

Mr Barrios – Dirigeant qualité de gestion projet 27 Mai 2019

**PLAN PROJET**

**En Gestion Projet**

Groupe

Yanis Delmas

Yoan Guiraud

Pierre-Marie Combalbert

Maël Le Folgoc Pontis

Table des matières

[I. Introduction 3](#_Toc9787487)

[II. Présentation du projet 3](#_Toc9787488)

[A. Définition générale du projet 3](#_Toc9787489)

[B. Cahier des charges 3](#_Toc9787490)

[a) Description du jeu des grenouilles et des crapauds 3](#_Toc9787491)

[b) Description du casse-tête des grenouilles et des crapauds 4](#_Toc9787492)

[c) Application à développer 5](#_Toc9787493)

[d) Configurations initiales 5](#_Toc9787494)

[e) Remarque 6](#_Toc9787495)

[C. Charte du projet 6](#_Toc9787496)

[f) Objectif du projet 6](#_Toc9787497)

[g) Périmètre du projet 6](#_Toc9787498)

[h) Demandes hors périmètre 6](#_Toc9787499)

[i) Principaux livrable 6](#_Toc9787500)

[j) Acteur du projet : équipe MOE, MOA, contrôle qualité 7](#_Toc9787501)

[k) Condition d’acceptation : exigences, contraintes, qualité, critères et mesures 7](#_Toc9787502)

[l) Risque identifier 8](#_Toc9787503)

[m) Politique de gestion des risques 8](#_Toc9787504)

[III. Organisation du projet 8](#_Toc9787505)

[D. Présentation du cycle de vie et ses conséquences 8](#_Toc9787506)

[E. Identifier les rôles 8](#_Toc9787507)

[F. Plan de communication 9](#_Toc9787508)

[n) Localisation géographique des intervenants 9](#_Toc9787509)

[o) Moyens de communication utilisés 9](#_Toc9787510)

[p) Réunion projet MOE 9](#_Toc9787511)

[q) Comité de pilotage 9](#_Toc9787512)

[G. Assurance qualité 9](#_Toc9787513)

[r) Manuel qualité et démarche qualité 10](#_Toc9787514)

[H. Ressource matérielle et logicielles 10](#_Toc9787515)

[IV. Pilotage du projet 12](#_Toc9787516)

[I. Cycle de vie 12](#_Toc9787517)

[J. Planification prévisionnel initiale 12](#_Toc9787518)

[Iteration 1 : 12](#_Toc9787519)

[s) Ordonnancement des principales itérations/taches 12](#_Toc9787520)

[t) Identification des principaux jalons 12](#_Toc9787521)

[u) Calendrier prévisionnelle 12](#_Toc9787522)

[Iteration 2 : 13](#_Toc9787523)

[v) Ordonnancement des principales itérations/taches 13](#_Toc9787524)

[w) Identification des principaux jalons 13](#_Toc9787525)

[x) Calendrier prévisionnelle 13](#_Toc9787526)

[Iteration 3 : 13](#_Toc9787527)

[y) Ordonnancement des principales itérations/taches 13](#_Toc9787528)

[z) Identification des principaux jalons 13](#_Toc9787529)

[aa) Calendrier prévisionnelle 13](#_Toc9787530)

[K. Organisation des réunions et comités de pilotage 14](#_Toc9787531)

[L. Suivi du projet 14](#_Toc9787532)

[bb) Synthèse par tableau de bord 14](#_Toc9787533)

[cc) Résultat des tests et recette de prototype de la période 15](#_Toc9787534)

[dd) Identification des principaux écarts et problèmes constaté, solutions possibles 15](#_Toc9787535)

[ee) Proposition de modification de la planification prévisionnelle pour tenir compte des corrections à apporter 16](#_Toc9787536)

[ff) Comptes-rendus des réunions projets de la période 17](#_Toc9787537)

[gg) Compte-rendu du comité de pilotage de la période 17](#_Toc9787538)

[V. Bilan projet 17](#_Toc9787539)

[M. Résultats du projet en therme de satisfaction du besoin 17](#_Toc9787540)

[N. Bilan de la gestion du projet 17](#_Toc9787541)

[hh) Bilan des écarts entre la planification prévisionnelle et le réel observé 17](#_Toc9787542)

[ii) Bilan de la gestion de communication 17](#_Toc9787543)

[jj) Bilan de la « qualité du projet » 17](#_Toc9787544)

[O. Bilan personnel du chef de projet 17](#_Toc9787545)

[P. Bilan des ressentit de l’équipe projet 17](#_Toc9787546)

[Q. Conclusion générale sur la gestion du projet 18](#_Toc9787547)

[VI. Glossaire 18](#_Toc9787548)

# Introduction

Nous avons choisi le jeu des grenouilles et crapauds programmé en Java permettant de jouer au jeu et casse-tête des grenouilles et des crapauds. Notre plan projet a pour but d’identifier les éléments importants d’un projet comme la spécification, la communication et les outils exploités au sein du groupe de projet. Ensuite, les parties les plus importantes du projet sont l’organisation du projet et le pilotage du projet qui permettent une bonne construction du projet.

# Présentation du projet

## Définition générale du projet

Le projet « Grenouille et crapaud » est un jeu où deux équipes (une équipe crapaud et l’autre celle des grenouilles) situé des deux extrémités du plateau de jeu vont s’affronter jusqu’à bloquer l’adversaire ou être arrivé à l’autre bout du plateau le premier. Il y pourra aussi jouer dans un casse-tête, le but du casse-tête est que le joueur pourra contrôler les deux équipes, le jeu se terminera quand tous les pions auront traversé le plateau ou quand des pions seront bloqués.

## Cahier des charges

Le but de ce projet est de développer un programme permettant de jouer au jeu et au casse-tête des grenouilles et des crapauds.

### Description du jeu des grenouilles et des crapauds

Le jeu oppose deux joueurs et consiste à déplacer des grenouilles et des crapauds sur un plateau quadrillé de N lignes et M colonnes. En début de partie, les grenouilles sont attribuées à l'un des joueurs, les crapauds à l'autre. Le détenteur des grenouilles commence et ensuite les joueurs jouent à tour de rôle. Le premier qui ne peut plus avancer ses batraciens a perdu.

Les grenouilles avancent horizontalement de la gauche vers la droite, et les crapauds de la droite vers la gauche.

Si la case adjacente dans le sens de déplacement est libre, la grenouille (respectivement le crapaud) déplacée vient occuper cette case. Si, par contre elle est occupée par un crapaud (respectivement une grenouille) et que la case suivante est libre, le batracien vient occuper cette case libre. Dans tous les autres cas, le batracien ne peut être déplacé.

**Exemple avec un plateau de 1 \* 5 :**

1. Plateau initial

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **G** | **G** |  | *C* | *C* |

* 1. 2 3 4 5

1. La grenouille de la case 2 avance en 3

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **G** |  | **G** | *C* | *C* |

* 1. 2 3 4 5

1. Le crapaud de la case 4 avance en 2

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **G** | *C* | **G** |  | *C* |

* 1. 2 3 4 5

1. La grenouille de la case 3 avance en 4

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **G** | *C* |  | **G** | *C* |

* 1. 2 3 4 5

1. Le crapaud de la case 5 avance en 3

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **G** | *C* | *C* | **G** |  |

* 1. 2 3 4 5

1. La grenouille de la case 4 avance en 5

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **G** | *C* | *C* |  | **G** |

* 1. 2 3 4 5

1. Le joueur qui détient les crapauds a perdu puisque ceux-ci ne peuvent plus avancer.

### Description du casse-tête des grenouilles et des crapauds

Les règles de déplacement sont identiques à celles du jeu. Cependant, les déplacements des grenouilles et des crapauds n'alternent pas obligatoirement. Le but de l'unique joueur est d'amener toutes les grenouilles à droite et tous les crapauds à gauche du plateau. Si cette disposition est atteinte, la partie de casse-tête est gagnée.

Exemple de disposition gagnante :

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *C* | *C* |  | **G** | **G** |
|  |  | | |  |
| **G** | *C* | *C* |  | **G** |

Exemple de disposition perdante :

### Application à développer

A son lancement l’application proposera à l’utilisateur d’afficher une aide pour son utilisation.

L’application devra permettre :

* À deux joueurs de s’affronter
* À un unique joueur de jouer contre l’ordinateur
* À un unique joueur d’utiliser le casse-tête
* De gérer des configurations initiales

Dans le cas d’une partie entre deux joueurs, ceux-ci devront saisir leur nom ce qui permettra ensuite à l’application de bien spécifier lequel des deux doit jouer. Les grenouilles et les crapauds seront attribués aux deux joueurs de manière aléatoire. En fin de partie, le nom du gagnant sera clairement affiché.

Dans le cas d’une partie contre l’ordinateur, ou bien lors de l’utilisation du casse-tête, l’application proposera à l’unique joueur d’entrer son nom. S’il n’en donne aucun, il sera désigné par un « joueur humain ». En fin de partie, un message indiquera le résultat de celle-ci et spécifiera le nom de l’unique joueur.

Après une partie, l’application affichera à nouveau le menu principal, ce qui permettra à l’unique joueur ou aux deux joueurs de lancer, éventuellement, une nouvelle partie.

### Configurations initiales

Une **configuration** est la description d'un plateau dans un état précis. Elle indique donc la taille du plateau (nombre de lignes et de colonnes) et la disposition des grenouilles et des crapauds présents sur le plateau. La taille du plateau aura une limite supérieure à fixer (liée aux contraintes d’affichage).

Pour spécifier la configuration initiale d’une partie, l'utilisateur choisira l'une des deux alternatives suivantes :

* Entrer au clavier cette configuration.
* Sélectionner une configuration préétablie. Par conséquent, le programme devra permettre de gérer un ensemble de configurations préétablies que l'utilisateur pourra modifier à son gré par insertion ou retrait d'une configuration. En particulier, lorsque l'utilisateur aura saisi une configuration initiale, le programme lui demandera s'il souhaite l'inclure dans l'ensemble des configurations préétablies

Le placement des batraciens sur le plateau est libre. Toutefois, l’application vérifiera qu’il y a autant de batraciens de chaque catégorie. Si ce n’est pas le cas, la configuration sera considérée comme invalide.

Exemple de configuration initiale :

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **G** |  |  | *C* |  | *C* |
|  | *C* | **G** |  | **G** | **G** |  |
| **G** | **G** |  |  | *C* | *C* |  |
|  | **G** | *C* | **G** |  | *C* | ***G*** |
|  | **G** | *C* |  | *C* | *C* |  |

Les configurations préétablies devront être stockées en mémoire permanente afin d’être retrouvées lors du prochain lancement de l’application.

### Remarque

L’application fonctionnera en mode console ou bien sera dotée d’une interface graphique, au choix du groupe réalisant le projet.

## Charte du projet

### Objectif du projet

L’objectif de ce projet sera la réalisation d’un jeu vidéo « Grenouille et crapaud » avec une interface.

### Périmètre du projet

Les périmètres du projet sont :

* Gestion des Pions et des plateaux.
* Sauvegarde des données du jeu dans un plateau.
* Création d’une IA.
* Création d’une interface.

### Demandes hors périmètre

Les demandes hors périmètre sont :

* Création d’un compteur pour le nombre de déplacement dans le casse-tête.
* Création de plusieurs niveaux d’IA.

### Principaux livrable

1er Livrable :

Premier aperçu d’un jeu fonctionnel permettant de jouer contre un joueur, mais aussi en casse-tête.

Premier aperçu de l’interface fonctionnelle, mais qui n’est pas lié avec le code.

2eme Livrable :

Premier aperçu du jeu et de l’interface ensemble.

Possibilité de jouer contre une IA très facile à battre.

Possibilité de sauvegarder une configuration

Apparition d’un nombre de coups dans le casse-tête

3eme Livrable :

Application finale du jeu vidéo

### Acteur du projet : équipe MOE, MOA, contrôle qualité

Les différentes équipes dans le projet :

L’équipe MOE :

* Mael Le Folgoc Pontis
* Pierre-Marie Combalbert (Gestionnaire de configuration)
* Yoan Guiraud (secrétaire)
* Yanis Delmas (Chef de projet)

L’équipe MOA :

* Corinne Servières

Contrôle qualité :

* Mael Le Folgoc Pontis

### Condition d’acceptation : exigences, contraintes, qualité, critères et mesures

Nos exigences sont :

* Une application fonctionnelle.
* Un code bien commenté.
* Des tests bien réalisés.
* Les conventions respectées.

Nos contraintes sont :

* Avoir des commentaires précis.
* Surveillance des conventions sur nos codes.

Les critères sont :

* Respecte des conventions de java (Packages, Classes, Variables, Constants, …)
* Création d’une javadoc.
* Création d’un jeu de test essayant de gérer le plus de problèmes

Les mesures :

* Une vérification des conventions par le Contrôle de qualité avec l’aide d’éclipse et de ces connaissances, mais juste avant il doit y avoir une vérification du code par le programmeur.
* Vérification par le Contrôle de qualité des tests et des javadocs avec le programmeur pour trouver des erreurs ou des améliorations.
* Chaque programmeur doit revérifier si son code est pertinent et le plus optimisé, le contrôle de qualité peut vérifier. Le contrôle de qualité peut alors vérifier un code s’il y a des doutes sur une partie du code.

### Risque identifier

Les risques que nous avons identifiés :

* Des javadocs mal écrites ou oublié d’être remplis.
* Erreurs dans les conventions java.
* Erreur simple dans le code (oublie de vider le tampon, etc.).
* Mauvaise compréhension du code à programmer par les programmeurs.

### Politique de gestion des risques

* Création automatique de « TODO Commentaire n’est pas terminer » lors de la création d’un commentaire par Eclipse.
* Relecture du code par l’équipe du contrôle de qualité et vérification des conventions par Eclipse.
* Vérification par les tests et vérification usuelle par l’équipe du contrôle de qualité (Test durant une partie de jeu pour découvrir des erreurs ou voir le bon fonctionnement du jeu, par exemple).
* Réunion organisée pour éviter de mauvaises compréhensions et réorganiser les idées du groupe.

# Organisation du projet

## Présentation du cycle de vie et ses conséquences

Le cycle itératif permet de diviser le délai du projet en itération. Les itérations du cycle itératif permettent de s’adapter au changement futur, mais aussi au changement causé par les itérations précédentes.

Les conséquences de ce cycle sont que nous devons proposer un prototype pour chaque fin d’itération, ce qui oblige à devoir réaliser des prototypes fonctionnels pour chaque itération.

## Identifier les rôles

Notre projet est composé de 4 membres avec différents rôles :

* Responsable MOA : Corinne Servières
* Responsable gestion projet : Frédéric Barrios
* Chef de projet MOE et MOA : Yanis Delmas
* Secrétaire de projet : Yoan Guiraud
* Gestionnaire de configuration : Pierre-Marie Combalbert

Programmeur : Maël Le-Folgoc-Pontis

## Plan de communication

### Localisation géographique des intervenants

Yoan : Foyer saint pierre Rodez (12)

Pierre-Marie : 26 avenue de Montpelier Rodez (12)

Yanis : Rue de Saunhac Rodez (12)

Maël : Foyer Sainte Thérèse Rodez (12)

### Moyens de communication utilisés

Les moyens de communication utilisées pour les réunions et les échanges de messages sont faits avec Discord, le compte rendu et les autres documents comme le plan projet par exemple sont réalisés avec Word (format .docx) et peuvent être fournis en PDF (format .pdf) et tout le suivi de l’avancement du projet avec les sauvegardes continuent des documents et de l’avancer du projet se fait à l’aide de GitHub.

Nous utilisons une communication centralisée grâce au logiciel de communication Discord, ce qui implique que toutes les parties prenantes soient reliées à une interface centrale qui stocke toutes les communications en un seul endroit, y compris toutes les documentations et informations nécessaires.

### Réunion projet MOE

Pour l’organisation des réunions de groupe (MOE), nous nous sommes fixés à une ou deux réunions par semaines et elles ont pour but d'organiser, de planifier et d’améliorer les différents objectifs prévus pour l’itération en cours. Et pour l’organisation des réunions MOA avec Corinne Servières, nous nous sommes fixés à une réunion toute les deux à une semaine et demie (chaque itération) qui a pour objectif de faire un point sur l’itération en cours et la prochaine itération prévue, avec des améliorations, modifications à apporter sur le projet et avec des points incompris ou confus à éclaircir. Ces types de réunions durent en moyenne 30 minutes et se font principalement entre les inters cours ou en fin de journée.

### Comité de pilotage

//TODO

## Assurance qualité

Nous suivons la convention Java du 12 septembre 1997 pour toute la programmation Java et le formalisme de modélisation est l’UML2 pour toute la partie conception orientée objet. Ensuite, nous utilisons la version 11 de Eclipse et Java Fx pour une corrélation des versions entre l’IDE et la programmation de l’interface graphique. Le cycle itératif nous impose d’avoir un contrôle de suivi toutes les deux semaines avec la MOA et nous permet de livrer des livrables à chaque itération.

### Manuel qualité et démarche qualité

Le suivi et le contrôle de qualité adoptée nous permet d’avoir un suivi continuel et permet d’avoir du recul sur le projet et les éléments manquants pour la globalité du projet.

## Ressource matérielle et logicielles

Logiciels et modules pour la programmation du projet :

|  |  |
| --- | --- |
|  | **Sublime Text 3 :**  Programmation, Mise en page du code… |
|  | **Eclipse :**  Programmation de l’application |
|  | **Version JAVA utilisée :**  JDK 11 |
|  | **Module pour la création de l’interface :**  JAVA FX 11  Efxclipse  Java Scene Builder 2.0 |

Logiciels de partage fichier :

|  |  |
| --- | --- |
|  | **Git hub desktop :**  Mise en commun des fichiers de l’application après chaque modification |
|  | **Google Drive :**  Partage des documents du plan de projet qui sont plus facilement accessible sur les ordinateurs non équipés de git hub desktop |

Logiciels de communication dans le projet :

|  |  |
| --- | --- |
|  | **Discord :**  Logiciel pour communiquer et mettre en commun, organiser les réunions et les tâches à accomplir |
|  | **Modules de Discord :**  Module pour avertir après chaque commit tous les utilisateurs du projet en les notifiant après chaque commit d’un message.  De plus chaque membre du projet possède l’application sur son pc et son téléphone pour être notifié en permanence |

Logiciels pour la rédaction des documents du projet :

|  |  |
| --- | --- |
|  | **Google docs :**  Pour rédiger des documents en direct avec les différents membres pour penser la rédaction des documents ensemble |
|  | W**ord :**  Rédactions des documents ne nécessitants pas d’être rédigé par plusieurs personne ou d’être accessible depuis plusieurs endroit |

Logiciels pour la présentation des documents :

|  |  |
| --- | --- |
|  | **Prezi :**  Afin de créer le diaporama final qui sera présenté le jour de l’oral de projet. |
|  | **Google Slides + Powerpoint :**  Création du diaporama de secours (Version hors ligne)  + Création au même moment des différentes diapositives et du choix des éléments à présenter (Non possible avec Prezi qui ne peut être modifié que par un seul utilisateur) |

Logiciels pour la modélisation :

|  |  |
| --- | --- |
|  | Modelio :  Création diagramme de cas d’utilisation + diagramme de classe |
|  | Win Design :  Création des premiers documents pour les premières échéances |

Logiciel pour la gestion de projet :

|  |  |
| --- | --- |
| C:\Users\yanis\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\7FF50CF6.tmp | **GanttProject :**  Création des diagrammes de Gantt |

Ressource matérielle :

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 1er PC  Ecran :  VG248QE  Processeur :  I7-6700K  Carte graphique :  GTX 970  Carte mère :  ASUS MAXIMUS VIII  Ram :  32GB de RAM | 2ème PC  Ecran :  ILLYAMA  Processeur :  I5  Carte Graphique :  GTX 960  Carte mère :  MSI  Ram :  8GB de RAM | 3ème PC  Ecran :  Processeur :  AMD  Carte Graphique :  Radeon sapphire 7550  Carte mère  ASUS  Ram :  8GB de RAM | 4ème PC  ASUS X302LJ  (PORTABLE) | 5ème PC  PC PORTABLE | 6ème PC  Ordinateur Région |

# Pilotage du projet

## Cycle de vie

Nous avons choisi un modèle de cycle itératif, car ce modèle nous permet de pouvoir de nous familiariser avec ce cycle de vie, mais aussi ce cycle permet de pouvoir ajouter des besoins qui n’auraient pas été demander au départ de l’application.

## Planification prévisionnel initiale

## 

## Itération 1 :

### Ordonnancement des principales itérations/taches

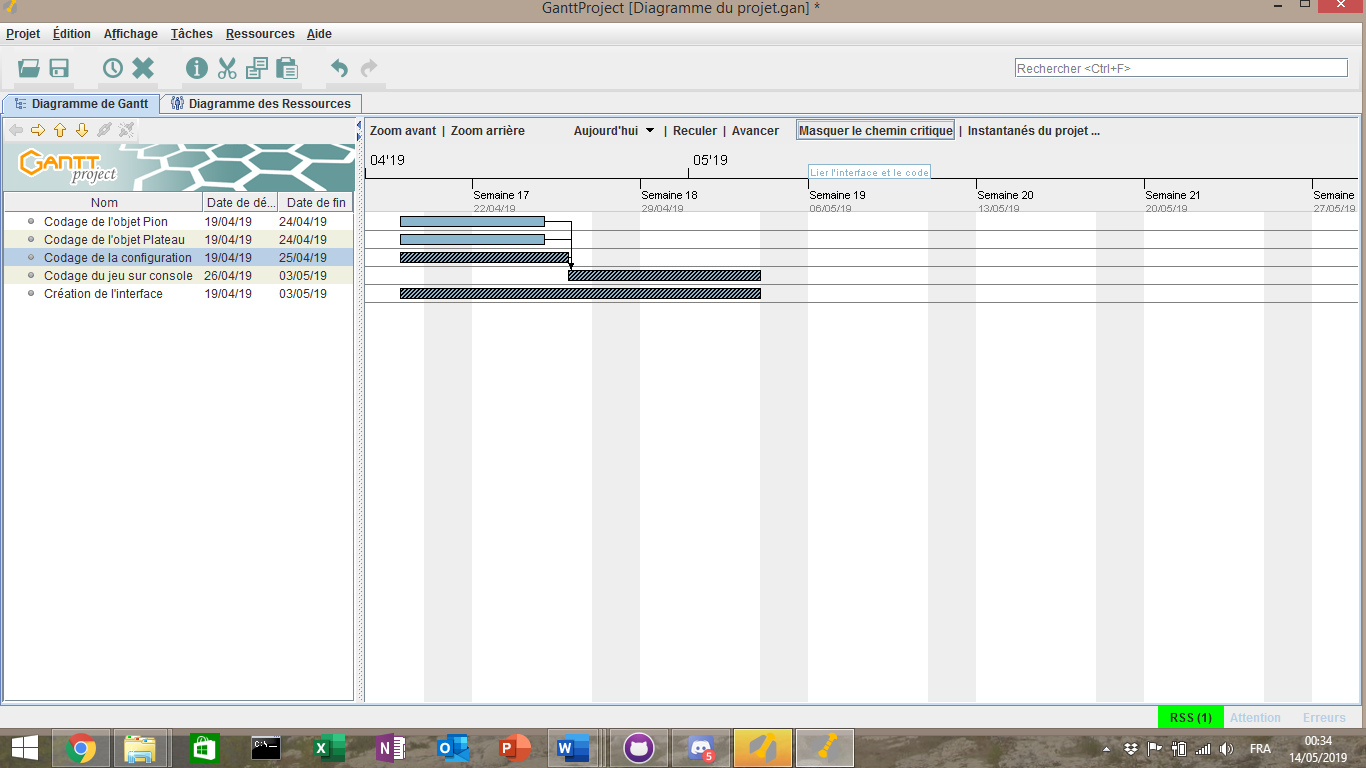
* Création de l’objet Pion et des fonctions permettant de les gérer.
* Création de l’objet Plateau et des fonctions permettant de le gérer.
* Création de la configuration par défaut d’un plateau.
* Création du jeu (joueur contre joueur et casse-tête) sur la console.
* Création d’une interface fonctionnelle

### Identification des principaux jalons

* Le jalon est la fin de l’itération 1, C’est à partir de ce moment que les class plateau et plateau devront être finis et que le code du jeu joueur contre joueur doit être fini.

### Calendrier prévisionnelle

Diagramme de Gantt de l’itération numéro 1 :



## 

## Itération 2 :

### Ordonnancement des principales itérations/taches

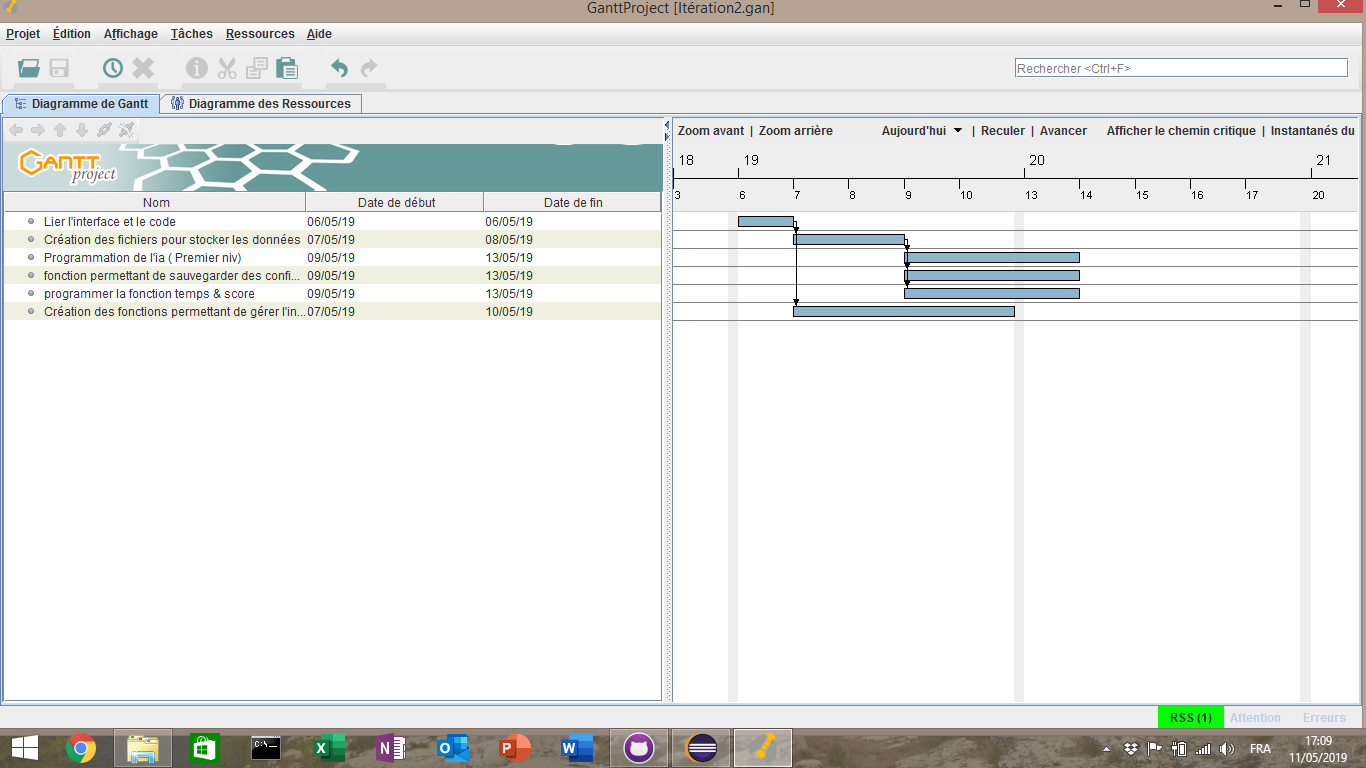
* Création des fichiers permettant la sauvegarde des données sauvegarder sur le jeu
* Création des fonctions permettant de faire les sauvegardes sur le jeu
* Création d’une IA simple pour le premier niveau.
* Lier l’interface et le code.

### Identification des principaux jalons

* Le jalon est la fin de l’itération 2, c’est à ce moment où toutes les configurations devront être finies ainsi que l’interface. On devra avoir un jeu avec une interface fonctionnelle avec des fonctions manquante ainsi qu’une IA simplifiée.

### Calendrier prévisionnelle

Diagramme de Gantt de l’itération numéro 2 :



## Itération 3 :

### Ordonnancement des principales itérations/taches

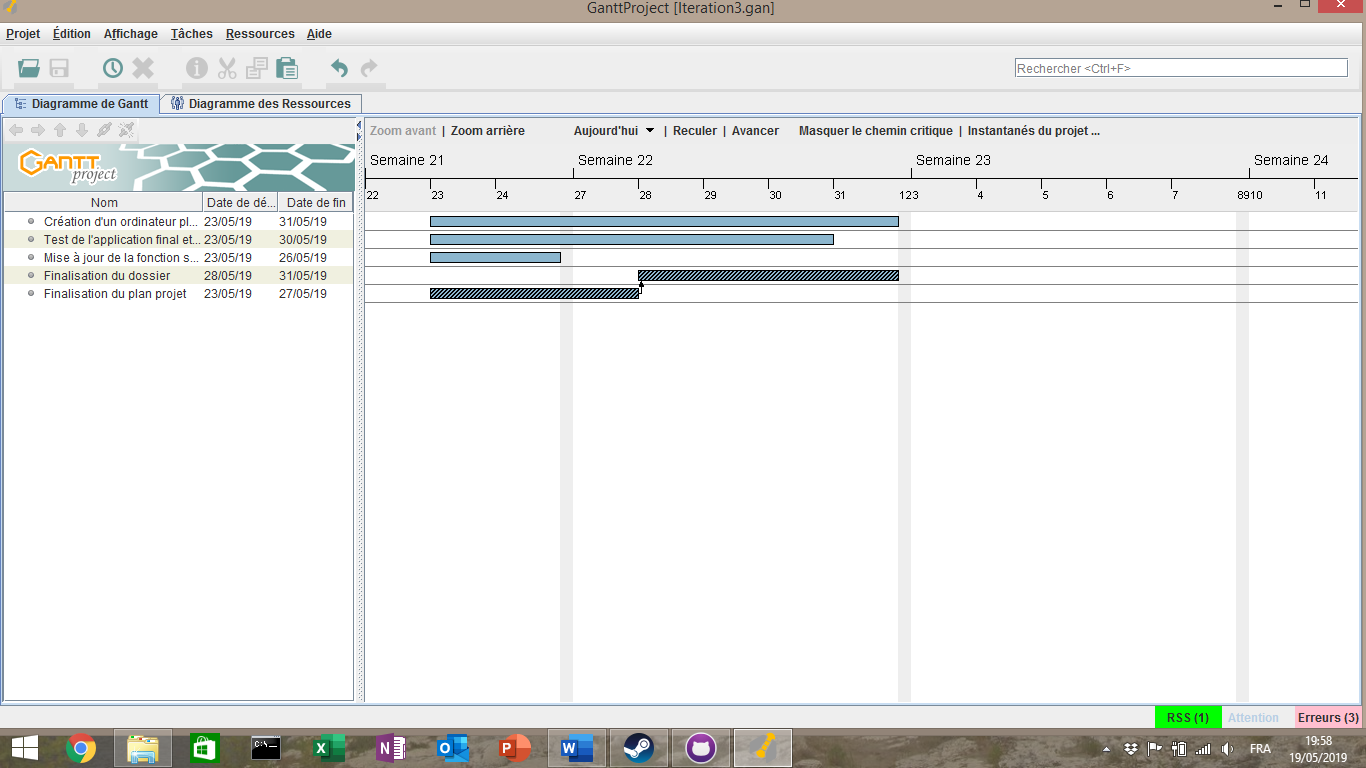
* Création de la fonction chrono
* Création d’un ordinateur plus intelligent.

### Identification des principaux jalons

* Le jalon est la fin du projet (donc fin de l’itération 3), nous devrons donc avoir un jeu avec une interface complet et une IA un peu plus complexe.

### Calendrier prévisionnelle

Diagramme de Gantt de l’itération 3 :



## Organisation des réunions et comités de pilotage

Nous organisons fréquemment de petites réunions entre les membres de l’équipe chaque semaine pour parler de tous les problèmes que chaque membre, on a eu et comment les régler.

Nous organisons aussi une grosse réunion chaque semaine pour s’organiser mieux que lors de nos petites réunions.

## Suivi du projet

### Synthèse par tableau de bord

Tableaux de bord de Maël :

|  |  |  |
| --- | --- | --- |
| Période | Intitulé | Description |
| 22/4 au 29/4 | Programmation de la classe Plateau et Pion | Programmation de la classe plateau et Pion. |
| 30/4 au 05/5 | Lier interface et code | L’objectif dans cette semaine c’est de pouvoir lier le code et l’interface |
| 05/5 au 11/5 | Création des configurations par défauts | L’objectif dans cette semaine, c’est de créer le code permettant de configurer une partie par défaut. |
| 11/5 au 18/5 | Amélioration interface et code | Amélioration du code avec l’interface et nous finissons les configurations. |
| 18/5 au 22/5 | Finir les configurations | Nous devons finir les configurations, car nous avons eu certains bugs |
| 22/5 au 29/05 | Amélioration du code de l’application. | Optimisation et modifications des éléments du code, ainsi que gérer toutes les erreurs vues durant les réunions et pendant l’itération en cours. |

J’ai compté que dans ce projet, j’ai fait 90 heures dessus.

Tableaux de bord de Yanis :

|  |  |  |
| --- | --- | --- |
| Période | Intitulé | Description |
| 22/4 au 29/4 | Programmation de la classe Plateau et Pion | Aide dans la programmation et la recherche de problèmes dans celle-ci |
| 30/4 au 05/5 | Création du premier plan projet | J’ai commencé à créer mes diagrammes de Gantt et le plan projet |
| 05/5 au 11/5 | Continuation du plan projet et aide dans des codes | Je continue mon plan projet, mais j’aide dans certains endroits du code et lors de la vérification. |
| 11/5 au 18/5 | Programmation de l’IA | Création d’une IA très simple |
| 18/5 au 22/5 | Améliorer le Plan projet et aide en programmation | Amélioration du plan projet et aide dans certains codes. |
| 22/5 au 26/5 | Finir et vérifier le plan projet et aider si besoins dans le code | Je vais devoir terminer le plan projet, mais aussi vérifier s’il ne manque rien. Je pourrais aussi aider dans la programmation. |
| 26/5 au 29/5 | Aide dans la partie programmation. | Programmation et tests de différentes méthodes, ainsi qu’optimisation du code. |

J’ai compté que dans ce projet, j’ai fait 64 heures dessus.

Tableaux de bord de Pierre Marie :

|  |  |  |
| --- | --- | --- |
| Période | Intitulé | Description |
| 22/4 au 29/4 | Création de l’interface | Programmation et création de l’interface |
| 30/4 au 05/5 | Lier interface et code | Lier le code et l’interface pour pouvoir jouer sur cette interface |
| 05/5 au 11/5 | Régler des problèmes avec l’interface et le code | Quelques problèmes ont été alors trouvé donc j’ai dû les régler. |
| 11/5 au 18/5 | Gestion des configuration | Gestion pour les configurations avec l’interface |
| 18/5 au 22/5 | Finir les configurations | Certains bugs, on put être trouvé donc nous avons dû les corrigés. |
| 22/5 au 29/5 |  |  |

Tableaux de bord de Yoan Guiraud :

|  |  |  |
| --- | --- | --- |
| Période | Intitulé | Description |
| 12/4 au 17/4 | Avancement du rôle de secrétaire et avancement des spécifications détaillées du projet | Rédaction du premier compte rendu et réalisation du diagramme et de la description des cas d’utilisation, ainsi que le diagramme de classe. |
| 22/4 au 29/4 | Continuation du rôle de secrétaire et avancement des spécifications détaillées du projet | Continuation des comptes-rendus et modification de la description des cas d'utilisation ainsi qu’aide à la liaison entre interfaces graphiques et code. |
| 30/4 au 05/5 | Continuation du rôle de secrétaire, finition des spécifications détaillées du projet et programmation | Continuation des comptes-rendus et finition des différents documents pour la prochaine itération, ainsi que programmation sur les configurations. |
| 06/5 au 14/5 | Continuation du rôle de secrétaire et gestion projet | Continuation, présentation et impression des comptes-rendus et des documents nécessaires pour la réunion de gestion projet. |
| 14/5 au 17/5 | Réalisation avec le chef de projet du plan projet | Réalisation en détail du plan projet. |
| 17/5 au 26/5 |  |  |
| 26/5 au 29/5 |  |  |

### Résultat des tests et recette de prototype de la période

Itération 1 :

Tous les tests ont été bien réalisés mais suite à des modifications dans des codes certaines fonctions test ont dû être amélioré, car il y a eu de nouvelles erreurs à devoir être géré par notre application.

Quant au prototype, nous avons eu quelque petit bug mineur qui ont été vite réglé.

Itération 2 :

Certaines difficultés pour les tests avec l’interface, mais ce fut réglé rapidement.

Ayant eu certains problèmes nous avons dû reporter la démonstration du prototype d’une semaines pour pouvoir proposer un prototype.

### Identification des principaux écarts et problèmes constaté, solutions possibles

Problèmes dans l’itération 1 :

* Problèmes dans les classes Pion et plateau qui ont pris plus de temps que prévue. Les solutions possibles sont de rajouter du temps donc devoir raccourcir le temps de programmations sur le code du jeu ou/et enlever des ressources dans l’itération pour rattraper le retard perdu.

Problèmes dans l’itération 2 :

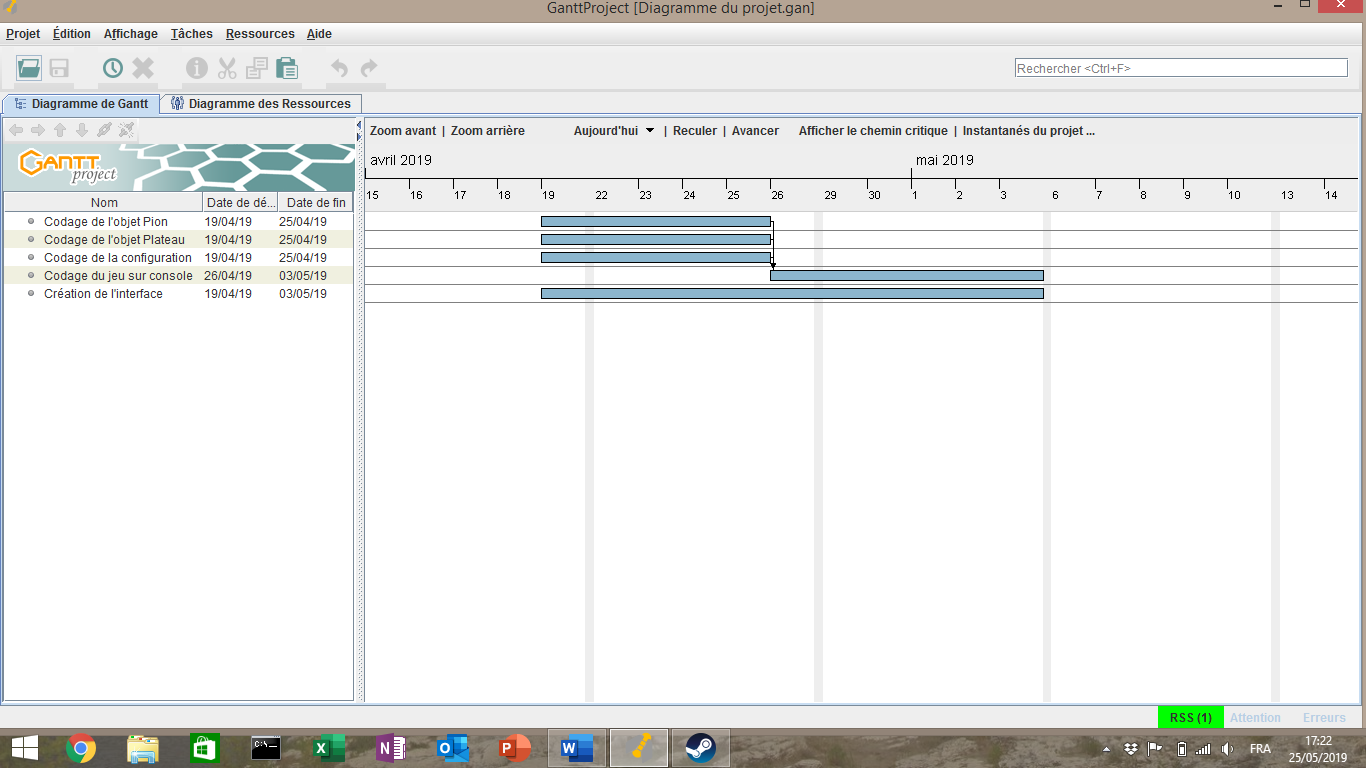
* Un problème entre l’interface et le code. Le code et l’interface n’étant pas programmer par la même personne, il y a eu des opinions différences dans la façon de programmer des fonctions.

La solution possible était de faire encore plus communiquer les programmeurs entre eux pour éviter qui est de tel problème pour les futures itérations.

### Proposition de modification de la planification prévisionnelle pour tenir compte des corrections à apporter

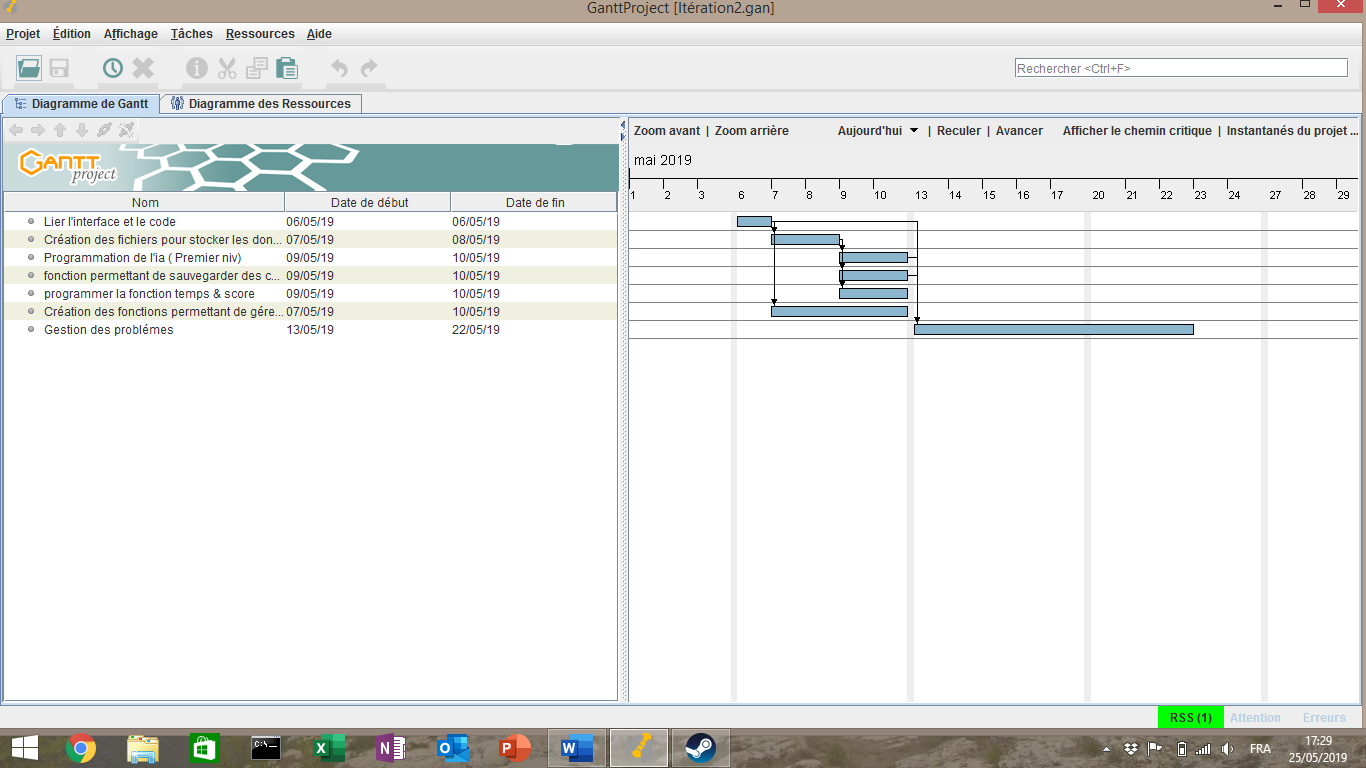
Itération 1 :

Nous avons eu un problème avec les objets plateau et pion donc nous avons dû ajouter 1j pour finir.



Itération 2 :

Nous avons eu un problème entre le code et l’interface donc nous avons dû rajouter 1 semaine en plus pour rendre un prototype fonctionnel.



### Comptes-rendus des réunions projets de la période

**Compte rendu de la réunion du 11/04/2019 :**

**Compte rendu de la réunion du 17/04/2019 :**

**Compte rendu de la réunion du 06/05/2019 :**

**Compte rendu de la réunion du /05/2019 :**

# Bilan projet

## Résultats du projet en therme de satisfaction du besoin

Nous avons réussi à avoir un résultat satisfaisant les besoins du cahier des charges.

Nous avons pu aussi faire une interface, ce qui n’était pas demandé dans le cahier des charges au départ, on a pu aussi ajouter certaines fonctionnalités à l’application.

## Bilan de la gestion du projet

### Bilan des écarts entre la planification prévisionnelle et le réel observé

On a eu quelque écart qui ont décalé nos planifications, mais dans l’ensemble nous avons réussie à rattraper ces erreurs. Nous avons du cependant allonger des tâches dans le temps, ce qui a causé certains écarts avec nos prévisions.

### Bilan de la gestion de communication

On a eu quelque problème de communication dans le projet, ce qui a amené à quelque problème dans la suite, mais nous avons quand même réussie à s’organiser pour que tout le monde puisse avoir les informations. Nous avons organisé des réunions pour que l’équipe communique entre elle.

### Bilan de la « qualité du projet »

Nous sommes plutôt contents de la qualité du projet que nous avons réussie à faire même s’il est probable qu’il y a des bugs.

## Bilan personnel du chef de projet

Mon bilan en tant que chef de projet est mitigé, car d’un côté nous avons réussie à créer une application « Grenouille et crapaud » fonctionnelle et une bonne entente tout le long du projet avec tous les membres de l’équipe, mais nous avons eu certaines difficultés dans certaines itérations qui nous fessaient prendre du retard. Souvent ces erreurs étaient du par un manque de communication sur certains points. Je retiens pour les prochains projets que si je suis chef de projet, il faudrait que je sois plus strict et que j’arrive à mieux organiser l’équipe pour éviter ces pertes inutiles par manque de communications.

## Bilan des ressentit de l’équipe projet

Maël :

Je pense que ce projet c'est bien passé dans son ensemble. Chaque membre a fait ce qu'il avait à faire et le faisait bien. Je n'ai pas reproduit les erreurs que j'ai pu commettre au premier projet tutoré et à l'application bus. Je pense cependant que nous avons sous-estimé à certains moment la quantité de travail occasionnant des retards et je pense également que c'était une erreur d’avoir créé le code et l'interface indépendamment.

Yoan :

Pierre-Marie :

## Conclusion générale sur la gestion du projet

Un projet qui même avec certains points négatifs, c’est quand même bien déroulé dans l’ensemble. Nous sommes ravis de ce projet, car il nous à permis de rencontrer certaines erreurs de devoir les gérer.

# Glossaire

**Comité de pilotage** : un groupe de dirigeants chargé de veiller au bon fonctionnement d'un  au sein d'une [entreprise](https://fr.wikipedia.org/wiki/Entreprise).

**Jalon** : un jalon, dans le cadre de la gestion de projet, est la fin d'une étape, la fin d'un travail.

**MOA** : la maîtrise d'ouvrage, aussi dénommée maître d'ouvrage est la personne pour qui est réalisé le projet. Elle est l'entité porteuse d'un besoin, définissant l'objectif d'un projet, son calendrier et le budget consacré à ce projet.

**Modules** : module d'extension ajoutée à une application pour étendre ses fonctions.

**MOE** : la maîtrise d'œuvre ou maître d'œuvre est la personne [physique](https://fr.wikipedia.org/wiki/Personne_physique) ou [morale](https://fr.wikipedia.org/wiki/Personne_morale) choisie par le [maître d'ouvrage](https://fr.wikipedia.org/wiki/Ma%C3%AEtre_d%27ouvrage) pour la conduite opérationnelle des [travaux](https://fr.wikipedia.org/wiki/Chantier_(b%C3%A2timent)) en matière de [coûts](https://fr.wikipedia.org/wiki/Co%C3%BBt), de [délais](https://fr.wikipedia.org/wiki/D%C3%A9lai) et de choix [techniques](https://fr.wikipedia.org/wiki/Technique), le tous conformément à un contrat et un [cahier des charges](https://fr.wikipedia.org/wiki/Cahier_des_charges). Un maître d'œuvre ne peut pas effectuer de travaux.

**Prototype** : exemplaire d'un modèle (de mécanisme, de véhicule) construit avant la fabrication en série

**Itérations** : [action](https://www.linternaute.fr/dictionnaire/fr/definition/action/)  répété [plusieurs](https://www.linternaute.fr/dictionnaire/fr/definition/plusieurs/) [fois](https://www.linternaute.fr/dictionnaire/fr/definition/fois/).