

# Projet Symfony

## *Gestion des notes du département Informatique*

### Objectifs pédagogiques

Il s'agit de développer une application **Symfony** complète en partant de (presque) rien afin d'explorer toutes les fonctionnalités offertes par ce framework. L'application à développer étant relativement conséquente, le travail devra être effectué par groupe de 3 à 5 étudiants (plutôt 4 à 5 si possible) et devra donc faire appel aux techniques de génie logiciel vues dans le cours de *Méthodes*.

#### Planning :

- début : semaine du 4 janvier ;
- rapport d'analyse détaillé (cas d'utilisation commentés, diagramme de classe commenté, diagramme relationnel des données) : semaine du 18/01 ;
- démo/soutenance finale : semaine du 7 mars.

## 1 Existant et besoins

La gestion des notes du département Informatique est assurée, comme pour toutes les composantes de l'université, par le logiciel de gestion de scolarité APOGEE. Il permet l'importation des notes pour un module donné et calcule les moyennes et les résultats (théoriques avant jury). Il permet l'édition des bulletins ainsi que des procès-verbaux (PV) de jury.

Cependant, ce logiciel est peu adapté à la prise de décision : les PV sont peu lisibles et ne font apparaître que les résultats du semestre en cours. De plus le département édite à la fin du S4 théorique un avis de poursuite d'études pour chaque étudiant, rappelant l'essentiel de son parcours.

Pour pallier les faiblesses d'APOGEE, le département maintient en parallèle des feuilles Excel de synthèse des résultats, alimentées par copier-coller depuis les feuilles d'importation d'APOGEE.

L'idée est de développer une application qui remplace ces feuilles Excel et ajoute des fonctionnalités :

- favorisant la prise de décision en jury (par exemple une fiche élève récapitulant le parcours complet d'un étudiant) ;
- simplifiant la réalisation des fiches de poursuite d'études, avec notamment la saisie en ligne des commentaires de chaque enseignant sur chaque élève.

## 2 Cahier des charges

L'application doit gérer les notes des étudiants à chaque semestre et permettre de visualiser le parcours d'un étudiant sur les quatre semestres passés à l'IUT, voire plus en cas de redoublement. Une année universitaire (2015/2016) sera identifiée par l'année de début (2015).

Les étudiants seront identifiés par leur numéro d'étudiant et on ne conservera ici que leur nom et leur prénom.

La liste des modules par UE est fournie dans le fichier **CASIR/PPN-Info-2013.csv** de l'archive **ProjetSymfony.tgz** accessible sur mon intranet : <http://intranet/genthial>

Notez qu'il manque à cette liste de module la matière optionnelle *Sport* qui ajoute une note hors UE à chaque semestre. Cette note vient donner un bonus sur la moyenne générale du semestre (quand elle est présente).

Les fichiers **S1-S2.pdf** et **S3-S4.pdf** contiennent une version plus lisible de la liste des modules.

Les enseignants seront identifiés par leur login et on ne gardera ici que leur nom et leur prénom. Le fichier **Enseignants.csv** de l'archive **ProjetSymfony.tgz** contient la liste des enseignants pour l'année en cours. Les étudiants n'auront pas accès à l'application (dans un premier temps).

Un enseignant devra pouvoir :

- visualiser toutes les notes, statistiques et pages d'aide à la décision ;
- visualiser la fiche détaillée d'un étudiant ;
- importer les notes d'un module dont il est responsable (1 seul responsable par module) ;
- modifier la note d'un étudiant dans un module dont il est responsable.
- remplir les avis de poursuite d'études pour la ou les modules dont il est responsable, dès que les notes du semestre 4 (théorique) sont toutes saisies.

### **Fiche d'un étudiant :**

La fiche doit faire apparaître la liste des semestres suivis par l'étudiant avec les moyennes par UE de chaque semestre, la moyenne générale du semestre ainsi que la décision de jury (voir **Decisions.csv** dans **ProjetSymfony.tgz**).

Elle doit également donner accès au détail des notes de chaque UE pour un semestre donnée.

### **Authentification :**

L'application sera intégrée à l'intranet dans lequel l'authentification est déjà gérée. On fera une page d'authentification mais on supposera que le mot de passe est toujours correct. Cela permettra de changer à volonté d'utilisateur.

### **Visualisation des résultats pour le jury :**

Le document utilisé par le jury est la liste des étudiants classés par ordre décroissant de moyenne générale du semestre en cours. L'affichage sous forme de liste est important et devra être conservé. On souhaite pouvoir identifier l'étudiant courant dans la liste de manière à pouvoir obtenir de manière simple ses résultats aux semestres précédant le semestre en cours.

On prévoira la saisie de la décision du jury pour le semestre en cours mais également pour le semestre précédent (le semestre 3 peut être utilisé pour compenser le semestre 2). Cette saisie ne sera accessible qu'à certains enseignants identifiés comme gestionnaires de scolarité (chef de département, directeurs des études).

### **Avis de poursuite d'études :**

Pour chaque étudiant, l'enseignant remplit un avis, mais ces avis sont regroupés : un enseignant ne donne qu'un avis par étudiant, même s'il est responsable de plusieurs modules (ce qui est souvent le cas puisque les modules sont ceux des deux semestres de la 2<sup>e</sup> année. Deux modèles d'avis de poursuite d'études est fourni dans **AvisAP.pdf** et **AvisAT.pdf**. Les deux versions correspondent aux deux parcours proposés au semestre 4.

Par défaut, les étudiants sont proposés à l'enseignant par ordre alphabétique, mais l'enseignant pourra choisir l'ordre décroissant de la moyenne générale.

Il peut être intéressant de rappeler les 5 derniers avis saisis lors de la saisie d'un nouvel avis.

L'application ne produit pas vraiment les avis de poursuite d'études, elle permet à l'enseignant de saisir ses avis et les avis seront imprimés ensuite par fusion avec une feuille Excel ou un document Word.

### **Importation :**

Pour simplifier le travail des enseignants, l'importation des notes pour une module s'appuiera sur le même fichier CSV que celui utilisé pour l'application APOGEE. Un exemple est fourni dans **Import.csv**.

### **Exportation :**

En cas de besoin, on doit pouvoir exporter les données (les notes) dans un fichier CSV. Pour faire

simple dans un premier temps, on exportera les notes de tous les étudiants pour un semestre donné, avec une colonne pour chaque module. Il est inutile ici de recalculer les moyennes par UE.

**Archivage :**

Les notes sont associées à un étudiant pour un semestre sur une année scolaire donnée, donc on conservera indéfiniment toutes les notes et tous les étudiants. Cela implique qu'on choisisse une année de référence au lancement de l'application (par défaut l'année en cours).

### 3 Contraintes techniques

L'utilisation du framework **Symfony** est imposée car l'objectif du projet est bien entendu la réalisation de l'application mais également l'acquisition d'une vraie expérience avec ce framework.

Le design n'est pas prioritaire, il est toutefois recommandé d'utiliser framework CSS/Javascript (typiquement **Bootstrap**) pour obtenir un design convenable avec un minimum d'efforts.

Pour ne pas être pollué par d'autres applications, il est recommandé de vider la base **MySQL** utilisée avant de commencer.

Un jeu d'essai réel vous sera fourni avec les étudiants et les notes des deux dernières années universitaires.