

Projet Programmation Système A3

*« Rapport de Gestion de Projet »*

Table des matières

[RAPPEL DU BESOIN 3](#_Toc532777705)

[I / Rappel 3](#_Toc532777706)

[II / Cahier des charges 3](#_Toc532777707)

[A / Fonctionnel 3](#_Toc532777708)

[B / Technique 4](#_Toc532777709)

[PRESENTATION DU GROUPE 4](#_Toc532777710)

[Dimitri Libessart : 4](file:///C:\Users\Dimitri\Documents\A3\Projet%20C%23\Rapport%20Gestion%20de%20Projet.docx#_Toc532777711)

[Théo Vanengelandt : 4](file:///C:\Users\Dimitri\Documents\A3\Projet%20C%23\Rapport%20Gestion%20de%20Projet.docx#_Toc532777712)

[Alexis Caron : 4](file:///C:\Users\Dimitri\Documents\A3\Projet%20C%23\Rapport%20Gestion%20de%20Projet.docx#_Toc532777713)

[Corentin Evrard : 4](file:///C:\Users\Dimitri\Documents\A3\Projet%20C%23\Rapport%20Gestion%20de%20Projet.docx#_Toc532777714)

[III / Planning Prévisionnel 5](#_Toc532777715)

[IV / Choix techniques et Réalisations 6](#_Toc532777716)

[V / Planning Réel de Projet 7](#_Toc532777717)

[VI / Problèmes rencontrés 8](#_Toc532777718)

[VII / MCD 9](#_Toc532777719)

[VIII / Diagramme de classes 11](#_Toc532777720)

[IX / Conclusion 12](#_Toc532777721)

# RAPPEL DU BESOIN

## I / Rappel

Une grande chaîne internationale de restaurants souhaite s’équiper d’une nouvelle application informatique pour améliorer l’accueil du public, le remplissage des salles, la gestion des réservations et l’organisation du travail en cuisine.

Le directeur du restaurant a décidé d’investir dans le développement d’une application de gestion et supervision du fonctionnement de son restaurant (salle de restauration et cuisine). Il fait appel à vous, étudiants ingénieurs CESI en A3, pour l’aider sur ce projet.

Vous devez analyser les résultats de la simulation et faire des propositions d’amélioration. Vos propositions seront dans l’obligation de reposer sur des éléments mesurables pour mettre en évidence, sans ambiguïté, les gains que vos propositions suggèrent et donc doivent être exprimées en unité de mesure (temps, pourcentage, argent, etc…).

## II / Cahier des charges

Pour la réalisation de ce projet, nous devons respecter un cahier des charges fonctionnel et technique.

Nous avons énuméré les différents points à aborder.

### A / Fonctionnel

* Diagrammes de cas d’utilisation
* Diagrammes d’activité de chaque poste de travail.
* L’application doit permettre de visualiser l’état de chaque personne (salariés ou clients) et de chaque objet modélisé ainsi que les situations limites.
* Prévoir le mode « PAUSE » de votre application pour pouvoir stopper tous les processus et analyser la situation en cours.
* Implémentation de plusieurs catégories de clients avec des comportements différents.
* Implémentation de tous les postes décrits dans la partie « description de postes ».
* Les temps de chaque tâche, des postes, le nombre de clients par type, le temps en mode accéléré, … tout doit être paramétrable dans l’application.

### B / Technique

* Diagrammes de composants
* Diagramme(s) de séquences
* Au moins un IPC par type (synchro, échange de données ou les deux) doit être utilisé. Les threads vous seront indispensables également. A vous de décider sur l’utilisation d’un processus léger ou pas en fonction des situations. Voir ressource « Aide-mémoire » IPC sur Moodle.
* Utilisation des pools
* Utilisation des sockets pour les échanges entre la salle de restauration et la cuisine.
* Utilisation du langage C# .NET.
* L’ensemble des scénarii d’activité doit être enregistré en BDD
* Base de données de stocks en SQL Serveur, actualisée avec les livraisons et mise à jour en temps-réel en fonction des commandes.
* Utilisation des DP MVC qui est obligatoire).

# PRESENTATION DU GROUPE

Pour la réalisation du projet, nous sommes une équipe de 4 collaborateurs (développeur C#, Conception UML et Base de données « MCD, script de création et jeux de données ») dont un chef de projet.

Le projet a débuté le 03 décembre 2019 et se termine le 17 décembre 2019.



Dimitri Libessart :

Chef de Projet

Théo Vanengelandt : Développeur/ Conception UML



Alexis Caron :

Développeur / Conception BDD

Corentin Evrard :

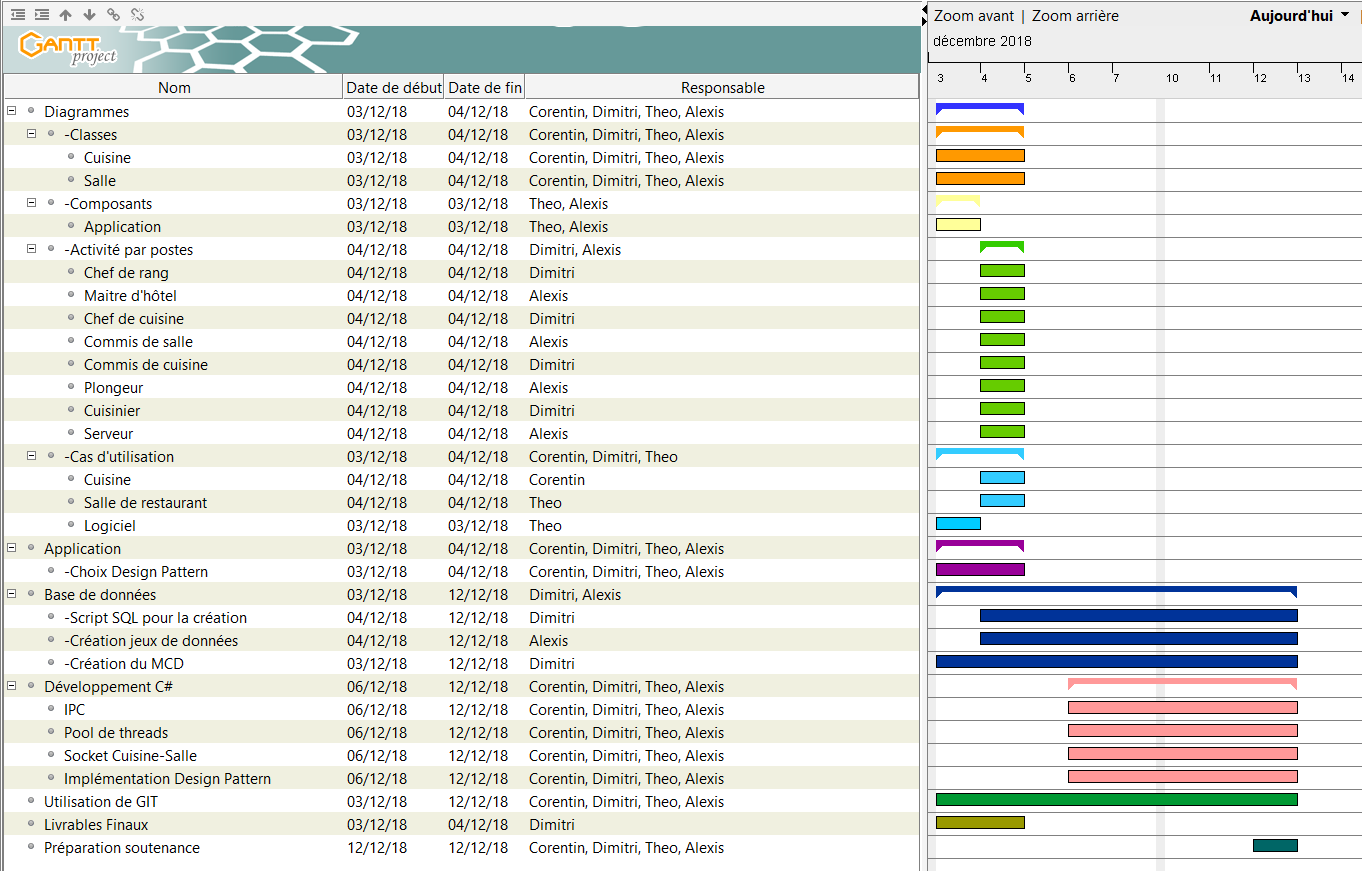
Développeur / Conception BDD

# III / Planning Prévisionnel

Pour réaliser à bien ce projet, nous avons mis en place un planning prévisionnel avec les tâches pour celui-ci.

Nous avons choisi celles-ci avec les compétences et les choix de chacun.

Ce qui en conclusion nous donne ce planning prévisionnel avec les différents délais définit avec l’ensemble du groupe pour bien optimiser le temps de travail et que chacun sache ce qu’il doit réaliser.



Nous avons réalisé les différents diagrammes UML (*Unified Modeling Language)* pour la conception de notre application mais aussi la conception de la Base de données avec le MCD (*Modèle Conceptuel de données*) et le script de création pour le rendu d’un livrable final le 5 décembre 2019.

D’après notre réflexion ce diagramme aller être quasiment identique au diagramme final mais à la suite de différentes difficultés et modifications du sujet nous avons grandement modifié ce diagramme qui sera expliqué dans la suite de ce rapport.

Nous avons donc séparé la base de données et les diagrammes UML en deux groupes de collaborateurs.

# IV / Choix techniques et Réalisations

Pour ce projet nous avons choisi un certain nombre de points à ne pas aborder par manque de temps ou de compétences techniques qui seront abordés dans la partie problèmes. Mais aussi à la suite de la modification du sujet le 10 décembre pour respecter au mieux.

Nous avons réalisé la plupart des éléments obligatoires, les diagrammes de composants et séquences n’ont pas été modifiées à la suite de l’envoi du 5 décembre.

Nous avions indiqué dans les livrables finaux utiliser les 6 Designs patterns demandés mais dans la réalisation nous avons utilisé le Design pattern MVC et singleton pour le maitre d’hôtel, le restaurant la cuisine et le displayer.

L’utilisation de thread à été réalisée pour les classes, maitre d’hôtel pour vérifier si une table est disponible., la classe Customer pour que le client mange, la classe ustensilWasher quand un ustensil est lavé, classe Washer pour le temps de lavage.

La mise en place d’un socket entre le chef de rang de la classe rowchief qui transmet la commande au chef de cuisine de la classe chef cuisine.

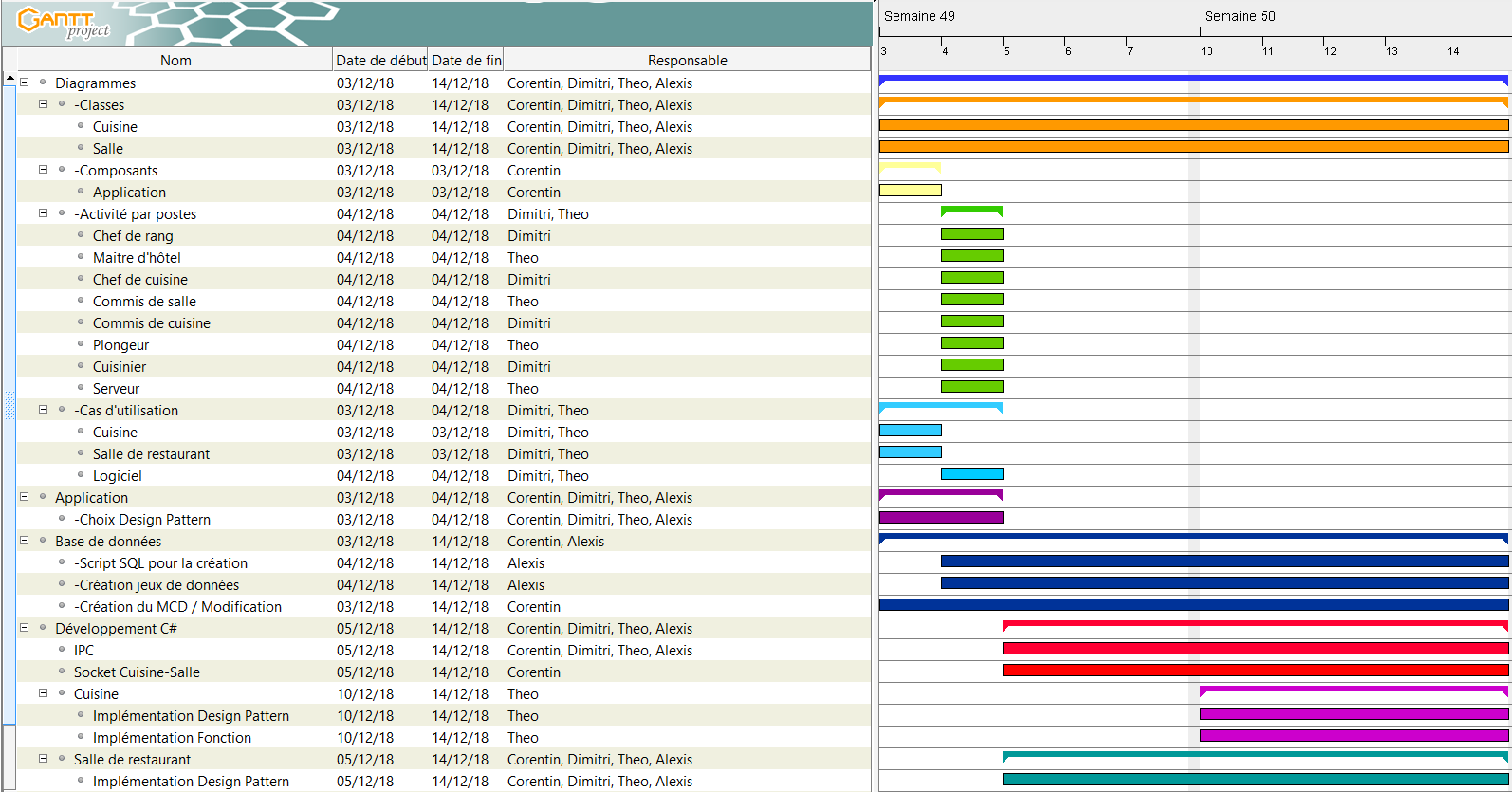
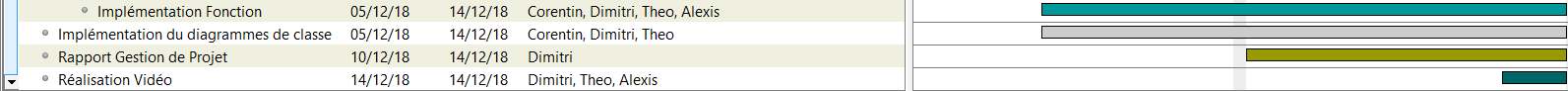
Nous avons donc un restaurant qui est instancié et s’affiche en console, de nouveaux clients qui arrivent dans notre restaurant et son reçu par le maitre d’hôtel qui va vérifier si une table est disponible malgré le nombre de places (exemple : 1 personnes sur une table de 8). Initialisation des pièces du restaurant, des tables de chaque pièce, du maitre d’hôtel, pour chaque chef salle des chefs de rang, des commis et des serveurs. Initialisation de la carte et création du premier groupe de clients pour le restaurant. Recherche des tables disponibles par le maitre d’hôtel.

Pour la cuisine, on a l’initialisation de la cuisine, la création du personnel de cuisine. Les machines pour le lavage du linge et des objets sont implémentées.

Nous n’avons pas mis en place un sémaphore par faute de temps mais aussi la View et le pool de thread de notre application mais nous avons les informations des différents processus du programme dans la console pour avoir un suivi lors du lancement du projet.

# V / Planning Réel de Projet

À la suite de ces 10 jours de projet, nous avons réaliser notre planning réel de projet avec lequel nous pouvons voir la différence entre le prévisionnel et celui-ci.

Nous pouvons voir que la répartition des tâches a été au mieux équilibré pour que l’ensemble de l’équipe ait le même nombre de tâches ou de temps de travail pour qu’il n’y ait pas de différence majeure au sein du projet en adaptant bien sûr en fonction des difficultés de chacun pour une meilleure optimisation du projet.

Une des grandes différences de ce diagramme de Gantt final et le Gantt prévisionnel et qu’il a fallu mettre un collaborateur chargé de la modification à chaque fois de la base de données donc du MCD pour les différents changements lors du développement du projet.

Le diagramme de classes à aussi été modifier et à évolué pour représenter au mieux notre application avec les différentes fonctions, méthodes et classes qui composent celui-ci. Nous avons donc tout au long du projet le mettre à jour ce qui n’avait pas été pensé lors de la rédaction du diagramme prévisionnel.

Nous avons commencé la partie développement à la suite de l’envoi du livrable final, nous avons eu des difficultés à la suite de cette période qui sera développée dans la partie problèmes.

# VI / Problèmes rencontrés

Lors de la réalisation de notre projet, nous avons eu des difficultés pour :

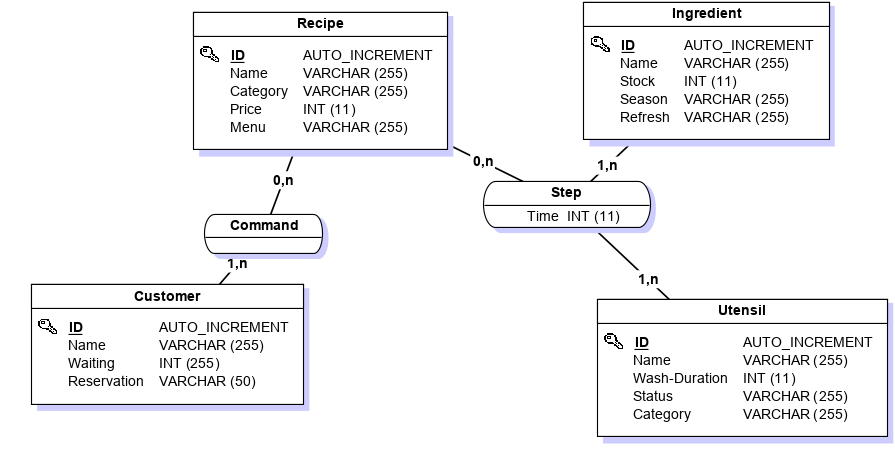
-La réalisation des différents diagrammes pour la conception de notre application parce qu’il y avait un livrable à rendre en deux jours et avec le nombre de diagrammes à réaliser nous n’avons pas pu réfléchir plus amplement aux différents points à aborder sur le sujet. Ce qui nous a ralentis lors du développement de l’application car il fallait encore réfléchir à ceux-ci et modifier le diagramme de classes dans le même temps pour le garder à jour.

-Problème avec le GitHub lors des différents push sur la branche master ce qui n’a a fait perdre pas mal d’heures pour la remise en place. Mais aussi l’utilisation des branches, seul Théo une maitrise de Git Bash.

-Problème de ressources humaines des collaborateurs durant le projet, difficile de mettre en place ou organiser des tâches, échéances souvent décalées d’où un léger retard tout au long du projet qui à la fin ne permets pas de réaliser toutes les fonctionnalités demandées.

# VII / MCD

Voici le MCD final de notre application, nous pouvons constater plusieurs modifications depuis le rendu le 5 décembre 2018 du livrable. Nous avons donc après plusieurs réflexions modifiées celui-ci par manque de temps pour la réalisation du projet ou que certaines classes n’avaient pas besoin d’être créé.



**Liste des tables :**

La table Ingrédient contient tous les ingrédients utilisés en cuisine, leurs noms, stocks, le niveau de fraicheur, la saison de l’ingrédient.

La table Utensil contient tous les ustensiles utilisés pour les recettes, nom, stock et temps de lavage.

La table Recipe contient toutes les recettes utilisées dans le restaurant, noms, prix, temps de préparation, catégories (entrée, plats, desserts, boisson).

La table Customer contient la liste de toutes les clientes et tous les clients ainsi que leurs noms, temps d’attentes, s’il y a une réservation.

**Relations :**

La table Recipe possède 2 relations dont 1 à triple branche :

- Une recette a besoin d’un 1 ou plusieurs Ingrédients pour faire une étape.

- Une recette emploie 1 ou plusieurs Ustensiles pour faire une étape.

La table Customer possède 2 relations :

- Un client commande 1 ou plusieurs Recipe

La table Utensil possède 1 relation à triple branche :

- Un ustensile est employé pour 0 ou plusieurs recettes pour réaliser une étape.

- Un ustensile est utilisé pour 0 ou plusieurs ingrédients pour réaliser une étape.

La table Ingrédient possède 1 relation à triple branche :

- Un ingrédient est contenu dans 0 ou plusieurs recettes pour réaliser une étape.

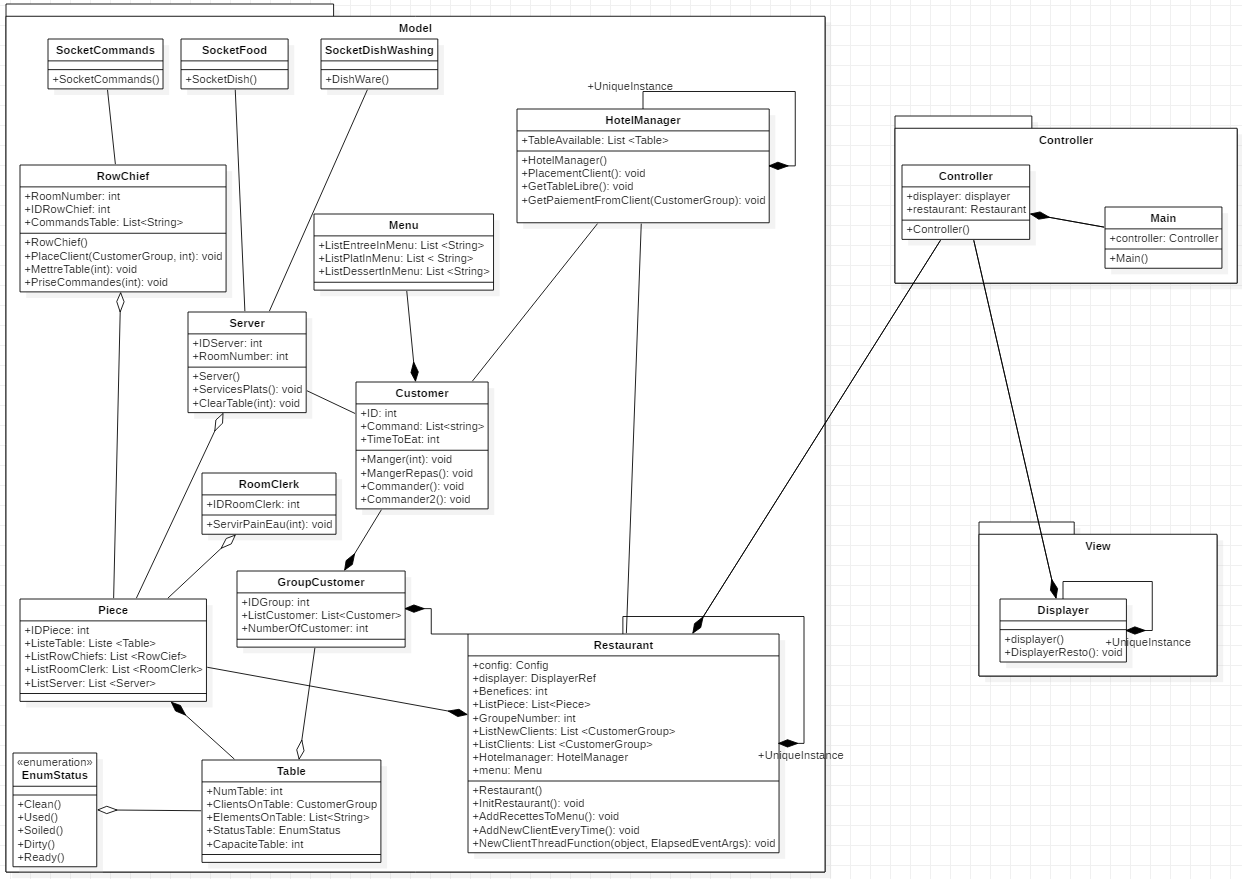
- Un ingrédient est utilisé avec 0 ou plusieurs ustensiles pour réaliser une étape.

# VIII / Diagramme de classes

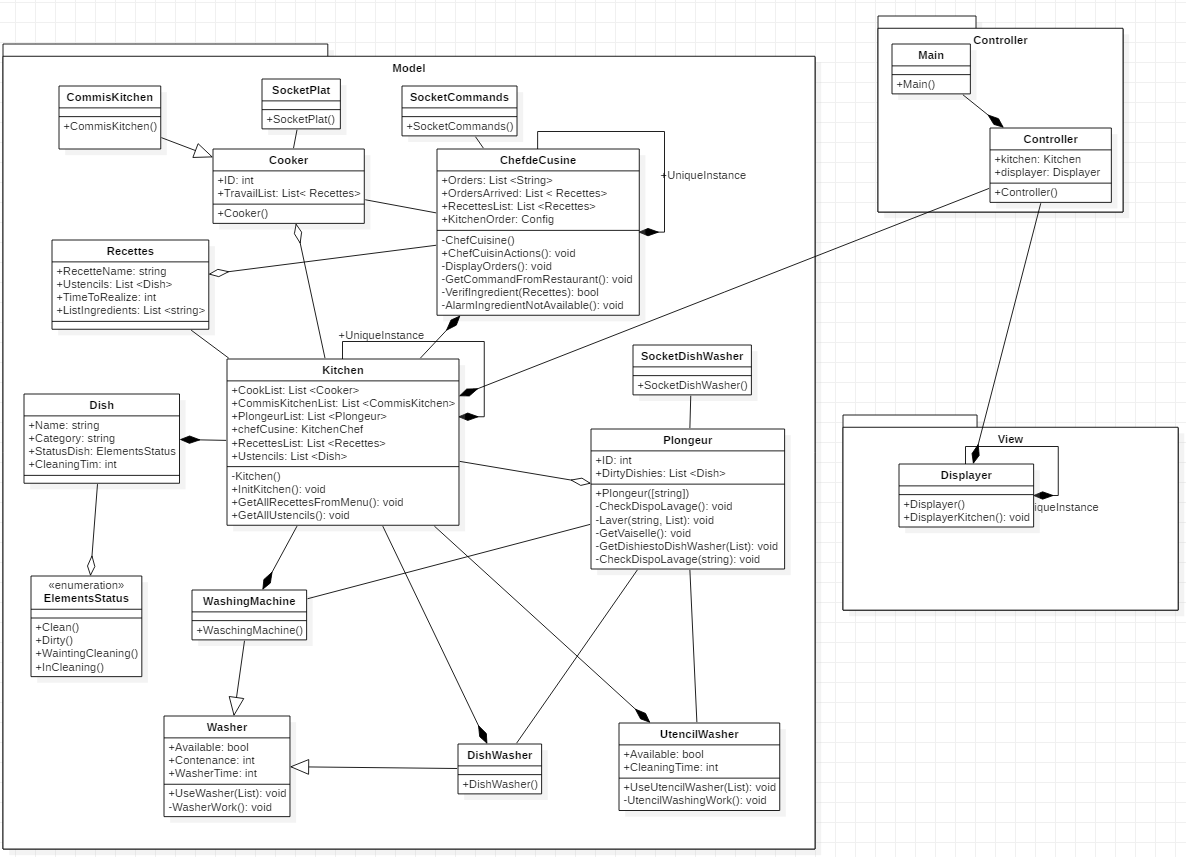
Comme cité plus haut dans le rapport nous avons modifié et mis à jour le diagramme de classes tout au long du projet pour permettre d’avoir d’être le plus précis possible sur celui-ci.

Nous pouvons constater que plusieurs classes et interfaces ont étaient modifiées ou supprimées à la suite de la réalisation du projet vu qu’il y a eu une modification du sujet en cours de semaine ce qui a eu un impact sur notre diagramme nous avons dû revoir les différents objectifs attendus et le modéliser.

Voici le diagramme de classes du restaurant à la fin de notre projet nous pouvons donc voir que plusieurs classes ont été supprimées ou modifiées.



Voici le diagramme de classes de la cuisine à la fin de notre projet nous pouvons donc voir que plusieurs classes ont été supprimées ou modifiées.



# IX / Conclusion

Pour la réalisation de ce projet de fin d’année, nous avons pu mettre en place dans la majeure partie du projet les points vus en prosit même si lors de certains jours des difficultés ont étaient rencontrés.

Nous avons essayé de répondre au mieux au besoin qui nous as été fournis ou d’expliquer les points non réalisés.

Le manque de temps à été un point très impactant dans ce projet car la conception d’une application avec tous les diagrammes est difficile à réaliser et à se projeter en si peu de temps. Ce qui s’est ressenti lors de la première semaine de projet où nous avons accumulés beaucoup de retard que nous avons rattrapé en partie lors de la seconde semaine de projet quand l’équipe à été au complet ce qui as permis une meilleure fluidité et entraide dans ce projet.

Projet très intéressant dans l’ensemble mais un groupe de 5 personnes n’aurait pas été un problème.

Pour la gestion de projet il est difficile de diriger les différentes tâches avec les difficultés et les absences ou le temps du projet difficile à gérer car une tâche de retard ralenti les autres tâches.

Un collaborateur doit toujours toucher à la base de données ce qui est aussi impactant.