

Groupe 4

# Systeme de Gestion de Ressources

Clients : Rachid Khoufache & Antoine Rozenknop



**Version finale**

## TABLE DES MATIERES

I.	INTRODUCTION.....	5
II.	ANALYSE DES BESOINS .....	6
1.	Les besoins fonctionnels .....	6
	Les comptes.....	6
a.	Compte Administrateur.....	6
b.	Compte Utilisateur .....	6
c.	Réservation .....	6
d.	Les salles .....	7
2.	Les besoins non fonctionnels .....	7
a.	Système d'exploitation .....	7
b.	Ergonomie.....	7
c.	Coût .....	7
III.	Architecture globale et intégration au système d'information .....	7
1.	Moyen matériel et logiciel.....	8
2.	Moyen fournisseur.....	8
3.	Moyen matériel .....	8
4.	Présentation de l'équipe .....	8
5.	Diagramme de GANTT.....	9
IV.	CONCEPTION DETAILLEE .....	11
1.	Acteurs primaires .....	11
2.	Les principaux cas d'utilisation.....	11
3.	Description détaillée des principaux cas d'utilisation.....	12
4.	Les cas d'utilisation secondaire.....	15
5.	Description détaillée des cas d'utilisation secondaire.....	15
6.	Diagramme des Cas d'utilisations.....	18
7.	Diagramme de Séquence .....	19
8.	Diagramme des classes.....	21
V.	ANALYSE DE LA SOLUTION EXISTANTE : GRR .....	23
1.	Etude de la solution existante.....	23
2.	Tableau Récapitulatifs des principales fonctionnalités attendues .....	23
3.	Structure de GRR .....	24
a.	Structure globale.....	24
b.	Page Admin_edit_room .....	25
c.	La base de données .....	26
4.	Solution pour la mise à jour automatique de la liste des logiciels installé.....	28

a.	Etude solution OCS inventory .....	28
b.	Fonctionnement.....	28
VI.	REALISATION .....	30
1.	Environnement client .....	30
a.	Présentation EasyPHP .....	30
b.	Installation de l'application.....	31
c.	Création de la base MySQL.....	32
2.	Configuration de GRR .....	32
a.	Effectuer une réservation .....	33
b.	Copier/Modifier/Supprimer une réservation.....	34
c.	Réservation en bloc .....	34
d.	Ajout manuel de logiciel.....	35
e.	Ajout manuel d'ordinateur.....	36
f.	Afficher planning.....	36
g.	Consultation Matériel disponible .....	37
3.	Difficultés rencontrées .....	38
VII.	CONCLUSION ET PERSPECTIVES .....	39
I.	NETOGRAPHIE .....	40

## TABLE DES FIGURES

Figure 1	Cas d'utilisation principaux.....	18
Figure 2	Effectuer une réservation / recherche de ressource.....	19
Figure 3	Ajouter un matériel .....	20
Figure 4	Ajout utilisateur.....	21
Figure 5	Modèle données .....	22
Figure 6	EasyPHP .....	31
Figure 7	Configuration GRR.....	32
Figure 8	Effectuer une réservation .....	33
Figure 9	Modifier /Supprimer/Reservation .....	34
Figure 10	Réservation par bloc .....	34
Figure 11	Ajout Matériel .....	35
Figure 12	Planning .....	36
Figure 13	Afficher description .....	37

# Modifications

---

Architecture globale intégration au SI.....	7
Diagrammes de séquence.....	20
Analyse de la solution existante.....	23

## I. INTRODUCTION

Le système de gestion des salles informatiques de Sup Galilée permet de réserver une salle et affiche un planning journalier de l'occupation de celle-ci. Néanmoins ce système ne permet pas une gestion optimisée des ressources disponibles dans les salles de TP et les logiciels installés.

Dans le cadre de la formation d'ingénieur en 2ème année informatique, nous avons choisi de développer un Système de réservation de salles et de gestion de leurs ressources comme projet. En effet ce projet est intéressant tant par l'aspect pratique du cadre de développement (université) que par la complexité de réalisation qui demande à la fois des compétences techniques à savoir la programmation (SQL et PHP) principalement mais aussi des compétences moins technique telles que la modélisation et la conduite de projet.

Dans ce rapport nous allons présenter les différentes étapes du développement de notre application. Pour cela nous commencerons par présenter le contexte et l'environnement de travail. Puis nous exposerons l'architecture globale du projet pour poursuivre avec la conception détaillée. Nous finirons par présenter nos réalisations et clôturerons avec une petite conclusion.

## II. ANALYSE DES BESOINS

### 1. Les besoins fonctionnels

#### Les comptes

Pour pouvoir réserver et accéder aux fonctionnalités autres que celle de la consultation des disponibilités et des attributions, l'application devra permettre la création de comptes utilisateurs.

Chaque utilisateur possédera un login et un mot de passe.

#### a. Compte Administrateur

- Cet utilisateur sera en charge de la gestion globale de la plateforme.
- Ajout/ suppression de compte.
- Distribution des droits d'accès.
- Ajout/ suppression de ressource.
- Mise à jour du système.
- Maintenance globale.

#### b. Compte Utilisateur

Il y aura un utilisateur par formation en charge de la réservation des salles.

- Effectuer/annuler une réservation.
- Consulter le planning.
- Consulter les ressources.

Ce compte permettra d'accéder à des informations globales concernant les formations qui lui sont attribuées.

#### c. Réservation

Un utilisateur possédant les droits nécessaires pourra accéder à une page réservation qui affichera tous les créneaux horaires disponibles.

- Consultation du planning des salles et des ressources disponibles.
- Effectuer une réservation.
- Annulation de réservation.
- Recherche par mot clé.
- Mise à jour automatique du planning.
- Sauvegarde d'un historique de réservation.

#### **d. Les salles**

Les salles seront classées en catégories. (Machines, Classique, TP, Amphithéâtre). Chaque salle sera identifiée par un numéro, un nom, une capacité, nombre de prises électriques et en fonction de sa catégorie on renseignera d'autres champs à savoir :

- Salle machine : Nombre de machine, logiciels installé, projecteur, système d'exploitation.
- Amphis: projecteurs, micros, haut-parleurs.
- Salle de TP

## **2. Les besoins non fonctionnels**

### **a. Système d'exploitation**

L'application sera portable et devra tourner au moins sous Linux (Ubuntu, Debian, Mandriva) et Windows (XP, Seven, Vista).

### **b. Ergonomie**

L'interface devra être simple et intuitive. Elle sera utilisable par des personnes n'ayant pas une formation d'informaticien.

### **c. Coût**

L'application sera peut coûteuse en argent mais aussi en mémoire.

## **III. Architecture globale et intégration au système d'information**

## **1. Moyen matériel et logiciel**

Pour le bon déroulement du projet, nous devons disposer d'au moins un ordinateur pouvant accueillir une base de données pour les tests, ainsi que le logiciel GRR.

Nous pourrions utiliser les machines mises à notre disposition par la formation pour effectuer le projet ce qui permet un investissement matériel nul. De plus le logiciel GRR étant open source, l'investissement logiciel est aussi nul.

## **2. Moyen fournisseur**

Le client fournit certaines technologies préexistantes, dans le cas présent, le logiciel GRR nous a été fourni, ainsi que le nom du logiciel qui est actuellement utilisé par le client « 4D ». De plus, le client nous a fourni une piste pour pouvoir s'adresser à des personnes aptes à expliquer le fonctionnement du logiciel 4D.

## **3. Moyen matériel**

Afin de mener à bien notre projet, nous devons avoir à notre disposition 4 ordinateurs compatibles avec de multiples navigateurs Internet (IE, Firefox, safari ...), afin de coder efficacement en PHP une plateforme compatible avec tous les navigateurs. Il nous faudra également une base de données MySQL afin de pouvoir créer une base de données nécessaire au fonctionnement du logiciel.

## **4. Présentation de l'équipe**

Notre équipe est composée de 4 étudiants en école ingénieurs informatique deuxième année dont les différents CV sont joints en annexe.

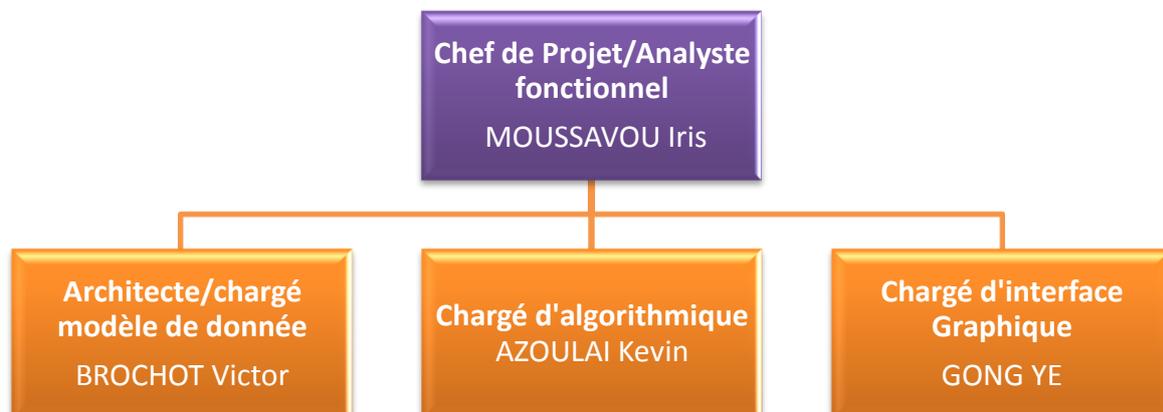


Figure 1 Equipe

## 5. Diagramme de GANTT

Aussi le développement de notre solution s'articule autour du planning suivant :



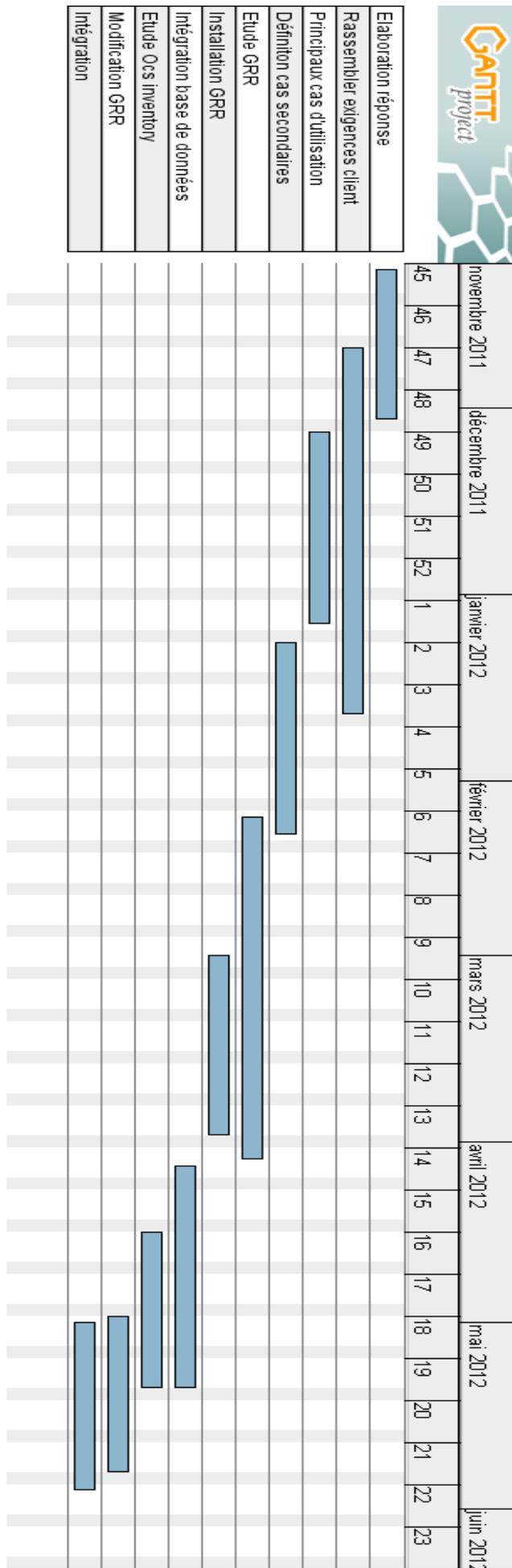


Figure 2 Diagramme de GANTT

## IV. CONCEPTION DETAILLEE

Le système que nous allons développer permet de gérer les réservations de salles machine au sein de l'Institut Galilée. Le système sera accessible via une plateforme Internet.

Chaque utilisateur autorisé pourra accéder à une page réservation qui affichera tous les créneaux horaires disponibles. Une réservation ne pourra se faire que si le créneau horaire est libre. Deux réservations pour une même salle à une même date ne seront pas possible.

### 1. Acteurs primaires

L'utilisateur générique de notre système peut être divisé en plusieurs catégories qui sont :

- ✚ **Les administrateurs**
- ✚ **Les responsables de formation**
- ✚ **Les étudiants et enseignants**

### 2. Les principaux cas d'utilisation

Globalement, le système doit permettre de gérer des salles machines. Ainsi, le système prend en charge les étapes de réservations et de gestion du matériel des salles.

- ✚ **Se connecter au système**
- ✚ **Ajouter un matériel**
- ✚ **Attribuer des droits d'accès**
- ✚ **Consulter créneau horaire**

+ Effectuer une réservation

+ Annuler une réservation

### 3. Description détaillée des principaux cas d'utilisation

+ Cas d'utilisation : Se connecter au système.

**Niveau :** But utilisateur

**Système :** SYSTÈME DE GESTION DE RESSOURCES

**But de ce cas d'utilisation :** permettre à un utilisateur de se connecter au système. Il peut le faire par internet.

**Acteur primaire :** Utilisateur

**Scénario de succès principal :**

- 1) L'utilisateur se rend sur la page d'accueil du système.
- 2) L'utilisateur s'identifie sur le système.

**Extensions :**

- 2)a) Si l'identifiant de l'utilisateur n'est pas valide il est invité à contacter l'administrateur.

+ Cas d'utilisation : Ajouter un matériel

**Niveau :** But utilisateur

**Système :** SYSTÈME DE GESTION DE RESSOURCES

**But de ce cas d'utilisation :** permettre à un administrateur d'ajouter un matériel dans les attributs d'une salle.

**Acteur primaire :** Administrateur ou responsable

**Scénario de succès principal :**

- 1) L'administrateur se rend sur la page d'ajout de matériel
- 2) L'administrateur ajoute le matériel
- 3) L'administrateur met à jour le système

+ Cas d'utilisation : Consulter un créneau horaire

**Niveau :** But utilisateur

**Système :** SYSTÈME DE GESTION DE RESSOURCES

**But de ce cas d'utilisation :** permettre à un utilisateur de consulter le planning des salles machine.

**Acteur primaire :** Utilisateur

**Scénario de succès principal :**

- 1) L'utilisateur se rend sur la page d'accueil du système.
- 2) L'utilisateur accède à la page planning.

3) L'utilisateur sélectionne une salle.

**Extensions :**

2)a) Si il n'y a pas de salle disponible à la date indiquée le cas d'utilisation reprend à l'étape 1.

 **Cas d'utilisation : Effectuer une réservation**

**Niveau :** But utilisateur

**Système :** SYSTÈME DE GESTION DE RESSOURCES

**But de ce cas d'utilisation :** permettre à un responsable d'effectuer la réservation de salle.

**Acteur primaire :** Responsable formation

**Pré-requis :** Le responsable s'est identifié et possède les droits nécessaires.

**Scénario de succès principal :**

- 1) L'utilisateur se rend sur la page d'accueil du système.
- 2) L'utilisateur consulte le matériel disponible.
- 3) L'utilisateur valide sa réservation.

**Extensions :**

2)a) Si il n'y a pas de salle disponible à la date indiquée le cas d'utilisation reprend à l'étape 1.

4)a) Si l'un de matériel dont l'utilisateur a besoin n'est pas présent dans la salle le cas d'utilisation reprend à l'étape 3.

 **Cas d'utilisation : Annuler une réservation**

**Niveau :** But utilisateur

**Système :** SYSTÈME DE GESTION DE RESSOURCES

**But de ce cas d'utilisation :** Permettre à un utilisateur d'annuler une réservation.

**Acteur primaire :** Responsable formation

**Pré-requis :** Le responsable s'est identifié et possède les droits nécessaires. L'annulation a lieu 48h avant la date de réservation.

**Scénario de succès principal :**

- 1) L'utilisateur se rend sur la page de la liste de ses réservations.
- 2) L'utilisateur sélectionne une réservation.
- 3) L'utilisateur annule la réservation.

 **Cas d'utilisation : Consulter Historique**

**Niveau :** But utilisateur

**Système :** SYSTÈME DE GESTION DE RESSOURCES

**But de ce cas d'utilisation :** permettre à un utilisateur de consulter l'ensemble de ses réservations

**Acteur primaire :** Responsable formation

**Pré-requis :** Le responsable s'est identifié et possède les droits nécessaires.

**Scénario de succès principal :**

- 1) L'utilisateur se rend sur la page d'accueil
- 2) L'utilisateur sélectionne « consulter l'historique ».
- 3) L'utilisateur indique une période et valide.
- 4) L'utilisateur consulte l'historique.

**Extensions :**

3)a) Si il n'y a pas eu de réservation le cas d'utilisation reprend à l'étape 3.

### **Cas d'utilisation : Effectuer une recherche de ressource**

**Niveau :** But utilisateur

**Système :** SYSTÈME DE GESTION DE RESSOURCES

**But de ce cas d'utilisation :** Permettre à un utilisateur de rechercher une ressource dans le système.

**Acteur primaire :** Responsable formation

**Pré-requis :** Le responsable s'est identifié et possède les droits nécessaires.

**Scénario de succès principal :**

- 1) L'utilisateur se rend sur la page de recherche
- 2) L'utilisateur entre le nom de la ressource et valide.
- 3) L'utilisateur consulte les salles qui possèdent cette ressource.

**Extensions :**

3)a) Si la ressource n'existe pas dans le système le cas d'utilisation reprend à l'étape 2.

### **Cas d'utilisation : Mettre à jour le matériel**

**Niveau :** But utilisateur

**Système :** SYSTÈME DE GESTION DE RESSOURCES

**But de ce cas d'utilisation :** permettre au système de mettre à jour automatiquement la liste des logiciels installés.

**Acteur primaire :** Le système

**Pré-requis :** Le responsable s'est identifié et possède les droits nécessaires.

**Scénario de succès principal :**

- 1) L'administrateur installe un logiciel sur une machine
- 2) Le système met à jour l'ensemble du matériel.

#### 4. Les cas d'utilisation secondaire

En plus des principaux cas d'utilisation le système inclus des fonctionnalités secondaires

- ✚ Consulter statistiques
- ✚ Envoi de mail automatique
- ✚ Réservation par block
- ✚ Fixer une deadline
- ✚ Copier une ressource
- ✚ Rendre une ressource indisponible

#### 5. Description détaillée des cas d'utilisation secondaire

##### ✚ Cas d'utilisation : Consulter statistiques

**Niveau :** But utilisateur

**Système :** SYSTÈME DE GESTION DE RESSOURCES

**But de ce cas d'utilisation :** permettre à un utilisateur de consulter les statistiques.

**Acteur primaire :** Utilisateur

**Scénario de succès principal :**

- 1) L'utilisateur s'identifie
- 2) L'utilisateur se rend sur la page statistique
- 3) L'utilisateur sélectionne un domaine
- 4) L'utilisateur sélectionne une période
- 5) L'utilisateur consulte les statistiques

**Extensions :**

4)a) Si l'utilisateur n'a pas effectué de réservation dans cette période le cas d'utilisation reprend à l'étape 4.

##### ✚ Cas d'utilisation : Envoi de mails d'information automatique

**Niveau :** But utilisateur

**Système :** SYSTÈME DE GESTION DE RESSOURCES

**But de ce cas d'utilisation :** Permet à l'utilisateur de recevoir des mails de rappel

**Acteur primaire :** L'administrateur

**Scénario de succès principal :**

- 1) L'utilisateur s'identifie
- 2) L'utilisateur se rend sur la page de configuration générale du système
- 3) L'administrateur sélectionne l'envoi de mails automatique

#### **Cas d'utilisation : Effectuer une réservation par bloc**

**Niveau :** But utilisateur

**Système :** SYSTÈME DE GESTION DE RESSOURCES

**But de ce cas d'utilisation :** permettre à un responsable de réserver ou de libérer très rapidement des journées entières simultanément sur plusieurs ressources de plusieurs domaines et selon un calendrier.

**Acteur primaire :** Responsable formation

**Pré-requis :** Le responsable s'est identifié et possède les droits nécessaires.

**Scénario de succès principal :**

- 1) L'utilisateur se rend sur la page d'accueil du système.
- 2) L'utilisateur consulte le matériel disponible.
- 3) L'utilisateur sélectionne une ressource.
- 4) L'utilisateur sélectionne réserver par bloc.
- 5) L'utilisateur sélectionne un créneau horaire.
- 6) L'utilisateur valide la réservation

**Extensions :**

- 5)a) Si il n'y a pas de salle disponible à la date indiquée le cas d'utilisation reprend à l'étape 1.
- 4)a) Si la ressource dont l'utilisateur a besoin n'est pas disponible le cas d'utilisation reprend à l'étape 3.

#### **Cas d'utilisation : Fixer une deadline**

**Niveau :** But utilisateur

**Système :** SYSTÈME DE GESTION DE RESSOURCES

**But de ce cas d'utilisation :** Permet à l'administrateur de fixer un délai au delà duquel un utilisateur ne peut pas annuler ou effectuer une réservation

**Acteur primaire :** L'administrateur

**Scénario de succès principal :**

- 1) L'utilisateur s'identifie
- 2) L'utilisateur se rend sur la page de configuration générale du système
- 3) L'utilisateur indique le nombre d'heure avant la réservation/annulation
- 4) L'utilisateur indique le nombre d'heure après la réservation/annulation
- 5) L'utilisateur sauvegarde la configuration

#### **Cas d'utilisation : Copier une ressource**

**Niveau :** But utilisateur

**Systeme :** SYSTEME DE GESTION DE RESSOURCES

**But de ce cas d'utilisation :** Permet à l'administrateur de dupliquer une ressource

**Acteur primaire :** L'administrateur

**Scénario de succès principal :**

- 1) L'utilisateur s'identifie
- 2) L'utilisateur se rend sur la page de configuration générale du système
- 3) L'utilisateur ajoute une ressource
- 4) L'utilisateur indique quelle ressource il souhaite dupliquer

### **Cas d'utilisation : Rendre une ressource indisponible**

**Niveau :** But utilisateur

**Systeme :** SYSTEME DE GESTION DE RESSOURCES

**But de ce cas d'utilisation :** Permet à l'administrateur de rendre une ressource indisponible

**Acteur primaire :** L'administrateur

**Scénario de succès principal :**

- 5) L'utilisateur s'identifie
- 6) L'utilisateur se rend sur la page de configuration générale du système
- 7) L'utilisateur sélectionne une ressource
- 8) L'utilisateur indique les dates d'indisponibilités de la ressource.

## 6. Diagramme des Cas d'utilisations

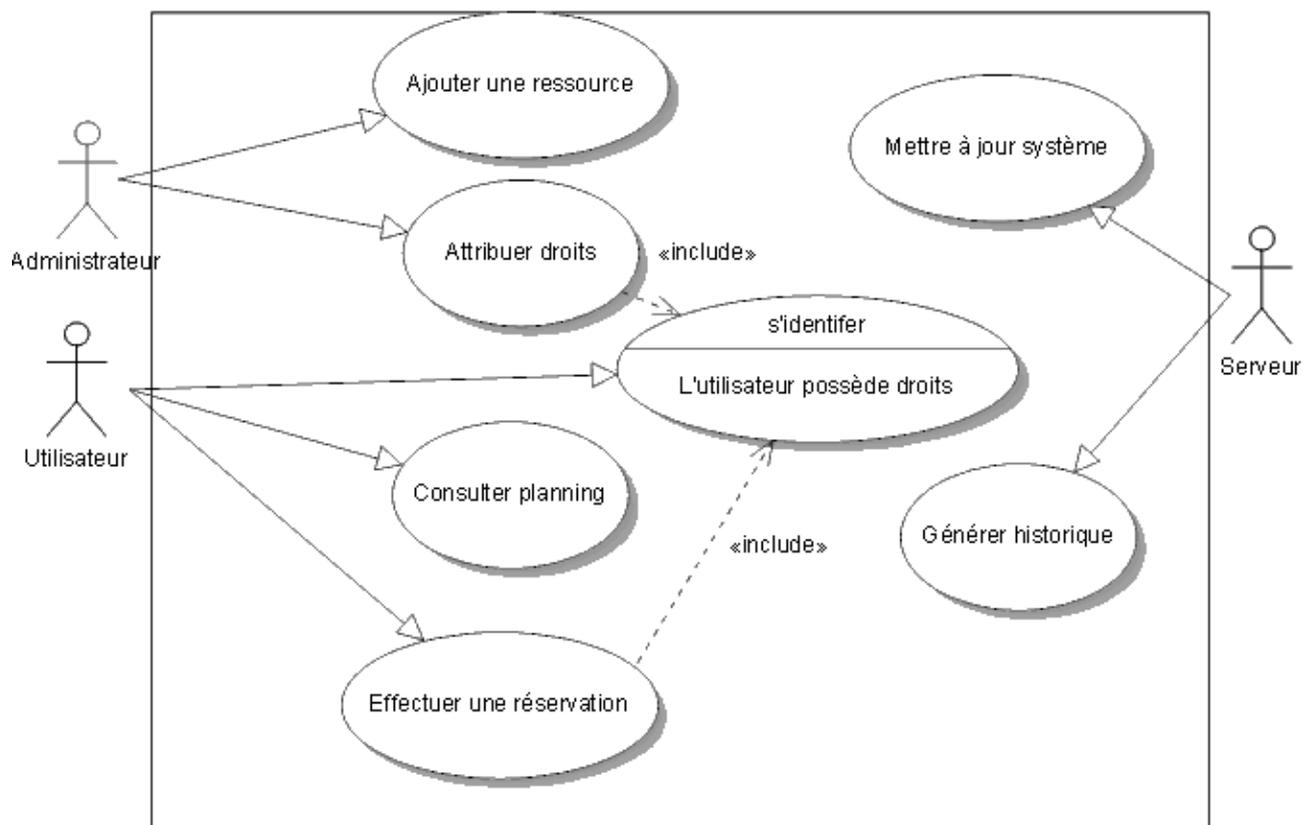


Figure 3 Cas d'utilisation principaux

## 7. Diagramme de Séquence

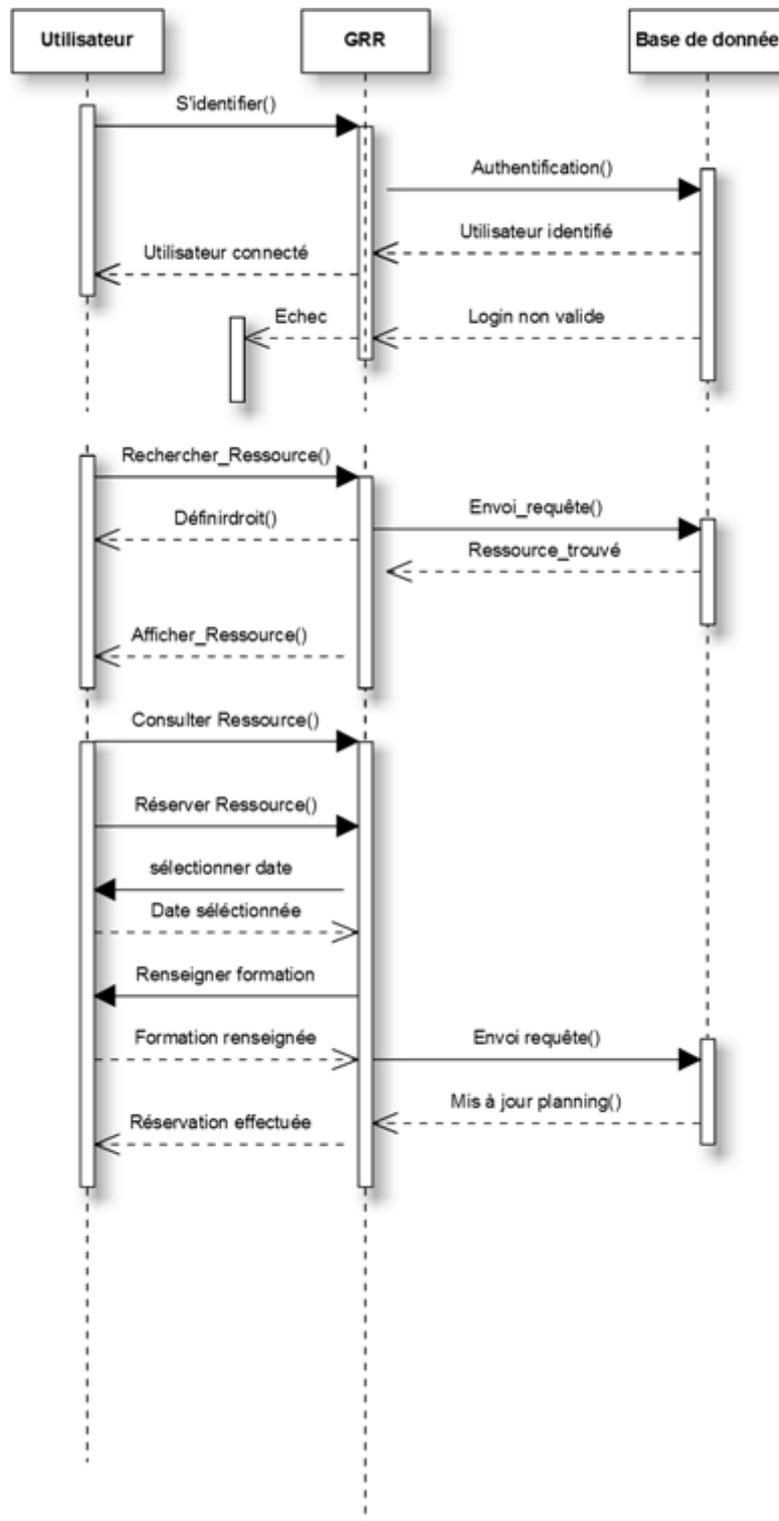


Figure 4 Effectuer une réservation / recherche de ressource

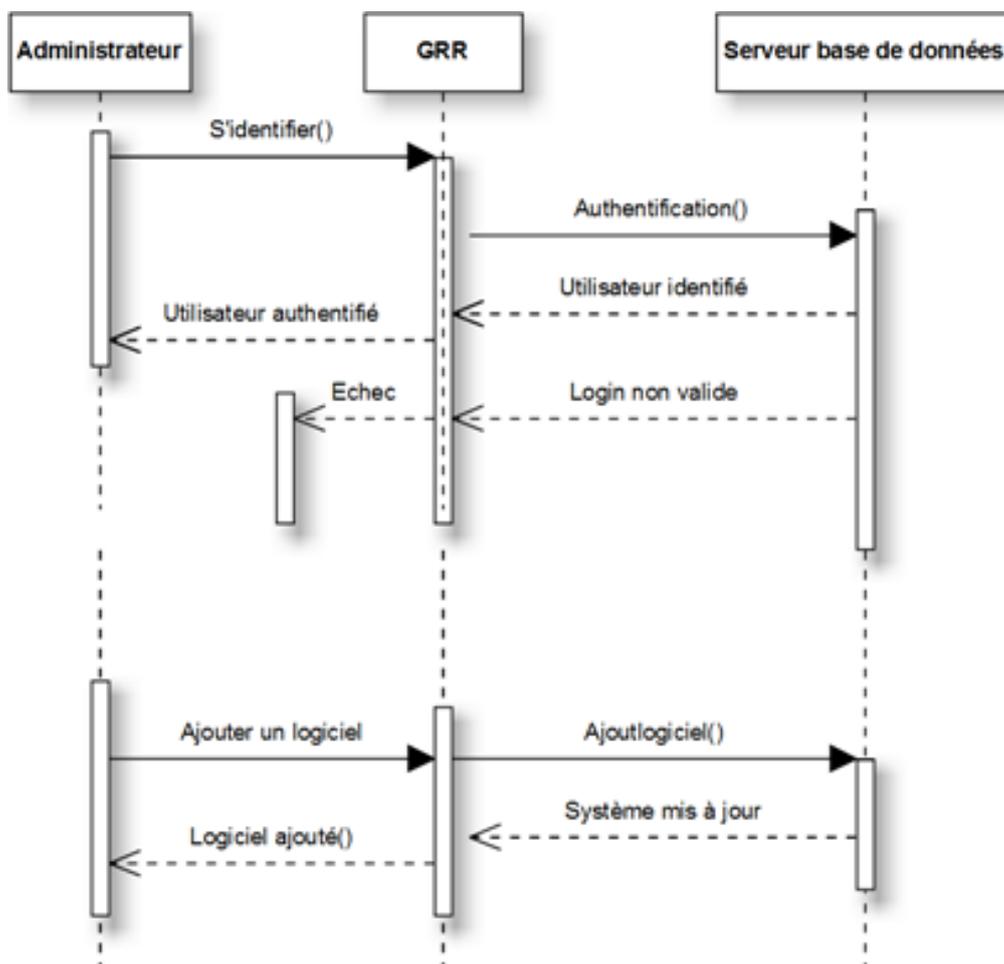


Figure 5 Ajouter un matériel

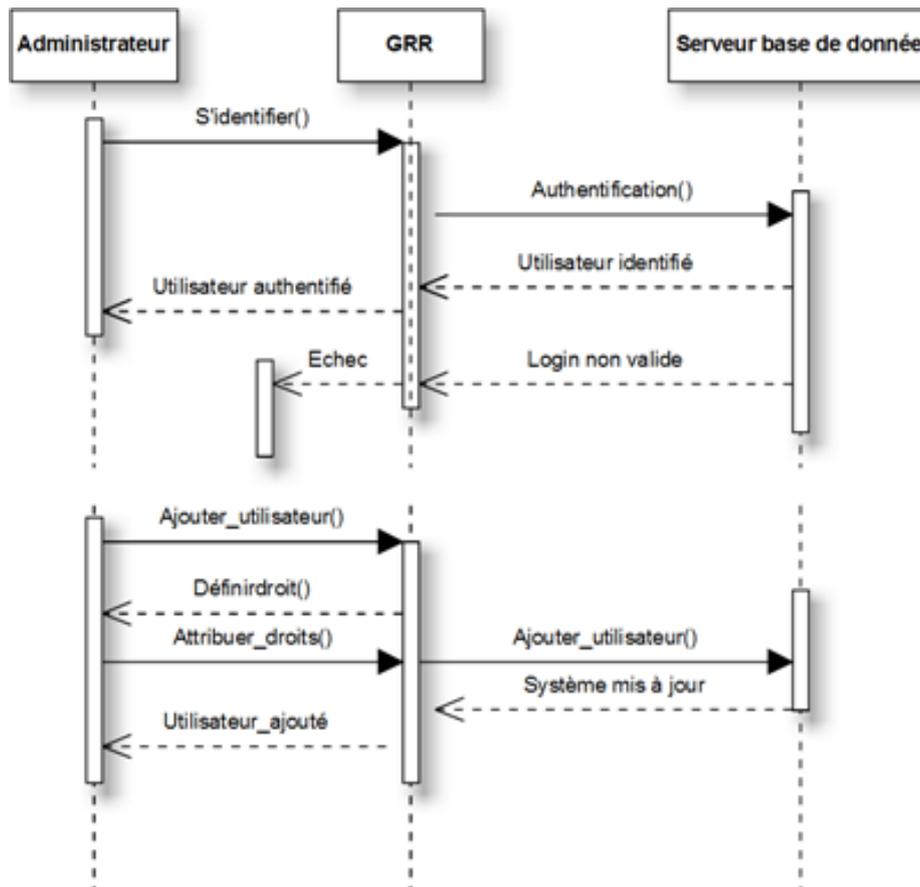


Figure 6 Ajout utilisateur

## 8. Diagramme des classes

Dans le cadre de la mise à jour manuel du matériel nous avons optés pour la définition de la base de données suivante :

Nous avons identifié les classes suivantes :

- ✚ UTILISATEUR
- ✚ RESERVATION
- ✚ MATERIEL
- ✚ SALLE
- ✚ ORDINATEUR
- ✚ LOGICIEL
- ✚ INSTALLATION

## Class Diagram

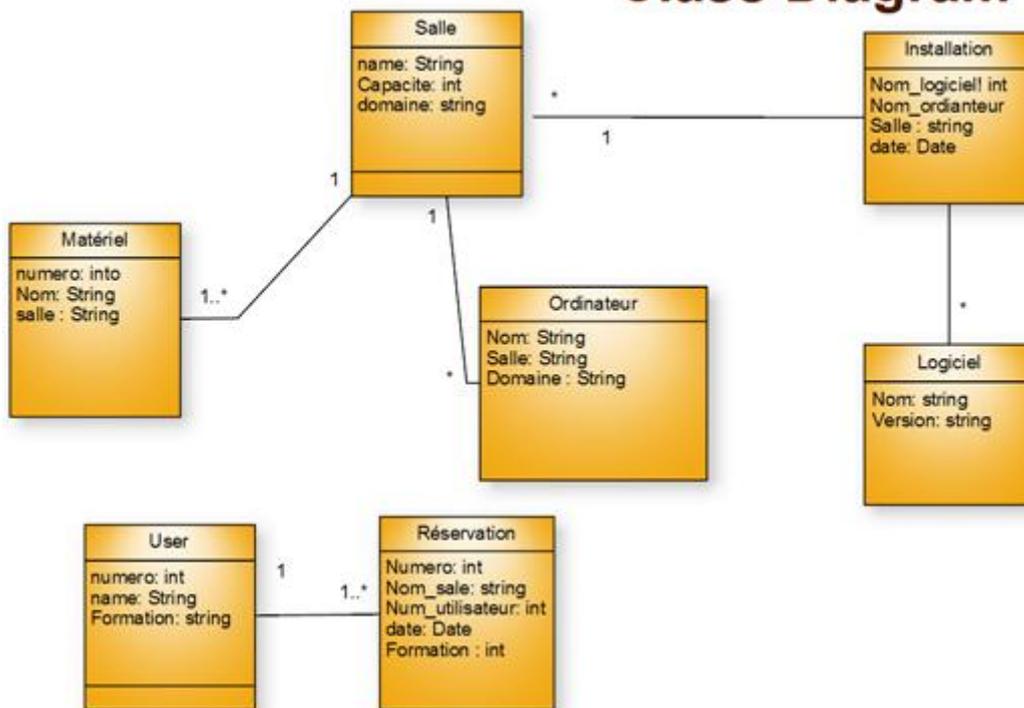


Figure 7 Modèle données

## V. ANALYSE DE LA SOLUTION EXISTANTE : GRR

### 1. Etude de la solution existante

Dans un premier temps nous avons étudié le GRR. Le GRR est un système de gestion de ressources. Ce logiciel est open source.

Dans cette étape nous avons analysé les fonctionnalités de ce logiciel et identifié les cas d'utilisation qui correspondent aux attentes de notre client.

### 2. Tableau Récapitulatifs des principales fonctionnalités attendues

Fonctionnalités	GRR	Amélioration en cours
Se connecter au système	OK	–
Ajouter un matériel	Possible manuellement	Automatiser la mise à jour via le client OCS Inventory
Attribuer des droits d'accès	Différent niveau de droits existant	
Consulter créneau horaire	Permet l'affichage d'un planning	–
Effectuer une réservation	OK	Configurer en rajoutant des champs obligatoires ou non selon les paramètres fixés par l'administrateur
Annuler une réservation	OK	Possibilité de configurer

Statistique des réservations	OK
Rechercher du matériel	- A IMPLEMENTER
Enregistrer historique	OK

### 3. Structure de GRR

#### a. Structure globale

GRR contient une structure arborescente de dossiers.

Le dossier parent est le dossier « gr197e » il contient les principaux fichiers PHP nécessaires au bon fonctionnement de l'application ainsi que des dossiers de configuration:

##### **Include**

Contient les fonctions appelées dans les fichiers principaux de GRR. Par exemple le fichier de connexion à la base de données.

##### **Img\_grr**

Contient les images utilisées dans le logiciel.

##### **Lasso**

Contient les fichiers nécessaires à l'implémentation de Lasso. Lasso est un logiciel gratuit de bibliothèque C visant à mettre en œuvre les standards Liberty Alliance, elle définit les processus des identités fédérées, simples protocoles de connexion sur et connexes. Lasso est construit au-dessus de libxml2, XMLSec et OpenSSL et est sous licence GNU General Public License (avec une exception OpenSSL).

##### **CKEditor**

CKeditor est un éditeur de texte utilisé dans les pages web de GRR.

##### **Phpmailer**

Full Featured Email Transfer Class for PHP. C'est une classe qui permet d'implémenter la fonction mail() de php.

##### **Language**

Contient les langages utilisés par GRR à savoir : anglais, français, espagnol et allemand.

## Themes

Contient les informations relatives aux thèmes disponibles dans GRR.

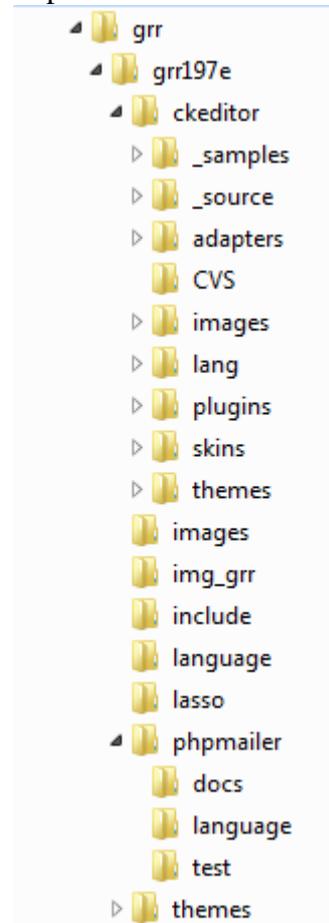


Figure 8 Structure globale GRR

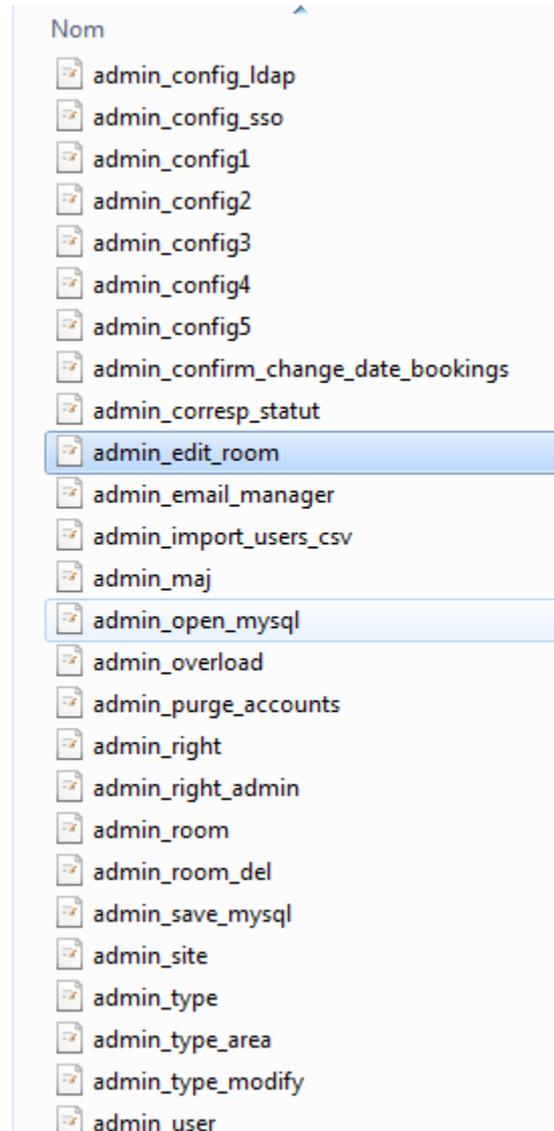


Figure 9 Fichier principaux

### b. Page Admin\_edit\_room

Cette page est celle où nous allons opérer des modifications afin qu'elle puisse répondre aux attentes de notre client. En effet, elle permet l'enregistrement d'une ressource. Elle permet également la modification des paramètres globaux qui décrivent une salle. Elle permet de modifier le nom de la salle, de définir sa capacité, de la mettre ou non en disponibilité, etc.

Elle permet également de rajouter la description (matériels disponibles) à l'aide du champ « description complète ». Toutefois, cette page ne répond pas aux attentes de notre client qui désire pouvoir avoir une liste de matériels avec un ajout manuel et/ou automatique.

Choisir une image de la ressource pour la fiche de présentation (png, jpg et gif uniquement).	<input type="text"/> <input type="button" value="Parcourir..."/>
Supprimer l'image actuelle de la ressource : (aucun)	<input type="checkbox"/>
Afficher la description complète dans le titre des plannings.	<input type="checkbox"/>
Description complète (visible dans la fiche de présentation) :	
<div style="border: 1px solid gray; height: 150px; width: 100%;"></div>	

#### Configuration des fonctionnalités

Pour une nouvelle réservation ou modification d'une réservation, l'utilisateur spécifie la date/heure de début de réservation et :	<input checked="" type="radio"/> la durée de la réservation <input type="radio"/> la date/heure de fin de réservation
Nombre de personnes maximum autorisé dans la salle (0 s'il ne s'agit pas d'une salle):	<input type="text" value="40"/>
Nombre max. de réservations par utilisateur (-1 si pas de restriction)	<input type="text" value="-1"/>
Nombre maximal de jours au-delà duquel l'utilisateur ne peut pas réserver ou modifier une réservation (-1 si pas de restriction).	<input type="text" value="-1"/>

Figure 10 Modifier ressource

### c. La base de données

**grr\_j\_mailuser\_room** : stock les salles réservées par un utilisateur.

**grr\_j\_user\_area** : associe à un utilisateur un domaine.

**grr\_j\_user\_room** : stock les salles réservées par un utilisateur.

**grr\_j\_useradmin\_area** : contient les paramètres généraux des espaces user.

**grr\_area** : table qui contient les paramètres généraux des réservations.

**grr\_log** : contient les informations relatives à une session.

**grr\_entry** : table réservations.

**grr\_repeat** : stock les réservations groupées.

**grr\_room** : table salle.

**grr\_utilisateurs** : contient les utilisateurs de GRR.

**grr\_setting** : qui contient le nom des paramètres.

**grr\_area\_periodes** : contient les périodes des différents domaines.

**grr\_type\_area** : contient le statut d'un domaine.

**grr\_j\_type\_area** : associe un type à un domaine.

**grr\_calendar** : stock les journées réservées depuis le début des réservations.

**grr\_calendrier\_jours\_cycle** : stock les journées réservées cycliquement.

**grr\_overload** : enregistre les dépassements en cas de délais non respectés.

**grr\_entry\_moderate** : contient les paramètres de modération.

**grr\_site** : contient les informations relatives à un site (localisation).

**grr\_j\_site\_area** : associe un site à un domaine.

**grr\_j\_useradmin\_site** : associe un site à un utilisateur.



## 4. Solution pour la mise à jour automatique de la liste des logiciels installés

### a. Etude solution OCS inventory

**Open Computers and Software Inventory Next Generation** est une solution de gestion technique de parc informatique. Ce logiciel a pour but d'aider l'administrateur réseau ou système à posséder un état de la configuration :

- Ordinateurs (Hard et périphériques)
- Logiciels



**Informations d'inventaire** pertinentes.



**Système évolué de télédiffusion** permettant de déployer des installations de logiciels ou d'exécuter des scripts et commandes sur les ordinateurs sans surcharger le réseau.



**Interface d'administration web.**



**Support de nombreux systèmes d'exploitation** incluant Microsoft Windows, Linux, \*BSD, Sun Solaris, IBM AIX, HP-UX, MacOS X.



**Architecture tierce** utilisant les standards courants, les protocoles HTTP/HTTPS et le formatage de données XML.



**Web service** accessible au travers de l'interface SOAP.



**Support de plugins** au travers des API.



**Recherche sur le réseau**



**Synchronisation avec GLPI**

### b. Fonctionnement

Ocs Inventory collecte les informations suivantes :

- ✚ **Type**
- ✚ **Disques logiques / partitions**
- ✚ **Système d'exploitation**
- ✚ **Logiciels**
- ✚ **Ecran**
- ✚ **Description de l'ordinateur**

Ces informations sont stockées dans une base de données et peuvent être récupéré grâce à un ensemble requêtes.

## VI. REALISATION

Lors du développement de notre application nous avons reconfiguré GRR et ajouter certains modules afin qu'il puisse répondre aux attentes de nos clients.

### 1. Environnement client

Nous avons choisi de déployer notre application en local sur les systèmes linux Debian et Windows 7. L'application est disponible via un simple navigateur web. Pour la développer nous avons utilisé les langages PHP MySQL et JAVAScript.

#### a. Présentation EasyPHP

Le langage PHP est un langage extrêmement puissant : il permet de créer des pages web, au travers desquelles l'utilisateur peut échanger des informations avec le serveur ; c'est ce qu'on appelle des pages web dynamiques. Programmer en PHP est assez simple. En revanche, PHP n'est pas un langage compilé, c'est un langage interprété par le serveur : le serveur lit le code PHP, le transforme et génère la page HTML. Pour fonctionner, il a donc besoin d'un serveur web. Il s'offre alors deux solutions :

- ✚ **Soit on utilise un programme FTP qui envoi les fichier sur un serveur web pour les tester.**
- ✚ **Soit on installe un serveur web en local, qui permet de tester directement les pages PHP.**

EasyPHP permet de réaliser cette dernière solution.

Enfin, EasyPHP permet également d'installer MySQL, une base de données, le troisième et inséparable membre du trio Apache/PHP/MySQL. On peut de plus accéder très facilement à une base de données MySQL à partir de PHP, ce qui permet de développer des sites web très performants et interactifs.

EasyPHP joint PHPMyAdmin à MySQL, un outil écrit en PHP permettant de gérer les bases de données MySQL.

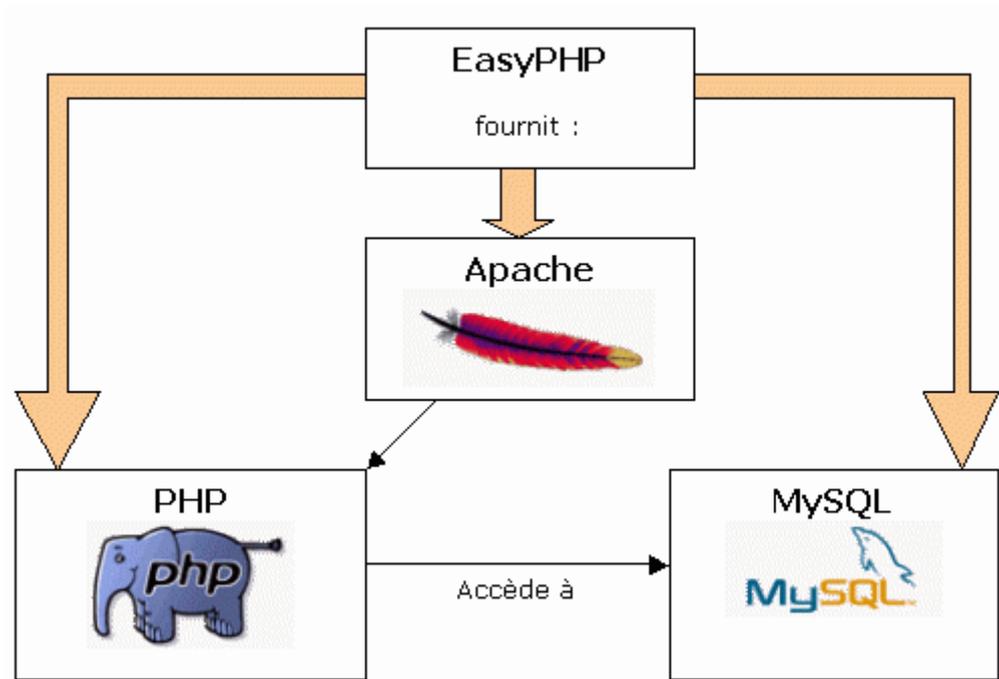


Figure 12 EasyPHP

### b. Installation de l'application

La première étape de l'installation consiste à transférer tous les fichiers de l'archive que vous avez téléchargée vers le serveur web/php.

Pour cela, il faut se munir des codes des paramètres de connexion au serveur et utiliser un logiciel de transfert de fichiers (FTP).

On pourra par exemple créer un répertoire "grr" dans le répertoire web du serveur ("htdocs" dans le cas d'Apache).

Modification des droits :

- Au répertoire "/images" (chargement d'images pour les fiches de présentation des ressources)
- Au fichier "include/connect.inc.php" (dans le cas d'une installation automatisée)
- Au fichier "include/config\_ldap.inc.php" (dans le cas d'une authentification ldap)

### c. Création de la base MySQL

Nous avons le choix entre deux types d'installation de la base de données Mysql:

- Une installation automatisée,
- Une installation manuelle, réservée aux experts.

## 2. Configuration de GRR

Mise en place des différents niveaux d'utilisateurs :

- ✚ Administrateur
- ✚ Responsable de formation
- ✚ Utilisateur

	Accueil - Sup-Galilée Bienvenue grr Administrateur Gérer mon compte -  Se déconnecter	Administration <input type="button" value="Lancer une sauvegarde"/> 1 personne connectée
---	---	--

### Ajout d'un nouvel utilisateur

[| Retour |](#)

Les champs marqués d'un \* sont obligatoires.

Identifiant \* :

Nom \* :  Prénom \* :

Email :  Statut :  Etat :

Mot de passe (6 caractères minimum) \* :

Mot de passe (à confirmer) \* :

Figure 13 Configuration GRr

## a. Effectuer une réservation

Cette page permet l'ajout d'une réservation en cliquant sur l'icône .



The screenshot shows the top navigation bar of the application. On the left is the logo for 'Ingénieurs sup Galilée' with a portrait of Galileo Galilei. The main navigation area contains: 'Accueil - Sup-Galilée', 'Bienvenue grr Administrateur', 'Gérer mon compte' with flags for France, Germany, UK, and Spain, and 'Se déconnecter'. A date selector shows '27 mai 2012' and 'Aujourd'hui', with buttons for 'Journée', 'Semaine', and 'Mois'. An 'Administration' section includes a 'Lancer une sauvegarde' button and '1 personne connectée'. On the right, the date and time are 'Dim. 27 Mai. 2012 - 19:08:13', with links for 'Aide', 'Recherche - Rapports - Stats.', and 'GRR 1.9.7e - Contacter le support technique'.

### Ajouter une réservation



The screenshot shows the 'Ajouter une réservation' form. It includes a dropdown for 'Administrateur grr' with a 'Définir par défaut' button. The 'Brève description \*' field contains 'Blabla'. The 'Description complète (facultative)' field contains 'blaba'. The 'Début de la réservation' is set to '27 mai 2012' at '08 : 00'. The 'Durée' is '15 minute(s)' with a checkbox for 'Journée entière (8 H - 19 H 15)'. The 'Ressources' field shows 'F207' with a note 'CTRL + clic souris pour sélectionner plus d'une ressource'. The 'Type \*' dropdown is set to 'Cours' with a 'Définir par défaut' button. A link on the right says '>>>Cliquez ici pour ouvrir les options de périodicité<<<'. At the bottom right are 'Annuler' and 'Enregistrer' buttons.

Figure 14 Effectuer une réservation

## b. Copier/Modifier/Supprimer une réservation

	<p>Accueil - Sup-Gallilée Bienvenue grr Administrateur Gérer mon compte -  Se déconnecter</p>
---	--

[Retour à la page précédente](#)

**Réservation : Blaba**

Description :	blaba
Ressource :	Salle Info - F207
Date de début :	08:00 - dimanche 27 mai 2012
Durée :	15 minute(s)
Date de fin :	08:15 - dimanche 27 mai 2012
Type :	Cours
Réservation effectuée par :	<b>grr Administrateur</b>
Dernière mise à jour :	19:09 - dimanche 27 mai 2012

[Modifier cette réservation](#) - [Copier cette réservation](#) - [Effacer cette réservation](#)

Figure 15 Modifier /Supprimer/Reservation

## c. Réservation en bloc

<p>Réservation au nom de : </p> <p>Administrateur grr <input type="button" value="Définir par défaut"/></p> <p>Brève description * : <input type="text"/></p> <p>Description complète (facultative) : <input type="text"/></p> <p>Début de la réservation : 27 mai 2012 Heure : 08 : 00</p> <p>Durée : 15 minute(s) <input type="checkbox"/> Journée entière (8 H - 19 H 15)</p> <p>Ressources : F206 CTRL + clic souris pour sélectionner plus d'une ressource F207 G210 G212</p> <p>Type * : (choisissez) <input type="button" value="Définir par défaut"/></p>	<p><a href="#">&gt;&gt;&gt;Cliquez ici pour fermer les options de périodicité&lt;&lt;&lt;</a> </p> <p>Type de périodicité :</p> <p><input checked="" type="radio"/> Aucune <input type="radio"/> Chaque jour <input type="radio"/> chaque semaine <input type="text"/> <input type="radio"/> Chaque mois, la même date <input type="text"/> <input type="radio"/> Chaque année, même date</p> <p>Date de fin de périodicité : 27 mai 2012</p> <p><input type="button" value="Annuler"/> <input type="button" value="Enregistrer"/></p>
--	---

Figure 16 Réservation par bloc

#### d. Ajout manuel de logiciel

Ce formulaire permet l'ajout manuel de logiciel dans la base de données. Pour simplifier nous avons choisit de ne retenir que 3 champs. Nom de la salle, nom du logiciel et la version.

#### Ajout\_logiciel

<b>Salle</b>	F207
<b>Nom Logiciel</b>	Eclipse
<b>Version logiciel</b>	2.0

#### Ajout\_Ordinateur

<b>Nom</b>	Dell Optiplex
<b>Salle</b>	F207

Figure 17 Ajout Matériel

### e. Ajout manuel d'ordinateur

Nous avons choisit d'ajouter ce formulaire pour pouvoir lister et répertorié l'ensemble des ordinateur d'une salle. Dans ce cas nous considérons qu'un logiciel doit être installé sur chaque machine de la salle.

### f. Afficher planning

Cette page permet l'affichage de planning du jour. La loupe permet l'affichage de la description de la salle.

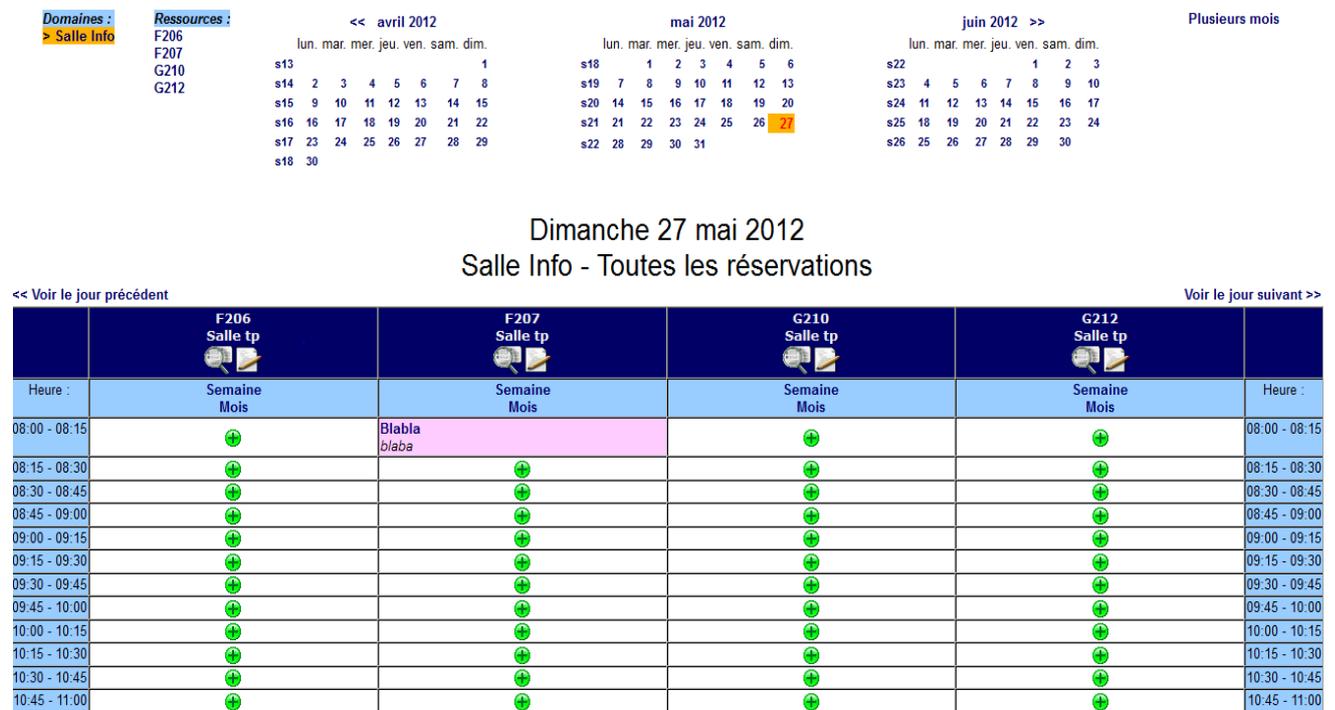


Figure 18 Planning

## g. Consultation Matériel disponible

Cette page permet d'afficher une description du matériel disponible dans une salle donnée.

**Domaine : Salle Info**  
**Modifier la ressource**

Nom :	
Description :	
Ordre d'affichage :	
Qui peut voir cette ressource (et éventuellement N'importe qui allant sur le site même sans être connecté)	
Déclarer cette ressource temporairement indisponible <i>La restriction ne s'applique pas aux gestionnaires de la ressource</i>	
Montrer la fiche de présentation de la ressource	
Choisir une image de la ressource pour la fiche de présentation	
Supprimer l'image actuelle de la ressource	
Afficher la description complète dans le détail	
Pour une nouvelle réservation ou modification	
Nombre de personnes maximum autorisées	
Nombre max. de réservations par utilisateur	
Nombre maximal de jours au-delà duquel la réservation est annulée <b>Exemple :</b> une valeur égale à 30 signifie que la réservation est annulée 30 jours après la date de la réservation <i>Cette limitation ne touche pas les gestionnaires de la ressource ainsi que les administrateurs du domaine.</i>	

GRR (Gestion et Réserveion de Ressources) : Sup-Gallilée - Mozilla Firefox  
127.0.0.1/grr197e1/view\_room.php?id\_room=2

**Ressource : F206  
(Salle Info)**

**Commentaire :**  
Salle tp

**Matériel:**  
Pas d'imprimante disponible  
Pas de retroprojecteur disponible  
Nombre de machines : 0

Pas d'image disponible

Parcourir...

de la réservation  
heure de fin de réservation

Figure 19 Afficher description

### 3. Difficultés rencontrées

#### **Difficulté d'installation du client OCS Inventory**

En effet pour installer ce client il nous fallait installer des machines virtuelles. Malheureusement faute de temps et de matériel dont nous n'avaons pas eu le temps de l'implémenter.

#### **Complexité du code GRR**

En effet le code de GRR est assez dense et pas très bien indenté. Il était difficile d'y apporter des modifications sans perturber tout le système.

#### **Difficulté d'installation sous linux**

## **VII. CONCLUSION ET PERSPECTIVES**

Ce projet nous a permis de mieux appréhender la gestion de projet et de mettre en application nos compétences en PHP et MySQL. De plus il nous a appris l'importance de développer des logiciels Open Source. Aussi humainement nous avons appris à travailler avec des personnes aux caractères différents. Pour finir outre le fait que nous avons pu implémenter une bonne partie des fonctionnalités attendues par nos clients et nous espérons développer la mise à jour automatique de logiciel via le client OCS INVENTORY .

## I. NETOGRAPHIE

*SiteDuZero.com/*

*Developpez.com/*

*Wikipédia.com/*

**grr.mutualibre.org/**

