I/ Structure du json de transmission (dans les deux sens)

* event\_type : Type de l’event (string)
* Id unique de l’expéditeur (transmis lors de la connexion au serveur, la seule exception étant le premier message de connexion si le client n’a jamais reçu d’id avant), ou -1 pour le serveur
* Id de l’évènement (à renvoyer absolument si il s’agit d’une réponse)
* Contenu (content)

II/ Processus de connexion et d’envoi de packets :

1. Lorsqu’un client souhaite se connecter, il se connecte au serveur, et doit immédiatement lui envoyer un message « header » présenté plus loin qui permet de signaler si il s’est déjà connecté avant (et donc s’il souhaite profiter d’un espace client déjà enregistré) ou pas.
2. Le serveur analyse le header et regarde s’il dispose en mémoire du client mentionné. Si oui, envoie un packet UserRecovered

III/ Evènements possibles :

* GetAiguillages
* GetAiguillage :
  + Int id
  + Int aiguillage\_handler\_id
* GotAiguillage :
  + Int aiguillageHandlerId
  + Int
* AddAiguillage :
  + String Tag
  + String type
  + Json params
  + Int idAiguillageHandler
  + JSON comp
* Connect :
  + String key
* Logged :
  + String client\_id
  + Bool success
* SendHeadMessage :
  + Bool wish\_recover
  + String previous\_key (si nécessaire
* SwitchAiguillage :
  + aiguillage\_id
  + aiguillage\_handler\_id