

# Protocole & Informations - Projet Speaping1 2016-17

*lundi 14 novembre 2016*

## Le cadre...

Le projet est à réaliser en groupe de quatre personnes (et seulement quatre). Sa durée est d'environ cinq mois (de janvier à mai). Un cahier des charges présentera la nature du projet, les différentes parties qui le composent et les délais de réalisation (planning).

## Le sujet...

Le projet que vous allez devoir réaliser, est un logiciel dans lequel l'algorithmique a une part très importante. Le sujet est libre, et doit être validé par Christophe Boullay. A titre d'indication, les sujets faisant appel à du traitement du signal, de l'imagerie ou du calcul scientifique pour ne citer que ceux-là ont de fortes chances d'être acceptés. En revanche, il est peu probable que le développement d'une base de données soit retenu, à moins que l'on ne parle du moteur de la base.

## Les restrictions...

**Ce projet sera développé sous Linux et sera codé en C** et devra fonctionner sur les machines de l'école (sous Linux). C'est d'ailleurs sur celles-ci qu'auront lieu les soutenances (y compris la dernière).

## le protocole...

Vous devez choisir un groupe de quatre personnes, donner un nom à ce groupe, choisir un sujet de projet et remettre le nom du groupe, le nom des membres, le login de chaque membre, le nom du projet ainsi que le nom de celui que vous aurez désigné comme chef de projet (cf. date limite de choix de groupe). Un chef de projet est nécessaire pour servir, entre autres, d'interlocuteur aux jurys des soutenances.

Accompagnant tout cela, vous devez en premier lieu fournir sur deux pages recto/verso un bref aperçu de ce que vous comptez réaliser, en décrivant l'intérêt algorithmique de ce sujet.

Si celui-ci est accepté, vous êtes tenus de réaliser à ce moment le cahier des charges. *Celui-ci doit commencer par une introduction rapide suivi par l'état de l'art<sup>1</sup> condensé correspondant aux technologies actuellement connues permettant de résoudre le(s) problème(s) posé(s) par le sujet choisi.* Chacune de ces technologies doit être expliquée, vous devez montrer ses capacités ainsi que ses limites. Ensuite vous devez préciser la technologie que vous avez retenue et pourquoi.

**Vous devez vérifier que les technologies que vous choisissez sont disponibles sur les machines de l'école (sous Linux). En cas de besoin particulier (bibliothèques), vous pourrez effectuer une demande d'ajout de paquets, seuls les paquets disponibles pour Archlinux seront acceptés. CELA NE DOIT PAS TOURNER QUE SUR VOS LAPTOPS !**

Pour terminer avec ce cahier des charges et avant une brève conclusion, vous devez fournir les éléments classiques, c'est à dire : Un tableau de répartitions des charges (accompagné d'un descriptif précis) entre les membres du groupe ainsi qu'un planning détaillé de réalisation aligné sur les dates de soutenances.

## Les dates à retenir...

- 02/12/2016** : Le choix du groupe et la remise du synopsis.
- 09/01/2017** : Remise du cahier des charges.
- 27/01/2017** : Validation du cahier des charges.

Viennent ensuite les dates des soutenances nous menant fièrement au mois de mai.

---

1. Et évitez les tas de lard...

## Les soutenances...

Les soutenances durent, à l'exception de la dernière, 15 minutes. Elles sont au nombre de trois. Elles se suivent et donc tout retard de votre part entraîne une diminution de votre temps de parole! Il faut donc faire une présentation claire, nette et précise de l'état d'avancement de votre projet et répondre de la manière la plus claire et la plus concise possible aux questions du jury. Les deux premières soutenances auront lieu en soirée (en fait dès que cela est possible).

## Dates des soutenances...

**Première** : du 13 au 17 mars 2017

**Deuxième** : du 24 au 28 avril 2017

**Finale** : du 29 au 31 mai 2017

## Le jury...

Toutes les soutenances seront assurées par Marwan Burelle et Christophe Boullay, Sous l'oeil exercé et intransigeant des ASMs qui auront envie d'y assister. Vos suivis seront assurés tout au long du projet par le même jury.

## Bonus...

Bien évidemment, des ajouts et/ou modifications pourront être effectués en cours d'année au cahier des charges validé.

## Règles à respecter...

Pour que le développement de ce projet se passe le mieux possible, nous allons respecter les règles de codage suivantes :

— **Pour le code :**

- Vous utiliserez le standard C99
- Votre code devra obligatoirement être correctement indenté
- Tous les noms d'identifiants (fonctions, variables, constantes, macros, etc.) devront être en anglais
- Vous devrez découper votre code en plusieurs unités et chaque fichier `.c` devra être accompagné de son en-tête (fichier `.h`)
- Le code ne devra pas dépasser 80 caractères par ligne
- Il ne doit y avoir d'espaces inutiles en fin de ligne
- Votre code devra être compilé avec au moins les options suivantes : `-Wall -Wextra -std=c99`
- Vous pourrez utiliser (au choix) les compilateurs `gcc` ou `clang`
- Votre code ne devra émettre aucun *warning* à la compilation

— **Pour le projet :**

- Vous devrez utiliser le gestionnaire de versions `git` (via votre compte à l'école) et être capable de montrer en soutenance (sous la forme que vous désirez) l'évolution de votre projet et les différents apports de chacun.
- Pour la construction de votre projet, vous devrez fournir un `Makefile` permettant de compiler les différents fichiers et de produire les différents binaires. Votre `Makefile` devra également permettre de nettoyer le répertoire des produits de compilation (`.o`, exécutable, ...)
- Votre `Makefile` devra \*autant que possible\* utiliser les règles de production implicites, il devra également ne compiler que les fichiers nécessaires en respectant les dépendances.
- Votre projet devra contenir au minimum un fichier `README` (au format texte simple ou éventuellement Markdown) expliquant succinctement comment compiler et lancer votre projet.

## Description des pièces à fournir...

**A chaque soutenance** vous devez fournir :

- Un plan de soutenance,
- Un rapport de soutenance,
- Un site Web.

**A la soutenance finale** vous devez fournir :

- Un plan de soutenance,
- Un rapport de projet,
- Le projet,
- Un site Web,
- Une aide (sous la forme de fichier texte/markdown) pour la compilation et l'installation,
- Une visualisation au choix de votre historique de développement.

### Un plan de soutenance (à chaque soutenance)

Il présente sur une page et sous la forme d'un plan, la façon dont va se dérouler votre soutenance. Cela sous-entend évidemment une préparation de celle-ci (ne serait-ce que bien gérer votre temps). Il doit être remis au début de chaque soutenance.

### Un rapport de soutenance (à chaque soutenance)

Il présente sur une vingtaine de pages (minimum) ce qui a été réalisé depuis la dernière soutenance ou depuis la validation du cahier des charges pour la première soutenance. Sa présentation <sup>1</sup> peut-être calquée sur celle du rapport de projet présenté plus loin. Il doit être remis au début de chaque soutenance. Il doit faire le bilan de ce qui a été fait (avances et/ou retards sur planning), par qui et présenter ce qui doit être fait pour la prochaine fois.

### Un site Web (à chaque soutenance)

Celui-ci doit contenir une page d'accueil permettant d'accéder aux différents éléments suivants :

- Une présentation du projet (l'historique, les membres, la chronologie de réalisation, les problèmes rencontrés, les solutions envisagées),
- Les liens sur les sites (des membres, des logiciels, images, sons, librairies, applets et autres éléments que vous auriez pu utiliser),
- Un download du rapport, du projet et d'une version lite de celui-ci (sans toutes les choses inutiles à l'exécution du projet).

---

1. Toute présentation, qu'elle soit écrite ou orale, doit contenir un plan ainsi qu'une introduction et une conclusion.

## Un rapport de projet (à la dernière soutenance)

Il est à remettre lors de la dernière soutenance, il doit contenir 50 pages minimum annexes non comprises (les sources du projet sont fournies hors rapport). Une structure<sup>1</sup> acceptable pourrait être :

- Reprise du cahier des charges,
- Plusieurs présentations possibles :
  - Chronologique (groupe),
  - Chronologique (individuelle),
  - Individuelles (répartition des tâches),
  - Autres.
- Récit de la réalisation :
  - Ses joies,
  - Ses peines,
  - Etc.
- Les annexes comprennent :
  - Les exemples d'impression,
  - les exemples d'écran,
  - les jeux d'essai,
  - les dessins d'origine,
  - Etc.

### **Remarques :**

Quelle que soit la structure retenue pour les rapports (chronologique, individuelle, etc.), elle doit dans tous les cas décrire de manière très EXPLICITE qui a fait quoi.

## Une visualisation de votre historique de développement (à la dernière soutenance)

Si vous avez correctement et régulièrement utilisé votre gestionnaire de version, celui-ci peut vous fournir un historique de l'ensemble des modifications apportées par chaque participant (commit) qui peuvent être visualisées sous différentes formes. Il existe de nombreux outils de visualisation pour `git`, à vous de trouver celui qui vous convient.

## Le projet (à la dernière soutenance)

Le projet (un exécutable) doit être accompagné des éléments suivants :

- Les sources,
- Les différentes bibliothèques utilisées par votre projet,