**GEN**

**Rapport Projet :**

**Poker**



**Auteurs** : Balestrieri Hakim & Gomes Christian

**Professeur :** Lachaize Patrick

**Date :** xx.06.2020

Table des matières

[Introduction 3](#_Toc42176820)

[Règles du jeu « Poker Texas Hold’em » 4](#_Toc42176821)

[Organisation 4](#_Toc42176822)

[Début d’une partie 4](#_Toc42176823)

[Combinaison de cartes 6](#_Toc42176824)

[Adaptation des règles 7](#_Toc42176825)

[Gestion de projet 8](#_Toc42176826)

[Sprint 1 8](#_Toc42176827)

[Sprint 2 8](#_Toc42176828)

[Sprint 3 8](#_Toc42176829)

[Sprint 4 8](#_Toc42176830)

# Introduction

Dans le cadre de la deuxième année concernant le cours de GEN, nous devons réaliser un projet qui comporte différents éléments du cours afin de les mettre en pratique. Notre choix s’est porté sur le jeu d’un « Poker » à deux cartes.

Ce choix nous permet de mettre en pratique :

* La programmation réseau & multi-threads
* L’utilisation de GitHub
* La gestion de projet avec IceScrum
* Les tests unitaires ainsi que Travis
* Les différents diagrammes

Le projet a été divisé en 4 sprints avec à chaque fois un entretien avec le « Project Owner » pour la validation des étapes.

Les dates des sprints ont été les suivantes :

* Sprint 1 : Du 1er Avril au 23 Avril 2020
* Sprint 2 : Du 24 Avril au 7 Mai 2020
* Sprint 3 : Du 8 Mai au 21 Mai 2020
* Sprint 4 : Du 22 Mai au 11 Juin 2020

Le choix de notre projet sur ce laps de temps n’est pas l’aisée au niveau de sa complexité, niveau interaction et jouabilité. L’objectif ici, est de s’approcher au maximum de la jouabilité « classique » d’un jeu de Poker afin d’en obtenir une version finale à la fin du projet.

# Règles du jeu « Poker Texas Hold’em »

Pour jouer à un jeu de Poker entre amis ou officiellement, il nous faut le strict minimum qui est :

* Un jeu de 52 cartes
* 2 joueurs minimum
* Des jetons afin de miser (différentes valeurs)

## Organisation

*« Pour commencer, vous devez désigner un donneur, afin de le distinguer des autres, le bouton doit être placé sur la table à côté de ce joueur. Le donneur est en charge de distribuer les cartes. A chaque tour, le donneur change dans le sens des aiguilles d’une montre.*

*Les deux joueurs situés à la gauche du donneur sont appelés petite blinde et grosse blinde. Ces deux joueurs sont obligés de miser une somme définie au départ entre l’ensemble des joueurs, cela avant même de distribuer les cartes.*

*Les joueurs qui sont petite blinde et grosse blinde change en même temps que le donneur, c’est-à-dire à chaque manche.*

*Les sommes misées pour la petite blinde et la grosse blinde augmentent au cours du jeu. Par exemple les joueurs peuvent se mettre d’accord pour augmenter les mises toutes les demi-heures.*

*Exemple : Selon la valeur que vous donnez pour chaque jeton, vous pouvez commencer par 5 et 10 pour la petite et grosse blinde, puis au bout d’une demi-heure, passer à 10 et 20. »*

## Début d’une partie

*« Le donneur distribue à chaque joueur deux cartes, une par une.*

*Le premier tour de mises démarre par le joueur immédiatement à la gauche de la grosse blinde. Ce joueur, en fonction des cartes qu’il a en main, peut :*

* ***Se coucher****. Il jette alors ses cartes et se retire définitivement de la partie.*
* ***Suivre la grosse blinde****, c’est-à-dire qu’il met dans le pot la même somme que celle déposée par la grosse blinde.*
* ***Relancer****. Il mise alors une somme supérieure à celle déposée par la grosse blinde.*

*Une fois que le premier joueur a pris sa décision, c’est au tour du joueur qui se trouve à sa gauche et ainsi de suite. La petite blinde et la grosse blinde peuvent soit s’aligner sur les mises des autres joueurs, soit relancer. Si relance il y a, un deuxième tour de mises s’effectue alors.*

*Lorsque le premier tour de mises est terminé, trois cartes sont retournées au centre de la table par le donneur. Ces cartes sont appelées le Flop.*

*A savoir : le donneur doit « griller » une carte à chaque fois qu’il doit poser des cartes, c’est-à-dire qu’il ne conserve pas la première carte du talon.*

*Le deuxième tour de mises démarre avec le joueur à gauche du donneur. Dans ce nouveau tour chaque joueur à la possibilité de dire :*

* ***Check****, le joueur peut cheker c’est-à-dire qu’il ne mise pas de somme supplémentaire dans le pot. Il peut cheker seulement si les joueurs d’avant n’ont pas misé eux non plus.*
* ***Miser****, il décide de mettre dans le pot la somme qu’il souhaite.*
* ***Relancer****, c’est-à-dire miser une somme plus importante si un joueur a déjà misé.*
* ***Suivre****, si un joueur a déjà misé, vous égalisé cette mise dans le pot.*
* ***Se coucher****. Il jette alors ses cartes et se retire définitivement de la partie.*

*Lorsque le deuxième tour de mises est terminé, une quatrième carte commune est retournée au centre de la table. Cette carte est appelée le Tournant.*

*Le troisième tour de mises peut commencer comme le précédent.*

*Lorsque ce tour est terminé, une cinquième carte commune appelée la Rivière est retournée au centre de la table. Une fois que les joueurs ont fini de miser, ceux qui sont toujours dans la course (c’est-à-dire les joueurs qui ne se sont pas couchés) montrent leurs cartes. C’est le joueur qui a la combinaison de cartes la plus importante qui gagne la manche, il remporte alors l’ensemble du pot.*

*Une seconde manche peut alors débuter.*

Les règles ci-dessous ont été reprises du site : <https://www.regles-de-jeux.com/regle-du-poker/>. Les adaptations pour notre projet seront expliqués au chapitre « Adaptation des règles ».

## Combinaison de cartes

Ci-dessous une image représentant les différentes combinaisons possibles au jeu du poker. Dans notre cas, entre les deux cartes en main d’un joueur et les 5 cartes de la table.



Figure 1 https://kingofpoker.org/wp-content/uploads/2014/10/mains-au-poker.png

# Adaptation des règles

La première approche au cours du projet a été de s’approcher le plus possible des règles de jeux classiques. Au fil du projet et à cause de la complexité ou encore du temps disponible pour toute l’implémentation et la sortie d’une release finale. Il a fallu adapter certaines règles qui génèrent une approche « à notre sauce » mais toujours dans le sens de la même jouabilité.

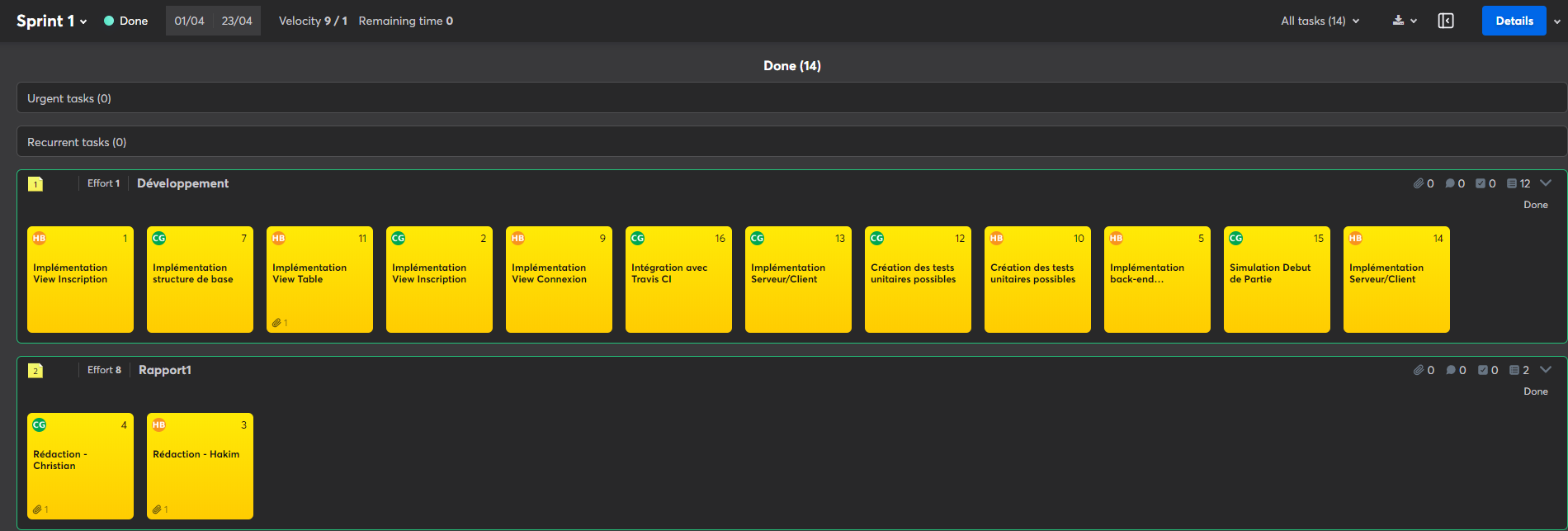
Les modifications suivantes ont été apportés : (avant-après)

# Gestion de projet

Dans le cadre de ce projet nous avons utilisé l’outil IceScrum. Celui-ci nous a permis de planifier diverses tâches réparties dans des « **Sprint** ».

Un « **Sprint** » est un ensemble de tâches découpées du projet. Celui-ci contient des « Stories » qui regroupent elles-mêmes un ensemble de tâches qui concernent une fonctionnalité du projet.

*Exemple d’un sprint*



Les « post-it » jaunes sont des **tâches**. On peut voir que celles du haut sont rattachées à la **story** « Développement » et le tout est lié au **sprint** « Sprint 1 ».

## Sprint 1

Concernant le premier sprint, nous nous sommes principalement concentrés sur la base du projet. C’est-à-dire avoir un environnement d’application fonctionnel graphique qui tourne. Nous avons donc créé les deux GUI principales lors de cette phase (GUI de la table de jeu et GUI de connexion).

Travis CI a été implémenté afin de pouvoir profiter d’un outil d’intégration continue, permettant ainsi de facilement lancer les tests de notre applications.

Nous avons notamment construit le squelette principal du rapport qui a été utilisé lors des différents rendus.

Le logo a lui aussi été créé durant cette phase.

## Sprint 2

Durant le deuxième sprint, nous avons établis un diagramme de communication ainsi qu’un diagramme de séquence. La GUI s’est vue modifiée afin de permettre une meilleure visualisation des joueurs et de leurs cartes respectives. On y implémenta également le fait de pouvoir miser / se coucher pour chaque joueur.

L’introduction de la communication Client- Serveur fût la tâche prenant le plus de temps. Celle-ci nous a permis de mettre en place une structure servant d’exemple afin de pouvoir ré—implémenter d’autre communications nécessaires au jeu (tel que le fait que les cartes du **Board**[[1]](#footnote-1) doivent êtres similaires pour tout le monde).

Afin de permettre une communication synchronisée entre le client et le serveur (tout ceci pendant que le jeu se déroule) nous avons mis en place différents **Thread**[[2]](#footnote-2) comme nous l’avons vu durant les différents cours de Génie logiciel.

Une première partie a pu être simulée (manuellement) afin de pouvoir visualiser les éventuels bogues / anormalités.

## Sprint 3

Lors du sprint 3, nous avons établis un diagramme d’activité.

Une gestion des différents états de la partie a notamment été introduit (reconnaissance de la phase actuelle : Flop[[3]](#footnote-3), River[[4]](#footnote-4), Turn[[5]](#footnote-5)). Nous avons notamment revu l’ensemble des actions possibles joueurs afin de pouvoir correctement les interprétés entre le back-end et le front-end. Ainsi un joueur pourra uniquement miser lors d’une phase de mise (qui est affichée sur la fenêtre).

Un pot commun aux joueurs a été implémenté. Celui-ci est doit être remporté par le gagnant de la partie.

## Sprint 4

Durant ce dernier sprint, nous avons ajouter diverses fonctionnalités. Le calcul de la force d’une main a été une tâche demandant beaucoup de temps d’analyse et d’implémentation.

Afin d’observer le déroulement d’une multitude de parties, nous avons fait en sorte qu’à la fin d’une partie, une nouvelle en est lancée. Cela nous a été utile car certaines fois il y eut des soucis indécelables (quint flush royale qui n’était pas prise en compte) si nous nous étions contentés de tester des parties manuellement. Ceci aurait été indétectable dans le sens où, il y a très peu de chance d’en avoir une. Le fait d’avoir pu automatiser cela nous a permis de jouer plus de parties.

Un modèle de domaine a été créé afin de permettre une meilleure visualisation du projet.

Pour finir, une rédaction finale du rapport fût effectuée.

# Modèle de domaine

1. Cartes communes à tous les joueurs. Elles se situent au milieu de la table. Il y en a cinq au total. [↑](#footnote-ref-1)
2. Un thread est une séquence d’instructions qui s’exécutent parallèlement à d’autres thread. Ceci est utile afin de permettre plusieurs actions simultanées. [↑](#footnote-ref-2)
3. Trois premières cartes du board [↑](#footnote-ref-3)
4. Quatrième carte du board [↑](#footnote-ref-4)
5. Cinquième carte du board [↑](#footnote-ref-5)