

	<p align="center">Concepts de programmation avancés</p> <p align="center">Projet de NFP 121</p> <p align="center">2021-2022</p>	<p align="right">17/02/2022</p> <p align="right">Par J. LAFORGUE</p>
--	--	--

CONS
 NATIONAL
 DES ARTS
 ET METIERS
Cnam
 CENTRE REGIONAL
 MIDI - PYRENEES

NFP 121 – Travaux pratiques

PROJET

*IPST-
CNAM*

Années 2021-2022
TOULOUSE
J. LAFORGUE

CORRECTION

1. INTRODUCTION

Groupe de projet : 0

Participants au projet : Jacques LAFORGUE

Objectif du projet : Programme permettant à au moins 4 joueurs de jouer au Poker.

Toutes les fonctionnalités décrites dans le §1 « Besoin » du sujet sont réalisées dans ce programme.

Les fonctionnalités décrites dans le §3 « Besoins complémentaires » qui ont été réalisées dans ce programme sont :

- Fil de discussion des joueurs
- Déterminer quel joueur à la meilleure main.
- Remplacer un joueur par un robot. Voir le commentaire de la classe RobotImpl pour les choix qui ont été fait sur le comportement du robot.
- Un minimum d'activation/désactivation des boutons a été réalisé, non pas en fonction des règles du jeu de poker, mais uniquement en fonction de si le joueur est le donneur et s'il est le joueur courant.

Ce programme est la version distante : les IHM sont des programmes séparés qui communiquent via les socket, avec le programme de serveur de jeu de poker.

2. LE DOSSIER DU PROJET

2.1. Conception préliminaire

Nous avons réalisé deux programmes :

- ServeurPoker
- PokerIHM

Le programme **ServeurPoker** doit être exécuté en premier. Il contient le Modèle et le Contrôleur permettant de gérer une partie de poker de 2 à 4 joueurs.

Le Modèle contient tous les traitements de gestion du jeu. Il maintient à tout instant un état cohérent du jeu (TablePoker). Il notifie à toutes les IHMs distantes cet état.

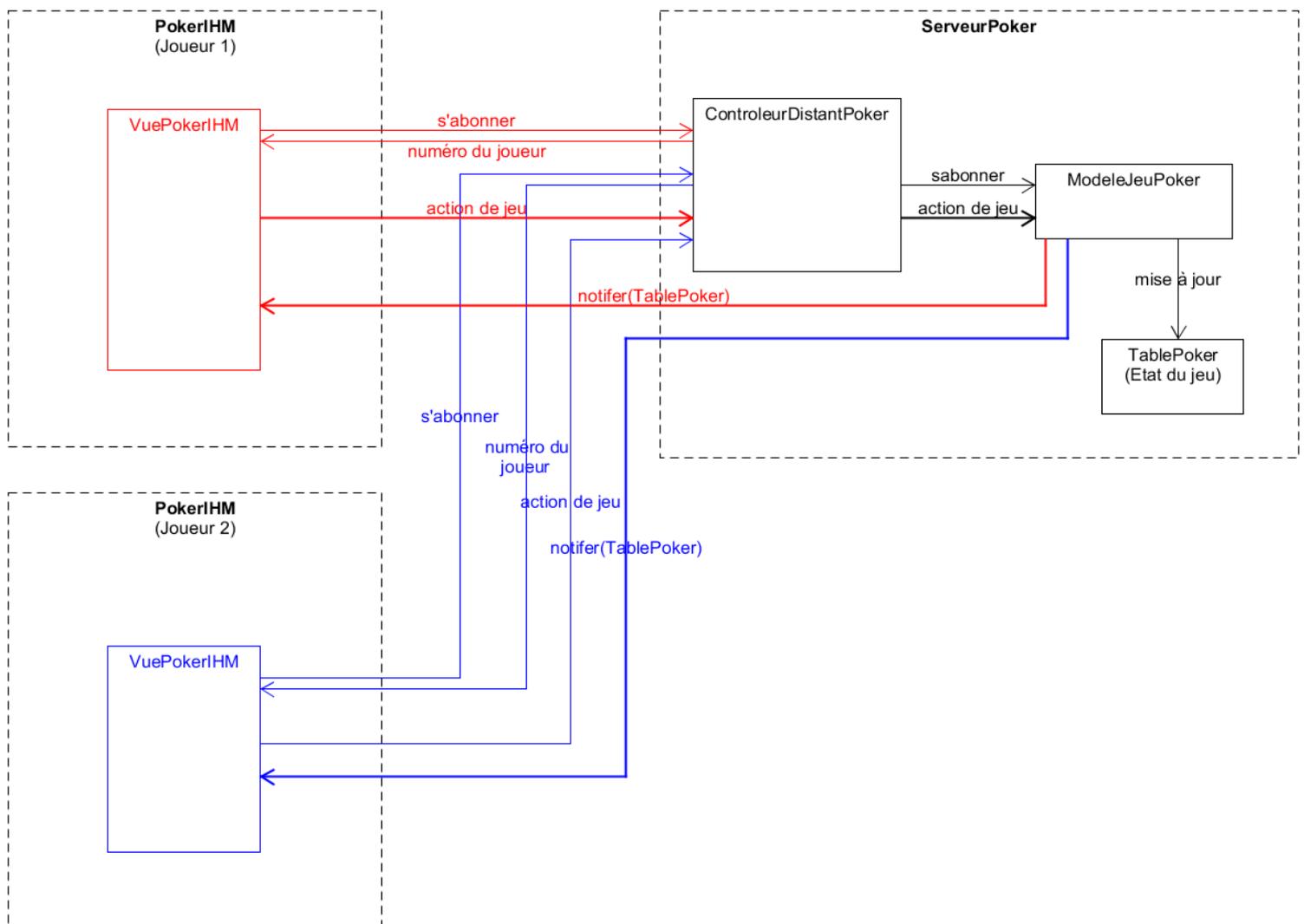
Le Contrôleur traite la réception des requêtes envoyées par PokerIHM.



Pour jouer, un joueur doit exécuter le programme PokerIHM qui affiche une IHM. L'IHM envoie une requête au serveur pour s'abonner. En retour, il reçoit son numéro unique.

L'IHM se met en attente de réception de la requête de notification de ServeurPoker dans laquelle se trouve l'état courant du jeu. En fonction de cet état, PokerIHM met à jour son IHM.

A chaque fois qu'un joueur réalise une action, il envoie une requête au serveur. Le serveur met à jour son état de jeu en fonction de l'action réalisée et notifie toutes les IHM distantes. Toutes les IHM sont ainsi mises à jour suite à l'action d'un des joueurs.



Les programmes sont implémentés suivant le **DP MVC**.

Le principe d'abonnement et de notification sera implémenté sur le serveur suivant le **DP Observateur** : suite à l'abonnement de chaque IHM un observer est créé qui enverra la requête de notification à cet IHM.

Etant donné que VuePokerIHM est unique dans le programme PokerIHM, VuePokerIHM sera un **DP Singleton**.

Etant donné que ControleurDistantPoker et ModeleJeuPoker sont uniques dans le programme ServeurPoker, ControleurDistantPoker et ModeleJeuPoker seront des **DP Singletons**.

	Concepts de programmation avancés Projet de NFP 121 2021-2022	17/02/2022 Par J. LAFORGUE
--	--	--------------------------------------

2.2. Les cas d'utilisation

Les cas d'utilisation que nous envisageons sont uniquement ceux qui correspondent au besoin exprimé dans le sujet :

2.2.1. Cas 1

Exécuter ServeurPoker et exécuter 4 PokerIHM

Le donneur est le premier joueur.

L'indicateur du donneur est positionné sur ce premier joueur.

Commencer la 1^{ère} manche entraîne la distribution des cartes à chacun des joueurs. Chaque joueur voit ses cartes et ne voit pas les cartes des autres.

L'indicateur du joueur courant est affiché sur le joueur courant.

Suite à l'action du joueur courant, le joueur suivant devient le joueur courant.

Si le donneur recommence une nouvelle manche alors le nouveau donneur est le joueur suivant.

L'indicateur du donneur est positionné sur ce premier joueur.

2.2.2. Cas 2

Exécuter ServeurPoker et exécuter 2 PokerIHM

Le pot affiché sur l'IHM est mis à jour en fonction des actions de mises de chacun des joueurs : miser blind, miser double-blind, miser 100.

Changer la valeur de la blind et miser des blinds et des doubles blinds.

Le joueur qui fait l'action de ramasser le pot, gagne tous les jetons du pot.

2.2.3. Cas 3

Exécuter ServeurPoker et exécuter 3 PokerIHM

Vérifier qu'un joueur qui a abandonné ne joue plus (il ne peut plus être un joueur courant)

Vérifier que à la manche suivante ce joueur peut jouer.

Vérifier qu'un joueur qui passe (checker), passe sont tour de jeu mais continue à jouer.

2.2.4. Cas 4

Exécuter ServeurPoker et exécuter 2 PokerIHM

Vérifier que le donneur peut distribuer le flop, le turn et la river. Les cartes sont visibles par tous les joueurs.

Vérifier que si les 2 joueurs font tapis alors le joueur qui a le plus de jetons et qui ramasse le port, ramasse tous les jetons du joueur adverse.

2.2.5. Cas 5

Exécuter ServeurPoker et exécuter 2 PokerIHM

Vérifier que si les 2 joueurs font tapis alors le joueur qui a le moins de jetons et qui ramasse le port, ne ramasse pas tous les jetons du joueur adverse mais uniquement le montant de la totalité de ses jetons (mise comprises).



2.3. L'aspect de l'IHM

JEU DE POKER

Jacques
Commencer
Flop Turn River
Blind 5 Valider
Miser Blind
Miser Double Blind
Mises
Miser
Checker
ALL-IN
Abandonner
Rammasser le pot
Montrer ses cartes
Gagnant ?
Pot : 0
bonjour
Envoyer

Jacques
Jetons 1000
Mises 0

Pierre
Jetons 1000
Mises 0

Jean
Jetons 1000
Mises 0

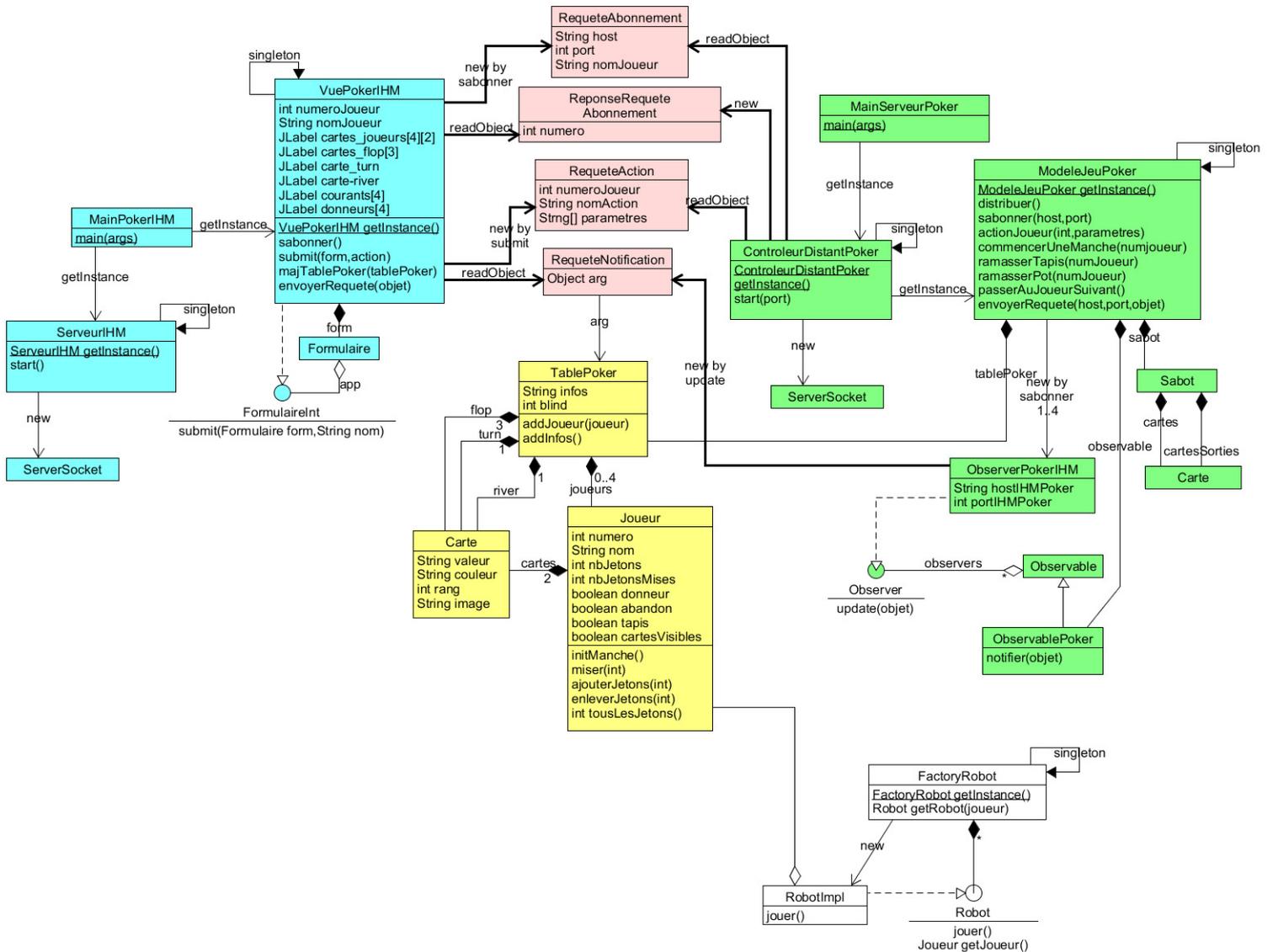
Paul
Jetons 1000
Mises 0

8♣ 4♦ A♣ 6♦ A♠

Jacques> bonjour
Jacques> RIVER
Jacques> TURN



2.4. Diagramme de classe



2.5. Les étapes de développement

- 1/ Mettre en place l'architecture globale du projet dans laquelle :
 - Le client IHM s'exécute contenant un seul bouton
 - Le client IHM s'abonne au serveur de jeu
 - Le client IHM lance son serveur de socket
 - Sur le bouton, l'IHM fait une action bidon
 - Le serveur de jeu traite l'action en envoyant une notification au client IHM
 - Le client IHM fait une trace ou affiche le fait qu'il a bien reçu la notification
- 2/ Vérifier que le projet fonctionne avec plusieurs IHM : les autres IHM sont également notifiées.
- 3/ Implémenter une seule action de jeu en vérifiant que les joueurs jouent à tour de rôle.
- 4/ Implémenter toutes les actions d'un joueur une par une en vérifiant à chaque fois que l'action est conforme.



3. LES SOURCES

Les sources pour la réalisation du jeu de poker sont structurés en 5 packages :

- **fr.cnam.poker.main**
Ce package contient les deux programmes principaux permettant d'exécuter le serveur et une IHM
- **fr.cnam.poker.serveur**
Ce package contient toutes les classes de réalisation du serveur de poker.
- **fr.cnam.poker.client**
Ce package contient toutes les classes de réalisation du client IHM de poker.
- **fr.cnam.pokercommun**
Ce package contient toutes les classes communes entre le serveur et le client : les classes de définition des requêtes et des paramètres de ces requêtes échangées entre le serveur et le client.
- **fr.cnam.poker.robot**
Ce package contient les classes permettant de créer un robot qui remplace un joueur humain.

Les programmes d'exécution :

- runServeurPoker.bat
Programme d'exécution du serveur de poker.
`java -classpath bin fr.cnam.poker.main.MainServeurPoker <port serveur>`
<port> est le port du serveur de socket du serveur de poker = **9100**
- runClientIHMPokerJoueur<N>.bat
Programme d'exécution d'une IHM d'un joueur
`java -classpath bin fr.cnam.poker.main.MainPokerIHM <nom> <host serveur> <port serveur> <port>`
<nom> est le nom du joueur
<host serveur> est le nom du host du serveur de poker
<port serveur> est le port du serveur de socket du serveur de poker
<port> est le port utilisé par le serveur de socket du client = **9101, 9102, 9103, 9104**
- runRobot<N>.bat
Programme d'exécution d'un robot.
`java -classpath bin fr.cnam.poker.main.MainPokerIHM ROBOT_<N> <host serveur> <port serveur> <port> = 9102, 9103, 9104`

Mode d'emploi :

- Exécuter d'abord runServeurPoker.bat.
Puis, les runClientIHMPokerJoueur<N>.bat pour chacun des joueurs

Pour jouer avec un robot :

Exécuter d'abord runServeurPoker.bat

Puis, un joueur humain : runClientIHMPokerJoueur1.bat

Puis, un robot : runRobot1.bat

Attention, ne pas lancer un robot et un joueur avec le même numéro de port (exemple : ne pas lancer runRobot2.bat et runClientIHMPokerJoueur2.bat en même temps).

Si vous êtes sur des machines différentes, il faut changer « localhost » par le nom de la machine du serveur dans le script que vous utilisez pour exécuter l'IHM d'un joueur.