

Rémi Gattaz  
Thibaut Nouguier

Germain Lecorps  
Régis Ramel

# Projet 2016-2017

## Présentation finale

### Gestionnaire de packages



# Sommaire

- ▶ I/ Rappel de la présentation de conception
- ▶ II/ Controller
- ▶ III/ Builder
- ▶ IV/ Frontend
- ▶ V/ Documentation et perspectives d'amélioration

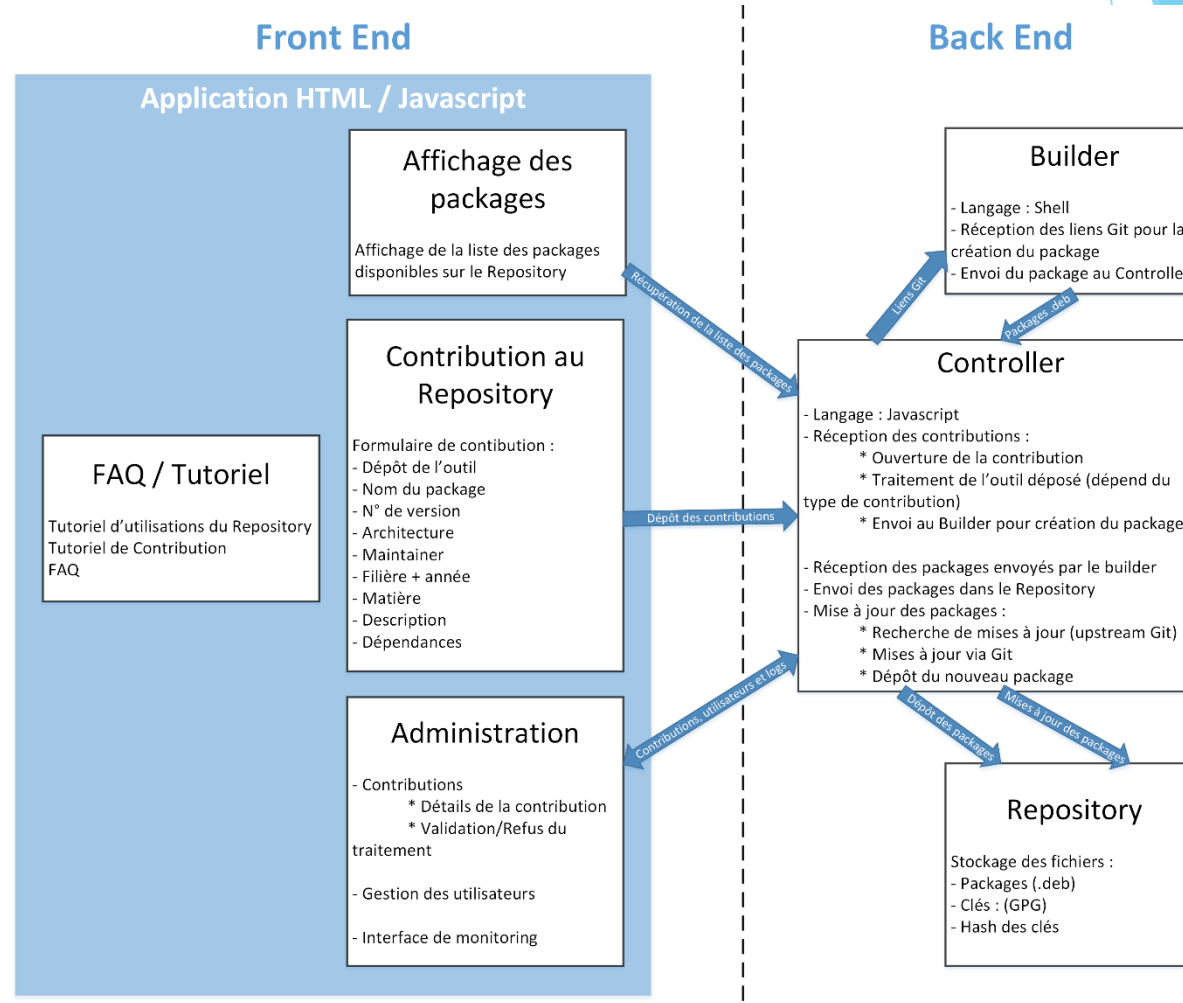
# I/ Structure de l'application

- ▶ But du projet : offrir un gestionnaire de packages pour les étudiants et enseignant de Polytech
- ▶ Objectifs :
  - ▶ Standardiser l'installation des outils nécessaires aux TP et Projets
  - ▶ Permettre aux enseignants de proposer des packages
  - ▶ Automatiser la gestion des mises à jour des packages

# I/ Structure de l'application

## ► Structure MVC

- Frontend Angular
- API en Javascript (Sails)
- Repository pour le stockage des packages



## II/ Controller

- ▶ Gestion des utilisateurs :
  - ▶ Génération de token après connexion avec Auth0
- ▶ Gestion des packages
- ▶ Gestion des tickets (ajout de tickets, système de vote)
- ▶ API testée avec Mocha et Dredd
  - ▶ Permet de tester les fonctionnalités d'authentification

# III/ Builder

- ▶ Générateur de packages
- ▶ Traite les ressources pour obtenir une arborescence de fichiers pour créer le package
- ▶ Supporte plusieurs types d'architectures de projet (Ant, Maven, Gradle, Makefile, binary)
- ▶ Documentation complète sur Git

# IV/ Interface utilisateur

## Login

- ▶ Login avec Auth0 opérationnel
  - ▶ Connexion avec un compte Google
  - ▶ Obtention d'un token propre à notre application
- ▶ Permet de tagger les requêtes POST avec l'id de l'utilisateur
- ▶ Administrateur : Valide les créations de packages

# IV/ Interface utilisateur

## Systeme de vote

- ▶ Idée : pouvoir estimer l'intérêt de l'outil
- ▶ Chaque utilisateur peut voter une fois par ticket
- ▶ Fonctionnement :
  - ▶ Vote inexistant : POST de création
  - ▶ Vote existant :
    - ▶ Vote différent : PUT pour mise à jour
    - ▶ Vote identique : PUT pour annuler le vote



# IV/ Interface utilisateur

## Formulaire de contribution

- ▶ Dépôt de ticket : POST sur l'API
- ▶ Renseignements basiques :
  - ▶ Nom du package, version, architecture, ...
  - ▶ Maintainer, Description, Filière, matière
- ▶ Dépôt des ressources non implémenté
  - ▶ Impossible de déposer sur l'API le lien Git / la tarball

# IV/ Documentation et améliorations

- ▶ Contenu du SRS :
  - ▶ Explication du sujet
  - ▶ Description du fonctionnement de l'application
- ▶ Autres documents sur le Wiki
  - ▶ Détails de conception
  - ▶ Manuel de déploiement
  - ▶ Fichiers README.md
- ▶ Code commenté

# IV/ Documentation et améliorations

- ▶ Fonctionnalités non implémentées :
  - ▶ Privilèges administrateurs
  - ▶ Déclenchement de la génération des packages
  - ▶ Gestion des ressources dans les dépôts de tickets (liens, tarballs)
- ▶ Travail à faire :
  - ▶ Déploiement de l'application
  - ▶ Étudier la solution du CAS de l'université
  - ▶ Améliorer la modularité et la compatibilité du Builder

Merci de votre attention

Questions ?