30/05/2023

Candidat : BARRAS Tristan

Chef de projet : ANDOLFATTO Frédérique

Application C# de gestion d’inventaire informatique

Table des matières

[Analyse préliminaire 3](#_Toc134083784)

[Introduction 3](#_Toc134083785)

[Objectifs 3](#_Toc134083786)

[Méthode de gestion de projet 3](#_Toc134083787)

[Planification initiale 4](#_Toc134083788)

[Analyse / Conception 5](#_Toc134083789)

[Concept 5](#_Toc134083790)

[Stratégie de test 11](#_Toc134083791)

[Planification 12](#_Toc134083792)

[Dossier de conception 13](#_Toc134083793)

[Réalisation 14](#_Toc134083794)

[Dossier de réalisation 14](#_Toc134083795)

[Description des tests effectués 15](#_Toc134083796)

[Erreurs restantes 15](#_Toc134083797)

[Liste des documents fournis 15](#_Toc134083798)

[Conclusions 16](#_Toc134083799)

[Annexes 17](#_Toc134083800)

[Résumé du rapport du TPI / version succincte de la documentation 17](#_Toc134083801)

[Sources – Bibliographie 17](#_Toc134083802)

[Journal de travail 17](#_Toc134083803)

[Manuel d'Installation 17](#_Toc134083804)

[Manuel d'Utilisation 17](#_Toc134083805)

[Archives du projet 17](#_Toc134083806)

*:*

# Analyse préliminaire

## Introduction

Ce projet est réalisé dans le cadre d’un TPI, il y a 90 heures de travail. Le projet est une application en C# de gestion d’inventaire de matériel informatique de l’école. Elle permettra de parcourir, ajouter, modifier du matériel informatique présent dans une base de données. Pour ce projet, j’ai des compétences en développement sur du C# ainsi que des compétences en modélisation et gestion de bases de données. J’ai fait un projet similaire le pré-TPI qui a duré 72 heures, ce qui me permet d’avoir un peu d’expérience pour la gestion d’un projet comme le TPI.

## Objectifs

Les objectifs de ce projet sont en premier de créer une partie administrateur disponible avec une authentification et qui pourra à l’aide d’une interface graphique insérer dans une base de données des informations sur du matériel informatique. Le deuxième objectif est de pouvoir créer/modifier des listes pour les marques du matériel, les modules ainsi que l’emplacement où ce qu’il se trouve. Le troisième objectif est le fait de pouvoir modifier les données du matériel ainsi que l’archivé. Le quatrième objectif est de faire une interface graphique pour que les utilisateurs puissent consulter le matériel informatique grâce à des critères et il peut voir toutes les informations sur le produit en cliquant dessus.

## Méthode de gestion de projet

La méthode de gestion de projet qui est utilisé est la méthode dite en cascade. Cette méthode a été choisie par confort en effet, ma méthode de travail correspond mieux à cette méthode que par exemple la méthode agile. Je me retrouve plus dans cette méthode qui me permet de bien suivre le projet étape par étape. La méthode se divise en 5 phases qui doivent se suivre successivement. La première phase l’analyse est dans ce cas l’analyse du cahier des charges normalement cette étape comprend la conception du cahier des charges, mais dans cette situation le cahier des charges a été reçu le premier jour. La seconde étape la conception comprend dans ce cas l’analyse initiale, la modélisation de la base de données, la maquette de l’application ainsi que le point analyse/conception dans le rapport du projet. La troisième étape est l’implémentation comprend la partie programmation. La quatrième partie, l’étape teste comprend tous les tests prévus dans la partie conception puis l’exploitation est la première version du produit est finie.

## Planification initiale

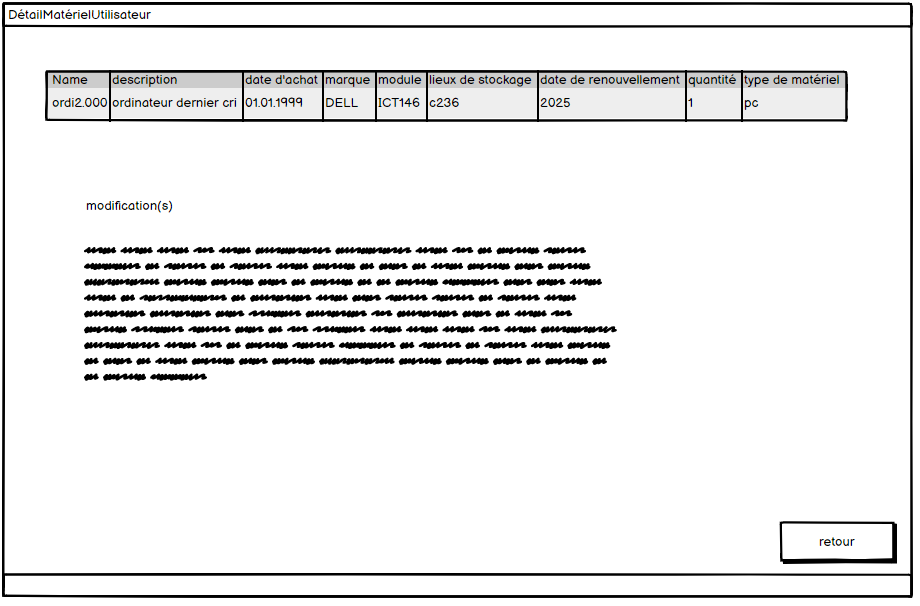
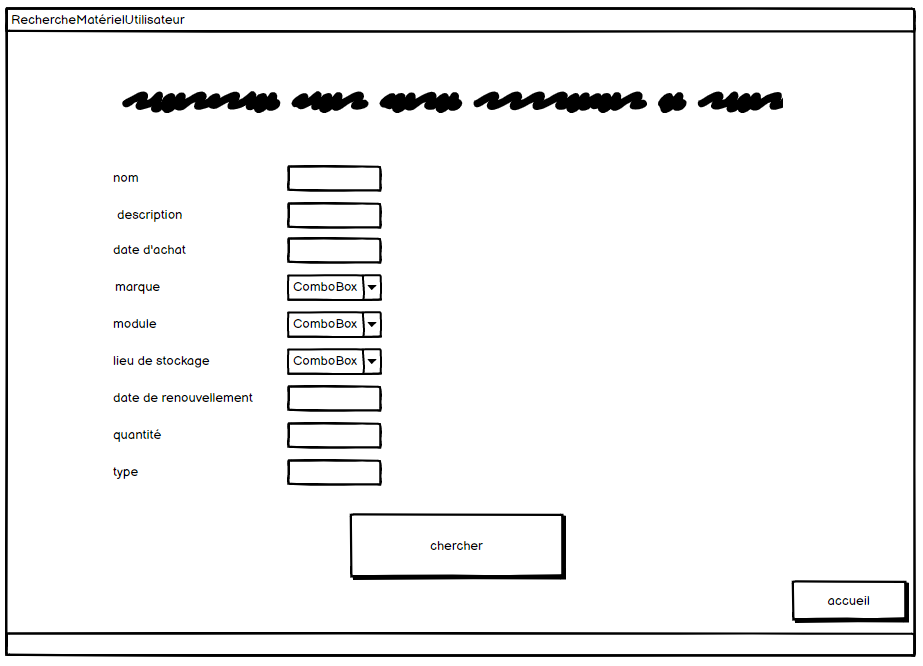
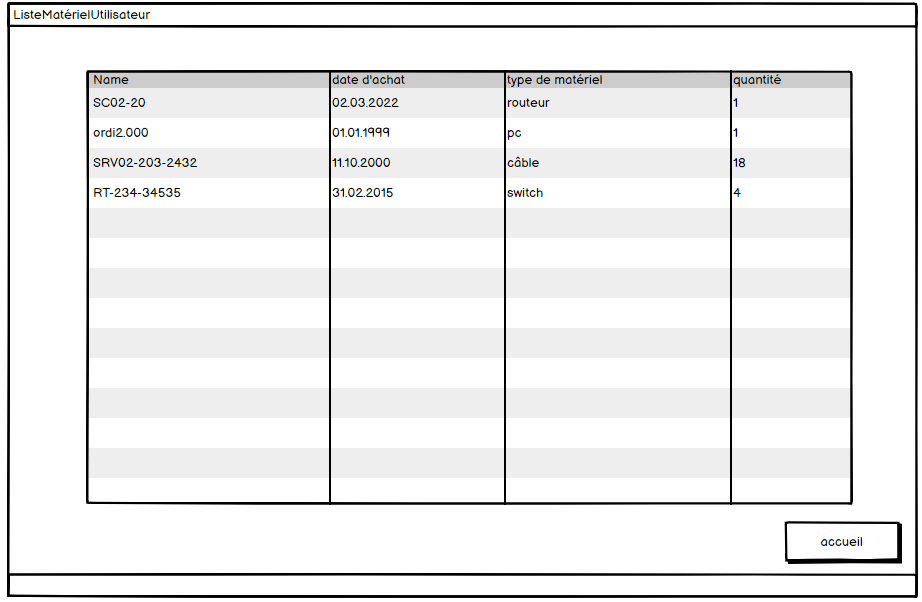
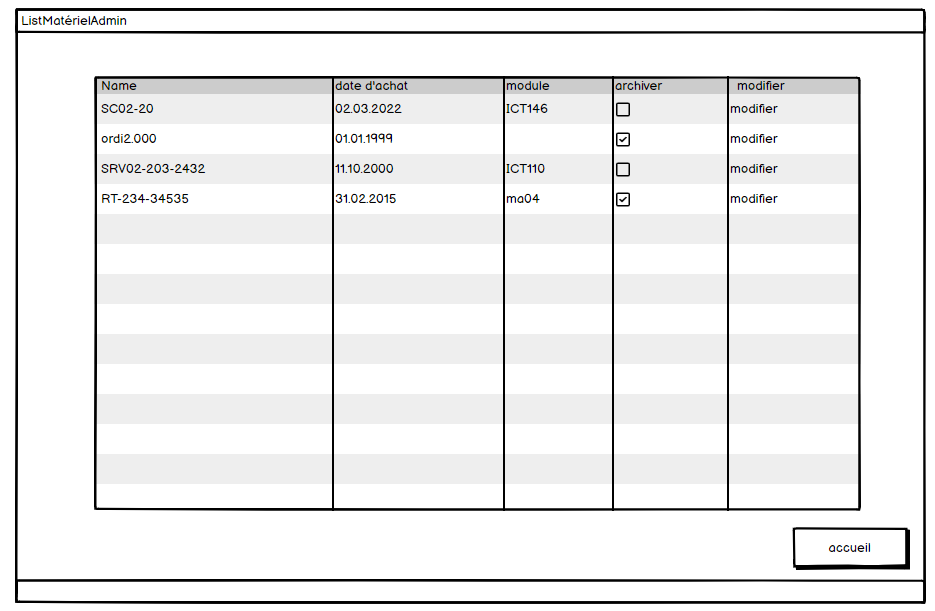
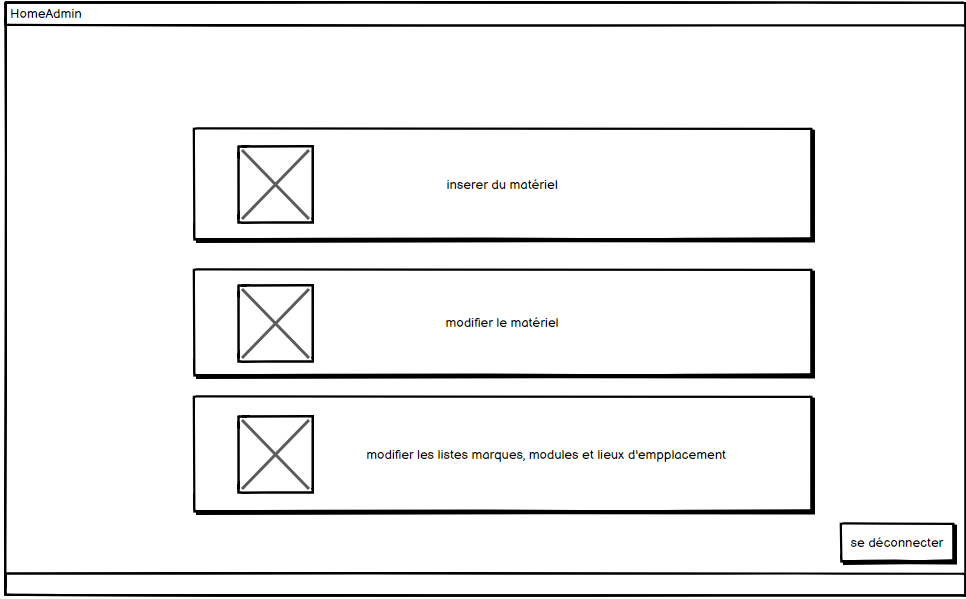
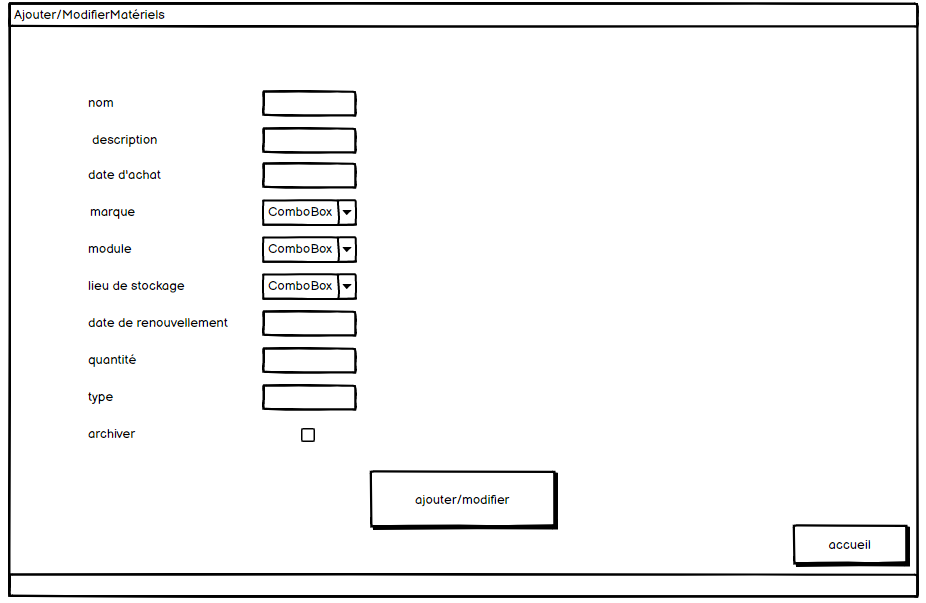
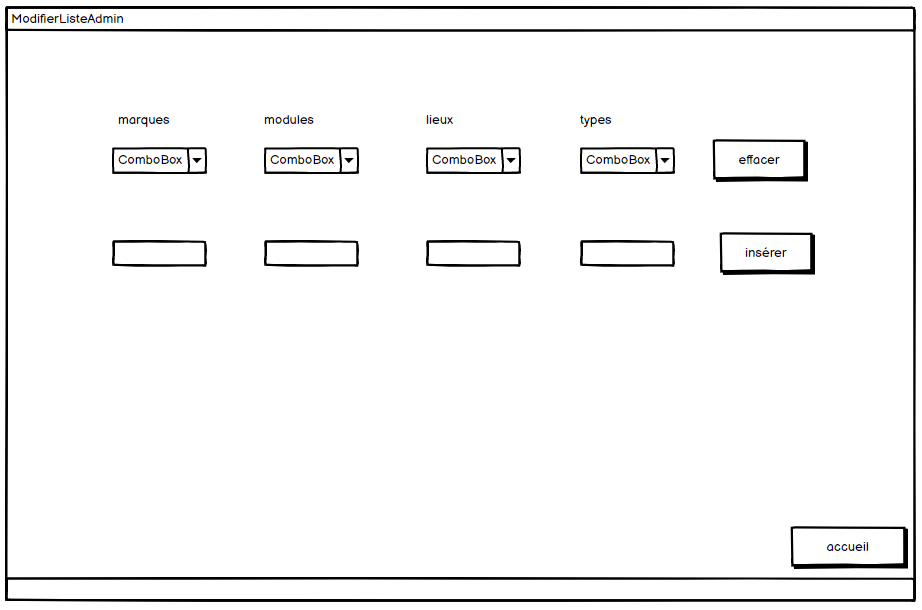
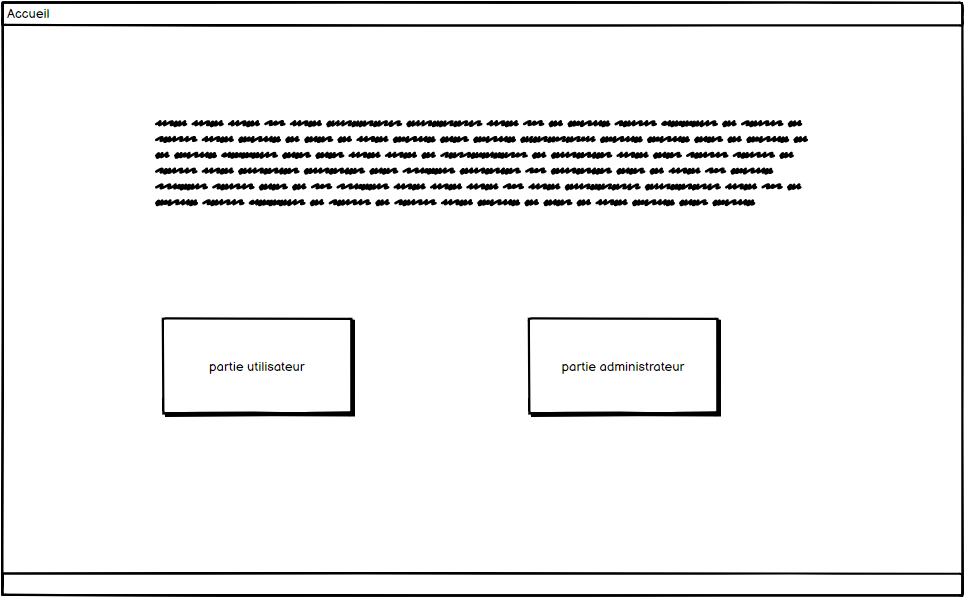
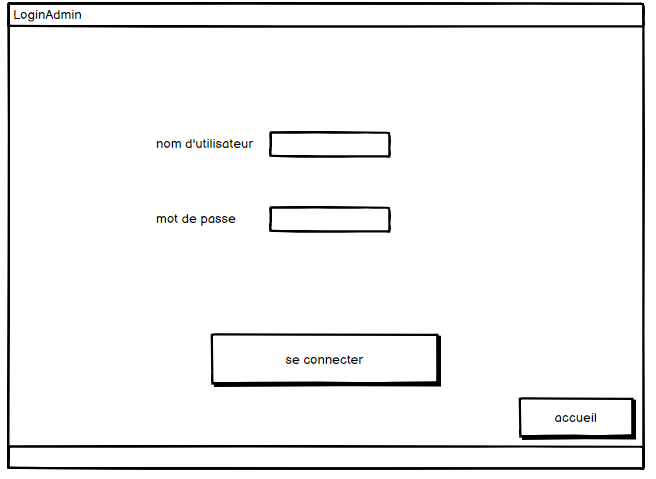
Voici un diagramme de Gantt pour représenter ma planification initiale.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **02.05.2023 - 30.05.2023** | | | | |
| 02.05.23 | 08.05.23 | 15.05.23 | 22.05.23 | 30.05.23 |
| Analyse |  |  |  |  |  |
| IHM |  |  |  |  |  |
| Code métier |  |  |  |  |  |
| Tests |  |  |  |  |  |
| Documentation final |  |  |  |  |  |

# Analyse / Conception

## Concept

Voici la maquette effectué pour voir concrètement l’apparence de l’application :



Voici les scénarios et Use case pour montrer les fonctionnalités de l’application, les scénarios seront utilisé pour effectuer les tests système car ils assurent que toutes les fonctionnalités marchent.





## Stratégie de test

La stratégie de teste sera faite en pyramide, en premier les tests unitaires pour chaque fonction importante, les tests d’intégrations seront fait quand plusieurs éléments interagissent entre eux pour s’assurer du fonctionnement et enfin les tests systèmes qui permettent de tester les scénarios écris dans la partie concept de ce projet. Une classe de test automatique sur une classe prévue pour ce projet devra être fait. Les données seront fictives mais elles seront inspirées des cours que j’ai effectué. Les tests seront effectués par moi-même.

## Planification



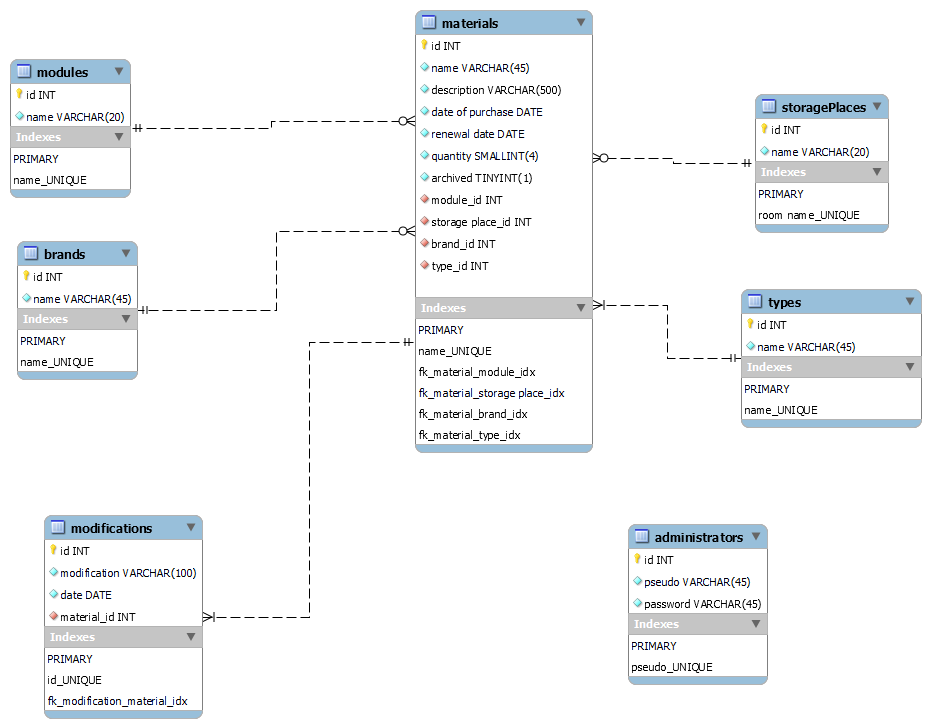
## Dossier de conception

Le matériel utilisé sera 1 PC en configuration standard CPNV avec accès à internet.

Le système d’exploitation sera Windows 10.

L’outil principale utilisé sera Microsoft Visual studio pour coder en C# ainsi que les applications de réalisation comme balsamiq, Excel, draw.io, MySQL Workbench, mariadb, HeidiSQL. Un outil de gestion de version git.

Voici le MCD de ce projet :

Voici Le MLD de ce logiciel

# Réalisation

## Dossier de réalisation

Le dossier est composé de deux parties, la première c’est le projet, code, donc qui contiennent Front qui contient l’affichage du projet, Classe qui contient les classes utiliser dans le code, Tests qui contient les tests automatiques effectuer et le .sln qui est la solution du projet. La deuxième partie concerne l’analyse/conception du projet avec comme contenue les User cases/scénarios, le rapport du TPI, la planification, le journal de travail, la maquette, le cahier des charges, le MCD/MLD, avec les scriptes de créations de base de données et d’ajout de données, et les captures d’écran.

## Description des tests effectués

### Tests unitaire/integration

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| numéro du test | description | date du test | résultat attendu | résultat |
| 1 |  | 08.05.23 | La fonction CheckAdmin retourne true si un admin similaire est trouvé dans la bdd | réussit |
| 2 |  | 08.05.23 | La fonction CheckAdmin retourne false si il n’y pas d’admin similaire dans la bdd | réussit |
| 3 |  | 08.05.23 | La fonction retourne la liste « storageplaces » disponible dans la bdd actuellement elle contient « SC-C111 » et « SC-C236 ». | réussit |
| 4 |  | 08.05.23 | La fonction InsertWord va ajouter un mot dans une liste de la bdd | réussit |
|  | La fonction DeleteWord va enlever un mot d’une liste dans la bdd |
| 5 |  | 08.05.23 | La fonction GetId va retourner l’id en int d’un élément d’une liste | réussit |
| 6 |  | 16.05.23 | La fonction InsertMaterial va insérer le matériel dans la base de données et la fonction GetMaterial va retourner une liste de matériel dont le dernier de la liste est le matériel qu’on vient de créer | réussit |
| 7 |  | 16.05.23 | La fonction UpdateMaterial va mettre à jour le matériel que l’on a modifier puis avec la fonction getMaterial l’on récupère le dernier matériel de la liste qui vient d’être modifier | réussit |
| 8 |  | 16.05.23 | La fonction InsertMessage qui va être lier avec le matériel « PCTEST » va insérer un message de modification dans la base de données, la fonction getMessages va rendre une liste de message et la dernière insérer va rendre le message inserer. | réussit |

### Tests système

|  |  |  |
| --- | --- | --- |
| **Numéro du scénario** | **date** | **résultat** |
| 1 | 17.05.2023 | réussit |
| 2 | 17.05.2023 | réussit |
| 3 | 17.05.2023 | réussit |
| 4 | 17.05.2023 | réussit |
| 5 | 17.05.2023 | réussit |
| 6 | 17.05.2023 | réussit |
| 7 | 17.05.2023 | réussit |
| 8 | 17.05.2023 | réussit |

## Erreurs restantes

*S'il reste encore des erreurs:*

* *Description détaillée*
* *Conséquences sur l'utilisation du produit*
* *Actions envisagées ou possibles*

## Liste des documents fournis

Voici la liste de tous les documents fournis qui sont disponible dans le dépôt github <https://github.com/Tristan1403/GestionInventaireInformatiqueTPI.git> :

Le rapport de TPi

La documentation initiale avec toutes la partie analyse, conception

Un manuel d’installation et d’utilisation

Les scriptes pour créer la base de données et y insérer des données

L’application avec un .exe pour l’ouvrir

# Conclusions

*Développez en tous cas les points suivants:*

* *Objectifs atteints / non-atteints*
* *Points positifs / négatifs*
* *Difficultés particulières*
* *Suites possibles pour le projet (évolutions & améliorations)*

# Annexes

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

## Sources – Bibliographie

*Liste des livres utilisés (Titre, auteur, date), des sites Internet (URL) consultés, des articles (Revue, date, titre, auteur)… Et de toutes les aides externes (noms)*

## Journal de travail

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Durée** | **Activité** | **Remarques** |
|  |  |  |  |
|  |  |  |  |

## 

## Manuel d'Installation

## Manuel d'Utilisation

## Archives du projet

<https://www.ionos.fr/digitalguide/sites-internet/developpement-web/modele-en-cascade/#:~:text=Le%20mod%C3%A8le%20en%20cascade%20reposant,%2C%20impl%C3%A9mentation%2C%20test%20et%20exploitation>. 03.05.2023

[https://stackoverflow.com/questions/3061042/how-do-i-set-combobox-read-only-or-user-cannot-write-in-a-combo-box-only-can-sel 08.05.2023](https://stackoverflow.com/questions/3061042/how-do-i-set-combobox-read-only-or-user-cannot-write-in-a-combo-box-only-can-sel%2008.05.2023)

[https://www.codeproject.com/Questions/646695/how-to-reload-form-or-refresh-form-in-csharp-net 08.05.2023](https://www.codeproject.com/Questions/646695/how-to-reload-form-or-refresh-form-in-csharp-net%2008.05.2023)

<https://stackoverflow.com/questions/29814305/how-to-load-list-data-into-datagridview-in-c-sharp> 09.05.2023

<https://learn.microsoft.com/en-us/dotnet/api/system.windows.forms.datagridview.rowheadermousedoubleclick?view=windowsdesktop-7.0&viewFallbackFrom=net-5.0> 09.05.2023

[https://learn.microsoft.com/en-us/answers/questions/372644/pass-from-data-grid-to-textbox-in-c 09.05.2023](https://learn.microsoft.com/en-us/answers/questions/372644/pass-from-data-grid-to-textbox-in-c%2009.05.2023)

messageInput [https://stackoverflow.com/questions/10797774/messagebox-with-input-field 09.05.2023](https://stackoverflow.com/questions/10797774/messagebox-with-input-field%2009.05.2023)

pour éviter une exception quand la premiere ligne est rendu invisible : <https://www.developpez.net/forums/d2100367/dotnet/langages/csharp/rendre-invisible-ligne-datagridview/> 11.05.2023

<https://www.developpez.net/forums/d511035/dotnet/langages/csharp/retour-chariot-string/> 15.05.2023

enlever les doublons dans une liste <https://www.tutorialspoint.com/remove-duplicates-from-a-list-in-chash#:~:text=Use%20the%20Distinct()%20method,from%20a%20list%20in%20C%23>. 15.05.2023