CESI 2023

Rapport Projet

Programmation Orientée Objet

Muriel RAYNAUD, Ghita BENCHEIKH

Une image contenant texte, Police, logo, Graphique

Description générée automatiquementPréparé par : Alexandre ROUSSEL, Sean LAMET et Marine Mazou

**Table des matières**

[**Présentation :** 2](#_Toc152684513)

[**Présentation de l’équipe :** 2](#_Toc152684514)

[**Reformulation du besoin :** 2](#_Toc152684515)

[**Etude du besoin :** 2](#_Toc152684516)

[**Réalisation du Diagramme de cas d’utilisation (Use Case)** 2](#_Toc152684517)

[**Réalisation du Diagramme d’activité** 3](#_Toc152684518)

[**Trame** 4](#_Toc152684519)

[**Réalisation du Diagramme de Séquence** 5](#_Toc152684520)

[**Réalisation du Diagramme de Classes** 7](#_Toc152684521)

[**Réalisation du Dictionnaire de Données** 8](#_Toc152684522)

[**Réalisation du Modèle conceptuel de données (MCD)** 9](#_Toc152684523)

[**Explication de la modélisation des données :** 9](#_Toc152684524)

[**Réalisation du Modèle logique de données (MLD)** 10](#_Toc152684525)

# **Présentation :**

Vous trouverez toutes les informations nécessaires, dans ce document, pour vous donner une idée claire de notre approche, de notre organisation générale pour ce projet.

## **Présentation de l’équipe :**

L’équipe se compose de trois membres : ROUSSEL Alexandre, LAMET Sean et MAZOU Marine. Ensemble, nous sommes convaincus que nous avons les compétences et l’expertises nécessaire pour mener à bien ce projet.

## **Reformulation du besoin :**

Une nouvelle entreprise développe son système d’information. Son cœur d’activité est la vente en ligne de composants électroniques. Vous devez concevoir et réaliser une solution digitalisant certains de ses processus métiers.

Nous avons pour objectif de réaliser une architecture de type client-serveur composée d’une application et d’une base de données.

Le domaine auquel appartient cette réalisation est l’informatique de gestion. Deux phases composent ce projet. Une première phase est consacrée à l’appropriation du présent cahier des charges, la composition du groupe de travail et son organisation, pour finir, la modélisation logicielle et des données.

Une deuxième phase vous permettra de réaliser votre solution (application et base de données) et de la soutenir.

# **Etude du besoin :**

Après avoir analysé le besoin du client nous pouvons nous attaquer à l'études de l’architecture de l’application client-serveur.

Pour cela nous avons réalisé multiple diagramme.

Nous avons commencé par le diagramme Use Case

## **Réalisation du Diagramme de cas d’utilisation (Use Case)**

Le diagramme de cas d'utilisation est un diagramme UML utilisés pour une représentation du comportement fonctionnel d'un système logiciel.

Ils sont utiles pour des présentations auprès de la direction ou des acteurs d'un projet, mais pour le développement, les cas d'utilisation sont plus appropriés.

Une image contenant texte, diagramme, croquis

Description générée automatiquementDans notre cas, nous nous occupons de la partie client-serveur. Ce diagramme montre les actions que le client va pouvoir réaliser sur le site

. Diagramme Use Case

Ici le client peut se connecter à la plateforme. Mais s’il n’a pas de compte il peut s’inscrire.

Un fois inscrit il pourra modifier ses informations personnelles et réaliser des commandes

Une fois la commande payée, la plateforme générera une facture à son nom.

Un fois que nous avons défini les différents comportements que peut avoir le client sur le serveur nous allons approfondir plus le sujet avec d’autre diagramme.

## **Réalisation du Diagramme d’activité**

Le diagramme d'activité est un diagramme comportemental d'UML, permettant de représenter le déclenchement d'événements en fonction des états du système et de modéliser des comportements parallélisables.

Le diagramme d'activité est également utilisé pour décrire un flux de travail.

Ce diagramme sera divisé en 3 grandes parties, la connexion, la prise de commande et le paiement.

Pour mieux vous expliquer le déroulement du diagramme nous avons écrit **une trame**.

## **Une image contenant diagramme, texte, origami, motif Description générée automatiquementTrame**

1. Page d'accueil :

L'utilisateur arrive sur la page d'accueil.

L'utilisateur a deux choix :

Appuyer sur un bouton pour créer un compte.

Remplir les informations de connexion (courriel et mot de passe) et appuyer sur le bouton "Se connecter".

1. Création de compte (si choisi) :

Une image contenant diagramme, croquis, texte, conception

Description générée automatiquementL'utilisateur est redirigé vers une page de création de compte.

L'utilisateur doit saisir son courriel, mot de passe, nom, prénom et numéro de téléphone.

Après avoir rempli les informations, l'utilisateur appuie sur un bouton pour créer le compte.

1. Connexion (si choisi) :

L'utilisateur est redirigé vers une page de connexion.

L'utilisateur saisit son courriel et son mot de passe.

Après avoir rempli les informations, l'utilisateur appuie sur un bouton pour se connecter.

1. Une image contenant texte, diagramme, motif

   Description générée automatiquementPage des articles (après connexion ou création de compte) :

. Diagramme d'activité

Si l'utilisateur se connecte avec succès, il est dirigé vers une page affichant différents articles.

Un message d'accueil personnalisé est affiché avec le prénom de l'utilisateur.

Trois boutons sont disponibles : "Quitter", "Ajouter", "Acheter".

1. Quitter :

Si l'utilisateur appuie sur le bouton "Quitter", la session se termine.

1. Ajouter un article :

Si l'utilisateur appuie sur le bouton "Ajouter", l'article actuellement affiché est ajouté au tableau.

1. Acheter un article :

Si l'utilisateur appuie sur le bouton "Acheter", il est redirigé vers une page de saisie d'adresse de livraison et d'adresse de facturation.

1. Confirmer l'achat :

L'utilisateur saisit les adresses de livraison et de facturation.

L'utilisateur confirme les informations pour recevoir une facture

## **Réalisation du Diagramme de Séquence**

Le diagramme de séquence est la représentation graphique des interactions entre les acteurs et le système selon un ordre chronologique dans la formulation UML.

Ici nous avons séparer le diagramme en deux diagrammes différents. Il y a le diagramme de séquence connexion et de création

1. Diagramme de séquence : Connexion

Une image contenant texte, diagramme, Parallèle, nombre

Description générée automatiquement

3. Diagramme de séquence \_ Connexion

Dans ce diagramme, l’utilisateur s’il a un compte rentre son mail avec son mot de passe. Ça va renvoyer les informations au client.

Si les informations données correspondent aux informations dans la data base la page contenant les articles s’affichera. Au contraire un pop-up apparaitra avec un message d’erreur.

1. Diagramme de séquence : Création

Une image contenant diagramme, texte, ligne, Parallèle

Description générée automatiquement

4. Diagramme de séquence \_ création

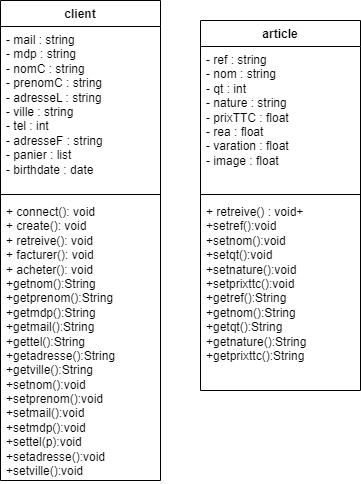
Ici l’utilisateur s’il n’a pas de compte va aller sur la page de création du compte, en cliquant sur le bouton « Créer un compte ». Après ça il aura un formulaire avec toute les informations nécessaires à sa création. Une fois remplie dès qu’il acceptera de créer son compte les données seront envoyé à la base de données et il pourra accéder aux articles de la plateforme.

Enfin nous arrivons au dernier diagramme concernant l’architecture client-serveur.

## **Réalisation du Diagramme de Classes**

Le diagramme de classes est un schéma utilisé en génie logiciel pour présenter les classes et les interfaces des systèmes ainsi que leurs relations.

Ce diagramme fait partie de la partie statique d'UML, ne s'intéressant pas aux aspects temporels et dynamiques.

****

5. Diagramme de classes

Nous avons fini d’étudier l’interface client-serveur maintenant nous allons nous intéresser aux données qui vont être stockés dans notre base de données. Pour cela nous allons commencer par réaliser un dictionnaire de données.

## **Réalisation du Dictionnaire de Données**

A l’aide des informations qui nous ont été mises à dispositions nous avons pu nous faire une idée des données qui devraient être dans la base de données.

Le dictionnaire de données est une description structurée des données utilisées dans un système informatique.

Une image contenant texte, nombre, Parallèle, Police

Description générée automatiquementIl contient des informations sur les données telles que leur format, leur type, leur source, leur signification et leur utilisation.

Une fois que nous avons bien analysé chaque éléments clé nous pouvons modéliser notre premier modèle.

## **Réalisation du Modèle conceptuel de données (MCD)**

Le MCD pour modèle conceptuel de données, également appelé schéma conceptuel de données, est une représentation claire des données du systèmes d’information à concevoir. Cette représentation en outre figure les relations entres ces données.

Une image contenant texte, capture d’écran, diagramme, conception

Description générée automatiquement

### **Explication de la modélisation des données :**

Avec notre dictionnaire de données nous allons modéliser nos données mais pour cela nous avons besoin de prendre en connaissance d’éléments clés pour notre modèle.

* **Les entités :** Les entités sélectionnes pour le MCD ont été choisies en fonction de leur importance pour le système d’information et de leur fréquence d’utilisation.
  + Par exemple, si nous créons une entité **Commande**, nous inclurons des sous entités telles que *RefC* (Référence Commande) car elle est **essentielle** pour le bon fonctionnement de la commande
* **Cardinalité des relations :** La cardinalité des relations entre les entités a été choisie en fonction de la nature de la relation.
  + Par exemple, si un client peut réaliser **plusieurs commandes** mais qu’une commande ne peut être **passé** que *par un client à la fois*, alors la relation entre client et commande aura une cardinalité de **1,1** et **1,N.**
* Attributs : Les attributs sélectionnées pour chacun entité ont été choisis en fonctions de leur pertinence pour le système d’information
  + Par exemple, si nous avons une entité « **Commande** », nous pouvons avoir des sous-entités telle que *DateP* (Date Paiement) et *Facture* pour gérer des informations spécifiques à chaque commande
* Normalisation : Les tables ont été normalisées pour éviter la redondance des données et pour améliorer la performance du système.
  + Par exemple, si nous avons des informations sur les articles stockés dans plusieurs commandes, nous normalisons les tables en regroupant les informations liées aux articles dans une seule table.

## **Réalisation du Modèle logique de données (MLD)**

Une image contenant texte, capture d’écran, diagramme, nombre

Description générée automatiquementLe modèle logique des données est une représentation plus détaillée et concrète du modèle conceptuel de données. Contrairement au MCD qui se concentre sur les entités, les attributs et les relations, le MCD prend en compte les types de données, les clés primaires et étrangères, les contraintes d’intégrité et les règles de normalisations.

Ici nous avons la relation **LISTER** qui est devenue une association en prenant en entité *RefA* (Référence Article) et *RefC* (Référence Commande)