

# **Projet JAVA**

## **1 Présentation**

Le sujet présente une problématique générale et une approche pour résoudre de façon pratique, cette problématique. Certains points ne sont pas totalement spécifiés car c'est justement à l'étudiant d'être capable de s'en accommoder. Le prof peut agir comme le "client" et aider l'étudiant à spécifier certains de ces points.

## **2 Documents définitif à fournir :**

Pour ce projet, vous aurez à fournir :

1. Un rapport sur papier comportant entre 8 et 15 pages. Celui-ci rappelle la problématique du projet et en donne les spécifications. Il détaille ensuite l'analyse effectuée par l'étudiant et comment celle-ci est implémentée (conception). Un soin particulier doit être pris pour expliquer comment les diverses tâches ont été réparties entre les différents étudiants travaillant sur le projet. Il est très important également de rappeler que ce rapport ne doit pas comporter les lignes de codes du logiciel. Enfin, le rapport doit présenter les tests effectués, une conclusion et des perspectives.

2. Le code source (sous la forme d'une archive .zip ou .tar.gz). Il s'agit de l'ensemble des fichiers permettant de compiler et exécuter votre logiciel. Le code doit être correctement présenté (indentation), clair (utilisation de norme précise pour les noms) et commenté (intelligemment).

L'ensemble de ces fichiers est à fournir sous la forme d'une archive (.zip ou .tar.gz).

## **3 Cahier de charge :**

### **3-1 Description :**

Ce projet consiste à développer une application de suivi des projets et tâches académiques.

Un professeur a deux fonctions :

-Enseignement : assurer des cours

-Recherche et encadrement : encadrer des PFE et des thèses de doctorat

Nous souhaitons réaliser une application qui aide le professeur à gérer ses tâches.

Le professeur gère ses tâches **en mode projet et/ou liste todo.**

Un projet est défini par :

- une catégorie (Enseignement, Encadrement, Autre)
- Un type (Thèse, PFE, PFA, Cours, Examen, Autre)
- Une description : Texte qui décrit le projet
- Une durée : Date de départ ( défini au démarrage de projet), date de fin peut être défini vers la fin du projet
- Chaque projet peut avoir plusieurs séances de travail. Ces séances peuvent être ajoutées directement par l'application ou importer de Google calendrier
- Une séance est défini par une description, une date de début et date de fin de même jour, on peut attacher à une séance une note écrite qu'on rédige à la fin
- On peut attacher plusieurs documents au projet ou à une séance de travail spécifique
- Un document est défini par une description et une date d'ajout
- Un projet peut être clôturé

Une tâche est défini par :

- une catégorie (Enseignement, Encadrement, Autre)
- Une description : Texte qui décrit la tâche
- Une durée : Date de départ ( défini au démarrage de tâche), date de fin
- On peut attacher plusieurs documents à la tâche
- Une tâche peut être attachée à un projet
- Une tâche peut être apportée de Google calendrier ou créée dans l'application
- Une tâche peut être dans une liste
- Une liste n'appartient à aucun projet.
- Une liste a les caractéristiques suivantes :
  - Description
  - Une liste des tâches

### **3-2 Fonctionnalités :**

L'application doit réaliser les fonctionnalités suivantes :

- 1-Le professeur doit se connecter avec un email de Google
- 2-Le professeur peut lister tous les projets et les filtrer par catégorie ou type
- 3-Le professeur peut faire une recherche sur les projets par mots clés (description) et consulter un projet (historique)
- 4-Le professeur peut ajouter, clôturer, cloner ou modifier les informations d'un projet
- 5-Le professeur peut créer, modifier la description, ajouter une tâche : existante ou créer une tâche, supprimer une tâche : une tâche supprimée de la liste est toujours existante, supprimer une liste : les tâches ne sont pas supprimées
- 5-Le professeur peut consulter les tâches par ordre chronologique et les filtrer par catégorie, état ( fini ou non) ou date
- 6-Le professeur peut faire une recherche sur les tâches par mots clés (description)

7-Le professeur peut ajouter, cloner ou modifier les informations d'une tâche  
8-Le professeur peut importer des tâches/séances de son calendrier Google et attribuer chaque séance/tâche à un projet existant ou créer une tâche avec la séance

9-Le professeur peut gérer ses séances : consultation, ajout, modification ( informations ou documents) et suppression

10- Le professeur peut effectuer une recherche par mots clés sur tous les documents ou les documents d'un projet ou une tâche

11- On veut visualiser les statistiques suivantes :

- Nombre d'heures de travail sur un projet
- Nombre de documents par projet
- Nombre d'heure de travail par semaine, mois et année
- Pourcentage d'heure de travail par catégorie ou type par semaine, mois et année

12- On souhaite ajouter les fonctionnalités optionnelles suivantes :

- OCR : ajout la recherche sur le contenu des documents
- Rédaction des notes de chaque séance par voix vocale

### **3-3 Contraintes techniques :**

On souhaite réaliser notre application avec les technologies suivantes :

1- **Java**

2-Interface graphique : Swing ou JavaFX

3-Stocage de données : Fichiers Json

4-Tous les paramètres de configuration de l'application doivent être stockés dans un fichier propriétés

L'application est une application bureau qui doit être déployée dans les OS (Windows, Mac, Linux). Vous devrez préparer les éléments suivants :

-Fichiers d'installation pour chaque OS

-Guide d'installation

## **4 Le rapport :**

Le rapport doit comporter :

1. Une introduction : elle replace ce rapport dans le contexte de votre formation.

2. Les spécifications : partie normalement rédigée conjointement avec le client qui comprend une description en langue naturelle de ce que doit faire le programme. Dans notre cas, les spécifications seront rédigées à partir des énoncés et des questions posées à l'enseignant.

3. L'analyse : cette partie explique comment les spécifications vont être réalisées. Elle comporte :

- La maquette des interfaces graphique + diagramme de navigation
- Le dictionnaire des données
- MCD

Toute autre ajout sera le bienvenu pour bien expliquer le projet.

4. La conception : cette partie explique comment l'analyse est implémentée, c'est-à-dire :
  - La conception générale de chaque partie de projet (Architecture de chaque partie du projet : voir l'autre document)
  - Documentation des API utilisés et leurs interactions avec le logiciel
  - Donnez un exemple commenté d'utilisation du logiciel du début à la fin.
5. La validation : spécifier les procédures de test de votre logiciel, c'est-à-dire :
  - les tests unitaires : comment les méthodes ont été testées individuellement ? Donnez quelques résultats (voir l'autre document) ;
  - les tests globaux : en utilisant votre interface, quels cas particuliers avez-vous testés ?
6. La maintenance : donnez quelques évolutions possibles du logiciel. Quelles sont les fonctionnalités que l'on peut rajouter et comment ? Pourquoi cela sera-t-il facile ou difficile ?
7. Une conclusion : quel est votre avis sur ce projet, sur les résultats produits et sur les perspectives éventuelles ?
8. Un guide d'industrialisation devra être ajouté en annexe.

## **6 Notes complémentaires :**

- Tout étudiant peut être appelé à présenter son projet à l'enseignant et à répondre à ses questions, même si le projet a été réalisé à plusieurs. Cette présentation peut rentrer dans le calcul de la note finale du projet.
  - Un programme qui fonctionne mais qui fait peu de choses sera mieux noté qu'un programme qui ne fonctionne pas du tout (par définition qui ne fait rien). Autrement dit : l'étendue des fonctionnalités réalisables par l'interface (et qui fonctionne) est un critère important.
  - Votre code doit respecter les normes des bonnes pratiques de la programmation et bien commenté
  - Plus les mécanismes objets et les particularités de JAVA seront mis à profit, plus la note sera élevée (et inversement).
  - Une bonne gestion du projet sera appréciée lors de la soutenance (Une bonne répartition des tâches, diagramme de gant : décrit le déroulement du projet ...)
  - L'utilisation du code d'autres auteurs (et libre de droit) est autorisée à condition qu'elle soit clairement signalée. Ces codes ne seront simplement pas pris en compte dans la note. Au contraire, une utilisation non signalée annule le reste du travail.
- Bonne chance !