**Initialisation**

= 1ère connexion d’un joueur

1) Obtention de l’ID du joueur :

Données en entrée :

* <aucune>

Données en retour :

* ID joueur
* Itinéraire : étapes (départ, étapes intermédiaires, arrivée) avec coordonnées

2) Récupération des données

Données en entrée :

* ID joueur

Données en retour :

* Main du joueur
* Cartes exposées / joueur
* Etape courante / joueur
* Position courante / joueur
* Distante parcourue / joueur
* Position de centrage = position de départ
* Inactif

3) Mise en attente d’événement du serveur

Données en entrée :

* ID joueur

Données en retour :

* True / False

**Nouveau tour du joueur actif**

1) Récupération des données

Données en entrée :

* ID joueur

Données en retour :

* Main du joueur
* Cartes exposées / joueur
* Etape courante / joueur
* Dernière carte jouée / joueur
* Position courante / joueur
* Distante parcourue / joueur
* Position de centrage = du joueur venant de jouer
* Commerces activables (nom, position)
* Actif

2) Tirage d’une carte

Données en entrée

* ID joueur
* Commerce sélectionné

Données en retour

* Carte tirée

3) Jeu d’une carte

Données en entrée

* ID joueur
* Carte jouée
* Adversaire visé (option)
* Défaussée (option)

Données en retour

* Opération interdite

ou

* Main du joueur
* Cartes exposées / joueur
* Etape courante / joueur
* Dernière carte jouée / joueur
* Position courante / joueur
* Distante parcourue / joueur
* Position de centrage = position du joueur venant de jouer
* Inactif

4) Mise en attente d’événement du serveur

Données en entrée :

* ID joueur

Données en retour :

* True / False

**Nouveau tour du joueur inactif**

1) Récupération des données

Données en entrée :

* ID joueur

Données en retour :

* Main du joueur
* Cartes exposées / joueur
* Etape courante / joueur
* Dernière carte jouée / joueur
* Position courante / joueur
* Distante parcourue / joueur
* Position de centrage = position du joueur venant de jouer
* Inactif

2) Mise en attente d’événement du serveur

Données en entrée :

* ID joueur

Données en retour :

* True / False