

PROJET LOGICIEL

---

# ESHOP

---

January 19, 2016

Axel Fahy / Rudolf Höhn / Benjamin Ganty / Robin Chappatte / Pierre-Yves Kugler  
hepia - ITI

## Contents

<b>1</b>	<b><i>User Stories</i></b>	<b>3</b>
<b>2</b>	<b><i>Use case</i></b>	<b>4</b>
2.1	Diagramme . . . . .	4
2.2	Détails . . . . .	4
<b>3</b>	<b>Architecture</b>	<b>6</b>
<b>4</b>	<b>Activités</b>	<b>7</b>
4.1	Recherche . . . . .	7
4.2	Gestion du panier . . . . .	7
4.3	Commande d'un produit . . . . .	7
<b>5</b>	<b>Séquences</b>	<b>8</b>
5.1	Recherche . . . . .	8
<b>6</b>	<b><i>Sprints</i></b>	<b>9</b>
6.1	Sprint 1 - Architecture et 3D . . . . .	9
6.2	Sprint 2 - Navigation, filtres et recherche . . . . .	9
6.3	Sprint 3 - Panier . . . . .	9
6.4	Sprint 4 - Utilisateurs . . . . .	10

## 1 *User Stories*

Description	Priorité
Le client veut voir les produits du magasin	1
Le client veut avoir la description du produit en cliquant dessus	2
Le client va voir dans le rayon d'à cote les produits proposés	3
Le client veut pouvoir filtrer les produits par catégories	4
Le client veut revenir au choix des catégories	5
Le client ne trouve pas le produit qu'il cherche et demande a un employe du magasin de l'aider	6
Le client dans le magasin veut ramasser un produit sur une étagere et le mettre dans son panier	7
Le client veut voir son panier	8
Le client n'est pas content avec un de ses articles dans le panier et souhaite le retirer	9
Le client veut modifier la quantité d'un element situé dans le panier	10
Le client veut vider son panier	11
Le client veut supprimer un element du panier	12
Le client veut connaitre le solde total de son panier	13
Le client avec son panier veut pouvoir aller à la caisse et va payer ses courses	14
Le client veut rentrer ses informations de paiement et payer	15
Le client veut se créer un compte	16
Le client veut payer mais il doit d'abord s'authentifier	17
Le client veut modifier son compte (nom, mot de passe, adresse)	18
Le client veut supprimer son compte	19
Le client veut se déconnecter de son compte	20



<b>Titre</b>	Achat de produits
Acteur	Client
But	Permettre au client d'acheter des produits.
Résumé métier	Le client doit pouvoir sélectionner le(s) produit(s) à acheter.
Pré-conditions	<ol style="list-style-type: none"> <li>1. Le client doit être authentifié.</li> <li>2. Les produits doivent être disponibles.</li> </ol>
Scénario alternatif	<ol style="list-style-type: none"> <li>1. Acheter plusieurs produits d'un coup.</li> <li>2. Avoir plusieurs fois le même produit.</li> </ol>
Exception	<ol style="list-style-type: none"> <li>1. Les produits ne sont pas en stock.</li> </ol>

<b>Titre</b>	Gestion de panier
Acteur	Client
But	Permettre au client de modifier, supprimer un élément du panier
Résumé métier	Le client doit pouvoir modifier la quantité d'un élément de son panier et de le supprimer
Pré-condition	Avoir des produits dans son panier

<b>Titre</b>	Gestion de son compte client
Acteur	Client
But	Permettre au client de modifier, supprimer son compte
Résumé métier	Le client doit pouvoir modifier ses informations de facturations, de paiements ou de supprimer son compte
Pré-condition	Etre authentifié

<b>Titre</b>	Consultation des commandes
Acteur	Client
But	Permettre au client de voir ses commandes en cours
Résumé métier	Le client doit pouvoir voir l'état de ses commandes
Pré-condition	Etre authentifié

### 3 ARCHITECTURE

Nous avons décidé de développer l'application en utilisant WebGL (Front-end), PHP (Front+Back end) et MySQL (Database). Le diagramme de composants ci-dessous montre le système complet.

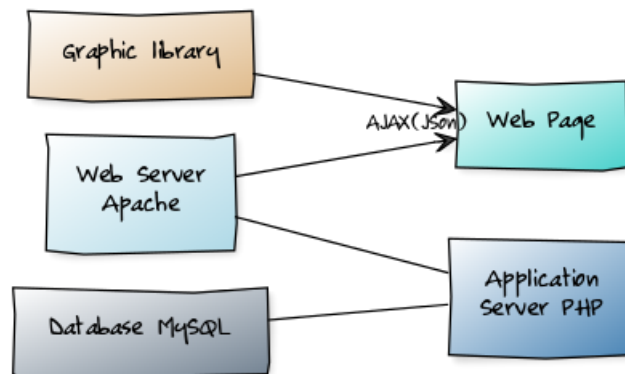


Diagramme UML de composants

## 4 ACTIVITÉS

### 4.1 Recherche

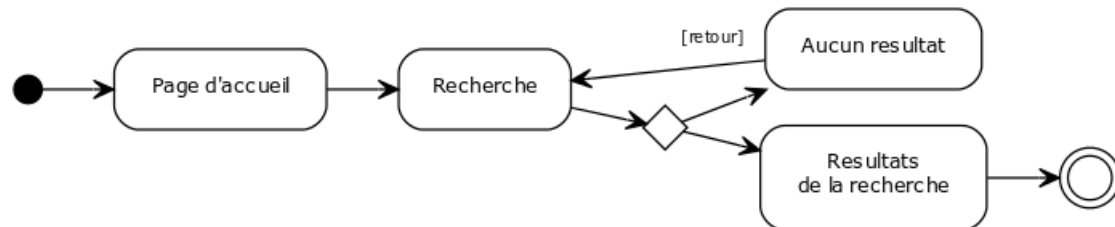


Diagramme UML de la recherche d'un produit

### 4.2 Gestion du panier

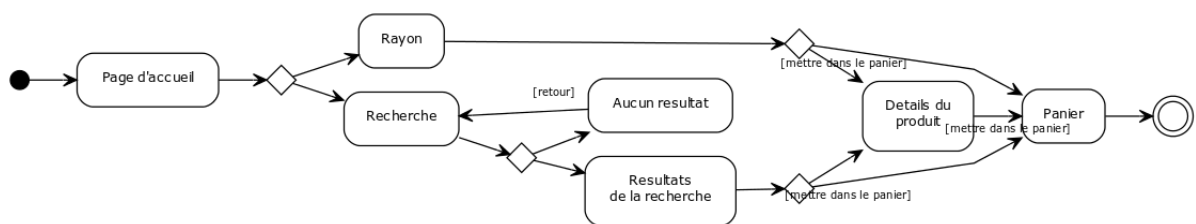


Diagramme UML de la gestion du panier

### 4.3 Commande d'un produit



Diagramme UML de l'accès à la commande d'un produit

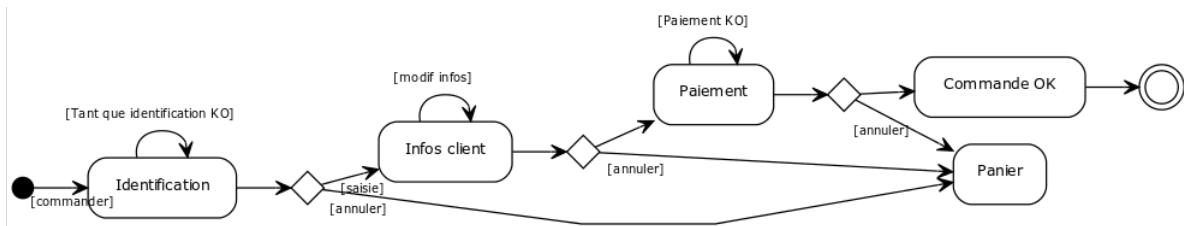


Diagramme UML d'une commande

## 5 SÉQUENCES

### 5.1 Recherche

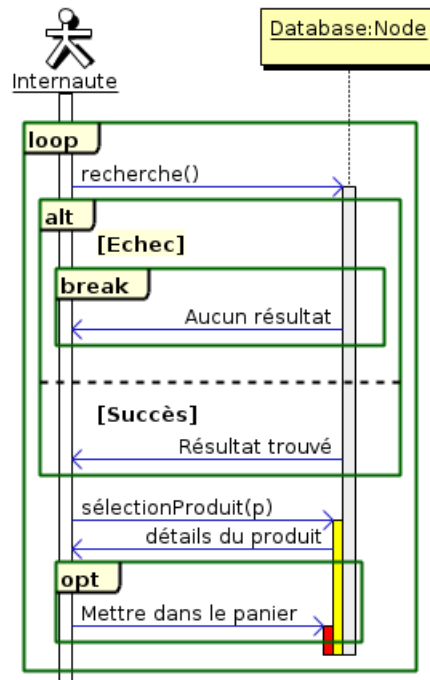


Diagramme UML de la recherche d'un produit



## **6 Sprints**

Un sprint dure entre 2 et 4 semaines. Le deuxième semestre dure 8 semaines (sans compter les vacances de Pâques), donc nous avons décidé de prévoir 4 sprints.

### **6.1 Sprint 1 - Architecture et 3D**

Dans ce sprint, on va mettre en place tout le système, développer le serveur PHP, très basique, et mettre en place la base données. Notre but est de faire fonctionner le lien entre la partie 3D et la base de données. À partir de ce moment-là, nous passerons à la visualisation des produits stockés dans la base de données. Cette tâche répond au usecase de "Visualisation de produits".

Le deuxième usecase traité dans ce sprint est "Afficher la description d'un produit". Cela permettra de tester le fonctionnement de l'architecture mise en place. A la fin du premier sprint, nous avons donc déjà un produit fonctionnel.

### **6.2 Sprint 2 - Navigation, filtres et recherche**

Dans ce 2ème sprint, nous allons mettre en place la vue par catégories des produits, le filtre des produits et la recherche des produits.

### **6.3 Sprint 3 - Panier**

Comme le titre du sprint l'indique, la gestion du panier se fera ici. Maintenant que l'utilisateur peut choisir ses produits en les filtrant, en naviguant par catégories et faisant des recherches, il pourra après ce sprint, ajouter des produits dans son panier, modifier son panier et enfin accéder au système de paiement. La partie de sélection de produit se faire via la 3D, mais toute la gestion se fera de manière classique. Le système de paiement étant externe à notre logiciel, le client sera juste redirigé vers la page du site partenaire. Le panier est géré par cookies, et donc la gestion des utilisateurs n'est pas requise à cette étape du projet.

## **6.4 Sprint 4 - Utilisateurs**

Le dernier sprint n'est pas la partie la plus dure du projet, c'est pour cela qu'elle est à la fin. Dans les comptes utilisateurs, nous pourrions stocker les adresses de facturation, les données clientes, et les informations de paiement.