

Rapport D'avancement

25/02/2019 - 01/05/2019

Par : Ayoub HTIOUI

Encadrant interne : Ali CHOUKRI

Encadrant externe : Mohamed IDRISSE

Problématique:

L'équipe de développement a un problème que chaque fois qu'on veut développer pour un client, on développe le projet à nouveau sans pouvoir réutiliser des codes déjà utilisés dans d'autres projets.

⇒ Perte de temps et consommation de beaucoup de ressources .

Solution:

Développer une architecture qui va faciliter la réutilisation des modules fonctionnels souvent demandés par les clients: la gestion des fichiers, l'authentification, internationalisation des langues ...

Idée générale du projet:

Mon projet de fin d'étude s'inscrit dans le cadre de développement d'un IDE de boilerplate, efficace, rapide mais surtout simple, qui a pour but de faciliter la réalisation des applications web, en offrant un boilerplate avec des modules fonctionnels-généralement très utilisés- prêts à être intégrés dans le projet, pour éviter de démarrer le développement "from scratch". Ces modules vont être représentés avec des fronts-end simples et flexibles.

Outils de développement utilisés: NodeJS, Angular, MongoDB, Postman, AWS

Méthodologie de travail SCRUM:

Contexte: Le projet se réalise dans une équipe, composée de trois stagiaires et un designer. Chaque stagiaire s'occupe du développement d'un module fonctionnel, en se basant sur une maquette fournie par le designer .

SCRUM est une méthode agile de gestion de projets, utilisée pour organiser les équipes pour produire une grande valeur métier avec une meilleure qualité. Elle découpe le projet en itérations nommées Sprints, un sprint peut avoir une durée de 2 à 4 semaines.

Scrum définit trois rôles :

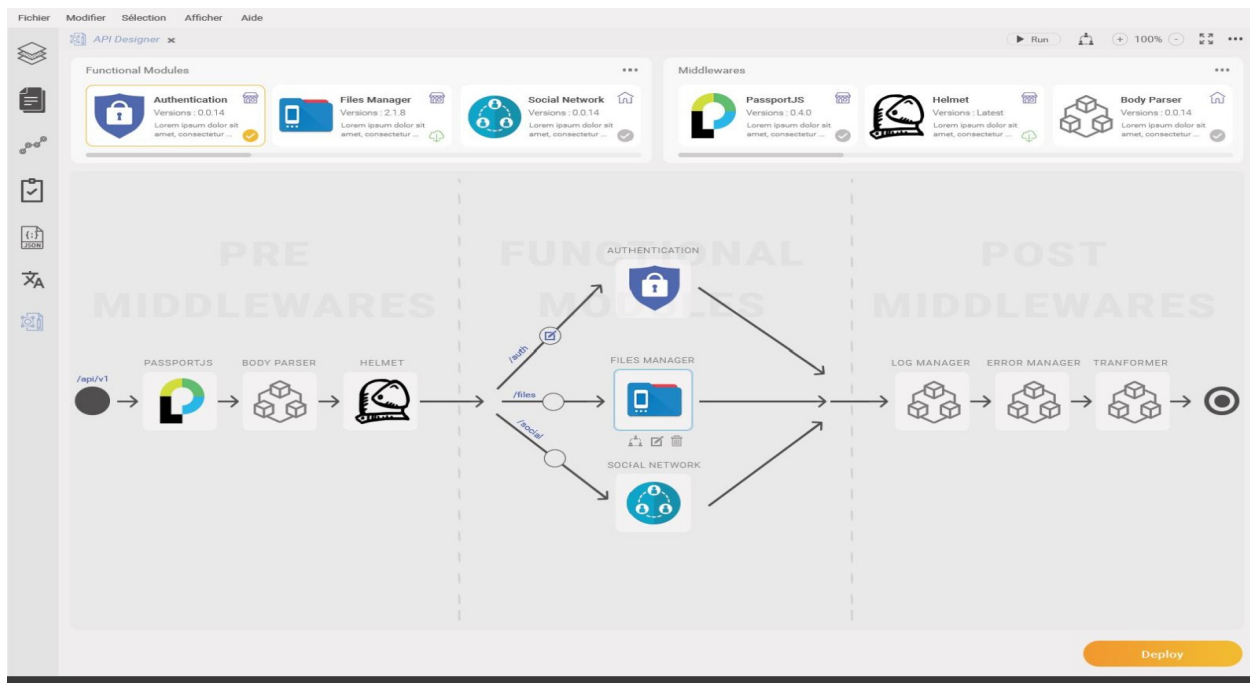
Le **Product Owner** qui porte la vision du produit à réaliser et qui travaille en interaction avec l'équipe de développement. Dans notre cas, le Product Owner est **Mr Mohamed IDRISSE**.

Le **Scrum Master** qui doit maîtriser Scrum et s'assurer que ce dernier est correctement appliqué. Il a donc un rôle de coach à la fois auprès du Product Owner et auprès de l'équipe de développement. Généralement le candidat tout trouvé au rôle de Scrum Master est le chef de projet

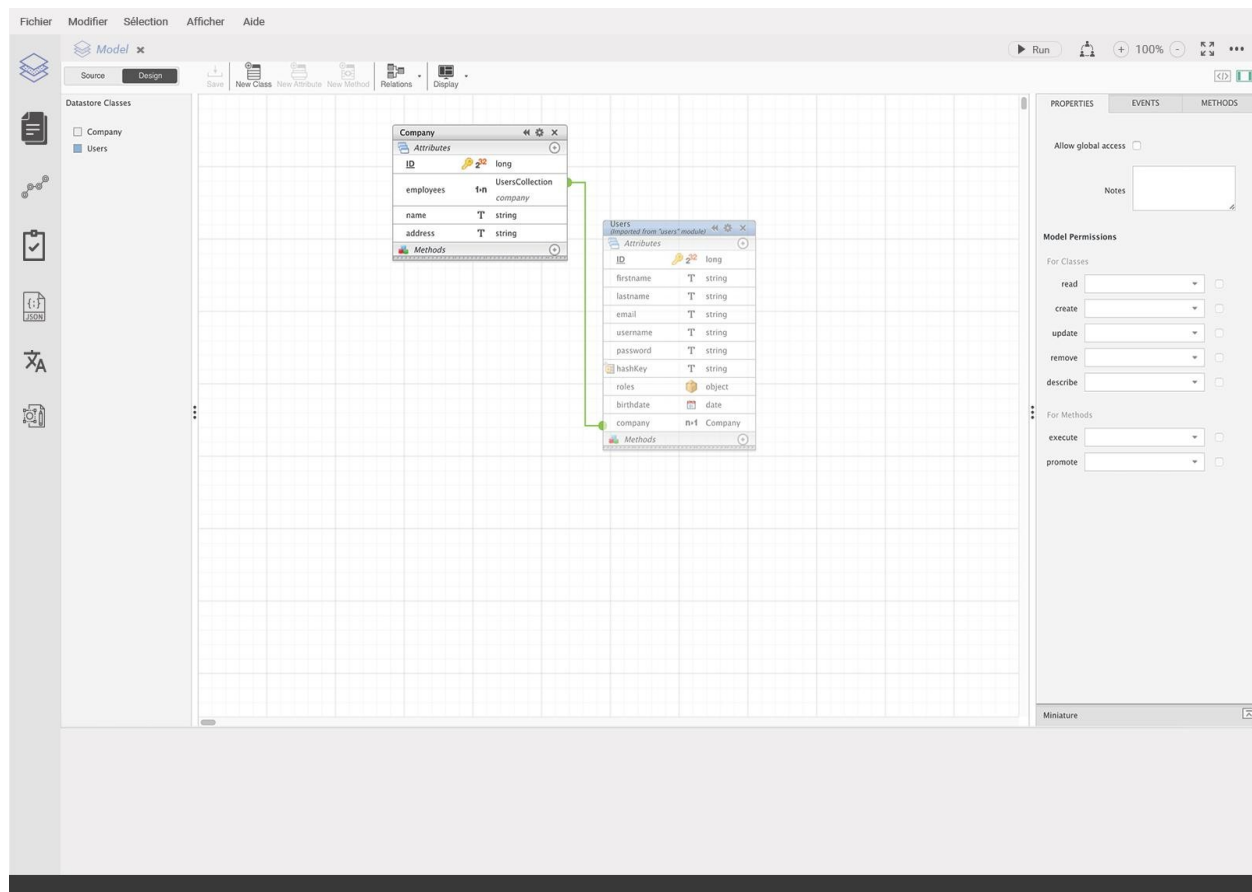
Scrum Team qui est chargée de transformer les besoins exprimés par le client en fonctionnalités utilisables. Elle est constituée d'un ensemble de développeurs qui assurent le développement de différentes fonctionnalités.

Les différentes parties du IDE:

L'IDE est constitué de 6 différentes parties: L'éditeur de code, API-designer, Routes-designer, i18n-editor, model-designer, manifest-editor



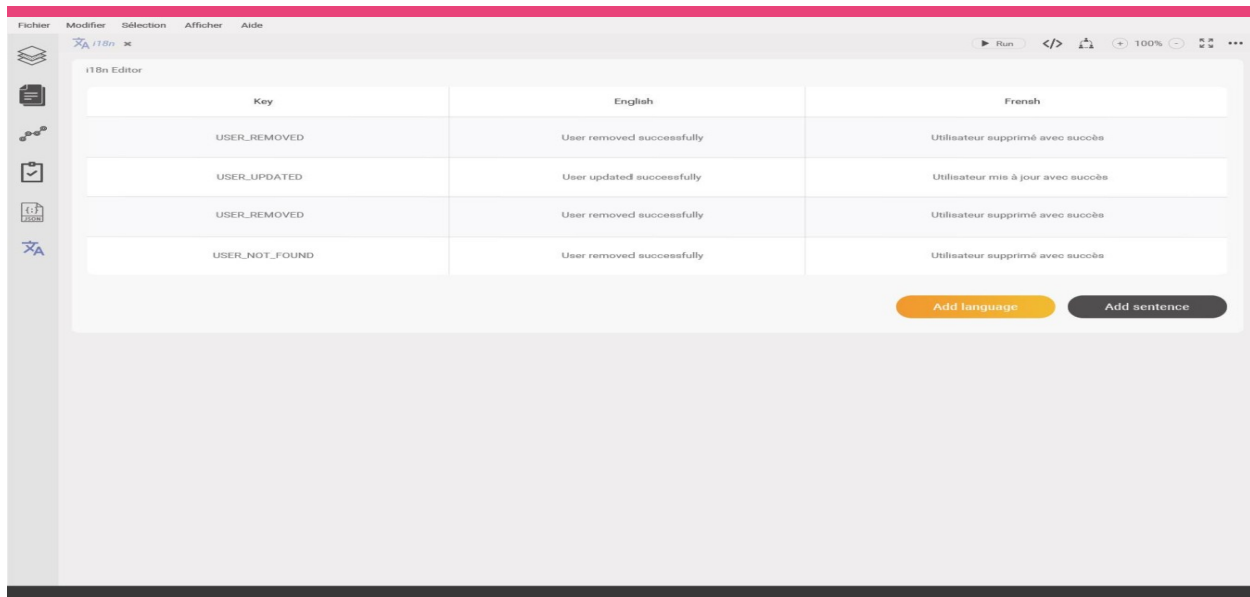
maquette 1: API-Designer



maquette 2: Model-designer

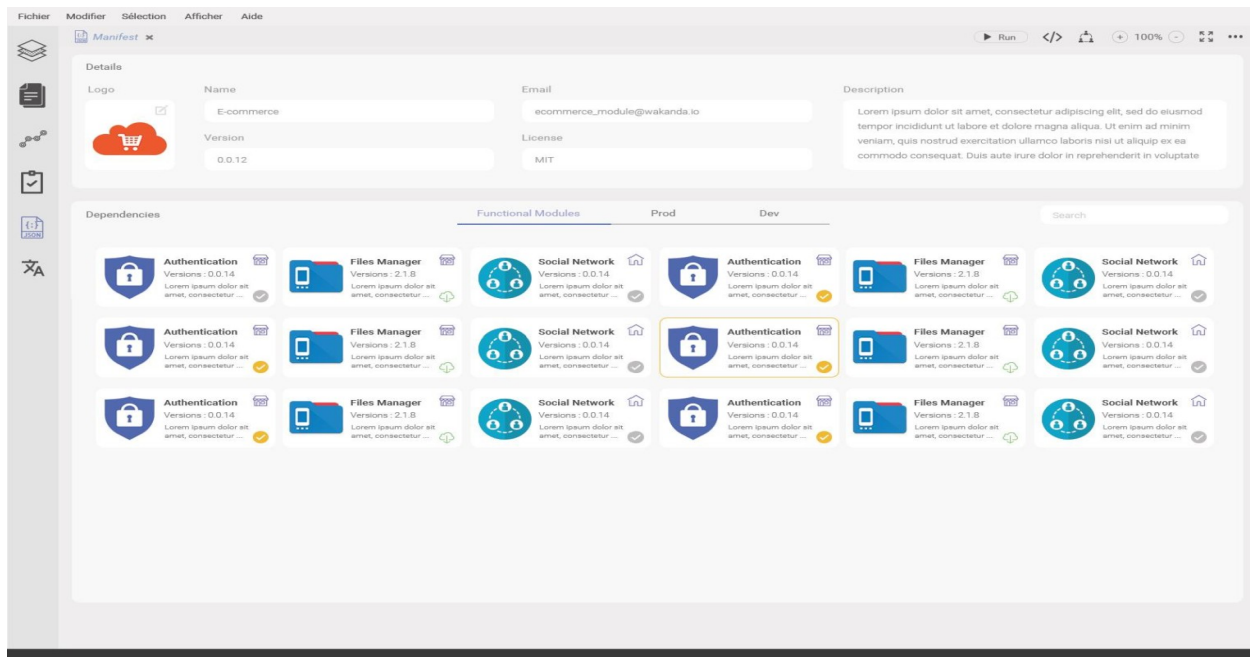
Ce module a pour but de faciliter la modélisation de la base de données à travers la définition des schémas de validation des données, indépendamment de la base de données, pour qu'il soit valable pour des utilisateurs de sql et non-sql à la fois .

À travers l'interface graphique le développeur peut facilement (par des clicks) ajouter des attributs, des méthodes aux objets à stocker et gérer les événements sur une table (presave/ postsave / postremove / preremove ...).



Maquette 3: Module internationalisation des langues

Le module i18n consiste à adapter le code afin de permettre la traduction littérale par la suite. Il fait appel à des API de traduction externes comme google translate.



maquette 4: manifest module

Ce module permet d'organiser les métadonnées relatives à chaque module fonctionnel (le nom, la version, la description, logo, license key ...) et ces dépendances (les modules fonctionnels et les bibliothèques desquels dépend ce module) .

Config-editor: ma tâche actuelle consiste à réaliser une interface dynamique de configuration, qui permet de visualiser/modifier/valider les entrées de configuration de chaque module fonctionnel.

Les outils à utiliser: Angular & Semantic UI

Formation Glt & Git-Flow:

Vu que le projet est collaboratif, il nous était indispensable d'opter pour Git-Flow comme technique qui va assurer l'efficacité de la collaboration, et la bonne gestion des différentes versions de l'application.

J'ai préparé, durant la deuxième semaine, une formation qui porte sur le workflow Git-Flow que j'ai faite pour mes collègues stagiaires le lundi 22 avril 2019.

Programme de la formation:

Installation et configuration

- ☐ Historique de Git
- ☐ Installation sous linux, mac ou Windows Les options de configuration

Principes de git

- ☐ Git : un outil décentralisé
- ☐ Gestion par branches et commit
- ☐ Travail en local
- ☐ Gestion des branches : Branches locales et remotes
- ☐ Fusion de deux branches et résolution des conflits

Erreurs, imprévus :

- ☐ Résoudre un merge compliqué ou conflit
- ☐ Les commandes Revert et Reset Mettre de côté avec Git Stash

Git en équipe

- ☐ Techniques avancées pour un commit lisible Git Flow : simple, robuste et efficace
- ☐ pull/push/fetch requests

Modèle de branchement Git FLOW

- ☐ Les bonnes pratiques pour mieux organiser les projets à plusieurs
- ☐ Séparer au maximum le travail sous plusieurs branches chacune dans un but différent