

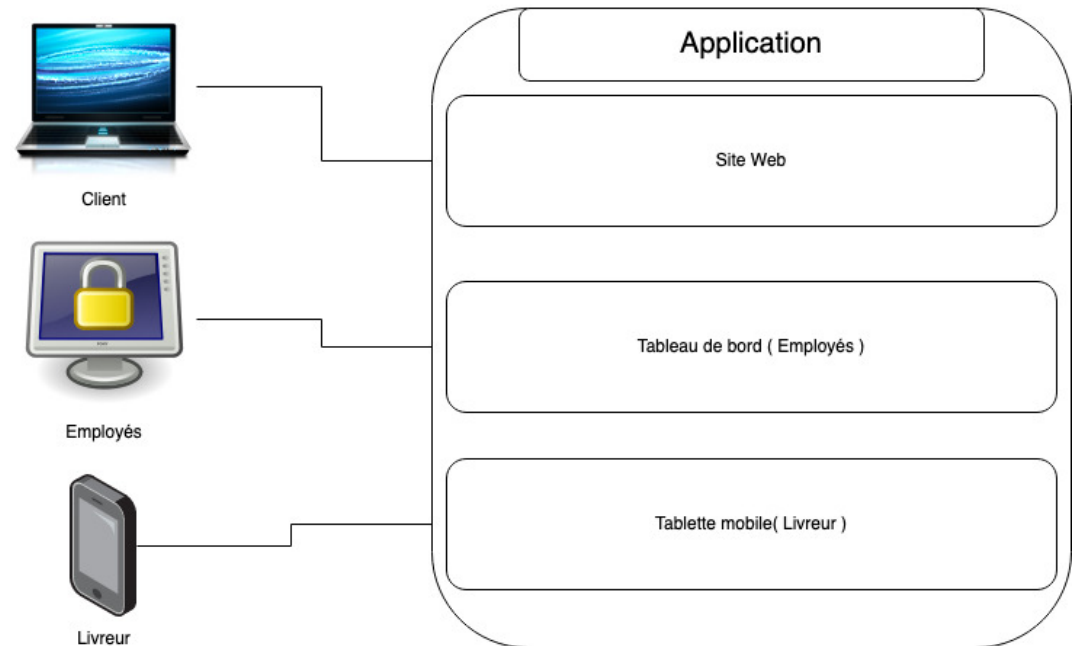
# Solution fonctionnelle

## Contexte :

« OC Pizza » est un jeune groupe de pizzeria en plein essor. Le groupe est spécialisé dans les pizzas livrées ou à emporter. Le système informatique actuel ne correspond plus aux besoins du groupe car il ne permet pas une gestion centralisée de toutes les pizzerias.

## Objectifs :

- Être plus efficace dans la gestion des commandes, de leurs réceptions à leurs livraisons en passant par leurs préparations.
- Suivre en temps réel les commandes passées, en préparation et en livraison.
- Suivre en temps réel le stock d'ingrédients restant pour savoir quelles pizzas peuvent encore être réalisées.
- Proposer un site internet pour que les clients puissent :
  - passer leurs commandes, en plus de la prise de commande par téléphone ou sur place;
  - payer en ligne leur commande s'ils le souhaitent - sinon, ils paieront directement à la livraison;
  - modifier ou annuler leur commande tant que celle-ci n'a pas été préparée;
- Proposer un aide-mémoire aux pizzaiolos indiquant la recette de chaque pizzas.

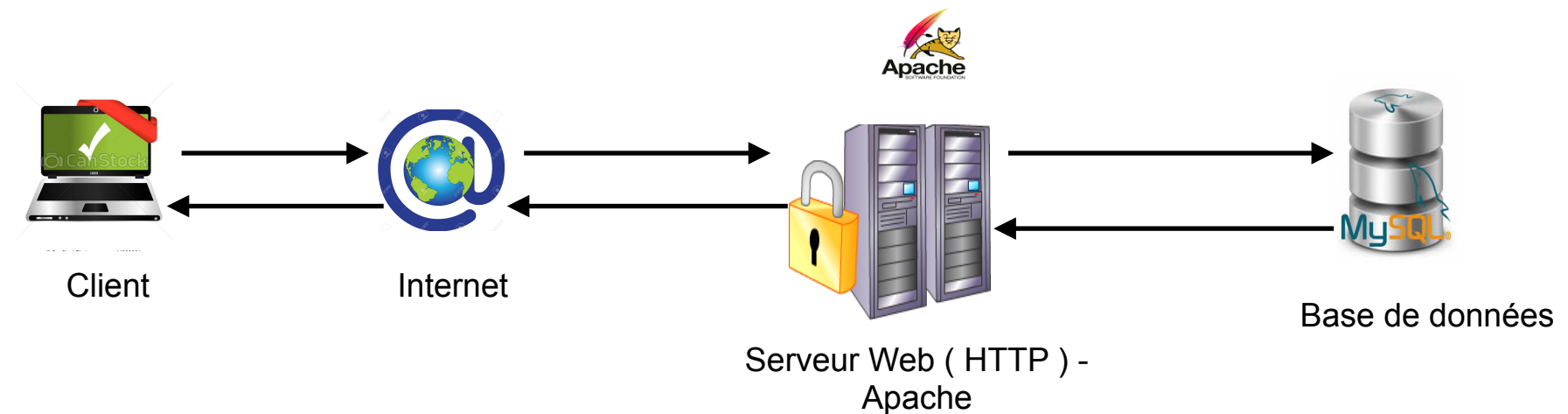


Site Web : Site internet qui permet de consulter les pizzas et passer une commande en ligne, etc...

Tableau de bord : Écran qui permet aux employés d'effectuer les tâches qui leur sont affectées.

Tablette mobile : Tablette qui permet au livreur de notifier le statut de sa livraison.

## Solution technique



**La solution technique est basée sur une combinaison Linux – Java/JEE - Apache Tomcat – MySQL**

### 1. Serveur Linux

Le serveur sur lequel la plateforme sera hébergée est un environnement [Linux – Debian](#). Cette distribution est stable, gratuite et proposée par la plupart des hébergeurs du marché (OVH, Amazon Web Service, Google Cloud Platform).

Sur ce serveur Linux, nous aurons les composants suivants :

- a. **Apache Tomcat** : [Apache Tomcat](#) est un serveur d'application qui va contenir l'application web (fichier .war), cœur de la plateforme web. En plus d'être un conteneur d'application, Apache Tomcat est un serveur Web et il répondra directement aux requêtes Web.

- b. **MySQL** : [MySQL](#) est un système de gestion de base de données (SGBD) open-source et gratuit. Il est facile à installer, configurer et maintenir. Il permet de stocker et d'organiser les données de la plateforme web.
- c. **JAVA/JEE** : écosystème avec le langage Java, les pages JSP/Servlet pour écrire des pages Web dynamiques. C'est un écosystème qui fournit un langage de programmation (Java) puissant et qui permet de communiquer avec le serveur MySQL. J'ai choisi l'écosystème Java/JEE parce qu'il est celui que j'ai étudié dans la formation Openclassrooms. Je suis donc apte à proposer un système d'information bâti sur cette technologie.

Tous les composants seront situés au début sur la même machine. Au fil du temps, il sera possible de mettre en place Apache Tomcat et l'application sur une machine et la base de données sur une 2ème machine.

Cette solution technique est open-source, simple et facile d'utilisation. J'ai choisi ce modèle pour ces raisons, la maintenance se fera facilement car ce sont des systèmes qui ont fait leurs preuves avec une large communauté d'utilisateurs.

## 2. **Déploiement de l'application, connexion au serveur**

Pour déployer l'application, il faut :

- installer un serveur FTP ( ProFTP par exemple ) sur le serveur. On utilisera Filezilla pour se connecter au serveur et déposer les fichiers que l'ont souhaite.

- installer Openssh sur la machine et ensuite avec un client SSH comme Putty, on se connecte de manière sécurisée pour exécuter les tâches administrative habituelles ( démarrer/arrêter Apache Tomcat, MySQL, vérifier les loges, etc... )

