



UNIVERSITÉ CLAUDE BERNARD

DOSSIER D'INITIALISATION

MEANTEX

Encadrant

Romuald THION

Directeurs de projet

Emmanuel COQUERY, Romuald THION

Equipe

MALIALIN Audréa

DAUPHIN Maxime

ALTIERI Guillaume

PATURAL Hugo

HARRAKA Mohamed

IZYDORCZYK Maxime

N° de révision	Date	Commentaire
03	15/10/2015	Révisions après commentaire de Romuald Thion et mise en forme LaTeX
02	14/10/2015	Révisions après rendez-vous avec le responsable de projet.
01	11/10/2015	Rédaction du dossier d'initialisation

Dépôt forge : <https://forge.univ-lyon1.fr/hg/p1002412-ti5-meantex>

Serveur de qualification : 192.168.74.199

15 Octobre 2015

Table des matières

1	Introduction	1
1.1	Contexte du projet	1
1.2	Positionnement	1
1.3	Objet	2
2	Résultats attendus	2
2.1	Livrables de gestion de projet	2
2.2	Livrables techniques	2
2.3	Autres livrables	3
3	Méthode et outils	3
3.1	Contraintes	3
3.2	Méthodologie	4
3.3	Outils	4
4	Organisation	4
4.1	Lots de travail	4
4.2	Répartition du travail	5
4.3	Phasage	5

1 Introduction

1.1 Contexte du projet

Actuellement, les sujets concernant les travaux dirigés, les travaux pratiques ou encore les contrôles sont, pour une majorité des enseignants en informatique ou mathématiques, rédigés en LaTeX. Cependant, ces sujets étant destinés à un travail collaboratif (plusieurs enseignants pour une même UE) il est essentiel de pouvoir partager les exercices associés. Toujours en considérant un cadre universitaire, il peut être nécessaire d'une année à une autre d'effectuer des modifications sur un sujet d'exercices : en supprimer un, en ajouter un autre ou tout simplement pour s'adapter au changement de contenu.

A contrario des outils de traitement de texte où il existe de nombreuses plateformes de travail collaboratif (Google Doc, Microsoft Online...) de tels outils pour le LaTeX sont rares et payants (Share-LaTeX, Overleaf). De plus, le problème n'étant pas en soit de rédiger ensemble un document mais de pouvoir facilement générer des sujets d'exercices venant d'une base commune.

C'est dans cette optique qu'a émergé le projet MeanTeX.

1.2 Positionnement

Deux projets (en 2013 et 2014) ont déjà été réalisés dans l'objectif de répondre à cette problématique. Chacun ayant des visions différentes de la solution. Si de bonnes idées émergent de chacun de ces projets, nous partirons ici d'un tableau blanc en essayant cependant de s'inspirer des travaux précédents.

Les idées retenues :

- Idée de panier : l'utilisateur comme il le ferait sur un site de commerce remplit un panier dans lequel on retrouve les exercices sélectionnés.(Groupe 2013)
- Utiliser uniquement du LaTeX et ne pas passer par langage pivot. (Groupe 2014)
- La gestion des métadonnées d'exercice qui peut être implicite ou explicite.(Groupe 2013)
 - Implicite : à partir de l'historique d'utilisation de l'exercice.
 - Explicite : on demande à l'utilisateur de remplir un formulaire.

En parcourant ces anciens projets nous avons pu mettre en évidence les points délicats de MeanTeX :

- Parsing des documents envoyés par les utilisateurs :
 - Du contenu hors balise exercice une solution a été de définir un environnement
 - Les dépendances à d'autres fichiers
 - Des balises qui diffèrent pour un exercice `\beginexercice` ou `/Exercice` etc.
 - Gestion des packages LaTeX.
 - Qui peuvent rentrer en conflits => pour la génération on ne pourra donc pas mettre comme des bourrins tous les packages des exercices.
 - Qui peuvent être absents.
 - Gérer les définitions et redéfinitions de commandes.
- Gestion des fichiers inclus : l'utilisateur va devoir envoyer non seulement son fichier .tex mais aussi toutes les dépendances à ce dernier (hors package). Par exemple les images ou les codes sources de programmes.

1.3 Objet

L'objectif du projet que nous avons à réaliser est de développer une application web. Cette solution aura pour but d'aider les enseignants à non seulement partager les exercices qu'ils auront rédigé en LaTeX mais aussi de leur permettre de créer des sujets de TD, TP et de contrôle à partir du panel d'exercices enregistrés dans la base.

Niveau fonctionnel, l'application devra permettre à l'utilisateur de réaliser les actions suivantes :

- Dépôt et analyse de feuilles d'exercices existantes : division en exercices, extraction des métadonnées
- Recherche d'exercices par des critères de tri et en texte plein pour la composition de nouvelles feuilles d'exercices à partir de ceux existants
- Aperçu des exercices et de la feuille à générer puis génération de la feuille aux formats PDF ou .tex
- Consultation de l'historique des feuilles créées et gestion des variantes d'exercices sur un modèle commun

2 Résultats attendus

2.1 Livrables de gestion de projet

- D1 : Dossier 1 Gestion de projet post-sprint 1 le **26/10/2015**
- D2 : Dossier 2 Gestion de projet post-sprint 2 le **07/13/2015**
- D3 : Dossier 3 Gestion de projet post-sprint 3 le **08/02/2016**

2.2 Livrables techniques

Pré-sprint 1 :

18/10/2015 : Période durant laquelle nous allons fixer toutes les bases nous permettant d'avancer sereinement sur le développement de notre application. L'objectif principal ici est de faire des choix de conception et de modélisation. Les tâches sont les suivantes :

- Choix des technologies donnant lieu à un proof of concept
- Prise en main de LaTeX
- Réalisation de moqups pour le frontend
- Modélisation de la base de données
- Réalisation de cas d'utilisations

Post-sprint 1 :

23/10/2015- Application minimale :il s'agit de livrer le code de l'application et une démonstration sur le serveur de qualification.

Un parsing simple. On se contentera de repérer les balises « exercice » sans essayer de gérer le contenu. On permettra à l'utilisateur de spécifier des métadonnées concernant les exercices repérés (auteurs, UE, difficulté, titre, semestre, type).

La recherche d'exercices se fera en considérant les métadonnées uniquement. L'utilisateur pourra ajouter les exercices souhaités dans un panier. Ces exercices seront représentés juste par leurs métadonnées au client (titre, auteurs, UE,...).

Pour ce qui concerne la génération d'exercices on se contentera d'afficher le titre de l'exercice (affichage du panier) sans gérer d'aperçu puis on enverra le fichier tex à l'utilisateur.

25/10/2015- Bilan du sprint 1 :

Dossiers technique/conception (à mettre à jour à chaque sprint) :

- Maquettes de l'interface,
- Structure des documents Mongo,
- Cas d'utilisation,
- Architecture de l'application.

Post-sprint 2 :

04/12/2015-Application améliorée

Au sprint 2 nous essayerons d'améliorer nos fonctionnalités. Tout d'abord, nous figurerons le parsing en extrayant les questions et leur correction pour chaque exercice. Nous rajouterons le fait de pouvoir voir des historiques de feuille d'exercices (sous forme d'arborescence).

Ensuite, nous essayerons d'avoir un aperçu de la feuille d'exercices générée. Et permettrons à l'utilisateur de générer au choix un fichier PDF ou le fichier tex.

06/12/2015- Bilan du sprint 2

Post-sprint 3 :

05/02/2015- Version finale de l'application L'envoi d'exercices devra gérer les dépendances du documents (images, entêtes, etc.). Pour ce fait il faudra dans un premier temps les détecter et demander à l'utilisateur d'envoyer les fichiers associés. La recherche devra intégrer la recherche de texte plein et dès ce moment nous essayerons de donner des aperçus des exercices trouvés. La génération devra gérer le cas où l'utilisateur ne veut que certaines questions d'un exercice et donc lui permettre d'en sélectionner à sa guise en tâchant bien de retirer la correction concernée.

07/02/2016- Bilan du sprint 3

2.3 Autres livrables

- Dossier de synthèse final
- Support de présentation (entre le 16/02 et le 18/02)
- Documentation de développement du logiciel
- Documentation de déploiement et d'administration du logiciel
- Documentation d'utilisation du logiciel
- Tutoriel d'utilisation du logiciel (vidéo)

3 Méthode et outils

3.1 Contraintes

Contraintes techniques : un des objectifs techniques ici est d'utiliser de nouvelles technologies web et plus particulièrement la stack MEAN. Ce qui impose l'utilisation de MongoDB pour la base de données, Express pour le back end de l'application, Angular pour le front et enfin NodeJS.

Contraintes de délais : avoir une version de production d'ici la fin du temps imparti c'est-à-dire le 08 Février 2016.

Contraintes de coûts : Pour 6 personnes, 558 heures de travail sont prévues au total.

3.2 Méthodologie

Tout au long du développement du projet nous travaillerons selon la méthode Scrum qui sera composée de trois sprints. A la fin de chacune de ces périodes nous tâcherons d'avoir une application fonctionnelle grâce à de l'intégration continue.

3.3 Outils

En plus des éléments de la stack Mean nous utiliserons un certain nombre d'outils.

Choix techniques

Côté client :

- Framework Bootstrap pour la mise en page et le design.
- Bower pour la gestion des packages côté client RequireJS

Côté serveur :

- RequireJS
- Child Process pour l'exécution de lignes de commandes.
- Et d'autres modules NPM qui évolueront au cours du temps.

Gestion de projet

Pour réaliser le projet nous utiliserons plusieurs outils :

- Slack : outil de collaboration pour des discussions en temps réel
- Trello : permet de définir les tâches et de suivre leur avancement
- Forge : gestionnaire de versions de l'Université
- Google Drive : pour la rédaction des documents demandés.

4 Organisation

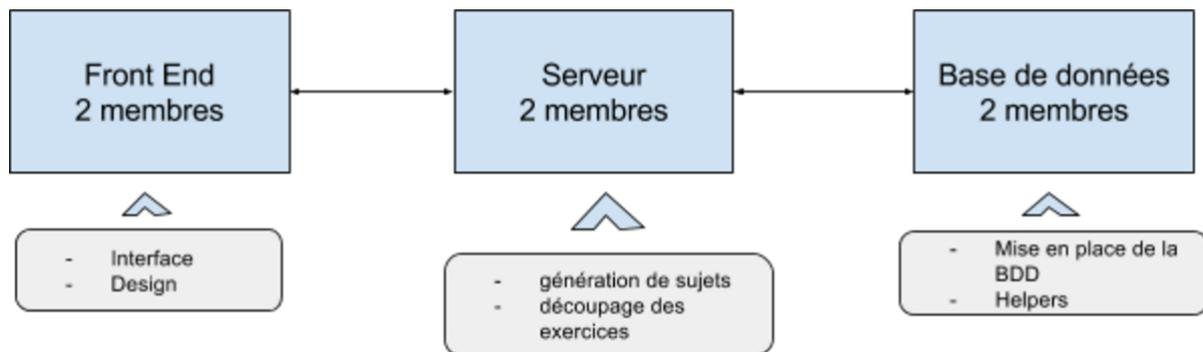
4.1 Lots de travail

- Mise en place de la base de données :
 - Configuration correct du module MongoDB
 - Mise en place des helpers
- Développement de la partie backend
 - Dépôt
 - Recherche
 - Génération aperçu
 - Génération sujet (+ enregistrement)
- Mise en place du frontend :
 - Dépôt document avec métadonnées
 - Interface de recherche
 - Aperçus
 - Génération de sujet (drag and drop, données supplémentaires)

- Documentation
 - Du code (penser qu'il y aura des gens qui vont reprendre le projet)
 - Les documents à rédiger (pour les utilisateurs et les responsables de projets)
- Tests
 - Tests unitaires
 - Tests fonctionnels
 - Déploiement sur VM

4.2 Répartition du travail

Nous avons décidé de partager la charge de travail de la manière suivante :



Au cours du développement les membres chargés de la base de données migreront vers la partie serveur.

4.3 Phasage

La répartition des tâches au niveau du temps se fera comme il suit :

MEANTEX	TIS - MEANTEX 2015																							
	RUSH 1							RUSH 2							RUSH 3									
Délivrables	DJ 1	DJ 2	DJ 3	DJ 4	DJ 5	DJ 6	DJ 7	DJ 1	DJ 2	DJ 3	DJ 4	DJ 5	DJ 6	DJ 7	DJ 1	DJ 2	DJ 3	DJ 4	DJ 5	DJ 5	DJ 6	DJ 7	DJ 8	
Mise en place de la BDD																								
Configuration module Mongo DB	█																							
Mise en place des schémas		█	█																					
Helpers	█																							
Back-end																								
Dépôt de document	█	█	█	█				█	█	█	█				█	█	█	█	█	█	█	█	█	
Recherche	█	█	█	█	█			█	█	█	█				█	█	█	█	█	█	█	█	█	
Génération de sujet	█	█	█	█	█			█	█	█	█				█	█	█	█	█	█	█	█	█	
Front-end																								
Dépôt document + méta-données	█	█	█	█											█	█	█	█	█	█	█	█	█	
Interface recherche			█	█	█	█	█																	
Génération sujet			█	█	█	█	█																	
Aperçu				█	█	█	█	█	█	█	█				█	█	█	█	█	█	█	█	█	
Page historique								█	█	█	█				█	█	█	█	█	█	█	█	█	
Annotations																								
DJ 1/2 journée																								