# Expressions des besoins :

Le client demande la réalisation d’une structure informatique pouvant gérer au mieux son activité de production, et cela au travers d’une interface graphique la plus agréable possible.

Le client n’est d’autre qu’une société de production d’articles ou d’ensemble composé de plusieurs articles. Il nous est demandé de pouvoir géré toutes les actions pouvant apparaitre dans sa chaine de production.

Il nous faudra donc gérer, stocker, modifier et mettre à jour toute les données nécessaires au traitement d’une commande, mais aussi à son aspect technique et comptable. Pour résumer, toute le fonctionnement proprement dit de l’entreprise doit être gérer par notre programme. De la production à la livraison en passant par le stockage et la gestion.

D’un point de vue technique, il nous faudra :

* Mettre en place une décomposition des articles car certains pourrait être un ensemble de sous-articles
* Détailler chaque article avec toutes les informations nécessaire à son traitement
* Mettre en place un système de classification des articles ainsi que de touts les composants de cette entreprise. Et cela de la façon la plus optimum possible pour facilité la rapidité du traitement des articles.
* Mettre en place la logique des gammes d’opérations de l’entreprise pour gérer un assemblage ou une production.

Cela nécessitera l’utilisation de méthode programmations afin de pouvoir s’approcher au mieux de la réalité de l’entreprise et du monde du travail. Nous devrons utiliser des méthodes de calcul afin de gérer au mieux les délais de livraison et de production, dans le cas d’un environnement déterministe mais aussi au sein d’un environnement stochastique.

Pour tout ceci, il nous faudra respecter les contraintes techniques imposé par le client :

* Langage de développement : JAVA
* Persistance des données : Sérialisation des objets proposée par Java.

Cette technique de sérialisation doit être masquée par une interface (au sens UML) afin de pouvoir changer simplement de méthode de persistance de données.

* Analyse et conception en UML. En particulier le pattern MVC est impératif.
* Dans un souci de maintenance et d'évolution possible, les objets doivent communiquer entre eux à travers des interfaces
* A chaque lancement du programme, l'ensemble des données persistantes doivent être chargées en mémoire vive. La sauvegarde des données se fait automatiquement à la sortie du programme ou à chaque fois que l'utilisateur le demande. Seules les données modifiées, crées ou supprimées doivent être mises à jour lors d'une sauvegarde.
* Les seules actions possibles sur les classes seront celles permettant de faire le CRUD (Create, Read, Update, Delete)