**Contexte de notre application**

Notre projet se pose comme une initiative proposant l’étude, la conception ainsi que la réalisation d’une plate‐forme micro-services de commerce électronique destinée à la société EURL PRO VISION SECURITY SYSTEMS.

1. **Les différents services de l’application :**

L’application va proposer trois types de services :

* **le service commandes** : est un système d’automatisation et de suivi de l’ensemble des demandes du client, ce service permettra donc au client d’effectuer ses achats en ligne et par conséquent augmenter le chiffre d’affaire de la société et en même temps faciliter les tâches pour les employées
* **le service catalogue** : consiste à exposer le catalogue des produits et des services de la société en donnant a ses clients la possibilité d’avoir une idée plus ou moins général sur chaque produit.que ce soit pour ses clients en business to business ou encore pour ses client business to client .
* **le service Business Intelligence** : qui a pour but de facilite la prise de décisions au sein de l’entreprise.

1. **Architecture de l’application :**

Pour réaliser notre projet on a [opté pour une architecture micro-services qui favorise la réalisation des applications modulaires c'est-à-dire constituées de petits composants appelés « services ».la figure ci dessous montre l’architecture globale de notre application :](http://blog.ippon.fr/2021/01/18/quand-opter-pour-une-architecture-de-micro-services/)

BDD

BDD

**Service catalogue**

**Service commandes**

**Service Business Intelligence**

Quelques Avantages de cette architecture

* Fonctionnalité modulaire, modules indépendants.
* pas de liaison à des langages et technologies spécifiques.
* intégration facile de nouveau services et possibilité de réutilisation.
* plus rapide et facile à mettre à jour.
* tolérance aux pannes et facilité de maintenance.

1. **Environnement de développement :**

**Laravel :** Framework php, coté serveur

**React :** librairie javascript, coté client

**Postgresql :** système de gestion de base de données relationnelle et objet

**Redis :** base de données Nosql pour la mise en cache

**Doker :** une plateforme permettant de lancer certaines applications dans des conteneurs logiciels.

**Linter :** pour la qualité du code

### LES BESOINS FONCTIONNELS :

### Le système doit permettre :

### Aux visiteurs :

### De feuilleter le catalogue des produit

### Créer un compte

### Au client :

### La gestion de son panier d’achat:

### -l’ajout d’un produit a son panier,

### -la suppression d’un produit de son panier

### -la validation des achats de son panier

### -vider le panier

### La consultation de l’ensemble des produits

### La recherche d’un produit

### La consultation de ses commande

### La gestion de son profil

### L’authentification

### A l’administrateur de :

### Gérer ses produits :

### - l’ajout et la suppression d’un produit

### - la modification des caractéristiques d’un produit

### -consulter la liste des produit

### La gestion des commandes :

### - confirmation d’une commande d’un client

### - consultation des commandes

### - suppression d’une commande

### Gestion de stock des produits

### -supprimer du stock

### -ajout au stock

### -modifier le stock

### - consulter les produits du stock

### L’authentification

### Gestion des clients :

### -valider le compte d’un client

### -Supprimer un client

### -consulter la liste des clients

### LES BESOINS NON FONCTIONNELS :

### Dans le cadre de ce travail, l'application devra accomplir un ensemble de caractéristiques qui sont :

### Gestion de la mémoire pour éviter les bugs et l’indisponibilité de l’application.

### Optimisation applicatif : mise en cache pour avoir de meilleurs temps de réponse

### L’application doit se reposer sur l’architecture micro-services

### **L’application doit** être hautement sécurisée **et performante**

### L’application doit être extensible, c'est-à-dire qu'il pourra y avoir une possibilité d'ajouter ou de modifier de nouvelles fonctionnalités