# Tâches pour le Gantt :

## 1. Game Design

## 1.1 Niveaux avec énigme de type « code »

## 1.2 Niveaux avec énigme de type « utilisez le(s) composant(s) »

## 2. Elaboration des cours, mémos, lexiques

## 3. Programmation

### 3.1 Programmation Arduino

#### 3.1.1 Programmation des éléments INTPUT

##### 3.1.1.1 Programmation des capteurs

###### 3.1.1.1.1 Programmation du bouton

###### 3.1.1.1.2 ­Programmation du potentiomètre

###### 3.1.1.1.3 Programmation de capteurs tels que luminosité, humidité…

##### 3.1.1.2 Programmation des moteurs

#### 3.1.2 Programmation des éléments OUTPUT

##### 3.1.1.1 Programmation du piézoélectrique

##### 3.1.1.2 Programmation de l’écran LCD

### 3.2 Programmation Java

#### 3.2.1 Menu d’accueil

#### 3.2.2 Menu d’options

#### 3.2.3 Implémentation des cours, mémos et lexiques

#### 3.2.4 Système de chargement des niveaux

#### 3.2.5 Déplacement du personnage (clavier : flèches droite/gauche)

#### 3.2.6 Gestion vie/mana du personnage

#### 3.2.8 Menu de sauvegarde + sauvegarde

### 3.3 Programmation Java/Arduino

#### 3.3.1 Communication Java/Arduino (Windows, Linux, Mac)

#### 3.3.2 Tests des branchements des composants

#### 3.3.3 Interaction jeu/Arduino liées au personnage (via saisie d’instructions au clavier)

#### 3.3.4 Interaction jeu/Arduino liées aux niveaux (via saisie d’instructions au clavier)

## <!> Précisions :

* Les numéros sont seulement là pour s’organiser, des tâches peuvent être effectuées simultanément.
* L’UML pour la classe est à faire avant la programmation de la classe.
* On génèrera la classe à partir de l’UML.
* On créera les tests unitaires avant de programmer les méthodes, bout par bout.
* On fera la javadoc (package, sous-package, constructeur, méthode…) au fur et à mesure de la programmation.
* On reformatera le code régulièrement.
* Une fois tous les tests unitaires effectués, on exécutera JaCoCoverage.
* Lorsque la classe est terminé, on créer le diagramme UML final de cette classe
* On ne commitera pas les fichiers compilé (possibilité de conflit lors de merge).
* On créera régulièrement des branches Git pour la programmation et mergera après avoir effectué les points ci-dessus.