**Construction du Livrable 1.**

Précisions du guide de déroulement :

***3.1 Livrable 1 (15 pts)*** *Le Livrable 1 est un pré-rapport. Il doit au moins contenir :  
— une analyse des besoins (cahier des charges),  
— une spécification fonctionnelle générale (description des fonctionnalités  
de l’application),  
— un regroupement modulaire (par "affinité") des fonctionnalités,  
— une description du flux des données entre les modules.  
Autrement dit, le Livrable 1 doit consigner les résultats obtenus par le parcours des premières étapes du "Cycle en V" du développement d’une application, jusqu’à celle intitulée "Conception préliminaire" incluse. Le contenu de ce document doit évoluer de manière à refléter l’état d’avancement du projet et devenir, à la fin du temps imparti à la réalisation du projet, le rapport final. Pré-rapport et rapport final ont, pour cette raison, la même  
forme et sont désignés "rapport" dans la suite de ce guide. Le rapport doit être consultable, à tout moment du déroulement du projet, par l’ensemble des participants au projet. La rédaction du rapport doit commencer dès l’issue de la réunion de lancement du projet et être partagée grâce à un outil collaboratif : git.  
Ci-après sont énoncés quelques conseils relatifs à la forme et au contenu de ce rapport.  
— Le rapport peut être édité, à l’aide d’un traitement de textes choisi par les membres du groupe-projet.  
— Le rapport doit être soumis au format pdf (Portable Document Format).  
— Le nom du fichier doit être explicite (par exemple : Rapport.pdf).  
— La page de garde doit indiquer le titre du projet, le nom des étudiants et de l’encadrant, la version courante et sa date.  
— Les pages doivent être numérotées.  
— Le rapport doit contenir une table des matières avec les numéros de pages.  
— Des annexes peuvent si nécessaire être utilisées (par exemple, un lexique  
pour les définitions des abréviations éventuelles et/ou des termes techniques  
relatifs au sujet traité).  
— Les références bibliographiques des articles ou livres éventuellement  
consultés doivent être citées.  
— Il faut éviter le style de rédaction « sténographique »avec des abréviations  
(sauf si elles sont définies dans un lexique).  
— Il faut éviter les fautes d’orthographe.  
— Les phrases doivent être syntaxiquement et sémantiquement correctes.*

A déposer sur moodle pour lundi prochain 12/02. Mais il faudra peut-être aussi l’envoyer à l’encadrant ou le déposer sur un support git partagé plus tard.

Il faut aussi utiliser les Slides du cours "Génie Logiciel et Gestion de Projets Informatiques".

Reprendre le teasing du sujet comme base :

*Le sujet que nous avons choisi est un jeu mobile* ***Android****, où s'opposent* ***deux joueurs en ligne****. L'image Paint ci-jointe résume l'esprit du jeu.*

*Le joueur défie un ami avec son identifiant ou rejoint un adversaire en ligne au hasard. Ils s'affrontent ensuite dans une* ***série de mini-jeux rapides*** *à deux (ou plus) parties gagnantes. C'est en quelque sorte un Duel Quiz avec des mini-jeux à la place des questions.*

*Ces mini-jeux auront des graphismes et un gameplay simples et efficaces. Parmi nos idées de base, du plus au moins simple à mettre en oeuvre :*

*- Presser un bouton lorsqu'un compteur atteint zéro.*

*- Un morpion.*

*- Une course d'escargots où il faut balayer l'écran pour faire avancer son escargot de quelques mm.*

*- Un hockey.*

*- Attraper un maximum de fruits qui tombent des cieux avec un personnage en bas de l'écran.*

*- D'autres mini-jeux plus complexes si le temps et les moyens sont avec nous : une course 2D, un combat de plateformes, un mini-golf en une manche...*

***Les moyens utilisés*** *: le langage Java, le logiciel Android Studio, peut-être la bibliothèque java.sockets pour les échanges entre les deux joueurs.*

*Nous pensons d'abord programmer un mode* ***Training****, où l'adversaire sera l'ordinateur.*

*Cela nous permettrait d'attaquer les difficultés de la communication en ligne à deux joueurs dans une deuxième étape, avec le mode* ***2 joueurs*** *en temps réel.*