



## **Réalisation d'application**

### **Gestion des stocks**

**Groupe 6:**

- BERRAI Dyhia N°15608005**
- HUARD Alicia N°15601834**

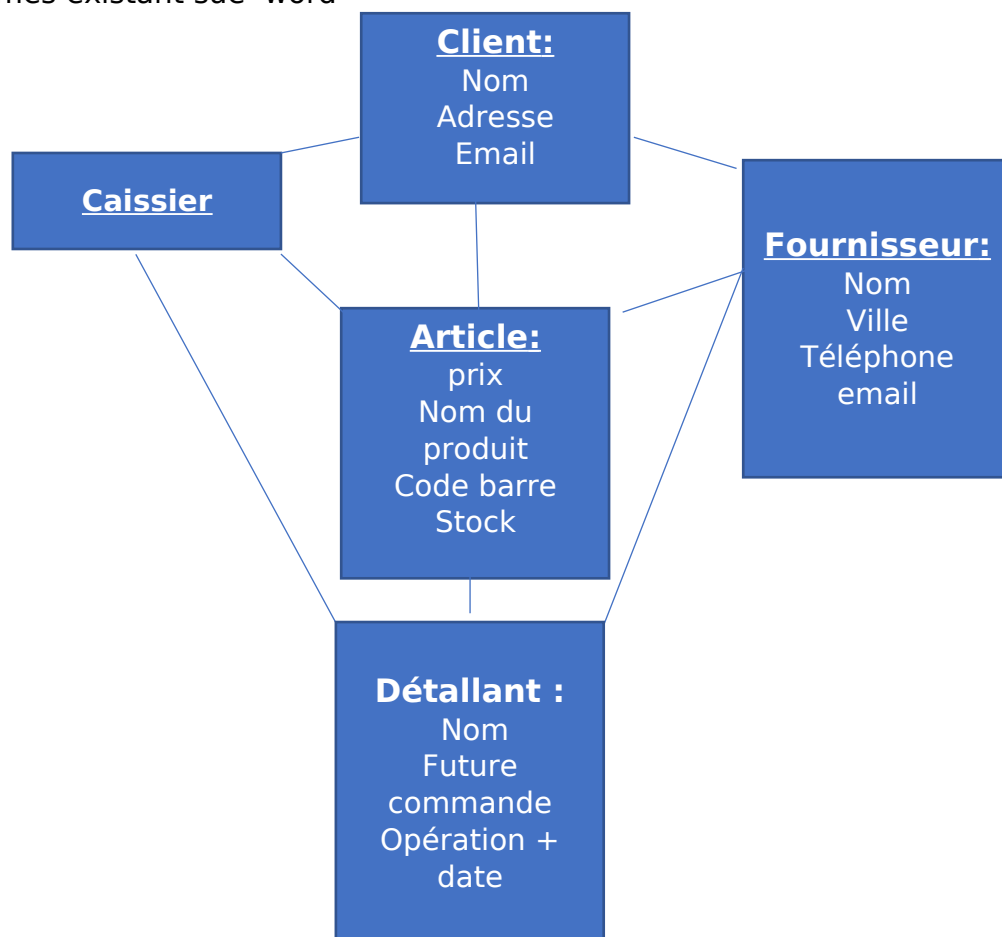
# Ce document est une suite de l'itération II.

## Etape 2 : Etudes du modèle statique

### Diagramme des classes

Le diagramme de classes est un schéma utilisé en génie logiciel pour présenter les classes et les interfaces des systèmes ainsi que les différentes relations entre celles-ci

Suite a des soucis d'installation du logiciel Modelio on a du le faire en utilisant les formes existant sue word



#### La classe article :

Permet la création d'objets "article".

#### La classe client :

Se résume à acheter un article et le retourner.

## Fournisseur :

Est géré par le détaillant, permet de géré l'entrée des articles

## Caissier :

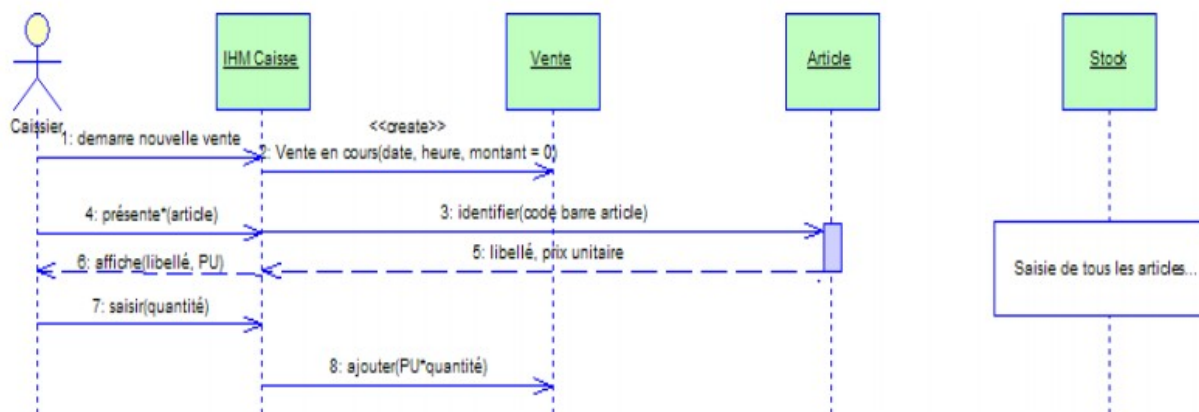
Cette class permet géré l'entrée sortie des articles

## Le détaillant :

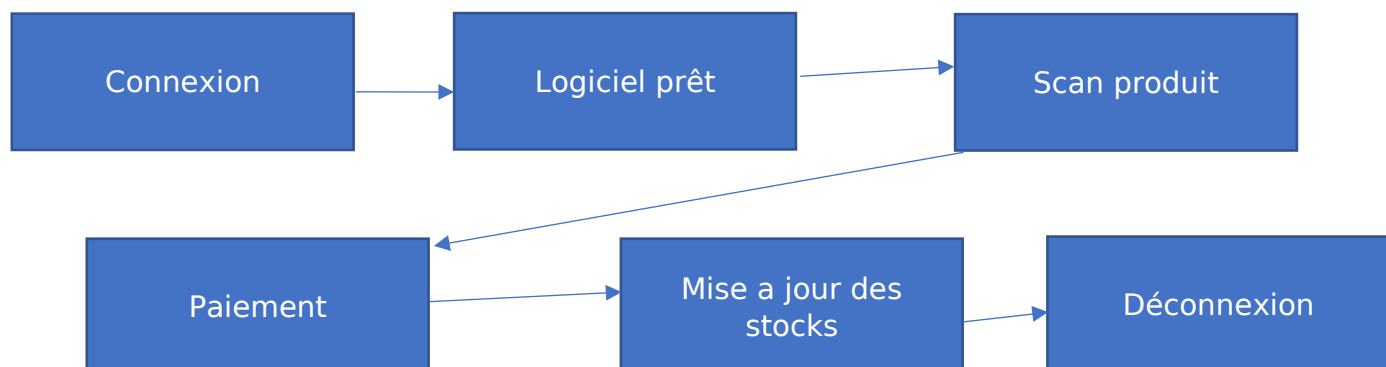
La classe détaillant permet de gérer toutes les fonctions pour la gestion du stock, et les fournisseurs.

## Etape 3 : Étude du modèle dynamique

### Diagramme de séquence



### Diagramme d'état :



Dans un premier temps le caissier va se connecter à l'aide d'un mot de passe ou d'un code utilisateur.

Une fois l'ouverture réussi le caissier pourra commencer à scanner les articles client, à la fin il va rediriger le client vers le paiement.

Ensuite le client paye ses articles.

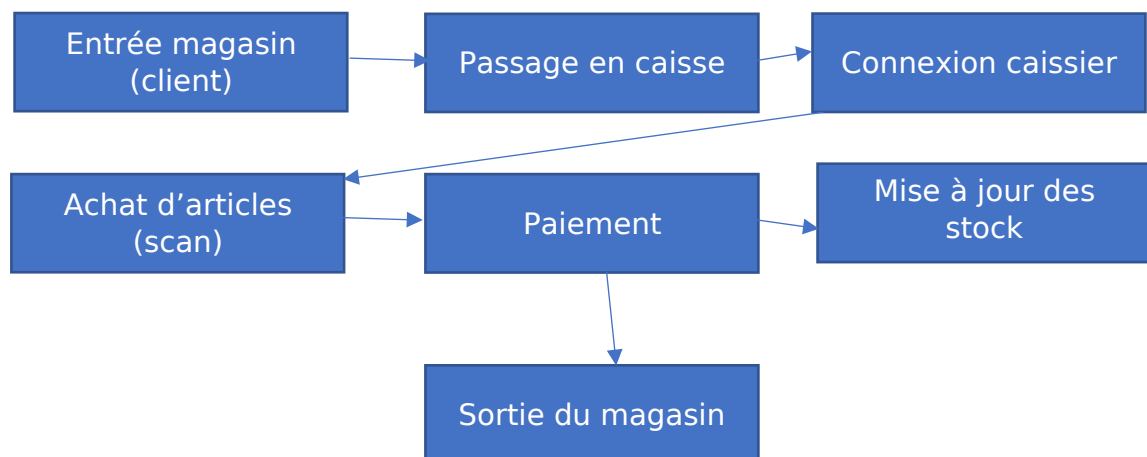
Si paiement réussi, les stock seront mis à jour automatiquement.

Si aucun client n'est en attente, le caissier pourra mettre fin a sa session sans se déconnecter « mettre en veille ».

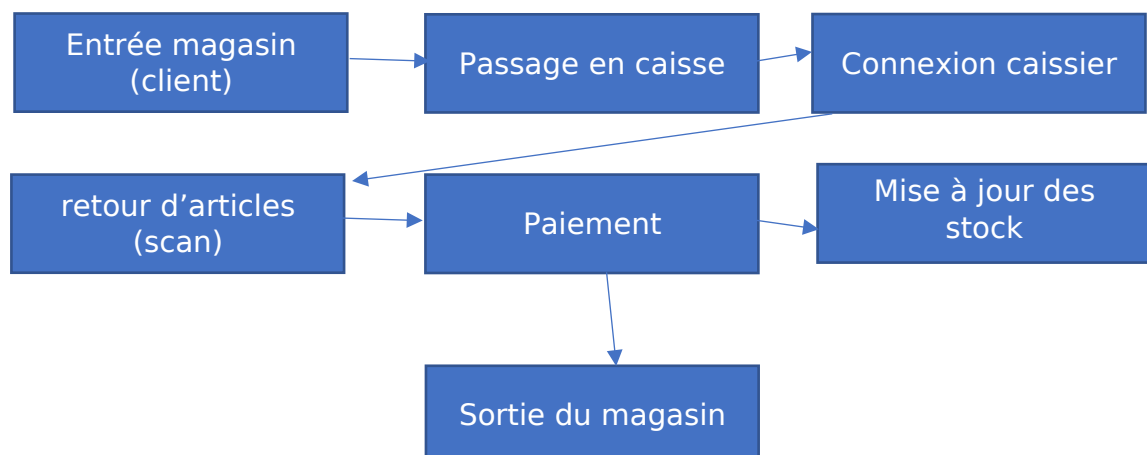
Si retour article, le caissier scan le produit et déclenche le remboursement, les stock seront mis à jour automatiquement.

## Diagramme montrant le parcours client avec le système Client & système

### Cas achat article :



### Cas retour article :



## Itération III & IV:

Nous avons essayé de coder une version du logiciel gestion des stock avec une classe (article, client, détaillant, caissier, caisse) sans oublié que nous nous sommes retrouvés à le faire qu'à deux étant données que nous n'avons plus de nouvelles de deux de nos camarades .

Voir code

```
public class Article {  
  
    protected String    nom;  
    protected int       codeBarre;  
    protected int       quantiteEnStock;  
    protected int       prixDevente;  
  
    public Article(String nom, int codeBarre, int quantiteEnStock, int prixDevente) {  
        this.nom = nom;  
        this.codeBarre = codeBarre;  
        this.quantiteEnStock = quantiteEnStock;  
        this.prixDevente = prixDevente;  
    }  
}
```

```
public class Caissier {  
  
    private String name;  
    private String cleId;  
  
    public Caissier(String cleId,String name){  
        this.cleId=cleId;  
        this.name=name;  
    }  
}
```

```

public class Caisse {

    private ArrayList<Double>                historiqueDeCaisse;
    private HashMap<String,Article>         ticketDeCaisse;
    private ArrayList<Integer>              cleIds;

    public Caisse(){
        this.ticketDeCaisse =new HashMap<String,Article>();
        this.historiqueDeCaisse =new ArrayList<Double>();
    }
}

```

```

public class Client {
    public double            solde;
    public String            identifiant;
    public ArrayList<Article> panier;

    public Client(String identifiant,double solde) {
        this.identifiant=identifiant;
        this.solde=solde;
        this.panier = new ArrayList<Article>();
    }
}

```

```

public class Détaillant {

    private String nom;
    private String nomDeMagasin;
    private ArrayList <Article>  listOfArticle;
    private ArrayList <ProviderInfo>  listOfProviders;

    public Détaillant(String nom,String nomDeMagasin) {
        this.nom=nom;
        this.setNomDeMagasin(nomDeMagasin);
        this.listOfArticle=new ArrayList<Article>();
        this.listOfProviders = new ArrayList<ProviderInfo>();
    }

    public Détaillant() {
        this.listOfArticle=new ArrayList<Article>();
        this.listOfProviders = new ArrayList<ProviderInfo>();
    }
}

```