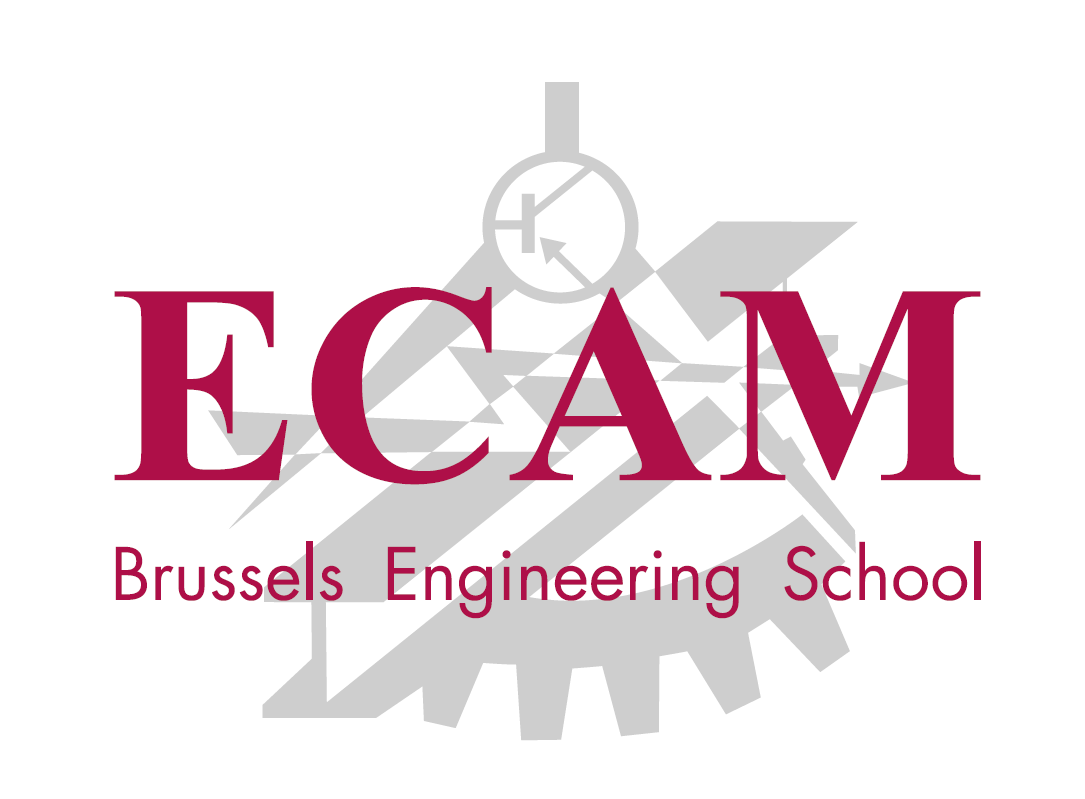
rapport du Projet informatique



Auteurs : KUE Guy & KOLAWOLE Abdoulaye

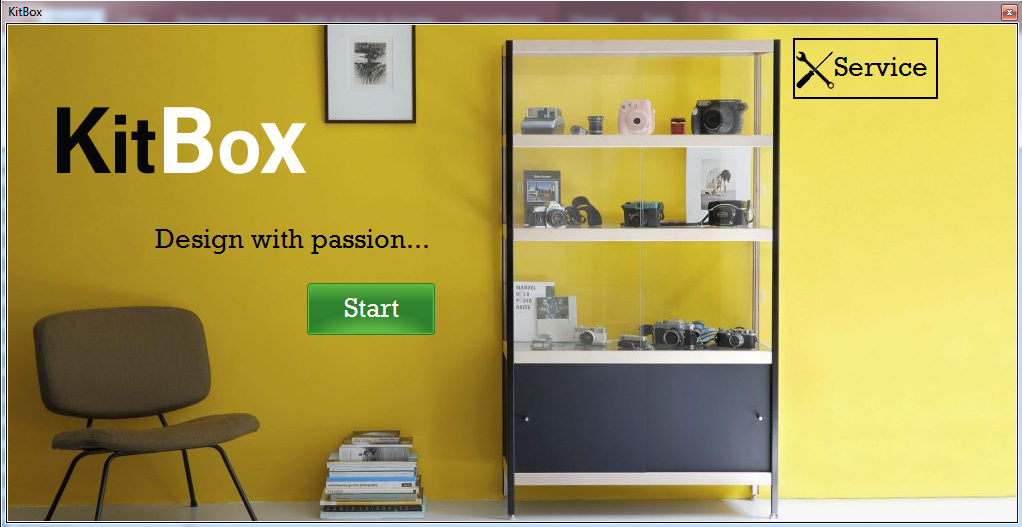


Table Des matières

[Introduction 2](#_Toc517161469)

[1. Fonctionnement 2](#_Toc517161470)

[1.1. Fonctionnement d’une commande standard 2](#_Toc517161471)

[1.2. Commande non finalisée 6](#_Toc517161472)

[1.3. Interface administrateur 8](#_Toc517161473)

[1.3.1. Recherche dans base de données 9](#_Toc517161474)

[1.3.2. Chercher une commande 10](#_Toc517161475)

[2. Base de Donnée 11](#_Toc517161476)

[3. Analyse et conception 12](#_Toc517161477)

[3.1. Diagramme des cas d’utilisation métier 12](#_Toc517161478)

[3.2. Diagramme d’activité métier 13](#_Toc517161479)

[3.3. Glossaire Anglais 14](#_Toc517161480)

[3.4. Diagramme des cas d’utilisation système 15](#_Toc517161481)

[3.5. Diagramme d’activité système 15](#_Toc517161482)

[3.6. Diagramme de séquence 17](#_Toc517161483)

[3.7. Diagramme de classe 19](#_Toc517161484)

[Conclusion 20](#_Toc517161485)

# Introduction

L’informatique peut se définir comme étant la science du traitement automatique et rationnel de l’information par le biais de logiciel ou d’application.

Ainsi dans le cadre de son informatisation, la société KitBox a fait appel à nous pour la conception d’une application pour l’exécution de ses commandes clients et de sa gestion de stock. Ce travail est alors l’objet de ce rapport.

Pour présenter l’application que nous avons développée, nous allons dans un premier temps présenter son fonctionnement à travers des captures d’écrans.

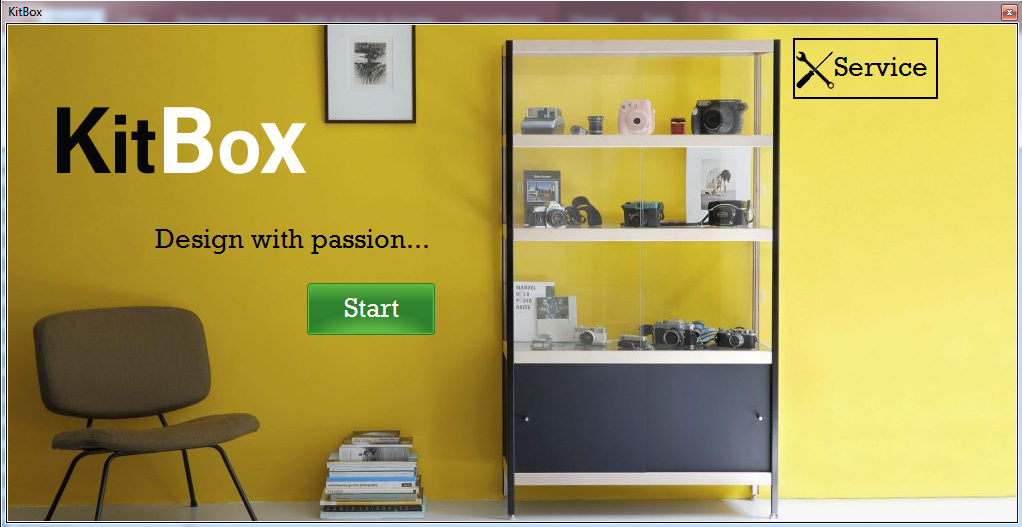
Par la suite, nous allons parler de sa conception de par les différentes analyses explicitées avec les différents diagrammes créés et enfin, nous allons terminer par une conclusion.

# Fonctionnement

## Fonctionnement d’une commande standard

Pour cet exemple nous allons effectuer une commande composée de 3 casiers, les captures d’écrans qui vont suivre représentent les différentes étapes de cet achat.

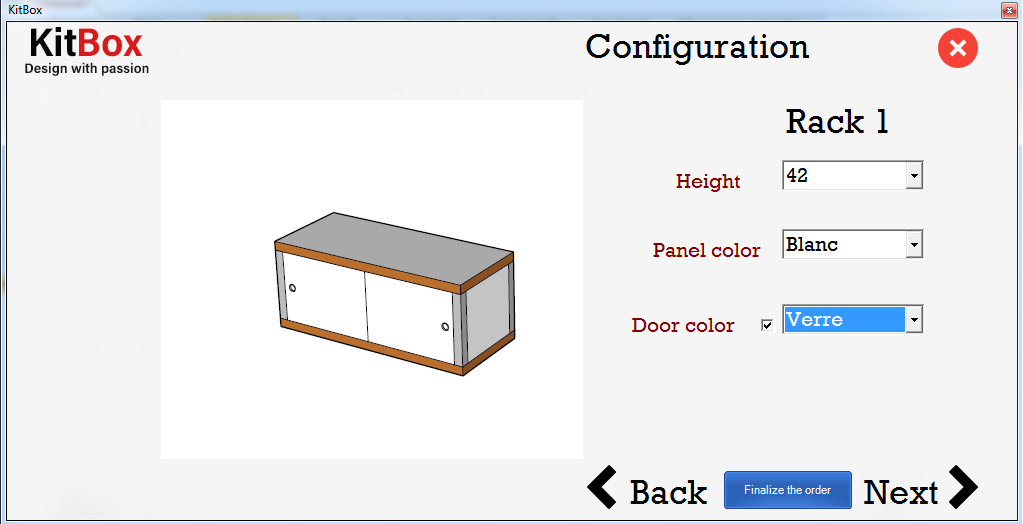
Démarrage de l’application :

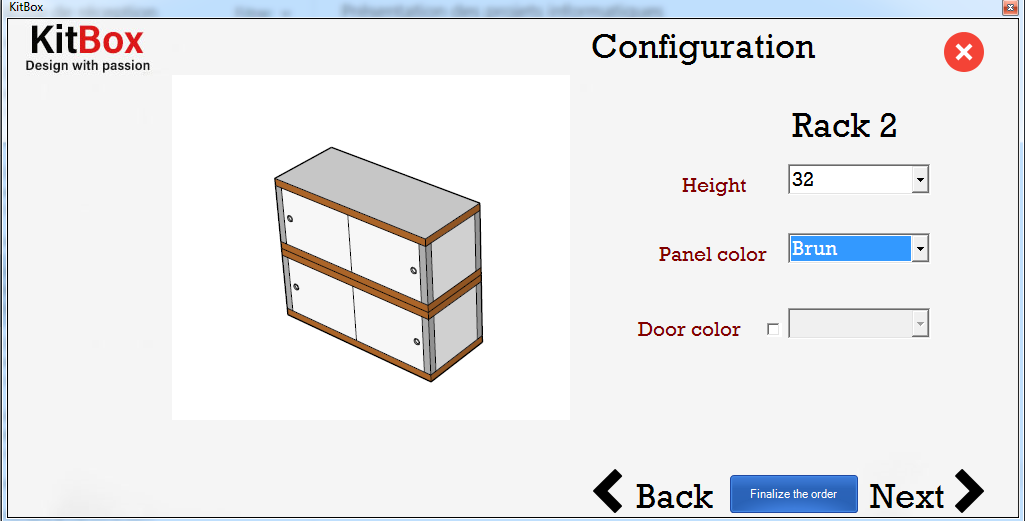


Choix des dimensions de base du casier et de la couleur de la cornière :

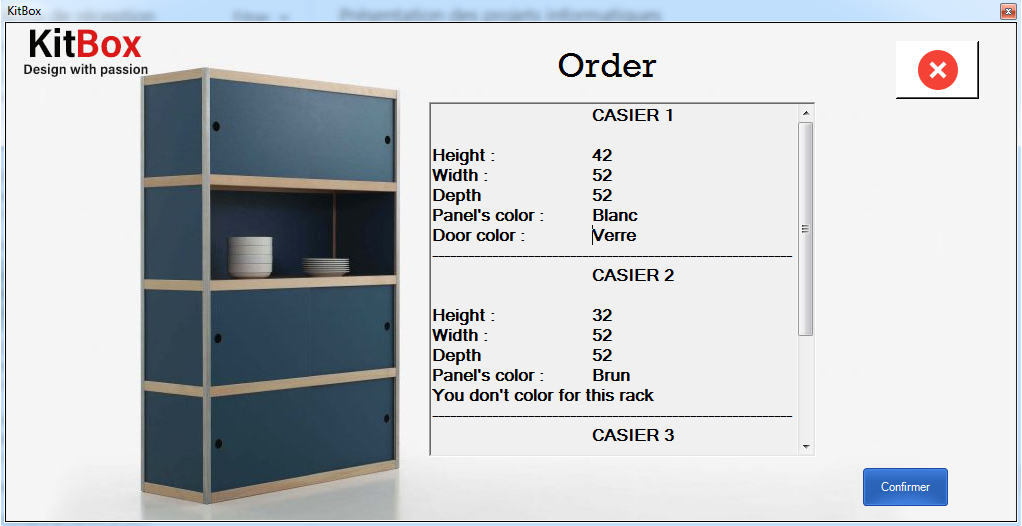


Choix des paramètres des différents casiers :

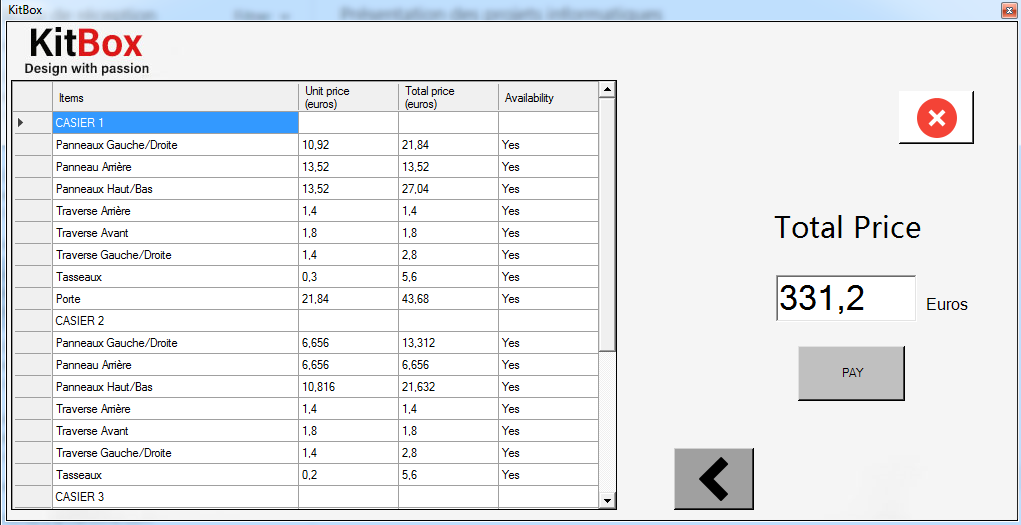


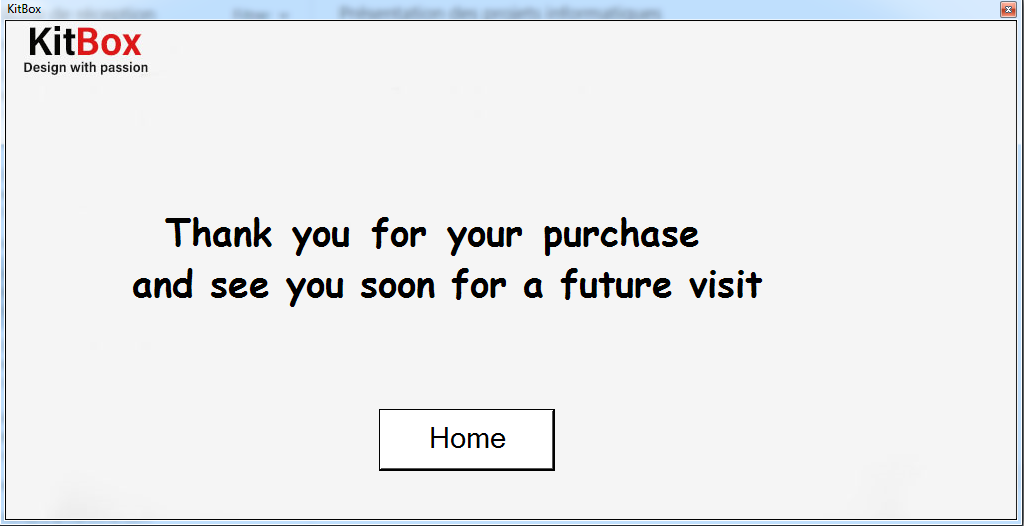


Visualisation de la commande :



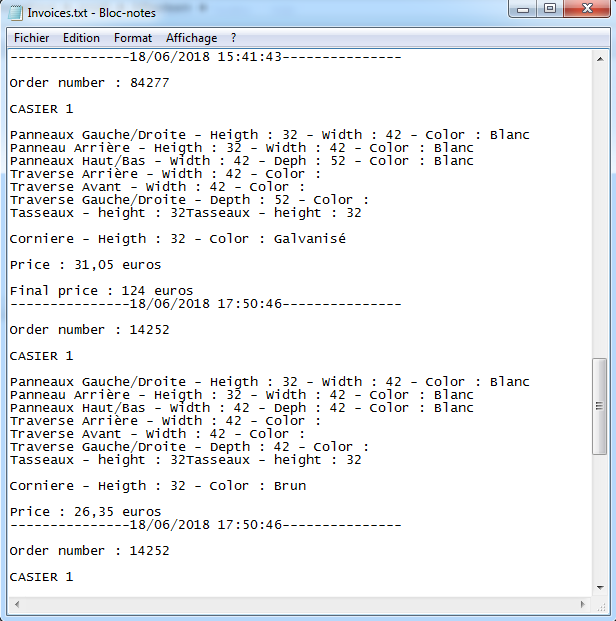
Facture à payer :





Après validation de la commande, les différents éléments sont déduits du stock avec le bon nombre entrant dans la constitution d’un casier. Parallèlement, une copie de la facture est enregistrée dans un fichier texte afin de pouvoir effectuer une comptabilité des ventes effectuées.

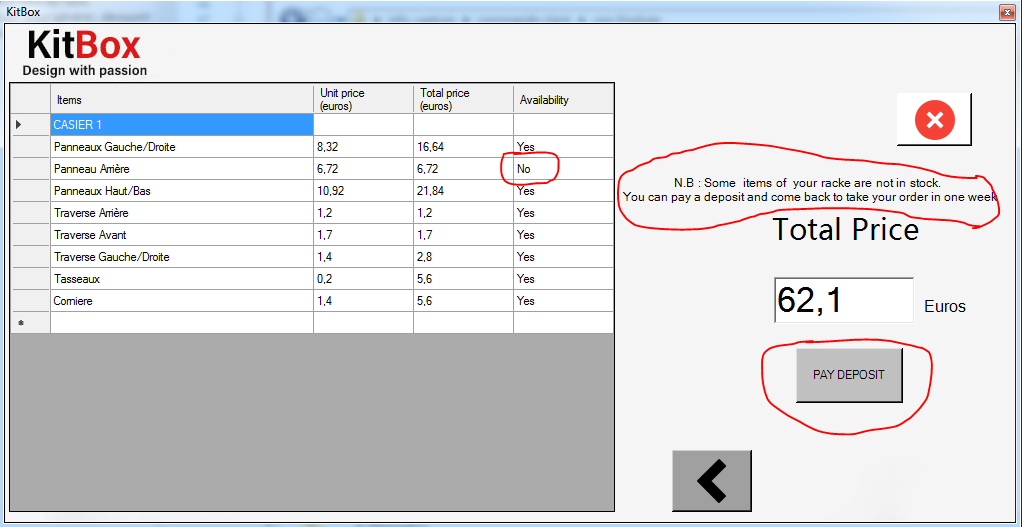
Chaque commande est enregistrée avec un numéro de commande unique ainsi que l’heure et la date de validation de la commande.



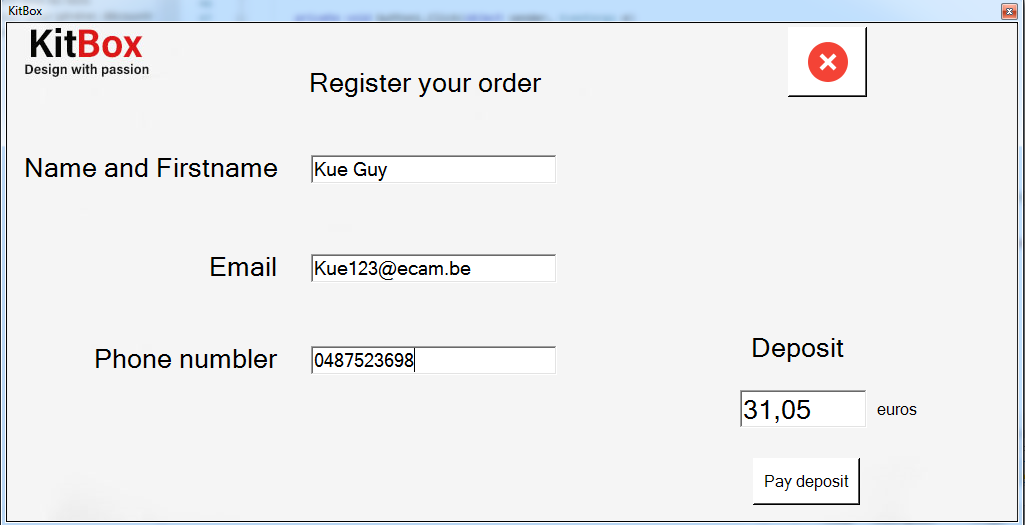
## Commande non finalisée

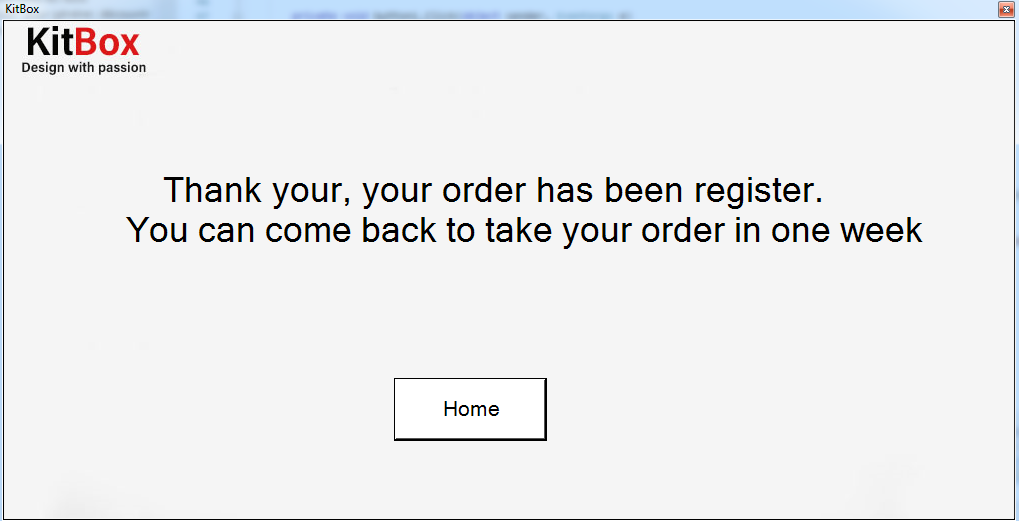
Dans cet exemple une ou plusieurs pièces sont manquantes :

Après la visualisation de la commande, on arrive à cette facture où on peut procéder à une réservation et au payement d’un acompte.



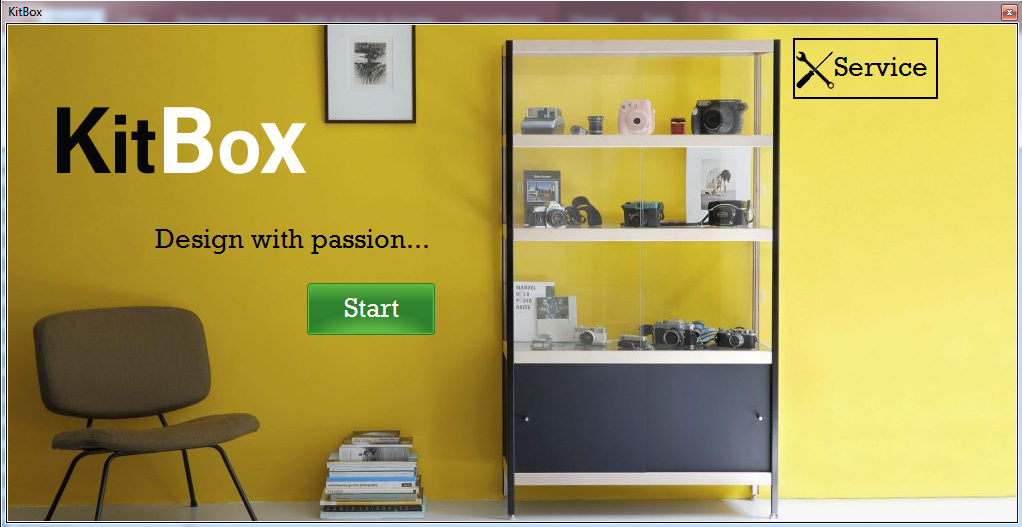
Information personnelle pour la réservation de commande :

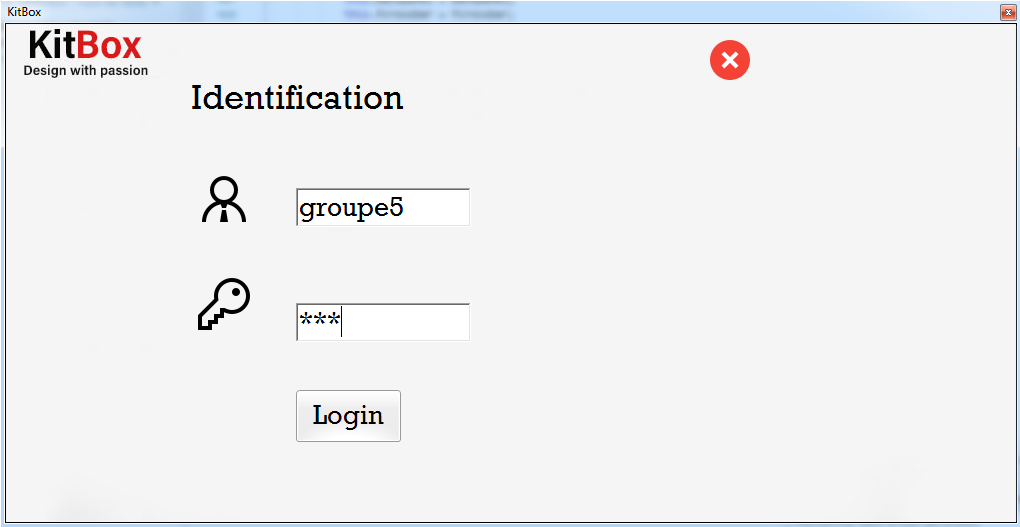




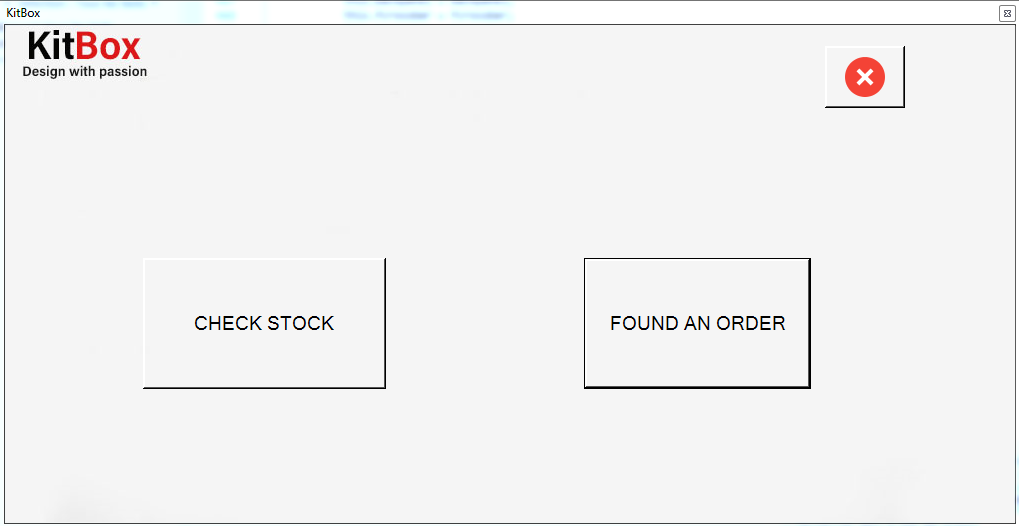
## Interface administrateur

On accède à la partie administrateur en cliquant sur le bouton « Service » :



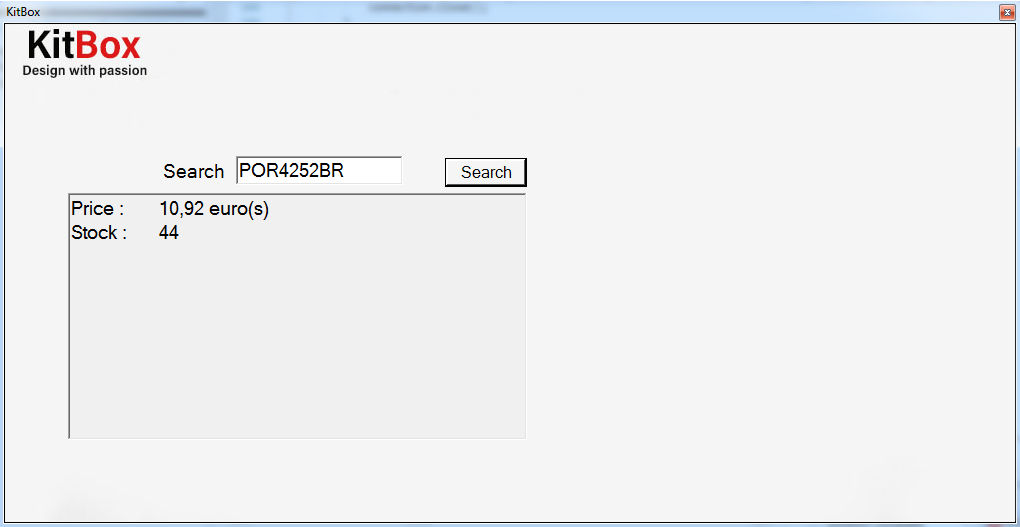


Choix de la tâche à effectuer :



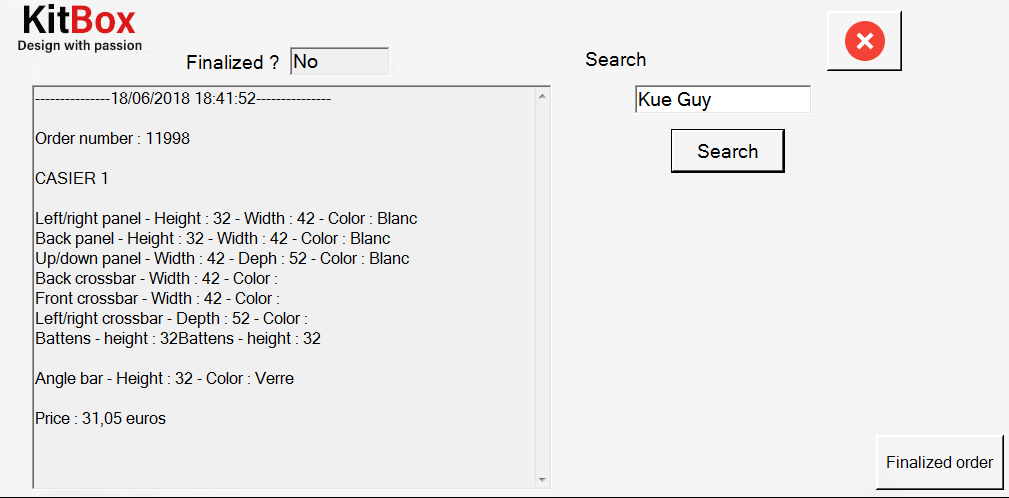
### Recherche dans base de données

En entrant le code de l’article dans la base de données on retrouve son prix et sa quantité en stock :



### Chercher une commande

Il suffit de rentrer le nom et le prénom ou le numéro de la commande, et on sait aussi avoir l’état de la commande, si elle est finalisée ou non.

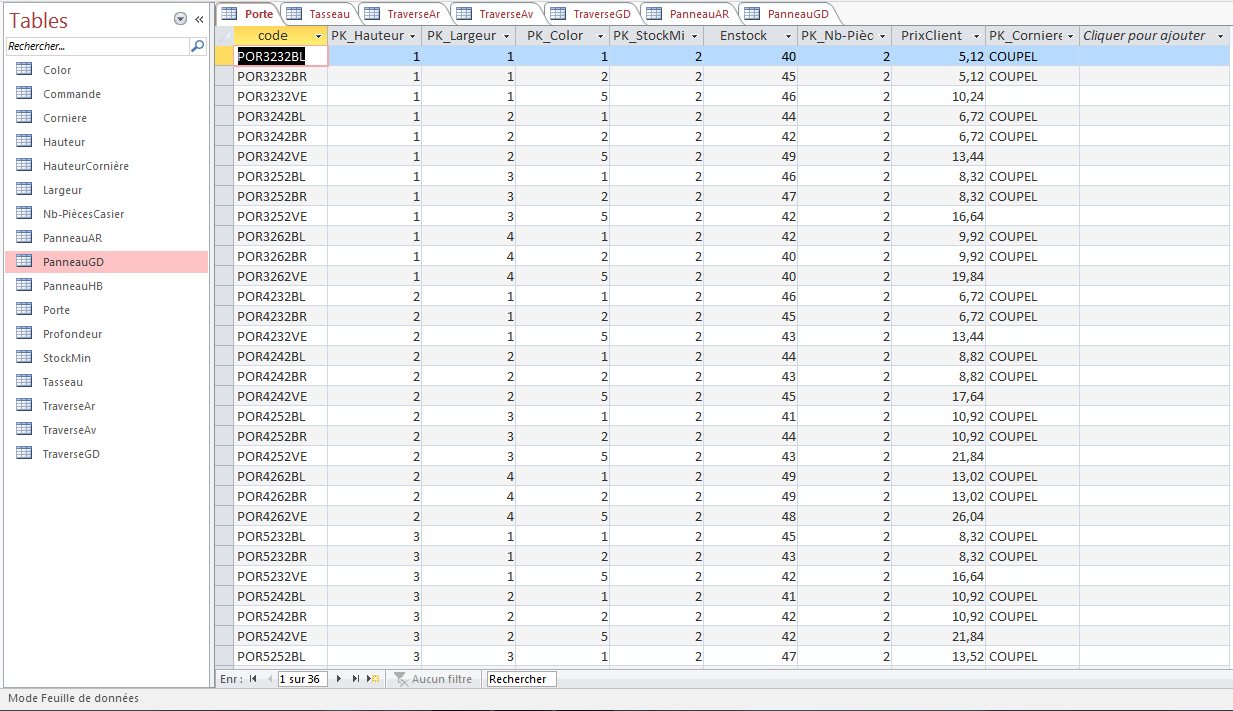


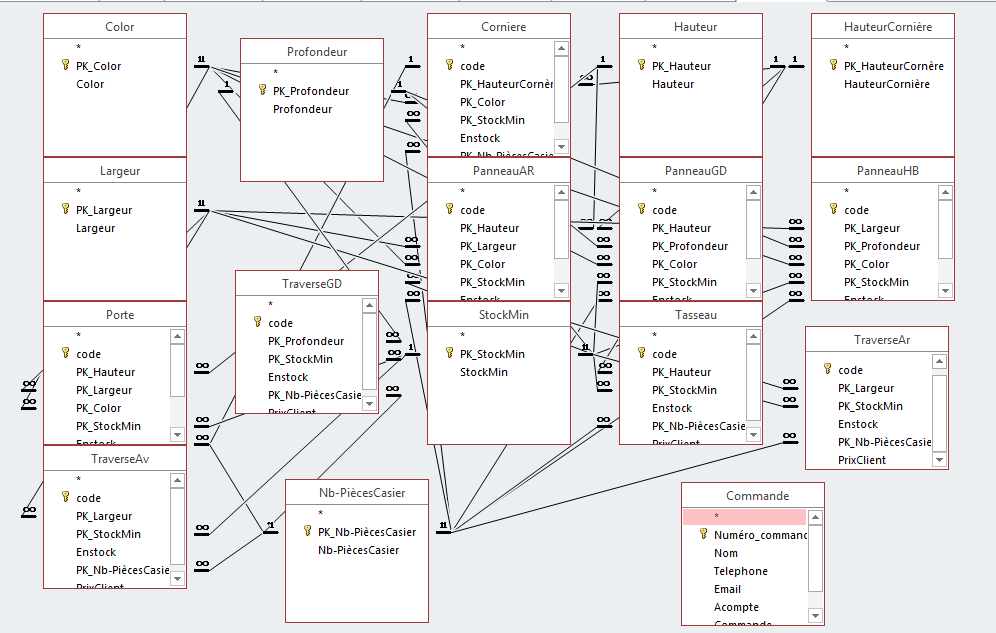
# Base de Donnée

La base de données que nous avons utilisée est une base de données accès, créé à partir du fichier Excel utilisé par le passer et qui faisait office de base de données. Nous avons classé et restructuré ces données dans des tables afin de les utiliser d’une façon plus efficace.

Pour se faire nous avons créé une table pour chaque type d’élément de la base de données, que nous avons mis en relation par utilisation de clé primaires et étrangères.

Les clés primaires de chaque élément de la base de données étant son code ceci dans le but de faciliter les recherches en base de données.





# Analyse et conception

Pour pouvoir mettre sur pied notre application, nous avons eu besoin d’analyser le métier, c’est-à-dire de déterminer les différents cas d’utilisation ainsi que les différents acteurs entrant en jeux dans le métier, mais également établir la séquence du processus complet, de la visite du client à sa sortie avec sa commande. Pour nous aider nous avons mis sur pied différents diagrammes.

## Diagramme des cas d’utilisation métier



Ce diagramme nous a permis de mettre en lumière les différents cas d’utilisation en lien avec les acteurs dans le métier

## Diagramme d’activité métier



Ce diagramme d’activité métier présente les différents flux possibles lors des interactions entre les acteurs du métier.

## Glossaire Anglais



Ce glossaire nous donne la traduction en anglais des différents éléments de la constitution du casier afin d’avoir un code avec le bon nom de variable utilisées.

## Diagramme des cas d’utilisation système



Ce diagramme nous a permis de voir les différentes fonctionnalités à implémentent dans notre application. De plus, nous pouvons y déduire les accessibilités des acteurs aux différentes fonctionnalités de notre application.

## Diagramme d’activité système

Le Diagramme d’activité système nous permet de définir les inputs et les outputs de notre système ainsi que le traitement des données à effectuer afin d’obtenir un système le plus automatisé possible.

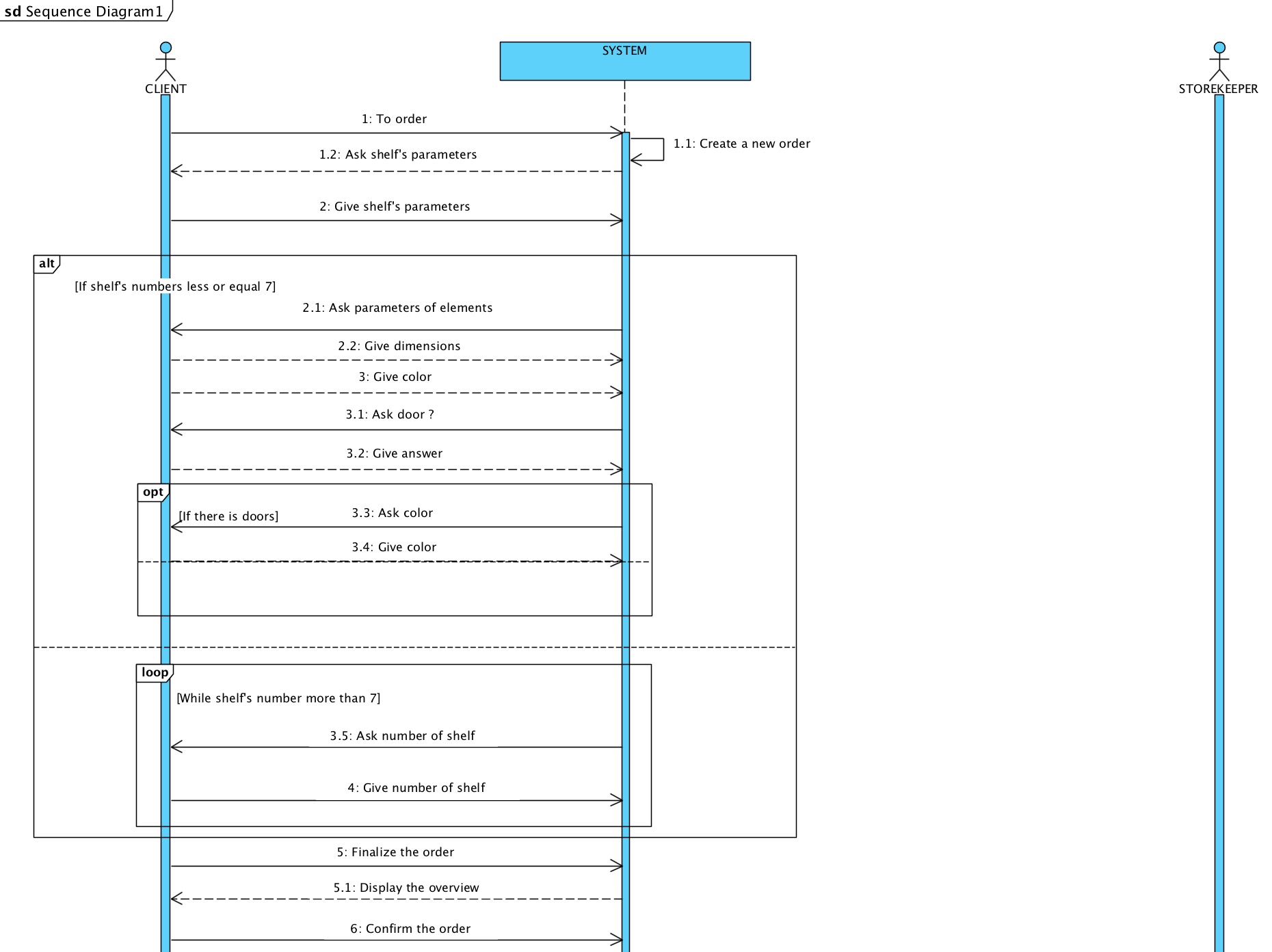
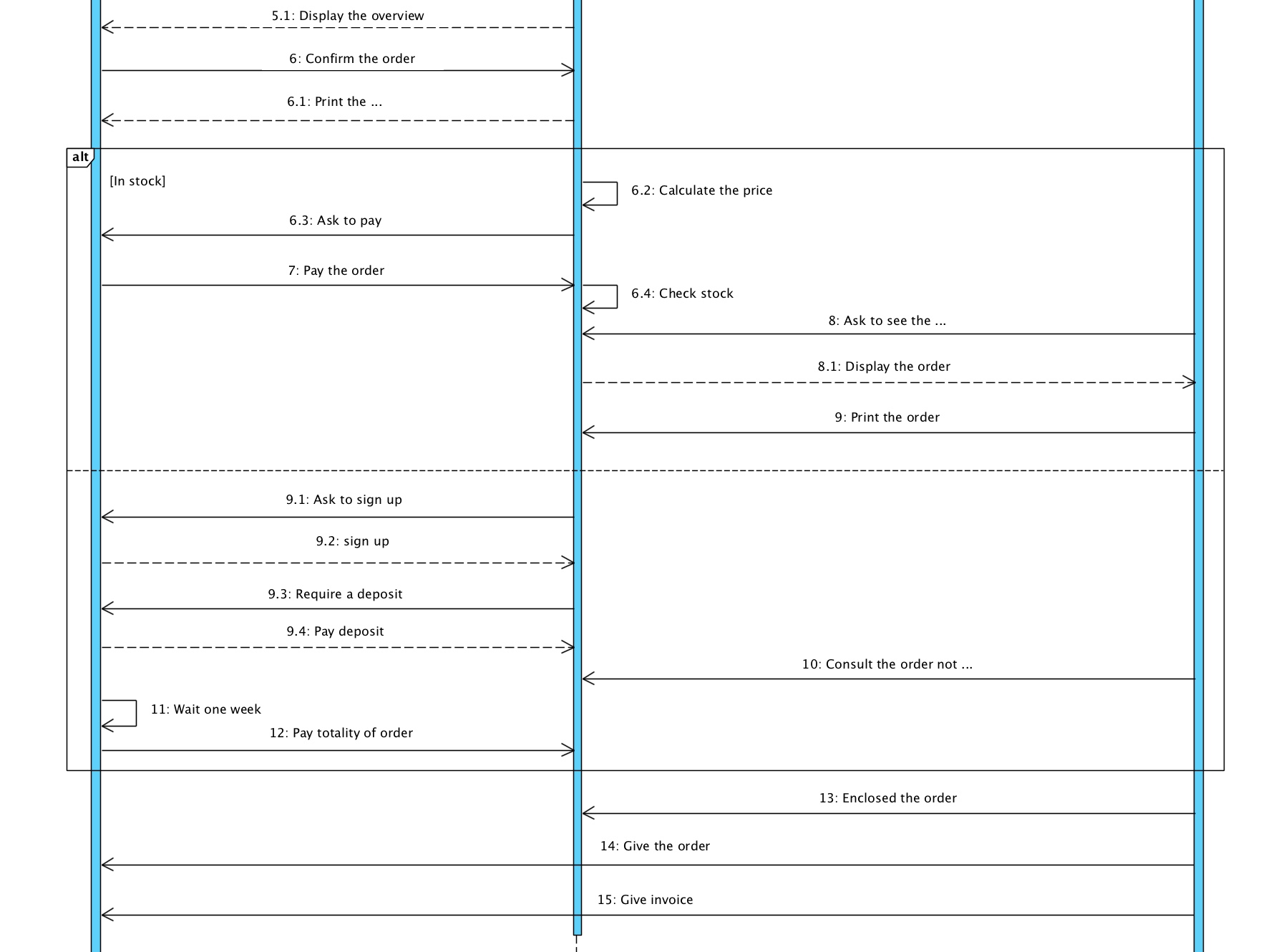
Pour le rendre plus explicite nous avons présenté plus en détail la partie « remplir bon de commande » dans un autre diagramme d’activité appelé Order activity Diagram.





loop

## Diagramme de séquence



## Diagramme de classe

Le diagramme de classe nous montre les liens qui existent entre nos différentes classes :

La classe « *Item »* est classe abstraite de notre application à partie de laquelle nous allons créer les autres éléments, ceci dans le but de pouvoir ouvrir notre application à une évolution plus facile grâce à l’Héritage. Cette classe est composé d’un objet de type « dimensions » qui renferme les dimensions de l’objet qui héritera de l’Item.

La classe « Rack » est une classe qui utilise tous les objets héritant de « Item ». Les objets de type « Rack » qui seront créés serviront à créer l’armoire simplement en les regroupant dans une liste de maximum 7 éléments qui définissent le nombre maximal de casier d’une étagère.

La méthode « Code() » est une méthode centrale de nos classes, elle renvoie le code unique à chaque élément et nous permet ainsi de retrouver cet objet dans la base de données.

# Conclusion

Pour conclure, notre application répond au cahier des charges imposé par le client qui est :

* L’encodage de la commande par le client ou le magasinier
* La complétion automatique des bonnes pièces d’un casier
* La visualisation de la commande effectuée ainsi que leur disponibilité en stock
* La suppression des éléments en stock lors de la validation de l’achat par le client
* Générer la facture du client ainsi que la liste de pièce
* L’enregistrement des commandes non clôturées
* Retrouver la commande d’un client sur base de son nom ou du numéro de sa commande

Pour ce qui est de la gestion de stock, notre groupe en a été dispensé compte tenue de notre nom de membre limité.

Notre application pourrait être améliorée notamment au niveau de l’interface de commande en montrant en temps réel les ajustements fait à un casier

Une interface plus dynamique pour captiver l’utilisateur.