Dans le cadre du développement du projet de bataille de pixels, nous devions développer les architectures du serveur et du client.

La communication se faisait via le protocole TCP.

Par défaut, le port du serveur devait être le 5000, la matrice de pixel avait pour dimension 80 par 60 et 10 pixels par minute pour chaque client. Le serveur devait pouvoir utiliser des valeurs référencées par l’utilisateur à la place des valeurs par défaut lors du lancement (port, dimensions et pixels par minute).

Le serveur devait se charger de l’interaction avec les utilisateurs de façon individuel et identifier les commandes qu’ils lui envoyaient, pour cela il possédait une liste chaînée référençant tous les clients connectés. Il devait aussi initialiser la matrice de pixels, la modifier à la demande de clients, si ces derniers n’avaient pas atteint leur quota de pixels par minutes, suite à la modification (ou non) de la matrice, un message récapitulant l’action devait être retournée au client. Chaque minute, le quota de pixels par minute devait être remit à 0 pour tous les clients.

Du côté du client, ce dernier pouvait demander les dimensions de la matrice, à actualiser la matrice, changer un pixel (dans le cas où il aurait dépassé son quota de pixels par minute, le serveur devait renvoyer un message pour en informer l’utilisateur), demander le temps restant avant de pouvoir placer à nouveau des pixels, demander la version du protocole et se déconnecter.

La communication des valeurs RGB de pixels était envoyés en base 64. Le reste des communications étant des chaînes de caractères pouvant être lu et compris par l’Homme.

Pour son fonctionnement, le serveur avait un programme lui permettant d’identifier et implémenter des valeurs par des option au démarrage. D’autres fonctions permettent de créer, gérer et utiliser une liste chaînée de clients. Une fonction permettant d’identifier la commande reçu d’un utilisateur et agir en fonction.

Des fonctions pour la gestion de ma matrice et des quota de modification des clients.