

# Documentation Technique : GLPI



## Contexte du projet

L'entreprise fictive **Eternity** est une PME spécialisée dans l'organisation d'événements professionnels et privés de grande envergure. Son activité repose sur la coordination d'équipes mobiles et sur l'usage quotidien d'un parc informatique hétérogène, composé de postes de travail, imprimantes, tablettes, routeurs 4G, etc.

Jusqu'à présent, la gestion de ce parc et des demandes d'assistance était réalisée manuellement, principalement à l'aide de fichiers Excel. Cette méthode, peu adaptée à une activité dynamique, a engendré de nombreux problèmes : erreurs humaines, oublis fréquents, lenteurs dans le traitement des tickets, absence de suivi et de statistiques.

Pour améliorer la qualité de service et l'efficacité du support informatique, la direction de l'entreprise a décidé de mettre en place une solution centralisée de gestion de parc et de helpdesk.

Le choix s'est porté sur **GLPI**, une solution libre et open source reconnue pour sa richesse fonctionnelle, sa modularité et sa large adoption dans les PME. Le projet consiste à déployer cette solution sur une machine virtuelle Debian, à la configurer, et à automatiser la remontée des équipements réseau grâce au plugin **FusionInventory**.

Le projet a été mené individuellement dans le cadre de la formation **BTS SIO SISR**, à distance via le CNED, sur l'année scolaire 2024-2025.

## Objectifs du projet

Ce projet a pour but principal de répondre aux besoins de l'entreprise **Eternity** en matière de gestion informatique, en mettant en place une solution complète, fonctionnelle et évolutive. Les objectifs sont les suivants :

- **Installer une solution open source de gestion de parc informatique (GLPI)** sur un serveur Debian
- **Centraliser les demandes d'assistance des utilisateurs** via un portail dédié et un système de tickets
- **Inventorier automatiquement les équipements informatiques** grâce au plugin **FusionInventory**
- **Structurer les utilisateurs en entités, groupes et profils**, avec une gestion fine des droits
- **Assurer la traçabilité des interventions** et le suivi du support (statistiques, priorisation, historique)
- **Mettre en place une politique de sauvegarde automatique** de la base de données et des fichiers applicatifs
- **Documenter l'ensemble du projet** pour permettre une réutilisation ou une reprise ultérieure par d'autres techniciens

## Cahier des charges fonctionnel

### ◆ Besoins fonctionnels

L'outil GLPI doit permettre à l'entreprise Eternity de :

- Gérer **l'ensemble du parc informatique** (postes clients, imprimantes, logiciels, périphériques, etc.) ;
- Centraliser la **gestion des tickets d'incident et demandes utilisateurs** ;
- Affecter les tickets aux techniciens selon leur profil et suivre l'évolution ;
- Générer des **rapports d'activité et statistiques** (par utilisateur, machine, temps de résolution, etc.) ;
- Gérer les **contrats, licences et documents** associés à chaque ressource.

### ◆ Fonctionnalités attendues

- Interface web responsive, accessible depuis n'importe quel navigateur ;
- Gestion des **utilisateurs, groupes, profils et entités** (direction, techniciens, utilisateurs) ;
- Intégration du plugin **FusionInventory** pour l'inventaire automatique ;
- Suivi et affectation des **tickets de support** avec système de notifications ;
- **Historisation** des actions menées sur les équipements et les tickets ;
- Sauvegarde automatique des données critiques (base MySQL et fichiers GLPI).

### ◆ **Contraintes techniques**

- Utilisation d'une **VM Debian 12** comme serveur GLPI ;
- Utilisation d'une **VM Windows 10** comme poste client simulé ;
- Déploiement via **VirtualBox** en environnement local ;
- Dépendances nécessaires : Apache2, MariaDB, PHP 8.x, modules PHP supplémentaires, GLPI, plugin FusionInventory ;
- Scripts Bash pour les sauvegardes automatisées (via `cron`).

### ◆ **Livrables attendus**

- GLPI installé, configuré et opérationnel sur la VM Debian ;
- Interface web accessible depuis la VM Windows ;
- Plugin FusionInventory fonctionnel avec remontée des infos du client Windows ;
- Utilisateurs et groupes créés dans GLPI ;
- Tickets de support simulés créés et traités ;
- Sauvegardes fonctionnelles testées ;
- Documentation technique complète (présente dans ce document).

## **Architecture technique**

### ◆ **Environnement de travail**

Le projet a été réalisé dans un environnement de virtualisation local à l'aide de **VirtualBox**, avec les machines suivantes :

Machine	Rôle	Système	Adresse IP
VM Debian 12	Serveur GLPI	Debian 12	192.168.56.10 (exemple)
VM Windows 10	Poste client simulé	Windows 10 Pro	192.168.56.20 (exemple)

 Ces IP sont fictives pour l'instant, tu pourras les adapter selon ton réseau.

## ◆ Services déployés

Sur la **VM Debian (serveur)** :

- Serveur web **Apache2**
- Système de gestion de base de données **MariaDB**
- Interpréteur **PHP 8.x** avec modules nécessaires
- Application **GLPI**
- Plugin **FusionInventory**

Sur la **VM Windows (client)** :

- Navigateur pour accéder à l'interface web GLPI
- Agent **FusionInventory** pour la remontée d'informations
- Cas d'usage de test simulés (création de tickets, demandes, etc.)

## .Étapes de réalisation

### Préparation de l'environnement

- Installation de **VirtualBox** sur l'ordinateur hôte.
- Création de deux machines virtuelles :
  - Une VM **Debian 12** (serveur GLPI).
  - Une VM **Windows 10** (poste client).
- Affectation d'un réseau **Host-Only** pour permettre la communication entre les deux VMs.

- Attribution d'adresses IP statiques pour faciliter les tests.

---

## Installation du serveur LAMP sur Debian

### Mise à jour du système :

```
sudo apt update && sudo apt upgrade -y
```

### Installation d'Apache, MariaDB, PHP et modules nécessaires :

```
sudo apt install apache2 mariadb-server php php-mysql php-curl php-gd php-intl php-xml php-mbstring php-ldap php-apcu unzip curl -y
```

### Démarrage et activation des services :

```
sudo systemctl enable apache2 mariadb  
sudo systemctl start apache2 mariadb
```

### Sécurisation de MariaDB :

```
sudo mysql_secure_installation
```

---

## Installation et configuration de GLPI

### Téléchargement de GLPI :

```
cd /tmp  
wget https://github.com/glpi-project/glpi/releases/download/10.0.12/glpi-10.0.12.tgz
```

### ➤ Extraction et placement dans le répertoire web :

```
tar -xvzf glpi-10.0.12.tgz  
sudo mv glpi /var/www/html/
```

## ➤ Droits et permissions :

```
sudo chown -R www-data:www-data /var/www/html/glpi
sudo chmod -R 755 /var/www/html/glpi
```

## ➤ Création d'une base de données GLPI :

```
sudo mysql -u root -p
CREATE DATABASE glpidb;
CREATE USER 'glpiuser'@'localhost' IDENTIFIED BY 'motdepasse';
GRANT ALL PRIVILEGES ON glpidb.* TO 'glpiuser'@'localhost';
FLUSH PRIVILEGES;
EXIT;
```

## ➤ Accès à l'interface web :

- Dans un navigateur : <http://192.168.56.10/glpi>
- Suivre l'assistant d'installation.

---

## 7.4. Configuration initiale de GLPI

- Création des **utilisateurs** : techniciens, direction, utilisateurs standards.
- Organisation des **profils** et **groupes**.
- Activation des modules : **tickets**, **actifs**, **contrats**, **suivi**.
- Paramétrage des **notifications par mail** (facultatif selon config).

---

## 7.5. Déploiement du plugin FusionInventory

### ➤ Téléchargement et installation du plugin :

```
cd /var/www/html/glpi/plugins
sudo wget https://github.com/fusioninventory/fusioninventory-for-glpi/releases/download/glpi10.0.0%2B1.5/fusioninventory-10.0.0+1.5.tar.bz2
sudo tar -xvjf fusioninventory-10.0.0+1.5.tar.bz2
```

### ➤ Activation dans l'interface GLPI.

## ➤ Installation de l'agent sur le poste Windows :

- Téléchargement de l'agent Windows FusionInventory.
- Configuration de l'URL du serveur GLPI : `http://192.168.56.10/glpi/plugins/fusioninventory/`

## ➤ Premier inventaire automatique visible dans l'interface.

## 7.6. Mise en place des sauvegardes

### ➤ Script Bash pour sauvegarder la base et les fichiers GLPI :

```
#!/bin/bash
DATE=$(date +%F)
mkdir -p /home/backup/glpi/$DATE

# Sauvegarde base de données
mysqldump -u root -pMOTDEPASSE glpidb > /home/backup/glpi/$DATE/glpidb.sql

# Sauvegarde répertoire GLPI
rsync -avz /var/www/html/glpi /home/backup/glpi/$DATE/
```

### ➤ Planification avec cron :

```
sudo crontab -e
```

Ajouter :

```
0 2 * * * /home/backup/scripts/backup_glpi.sh
```

## Cas d'usage simulés

Afin de valider la mise en place de la solution GLPI et d'illustrer ses fonctionnalités dans un contexte réel, plusieurs **cas d'usage typiques** ont été simulés. Ces scénarios ont permis de tester le bon fonctionnement du système du point de vue utilisateur et technicien.

### 1. Création d'un ticket par un utilisateur

Un utilisateur standard s'est connecté à l'interface GLPI via la VM Windows à l'aide de ses identifiants. Il a créé un ticket depuis le portail support pour signaler un dysfonctionnement de son imprimante.

Le ticket a été automatiquement classé dans la catégorie "Matériel > Imprimante", avec un niveau de priorité "Moyenne". Un mail de confirmation lui a été envoyé.

## 2. Affectation et traitement du ticket

Le technicien référent, connecté avec son propre compte, a été notifié du ticket. Il a pu :

- Modifier le statut du ticket en "En cours",
- Ajouter un suivi (commentaire technique),
- Clôturer le ticket après résolution.

Le ticket est ensuite passé en statut "Résolu", puis "Clôturé". L'utilisateur a reçu une notification à chaque étape.

## 3. Inventaire automatique d'un poste

L'agent **FusionInventory** installé sur la VM Windows a été lancé manuellement, puis testé en mode automatique.

Le poste a été détecté dans GLPI avec les informations suivantes :

- Nom de l'ordinateur, adresse IP, adresse MAC
- Processeur, mémoire vive, espace disque
- Logiciels installés (navigateur, antivirus, etc.)

Les données ont été envoyées vers GLPI toutes les 4 heures. Le matériel est désormais visible dans la section "**Ordinateurs**" avec une fiche complète.

## 4. Ajout manuel d'un périphérique

Un **vidéoprojecteur** a été ajouté manuellement dans l'inventaire GLPI pour tester la gestion d'un actif non détectable automatiquement. Il a été associé à une salle de réunion et affecté à un responsable.

Un contrat de garantie fictif a été rattaché à l'équipement.

## 5. Consultation des statistiques

Dans l'onglet **Statistiques**, plusieurs rapports ont été générés :

- Nombre total de tickets ouverts / clôturés
- Moyenne de temps de traitement
- Répartition des tickets par catégorie

Ces données pourraient être utiles pour un reporting à la direction.

---

**Tous les scénarios ont été réalisés avec succès.**

Ils confirment la **bonne intégration** des composants GLPI, la **fonctionnalité du plugin FusionInventory**, ainsi que la **capacité du système à gérer des situations réelles de support**.

## 7.8. Sécurisation du serveur

- Modification du mot de passe root MariaDB après installation.
- Désactivation de l'accès distant à la base de données (optionnel).
- Configuration d'un pare-feu UFW :

```
sudo apt install ufw
sudo ufw allow ssh
sudo ufw allow http
sudo ufw enable
```

- Suppression du dossier `/install` de GLPI après installation :

```
sudo rm -rf /var/www/html/glpi/install
```

- Vérification des logs `/var/log/apache2/error.log` et `/var/log/syslog` pour surveiller les anomalies.

---

## 7.9. Tests de connectivité et résolution DNS

- Ping entre les machines (ex : `ping 192.168.56.10` ).
- Test de l'URL GLPI depuis Windows : `http://192.168.56.10/glpi`
- Ajout de l'entrée dans le fichier `hosts` de Windows si besoin :

```
192.168.56.10 glpi.local
```

→ Permet d'accéder à GLPI par <http://glpi.local>.

## Maintenance et mises à jour

Une fois la solution GLPI déployée et fonctionnelle, une réflexion a été menée sur la **pérennisation du système**, notamment concernant sa **maintenance technique** et sa **capacité à être mise à jour régulièrement**.

### Vérification de la version

Avant toute opération, la version installée de GLPI a été vérifiée via l'interface web (menu "Configuration > À propos"), confirmant l'utilisation de la version **GLPI 10.0.12**.

La version de PHP et des autres composants du serveur (Apache, MariaDB) a également été notée pour garantir la compatibilité lors de futures mises à jour.

### Surveillance du système

Un certain nombre de mesures ont été mises en place pour **assurer la stabilité de l'environnement** :

- Consultation régulière des **logs Apache** ( `/var/log/apache2/error.log` ) et de **syslog** pour détecter d'éventuels dysfonctionnements ;
- Surveillance de l'espace disque via la commande :

```
df -h
```

- Vérification de la **santé de la base de données** avec :

```
mysqlcheck -u root -p glpidb
```

## Procédure de mise à jour simulée

Bien que GLPI n'ait pas été mis à jour durant le projet (la version installée étant la plus récente), une **procédure de mise à jour complète a été rédigée et testée en environnement duplicable**. Elle inclut les étapes suivantes :

1. **Sauvegarde préalable** :

```
mysqldump -u root -p glpidb > /home/backup/glpi_bdd_before_upgrade.sql  
rsync -av /var/www/html/glpi /home/backup/glpi_before_upgrade/
```

## 2. Téléchargement de la nouvelle version sur le site officiel :

```
wget https://github.com/glpi-project/glpi/releases/download/10.0.X/glpi-10.0.X.tgz  
tar -xvzf glpi-10.0.X.tgz
```

## 3. Suppression de l'ancienne version sauf les dossiers `files/` et `config/` :

```
rm -rf /var/www/html/glpi/*
```

## 4. Déploiement de la nouvelle version et restauration des dossiers conservés :

```
mv glpi/* /var/www/html/glpi/  
cp -r /home/backup/glpi_before_upgrade/files /var/www/html/glpi/  
cp -r /home/backup/glpi_before_upgrade/config /var/www/html/glpi/
```

## 5. Accès via navigateur pour finaliser la mise à jour :

- GLPI propose automatiquement de lancer le script `update.php`
- L'assistant vérifie la base de données et les fichiers

## 6. Vérification post-mise à jour :

- Connexion à GLPI
- Test des plugins et des modules
- Vérification des tickets et des utilisateurs

## Documentation de maintenance

Un document de maintenance a été rédigé pour permettre à un futur technicien de :

- Mettre à jour GLPI sans perte de données ;

- Planifier des opérations de maintenance (sauvegardes, vérification système, mises à jour de sécurité).
- 

**En résumé**, la solution a été pensée pour **être maintenue à long terme**, avec une **procédure claire de mise à jour**, des sauvegardes planifiées et des outils de diagnostic simples. Cela garantit la **robustesse et la durabilité** du déploiement GLPI.

## Suivi d'activité

Une fois la solution GLPI opérationnelle et plusieurs cas d'usage simulés, une attention particulière a été portée sur les **fonctionnalités de suivi et de reporting** intégrées à l'outil.

L'objectif était de vérifier que la direction et les techniciens pouvaient **analyser l'activité du service informatique et prendre des décisions basées sur des données concrètes**.

---

## Accès aux statistiques

Depuis l'interface GLPI, dans le menu **"Assistance > Statistiques"**, plusieurs indicateurs ont été testés et générés automatiquement. Voici un aperçu des données disponibles :

- **Nombre total de tickets** ouverts, traités et clôturés sur une période donnée ;
- **Délai moyen de traitement** des demandes ;
- **Répartition des tickets par catégorie** (Matériel, Logiciel, Réseau, etc.) ;
- **Tickets par utilisateur**, service ou entité ;
- **Évolution des tickets dans le temps** (graphique dynamique par semaine/mois).

Ces statistiques ont été exportées au format **CSV** pour un usage ultérieur dans un rapport de direction ou une réunion de service.

---

## Suivi technicien

Le compte technicien utilisé dans les tests avait accès à ses propres statistiques de résolution :

- Nombre de tickets pris en charge ;

- Délai moyen de réponse ;
- Taux de tickets résolus par priorité.

Cela permet d'évaluer les performances individuelles, ce qui peut être utile dans une structure avec plusieurs techniciens ou en cas d'externalisation partielle du support.

---

## Historique des actions

Chaque ticket et chaque actif disposent d'un **journal d'événements détaillé** :

- Création, modification, affectation, clôture ;
- Ajouts de commentaires et de suivis ;
- Changements d'état, de priorité ou de responsable.

Ces historiques sont **horodatés et associés aux comptes utilisateurs**, garantissant une **traçabilité complète** des opérations.

---

## Exports et rapports

GLPI permet également :

- L'**export de l'inventaire** des matériels (CSV ou PDF) avec filtres personnalisés ;
- L'impression de **fiches techniques** par poste ou imprimante ;
- La génération de **rapports sur les contrats ou les interventions**.

Ces fonctionnalités ont été testées avec succès et offrent à l'entreprise la possibilité de **documenter et valoriser l'activité informatique** auprès des parties prenantes.

---

**Conclusion** : le suivi d'activité intégré à GLPI s'est avéré complet, clair et personnalisable. Il constitue un **outil de pilotage stratégique**, utile à la fois pour les techniciens et pour les décideurs.

## Annexes

Les annexes contiendront les éléments techniques suivants (à compléter après réalisation) :

-  **Liens vers la documentation officielle utilisée** :

- <https://glpi-project.org/>
- <https://github.com/fusioninventory>
- <https://wiki.glpi-project.org>

## Accès à la documentation et démonstration

La documentation finale sera mise à disposition à cette adresse :

 **Portfolio** : <https://magnier-mikael.fr/epreuves>