

# RAPPORT DU PROJET IHM

**Enora GERMOND**

**Aléxia BERNARD**

27/11/2021

groupe 4

## INTRODUCTION

Ce projet IHM, à faire en binôme, avait pour but de réaliser deux IHM pour un hôtel. La première permet à un employé d'allouer une chambre à un client grâce à une préréservation. La seconde permet au gérant de l'hôtel d'avoir accès à des statistiques (taux d'occupation, graphique). Nous avons pour cela utilisé JAVA ainsi que GIT.

Notre git est sur le dépôt d'Aléxia et a pour nom FIprojetIHM2021.

## RÉALISATION

Tout d'abord, nous avons commencé par faire l'IHM numéro une. N'ayant pas vraiment compris ce que nous devions faire, nous avons débuté en reprenant l'exemple du professeur qui montrait comment utiliser la partie MNP et l'API. Ainsi nous avons codé la première IHM avec uniquement la partie MNP. Cette partie fût la plus longue à faire car c'est durant cette étape que nous avons pu comprendre ce que voulait M Madelaine. Nous avons donc séparé les dossiers API et MNP de notre IHM1 que nous avons elle-même séparé en trois: vues, modèles et contrôleurs. Nous avons décidé de mettre tout de même une partie modèle qui fera le lien avec la partie MNP et MP dans notre IHM afin de mieux distinguer les parties, même si le vrai modèle est dans MP et MNP (Modèle (Non) Persistant).

Une fois la première IHM finie, nous avons donc décidé de coder les fichiers MP qui font la connexion avec la base de données et de créer le main qui va avec. Cette partie était assez rapide pour la première IHM, et assez complexe pour la seconde où j'ai dû faire des modifications au fur et à mesure.

Par la suite nous avons donc commencé à coder la partie de l'IHM2 directement avec notre MP.

Nous avons fini en ajustant la taille des boutons, changé quelques locations de Composant de la fenêtre afin de rendre l'IHM visuellement plus agréable et compréhensible.

Nous avons tout de même eu quelques soucis :

Dans la partie MP dans le fichier ReservationFactoryMP, dans plusieurs méthodes, par exemple getReservation(LocalDate d1, LocalDate d2, TypeChambre t) : ici, nous avons

récupérer toutes les réservations grâce à la méthode `getAllReservations()` que nous avons stocké dans un `Set<Reservation>` le problème était que une fois les mauvaises réservations trouvés nous n'avons pas réussi à les supprimer de ce Set. C'est pourquoi nous avons dû faire l'inverse en créant un deuxième `Set<Reservation>` initialisé à vide où nous avons ajouté les bonnes réservations.

Comme je le mentionne ci-dessous, lorsque nous essayons de cliquer sur le bouton effacer pour effacer un graphique après avoir cliqué sur un autre bouton il ne s'efface pas. Nous avons essayé de mettre le graphique dans le JPanel centre mais sans succès, c'est pourquoi nous avons dû laisser comme ça.

## TESTS

Tests à effectuer :

DANS IHM1 :

Client disponibles :

Marine Carpentier	PAS DE PRERESERVATION
Kimberley Leroux	PAS DE PRERESERVATION
Aaron Dumas	DEUX PRERESERVATIONS
Florian Faure	TROIS PRERESERVATIONS

Quelques Références :

9763-5716-PHKX  
7780-7814-ARJO  
5468-4123-TRDT

DANS IHM2 :

Date disponibles pour trouver un ratio ou pour construire le graphique :  
03/01/2018 à 07/01/2018

Essayer d'effacer le graphique après avoir cliqué sur un autre bouton : le graphique ne s'efface plus.

Mettre que si on essaie d'effacer dans IHM2 le graphique après avoir cliqué sur un autre bouton, celui-ci ne s'efface plus.

Pour lancer la première IHM en MP : `make run_ihm1`

Pour lancer la première IHM en MNP : make run\_ihm1\_mnp

Pour lancer la seconde IHM : make run\_ihm2

## DIAGRAMMES

Nous avons fait le diagramme de classe des IHM1 et IHM2, ce qui nous permet de mieux visualiser les liens entre les classes et de voir la complexité du projet. Ils sont également dans le git afin que vous puissiez les consulter dans leur intégralité avec une meilleure qualité.

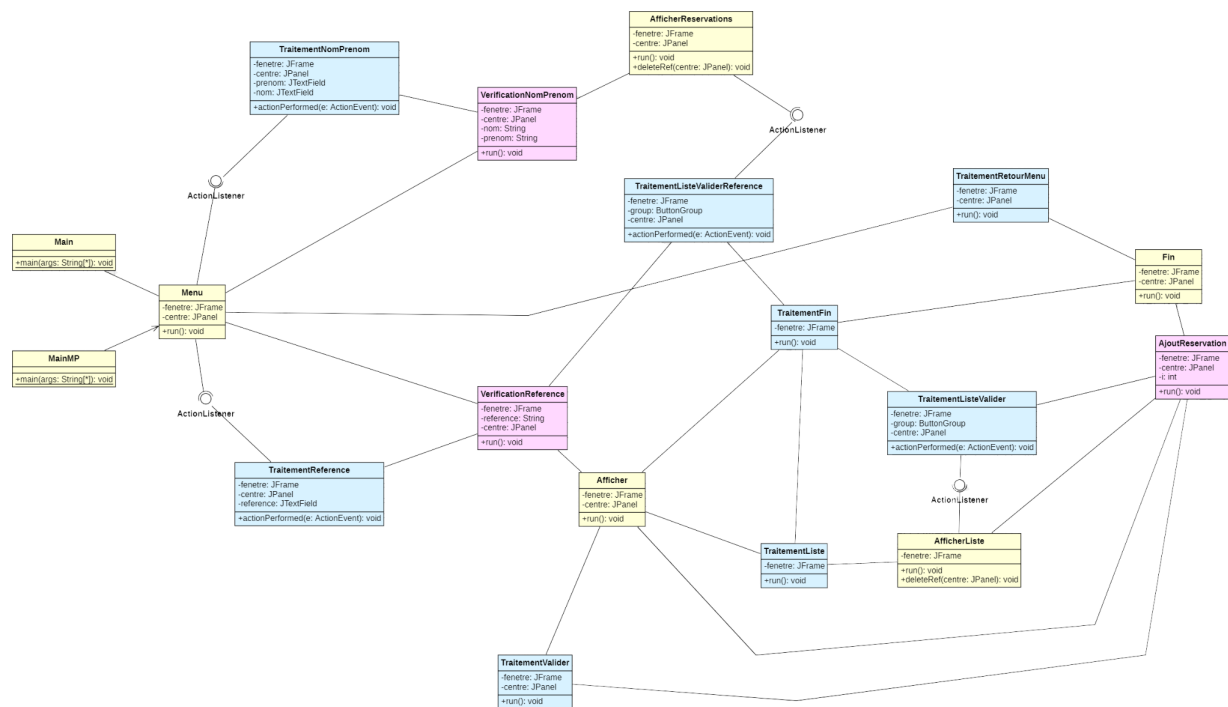


Diagramme de classe IHM1

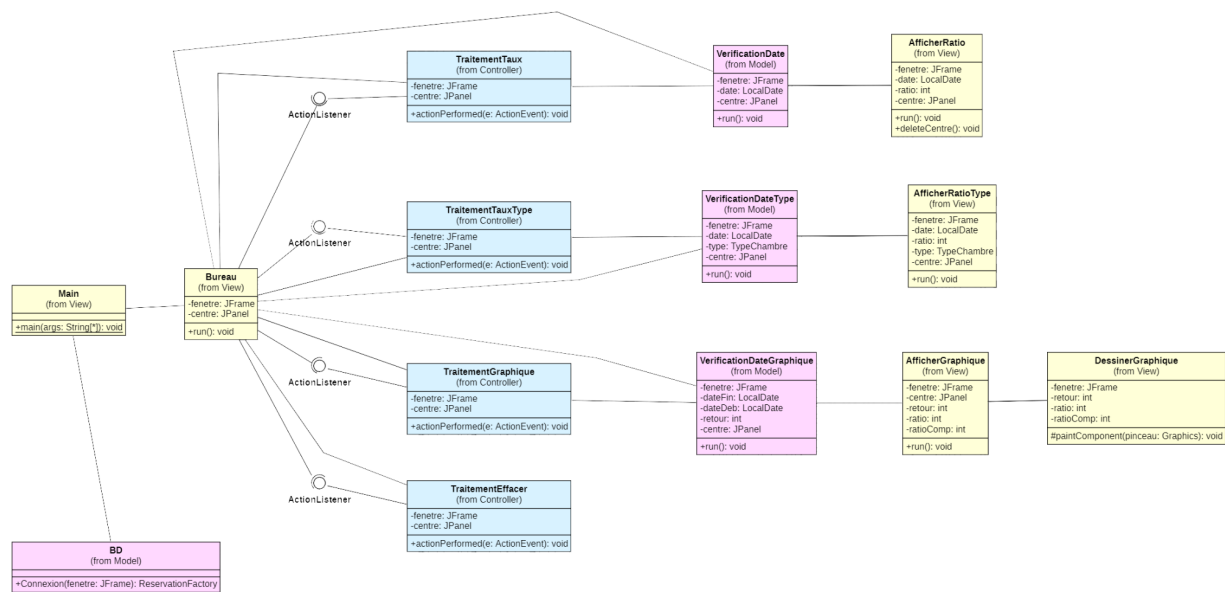


Diagramme de classe IHM2

## CONCLUSION

### D'Aléxia

Pour conclure, j'ai beaucoup aimé faire ce projet car il m'a permis d'apprendre beaucoup de choses, comme par exemple l'utilisation des Set<E>, comment coder avec une API et quelques détails à propos du code en lien avec la Base de Données. Je regrette tout de même le fait que nous n'ayons pas réparti équitablement le travail.

Ce projet m'a permis de me conforter dans l'idée que j'aimerais faire du développement (en java par exemple) dans mes études après le DUT.

### D'Enora

J'ai trouvé ce projet très intéressant car je n'avais encore jamais travaillé avec un vrai modèle MVC. Il m'a également permis de mieux comprendre comment utiliser une API et une Base de Données dans un projet. Je regrette juste d'avoir perdu du temps sur des choses qui ne le méritaient pas forcément et qui n'ont pas aidé à la répartition du travail. J'ai donc noté les difficultés rencontrées afin de m'améliorer par la suite. Ce fut donc une expérience enrichissante.

### Commune

Pour conclure pour le groupe, nous sommes globalement d'accord que nous aurions dû mieux répartir le travail afin que chacune puisse s'améliorer sur des choses plus variées. Cependant nous sommes quand même satisfaites du projet dans sa globalité qui nous a tout de même appris de nombreuses choses.