

PRIM

Application web de gestion de l'UE M1-PRIM

SAVEIL NUEL

ROBIN KEMPF

DANG NGUYEN

ABDOULAYE KEITA

THOMAS MOREL (CHEF DE PROJET)

19/10/2014

Dossier post-sprint 1

Ce document résume le travail effectué durant le sprint 1.

Encadrants

- Marc PLANTEVIT
- Emmanuel COQUERY

Directeurs de projet

- Romuald THION
- Emmanuel COQUERY

TABLEAU DES REVISIONS DU DOCUMENT

01	Robin KEMPF Thomas MOREL	23/10/14	Première version
Révision	Auteur(s)	Date	Modifications

Table des matières

TABLEAU DES REVISIONS DU DOCUMENT	1
DECOMPOSITION DES TACHES.....	3
EVALUATION DE LA CHARGE DES TACHES ET REPARTITION.....	3
LIVRABLES TECHNIQUES.....	4
ETAT D'AVANCEMENT DES TACHES.....	4
TRAVAIL EFFECTUE.....	5
OBJECTIF DE LA PROCHAINE ITERATION	7

DECOMPOSITION DES TACHES

L'objectif de ce premier sprint est de mettre en place une première version de l'application qui devra permettre à l'utilisateur de se connecter à celle-ci. Les enseignants peuvent enregistrer des sujets et les étudiants peuvent les visualiser. Pour parvenir à cet objectif, nous devons effectuer les tâches suivantes :

- Mise en place du projet
- Mise en place de la base de données
- Faire l'interface du client en SPA
- Mise en place du CAS/LDAP pour pouvoir s'authentifier et rediriger l'utilisateur vers la bonne interface suivant son rôle
- Ecrire les scripts PHP côté serveur pour répondre aux requêtes du client en mettant en place une API REST
- Ecrire les scripts JS pour requêter le serveur afin de récupérer les données dans la base de données
- Rédiger les documents post-sprint
- Gestion de projet

EVALUATION DE LA CHARGE DES TACHES ET REPARTITION

Dans le tableau suivant, nous avons les différentes tâches avec le nombre de journées passé au total sur ces tâches, ainsi que les personnes affectés dessus.

Tâches	Evaluation de la charge	Personnes Affectées
Mise en place du projet	½ journée	Dang
Formation Symfony	½ journée	Dang, Robin
Formation AngularJS	½ journée	Saveil
Mise en place de la BD	½ journée	Robin
Interface étudiant	½ journée	Saveil
Interface enseignant	½ journée	Saveil
API REST PHP	1,5 journée	Dang, Robin
Script JS pour requêter le serveur	½ journée	Saveil
Mise en place du CAS	3 journées	Dang, Robin, Thomas
Récupérer les données du LDAP	1 journée	Saveil
Dossier Post-sprint 1	1 journée	Robin
Gestion de projet	3,5 journées	Thomas
Document de conception	½ journée	Thomas, Robin

LIVRABLES TECHNIQUES

À la fin de ce sprint, nous devons rendre deux livrables techniques :

- L'application minimale permettant les fonctionnalités décrites plus haut que l'on présentera à nos encadrants à la prochaine réunion
- Document de retour sur les technologies utilisées jusqu'à présent

ETAT D'AVANCEMENT DES TACHES

Le tableau récapitule l'avancement des tâches après ce sprint, certaines tâches ne sont pas à 100% car elles sont amenées à évoluer encore dans les prochaines itérations.

Tâches	Prévisions	Etat d'avancement
Mise en place du projet	100%	100%
Mise en place de la BD	100%	100%
Interface étudiant	50%	60%
Interface enseignant	50%	50%
API REST PHP	50%	50%
Récupérer les données du LDAP	50%	25%
Mise en place du CAS	100%	20%

À la fin de ce sprint, nous avons une application minimale qui permet d'utiliser les interfaces étudiant et enseignant. Avec l'interface étudiant, on peut visualiser les différents sujets postés par les enseignants. Et avec l'interface enseignant, on peut rajouter des sujets en spécifiant les différents champs.

Au niveau du CAS, celui-ci fonctionne hors projet sans vérification du certificat de l'université (environnement de développement). L'intégration du CAS au projet aura nécessité beaucoup plus de temps que prévu (3 jours), sans pour autant être fonctionnel en fin de sprint. Le problème venant de l'impossibilité d'intégrer un « Bundle » Symfony existant qui utilise SSLv3. En effet, une faille découverte récemment sur SSLv3 (faille Poodle) a contraint l'université à bloquer ce protocole, ce qui nous a fait perdre du temps sur le débogage qui s'est avéré vain.

Pour LDAP, le problème est similaire à celui du CAS. En effet, nous sommes passées par une implémentation hors projet. Cependant sa réalisation nous a posé moins de problèmes.

TRAVAIL EFFECTUE

Pour mettre en place la base de données, nous avons utilisé WampServer pour créer les tables, puis exporter le script SQL.

Au niveau de la mise en place de l'API REST PHP, nous avons créé un Bundle « RESTBundle » qui contient des contrôleurs et des entités.

Ces entités représentent les différentes tables de la base de données. Elles contiennent des annotations ORM pour permettre de faire le lien avec la base de données. Pour générer ces entités et ces annotations, nous avons utilisé la génération par ligne de commande et la librairie Doctrine.

Les contrôleurs, quant à eux contiennent les différents traitements pour interroger la base de données avant de pouvoir récupérer les données au format JSON ou d'insérer des données dans la base de données.

Reprise des maquettes pour l'intégration avec AngularJS, dont la création de contrôleurs, routes et vues. Dans le contrôleur, les requêtes REST vers le serveur incluent la récupération de la liste des sujets, pour leur affichage côté étudiant (cf. Figure 1) et enseignant. De plus, l'interface enseignant (cf. Figure 2) requête le serveur afin d'y ajouter un nouveau sujet (cf. Figure 3). Pour le moment, aucune vérification n'a été effectuée sur les données du formulaire.

Pour intégrer les pages HTML dans le projet, nous sommes en train de mettre en place un autre Bundle qui s'occupera d'afficher les différentes interfaces grâce à des moteur template Twig.

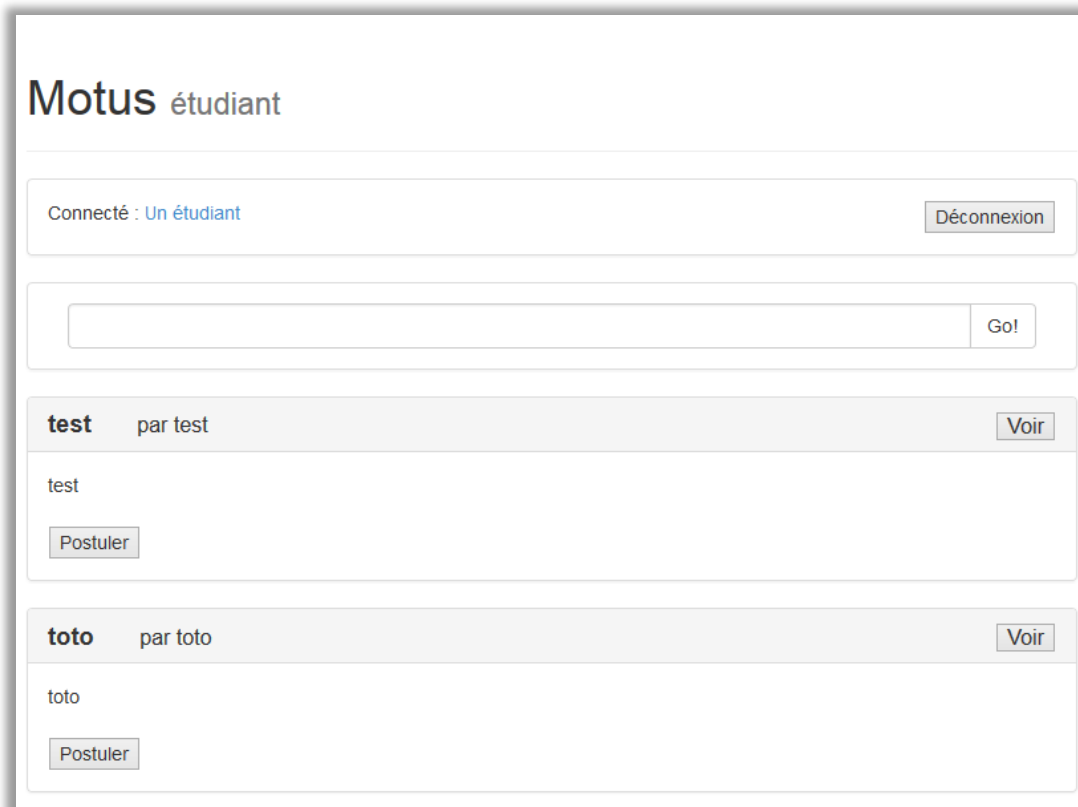


FIGURE 1 : LISTE DES SUJETS SUR L'INTERFACE ETUDIANT

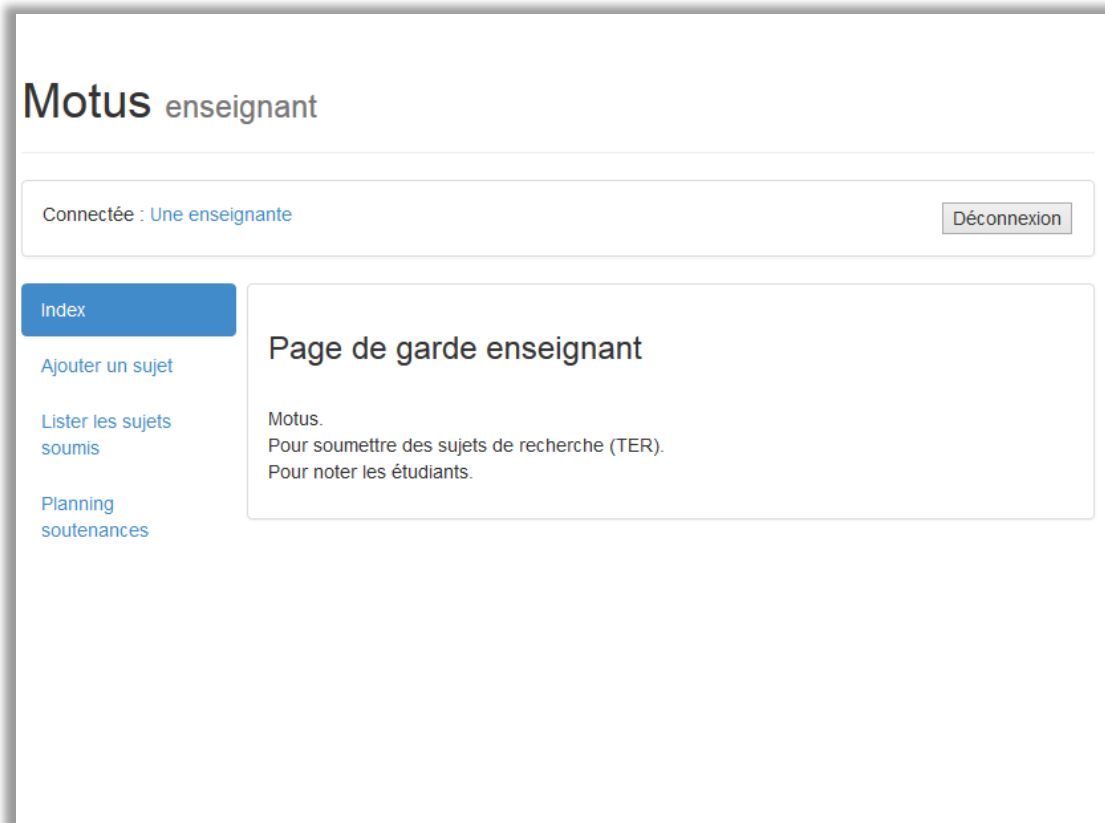


FIGURE 2 : PAGE DE GARDE ENSEIGNANT

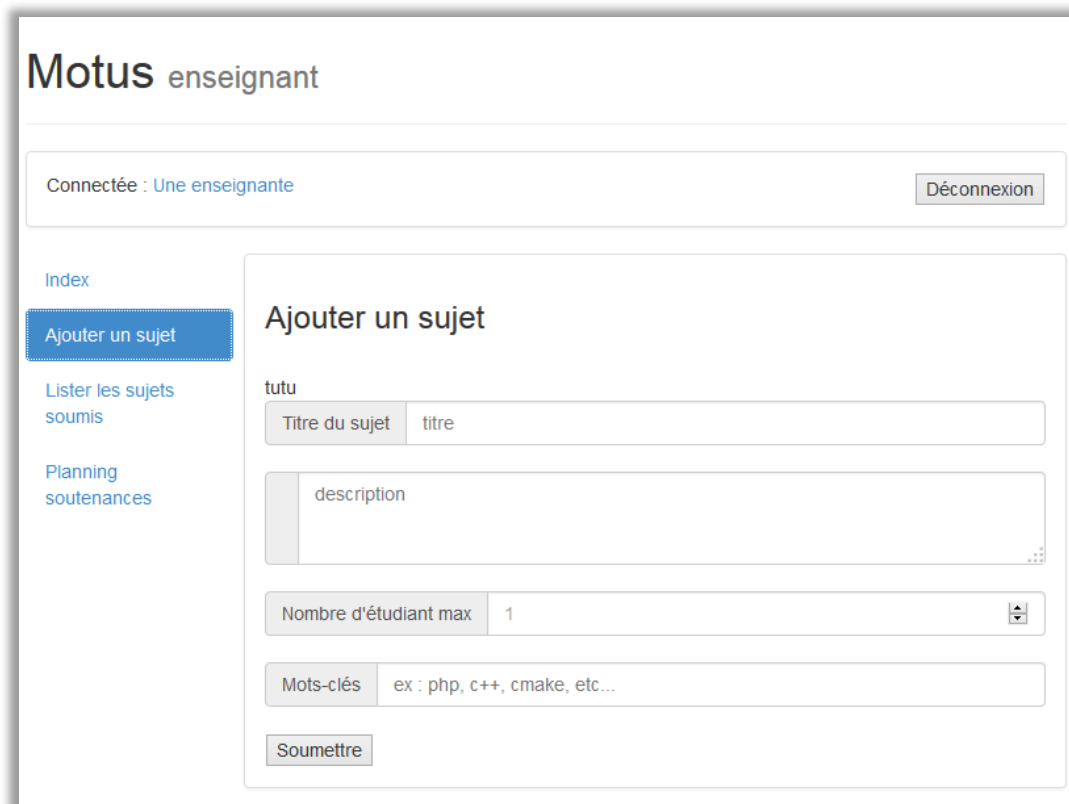


FIGURE 3 : AJOUT D'UN SUJET COTE ENSEIGNANT

OBJECTIF DE LA PROCHAINE ITERATION

À la prochaine itération, nous devons faire en sorte d'avancer sur les tâches suivantes :

- Intégrer le CAS au projet (Prioritaire)
- Continuer LDAP (Important)
- Finir l'API REST PHP
- Finir les scripts JS pour requêter le serveur
- Avancer sur les interfaces client