Rapport du 14/02/2020

Durant la séance j’ai commencé à me renseigner sur comment fonctionne MIT appinventor 2. J’ai regardé un tuto pour apprendre les bases mais je me suis rendu compte au fur et à mesure que j’avançais que les fonctionnalités de design étaient très limitées et surtout pas optimisées (impossible de placer un bouton au centre de l’écran à l’aide d’un réglage, il faut mettre du texte sur la gauche du bouton pour le décaler sur la droite par exemple). J’ai aussi perdu pas mal de temps à essayer de connecter le téléphone à l’émulateur pour avoir un aperçu direct de ce que cela donnait sur un écran de forme actuelle (plus allongé qu’auparavant et donc plus allongé que l’aperçu que j’avais de disponible sur le site). Malheureusement peu importe la méthode de connexion utilisée (QR code, ou code de connexion) j’obtenais le même message d’erreur. Je pense donc abandonner cette solution et passer sur android Studio qui utilise le langage de programmation java.

J’ai aussi aidé Arnaud qui rencontrait un problème avec l’intégration de la nouvelle fonction stroboscopique à l’effet lumineux « fade ». En effet dans la version précédente du code les « delay » bloquaient la progression rapide du « for » qui faisait varier les variables de couleurs ce qui nous permettait d’incrémenter les variables de couleurs de 3 à chaque boucle (par exemple « for(v=0 ; v<255 ; v+=3){} » qui fait augmenter la quantité de vert). Ici étant donné que le code ne contient plus de « delay » les boucles for s’enchainent et l’effet fade était devenu beaucoup trop rapide, j’ai donc eu l’idée de diminuer la vitesse d’incrémentation des variables de couleurs lorsque je me suis rendu de cela (par exemple « for(v=0 ; v<255 ; v+=0.0005){} » ).