



Little Town Dossier

Projet tutoré

Équipe 14

Dorian Lemerrier

Mathéo Meinerad

Quentin Savéan

Théo Lavie

Table des matières

INTRODUCTION.....	3
DIAGRAMME UML.....	4
SUIVI D'ACTIVITÉ.....	6
Diagramme de Gantt.....	7
Diagramme des ressources.....	7
DOCUMENTATION UTILISATEUR.....	8
JEUX DE TESTS.....	10
Scénario A : Lancement de la partie.....	11
1. Paramétrage(s) de la partie.....	12
Scénario B : Partie en cours.....	14
1. Gestion du menu.....	14
2. Gestion du score.....	15
3. Placement d'un ouvrier sur une tuile.....	16
4. Construction d'un bâtiment.....	17
5. Activer une tuile.....	19
6. Information(s) d'un bâtiment.....	20
7. Echanger les pièces contre des ressources.....	21
8. Gestion des cartes objectifs.....	22
Scénario C : Banque vide.....	23
Scénario D : Fin de manche.....	24
1. Passer au joueur suivant.....	24
2. Nourrir les ouvriers.....	24
Scénario E : Fin de la partie.....	26
1. Échanger les pièces du joueurs.....	26
2. Afficher et enregistrer les scores.....	27
3. Déterminer le gagnant.....	27
COMMENTAIRES.....	28
CONCLUSION.....	31

INTRODUCTION

Dans le cadre du projet tutoré de fin semestre 2, il nous a été demandé de réaliser un jeu de société du nom de "Little Town" au format numérique : console (CUI) et interface graphique (IHM). Il s'agit d'un projet de deux semaines, où à la fin, une démonstration du code ainsi qu'une soutenance orale se tiendra pour la présentation de celui-ci.

Nous avons tout d'abord procédé à une répartition des tâches à l'aide de Gantt projet dans vous trouverez les schémas par la suite. Au cours du développement, nous avons programmé une dizaine de classe dont vous trouverez le récapitulatif dans le digramme UML. Afin de tester notre programme et vérifier la fiabilité de ce dernier, nous avons rédigé une liste de tests et scénario que vous retrouverez également par la suite.

DIAGRAMME UML

SUIVI D'ACTIVITÉ

Diagramme de Gantt

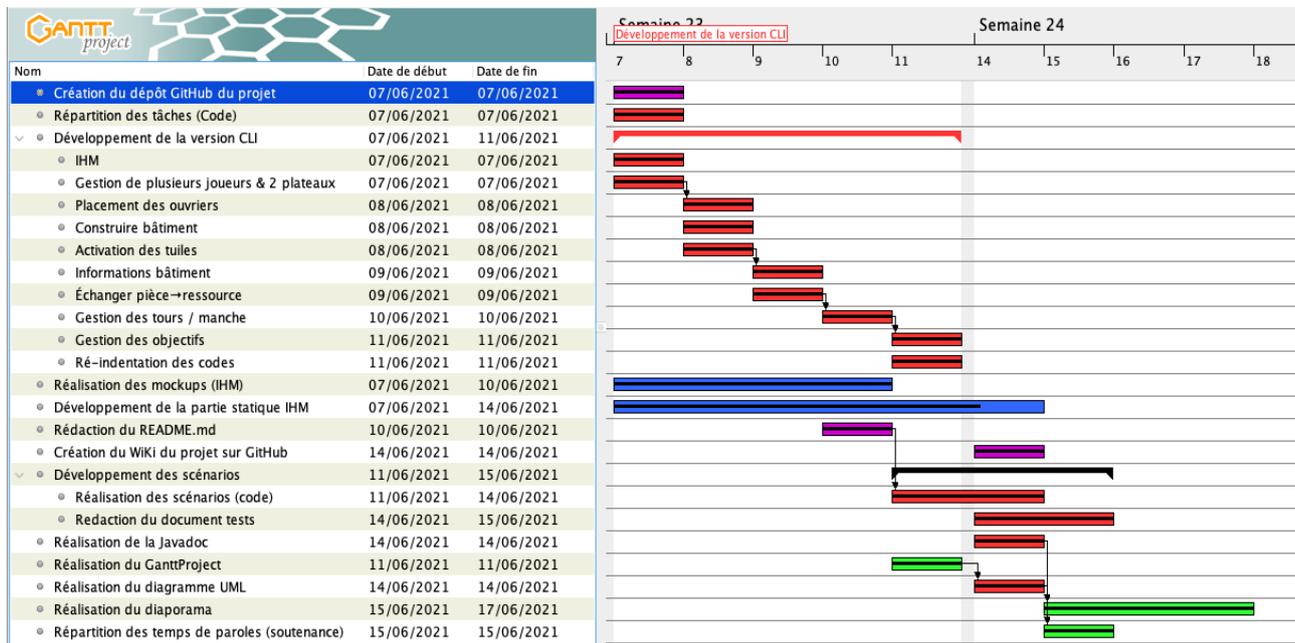
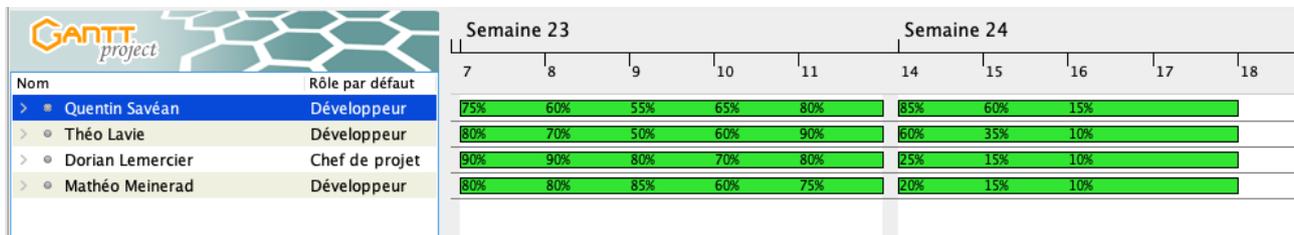


Diagramme des ressources



DOCUMENTATION UTILISATEUR

Placer ouvrier	<p>> Choix numéro 1 :</p> <p>> Rentrez les coordonnées entre [A,I] et [1,6]. <i>Exemple : B2</i></p>
Construire bâtiment	<p>> Choix numéro 2 :</p> <p>> Rentrez les coordonnées du bâtiment à construire dans plateau bas entre [C,H] et [7,8] mais aussi B7 pour le blé. > Rentrez les coordonnées ou vous voulez placer le bâtiment. <i>Exemple :</i> : B7 : B4</p>
Activer tuile	<p>> Choix numéro 3 :</p> <p>> Rentrez les coordonnées de la tuile à activer autour du dernier ouvrier que vous avez posé. <i>Exemple :</i> : B3</p>
Info bâtiment	<p>> Choix numéro 4 :</p> <p>> Rentrez les coordonnées de la tuile dont vous voulez connaître les informations. Cette tuile peut se trouver sur le plateau bas comme sur le plateau haut. <i>Exemple : G7</i></p>
Échanger pièce → ressource	<p>> Choix numéro 5 :</p> <p>Sélectionner le caractère représentant le ressource à recevoir : (On peut recevoir qu'une ressource à la fois) <i>Exemple : C</i></p>
Valider un objectif	<p>> Choix numéro 6 :</p> <p>Sélectionner le numéro de l'objectif à valider. <i>Exemple : 1 ou 2</i></p>

Pierre	P
Arbre	A
Eau	E
Céréale	C
Pièce	M

JEUX DE TESTS

Scénario A : Lancement de la partie

Lancer le scénario : `java littletown.cui.Controleur`

```
quentin@Air-de-Quentin cui % java littletown.cui.Controleur
Sur quel plateau voulez vous jouer ? [1 ou 2] : 1
Combien de joueurs voulez vous choisir : 4
Joueur n°1 quel est votre prenom ? : Quentin
Joueur n°2 quel est votre prenom ? : Théo
Joueur n°3 quel est votre prenom ? : Dorian
Joueur n°4 quel est votre prenom ? : Mathéo
> Plateau actuel :
  A   B   C   D   E   F   G   H   I
+---+---+---+---+---+---+---+---+
| P- |   | A- | P- |   | E- |   | P- | P- | 1
+---+---+---+---+---+---+---+---+
|   |   |   |   |   |   |   |   |   | 2
+---+---+---+---+---+---+---+---+
| E- |   |   |   |   |   |   |   |   | 3
+---+---+---+---+---+---+---+---+
| A- |   |   |   |   |   |   |   | E- | 4
+---+---+---+---+---+---+---+---+
|   |   |   |   |   |   | A- |   |   | 5
+---+---+---+---+---+---+---+---+
|   | A- | A- |   | E- | E- |   |   | A- | 6
+---+---+---+---+---+---+---+---+
+---+---+---+---+---+---+---+---+
| 5 | B | C | E | R | E | F | B |   | 7
+---+---+---+---+---+---+---+---+
| 0 | 0 | P | A | B | B | G | P |   | 8
+---+---+---+---+---+---+---+---+
| 0 | 0 |   |   |   |   |   |   |   |
+---+---+---+---+---+---+---+---+
```

1. Paramétrage(s) de la partie

Le programme demande, dès le lancement du jeu, les différentes caractéristiques de la partie telle que le nombre de joueurs, le numéro de plateau et leurs prénoms/pseudonymes.

Nous testerons une partie à 4 joueurs avec le plateau 1 et une de 2 joueurs avec le plateau 2.

Test 1.1

L'utilisateur saisit son numéro de plateau (1 ou 2).

Test 1.2

L'utilisateur saisit le nombre de joueurs présents dans la partie (comprise entre 2 et 4).

Test 1.3

L'utilisateur saisit les noms des joueurs selon leur nombre choisi.

Test 1.4

Le plateau saisi correspond à l'affichage du plateau actuel.

Plateau 1, 4 joueurs :

```
quentin@Air-de-Quentin cui % java littletown.cui.Controleur
[Sur quel plateau voulez vous jouer ? [1 ou 2] : 1
Combien de joueurs voulez vous choisir : 4
Joueur n°1 quel est votre prenom ? : Quentin
Joueur n°2 quel est votre prenom ? : Théo
Joueur n°3 quel est votre prenom ? : Dorian
Joueur n°4 quel est votre prenom ? : Mathéo
> Plateau actuel :
+---+
| A | B | C | D | E | F | G | H | I |
+---+
| P- |   | A- | P- |   | E- |   | P- | P- | 1
+---+
|   |   |   |   |   |   |   |   |   | 2
+---+
| E- |   |   |   |   |   |   |   |   | 3
+---+
| A- |   |   |   |   |   |   |   | E- | 4
+---+
|   |   |   |   |   |   | A- |   |   | 5
+---+
|   | A- | A- |   | E- | E- |   |   | A- | 6
+---+
+---+
| 5 | B | C | E | R | E | F | B |   | 7
+---+
| 0 | 0 | P | A | B | B | G | P |   | 8
+---+
| 0 | 0 |   |   |   |   |   |   |   |
+---+
```

Plateau 2, 2 joueurs :

```
Sur quel plateau voulez vous jouer ? [1 ou 2] : 2
Combien de joueurs voulez vous choisir : 2
Joueur n°1 quel est votre prenom ? : Quentin
Joueur n°2 quel est votre prenom ? : Théo
> Plateau actuel :
+---+
| A- |   |   | P- | P- |   | E- |   | A- | 1
+---+
|   |   | A- |   |   |   |   |   |   | 2
+---+
| A- |   |   |   |   |   |   |   | E- | 3
+---+
|   |   |   |   |   |   |   |   | A- | 4
+---+
| E- |   |   |   | A- |   |   |   |   | 5
+---+
| E- |   |   | P- |   |   | E- |   | P- | 6
+---+
+---+
| 5 | B | F | P | G | A | C | C |   | 7
+---+
| 0 | 0 | S | E | E | F | M | P |   | 8
+---+
```

Cas d'erreurs :**Test 1.5**

L'utilisateur saisit un numéro de plateau inférieur à 0 ou supérieur à 2 sinon le programme redemande la saisit.

Test 1.6

L'utilisateur saisit un nombre de joueurs inférieur à 2 ou supérieur à 4 ou une chaîne de caractère, le programme redemande la saisit.

Test 1.7

L'utilisateur saisit les prénoms des joueurs. Le nombre de prénoms à saisir est identique au nombre de joueur choisi dans la partie.

```
quentin@Air-de-Quentin cui % java littletown.cui.Controleur test
Sur quel plateau voulez vous jouer ? [1 ou 2] : 4
Sur quel plateau voulez vous jouer ? [1 ou 2] : -1
Sur quel plateau voulez vous jouer ? [1 ou 2] : █
```

```
Combien de joueurs voulez vous choisir : 6
Combien de joueurs voulez vous choisir : -1
Combien de joueurs voulez vous choisir : 0
Combien de joueurs voulez vous choisir : 1
Combien de joueurs voulez vous choisir : █
```

```
Combien de joueurs voulez vous choisir : 2
Joueur n°1 quel est votre prenom ? : Quentin
Joueur n°2 quel est votre prenom ? : Théo
```

Scénario B : Partie en cours

Commande pour lancer le scénario : `java littletown.cui.Controleur mp`

D'après le scénario, nous partons du principe que nous avons lancé une partie de quatre joueurs avec le plateau 1 et que nous sommes à la manche 2, comme suit :

```

cui - java littletown.cui.Controleur mp - 204x54
  A B C D E F G H I
  P- |  | A- | P- |  | E- |  | P- | P- | 1
  |  |  |  |  |  |  |  |  |  | 2
  E- | L4 |  |  | E2 |  |  |  |  | 3
  A- |  |  |  |  |  |  |  |  | E- | 4
  |  |  |  |  |  |  |  |  |  | 5
  A- | A- |  | E- | E- |  |  |  | A- | 6
  |  |  |  |  |  |  |  |  |  | 7
  |  |  |  |  |  |  |  |  |  | 8
  |  |  |  |  |  |  |  |  |  | 9
  |  |  |  |  |  |  |  |  |  | 0

> La manche actuelle est ( 2 / 4 ).
Banque
Nombre de Pierre : 15
Nombre de Arbre : 15
Nombre d'eau : 15
Nombre de Ble : 15
Nombre de Piece : 40

Joueur n°2, ( Dorian ), couleur : Bleu
Nombre de Pierre : 9
Nombre de Arbre : 1
Nombre d'eau : 1
Nombre de Ble : 2
Nombre de Piece : 13
Nombre de Point : 0

Objectif(s) restant(s):
[1] Construire 1 Bâtiment qui produit et, ou consomme de l'eau.
Cet objectif rapporte : 2 points.
[2] Construire 2 Bâtiments qui produisent et, ou consomment des pièces.
Cet objectif rapporte : 3 points.

> Action possibles:
1 - Placer Ouvrier
2 - Construire batiment
3 - Activer tuile
4 - Info Batiment
5 - Echanger pièce -> ressources
6 - Valider un objectif
7 - Fin de tour
  
```

1. Gestion du menu

Cas d'erreurs :

Test 1.1.

L'utilisateur saisie un nombre inférieur à 1, supérieur à 7 ou une chaîne de caractères redemande la saisie.

Test 1.2.

L'utilisateur ne peut pas passer son tour s'il celui-ci n'a pas placé d'ouvrier sur le plateau.

```

cui - java littletown.cui.Controleur - 84x28
Nombre de Arbre : 0
Nombre d'eau : 0
Nombre de Ble : 0
Nombre de Piece : 3
Nombre de Point : 0

Objectif(s) restant(s):
[1] Nourrir tous les Ouvriers avec de l'eau.
Cet objectif rapporte : 2 points.
[2] Activer 1 Bâtiment qui consomme 2 types de Ressource différents.
Cet objectif rapporte : 2 points.

> Action possibles:
1 - Placer Ouvrier
2 - Construire batiment
3 - Activer tuile
4 - Info Batiment
5 - Echanger pièce -> ressources
6 - Valider un objectif
7 - Fin de tour

Faites votre choix : 0
Faites votre choix : -1
Faites votre choix : test
Faites votre choix : 18
Faites votre choix :
  
```

2. Gestion du score

Quand l'utilisateur gagne du score (des points), celui-ci doit s'afficher et s'incrémenter par rapport aux bâtiments qu'ils construits, cartes objectifs qu'ils complètent ou bien par rapport à certaines tuiles qu'ils activent.

Par exemple, nous allons ici construire un bâtiment.

Test 2.1

L'utilisateur détient les ressources nécessaires à la construction de son bâtiment.

Test 2.2

L'utilisateur place un bâtiment.

Test 2.3

L'utilisateur gagne les points définis dans les informations du bâtiment et l'affiche dans le récapitulatif de ces ressources.

```
Faites votre choix : 2
Une position est constituer d'une lettre entre 'A' et 'I' et un chiffre entre 1 et 8
Entrez la position du batiment : C8
Une position est constituer d'une lettre entre 'A' et 'I' et un chiffre entre 1 et 6
Entrez la position de la tuile vide : D2
> Plateau actuel :
  A  B  C  D  E  F  G  H  I
+---+---+---+---+---+---+---+---+
| P- |   | A- | P- |   | E- |   | P- | P- | 1
+---+---+---+---+---+---+---+---+
|   |   |   | F2 |   |   |   |   | B1 | 2
+---+---+---+---+---+---+---+---+
| E- |   | B4 |   |   | P2 |   |   |   | 3
+---+---+---+---+---+---+---+---+
| A- |   |   |   |   |   |   |   | E- | 4
+---+---+---+---+---+---+---+---+
|   |   | A3 |   |   |   |   | A- |   | 5
+---+---+---+---+---+---+---+---+
|   | A- | A- |   | E- | E- |   |   | A- | 6
+---+---+---+---+---+---+---+---+
| 5  | B  |   | C  | E  |   | P  | G  |   | 7
+---+---+---+---+---+---+---+---+
| 1  | 0  |   |   |   |   | P  | B  | C  | 8
+---+---+---+---+---+---+---+---+
| 0  | 0  |   |   |   |   |   |   |   |
+---+---+---+---+---+---+---+---+
```

```
Nombre de Pierre : 999
Nombre de Arbre  : 1000
Nombre d'eau     : 1000
Nombre de Ble    : 1001
Nombre de Piece  : 1012
Nombre de Point  : 0
```

```
Nombre de Pierre : 999
Nombre de Arbre  : 996
Nombre d'eau     : 1000
Nombre de Ble    : 1001
Nombre de Piece  : 1012
Nombre de Point  : 6
```

3. Placement d'un ouvrier sur une tuile

Le joueur pourra placer un ouvrier sur une tuile vide à chaque tour afin de collecter les différentes ressources environnantes depuis le plateau du haut.

On testera les différentes positions possibles et traiterons les cas d'erreurs.

<p>Test 3.1 L'utilisateur saisit la position de l'ouvrier sur le plateau (menu 1).</p> <p>Test 3.2 L'ouvrier est placé sur la tuile vide souhaitée (ici B2).</p> <p>Test 3.3 On précise que la tuile possède un ouvrier (O) et son propriétaire (Joueur 1 ici).</p>	<p>Ouvrier placé (B2) :</p> <pre> Faites votre choix : 1 Une position est constituer d'une lettre entre 'A' et 'I' et un chiffre entre 1 et 6 Entrez la position de la tuile vide : B2 > Plateau actuel : A B C D E F G H I +---+---+---+---+---+---+---+---+ P- A- P- E- P- P- 1 +---+---+---+---+---+---+---+---+ O1 2 +---+---+---+---+---+---+---+---+ E- 3 +---+---+---+---+---+---+---+---+ A- E- 4 +---+---+---+---+---+---+---+---+ A- 5 +---+---+---+---+---+---+---+---+ A- A- E- E- A- 6 +---+---+---+---+---+---+---+---+ 5 B S B E C P M 7 +---+---+---+---+---+---+---+---+ 0 0 G B F E P R 8 +---+---+---+---+---+---+---+---+ </pre>
--	---

Cas d'erreurs :

Nous avons le plateau suivant :

	A	B	C	D	E	F	G	H	I	
1	P-		A-	P-		E-		P-	P-	
2								F1		
3	E-		C4			B2				
4	A-								E-	
5			B3				A-			
6		A-	A-		E-	E-			A-	

<p>Test 3.4 L'utilisateur ne peut placer d'ouvrier sur une tuile comportant un bâtiment, une ressource ou un ouvrier.</p> <p>Test 3.5 L'utilisateur ne peut placer d'ouvrier à l'extérieur du plateau.</p> <p>Test 3.6 L'utilisateur ne peut placer d'ouvrier sur le plateau bas.</p>	<pre> Faites votre choix : 1 Une position est constituer d'une lettre entre 'A' et 'I' et un chiffre entre 1 et 6 Entrez la position de la tuile vide : C3 Une position est constituer d'une lettre entre 'A' et 'I' et un chiffre entre 1 et 6 Entrez la position de la tuile vide : █ Faites votre choix : 2 Une position est constituer d'une lettre entre 'A' et 'I' et un chiffre entre 1 et 6 Entrez la position de la tuile vide : A7 Une position est constituer d'une lettre entre 'A' et 'I' et un chiffre entre 1 et 6 Entrez la position de la tuile vide : █ Faites votre choix : 3 Entrez la position de la tuile vide : E8 Une position est constituer d'une lettre entre 'A' et 'I' et un chiffre entre 1 et 6 Entrez la position de la tuile vide : █ </pre>
--	---

4. Construction d'un bâtiment

L'utilisateur pourra placer à chaque tour un bâtiment, s'il détient les ressources nécessaires à sa construction sur une tuile vide.

Test 4.1

L'utilisateur construit un bâtiment sur une tuile vide.

Test 4.2

L'utilisateur perd les ressources échangées contre la construction du bâtiment.

Test 4.3

L'utilisateur gagne les points du bâtiment.

```
Faites votre choix : 2
Une position est constituer d'une lettre entre 'A' et 'I' et un chiffre entre 1 et 8
Entrez la position du batiment : C8
Une position est constituer d'une lettre entre 'A' et 'I' et un chiffre entre 1 et 6
Entrez la position de la tuile vide : D2
> Plateau actuel :
```

A	B	C	D	E	F	G	H	I	
P-	A-	P-		E-		P-	P-		1
			F2				B1		2
E-	B4			P2					3
A-							E-		4
		A3				A-			5
	A-	A-		E-	E-		A-		6
5	B		C	E		P	G		7
1	0					P	B	C	8
0	0								

```
Nombre de Pierre : 999
Nombre de Arbre : 1000
Nombre d'eau : 1000
Nombre de Ble : 1001
Nombre de Piece : 1012
Nombre de Point : 0
```

```
Nombre de Pierre : 999
Nombre de Arbre : 996
Nombre d'eau : 1000
Nombre de Ble : 1001
Nombre de Piece : 1012
Nombre de Point : 6
```

Cas d'erreurs :

Test 4.4

L'utilisateur ne peut placer d'ouvrier sur le plateau haut avec un bâtiment.

Test 4.5

L'utilisateur ne peut construire un bâtiment si celui-ci ne détient pas les ressources nécessaires.

Test 4.6

L'utilisateur ne peut placer un bâtiment sur une tuile déjà occupée.

```
Faites votre choix : 2
Une position est constituer d'une lettre entre 'A' et 'I' et un chiffre entre 1 et 8
Entrez la position du batiment : E7
Une position est constituer d'une lettre entre 'A' et 'I' et un chiffre entre 1 et 6
Entrez la position de la tuile vide : E2
> Plateau actuel :
  A  B  C  D  E  F  G  H  I
+---+---+---+---+---+---+---+---+
| P- |  | A- | P- |  | E- |  | P- | P- | 1
+---+---+---+---+---+---+---+---+
|  |  |  |  |  | P2 |  | G1 |  | 2
+---+---+---+---+---+---+---+---+
| E- |  | A4 |  |  | R2 |  |  |  | 3
+---+---+---+---+---+---+---+---+
| A- |  |  |  |  |  |  |  | E- | 4
+---+---+---+---+---+---+---+---+
|  |  | E3 |  |  |  | A- |  |  | 5
+---+---+---+---+---+---+---+---+
|  | A- | A- |  | E- | E- |  |  | A- | 6
+---+---+---+---+---+---+---+---+
| 5 | B |  | P |  |  | E | B | 7
+---+---+---+---+---+---+---+---+
| 1 |  | S |  |  | C | E | B | 8
+---+---+---+---+---+---+---+---+
|  |  |  |  |  |  |  |  |  |
+---+---+---+---+---+---+---+---+
```

```
Faites votre choix : 1
Vous avez déjà réalisé votre action.
```

```
Faites votre choix : 2
Une position est constituer d'une lettre entre 'A' et 'I' et un chiffre entre 1 et 8
Entrez la position du batiment : G8
Une position est constituer d'une lettre entre 'A' et 'I' et un chiffre entre 1 et 6
Entrez la position de la tuile vide : E2
> Plateau actuel :
  A  B  C  D  E  F  G  H  I
+---+---+---+---+---+---+---+---+
| P- |  | A- | P- |  | E- |  | P- | P- | 1
+---+---+---+---+---+---+---+---+
|  |  |  |  |  |  |  | P1 |  | 2
+---+---+---+---+---+---+---+---+
| E- |  | L4 |  |  | B2 |  |  |  | 3
+---+---+---+---+---+---+---+---+
| A- |  |  |  |  |  |  |  | E- | 4
+---+---+---+---+---+---+---+---+
|  |  | E3 |  |  |  | A- |  |  | 5
+---+---+---+---+---+---+---+---+
|  | A- | A- |  | E- | E- |  |  | A- | 6
+---+---+---+---+---+---+---+---+
| 5 | B |  | R | G |  | P | C | 7
+---+---+---+---+---+---+---+---+
|  |  | A |  |  | F | E | B | 8
+---+---+---+---+---+---+---+---+
|  |  |  |  |  |  |  |  |  |
+---+---+---+---+---+---+---+---+
```

```
Faites votre choix : 2
Une position est constituer d'une lettre entre 'A' et 'I' et un chiffre entre 1 et 8
Entrez la position du batiment : D7
Une position est constituer d'une lettre entre 'A' et 'I' et un chiffre entre 1 et 6
Entrez la position de la tuile vide : F3
Une position est constituer d'une lettre entre 'A' et 'I' et un chiffre entre 1 et 6
Entrez la position de la tuile vide :
```

5. Activer une tuile

L'utilisateur pourra activer une tuile si celui-ci le décide depuis le menu « **activer tuile** » et si celui-ci se situe à côté de celle-ci en détenant les ressources nécessaires pour l'activer.

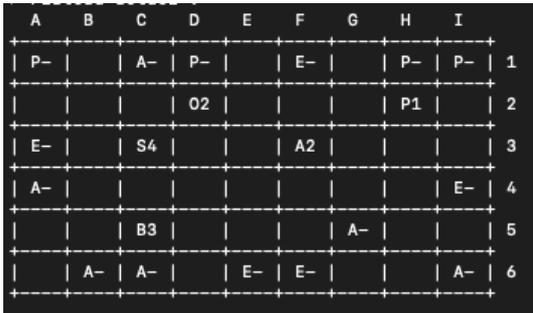
Une tuile correspond à une case du plateau.

Nous activerons une tuile d'un bâtiment. Nous partons du principe que l'utilisateur a les ressources nécessaires pour l'activer.

Soit le plateau suivant où nous sommes le jouer 2 :

```

> Plateau actuel :
  A  B  C  D  E  F  G  H  I
+---+---+---+---+---+---+---+---+
| P- |   | A- | P- |   | E- |   | P- | P- | 1
+---+---+---+---+---+---+---+---+
|   |   |   |   |   |   |   | P1 |   | 2
+---+---+---+---+---+---+---+---+
| E- |   | S4 |   |   | A2 |   |   |   | 3
+---+---+---+---+---+---+---+---+
| A- |   |   |   |   |   |   |   | E- | 4
+---+---+---+---+---+---+---+---+
|   |   | B3 |   |   |   |   | A- |   | 5
+---+---+---+---+---+---+---+---+
|   | A- | A- |   | E- | E- |   |   | A- | 6
+---+---+---+---+---+---+---+---+
    
```

<p>Test 5.1 L'ouvrier se situe autour d'une tuile.</p> <p>Test 5.2 L'utilisateur sélectionne le menu « 3 » pour activer une tuile.</p> <p>Test 5.3 Saisis la position sur le plateau de/des tuile(s) à activer.</p> <p>Test 5.4 Gagne les ressources ou points que procure celle-ci.</p>	 <pre> Faites votre choix : 3 Une position est constituer d'une lettre entre 'A' et 'I' et un chiffre entre 1 et 8 Entrez la position du batiment : C1 > Activation de la tuile : Arbre Une position est constituer d'une lettre entre 'A' et 'I' et un chiffre entre 1 et 8 Entrez la position du batiment : D1 > Activation de la tuile : Pierre </pre>
--	--

Cas d'erreurs :

<p>Test 5.5 L'utilisateur ne peut activer une tuile s'il y en a aucune autour de lui.</p> <p>Test 5.6 L'utilisateur ne peut activer une tuile bâtiment si celui-ci ne possède pas les ressources nécessaires.</p>	<pre> Faites votre choix : 3 Une position est constituer d'une lettre entre 'A' et 'I' et un chiffre entre 1 et 8 Entrez la position du batiment : F5 Une position est constituer d'une lettre entre 'A' et 'I' et un chiffre entre 1 et 8 Entrez la position du batiment : █ Une position est constituer d'une lettre entre 'A' et 'I' et un chiffre entre 1 et 8 Entrez la position du batiment : I7 Une position est constituer d'une lettre entre 'A' et 'I' et un chiffre entre 1 et 8 Entrez la position du batiment : █ </pre>
---	--

Test 5.7

L'utilisateur ne peut activer sur une position inexistante de tuile.

6. Information(s) d'un bâtiment

L'utilisateur pour construire un bâtiment où savoir quelles ressources ou points qu'il gagnera en construisant celui-ci où en l'activant pourra consulter ces diverses informations depuis le menu « info bâtiment ».

Soit le plateau suivant :

A	B	C	D	E	F	G	H	I	
P-		A-	P-		E-		P-	P-	1
							C1		2
E-		G4			S2				3
A-								E-	4
		L3				A-			5
	A-	A-		E-	E-			A-	6

Test 6.1

L'utilisateur sélectionne le menu « info bâtiment ».

Test 6.2

L'utilisateur saisit la position du bâtiment.

Test 6.3

Les informations du bâtiment s'affichent.

```
Faites votre choix : 4
Une position est constituer d'une lettre entre 'A' et 'I' et un chiffre entre 1 et 8
Entrez la position du batiment : D7

Nom du batiment   : Foire
Cout              : A4
Cout de Production : P1A1C1E1
Revient Production : S7
Gain de score    : 6
```

Test 6.4

Le programme redemande là saisi à l'utilisateur si celui-ci saisit une position ou il n'y a aucun bâtiment.

```
Faites votre choix : 4
Une position est constituer d'une lettre entre 'A' et 'I' et un chiffre entre 1 et 8
Entrez la position du batiment : E1
Une position est constituer d'une lettre entre 'A' et 'I' et un chiffre entre 1 et 8
Entrez la position du batiment : █
```

7. Echanger les pièces contre des ressources.

L'utilisateur pourra échanger ces pièces contre des ressources (blé, bois, eau, pierre...) à chaque tour.

Ici nous échangerons 2 blés contre 6 pièces. 1 ressource = 3 pièces. Nous partons du principe pour les cas de tests hors cas d'erreurs qu'ils possèdent les pièces nécessaires à l'échange.

<p>Test 7.1 L'utilisateur sélectionne le menu «Echanger pièce -> ressources » en saisissant le menu « 5 ».</p> <p>Test 7.2 L'utilisateur saisit la ressource qu'ils souhaitent échanger (C1 (blé), E1 (eau), P1 (pierre), A1 (arbre)). Ici C1.</p> <p>Test 7.3 L'utilisateur a bien été créditée de son blé et débité de ces pièces.</p>	<pre>Joueur n°2, (Quentin), couleur : Bleu Nombre de Pierre : 0 Nombre de Arbre : 1 Nombre d'eau : 1 Nombre de Ble : 2 Nombre de Piece : 13 Nombre de Point : 0</pre> <pre>Faites votre choix : 5 Entrez Ressources : C1</pre> <pre>Joueur n°2, (Quentin), couleur : Bleu Nombre de Pierre : 0 Nombre de Arbre : 1 Nombre d'eau : 1 Nombre de Ble : 3 Nombre de Piece : 10 Nombre de Point : 0</pre>
--	--

Cas d'erreurs :

Test 7.4

L'utilisateur ne peut qu'échanger une ressource par une ressource.

Test 7.5

L'utilisateur ne peut échanger une ressource si celui-ci ne possède pas les pièces nécessaires.

```
Faites votre choix : 5
Entrez Ressources : C3
```

```
Joueur n°2, ( Quentin ), couleur : Bleu
Nombre de Pierre : 0
Nombre de Arbre : 1
Nombre d'eau : 1
Nombre de Ble : 4
Nombre de Piece : 7
Nombre de Point : 0
```

```
Joueur n°2, ( Quentin ), couleur : Bleu
Nombre de Pierre : 0
Nombre de Arbre : 1
Nombre d'eau : 1
Nombre de Ble : 6
Nombre de Piece : 1
Nombre de Point : 0
```

```
Faites votre choix : C5
Faites votre choix : █
```

Piste d'améliorations :

Dans notre programme nous ne pouvons pas échanger un ensemble de ressources dans une seule saisie (exemple C5 pour 5 blés), il n'est possible que d'échanger une unité de ressource par saisie. Il pourrait être intéressant de le faire dans une prochaine version.

8. Gestion des cartes objectifs**Test 8.1**

L'utilisateur active la carte objectif

Test 8.2

L'utilisateur reçoit les points de la carte

Test 8.3

La carte disparaît du menu du joueur

```
Joueur n°1, ( Mathéo ), couleur : Rouge
Nombre de Pierre : 0
Nombre de Arbre : 0
Nombre d'eau : 0
Nombre de Ble : 0
Nombre de Piece : 3
Nombre de Point : 0
```

```
Objectif(s) restant(s):
[1] Construire 3 Bâtiments, respectivement adjacents ou en diagonale d'un Lac, d'une Montagne et d'une Forêt.
Cet objectif rapporte : 3 points.
[2] Construire 1 Bâtiment dont le coût est d'au moins 3 pierre.
Cet objectif rapporte : 2 points.
```

```
> Action possibles:
1 - Placer Ouvrier
2 - Construire bâtiment
3 - Activer tuile
4 - Info Bâtiment
5 - Echanger pièce -> ressources
6 - Valider un objectif
7 - Fin de tour
```

```
Faites votre choix : 6
Quel est le numéro de l'objectif ? [1 ou 2] : 1
```

```
Joueur n°1, ( Mathéo ), couleur : Rouge
Nombre de Pierre : 0
Nombre de Arbre : 0
Nombre d'eau : 0
Nombre de Ble : 0
Nombre de Piece : 3
Nombre de Point : 3
```

```
Objectif(s) restant(s):
[2] Construire 1 Bâtiment dont le coût est d'au moins 3 pierre.
Cet objectif rapporte : 2 points.
```

Scénario C : Banque vide

Dans ce scénario, nous simulerons une banque vide.

Commande pour lancer le scénario : `java littletown.cui.Controlleur bv`

Test 1.1

Construire un bâtiment ajoute des ressources à la banque.

Test 1.2

On ne peut pas recevoir des ressources de la banque s'il elle est vide.

```

Entrez la position du bâtiment : A1
> Activation de la tuile : Pierre
> Plateau actuel :
+---+
| P- | | A- | P- | | E- | | P- | P- | 1
+---+
|   | O1 |   |   |   |   |   |   | 2
+---+
| E- |   |   |   |   |   |   |   | 3
+---+
| A- |   |   |   |   |   |   | E- | 4
+---+
|   |   |   |   |   | A- |   |   | 5
+---+
| A- | A- |   | E- | E- |   | A- |   | 6
+---+

+---+
| 5 | B | A | B | C | G | F | P | 7
+---+
| 0 | 0 | P | M | C | B | S | E | 8
+---+

> La manche actuelle est ( 1 / 4 ).
Banque
Nombre de Pierre   : 0
Nombre de Arbre   : 0
Nombre d' Eau     : 0
Nombre de Céréale : 0
Nombre de Piece   : 86

Joueur n°1, ( Theo ), couleur : Rouge
Nombre de Pierre   : 0
Nombre de Arbre   : 0

```

```

Faites votre choix : 2
Une position est constituer d'une lettre entre '
Entrez la position du bâtiment : B7
Une position est constituer d'une lettre entre '
Entrez la position de la tuile vide : B2
> Plateau actuel :
+---+
| P- | | A- | P- | | E- | | P- | P- | 1
+---+
|   | B1 |   |   |   |   |   |   | 2
+---+
| E- |   |   |   |   |   |   |   | 3
+---+
| A- |   |   |   |   |   |   | E- | 4
+---+
|   |   |   |   |   | A- |   |   | 5
+---+
| A- | A- |   | E- | E- |   | A- |   | 6
+---+

+---+
| 4 | B | B | S | F | B | B | M | 7
+---+
| 1 | 0 | A | P | C | R | P | L | 8
+---+

> La manche actuelle est ( 1 / 4 ).
Banque
Nombre de Pierre   : 0
Nombre de Arbre   : 1
Nombre d' Eau     : 0
Nombre de Céréale : 0
Nombre de Piece   : 86

```

Scénario D : Fin de manche

Dans ce scénario, nous simulerons une fin de manche, c'est-à-dire au moment de nourrir les ouvriers et passer à la prochaine manche une fois cela fait.

Commande pour lancer le scénario : `java littletown.cui.Controleur no`

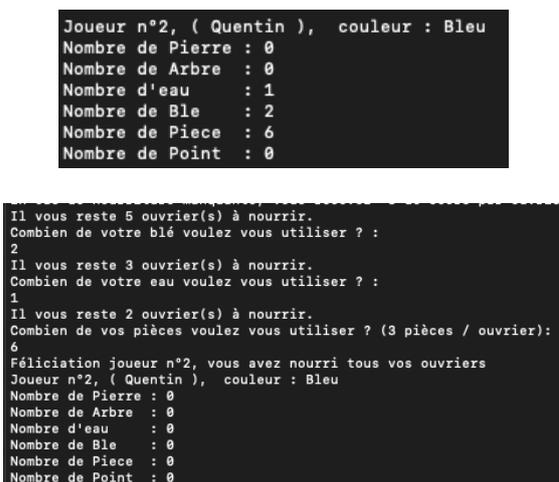
1. Passer au joueur suivant

Nous ferons en sorte que le même joueur ne puisse commencer à chaque début de manche. Cela tournera au fur et à mesure des tours.

<p>Test 1.1 L'utilisateur change à chaque début de nouvelle manche.</p>	
--	--

2. Nourrir les ouvriers

À chaque fin de manche, les joueurs devront nourrir leurs ouvriers en utilisant leurs ressources pour chacun d'entre eux (soit du blé ou/et de l'eau) ou 3 pièces pour obtenir l'une de ces ressources s'il n'a pas assez sinon il perd des points. Nous partons en principe que nous sommes en fin de manche d'après le scénario pour une partie **de deux joueurs**.

<p>Test 2.1 L'utilisateur saisit le blé qu'il souhaite échanger et son eau.</p> <p>Test 2.2 Il manque une ressource au joueur, l'utilisateur saisit le nombre de pièces qu'il souhaite échanger (soit 3 pièces = 1 ressource).</p> <p>Test 2.3 L'utilisateur perd des points s'il ne parvient pas à nourrir ces ouvriers (3 points par ouvrier non nourri). Le score peut être négatif.</p>	 <pre> Joueur n°2, (Quentin), couleur : Bleu Nombre de Pierre : 0 Nombre de Arbre : 0 Nombre d'eau : 1 Nombre de Ble : 2 Nombre de Piece : 6 Nombre de Point : 0 Il vous reste 5 ouvrier(s) à nourrir. Combien de votre blé voulez vous utiliser ? : 2 Il vous reste 3 ouvrier(s) à nourrir. Combien de votre eau voulez vous utiliser ? : 1 Il vous reste 2 ouvrier(s) à nourrir. Combien de vos pièces voulez vous utiliser ? (3 pièces / ouvrier): 6 Félicitation joueur n°2, vous avez nourri tous vos ouvriers Joueur n°2, (Quentin), couleur : Bleu Nombre de Pierre : 0 Nombre de Arbre : 0 Nombre d'eau : 0 Nombre de Ble : 0 Nombre de Piece : 0 Nombre de Point : 0 </pre>
--	---

Cas d'erreurs :

<p>Test 2.4</p>	
------------------------	--

L'utilisateur ne peut nourrir un ouvrier s'il n'a pas assez de ressources. Il perd alors des points (3 points par ouvrier non nourri).

Test 2.5

L'utilisateur peut échanger uniquement par tranche de 3 pièces en prenant en compte la/les ressources nécessaires à échanger pour nourrir l'ouvrier.

```
Il vous reste 5 ouvrier(s) à nourrir.
Combien de votre blé voulez vous utiliser ? :
0
Il vous reste 5 ouvrier(s) à nourrir.
Combien de votre eau voulez vous utiliser ? :
0
Il vous reste 5 ouvrier(s) à nourrir.
Combien de vos pièces voulez vous utiliser ? (3 pièces / ouvrier):
0
Vous n'avez pas nourri 5 ouvrier(s)
Vous allez recevoir -15 de score.
```

```
Vous allez recevoir -15 de score.
Joueur n°2, ( Quentin ), couleur : Bleu
Nombre de Pierre : 0
Nombre de Arbre : 0
Nombre d'eau : 1
Nombre de Ble : 2
Nombre de Piece : 6
Nombre de Point : -15
```

```
Il vous reste 5 ouvrier(s) à nourrir.
Combien de votre blé voulez vous utiliser ? :
2
Il vous reste 3 ouvrier(s) à nourrir.
Combien de votre eau voulez vous utiliser ? :
1
Il vous reste 2 ouvrier(s) à nourrir.
Combien de vos pièces voulez vous utiliser ? (3 pièces / ouvrier):
4
Vous n'avez pas nourri 1 ouvrier(s)
Vous allez recevoir -3 de score.
Joueur n°2, ( Quentin ), couleur : Bleu
Nombre de Pierre : 0
Nombre de Arbre : 0
Nombre d'eau : 0
Nombre de Ble : 0
Nombre de Piece : 3
Nombre de Point : -3
```

Scénario E : Fin de la partie

Dans ce scénario, nous simulerons la fin de partie avec les différents événements qui en suit.

Commande pour lancer le scénario : **java littletown.cui.Controleur fp**

1. Échanger les pièces du joueurs

En fin de partie, dès l'affichage du classement, le programme échangera automatiquement les pièces des joueurs contre des points (soit 3 pièces = 1 point).

<p>Test 1.1 Les joueurs finissent de nourrir leurs ouvriers.</p> <p>Test 1.2 Leurs pièces sont converties en point(s).</p>	<pre> Il vous reste 5 ouvrier(s) à nourrir. Combien de votre blé voulez vous utiliser ? : 2 Il vous reste 3 ouvrier(s) à nourrir. Combien de votre eau voulez vous utiliser ? : 1 Il vous reste 2 ouvrier(s) à nourrir. Combien de vos pièces voulez vous utiliser ? (3 pièces / ouvrier): 6 Félicitation joueur n°2, vous avez nourri tous vos ouvriers Joueur n°2, (Quentin), couleur : Bleu Nombre de Pierre : 0 Nombre de Arbre : 1 Nombre d'eau : 0 Nombre de Ble : 0 Nombre de Piece : 7 Nombre de Point : 0 Premier : Théo avec un score de 2 Deuxieme : Quentin avec un score de 2 </pre>
--	--

Cas d'erreurs

<p>Test 1.3 Le joueur peut gagner un point par tranche de 3 pièces.</p>	
--	--

2. Afficher et enregistrer les scores

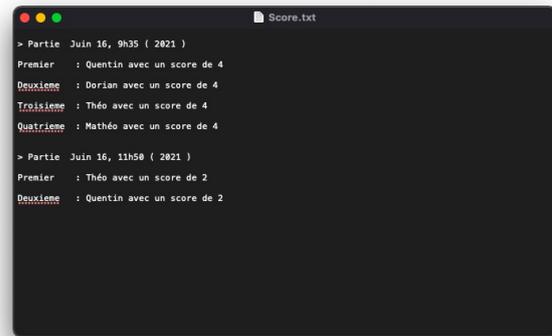
Dès la fin de la partie et l'affichage du classement, nous enregistrons le score de chaque joueur dans un fichier « score.txt » avec les noms des joueurs, leurs triés par ordre du premier au dernier avec la date de la partie comprise avec.

Test 2.1

Les scores s'enregistrent dans un fichier « score.txt ».

Test 2.2

L'utilisateur peut consulter le fichier depuis le répertoire « Code/data/score.txt ».



```
Score.txt
> Partie Juin 16, 9h35 ( 2021 )
Premier : Quentin avec un score de 4
Deuxieme : Dorian avec un score de 4
Troisieme : Théo avec un score de 4
Quatrieme : Mathéo avec un score de 4

> Partie Juin 16, 11h50 ( 2021 )
Premier : Théo avec un score de 2
Deuxieme : Quentin avec un score de 2
```

3. Déterminer le gagnant

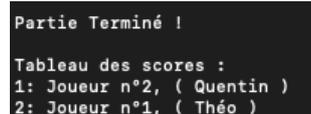
L'utilisateur ayant cumulé le plus de points est désignée grand bâtisseur du jeu (soit le gagnant).

Test 3.1

Les pièces sont converties en points.

Test 3.2

Les scores s'affichent par classement des joueurs (du gagnant aux perdants).



```
Partie Terminé !
Tableau des scores :
1: Joueur n°2, ( Quentin )
2: Joueur n°1, ( Théo )
```

COMMENTAIRES

Théo	<p>Ce projet de fin de première année a été très enrichissant. Le travail de groupe en présentiel était agréable. La liberté du sujet nous a permis de confectionner notre propre ihm et pas beaucoup de choix stratégique. Nous aurons aimé confectionner une version GUI mais malheureusement nous avons manqué de temps.</p> <p>Utiliser git a été un atout. Le partage de code était beaucoup plus facile ainsi que le retour à un précédent backup en cas de problème.</p> <p>Dans nos précédents projets le partage de code était parfois une tâche délicate. Github nous a perdu de résoudre ce problème. De plus la fonctionnalité "issues" nous a permis de facilement nous répartir les tâches.</p> <p>Bien connaître cet outil est un avantage dans le monde professionnel.</p> <p>Une bonne cohésion de groupe, un projet dynamique. La réalisation de ce dernier en présentiel était agréable.</p>
Mathéo	<p>J'ai rencontré des difficultés à rester concentré dû au bruit et au nombre de personnes venue pour savoir où on en était le projet.</p> <p>Sinon en général, je n'ai pas rencontré de difficulté.</p> <p>J'ai une suggestion pour le projet peut être qu'on pourrait utiliser une autre plateforme pour coder, car avec GitHub il peut y avoir des problèmes. Je maîtrise beaucoup mieux GitHub grâce au projet et je me suis amélioré pour réfléchir à comment aborder un sujet.</p>
Quentin	<p>Ce projet fut enrichissant que ça soit au niveau technique ou communicationnel. Nous avons pu restituer nos connaissances entre membres afin de mener à bien ce projet tout en répartissant les tâches de manières équitables. L'ambiance était bonne.</p> <p>Nous avons manqué un peu de temps sur certains points notamment pour la réalisation du GUI.</p>

	<p>Cela m'a apporté des connaissances en plus que ça soit au niveau du code ou au niveau du travail d'équipe qui fut plus importante en présentiel qu'en distanciel par rapport au premier projet de S1.</p> <p>L'utilisation d'outils comme Git nous a permis de partager nos codes, répertorié les bugs et tâches à faire assez facilement malgré les quelques conflits entre les modifications de deux mêmes fichiers. Le système de branche nous a permis dans le futur de pallier ce problème.</p> <p>Pour conclure, ce projet m'a beaucoup apporté en matière de connaissances techniques et travail d'équipe à travers un projet enrichissant à faire.</p>
Dorian	<p>Durant ces semaines, de projet. Mes camarades et moi avons réfléchi à savoir comment le réaliser. Notre équipe a préféré se concentrer sur la conception de notre application et méthodes.</p> <p>D'abord pendant ce projet, notre objectif principal était de concevoir un jeu tout d'abord fonctionnel quitte à perdre en confort sur le jeu. De ce fait, c'est pour cela que notre jeu n'est pas d'une grande simplicité d'utilisation.</p> <p>Par la suite, nous pourrions tout d'abord pour améliorer ce projet travailler plus dans une ambiance calme ce qui nous permettra d'être plus souvent concentré. Pour un futur projet, il nous faudra aussi plus se focaliser sur l'aspect esthétique sur l'aspect de notre interface.</p> <p>Pour conclure notre première année et voir si nous sommes capables de fournir un travail conséquent en équipe.</p>

CONCLUSION

Le projet nous a confronté à certaines difficultés avec la synchronisation des fichiers GitHub ce qui nous a obligé à revenir sur une ancienne version. À certains moments, il y avait des nuisances sonores qui empêchaient notre concentration.

Sur le projet, on aurait pu faire une interface graphique, mais le manque de temps à jouer en notre défaveur. L'ihm qui aurait pu être plus clair.

Ce projet nous a permis à tous d'améliorer notre travail en équipe et des connaissances en Java.