

# Document de Conception - Projet Fil Rouge

Alexandre  
STEFANOS  
Hanafi  
EL-MOUGHARTI  
Constant  
COINTE

## Introduction

Dans le cadre de la pandémie actuelle, un restaurant traditionnel a fait appel à nous pour automatiser différentes étapes de sa chaîne de production et du service afin de réduire les contacts humains et de pouvoir laisser ouvert le restaurant dans toutes les circonstances.

Dans le langage UML, il existe trois catégories de diagramme (de structure, de comportement et d'interaction) qui peuvent définir le projet sous différents points de vue. Nous avons décidé de réaliser un diagramme de chaque catégorie dans le but d'avoir une bonne représentation de notre projet. Nous utiliserons, dans un premier temps, un diagramme de cas d'utilisation pour une meilleure visualisation du comportement de notre projet, puis un diagramme de séquence pour décrire les interactions entre les objets et enfin un diagramme de classe pour décrire la structure.

Pour les documents cités ci-dessus, nous vous invitons à consulter l'annexe.

## Description des interfaces entre les différentes tâches du logiciel

Notre interface principale est une base de donnée qui permet de gérer une grande partie de l'automatisation. Elle permet de gérer les stocks et les commandes de marchandises, le nombre et la nature des commandes.

L'interface permettant au client de commander est le site internet. Ce site est accessible grâce aux QR Codes disposés sur les tables du restaurant.

Pour les règlements après un repas, l'interface est un terminal de paiement bancaire.

Enfin, les repas sont acheminés à l'aide d'un rail qui remplace les serveurs traditionnels. C'est donc notre interface permettant de gérer le service des repas.

## Description détaillée

- Pour chaque module :
  - Gestion des stocks de marchandises
    - L'objectif est d'avoir une gestion optimale des marchandises en commandant régulièrement les produits frais. Concernant les produits ne nécessitant pas une consommation rapide. Lorsqu'un produit descend sous un certain seuil, une commande automatique gérée par la base de donnée est effectuée.
    - La gestion des stocks est reliée au module de commande des clients. À chaque commande d'un produit, il est décrementé dans le compte des stocks. Dans le cas où un produit n'est plus disponible, on ne peut plus faire une commande contenant ce produit.
    - Les fonctions visibles par les autres modules sont celles permettant de suivre l'évolution des stocks.
  - Commande des clients

- L'objectif est d'avoir un service automatique garantissant la continuité entre les clients et le personnel en cuisine. Pour cela, nous avons pensé à donner accès à un site internet pour que les clients puissent faire leurs commandes.
  - La commande des clients est reliée à la gestion des stocks comme expliqué précédemment. C'est aussi relié au paiement des clients car le prix de la commande est déterminé en fonction de cette dernière. Ce module est aussi relié au service des clients comme expliqué dans le module service des clients.
  - Les fonctions visibles par les autres modules sont celles permettant d'accéder à la composition de la commande et à son prix.
- Service des clients
  - L'objectif est d'avoir un service automatique provenant de la cuisine jusqu'au clients par l'intermédiaire d'un rail automatique.
  - Le service des clients est relié au module commande des clients. Le client peut voir sur le site internet le temps d'attente restant jusqu'à l'arrivée de sa commande.
  - Les fonctions visibles sont celles permettant d'accéder au temps de commande.
- Paiement des clients
  - L'objectif est de permettre au client de payer en utilisant un terminal de paiement.
  - Ce module est relié à la commande des clients comme nous l'avons expliqué précédemment.
  - Les fonctions visibles sont celles permettant de payer.
- Pour chaque classe :
  - Restaurant
    - Classe principale du projet englobant les autres classes
    - Elle est reliée à toutes les classes (hors livreur)
    - Attributs : String nom, int nbClients
    - Méthodes : String getNom(), int getNbClients() et static void main(String args[])
  - Humain
    - Classe abstraite permettant de créer des sous-classes
    - Elle est reliée à la classe cuisinier, client et livreur
    - Attributs : String nom, String prenom et String metier
    - Méthodes : String getNom(), String getPrenom() et abstract String getMetier()
  - Cuisinier
    - Classe concrète représentant un cuisinier. Ce dernier permet de préparer les commandes et de les envoyer aux clients par l'intermédiaire des rails automatiques.
    - Elle est reliée à la classe cuisine, au rail automatique et à l'écran
    - Attributs : String commande
    - Méthodes : void regarderCommandeEcran(), void prepareCommande(), void mettreSurRail()
  - Client
    - Classe concrète et statique qui hérite d'Humain représentant un Humain qui passe une commande

- Elle est reliée au site Internet et au terminal de paiement
  - Méthodes : void envoieCommande() et void payer()
- Livreur
  - Classe concrète qui hérite d'Humain représentant un Humain qui livre les marchandises
  - Elle est reliée au restaurant
  - Attribut : String nom
  - Méthode : void livrerMarchandises()
- Matériel Informatique
  - Classe abstraite représentant un ensemble de matériel informatique
  - Elle est reliée à aucune classe
  - Attribut : String nom
  - Méthode : String getNom()
- Ecran
  - Classe concrète qui hérite de matériel informatique permettant d'afficher les commandes sur l'écran des cuisiniers
  - Elle est reliée au cuisinier et à la base de donnée
  - Attributs : String marque et int numEcran
  - Méthodes : void afficherRepas(), void afficherNbCommandeRestante() et void afficherNbCouverts()
- Site Internet
  - Classe concrète qui gère le site Internet et les commandes des clients
  - Elle est reliée au client et au cuisinier
  - Attribut : int numCommande
  - Méthodes : int getNumCommande(), void afficherSite() et void afficherTempsAttente()
- Rail Automatique
  - Classe concrète qui hérite de matériel informatique représentant le rail automatique qui permet de servir le client
  - Elle est reliée au cuisinier et au client
  - Attributs : int num et String marque
  - Méthodes : void avancer(), void reculer(), void trouverTable() et void déposerPlat()
- QR Code
  - Classe concrète qui hérite de matériel infomartique permettant de gérer le QR Code et de permettre au client d'accéder au site Internet
  - Elle est reliée au site internet et au client
  - Attributs : int numTable et String lien
  - Méthodes : int getNumTable(), String getLien() et String envoyerLien()
- Base de donnée
  - Classe concrète qui hérite de matériel informatique permettant la gestion de la base de donnée comme par exemple la gestion des stocks
  - Elle est reliée au site Internet et au cuisinier
  - Attributs : Fichier menu et String marchandise
  - Méthodes : void actualiserMarchandise() et void envoyerMenu()
- Terminal de Paiement
  - Classe concrète qui hérite de matériel informatique permettant la gestion des paiements des clients
  - Elle est reliée au client

- Attribut : String banque et int paiement
- Méthodes : String getBanque(), int recupPaiement() et int envoiePaiement()