Pa$$w0rd » | L’application affiche le menu | Ok |
| Cliquer sur « Voir l’inventaire » |  | OK |
| Dans la barre de recherche mette « 2 » puis cliquer sur « Chercher » | L’application affiche le résultat de la recheche | OK |
| Cliquer sur « Voir tous les produits et locations » | L’application affiche tous les produits et toutes les location | OK |

### Test retour d’un produit

|  |  |  |
| --- | --- | --- |
| Démarche | Résultat attendu | Validé |
| Ouvrir l’application | L’application s’ouvre | OK |
| Mettez comme login « vendeur » et comme mot de passe « Pa$$w0rd » | L’application affiche le menu | Ok |
| Cliquer sur « Retour d’un produit » |  | OK |
| Dans la combobox sélectionner le produit à faire le retour, puis cliquer sur « Retour du produit » |  | OK |
|  | Une fenêtre de confirmation apparée | OK |

### Test de la recherche par mot-clé

|  |  |  |
| --- | --- | --- |
| Démarche | Résultat attendu | Validé |
| Ouvrir l’application | L’application s’ouvre | OK |
| Mettez comme login « vendeur » et comme mot de passe « Pa$$w0rd » | L’application affiche le menu | Ok |
| Cliquer sur « Afficher le menu pour la recherche » |  | OK |
| Cliquer sur « Mot-clé » |  | OK |
| Dans le champ texte mettez soit un nombre, soit un mot puis cliquer sur « Chercher » |  | OK |

### Test historique d’un produit

|  |  |  |
| --- | --- | --- |
| Démarche | Résultat attendu | Validé |
| Ouvrir l’application | L’application s’ouvre | OK |
| Mettez comme login « vendeur » et comme mot de passe « Pa$$w0rd » | L’application affiche le menu | Ok |
| Cliquer sur « Afficher le menu pour la recherche » |  | OK |
| Cliquer sur « Historique » |  | OK |
| Dans la combobox sélectionner le produit 1 puis cliquer sur « Chercher » |  | OK |
|  | L’application affiche l’historique du produit « 1 » | OK |

### Test Hors-délai des locations

Avec MySQL Workbench il faut créée une connexion avec la base de donnée pour créer manuellement une location qui sera en retard d’un jour

|  |  |  |
| --- | --- | --- |
| Démarche | Résultat attendu | Validé |
| Ouvrir l’application | L’application s’ouvre | OK |
| Mettez comme login « vendeur » et comme mot de passe « Pa$$w0rd » | L’application affiche le menu | Ok |
| Cliquer sur « Afficher le menu pour la recherche » |  | OK |
| Cliquer sur « Hors-délai » | L’application affiche la liste des location qui sont en retard | OK |

# Conclusions

Tous les objectifs ont été atteints

Les points positifs :

* J’ai appris à faire des recherches sur le net plus pousser pour trouver des réponses a question sur le développement en C#.

Les points négatifs :

Difficultés particulières, il y a eu un problème avec la dll Mysql.data.dll, ce problème a été résolu avec un changement de la version de la dll

Amélioration possible :

* Rajouter une rechercher pour les loueurs pour savoir qui à loueur quel produit

# Annexes

## Résumé du rapport du TPI / version succincte de la documentation

## Sources – Bibliographie

<https://openclassrooms.com/courses/creer-une-installation> Comment créer une installation avec un .exe

<https://stackoverflow.com/questions/2818179/how-do-i-force-my-net-application-to-run-as-administrator> Comment lancer l’application en mode administrateur

<https://www.digitalocean.com/community/tutorials/how-to-create-a-new-user-and-grant-permissions-in-mysql> comment crée un utilisateur et lui donner des droit.

<https://stackoverflow.com/questions/3036829/how-do-i-create-a-message-box-with-yes-no-choices-and-a-dialogresult> comment crée une messageBox avec bouton oui non et regarder le résultat.

<http://davismj.me/blog/bcrypt/> utiliser pour faire le cryptage du login et du mot de passe

<https://www.mockaroo.com/> utiliser pour faire des données réel dans la base de donnée

<https://stackoverflow.com/questions/11492705/how-to-create-xml-document-using-xmldocument> exemple avec xmldocument et xdocument

<https://www.youtube.com/watch?v=sfDPdflXbiM> exemple lecture xml avec linq

<https://www.youtube.com/watch?v=uwa9op2IrM4> exemple lecture xml avec linq

<https://stackoverflow.com/questions/1082852/connect-to-mysql-with-hashed-password> explique que on ne peut pas mettre le mot de passe de connexion à MySQL en hash

M. ITHURBIDE

## Journal de travail

## Journal de bord

## Planification initial

## Planification final

## Manuel d'Installation

## Manuel d'Utilisation