

EFREI – L3 new

Rapport d'avancement n°1

Projet Java 2

Thomas LINTANF - Batiste MONGAI – Camille Vinet
19/12/2019

Introduction

Ce document vise à rendre compte de l'état d'avancement du projet java. Il doit être réalisé régulièrement afin de suivre l'évolution du projet et détecter les éventuelles point de blocage. Il doit contenir un image du développement du projet à un moment donné, les problèmes rencontrés ainsi que les prochaines actions.

Table des matières

| | |
|--------------------------------|---|
| Introduction..... | 1 |
| I. Etat du projet..... | 2 |
| A. Conception | 2 |
| 1. Diagramme UML..... | 2 |
| B. Réalisation | 3 |
| II. Problèmes rencontrés | 3 |
| III. Prochaines actions..... | 3 |

I. Etat du projet

A. Conception

1. Diagramme UML

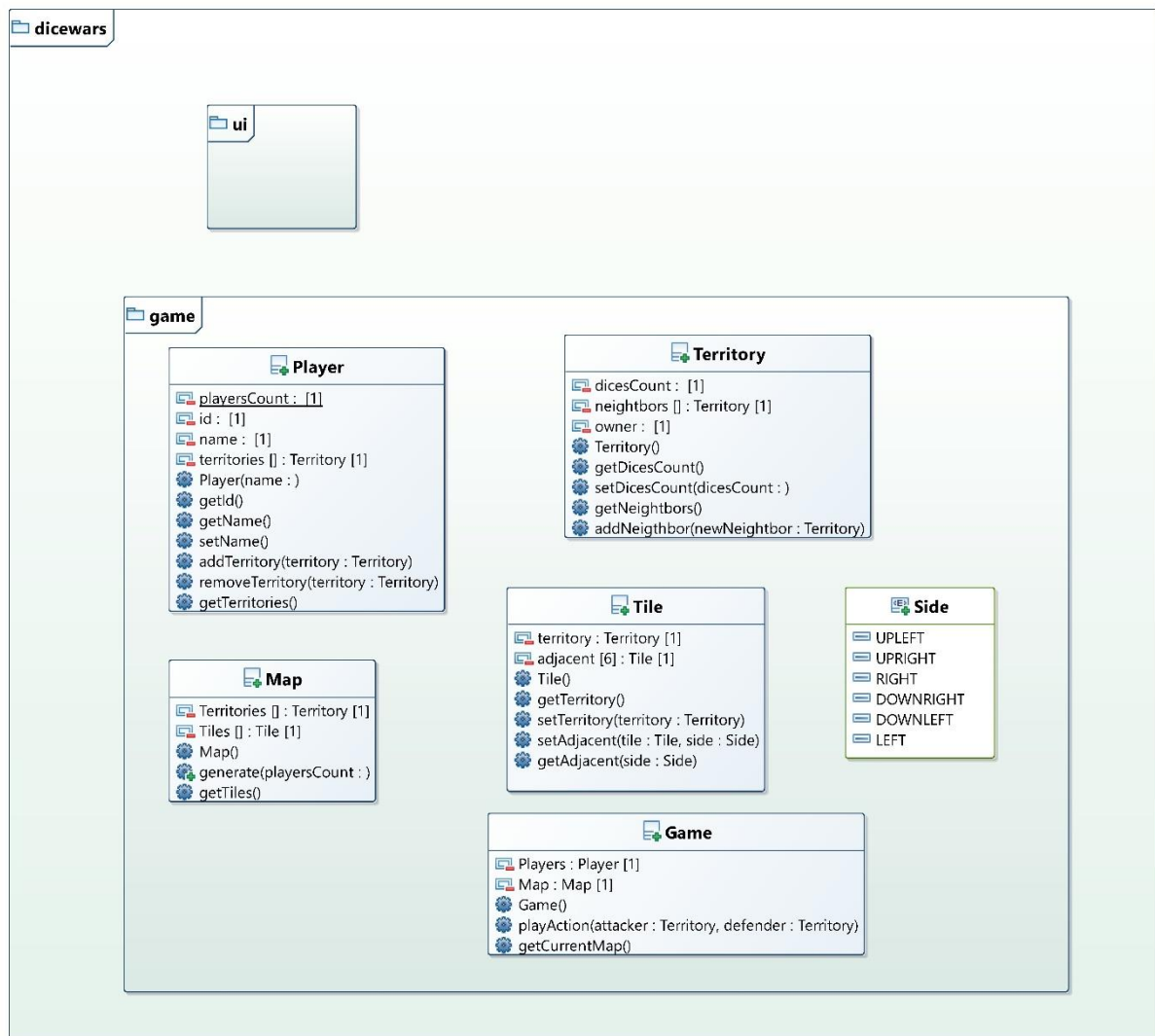


Figure 1 - Diagramme de classe

La conception du projet peut-être résumée par ce diagramme UML. Le projet est structuré en deux packages, le package game qui contient les classes fonctionnelle du projet et le package ui qui contient les classes d'interfaces graphiques. L'idée est qu'il soit possible de modifier ces deux partie indépendamment l'une de l'autre.

Seule une partie des classes est actuellement définie nous allons les voir en détails

La classe Player permet de stocker les informations de chaque joueur elle contient un identifiant unique, un nom et la liste des territoires du joueurs (stockage des références).

La classe Territory permet de stocker les informations d'un territoire, elle contient le nombre de dès du territoire, la liste de ses voisins (stockage des références) ainsi que la référence du joueur auquel il appartient.

La classe Tile permet de stocker une case de la carte, chaque case peut faire partie d'un territoire. Elle contient la référence de son territoire et les références de ses voisins (auquel on accède via un getter customiser qui utilise l'enum Side).

La classe Map représente la carte du jeu, celle-ci est composée de cases hexagonales qui forment des territoires. Elle contient la liste des cases (Tile) et des territoires (Territory). Elle est initialisée par la méthode generate(int) qui prend pour argument le nombre de joueurs dans la partie.

La classe Game est la classe qui représente une partie, elle contient la liste des joueurs et la carte

B. Réalisation

La réalisation n'a pas encore commencé

II. Problèmes rencontrés

Le principal défi rencontré pour l'instant a été d'imaginer une solution technique pour être capable de représenter les territoires de façon similaire à l'illustration. Pour y parvenir nous avons pensé au système de case. La carte est ainsi découpée en cases hexagonales. Lors de la génération de la carte un premier territoire composé d'un nombre aléatoire de cases contenu dans un intervalle à définir est placé de façon aléatoire. Les autres territoires sont ensuite insérés de façon aléatoire avec la contrainte que leur première case doit être adjacente à celle d'un territoire existant.

III. Prochaines actions

- Finir le diagramme UML → pour le 20 avant 15h
- Préparer le projet eclipse
- Mettre en place le repository GitHub
- Réunion de lancement 20/12 à 15h