

Personnes présentes :

Jean-Sébastien LERAT	Fabrice Bellarmin NOLACK FOTE
Virginie VANDEN DRIES	Amine ROUKH

1 Introduction

Notre projet est un sous-projet (projet d'amorçage) d'un projet de 6 ans impliquant divers partenaires wallons. Notre "client" est l'Association Wallonne de l'Élevage (AWE), représenté par François RIGA, et les responsables UMONS sont Saïd MAHMOUDI et Sidi MAHMOUDI. Les personnes en charge du développement sont Fabrice Bellarmin NOLACK FOTE (doctorant) et Amine ROUKH (post-doctorant).

À la suite de notre discussion, il en ressort que l'AWE souhaiterait obtenir une plateforme qui regroupe les différentes applications (proposées par différentes entités) qu'ils utilisent. La plateforme devrait permettre de gérer automatiquement toutes les données des agriculteurs (en respectant le RGPD1) afin d'automatiser certaines tâches administratives, de proposer un suivi des données personnelles via une présentation des données (cartographie, graphique, tableau, ...), de proposer une visualisation des données à un conseiller (comptable, . . .) ou bien de fournir des données à des partenaires de recherche.

Nous avons délimité le cadre du projet en discutant avec les différents interlocuteurs. Quelques remarques préliminaires sont nécessaires :

- Nous n'aurons pas d'échantillons de données d'ici la fin du projet.
- Nous n'aurons aucune application de l'AWE disponible d'ici la fin du projet.
- M. François RIGA se concentre principalement que sur son projet pour 6 ans. Il souhaite intervenir lors des présentations intermédiaires avec un ou deux autres collègues (informaticiens) de l'AWE. Lors de la dernière échéance, nous lui présenterons une interface conviviale afin de lui montrer le potentiel du travail réalisé.
- Notre travail est principalement une proof of concept.
- Nous pouvons avoir accès à un cluster (groupement de ressources informatiques) de calculs.
- Nous pouvons avoir accès à un service de cloud computing payant.
- Nous pouvons avoir du matériel mais pas avant la fin du projet.
- Les données que nous exploiterons seront structurées.

2 Cadre de travail

En tant que proof of concept, nous devons réaliser une interface Web vers différentes sources de données.

Ces données peuvent être hébergées chez des agriculteurs, chez des entreprises qui fournissent des services à l'AWE ou à l'AWE elle-même. Le travail consiste principalement à

concevoir une API d'interfaçage pour l'interrogation et l'échange de données. Il faudra toutefois présenter une interface Web simple qui permet d'authentifier un utilisateur et de présenter les données auxquelles il a accès sous forme de tableau ou de graphique.

Le portail Web doit pouvoir autoriser 6 rôles d'utilisateur (un utilisateur peut avoir plusieurs rôles) :

- **Gestionnaire** peut créer des comptes (ou inviter quelqu'un à créer un compte), peut fournir des droits à une entité (attention aux droits sur les données personnelles, seul l'agriculteur gère ses droits).
- **Agriculteur** peut importer des données (excel, access, fichier texte (style csv), ...) et autoriser des personnes à accéder à ses données. Il peut également supprimer toutes ses données personnelles ou bien les modifier. Attention certaines données peuvent être envoyées à une autre application et ne plus être modifiable sur cette autre application.
- **Conseiller** peut accéder à des données générales et des données propres à un ou plusieurs agriculteurs (si ceux-ci le permettent).
- **Scientifique** peut accéder à des données thématiques générales (définies par le gestionnaire) et accéder à des données privées d'agriculteurs (si ceux-ci le permettent) à la suite de la demande d'un gestionnaire.
- **Citoyen** peut visualiser les données générales (exemple : quelle est la quantité de méthane produite en région wallonne à la suite des pets des vaches).
- **Développeur** peut avoir un accès particulier (publier une nouvelle application), obtenir un échantillon de données de test, environnement de test, ...

Les données générales doivent également contenir des informations (stats) à propos de l'utilisation du site Web (ex : nombre de visiteurs moyen, nombre d'agriculteurs inscrits, ...).

M. François RIGA est en train de créer un schéma de base de données d'exemple avec M. Fabrice Bellarmin NOLACK FOTE et M. Amine ROUKH qu'ils devraient nous fournir assez rapidement.

3 Données

Les données peuvent provenir de sources hétérogènes : Fichier Excel, Access, csv, ...

Base de données relationnelle Oracle, Postgresql, MySQL, SQL Server, ...

Echange de données SOAP3, REST4, ...

Il faut donc permettre d'ajouter facilement de nouvelles sources de données et d'y accéder facilement.

4 Authentification

N'ayant pas de service d'authentification disponible, nous devons mettre en place notre propre système (table utilisateurs) d'authentification mais il faut également permettre le mode d'authentification ou d'en ajouter d'autres. M. Amine ROUKH souhaite que nous installions un serveur LDAP5 mais nous lui avons fait part des limites de notre matériel. Il a accepté nos arguments tant qu'il est possible d'intégrer facilement un serveur LDAP.

5 Présentation des données

Les données issues des différentes sources et disponibles doivent être présentées sous différents formats d'affichage, deux formats dans le cadre de notre projet :

Tableau Affichage du contenu des différentes tables. Attention, nous n'avons pas forcément accès à toutes les tables (exemple : service REST).

Génération simple de graphique. Il doit être possible de choisir les données à afficher (différentes sources), de choisir le type de graphique et de produire le graphique correspondant. Cela doit permettre par la suite d'afficher un graphique spécifique.

6 Résumé

En résumé, il faut que nous créions une structure de portail Web afin d'accéder à un data warehouse dans le domaine de l'agriculture, fournissant des données similaires à de l'OpenData mais il y a des restrictions d'accès.

La suite consiste à choisir les technologies, à mettre en place une infrastructure de tests, créer des comptes github, créer la modélisation de la base de données, créer la modélisation UML de l'application, créer les histoires scrum.

Procès verbal - réunion 2 – 21/02/2019

Membres du groupe présents :

Aurélie COOLS	François ROBERTS
Inès GAUTHIER	Michael ROMBAUX
Emilie GENIN	Virginie VANDEN DRIES
Jean-Sébastien LERAT	

1. Introduction

La réunion a pour but de :

- Discuter de la précédente réunion du 12/02/2019 à laquelle tout le monde ne pouvait être présent ;

- Fixer les technologies à utiliser ;
- Discuter de la répartition des tâches.

Un partage de connaissance sur la mise en place des différentes technologies est aussi abordé.

2. Discussion sur la réunion du 12/02/2019

Quelques questions subsistent sur le cadre de travail, le type de données à traiter et les moyens que nous devrons mettre en œuvre pour y parvenir.

3. Technologies

Jean-Sébastien propose et présente une série de technologies aux membres.

Pour la gestion de version des codes sources ainsi que les documents relatifs au projet, github (<https://github.com/>) est proposé et accepté.

L'outil d'intégration continue est Jenkins (<https://jenkins.io/>).

Pour les tests en local (serveur web avec php & mysql), XAMPP, WAMP ou MAMP sera utilisé au choix (<https://www.apachefriends.org/fr/index.html>).

Concernant l'IDE, différentes possibilités sont abordées. Jean-Sébastien utilise habituellement kate (<https://kate-editor.org/>) et les commandes git dans une console. Mais, pour les membres qui n'ont pas l'habitude, Eclipse pour PHP (<https://www.eclipse.org/pdt/>) avec un plugin GIT (<https://www.eclipse.org/egit/download/>) semble plus simple (beaucoup de documentation disponible sur internet).

Les membres sont d'accord pour l'utilisation d'Eclipse et d'un plugin GIT.

Pour la business infrastructure, il faut encore choisir entre 2 logiciels libres Zeppelin et Jupyter. Jupyter fournit une solution aboutie mais limitée. Zeppelin est soutenu par la fondation Apache, combinaison de langages de programmation, est multi-utilisateur et fournit de la sécurité. Il est donc préféré car plus pérenne.

Pour le framework, la discussion porte sur Laravel, CodeIgniter et Symfony. Laravel est assez compliqué à prendre en main. Symfony est plus lent et lourd et présente moins de fonctionnalités. CodeIgniter présente pratiquement les mêmes fonctionnalités que Laravel mais est plus léger, simple et rapide. Ce dernier est donc choisi.

Enfin, pour la base de données centrale, le choix est partagé entre MySQL et MariaDB. C'est ce dernier SGBD qui est finalement choisi.

Pour le suivi du projet, un diagramme de Gantt sera disponible où les tâches seront réparties en histoires avec des dates butoirs.

4. Répartition des tâches

La méthode xp promeut le pair programming. C'est ce qui sera utilisé dans le cadre de ce projet.

Pour la répartition des tâches impliquant le développement :

- Virginie et Emilie se proposent pour travailler dans un premier temps sur la base de données. Elles s'occuperont ensuite du frontend du site ;
- Inès et Aurélie se proposent pour la réalisation des tests ;
- Jean-Sébastien se propose pour mettre en place de l'environnement de développement (matériel, configuration,...) ;
- Enfin, Michael et François s'occuperont du backend.

Pour le premier point de contrôle, sont à faire :

- La planification des tâches et du PID ;
- La réalisation des maquettes ;
- La réalisation d'un premier MCD ;
- La réalisation d'un diagramme de cas d'utilisations.

Procès verbal - 1^{er} point de contrôle – 28/02/2019

Membres du groupe présents :

Aurélie COOLS	François ROBERTS
---------------	------------------

Inès GAUTHIER	Michael ROMBAUX
Emilie GENIN	Virginie VANDEN DRIES
Jean-Sébastien LERAT	

1. Introduction

Lors du premier point de contrôle, différents points ont été abordés :

- Retards ;
- Technologies choisies ;
- Modélisation ;
- Répartition des tâches ;
- Problèmes techniques.

2. Retards

Des retards initiaux ont été abordés, liés à la difficulté de rencontre avec le client et de mise en place des ressources matérielles nécessaires.

3. Technologies

L'ensemble des technologies a été approuvé mais l'utilisation de Trello a été recommandée à la place d'EasyBacklog initialement prévue.

4. Modélisation

Le MCD présenté est validé, tout comme la maquette. Quelques modifications sont proposées pour le diagramme de cas d'utilisation. L'environnement en place est également validé.

5. Répartition des tâches

À la suite du premier point de contrôle, la répartition des tâches a été revue pour la partie « modèle » du Backend. Cette tâche sera réalisée par Emilie et Virginie dès l'implémentation de la base de données finie. Le MCD étant validé, une ébauche de diagramme de classes du modèle est discutée.

6. Problèmes techniques

Vu les problèmes liés à l'apprentissage d'un certain nombre de nouvelles technologies, une intégration de git à Eclipse est réalisée. Les opérations de base pour la mise en ligne et la récupération du code sont montrées. De même, l'utilisation du framework codeigniter est introduite. La documentation disponible sur codeigniter est recommandée pour faciliter

l'apprentissage. De même, les difficultés de mise en place et de paramétrage de Jenkins sont abordées.

Procès verbal - 2è point de contrôle – 28/03/2019

Personnes présentes :

Aurélie COOLS	François ROBERTS
Inès GAUTHIER	Michael ROMBAUX
Emilie GENIN	Virginie VANDEN DRIES
Jean-Sébastien LERAT	

1. Introduction

Différents points ont été abordés :

- Modélisation ;
- Réalisation des tests unitaires ;
- Technologies ;
- Gestion des retards ;
- Demande du client ;
- Problèmes techniques.

De plus à la fin du point de contrôle, il est conseillé de répartir le temps de paroles plus équitablement entre les membres du projet pour les futures présentations.

2. Modélisation

Un premier diagramme de classes du backend est proposé et validé.

3. Réalisation des tests unitaires

La mise en place efficace de Jenkins et la réalisation des premiers tests unitaires a été démontrée.

4. Technologies

Pour le Front-End, les moyens de mise en œuvre n'ont toujours pas été fixés et sont discutés au cours de la présentation. Les possibilités étudiées sont Bootstrap et les flexbox. Les

inconvénients de Bootstrap font que le groupe est favorable à l'utilisation des flexbox. Après discussion lors du point de contrôle, ce sont ces derniers qui sont choisis.

5. Gestion des retards

Des retards subsistent mais sont partiellement rattrapés. La vélocité du groupe augmente vu l'augmentation des connaissances dans les technologies utilisées.

6. Demande du client

Lors de ce point de contrôle, le client a demandé d'ajouter aux fonctionnalités un système de recherche d'utilisateurs. Ce point est à ajouter au planning.

7. Problèmes techniques

Certaines questions subsistent quant aux moyens à mettre en œuvre pour automatiser les tests unitaires dans Jenkins. De même, les moyens à mettre en œuvre pour la réalisation des tests fonctionnels sont à résoudre. Aurélie penche pour l'utilisation de Sélénium.

Procès verbal - 3è point de contrôle – 2/05/2019

Personnes présentes :

Aurélie COOLS	François ROBBERTS
Inès GAUTHIER	Michael ROMBAUX
Emilie GENIN	Virginie VANDEN DRIES
Jean-Sébastien LERAT	

1. Introduction

Les points abordés suite à ce troisième point de contrôle sont :

- Gestion des retards ;
- Modélisation ;
- Tests ;
- Réalisation du Front-End ;
- Réalisation de la documentation ;
- Problèmes techniques.

2. Gestion des retards

Tous les retards ont été rattrapés.

3. Modélisation

Les classes du modèle et les principales classes du contrôleur ont été présentées et décrites. Elles ont été validées.

4. Tests

Des graphiques de tests unitaires ont été présentés. Pour les tests fonctionnels, ils ont été implémentés sur Sélénium mais n'ont pu être présentés que dans les grandes lignes.

5. Réalisation du Front-End

Le Front-End est à peaufiner pour la remise du projet. Il est conseillé au groupe l'utilisation de maximum 3 couleurs avec lesquelles jongler pour la mise en page du site. Aurélie se propose pour la mise en page du site. De plus, l'application ne possédant pas encore de logo, elle se propose pour la confection d'un logo.

6. Réalisation de la documentation

La documentation du code doit être fournie pour la remise du projet. Virginie utilise Doxygen pour générer la documentation. Jean-Sébastien s'occupe du manuel d'installation après avoir fini la partie « REST ».

7. Problèmes techniques

Pour la réalisation des tests unitaires sur la classe DataSourceModel, Emilie et Ines discutent des cas d'échecs aux tests unitaires et travaillent à les résoudre.