

# GESTUM

03/10/2017



## GESTion du Temps d'Utilisation des MAchines

- Projet de Partenariat entre Coderoom Orange/Lyon Part-Dieu et La Fabrique d'Objets Libres

# Besoin

**Le Fab lab de grenoble « La Casemate » a développé une application de gestion des FAB LABS appelée fab-manager.**

**Les FabLabs souhaitent lui adjoindre un module de comptabilisation du temps d'utilisation des machines par adhérent.**

**La code room de la DSIF de Lyon propose de réaliser un prototype qui réponde aux besoins exprimés par les FabLabs.**

# Objectif

**Enrichir les données de l'application fab-manager et lui adjoindre un dispositif permettant de décompter le temps d'utilisation d'une machine par un utilisateur.**

**Technologies:**

**- Un serveur local reçoit de chaque machine une information identifiant l'utilisateur connecté, la machine, et l'opération : start/stop.**

**- Un lecteur de carte/arduino équipe chaque machine.**

**- La connectique est assurée par un réseau privé (RJ45/switch)**

**- Les APIS du FabManager devront être utilisées pour le dialogue fabmanager/gestum**

*(non réalisé dans le prototype)*

# Fonctionnalités générales

**Récupération des utilisateurs/machines/certifications depuis le fab manager (par connection direct sur postgresql)**

**Gestion des cartes :**

- Attribution
- Désactivation (non réalisé dans le prototype)
- Carte 'Manager' pour dépannage (non réalisé dans le prototype)

**Activation de la machine si utilisateur possède la compétence (non réalisé dans le prototype)**

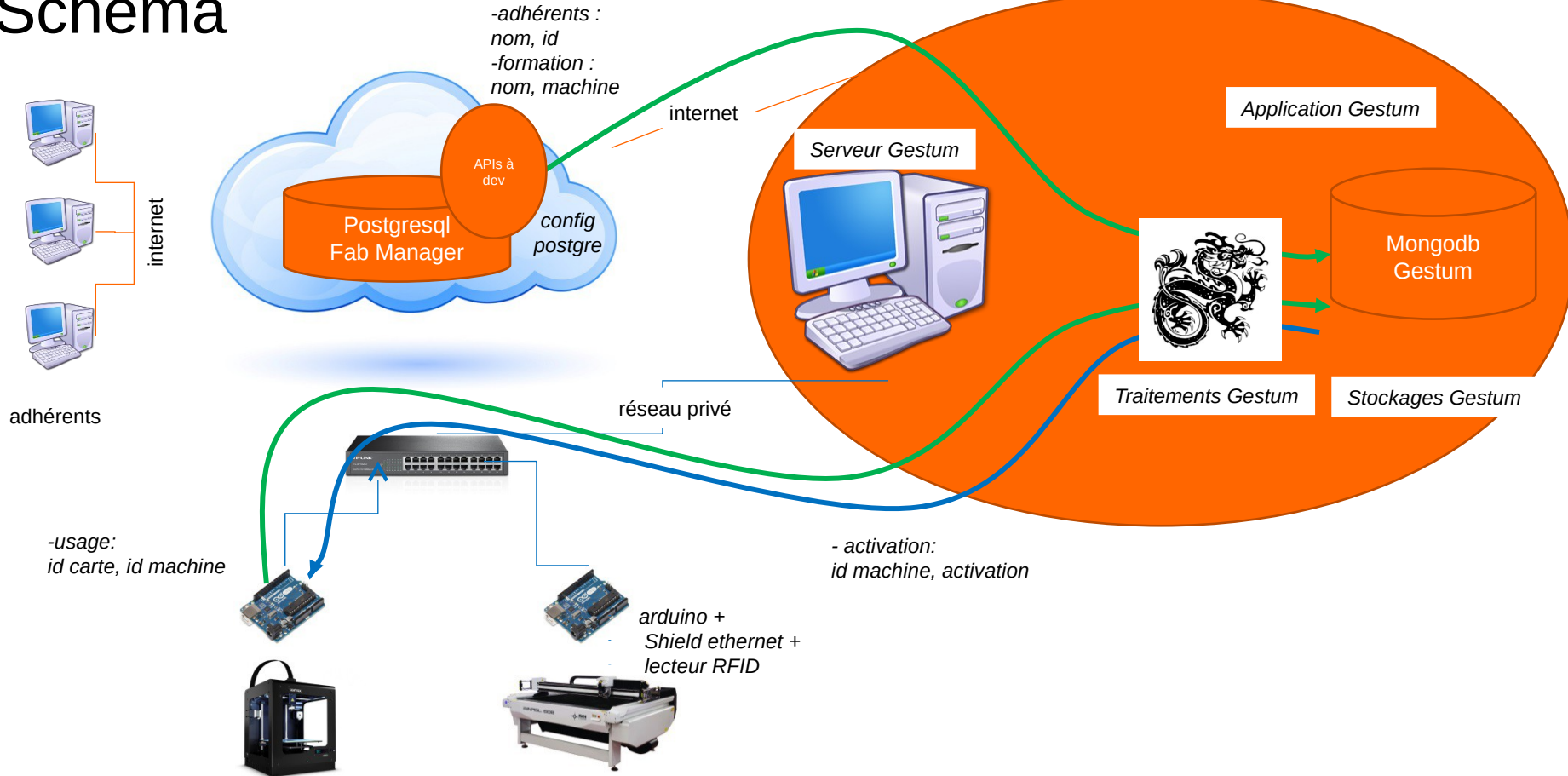
- Gestion intelligente pour éviter de couper en cours une opération longue.

**Enrichissement à la volée et stockage bigdata (id machine, id carte, timestamp)**

**Gestion consommation**

- Journal d'utilisation global
- Etat par utilisateur (non réalisé dans le prototype)
- Graphiques par machine (non réalisé dans le prototype)

# Schéma



# Fonctionnalités détaillées : Connection et administration

**Des utilisateurs de Gestum sont créés.**

- **L'administrateur a accès a toutes les fonctions**
- **Les gestionnaires : (non réalisé dans le prototype)  
créent et affectent les cartes (sauf la carte administrateur)  
consultent les rapports**

# Fonctionnalités détaillées : Récupération utilisateurs/machines/certifications depuis le fab manager

## Lecture asynchrone

- Automatique a intervalle régulier (non réalisé dans le prototype)
- A la demande sur clic depuis Gestum
  - lors de chaque création de adhérent, modification des compétences, ajout de machine

## Copie locale dans MongoDB

- adhérent : nom, prénom, id
- compétences : id, compétence (non réalisé dans le prototype)
- machine : id, nom

## Configuration particulière du prototype :

*Sur serveur fab-manager : Autoriser le serveur Gestum dans pg\_hba.conf*

# Fonctionnalités détaillées : Gestion des cartes :

## Création des identifiants de carte

- Les cartes sont créées à la volée lors d'une première utilisation

## Attribution

- Un identifiant de carte dans la liste est lié à un utilisateur.

*La carte manager ne peut être manipulée que par l'admin (non réalisé dans le prototype)*

## Désactivation

- Une case à cocher permet de désactiver une carte(non réalisé dans le prototype)



# Fonctionnalités détaillées : Activation de la machine si utilisateur possède la compétence

Réception d'un signal venant d'une machine :

- Recherche utilisateur, machine, compétence
- Cas utilisateur inconnu => Signal KO (non réalisé dans le prototype)
- Si compétence présente => Signal OK (non réalisé dans le prototype)

*3 versions aduino:*

*Carte en permanence*

*Carte on / Carte Off*

*Carte On/signal de la machine pour le off*

# Fonctionnalités détaillées : Enrichissement à la volée et stockage bigdata

Réception d'un signal venant d'une machine : (id machine, id carte)

- Enrichissement :

  - rajout de infos utilisateur connu, machine, timestamp

- Stockage MongoDB

*Intérêt de l'enrichissement à la volée :*

*Permet de ne pas avoir à gérer les changements de propriétaire dans les historiques*

# Fonctionnalités détaillées : Gestion consommation

- Etat par utilisateur (non réalisé dans le prototype)
- Graphiques par machine (non réalisé dans le prototype)
- Autres à définir (non réalisé dans le prototype)

# Besoins supplémentaires à étudier

- Une carte peut servir sur plusieurs machines en même temps
- Un afficheur sur le lecteur peut donner des infos en temps réel à l'utilisateur
- Accès aux infos d'un utilisateur depuis internet
- Remontée des infos vers module stat du fab manager
- Gestion du planning? Si la machine est réservée, alerter la personne qui veut lancer un job (NOTA :comment est ce que cela se gère actuellement?) la fonctionnalité est complexe pour un résultat pas forcément probant.

# Suite court terme pour la casemate

**-Installation du proto pour machine en libre service.**

**La carte est donnée par la personne responsable à un utilisateur (existant ou pas dans fabmanager)**

**Le responsable attribue la carte à un utilisateur.**

**Version à utiliser : carte Start, machine STOP <= à valider**

**Le signal start provoque le démarrage si la carte est activée.**

**Besoin de reporting à étudier.**

**Matériel nécessaire : arduino UNO**

**- SODIAL (R) carte capteur d'induction CRFM-522 RC522 Module RFID IC avec la chaine de cle de carte de S50 libre**



# Détail du matériel que j'utilise à commander pour le proto machine Casemate

SODIAL(R) Funduino UNO R3 ATmega328P-PU Carte d'extension Plaque + Cable USB pour Arduino

SODIAL (R) carte capteur d'induction CRFM-522 RC522 Module RFID IC avec la chaine de cle de carte de S50 libre

2 SODIAL(R) Ethernet Carte Extension pour Arduino Mega W5100 2009 UNO 1280 2560

# Merci

Bouton de simulation  
de fin de job par machine

Bouton de simulation  
de fin de job par machine

