# Fiche de synthèse du projet d’année : FEG Project

**Titre du projet :** Création d'une application en ligne pour la gestion des candidatures étudiants de la Faculté d'Economie et Gestion.

**Tuteurs :** Nicola Olivetti, Jean Caussanel

**Equipe projet :** David Cohen, Lionel Guissani, Hasan Karagoz, Kevin Meas

Contenu

[Fiche de synthèse du projet d’année : FEG Project 1](#_Toc377931326)

[1. Présentation détaillée du projet (3 pages) 2](#_Toc377931327)

[1.1. Cahier des charges 2](#_Toc377931328)

[1.1.2 Contraintes 3](#_Toc377931329)

[1.1.3 Budget 4](#_Toc377931330)

[1.1.4 Suivi projet 4](#_Toc377931331)

[1.2. Etude fonctionnelle 4](#_Toc377931332)

[1.3. Etude technique 8](#_Toc377931333)

[2. Gestion de projet 16](#_Toc377931334)

[2.1. Description générale 16](#_Toc377931335)

[2.2. Outils utilisés 16](#_Toc377931336)

[2.3. Réunion 17](#_Toc377931337)

[2.4. Difficultés rencontrés 17](#_Toc377931338)

[3. Etat d’avancement 17](#_Toc377931339)

[3.1. Ce qu’il nous reste à faire 17](#_Toc377931340)

[3.2. Ce que nous sommes en train de faire 17](#_Toc377931341)

[3.3. Ce que nous avons fait 18](#_Toc377931342)

## 1. Présentation détaillée du projet (3 pages)

### 1.1. Cahier des charges

#### 1.1.1 Le projet

##### 1.1.1.1 PRÉSENTATION

Le projet est de réaliser une application en ligne permettant la gestion (création, consultation, mise à jour et suppression) des dossiers de candidature et de pré-inscription de la Faculté d’Economie Gestion.

##### 1.1.1.2 Enjeux

Les enjeux de la création de cette application sont :

• Le gain de temps pour le traitement des dossiers

• Apport de plus de facilités pour la gestion de la structure des dossiers

• Harmonisation

##### 1.1.1.3 Objectifs

L’objectif est de fournir au personnel de la faculté une application efficace, ergonomique et fonctionnelle

##### 1.1.1.3 Prestations complémentaires

###### 1.1.1.3.1 Design

Le design du site suivra la charte graphique de la Faculté d’Economie et de Gestion.

###### 1.1.1.3.2 Architecture

L’architecture du site suivra le modèle MVC (Modèle – Vue – Contrôleur).

###### 1.1.1.3.3 Ergonomie

Le site sera conforme aux dernières innovations en termes d’ergonomie.

Le site ne sera développé qu’en version ordinateur, une version mobile du site ou application pour smartphone n’est pas prévue.

### 1.1.2 Contraintes

#### 1.1.2.1 Contraintes techniques

##### 1.1.2.1.1 TECHNOLOGIES

###### 1.1.2.1.1.1 Base de données

Le SGBD cible est MySQL.

###### 1.1.2.1.1.2 Langage serveur

Le langage serveur est PHP.

###### 1.1.2.1.1.2 CMS

Le site sera construit sur une architecture originale est ne se basera pas sur un CMS.

###### 1.1.2.1.1.3 COMPATIBILITÉ

Le site doit être compatible tous navigateurs.

###### 1.1.2.1.1.4 HEBERGEMENT

Le site sera hébergé sur un serveur mutualisé chez OVH.

##### 1.1.2.1.2 Ressources

Notre équipe projet regroupe diverses compétences :

###### 1.1.2.1.2.1 Programmation WEB

Langages :

• PHP

• JavaScript, AJAX, JSON

• HTML & CSS

• XML

###### 1.1.2.1.2.2 Bases de données

MySql :

• Conception et administration de base de données

• Maîtrise des langages SQL et SQL/PSM

###### 1.1.2.1.2.3 Compétences complémentaires

Compétences dans les framework Bootstrap, jQuery, Symfony.

Compétences en ergonomie des sites internet.

### 1.1.3 Budget

Il n’y a pas de budget prévu pour le projet.

### 1.1.4 Suivi projet

Après livraison du produit, la maintenance et l’évolution du site seront effectuées par le client.

### 1.2. Etude fonctionnelle

#### 1.2.1. Présentation générale

L’application a pour but de faciliter les pré-inscriptions et candidatures des étudiants pour la faculté d’économie et de gestion d’Aix-Marseille. Elle permet d’informatiser et d’harmoniser tous les dossiers papiers qui existent à l’heure actuelle. Afin que les dossiers PDF correspondent au plus aux exigences des formations, il est possible de les personnaliser via l’intranet.

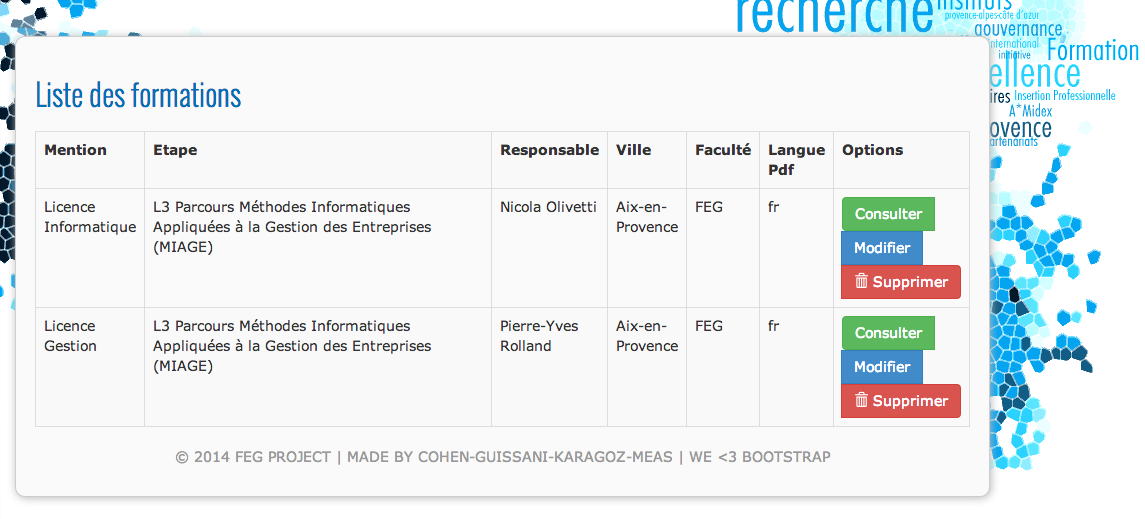
#### 1.2.2. Fonctionnalités

##### 1.2.2.1. Gestion de l’intranet

Le personnel peut s’identifier sur l’intranet afin de gérer les dossiers de pré-inscription et de candidature. Il peut en outre, ajouter, modifier et supprimer :

* Les pièces à joindre commune à toutes les formations
* Les pièces à joindre spécifiques à chaque formation
* Les informations communes à toutes les formations
* Les informations spécifiques à chaque formation
* Les formations (i.e. une nouvelle formation qui vient de s’ouvrir dans l’université pourra utiliser l’application)

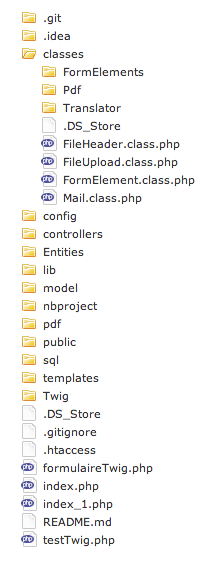
Le dossier de candidature PDF est commun à toutes les formations (c’est-à-dire que les champs sont tous les mêmes). Pour les formations qui souhaitent demander plus d’informations aux étudiants, le personnel pourra compléter les dossiers de candidature/ préinscription par des champs supplémentaire depuis l’intranet. Les dossiers de candidatures PDF sont ainsi dynamique et pourront être adaptés aux besoins futurs. Les champs ainsi créés pourront être supprimés par la suite.



*Gestion des formations sur l’intranet.*

**1.2.2.2. Explorateur de fichiers**

Le personnel a la possibilité de télécharger les divers répertoires de candidature des étudiants. Les répertoires peuvent être soit téléchargés en totalité soit un par un par. Une fois les répertoires téléchargés, il est possible de les supprimer en cliquant sur une petite croix rouge. Lors du clic, une fenêtre s’ouvre afin de demander la confirmation de suppression à l’utilisateur.



*Exemple d’explorateur de fichier qui va être mis en place pour l’intranet.*

##### 1.2.2.3. Formulaire de formation

Ce formulaire contient deux listes déroulantes :

* Dernière formation effectuée et validée
* Formation souhaitée

##### 1.2.2.4. Formulaire d’inscription

Ce formulaire permet à l’étudiant d’indiquer toutes ses informations personnelles (nom, prénom, numéro INE …), la spécialité qu’il souhaite effectuer, son cursus post-bac et les pièces à joindre.

##### 1.2.2.5. Générer un document de candidature PDF

Un document PDF va être généré en fonction de la nature de la demande (pré-inscription ou candidature). Tous les champs du PDF sont pré-remplis grâce aux informations récoltés en amont via les formulaires.

##### 1.2.2.6. Uploader des documents

Plusieurs champs vont permettre l’upload des pièces justificatives (différents selon les formations). Certains champs sont obligatoires et doivent être obligatoirement complétés.

##### 1.2.2.7. Envoyer un mail à l’étudiant

Un mail est envoyé à l’étudiant lorsqu’il a finalisé son inscription. Ce mail confirme que la candidature de l’étudiant a bien été pris en compte. Le document PDF généré précedemment est en fourni en pièce jointe comme récapitulatif.

##### 1.2.2.8. Création de répertoire de candidature sur le serveur

Une fois que l’étudiant a finalisé son inscription, toutes les informations récoltées sont stockées dans un répertoire compressé au format ZIP qui a pour nom, le numéro INE, le nom et le prénom de l’étudiant. Ce répertoire contient toutes ses pièces justificatives et le dossier de candidature au format PDF.

##### 1.2.2.8. Grille de synthèse des candidatures au format CSV stockée sur le serveur

Un fichier de synthèse excel sera généré avec les données renseignées dans le formulaire et d’autres informations pour le personnel et la commission.

Les colonnes sont composées des éléments suivants :

* Classe de diplôme
* Nom
* Prénom
* E-mail
* Téléphone
* Diplômes d’origine
* Année de naissance
* Candidate en
* Avis d’admissibilité
* Avis d’admission
* Année du BAC
* Dernier diplôme préparé
* Obtenu (non/oui/en cours)
* Moyenne année N
* Moyenne année N-1
* Moyenne année N-2
* Nombre d’années depuis le dernier diplôme obtenu
* Lettre de recommandation
* Commentaires

**1.2.2.9. Liens disponibles sur le site de l’AMU (Drupal)**

Sur chaque page de formation du site de l’université se trouve un lien qui va rediriger l’utilisateur vers l’application afin qu’il puisse postuler à la formation qui correspond à sa demande.

### 1.3. Etude technique

#### 1.3.1. Présentation générale

L’architecture du projet FEG Project est basée sur le modèle MVC.

Afin de développer l’application, nous avons utilisés les langages suivants :

* PHP 5.5.3
* MySQL
* JavaScript avec jQuery 2.0.3
* HTML5
* CSS3 avec Bootstrap3

Nous avons également utilisés les ressources suivantes :

* Un serveur OVH mutualisé
* Moteur de template TWIG
* jquery-form-validator.js : Permet de vérifier la validité des champs d’un formulaire
* jqueryFileTree.js : Permet d’afficher l’arborescence complète d’un répertoire
* tourist.js : Permet de guider pas à pas graphiquement un utilisateur
* html2pdf : Permet de générer un document PDF à partir d’un code HTML et CSS
* Framework PHP personnel de Lionel Guissani
* PHPUnit

#### 1.3.2. Description des processus

##### 1.3.2.1. Identification sur l’intranet

Les utilisateurs internes (personnel de l’AMU) de l’application peuvent se connecter sur l’intranet grâce à un unique identifiant et mot de passe qui sont stockés dans la base de donnée.

##### 1.3.2.2. Explorateur de fichiers

Le plugin jqueryFileTree.js permet de récupérer tous les noms de répertoires et sous répertoires ainsi que les noms de fichiers. Les repertoires sont ensuite affichés dans l’intranet. Ils sont cliquables et l’on peut accèder aux fichiers qu’ils contiennent. Tout est réalisé de manière dynamique en AJAX.

##### 1.3.2.3. Liens disponibles sur le site de l’AMU (Drupal) et formulaire de formation

Afin d’accèder à notre application, l’étudiant souhaitant postuler pour une formation doit d’abord passer par le site de l’AMU dans la section de la formation souhaitée. Sur cette rubrique se trouve un lien dont l’URL est de la forme :

[www.candidature-feg.fr/?uc](http://www.candidature-feg.fr/?uc)=formulaireChoix&formation=code\_étape

Le contrôleur de l’application analyse l’URL et génère le formulaire de formation adapté pour l’étudiant. Ensuite selon les choix effectués, un formulaire d’inscription correspondant à la formation demandée est créé.

##### 1.3.2.4. Formulaire d’inscription

Le formulaire d’inscription est généré grâce à la fonctionnalité décrite précedemment. Le plugin jquery-form-validator.js permet de vérifier que tous les champs du formulaire qui sont obligatoires ont bien été remplis afin que le formulaire puisse être validé. Une fois le formulaire validé, toutes les données rentrées sont sauvegardées dans la base de données.

##### 1.3.2.5. Générer un document PDF

Le PDF est généré grâce à une requête vers la base de données et la classe HTML2PDF. Le PDF généré est conçu dynamiquement (i.e. les champs ne sont pas les mêmes selon les données stockées en base de données). Cependant il contient une partie commune à toutes les formations. Il n’est pas possible de modifier cette partie.

##### 1.3.2.6. UPLOADEr des documents

Les documents qui sont uploadés vont être stockés sur le serveur, dans le même répertoire que le fichier PDF qui a été généré.

##### 1.3.2.7. CREATION DES répertoires étudiants sur le serveur

Les répertoires de candidature ont l’architecture suivante :

* Répertoire : Nom de la formation
  + Répertoire : Nom et prénom du candidat
    - Répertoire : Documents
      * Fichiers : Pièces justificatives …
    - Fichier : Dossier de candidature/pré-inscritpion

##### 1.3.2.8. Envoyer un mail à l’étudiant

Une fois l’inscription finalisé un e-mail de confirmation est envoyé à l’étudiant. Ce mail est envoyé grâce à une classe PHP.

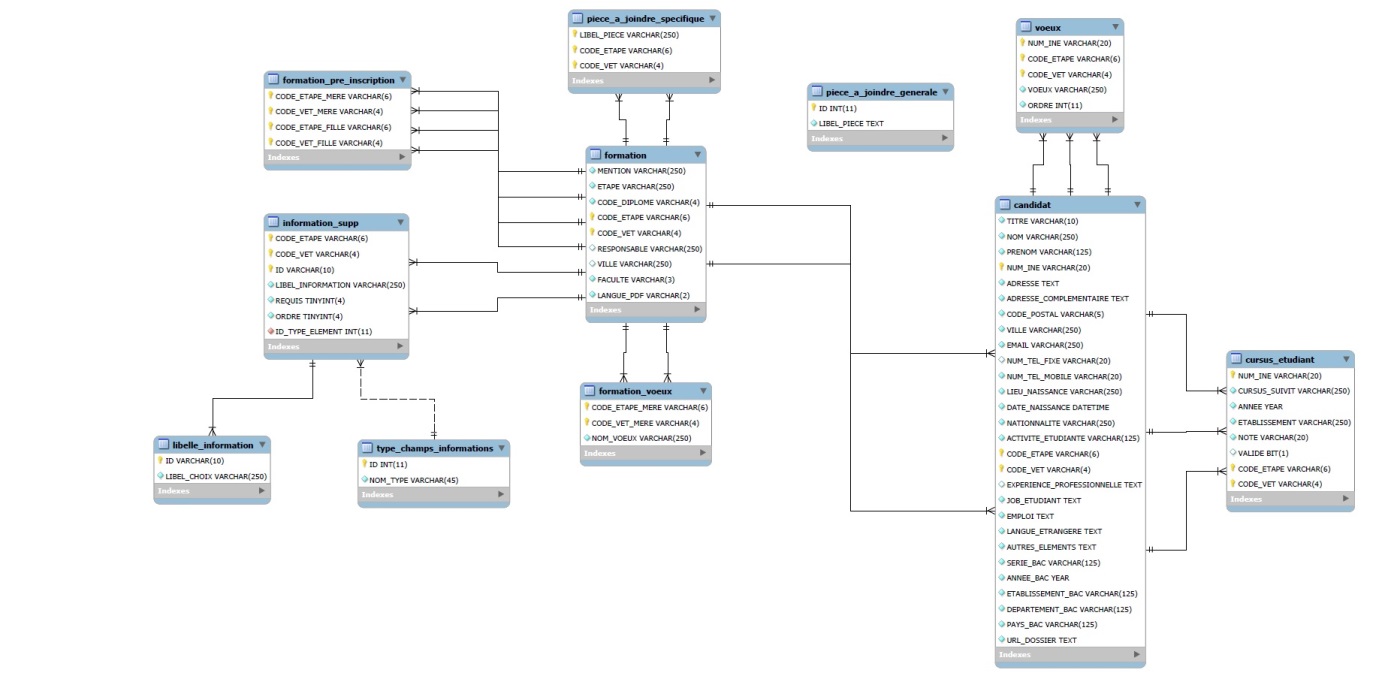
##### 1.3.2.9. Grille de synthèse des candidatures au format CSV

Un script PHP génère la grille de synthèse des candidatures.

#### 1.3.3. Description de la base de données

La base de données est géré par le SGBD MySQL.

La connexion se fait via le PhpMyAdmin du site OVH qui gère le serveur ou est installé le serveur de base de données.

La connexion se fait via <https://phpmyadmin.ovh.net/> , le login utilisateur est : **miageaixcandid**, le mot de passe est : **candidfeg13**, le serveur est le : **mysql51-63**, **pro**.

##### 1.3.3.1 Dictionnaire de données

Le dictionnaire de données retraçant les différents champs, types et commentaires pour les différentes tables de données se trouvent dans le document Excel ci-joint.

##### 1.3.3.2 Explications de la base de données

###### Table Candidat

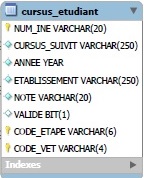
La table candidat est une des tables les plus importantes de cette base de données. En effet c’est dans cette table que seront stockées une grande partie des informations que les candidats rentreront sur le site Internet de candidature.

Clés primaires: NUM\_INE, CODE\_ETAPE, CODE\_VET

Cette table a été créée dans le but que chaque étudiant puissent postuler à une ou plusieurs formations, c’est pourquoi le Numéro INE de l’étudiant n’est pas une clé primaire simple, la clé primaire de cette table es composé de 3 champs. Le CODE\_ETAPE et CODE\_VET étant ce qui identifie de manière unique une formation, la table permet donc à un étudiant de postuler plusieurs fois mais pour des formations différentes.

Les informations que le candidat va rentrer dans cette table sont des informations générales, c’est à dire qu’importe que ce soit une candidature ou une pré-inscription ces informations seront demandées.

###### Table cursus\_etudiant



Cette table a pour but de contenir les dernières formations suivit par un étudiant candidatant sur une formation.

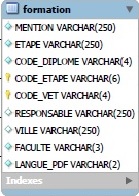
Clés Primaires : NUM\_INE, CODE\_ETAPE, CODE\_VET

Clés étrangères : NUM\_INE, CODE\_ETAPE, CODE\_VET qui font référence à la table candidat.

Le fait de d’avoir NUM\_INE, CODE\_ETAPE, CODE\_VET

pour clés étrangère permet de créer un lien entre la table candidat et cursus\_etudiant ce qui facilite l’accès et le regroupement aux données par candidat.

###### Table formation



La table formation est aussi l’une des principales table de cette base de données.

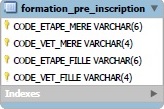
Clés Primaires : CODE\_ETAPE, CODE\_VET

Cette table regroupe l’ensemble des formations disponibles.

Dans un souci d’optimisation un champ faculté a été créé, celui-ci permettra si permettra si le besoin s’en fait ressentir de proposer ce projet à d’autres facultés que la Faculté d’Economie Gestion.

Dans cette sont enregistrées toutes les informations relatives à une formation, que ce soit le parcours dans lequel elle s’inscrit ou le responsable de celle-ci.

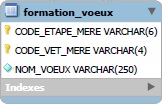
###### Table formation\_pre\_inscription

Cette table contient toutes les dépendances entre les différences formations, c’est grâce à cette table que l’on sait si une candidature sera une pré-inscription ou une simple candidature par rapport aux différents cursus que les étudiants auront suivit avant.

Clés primaires : CODE\_ETAPE\_MERE, CODE\_VET\_MERE, CODE\_ETAPE\_FILLE, CODE\_VET\_FILLE

Clés étrangères : CODE\_ETAPE\_MERE, CODE\_VET\_MERE, CODE\_ETAPE\_FILLE, CODE\_VET\_FILLE qui font référence aux clés primaires CODE\_ETAPE, CODE\_VET de la table formation.

###### table formation\_voeux

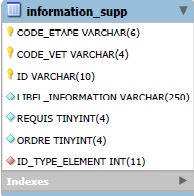


Cette table contient tous les différents vœux possibles qui sont disponibles pour chaque formation dans laquelle un étudiant peut postuler.

Clés Primaires : CODE\_ETAPE, CODE\_VET

Clés étrangères : CODE\_ETAPE, CODE\_VET qui font référence à la table formation.

###### Table information\_supp

Cette table va contenir les informations supplémentaires que demande chaque formation. Le problème étant que chaque formation ne demande pas les même. Cette table n’est donc qu’une partie du système que nous avons mis en place. Cette table contiendra le nom de l’information, son ordre d’apparition dans le système d’inscription et enfin. Elle contiendra aussi un ID qui donnera le type du champs ( CheckBox, textBox ..).

Clés primaires : CODE\_ETAPE, CODE\_VET, ID

Clé étrangère : ID\_TYPE\_ELEMENT qui référence la table type\_champs\_informations.

###### Table libelle\_information

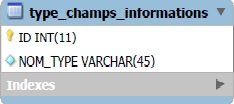
Cette table va contenir les différents choix possibles pour une information.

Par exemple pour un champ de type CheckBox, cette table contiendra les différents choix de cette CheckBox.

Clé primaire : ID

Clé étrangère : ID référence la table information\_supp

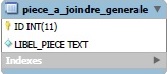
###### Table type\_champs\_information

Cette table contient les différents types de données qui définiront les informations supplémentaires. Il y’a 7 différents types : TextBox , CheckBox , CheckBoxGroup , TextArea, RadioButtonGroup, Select, SelectGroup .

Clé primaire : ID

Clé étrangère : ID référence la table information\_supp (ID\_TYPE\_ELEMENT).

###### Table piece\_a\_joindre\_generale

Cette table contient les différentes pièces à joindre générales, ces pièces sont celles qui sont demandés qu’importe la formation et le type de candidature. Cette table n’a pas de relation, elle ne dépend d’aucunes autres tables.

Clé primaire : ID

###### Table piece\_a\_joindre\_specifique

Cette table va contenir les différentes pièces à joindre spécifiques que chaque formation peut demander.

Le nom de la pièce à joindre est enregistré dans LIBEL\_PIECE, alors que CODE\_ETAPE et CODE\_VET identifie la formation à laquelle la pièce est rattaché.

Clé primaire : LIBEL\_PIECE, CODE\_VET, CODE\_ETAPE.

Clés étrangères : CODE\_VET, CODE\_ETAPE qui font référence à la table formation.

###### Table vœux

Cette table enregistre les différents vœux que le candidat a fait.

Les vœux seront ordonnés grâce au champ ORDRE.

Clés Primaires : NUM\_INE, CODE\_ETAPE, CODE\_VET

Clés étrangères : NUM\_INE, CODE\_ETAPE, CODE\_VET qui font référence à la table candidat.

## 2. Gestion de projet

### 2.1. Description générale

Les membres de l’équipe ont les rôles suivants :

* David Cohen : Administrateur & concepteur de la base de données
* Lionel Guissani : Développeur Back-end & concepteur de la base de données & Desgin
* Hasan Karagoz : Développeur Front-end & Design
* Kevin Meas : Développeur Back-end & Design

### 2.2. Outils utilisés

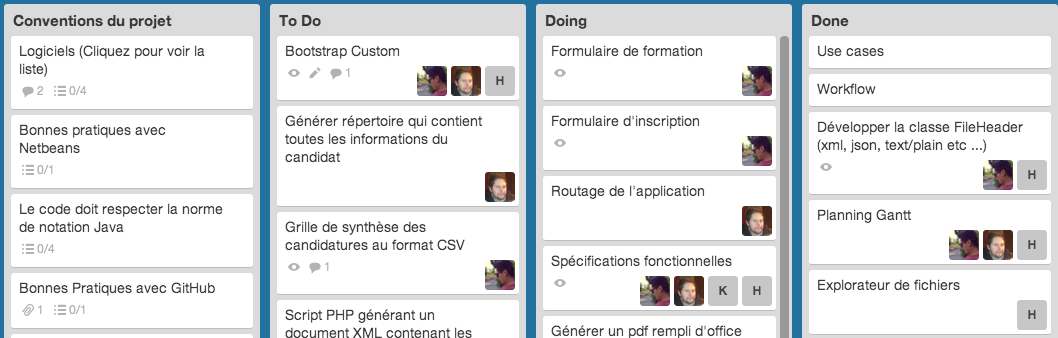
Afin de bien gérer notre projet nous avons utilisés divers outils.

#### 2.2.1. Système de gestion de version Git avec Github

Git couplé à Github permet de mettre en place un système de versionning pour notre projet. Il est plus facile pour les membres de l’équipe de récupérer ce que les autres ont fait via la plateforme de Github. Par ailleurs, si jamais une modification provoque un bug, il est très facile de pouvoir revenir en arrière et de corriger cette modification.

#### 2.2.2. Trello

Trello est un site qui permet d’organiser des projets. Il est possible d’assigner des tâches à des personnes. Ainsi grâce à cet outil, chaque membre de l’équipe sait ce qu’il a faire. Par ailleurs, une fois que la tâche est réalisée, il peut le signaler aux autres membre du groupe via le site. Ainsi tout le monde peut voir l’état d’avancement du projet.



*Apperçu du site Trello*

#### 2.2.3 Google drive

Nous utilisons Google drive afin que tout le monde puisse avoir accès aux divers documents techniques qui sont nécessaires pour la réalisation de l’application. Nous y notons également un petit résumé de toutes les réunions que nous effectuons avec nos tuteurs du projet.

#### 2.2.4. Diagramme de Gantt

Afin de bien planifier le projet, nous utilisons le diagramme de Gantt. Il nous permet d’avoir une idée de l’état d’avancement du projet et de ce qu’il reste à faire.

### 2.3. Réunion

En général, nous effectuons une réunion toutes les semaines. Ces réunions nous permettent de poser des questions aux tuteurs du projet afin d’avoir plus de précisions sur certaines tâches et de leur montrer ce que nous avons réalisés.

### 2.4. Difficultés rencontrés

* Problème de communication au sein de l’équipe et donc incompréhension des tâches à réaliser

## 3. Etat d’avancement

Il est prévu que nous finissons le projet d’ici début février afin de pouvoir réaliser une grande série de tests sur l’application développée. Nous pensons que nous sommes dans les temps que nous pourrons livrer le site à la date prévue.

### 3.1. Ce qu’il nous reste à faire

* Générer répertoire qui contient toutes les informations du candidat
* Grille de synthèse des candidatures au format CSV
* Script PHP générant un document XML contenant les liens vers les formations, ces liens iront sur Drupal.
* Changer dynamiquement les ressources par rapport à l'année d'inscription (2014/2015)
* Prévoir le site en Français/Anglais
* Contrôleurs général et spécifiques

### 3.2. Ce que nous sommes en train de faire

* Formulaire de formation
* Formulaire d'inscription
* Routage de l'application
* Générer un pdf rempli d'office pour les commissions pour chaque élève
* Architecture du site
* Compression des fichiers uploadés dans un zip
* Gestion de l'upload des fichiers
* Validation des formulaires en jQuery avec respect de l'ergonomie avec bootstrap.
* Base de données
* Gestion de l’intranet
* Vues TWIG

### 3.3. Ce que nous avons fait

* Use cases
* Workflow
* Développer la classe FileHeader (xml, json, text/plain etc ...)
* Planning Gantt
* Explorateur de fichiers
* Cahier des charges
* Ajout de Twig
* Spécifications fonctionnelles
* Spécifications techniques
* Système de Traduction