|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| N° | Méthode à tester | Valeur entrée | Résultat attendu | Résultat obtenu |
| T1 | verificationCartesRoute() C\_Joueur | Route=5  cWagon=3 | False | False |
| T2 | verificationCartesRoute() C\_Joueur | Route=5  cWagon=5 | True | True |
| T3 | verificationCartesRoute() C\_Joueur | Route=5  cWagon=6 | True | True |
| T4 | verificationWagonsRoute()  C\_Joueur | Route = 4  nbWagon = 2 | False | False |
| T5 | verificationWagonsRoute()  C\_Joueur | Route = 4  nbWagon = 4 | True | True |
| T6 | verificationWagonsRoute()  C\_Joueur | Route = 4  nbWagon = 6 | True | True |
| T7 | determinerVainqueur()  C\_Partie | Joueur 1 = 20  Joueur2 = 40 | Joueur 2 | Joueur 2 |
| T8 | determinerVainqueur()  C\_Partie | Joueur 1 =40  Joueur2 = 40 | Joueur 1 | Joueur 1 |
| T9 | determinerVainqueur()  C\_Partie | Joueur 1 = 60  Joueur 2 = 40 | Joueur 1 | Joueur 1 |
| T10 | initialiserWagons()  C\_Partie | Couleur= « rouge » | lesWagons[n]= « rouge » | 45/45 wagons rouge |
| T11 | initialiserWagons()  C\_Partie | Couleur= « bleu » | lesWagons[n]= « bleu » | 45/45 wagons bleu |
| T12 | piocher()  C\_PiocheDestination | - | OutOfCardsException() | OutOfCardsException() |
| T13 | piocher()  C\_PiocheDestination | - | Piochees[0] != null | Exception |
| T14 | conserverCarte()  C\_PiocheDestination | O | true | false |
| T15 | conserverCarte()  C\_PiocheDestination | N | false | false |
| T16 | distribuer()  C\_PiocheDestination | //1 carte choisie | "Vous devez conserver au moins 2 cartes !" | Exception |
| T17 | distribuer()  C\_PiocheDestination | //2 cartes choisies | Piochees contient 2 cartes choisies | Exception |
| T18 | distribuer()  C\_PiocheDestination | // 3 cartes choisies | Piochees contient 3 cartes choisies | Exception |
| T19 | piocher1Carte()  C\_PiocheWagon | « premiere carte » = WagonBleu | carteWagon.pollFirst() = bleu | bleu |
| T20 | piocher1Carte()  C\_PiocheWagon | « premiere carte » = WagonNoir | carteWagon.pollFirst() = noir | noir |
| T21 | piocher1Carte()  C\_PiocheWagon | « premiere carte » = Locomotive | Piochees = Locomotive | locomotive |
| T22 | Piocher()  C\_PiocheWagon | « premiere carte » = Locomotive | Piochees = Locomotive | Locomotive |
| T23 | Piocher()  C\_PiocheWagon | « premiere carte » = WagonBleu  S == O | Piochees.size() == 2 | Piochees.size() == 1  Carte = bleu |
| T24 | Piocher()  C\_PiocheWagon | « premiere carte » = rouge  S == N | Piochees.size() == 1 | Piochees.size() == 1  Carte = rouge |
| T25 | Distribuer()  C\_PiocheWagon | - | Cartes.length == 5 |  |

# Rapport de Test

## Fonctions à retravailler :

* verificationCartesRoute() Classe Joueur
* verificationWagonsRoute() Classe Joueur
* piocher() Classe PiocheDestination
* distribuer() Classe PiocheDestination

# Débogage

* verificationCartesRoute() **-> fonctionnelle**
* verificationWagonsRoute() **-> fonctionnelle**
* piocher() (classe PiocheDestination) **-> bug à cause de la méthode conserverCarte()**
* conserverCarte() **-> bug à cause du scanner**
* distribuer() **-> bug à cause de la méthode conserverCarte()**
* piocher **-> bug à cause du scanner de la méthode conserverCarte()**