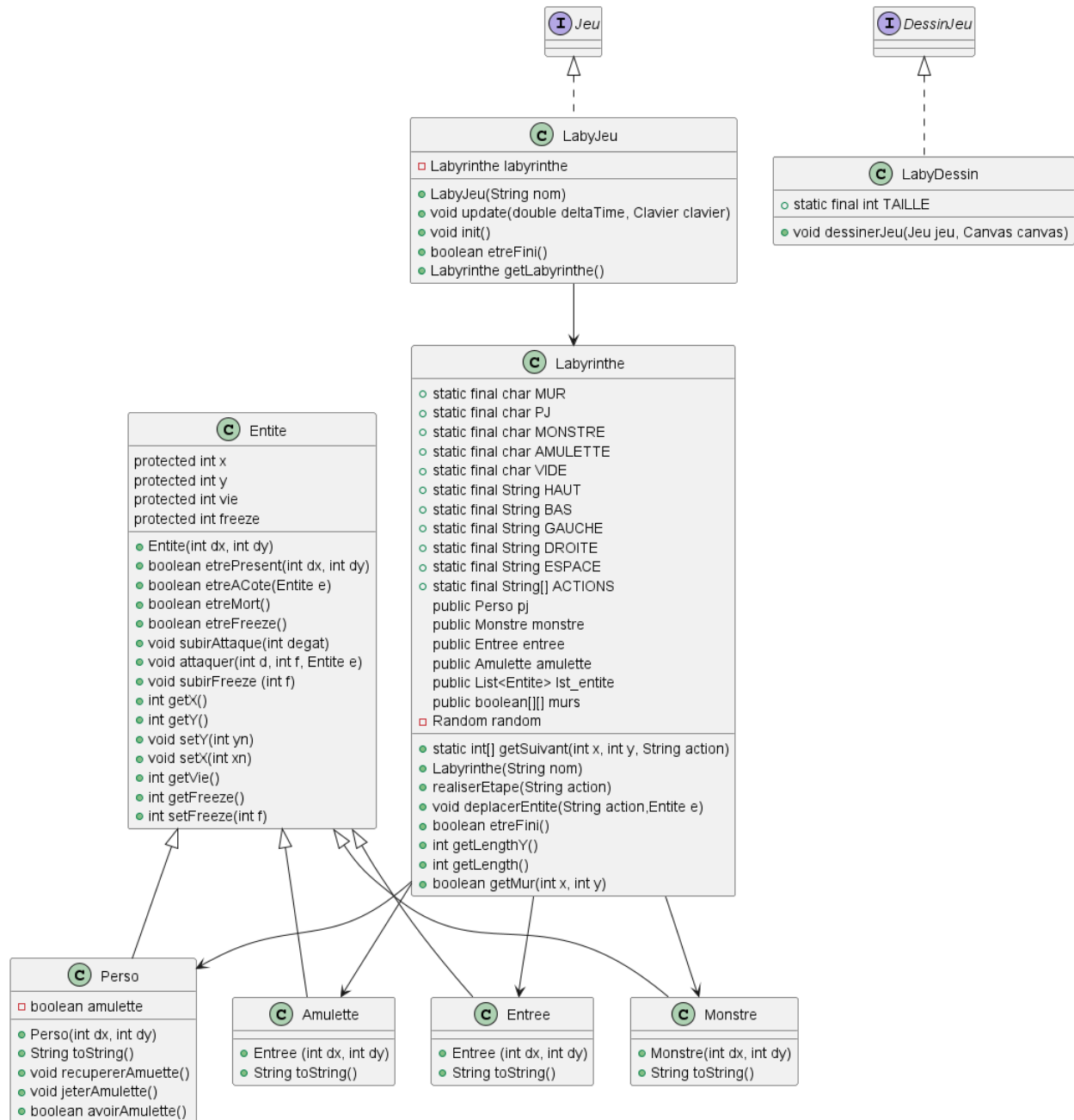
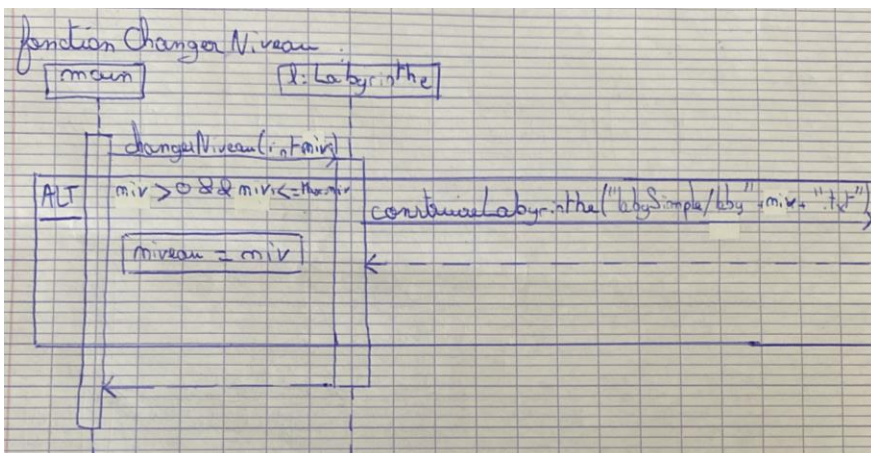
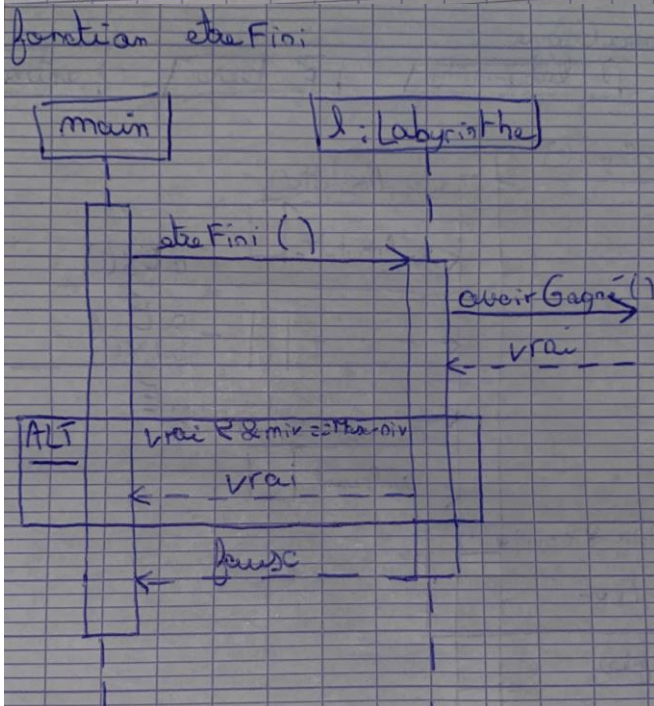
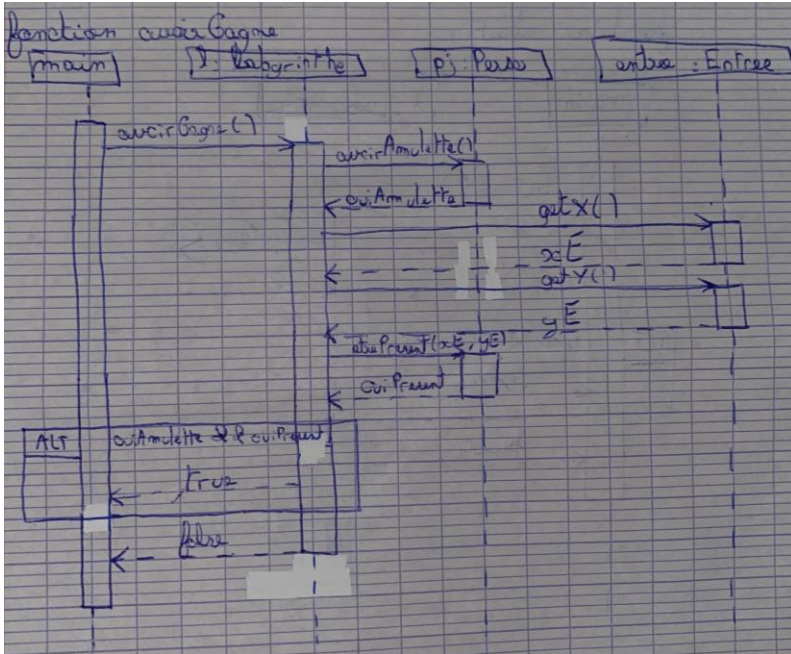


Pour cette itération, nous savions quoi faire dès le moment où nous avons passé la porte. L'idée pour les prochaines heures était d'implémenter la fonctionnalité de pouvoir changer de niveau lorsque celui-ci est terminé. Nous voulions donc encore approfondir notre manière de donner un but au jeu. Le personnage aurait alors la possibilité de Gagner en plus de celle de perdre qu'il avait déjà avant.

Nous avons commencé l'itération en changeant très brièvement notre diagramme de classe initial. En effet, il manquait quelques fonctions comme le `toString` du monstre qui n'apparaissait pas ou encore les flèches qui indiquaient que le Labyrinthe possédait des attributs `Entree` et `Amulette`. Le voici :



Après avoir pris connaissance de notre diagramme de début. Nous avons pu commencer les fonctionnalités à incrémenter. Pour ce faire, nous utilisons la même méthode que nous utilisons depuis le début de la SAE c'est-à-dire, dessiner les diagrammes de séquence et de classe au tableau Velléda. Monsieur Dosh nous a aidé pour cela et nous a donné plus de feutres pour que nous puissions aller plus vite. Nous avons donc réfléchi et fait nos fonctions comme « Avoir gagné » « etreFini » ou encore « changerNiveau ». Ces méthodes ont été dessinées et discutées au tableau pendant une trentaine de minute puis recopiées au propre avant d'être codées et implémentées.



Après s'être mis d'accord avec l'équipe, nous avons codé les fonctionnalités sur notre jeu. Nous avons également ajouté durant cette itération, la possibilité de mettre plusieurs monstres. Notre jeu est donc à un point où le personnage du joueur apparaît, il doit aller chercher une amulette dans un labyrinthe en évitant les monstres, il peut les empêcher de bouger mais ne peut pas les tuer. Il doit donc se libérer de ce labyrinthe pour en sortir et faire cela jusqu'à ce qu'il soit venu à bout de tous les labyrinthes sur son chemin. Il y a également un labyrinthe d'entraînement pour comprendre le but du jeu.

Le diagramme de classes final est donc terminé. Nous avons concentré une grande partie de la SAE à la compréhension de notre code au détriment du code en lui-même. Les fonctionnalités ajoutées sont sûres, testées et compilent correctement. Nous avons créé chacune de nos fonctions sur des bases solides et le travail d'équipe a beaucoup joué car la cohésion d'équipe lors des réunions et des explications au tableau nous ont permis un travail de meilleure qualité. De plus, la manipulation d'un dépôt Git a permis une synchronisation entre tous les membres du groupe. Voici alors notre diagramme final.



La dernière itération a été utilisée en grande partie dans le but de préparer la présentation Finale afin de vendre notre jeu le mieux possible.