

# Rapport Projet

Durant cette période de projet, nous devons réaliser la gestion de l'hôtel « Blue Bay Curaçao » c'est-à-dire la gestion des clients, des chambres, de son restaurant, de son casino ainsi que du service de spa qu'il propose. Le but final est de pouvoir directement enregistrer les clients et leurs informations sur la plateforme utilisateur. Puis, nous leur attribuerons une chambre. Nous leur proposerons ensuite s'ils désirent aller au restaurant, au casino ou au spa. Il a fallu donc créer les classes Client, Chambre, JeuCasino, Restaurant, Spa et enfin la classe Utilisateur qui permettra de créer la structure finale.

- La classe Client

Le constructeur de notre classe Client contient le nom, le prénom ainsi que l'année de naissance de la personne. Pour cette dernière nous avons utilisé la classe SimpleDateFormat qui nous aidera à analyser la validité des dates. La complexité de cela est moindre comparé à une vérification manuelle des dates. Le constructeur contient également l'adresse e-mail. On vérifie la validité de celle-ci en testant la présence d'un « @ » ainsi que d'un « . ». Enfin, le constructeur contient le numéro de téléphone du client. Il y aura deux types de numéros. Le premier commencera par un « + » qui sera suivi de onze chiffres. Le deuxième type de numéros contiendra uniquement 10 chiffres. On vérifie la présence dans un cas du « + » et le nombre exact de caractères (pour les deux cas). Enfin, on vérifie que le numéro est bien d'un entier grâce à la méthode estUnEntier. La méthode toString affichera la fiche client.

- Classe Chambre

Notre classe Chambre contient l'attribut tableauClient représentant l'ensemble des clients occupant la chambre. Grâce à cela on pourra attribuer le type de chambre (simple, double ou triple) en fonction du

nombre d'occupant de la chambre. Si deux personnes occupent une même chambre, alors ceux-ci auront le choix entre une chambre double et une chambre deluxe. Les méthodes `setDépart` et `setArrivée` vérifieront la validité des dates d'arrivée et de départ à l'hôtel des clients grâce à la classe `SimpleDateFormat`. Ensuite, la méthode `départAprèsArrivée` vérifiera que le départ des clients a bien lieu après la date d'arrivée. La méthode `calculNbJours` calculera le nombre de jours pendant lequel les clients restent à l'hôtel. Cela nous sera utile pour calculer le prix final du séjour. Grâce aux méthodes `compareDate` et `disponibilitéUneChambre` on vérifie si une chambre est disponible. Il faut savoir que l'hôtel contient 20 chambres. Les chambres 1 à 5 sont les simples, de 6 à 10 il s'agit des chambres doubles, de 11 à 15 il s'agit de chambres deluxe et les chambres de 16 à 20 sont les triples. A chaque réservation, le numéro de chambre et les dates d'arrivées et de départ sont ajoutées à des tableaux. Dans les deux méthodes, on prend en compte le type de la chambre demandée ainsi que les dates. On va ensuite vérifier dans le tableau stockant tous les numéros de chambres qui ont été réservées, toutes les réservations déjà effectuées correspondant à ce type de chambre. Par la suite, on comparera les dates pour savoir si au moment où l'on veut effectuer une réservation, une chambre de ce type est disponible. Puis, dans la méthode `getNumeroChambre` on renvoie le numéro d'une des chambres disponibles qui sera attribué à la réservation.

- La classe `JeuCasino`

Dans notre classe `JeuCasino`, il y a un attribut `cagnotteOfferte` récupérant le montant du ticket pour le casino offert par l'hôtel qui est initialement de 20 euros. Cet attribut se trouve initialement dans la classe `Client`. Il y a un tableau `argentMise` représentant chaque mise que le joueur effectuera et un tableau `tableauMise` représentant les éléments sur lesquels le joueur a misé. Il y a également une variable `gain` qui initialement est à 0 et une variable `argentInseree` qui représente l'argent que le joueur insère dans la machine ou donne au croupier en échange de jetons pour jouer. Lors de « l'insertion » de la monnaie, le joueur a le choix entre se servir de la cagnotte offerte, dans ce cas celle-ci sera changée avec un appel à la classe `Client`, et se servir de son argent personnel.

Dans `faitesVosJeux`, on récupère le tableau `argentMise` et on vérifie que `argentMise` et `tableauMise` soient de même dimension. Il faut par la suite convertir les éléments insérés dans `tableauMise` pour se servir de `Mise` (l'énumération). On tire ensuite un entier aléatoirement entre 0 et 36. Pour chaque élément du `tableauMise`, on vérifie si le nombre tiré correspond à

ce sur quoi le joueur a misé. En cas de victoire, on modifie gain en ajoutant à celui-ci 2 fois l'argent misé sur l'élément gagnant. On prend soin à la fin d'effacer le tableauMise. VersementGain permet de rajouter le montant gagné à l'argent inséré et réinitialise le gain. La méthode deroulementJeuComplet contient toutes les méthodes énoncées précédemment à la suite.

- **La classe Restaurant**

La classe Restaurant contient un attribut tableauClient contenant les clients présents à une table du restaurant. La méthode commande demande à chaque client ce qu'il désire commander c'est-à-dire s'il souhaite un menu (Mojito ou Daiquiri) ou s'il souhaite choisir à la carte par exemple. Puis, la méthode CalculTotalPrix calcule le prix total pour la table. Enfin, la méthode Règlement demande à chaque client s'il souhaite utiliser des tickets restaurants. Si sa réponse est positive, on lui demande combien il souhaite en utiliser. On va donc récupérer l'attribut concernant le nombre de tickets qu'il possède dans la classe Client. Grâce à cela, on vérifie que le nombre de tickets que le client souhaite utiliser est inférieur au nombre de tickets qu'il possède. Puis, nous vérifierons que le client ne met pas trop de tickets sur la table c'est-à-dire si le montant correspondant au nombre de tickets qu'il choisit de mettre n'est pas supérieur au prix total du repas. A la fin, on affiche le nombre total de tickets utilisés ainsi que le prix restant à payer.

- **La classe Spa**

Notre classe Spa possède les attributs tableauClient, tableauRelaxant, tableauPierre et tableauBougie. tableauClient répertorie les différents clients présents au spa. A chaque client, on demande s'il souhaite un massage relaxant, un massage aux pierres ou un massage à la bougie. Selon ses réponses, on remplit les différents tableaux à l'aide de booléens. On fixe également des valeurs correspondant au nombre de massage demandé si la réponse du client est positive. Pour calculer le prix total des soins on récupère les valeurs de chaque tableau de massage correspondant à l'indice du client concerné.

- **La classe Utilisateur**

Dans la classe Utilisateur, nous appelons toutes nos autres classes et nous nous servons des Scanners pour créer une plateforme d'utilisation de notre projet. Dans un premier temps, grâce à la classe Client, nous

enregistrons les clients en tapant leur nom, prénom, date de naissance, etc... Puis, nous effectuons la réservation d'une chambre en entrant le type de chambre désiré ainsi que les clients qui occuperont cette chambre. L'avantage de notre classe Chambre est que nous pouvons réserver des chambres pour le jour-même mais nous pouvons réserver des chambres pour le futur grâce à nos tableaux répertoriant toutes les réservations. (On leur demande ensuite s'ils désirent accéder aux activités). Par exemple, s'ils désirent aller au restaurant, on enregistre les clients occupant une table au restaurant puis on se sert de la classe Restaurant pour la gestion de la commande et du repas. On demande également si un client veut aller au casino. Dans ce cas, on enregistre le client qui sera le joueur. Puis, grâce à la classe JeuCasino on fait jouer ce client. Enfin, on demande aux clients s'ils souhaitent profiter du spa. On enregistre alors les clients souhaitant aller au spa et on leur proposera les services qu'ils désirent.