**Websockets**

**Installation**

Requis : *npm* (node packet manager) & *node.js*

*sudo apt-get update*

*sudo apt-get install npm* (installe aussi node)

*sudo npm install websocket*

**Utilisation de *node.js***

Créer un fichier pour le serveur, par exemple server.js

Pour l’exécuter : *node server.js*

Exemple de serveur : 

NB : Le serveur websocket a besoin d’un serveur http pour fonctionner. Ici le serveur va tourner sur le port 18000, url *ws://IP\_MACHINE :18000*.

Le serveur peut envoyer des messages grâce à l’objet *connection* : *connection.sendUTF(« Coucou »)* envoie coucou au client de la connexion.

Toute la gestion se fait dans l’événement *message,* c’est ici qu’il faut implémenter le protocole et la logique. Le serveur sert essentiellement d’intermédiaire entre les clients. Un client fait une mise à jour, il l’envoie au serveur, le serveur la fait suivre aux autres clients.

Exemple de client : 

**Logique générale de l’application**

*SERVEUR*

Evénement : Un client se connecte

* Le serveur reçoit la connexion, il la conserve dans sa liste de connexions actives
* Il informe tous les clients déjà connectés de la présence d’un nouveau client
* Il informe le nouveau client de la présence de tous les clients connectés

Evénement : Un client se déconnecte

* Le serveur enlève le client de sa liste de connexions actives
* Il informe tous les clients connectés de la déconnexion d’un client donné

Evénement : Un client envoie un message ou une position

* Le serveur informe tous les clients du nouveau message ou de la modification de la position

*CLIENT*

Evénement : Le serveur confirme la connexion au client

* Le client créé un nouveau joueur

Evénement : Le serveur informe le client de la présence d’un nouveau joueur

* Le client enregistre le joueur dans sa liste des joueurs actifs et l’inclut dans la logique d’affichage et de mise à jour

Evénement : Le serveur informe le client d’un nouveau message ou d’un changement de position d’un joueur

* Le client déplace le joueur concerné au bon endroit ou affiche le message

Evénement : Le serveur informe le client de la déconnexion d’un joueur

* Le client enlève le joueur de la liste des joueurs actifs et l’exclut de la logique d’affichage et de mise à jour.