

الجمهورية الجزائرية الديمقراطية الشعبية  
République Algérienne Démocratique et Populaire  
وزارة التعليم العالي والبحث العلمي  
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique



N° Réf :.....

Centre Universitaire  
Abd Elhafid Boussouf Mila

Institut des Sciences et Technologie

Département de Mathématiques et Informatique

## Mémoire préparé en vue de l'obtention du diplôme de Master

EN : Informatique

Spécialité : Sciences et Technologies de l'Information et de la Communication  
(STIC)

# Développement d'une application web d'achat en ligne

Préparé par : KELOUZ Maha  
LEGRIQUI Meriem

Devant le jury :

HEDJAZ Sabrine	MAA	C.U.Abd Elhafid Boussouf	Rapporteur
BOUBAKIR Mohamed	MAA	C.U.Abd Elhafid Boussouf	Président
ZEKIOUK Mounira	MAA	C.U.Abd Elhafid Boussouf	Examineur

Année Universitaire : 2017/2018

# Remerciement

---

*Nous remercions d'abord et avant tout Allah qui nous a donné le courage et la patience pour réaliser ce travail.*

*Un remerciement particulier à notre encadreur Madame Hedjaz Sabrina pour sa présence, ses précieux conseils et son aide durant toute la période du travail.*

*Nous remercions également les membres de jury Boubakir Mohamed Et Zekiouk Mounira Pour l'intérêt qu'ils ont porté à notre recherche en acceptant d'examiner notre travail Et de l'enrichir par leurs propositions.*

*Sans oublier tous les enseignants qui ont contribué à notre formation durant notre vie scolaire particulièrement les enseignants de notre institut.*

*Finalement, nous remercions très sincèrement tous nos familles pour leur encouragement sans limite.*

*Maha et Meriem*

# Dédicace

---

*Je dédie ce modeste travail :*

*A mes parents Djamel et Hafissa, qui ont toujours été là pour moi.*

*A mes frères : Imad et Wassim.*

*A ma sœur Racha.*

*A mon mari Mousaab.*

*A toute ma famille.*

*A mes très chères amies Nihad, Assia, Wassila, Manel.*

*À ma chère amie, collègue, et binôme Meriem.*

*A notre encadreur Hedjaz Sabrina et à nos enseignants.*

*A tous ceux qui nous ont soutenus au cours de notre travail notamment Aissam.*

*Maha.*

# Dédicace

---

*Je dédie ce modeste travail :*

*A mes parents Mohammed et Zikha, qui ont toujours été là pour moi.*

*A mon frère : Said.*

*A mes sœurs Nesrine et Djihane.*

*A toute ma famille.*

*A mes très chères amies*

*À ma chère amie, collègue, et binôme Maha.*

*A notre encadreur Hedjaz Sabrina et à nos enseignants.*

*A tous ceux qui nous ont soutenus au cours de notre travail notamment Aissam.*

*Meriem*

# Résumé

Les entreprises qui offrent des produits ou des services cherchent à adopter la meilleure gestion commerciale pour concurrencer sur le marché afin de préparer la plate-forme de commerce électronique pour mener leurs activités, basées sur la technologie et les médias.

L'objectif de notre projet est d'implémenter une application d'achat en ligne au sein du centre commercial "Beldi", dont le but principal est de développer une application web pour gérer les ventes et les achats en ligne afin d'élargir la base de sa clientèle et mieux faire connaître au public les produits vendus, développant ainsi de manière significative les ventes et le chiffre d'affaires du centre.

Afin de développer notre application, nous allons mener une étude conceptuelle de l'approche 2TUP en utilisant UML comme langage de modélisation, Framework "Laravel " basé sur PHP comme langage de programmation et MYSQL comme système de gestion de bases de données.

**Mots-clés** : FRAMEWORK LARAVEL, MYSQL, PHP, UML, 2TUP, XAMPP, HTML, CSS, JavaScript.

# *Abstract*

Companies that offer products or services search to adopt the best business management to compete in the market to prepare the electronic business platform to conduct their activities based on technology and the media.

The objective of our project is to implement an application of online purchase within the shopping mall "Beldi" the main purpose of which is to develop a Web application to manage sales and online purchases to widen the customer base and better become known to the public the sold products, so developing significantly sales and turnover of the center.

In order to develop our application, we will conduct a conceptual study of the 2TUP approach using UML as modeling language, "laravel" framework based on PHP as programming language and MYSQL as database management system.

**Keywords :** LARAVEL FRAMEWORK, MYSQL, PHP, UML, 2TUP, XAMPP, HTML, CSS, JavaScript.

## تلخيص

تسعى الشركات التي تقدم منتجات أو خدمات إلى تبني أفضل إدارة أعمال للتنافس في السوق لإعداد منصة للتجارة الإلكترونية من أجل إدارة أعمالها، اعتمادًا على التكنولوجيا والإعلام. الهدف من مشروعنا هو تنفيذ تطبيق للتسوق عبر الإنترنت داخل مركز التسوق "بلدي" الذي يتمثل هدفه الأساسي في تطوير تطبيق ويب لإدارة المبيعات والمشتريات عبر الإنترنت من أجل توسيع قاعدة البيانات. لعملائها وللإعلان على نحو أفضل عن المنتجات المباعة، وبالتالي زيادة كبيرة في المبيعات المركز.

من أجل تطوير تطبيقنا، سنجري دراسة مفاهيمي لنهج 2TUP باستخدام اللغة الموحدة (UML) كلغة نمذجة، وإطار "Laravel" القائم على PHP كلغة برمجة و MySQL كنظام لإدارة قواعد البيانات.

الكلمات الرئيسية: CSS، HTML، XAMPP، 2TUP، UML، PHP، MYSQL، FRAWMWORK LARAVEL، JavaScript.

# Table de matière

<b>Introduction général</b>	
<b>Chapitre 01 : Généralités</b>	
1. Processus de développement logiciel	<b>06</b>
1.1 Processus unifié (unified processus)	<b>06</b>
1.2 Le processus 2TUP	<b>07</b>
1.3 Un processus de modélisation avec UML	<b>10</b>
2. Application web	<b>11</b>
2.1 Pourquoi une application web	<b>11</b>
2.2 Le Framework	<b>12</b>
2.2.1 Le Framework Laravel	<b>12</b>
2.2.2 Pourquoi laravel et non pas PHP	<b>12</b>
2.3 Le modèle MVC	<b>13</b>
2.3.1 Les composants de modèle MVC	<b>14</b>
2.3.2 L'interaction entre les composants	<b>14</b>
3. E-commerce	<b>15</b>
3.1. L'entreprise et son approche du e-commerce	<b>15</b>
3.2. Avantages d'un site e-commerce	<b>15</b>
3.2.1 Pour les entreprises	<b>15</b>
3.2.2 Pour les internautes	<b>15</b>
4 .Le centre commercial Beldi	<b>16</b>
<b>Chapitre 02 : Etude préliminaire</b>	
1. Etude préliminaire	<b>18</b>
2. Elaboration du cahier des charges	<b>19</b>
2.1 Présentation du projet	<b>19</b>
2.2 Grands choix techniques	<b>19</b>
2.3 Recueil des besoins fonctionnels	<b>20</b>
2.4 Recueil des besoins opérationnels	<b>20</b>
3. Description de contexte du système	<b>20</b>
3.1 identification des acteurs	<b>21</b>
3.2 identification des messages	<b>22</b>



3.3 Modélisation du contexte	<b>23</b>
<b>Chapitre 03 : Capture des besoins fonctionnels</b>	
1. Capture des besoins fonctionnels	<b>26</b>
2. Identifier les cas d'utilisation	<b>27</b>
2.1 Liste préliminaire des cas d'utilisation	<b>28</b>
2.2 Diagramme de cas d'utilisation	<b>29</b>
3. Description détaillée des cas d'utilisation	<b>30</b>
3.1 Créer compte	<b>30</b>
3.2 Consulter article	<b>32</b>
3.3 Rechercher article	<b>33</b>
3.4 Gérer panier	<b>35</b>
3.5 S'authentifier	<b>36</b>
3.6 Passer commande	<b>38</b>
3.7 Consulter facture	<b>40</b>
3.8 Paiement en ligne	<b>41</b>
3.9 Modifier les informations du compte	<b>43</b>
3.10 Gérer article	<b>45</b>
3.11 Ajouter article	<b>47</b>
3.12 Modifier article	<b>49</b>
3.13 Supprimer article	<b>51</b>
3.14 Traiter commande	<b>53</b>
3.15 Gérer compte	<b>54</b>
3.16 Ajouter compte	<b>56</b>
3.17 Bloquer compte	<b>58</b>
3.18 Statistique	<b>60</b>
4 Identification des classes candidates	<b>62</b>
4.1 La liste des classes candidates	<b>62</b>
4.2 Responsabilités des classes	<b>63</b>

5. Diagrammes de classes participation	<b>65</b>
5.1 Cas d'utilisation : Créer compte	<b>65</b>
5.2 Cas d'utilisation : Consulter article	<b>66</b>
5.3 Cas d'utilisation : Rechercher article	<b>66</b>
5.4 Cas d'utilisation : Gérer panier	<b>67</b>
5.5 Cas d'utilisation S'authentifier	<b>68</b>
5.6 Cas d'utilisation : Passer commande	<b>69</b>
5.7 Cas d'utilisation : Consulter facture	<b>69</b>
5.8 Cas d'utilisation : Paiement en ligne	<b>70</b>
5.9 Cas d'utilisation : Modifier les informations du compte	<b>71</b>
5.10 Cas d'utilisation : Ajouter article	<b>72</b>
5.11 Cas d'utilisation : Modifier article	<b>72</b>
5.12 Cas d'utilisation : Supprimer article	<b>73</b>
5.13 Cas d'utilisation : traiter commande	<b>73</b>
5.14 Cas d'utilisation : Ajouter compte	<b>74</b>
5.15 Cas d'utilisation : bloquer compte	<b>74</b>
5.16 Cas d'utilisation : Statistique	<b>75</b>
<b>Chapitre 04 : Capture des besoins techniques</b>	
1. Capture des besoins techniques	<b>77</b>
1.1 Spécification technique du point de vue matériel	<b>78</b>
1.2 Spécification d'architecture et influence sur le modèle déploiement	<b>80</b>
1.3 Elaboration du modèle de spécification logicielle	<b>81</b>
1.3.1 Identification des cas d'utilisation techniques	<b>81</b>
1.4 Organisation du modèle de spécification logicielle	<b>82</b>
1.4.1 Organisation en couches du modèle de spécification	<b>83</b>
<b>Chapitre 05 : Analyse</b>	
1. Découpage en catégorie	<b>86</b>
1.1 Répartition des classes candidates en catégories	<b>87</b>

1.2 Elaboration des diagrammes de classes préliminaire par catégorie	<b>88</b>
1.3 Association entre catégorie	<b>89</b>
2. Développement du modèle statique	<b>89</b>
2.1 Diagramme de classe pour la catégorie compte	<b>90</b>
2.2 Diagramme de classe pour la catégorie commande	<b>91</b>
3. Développement du modèle dynamique	<b>91</b>
3.1 Les diagrammes d'interaction	<b>92</b>
3.2 Ajouter les opérations	<b>106</b>
<b>Chapitre 06 : Conception</b>	
1. Conception générique	<b>109</b>
1.1 Notion des classes	<b>110</b>
1.2 Définition d'un Framework technique	<b>110</b>
1.3 Interface	<b>110</b>
1.4 Elaboration du modèle logique de conception	<b>110</b>
1.4.1 Organisation des frameworks techniques	<b>110</b>
2. Conception préliminaire	<b>112</b>
2.1 Développement du modèle de déploiement	<b>112</b>
2.1.1 poste de travail	<b>112</b>
3. Conception détaillée	<b>113</b>
3.1 Dictionnaire des données	<b>113</b>
3.1.1 Les classes et les attributs	<b>113</b>
3.1.2 Les opérations	<b>115</b>
3.2 Diagramme de classe détaillé	<b>117</b>
3.3 Passage vers le model relationnel	<b>118</b>
3.3.1 Les règles de passage	<b>118</b>
3.3.2 Les tables de la base de données	<b>119</b>
<b>Chapitre 07 : Implémentation</b>	
1. Langage et outil de développement	<b>121</b>

1.1 Le Framework Laravel	<b>121</b>
1.2 Le serveur XAMPP	<b>121</b>
1.2.1 Apache	<b>122</b>
1.2.2 PHP	<b>122</b>
1.2.3 MYSQL	<b>122</b>
1.3. Les standards web	<b>123</b>
2. Description d'application	<b>124</b>
<b>Conclusion générale</b>	<b>129</b>
<b>Bibliographie</b>	<b>130</b>

## *Liste des figures*

<b>Chapitre 01 : Généralités</b>		
Figure 01	Le système d'information soumis à deux natures de contraintes	<b>08</b>
Figure 02	Le processus de développement en Y	<b>08</b>
Figure 03	Schéma MVC	<b>14</b>
<b>Chapitre 02 : Etude préliminaire</b>		
Figure 01	Phase d'étude préliminaire	<b>18</b>
Figure 02	Résumé des activités et des produits de l'étude préliminaire	<b>19</b>
Figure 03	Diagramme de contexte dynamique	<b>23</b>
<b>Chapitre 03 : Capture des besoins fonctionnels</b>		
Figure 01	Phase de capture des besoins fonctionnels	<b>26</b>
Figure 02	Démarche de capture des besoins fonctionnels	<b>27</b>
Figure 03	Diagramme de cas d'utilisation	<b>29</b>
Figure 04	Diagramme de séquence du cas d'utilisation « Créer compte »	<b>31</b>
Figure 05	Diagramme d'activité du cas d'utilisations « Créer compte »	<b>31</b>
Figure 06	Diagramme de séquence du cas d'utilisation « Consulter article »	<b>32</b>
Figure 07	Diagramme d'activité du cas d'utilisation « Consulter article »	<b>33</b>
Figure 08	Diagramme de séquence du cas d'utilisation « Rechercher article »	<b>34</b>
Figure 09	Diagramme d'activité du cas d'utilisation « Rechercher article »	<b>34</b>
Figure 10	Diagramme de séquence du cas d'utilisation « Gérer panier »	<b>35</b>
Figure 11	Diagramme d'activité du cas d'utilisation « Gérer panier »	<b>36</b>
Figure 12	Diagramme de séquence du cas d'utilisation « S'authentifier »	<b>37</b>
Figure 13	Diagramme d'activité du cas d'utilisation « S'authentifier »	<b>37</b>
Figure 14	Diagramme de séquence du cas d'utilisation « Passer commande »	<b>39</b>
Figure 15	Diagramme d'activité du cas d'utilisation « Passer commande »	<b>39</b>
Figure 16	Diagramme de séquence du cas d'utilisation « Consulter facture »	<b>40</b>
Figure 17	Diagramme d'activité du cas d'utilisation « Consulter facture »	<b>41</b>
Figure 18	Diagramme de séquence du cas d'utilisation « Paiement en ligne »	<b>42</b>
Figure 19	Diagramme d'activité du cas d'utilisation « Paiement en ligne »	<b>43</b>
Figure 20	Diagramme de séquence du cas d'utilisation « Modifier les informations du compte »	<b>44</b>

Figure 21	Diagramme d'activité du cas d'utilisation « Modifier les informations du compte »	<b>45</b>
Figure 22	Diagramme de séquence du cas d'utilisation « Gérer article »	<b>46</b>
Figure 23	Diagramme d'activité du cas d'utilisation «Gérer article »	<b>46</b>
Figure 24	Diagramme de séquence du cas d'utilisation « Ajouter article »	<b>48</b>
Figure 25	Diagramme d'activité du cas d'utilisation « Ajouter article »	<b>48</b>
Figure 26	Diagramme de séquence du cas d'utilisation « Modifier article »	<b>50</b>
Figure 27	Diagramme d'activité du cas d'utilisation « Modifier article »	<b>51</b>
Figure 28	Diagramme de séquence du cas d'utilisation « Supprimer article »	<b>52</b>
Figure 29	Diagramme d'activité du cas d'utilisation « Supprimer article »	<b>52</b>
Figure 30	Diagramme de séquence du cas d'utilisation « Traiter commande »	<b>53</b>
Figure 31	Diagramme d'activité du cas d'utilisation « Traiter commande »	<b>54</b>
Figure 32	Diagramme de séquence du cas d'utilisation « Gérer compte »	<b>55</b>
Figure 33	Diagramme d'activité du cas d'utilisation « Gérer compte »	<b>55</b>
Figure 34	Diagramme de séquence du cas d'utilisation « Ajouter compte »	<b>57</b>
Figure 35	Diagramme d'activité du cas d'utilisation « Ajouter compte »	<b>58</b>
Figure 36	Diagramme de séquence du cas d'utilisation « Bloquer compte »	<b>59</b>
Figure 37	Diagramme d'activité du cas d'utilisation « Bloquer compte »	<b>60</b>
Figure 38	Diagramme de séquence du cas d'utilisation « Statistique »	<b>61</b>
Figure 39	Diagramme d'activité du cas d'utilisation « Statistique »	<b>61</b>
Figure 40	Les responsabilités de la classe candidat « Compte »	<b>63</b>
Figure 41	Les responsabilités de la classe candidat « Administrateur »	<b>63</b>
Figure 42	Les responsabilités de la classe candidat « Client »	<b>63</b>
Figure 43	Les responsabilités de la classe candidat « Vendeur »	<b>63</b>
Figure 44	Les responsabilités de la classe candidat « Catégorie »	<b>64</b>
Figure 45	Les responsabilités de la classe candidat « Article »	<b>64</b>
Figure 46	Les responsabilités de la classe candidat « Panier »	<b>64</b>
Figure 47	Les responsabilités de la classe candidat « Commande»	<b>64</b>
Figure 48	Les responsabilités de la classe candidat « Facture »	<b>65</b>
Figure 49	Diagramme de classe participantes de « Créer compte»	<b>65</b>
Figure 50	Diagramme de classe participantes de « Consulter article »	<b>66</b>
Figure 51	Diagramme de classe participantes de « Rechercher article »	<b>66</b>
Figure 52	Diagramme de classe participantes de « Gérer panier »	<b>67</b>

Figure 53	Diagramme de classe participantes de « s’authentifier »	<b>68</b>
Figure 54	Diagramme de classe participantes de « Passer commande »	<b>69</b>
Figure 55	Diagramme de classe participantes de « Consulter facture»	<b>69</b>
Figure 56	Diagramme de classe participantes de « Paiement en ligne »	<b>70</b>
Figure 57	Diagramme de classe participantes de « Modifier les informations du compte »	<b>71</b>
Figure 58	Diagramme de classe participantes de « Ajouter article »	<b>72</b>
Figure 59	Diagramme de classe participantes de « Modifier article »	<b>72</b>
Figure 60	Diagramme de classe participantes de « Supprimer article »	<b>73</b>
Figure 61	Diagramme de classe participantes de « Traiter commande »	<b>73</b>
Figure 62	Diagramme de classe participantes de « Ajouter compte »	<b>74</b>
Figure 63	Diagramme de classe participantes de « Bloquer compte »	<b>74</b>
Figure 64	Diagramme de classe participantes de « Statistique»	<b>75</b>
<b>Chapitre 04 : Capture des besoins techniques</b>		
Figure 01	Phase de capture de besoins techniques	<b>77</b>
Figure 02	Construction de l’étape de capture de besoins techniques	<b>78</b>
Figure 03	Configuration matériel du système	<b>80</b>
Figure 04	Modèle de spécification logicielle du système	<b>82</b>
Figure 05	Style d’architecture en 5 couches	<b>83</b>
Figure 06	Organisation du modèle spécification de logicielle	<b>84</b>
<b>Chapitre 05 : Analyse</b>		
Figure 01	Phase d'analyse	<b>86</b>
Figure 02	Démarche du découpage en catégorie	<b>87</b>
Figure 03	Le découpage en catégories	<b>87</b>
Figure 04	Diagramme de classes préliminaires par catégorie	<b>88</b>
Figure 05	Dépendance entre catégorie	<b>89</b>
Figure 06	Démarche d’élaboration du modèle statique	<b>90</b>
Figure 07	Diagramme de classes pour la catégorie compte	<b>90</b>
Figure 08	Diagramme de classes pour la catégorie commande	<b>91</b>
Figure 09	Démarche d’élaboration du modèle dynamique	<b>91</b>
Figure 10	Le diagramme d’interaction de cas d’utilisation « Créer compte »	<b>92</b>
Figure 11	Le diagramme d’interaction de cas d’utilisation « Consulter article»	<b>92</b>
Figure 12	Le diagramme d’interaction de cas d’utilisation « Rechercher article »	<b>93</b>

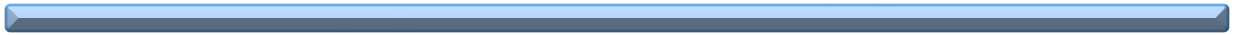
Figure 13	Le diagramme d'interaction de cas d'utilisation « Gérer panier »	<b>94</b>
Figure 14	Le diagramme d'interaction de cas d'utilisation « S'authentifier »	<b>95</b>
Figure 15	Le diagramme d'interaction de cas d'utilisation « Passer commande »	<b>96</b>
Figure 16	Le diagramme d'interaction de cas d'utilisation « Consulter facture »	<b>97</b>
Figure 17	Le diagramme d'interaction de cas d'utilisation « Paiement en ligne »	<b>98</b>
Figure 18	Le diagramme d'interaction de cas d'utilisation « Modifier les informations du compte »	<b>99</b>
Figure 19	Le diagramme d'interaction de cas d'utilisation « Ajouter article »	<b>100</b>
Figure 20	Le diagramme d'interaction de cas d'utilisation « Modifier article »	<b>101</b>
Figure 21	Le diagramme d'interaction de cas d'utilisation « Supprimer article »	<b>102</b>
Figure 22	Le diagramme d'interaction de cas d'utilisation « Traiter commande »	<b>103</b>
Figure 23	Le diagramme d'interaction de cas d'utilisation « Ajouter compte»	<b>104</b>
Figure 24	Le diagramme d'interaction de cas d'utilisation « Bloquer compte »	<b>105</b>
Figure 25	Le diagramme d'interaction de cas d'utilisation « Statistique »	<b>105</b>
Figure 26	Ajouter les opérations pour la catégorie compte	<b>106</b>
Figure 27	Ajouter les opérations pour la catégorie commande	<b>106</b>
<b>Chapitre 06 : Conception</b>		
Figure 01	Phase de conception	<b>109</b>
Figure 02	Organisation du modèle logique	<b>111</b>
Figure 03	Le modèle de déploiement	<b>113</b>
Figure 04	Diagramme de classe détaillé	<b>117</b>
<b>Chapitre 07 : Implémentation</b>		
Figure 01	Fenêtre d'accueil	<b>124</b>
Figure 02	Fenêtre d'inscription	<b>124</b>
Figure 03	Fenêtre d'authentification	<b>125</b>
Figure 04	Fenêtre panier	<b>125</b>
Figure 05	Fenêtre de la gestion des articles.	<b>126</b>
Figure 06	Fenêtre d'ajout d'un article	<b>126</b>
Figure 06	Fenêtre de la gestion des comptes	<b>127</b>



## *Liste des tableaux*

<b>Chapitre 02 : Etude préliminaire</b>		
Tableau 01	Identification des acteurs	<b>22</b>
Tableau 02	Identification des messages	<b>23</b>
Tableau 03	Description détaillée des messages	<b>24</b>
<b>Chapitre 03 : Capture des besoins fonctionnels</b>		
Tableau 01	Liste préliminaire des cas d'utilisation	<b>28</b>
Tableau 02	Description textuelle du cas d'utilisation « Créer compte »	<b>30</b>
Tableau 03	Description textuelle du cas d'utilisation « Consulter article »	<b>32</b>
Tableau 04	Description textuelle du cas d'utilisation « Rechercher article »	<b>33</b>
Tableau 05	Description textuelle du cas d'utilisation « Gérer panier »	<b>35</b>
Tableau 06	Description textuelle du cas d'utilisation « S'authentifier »	<b>36</b>
Tableau 07	Description textuelle du cas d'utilisation « Passer commande »	<b>38</b>
Tableau 08	Description textuelle du cas d'utilisation « Consulter facture »	<b>40</b>
Tableau 09	Description textuelle du cas d'utilisation « Paiement en ligne »	<b>42</b>
Tableau 10	Description textuelle du cas d'utilisation « Modifier les informations du compte »	<b>43</b>
Tableau 11	Description textuelle du cas d'utilisation « Gérer article »	<b>45</b>
Tableau 12	Description textuelle du cas d'utilisation « Ajouter article »	<b>47</b>
Tableau 13	Description textuelle du cas d'utilisation « Modifier article »	<b>49</b>
Tableau 14	Description textuelle du cas d'utilisation « Supprimer article »	<b>51</b>
Tableau 15	Description textuelle du cas d'utilisation « traiter commande »	<b>53</b>
Tableau 16	Description textuelle du cas d'utilisation « Gérer compte »	<b>54</b>
Tableau 17	Description textuelle du cas d'utilisation « Ajouter compte »	<b>56</b>
Tableau 18	Description textuelle du cas d'utilisation « Bloquer compte »	<b>58</b>
Tableau 19	Description textuelle du cas d'utilisation « Statistique »	<b>60</b>
<b>Chapitre 06 : Conception</b>		
Tableau 1	Conception des classes	<b>115</b>
Tableau 2	Tableau des opérations	<b>116</b>

# Introduction générale



# Introduction générale

---

## Préambule

Le monde a récemment connu à une révolution majeure de la technologie, cette dernière a pris une large place dans les activités des gens surtout dans le domaine des technologies de l'information et de la communication. En effet elle est devenue le centre d'intérêt dans tous les domaines, et elle est utilisée dans de nombreuses entreprises et administrations rendant très facile et plus accessible la diffusion et la réception de l'information.

Les applications web sont un outil très important pour la gestion des entreprises, ces dernières doivent donc s'équiper d'une ou plusieurs applications pour augmenter le gain de temps et le développement de la production, tels que le commerce électronique qui permet l'échange de biens et services sans frontières, ce type de commerce est une excellente occasion d'élargir la liste des clients potentiel.

Le centre commercial "Beldi" a besoin d'une application web d'achat en ligne pour élargir la base de sa clientèle et mieux faire connaître au public les produits vendus, développant ainsi de manière significative les ventes et le chiffre d'affaires du centre.

L'objectif de notre projet est de développer une application web pour gérer les achats en ligne au sein du centre commercial "Beldi".

## Problématique

Le commerce classique perd de plus en plus de sa popularité, le marketing est très concentré sur les stratégies qui peuvent apporter les plus hauts revenus pour la marque.

De nos jours, une grande majorité des achats se réalisent sur internet, le commerce traditionnel ne réalise pas des bénéfices considérables.

Les problèmes rencontrés dans ce type de commerce sont les suivants :

- Rencontre des acheteurs et des vendeurs sur un lieu physique c'est à dire que le client doit se déplacer, et faire le tour du centre pour pouvoir trouver le produit qu'il recherche.
- le champ de vente est limité par le nombre des clients qui visite le centre.

# Introduction générale

---

## Motivation

Plusieurs millions de personnes se connectent tous les jours à internet et beaucoup d'entre eux sont intéressés par le commerce électronique et l'idée de développer une application d'achat en ligne est devenue le souhait d'une grande majorité d'entre elles.

Le commerçant va arriver à commercialiser plusieurs types de produits ou d'articles en fonction des commandes des clients, ces derniers vont faire des achats en fonction des produits publiés par les vendeurs.

À l'évolution du développement d'applications Web dans le domaine du commerce électronique, on peut atteindre un grand nombre d'objectifs, notamment :

- Pour acheter un article le client n'a qu'à effectuer ses achats sur le Web. Il n'a plus besoin de se déplacer, Il dispose de l'intégralité des informations et des caractéristiques du produit souhaité (prix, couleurs disponibles, etc.).
- Offre plus large, c'est-à-dire que le vendeur n'est pas limité par la surface de son magasin pour exposer ces articles et ça sert à augmenter les performances du magasin et de l'entreprise.
- La livraison de la commande est effectuée à domicile, et ce en respectant les délais préconisés par le vendeur, ce qui permet de gagner le temps de travail.

## Démarche adoptée

Le déploiement d'une démarche qui s'appuie sur un langage de modélisation est nécessaire pour bien maîtriser le déroulement du projet. Dans le cadre de notre étude nous allons utiliser un processus de développement nommé 2TUP et un langage de modélisation qui est UML.

## Organisation du mémoire

Nous avons structuré ce mémoire en six chapitres :

**Chapitre 01 : « Etude préliminaire »** Ce chapitre consiste à recueillir les besoins des utilisateurs, c'est-à-dire d'élaborer le cahier des charges et de décrire le contexte du système en identifiant les acteurs qui interagiront avec le système et les messages entre eux.

**Chapitre 02 :** Ce chapitre intitulé « **Capture des besoins fonctionnels** » sera consacré à la présentation des besoins fonctionnels en identifiant les cas d'utilisation, et en donnant la

# Introduction générale

---

description de chaque cas d'utilisation pour recueillir toutes les informations d'aspect fonctionnel.

**Chapitre 03 :** Ce chapitre intitulé « **Capture des besoins techniques** » présente l'architecture globale utilisée par notre système sur les deux points de vue matériel et logiciel.

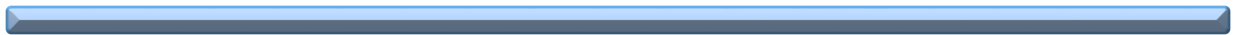
**Chapitre 04 :** « **Analyse** » Dans ce chapitre nous présentons trois activités de la phase d'analyse : le découpage en catégorie qui sert à regrouper logiquement des classes, le développement du modèle statique qui présente le diagramme de classe pour chaque catégorie, et la troisième activité le développement du modèle dynamique, qui présente le diagramme d'interaction pour chaque cas d'utilisation.

**Chapitre 05 :** « **Conception** » Ce chapitre comporte trois parties : la conception générique qui consiste à développer la solution qui répond aux spécifications techniques, la conception préliminaire qui sert à effectuer la fusion des études fonctionnelles et techniques et enfin la conception détaillée qui sert à construire les classes et montre le diagramme de classe détaillé, et le passage vers le modèle relationnel pour passer à la conception de la base de données.

**Chapitre 06 :** « **Implémentation** » Dans ce chapitre nous allons présenter les langages et outils de développement que nous avons utilisés et montrons quelques interfaces de l'application.

Nous terminons notre mémoire avec une conclusion générale

# Chapitre 01



## **Généralités**

## **Introduction**

Dans ce chapitre, nous présentons quelques généralités sur les méthodes, les concepts et les techniques utilisées pour la réalisation de notre projet.

Nous commençons par le processus de développement UP et le processus 2TUP qui l'implémente ainsi que le processus de modélisation avec UML, Ensuite, nous définissons les applications web et pourquoi implémenter des applications web. Après et comme notre application a été développée en utilisant le framework laravel en respectant le modèle MVC, nous complétons ce chapitre par une présentation de ce framework et du modèle MVC.

## **1. Processus de développement logiciel**

Un processus définit une séquence d'étapes, en partie ordonnées, qui concourent à l'obtention d'un système logiciel ou à l'évolution d'un système existant. L'objet d'un processus de développement est de produire des logiciels de qualité qui répondent aux besoins de leurs utilisateurs dans des temps et des coûts prévisibles. En conséquence, le processus peut se décomposer suivant deux axes de contrôle sur le développement : • l'axe de développement technique, qui se concentre principalement sur la qualité de la production ; • l'axe de gestion du développement, qui permet la mesure et la prévision des coûts et des délais. Nous ne traitons pas cet aspect dans « UML en Action... ». [3]

### **1.1 Processus unifié (unified processus)**

Un processus unifié est un processus de développement logiciel construit sur UML ; il est itératif et incrémental, centré sur l'architecture, conduit par les cas d'utilisation et piloté par les risques.

La gestion d'un tel processus est organisée suivant les 4 phases suivantes:

Pré étude (inception), élaboration, construction et transition.

Ses activités de développement sont définies par 5 disciplines fondamentales qui décrivent la capture des besoins, l'analyse et la conception, l'implémentation, le test et le déploiement.

Le processus unifié doit donc être compris comme une trame commune des meilleures pratiques de développement, et non comme l'ultime tentative d'élaborer un processus universel.

La définition d'un processus UP est donc constituée de plusieurs disciplines d'activité de production et de contrôle de cette production.

Tout processus UP répond aux caractéristiques ci-après :

- **Il est itératif et incrémental :** La définition d'itérations de réalisation est en effet la meilleure pratique de gestion des risques d'ordre à la fois technique et fonctionnel. On peut estimer qu'un projet qui ne produit rien d'exécutable dans les 9 mois court un risque majeur d'échec. Chaque itération garantit que les équipes sont capables d'intégrer l'environnement technique pour développer un produit final et fournit aux utilisateurs un résultat tangible de leurs spécifications. Le suivi des itérations constitue par ailleurs un excellent contrôle des coûts et des délais.

**Il est piloté par les risques:** dans ce cadre, les causes majeures d'échec d'un projet logiciel doivent être écartées en priorité. Nous identifions une première cause provenant de l'incapacité de l'architecture technique à répondre aux contraintes opérationnelles, et une seconde cause liée à l'inadéquation du développement aux besoins des utilisateurs.

- Il est construit autour de la création et de la maintenance d'un modèle, plutôt que de la production de montagnes de documents, Le volume d'informations de ce modèle nécessite une organisation stricte qui présente les différents points de vue du logiciel à différents degrés d'abstraction, l'obtention de métriques sur le modèle fournit par ailleurs des moyens objectifs d'estimation.

- **Il est orienté composant:** tant au niveau modélisation que production, c'est une garantie de souplesse pour le modèle lui-même et le logiciel qu'il représente, cette pratique constitue le support nécessaire à la réutilisation logicielle et offre des perspectives de gains non négligeables.

- **Il est orienté utilisateur:** car la spécification et la conception sont construites à partir des modes d'utilisation attendus par les acteurs du système. [3]

## 1.2 Le processus 2TUP

2TUP signifie « 2 Track Unified Process ». C'est un processus UP qui répond aux caractéristiques que nous venons de citer, le processus 2TUP apporte une réponse aux contraintes de changement continu imposées aux systèmes d'information de l'entreprise, en ce sens, il renforce le contrôle sur les capacités d'évolution et de correction de tels systèmes.



« 2 Track » signifie littéralement que le processus suit deux chemins. Il s’agit des chemins « fonctionnels » et « d’architecture technique », qui correspondent aux deux axes des changements imposés au système informatique. [3]

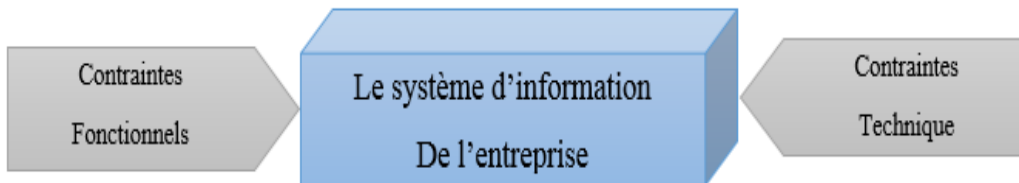


Figure 01 : Le système d'information soumis à deux natures de contraintes

À l’issue des évolutions du modèle fonctionnel et de l’architecture technique, la réalisation du système consiste à fusionner les résultats des deux branches. Cette fusion conduit à l’obtention d’un processus de développement en forme de Y, comme illustré par la figure 2.

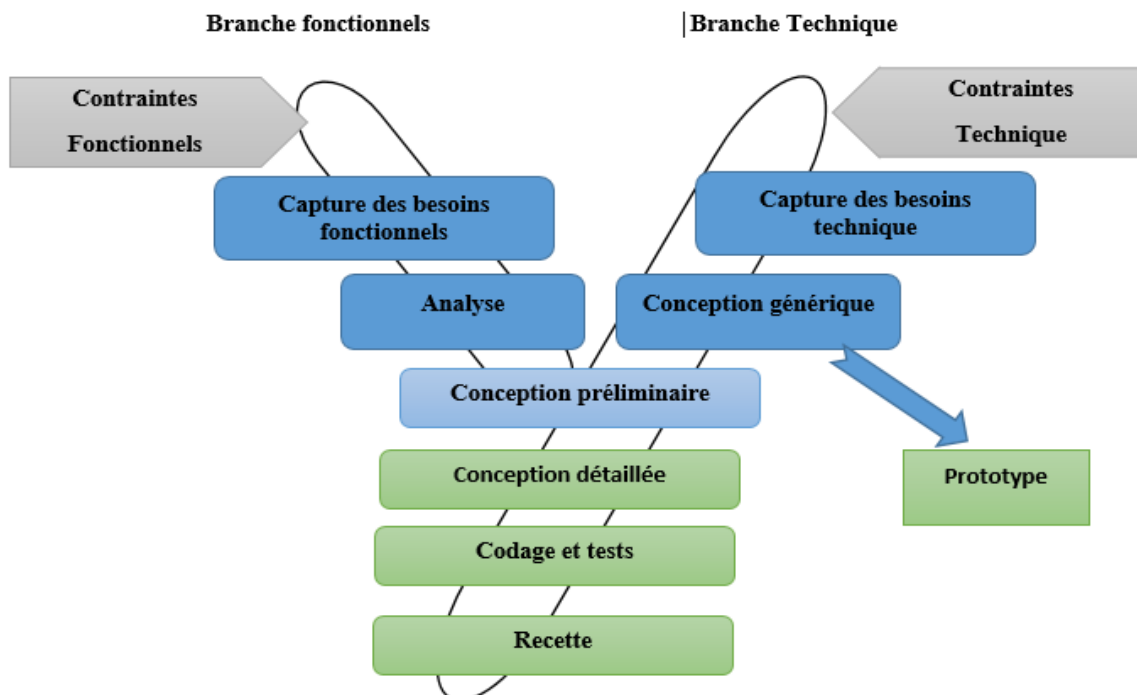


Figure 02 : Le processus de développement en Y.

✓ **La branche gauche (fonctionnelle) comporte :**

- la capture des besoins fonctionnels, qui produit un modèle des besoins focalisé sur le métier des utilisateurs. Elle qualifie au plus tôt le risque de produire un système inadapté aux utilisateurs. De son côté, la maîtrise d'œuvre consolide les spécifications et en vérifie la cohérence et l'exhaustivité l'analyse, qui consiste à étudier précisément la spécification fonctionnelle de manière à obtenir une idée de ce que va réaliser le système en termes de métier. Les résultats de l'analyse ne dépendent d'aucune technologie particulière. [3]

✓ **La branche droite (architecture technique) comporte :**

- la capture des besoins techniques, qui recense toutes les contraintes et les choix dimensionnant la conception du système. Les outils et les matériels sélectionnés ainsi que la prise en compte de contraintes d'intégration avec l'existant conditionnent généralement des prérequis d'architecture technique.

- la conception générique, qui définit ensuite les composants nécessaires à la construction de l'architecture technique, cette conception est la moins dépendante possible des aspects fonctionnels, le a pour objectif d'uniformiser et de réutiliser les mêmes mécanismes pour tout un système. L'architecture technique construit le squelette du système informatique et écarte la plupart des risques de niveau technique, l'importance de sa réussite est telle qu'il est conseillé de réaliser un prototype pour assurer sa validité.

La branche du milieu comporte :

- la conception préliminaire, qui représente une étape délicate, car elle intègre le modèle d'analyse dans l'architecture technique de manière à tracer la cartographie des composants du système à développer ;

- la conception détaillée, qui étudie ensuite comment réaliser chaque composant ;

- l'étape de codage, qui produit ces composants et teste au fur et à mesure les unités de code réalisées ;

- l'étape de recette, qui consiste enfin à valider les fonctions du système développé.

L'ensemble de ces étapes de développement sera illustré tout au long de cet ouvrage par la mise en application du processus 2TUP à l'étude de cas.

Seules les deux dernières étapes, ne relevant pas de l'utilisation d'UML, ne seront pas abordées dans cet ouvrage. [3]

## 1.3 Un processus de modélisation avec UML :

Le processus 2TUP s'appuie sur UML tout au long du cycle de développement, car les différents diagrammes de ce dernier permettent de part leur facilité et clarté, de bien modéliser le système à chaque étape.

« *Unified Modeling Language* »: UML se définit comme un langage de modélisation graphique et textuel destiné à comprendre et décrire des besoins, spécifier, concevoir des solutions et communiquer des points de vue. (Pitman, 2006)

UML unifie à la fois les notations et les concepts orientés objet. Il ne s'agit pas d'une simple notation, mais les concepts transmis par un diagramme ont une sémantique précise et sont porteurs de sens au même titre que les mots d'un

Langage, c'est pour ça qu'UML est présenté parfois comme une méthode alors qu'il ne l'est absolument pas.

UML unifie également les notations nécessaires aux différentes activités d'un processus de développement et offre, par ce biais, le moyen d'établir le suivi des décisions prises, depuis la définition des besoins jusqu'au codage.

(Roques, 2006)

Voici une présentation rapide des différents diagrammes UML qui vont être utilisés tout au long du projet :

- **Le diagramme des cas d'utilisation** : représente la structure des fonctionnalités nécessaires aux utilisateurs du système. Il est normalement utilisé lors des étapes de capture des besoins fonctionnels et techniques.
- **Le diagramme d'activités** : représente les règles d'enchaînement des activités et actions dans le système. Il peut être assimilé comme un algorithme mais schématisé.
- **Le diagramme de packages** : présent depuis UML 2.0, ce diagramme modélise des catégories cohérentes entre elles, pour un souci de partage des rôles.

Correspond à l'étape de modélisation des différents scénarios d'un cas d'utilisation.

- **Le diagramme de classes** : sûrement l'un des diagrammes les plus importants dans un développement orienté objet. Sur la branche fonctionnelle, ce diagramme est prévu pour développer la structure des entités manipulées par les utilisateurs.

En conception, le diagramme de classes représente la structure d'un code orienté objet

- **Le diagramme de séquence** : représente les échanges de messages entre objets, dans le cadre d'un fonctionnement particulier du système.
- **Le diagramme d'états** : représente le cycle de vie d'un objet. Il spécifie les états possibles d'une classe et leur enchaînement.

Ce diagramme est utilisé lors des étapes d'analyse et de conception. [16]

## 2. Application web

Une application web est un logiciel applicatif, d'où son nom, hébergé sur un serveur et accessible via les navigateurs Internet (Explorer, Mozilla Firefox, Safari...) ou sur un réseau intranet.

Contrairement à un logiciel classique, les applications web n'ont pas besoin d'être installées sur les ordinateurs de leurs utilisateurs : il leur suffit en effet de se connecter à l'application à l'aide de leurs navigateurs favoris pour pouvoir y accéder, leur but est d'offrir une expérience à l'utilisateur et des fonctionnalités équivalentes aux logiciels habituellement installés sur les ordinateurs, en utilisant de technologies de développement identiques à celles employées habituellement dans la création des sites internet.

La création des applications web demande la maîtrise parfaite de différents langages de codage, comme le HTML, le CSS, JavaScript... Le dynamisme et l'ergonomie des applications web tiennent dans le fait qu'elles contiennent divers widgets qui permettent de les manipuler et d'envoyer de nouvelles requêtes. [17]

### 2.1 Pourquoi une application web :

Les applications web se déclinent sous de multiples formes : jeux, moteurs de recherche, géolocalisation, messagerie, logiciels professionnels...

Les applications web offrent donc de nombreux avantages aux utilisateurs : système ouvert et libre de droit, coûts de développement réduits, compatibilité totale avec les navigateurs, adaptation à tous les formats d'écrans, consultation possible hors connexion.

L'avantage majeur des applications web utilisant par exemple des langages modernes comme HTML 5 est qu'elles n'ont pas besoin d'utiliser d'autres composants comme Google Gears, Adobe Flash Player ou Microsoft Silverlight pour fonctionner. Elles peuvent afficher des vidéos et des plages sonores, ou intégrer le drop and drag, ce qui les rend consultables depuis n'importe

quel terminal. De plus, leurs coûts de développement sont très réduits et elles sont aussi plus rapides à mettre en place. Enfin, elles ne cessent d'évoluer et proposent sans cesse de nouvelles fonctionnalités. Par exemple, le stockage des fichiers et la gestion du mode hors-ligne sont maintenant possibles.

Par ailleurs, l'annonceur est indépendant vis-à-vis de ses applications web et de son application store. En effet, il ne subit pas le contrôle et la pression permanents des plateformes d'application comme Appstore ou Android Market, et ne doit pas leur verser de commission prohibitive.

Les applications sont donc désormais incontournables pour toutes les entreprises, et permettent de s'imposer face à la concurrence en offrant de nombreuses possibilités d'exploitation. [17]

## 2.2 Le Framework

Un Framework informatique est un « ensemble cohérent de composants logiciels structurels, qui sert à créer les fondations ainsi que les grandes lignes de tout ou d'une partie d'un logiciel » ; autrement dit, c'est une base cohérente avec des briques toutes prêtes à disposition. Il existe des Framework pour tous les langages de programmation et en particulier pour PHP.

L'utilité d'un Framework est d'éviter de passer du temps à développer ce qui a déjà été fait par d'autres, souvent plus compétents, et qui a en plus été utilisé et validé par de nombreux utilisateurs. On peut imaginer un Framework comme un ensemble d'outils à disposition, Pour notre projet on s'intéresse au Framework Laravel. [4]

### 2.2.1 Le Framework Laravel

C'est un framework PHP open sources basé sur l'architecture MVC. Sortie en juin 2011, il est actuellement l'un des frameworks PHP les plus populaires grâce à sa documentation bien fournie, son plus gros point fort est sa simplicité, le code que l'on écrit avec Laravel est toujours simple, lisible, concis et agréable à lire. On y retrouve les standards de programmation ainsi que la possibilité d'ajouter facilement des extensions. [18]

### 2.2.2 Pourquoi laravel et non pas PHP

Plonger dans le code de Laravel, c'est recevoir un cours de programmation tant le style est clair et élégant et le code merveilleusement organisé. La version actuelle de Laravel est la 5.2 et nécessite au minimum la version 5.5.9 de PHP. Pour aborder de façon efficace ce framework, il est souhaitable de se familiariser avec les notions suivantes.

- Les espaces de noms : c'est une façon de bien ranger le code pour éviter des conflits de nommage. Laravel utilise cette possibilité de façon intensive. Tous les composants sont rangés dans des espaces de noms distincts, de même que l'application créée.
- Les fonctions anonymes : ce sont des fonctions sans nom (souvent appelées clo- sures) qui améliorent le code. Les utilisateurs de JavaScript y sont habitués, ceux de PHP un peu moins parce qu'elles y sont plus récentes. Laravel les utilise aussi de façon systématique.
- Les méthodes magiques : ce sont des méthodes qui n'ont pas été explicitement décrites dans une classe, mais qui peuvent être appelées et résolues.
- Les interfaces : une interface est un contrat de constitution des classes. En programmation objet, c'est le sommet de la hiérarchie. Tous les composants de Laravel sont fondés sur des interfaces. La version 5 a même vu apparaître un lot de contrats pour étendre de façon sereine le framework.
- Les traits : c'est une façon d'ajouter des propriétés et méthodes à une classe sans passer par l'héritage, ce qui aide à passer outre certaines limitations de l'héritage simple proposé par défaut par PHP [4]

## 2.3 Le modèle MVC

MVC est un patron de conception (*design pattern* en anglais) très répandu pour réaliser des sites web. Ce patron de conception est une solution éprouvée et reconnue permettant de séparer l'affichage des informations, les actions de l'utilisateur et l'accès aux données.

MVC signifie Modèle-Vue-Contrôleur, c'est un modèle qui a été conçu au départ pour des applications dites « client lourd », c'est-à-dire dont la majorité des données sont traitées sur le poste client (par exemple : un traitement de texte comme Word). MVC était tellement puissant pour ces applications « client lourd », qu'il a été massivement adopté comme modèle pour la création d'applications web (dites « client léger »). C'est cette utilisation de MVC pour le web que je vais bien sûr décrire ici.

MVC n'est pas associé à un langage de programmation et il peut être utilisé avec énormément de langages.

MVC est beaucoup utilisé avec PHP par exemple ou même JAVA. En général MVC agit comme une couche supplémentaire (qu'on appelle un « framework ») qui vient se greffer par-dessus un langage.[19]

### 2.3.1 Les composants de modèle MVC

- ✓ **M comme modèle** : Il s'agit des données et de l'état de votre application web (par exemple votre panier de courses sur un site d'e-commerce, ou la liste des produits à acheter,...). En général, ces données sont représentées par un ensemble de classes qui permettent d'accéder à une base de données, mais vos données pourraient tout aussi bien être stockées dans des fichiers plats ou XML, ou même vos classes pourraient utiliser des services web ou autre...
- ✓ **V comme vue**: Le V de MVC signifie la **Vue** et traite de ce qu'on voit à l'écran dans le navigateur web. Il s'agira globalement de code HTML et de CSS. Le but de la vue est de présenter les données issues du modèle mais sans les modifier, sans en interpréter le contenu.

Dans un site web, il y a en général plusieurs vues et une vue correspond bien souvent à une unique page. Dans notre site d'e-commerce par exemple, une vue permettra d'afficher les caractéristiques d'un produit, une autre affichera la page d'accueil, une autre encore permettra de visualiser sa commande, etc.

- ✓ **C comme contrôleur** : le contrôleur fait le lien entre la vue et le modèle, il gère les interactions avec l'utilisateur et détermine quels traitements doivent être effectués pour une action donnée. D'une manière générale, il va utiliser les données du modèle, les traiter en fonction de l'action de l'utilisateur, et les envoyer à la vue afin qu'elle les affiche. [19]

### 2.3.2 L'interaction entre les composants

Voici un petit schéma synthétique pour rappeler comment les différentes couches interagissent entre elles :

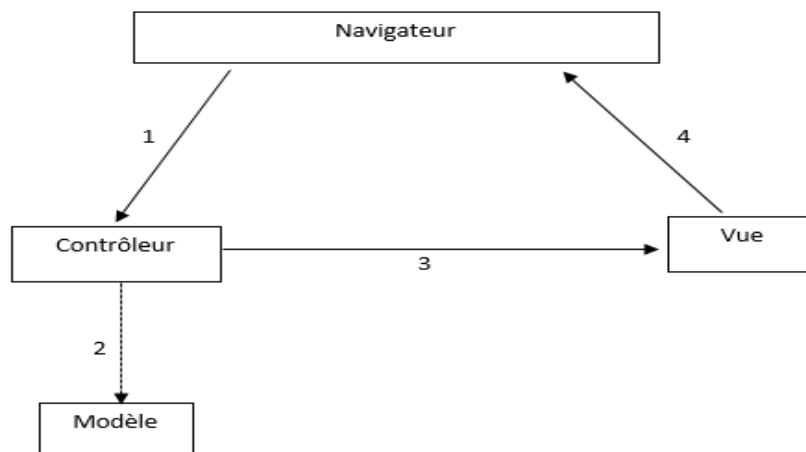


Figure 03 : Schéma MVC

1. Action utilisateur via une requête HTTP

2. Consultation et/ou mise à jour du modèle (facultatif)
3. Le contrôleur décide de la vue à afficher
4. La vue renvoi le HTML au navigateur [19]

### **3. E-commerce**

Le e-commerce ou commerce électronique regroupe l'ensemble des transactions commerciales s'opérant à distance par le biais d'interfaces électroniques et digitales, ce type de commerce englobe essentiellement les transactions commerciales s'effectuant sur Internet à partir des différents types de terminaux (ordinateurs, tablettes, smartphone, etc.). Le e-commerce est devenu le principal canal de la vente à distance ce qui explique le remplacement du terme de « vente par correspondance » par celui de « vente à distance », il ne recouvre qu'une partie des enjeux du commerce sur Internet car la présence des entreprises sur Internet influence également les achats réalisés en points de vente ou par d'autres canaux. [20]

#### **3.1. L'entreprise et son approche du e-commerce**

Depuis l'apparition du commerce en ligne, l'entreprise est confrontée à un choix délicat à savoir rester dans un circuit traditionnel (c'est-à-dire de vente en « face à face »), ou intégrer le e-commerce. La décision que prendra l'entrepreneur aura donc inéluctablement des répercussions sur le plan stratégique. [21]

#### **3.2. Avantages d'un site e-commerce**

##### **3.2.1. Pour les entreprises**

- Il représente un nouveau canal de distribution non négligeable.
- Il permet de se développer sur des marchés, jusque-là inaccessibles avec un magasin physique.
- Il permet de développer l'interactivité entre l'entreprise et ses clients (sons, images, vidéos...)
- Il facilite les achats, sans que l'acheteur n'ait besoin de se déplacer.
- Il permet de proposer à la clientèle des prix moindres, les charges étant réduites.
- Il permet de recueillir des données sur les clients, ce qui permettra de mettre en place des actions de marketing direct [21].



### 3.2.2. Pour les internautes :

- Un site de e-commerce permet aux clients d'effectuer un repérage avant d'acheter en magasin.
- Ce type de site permet aux clients de rechercher le meilleur prix.
- Le client effectue ses achats de son domicile.
- Possibilité d'accéder au magasin de n'importe où dans le monde, 24h/24 et 7j/7.
- Gain de temps non négligeable [21].

## 4. Le centre commercial Beldi

Le centre commercial 'Beldi', est une galerie marchande qui contient un nombre de magasins de différentes catégories, dont l'attractivité attire en consommateurs ces magasins, il est destiné à rendre les moments de shopping agréables pour les visiteurs.

Chaque magasin dans ce centre est géré par un vendeur, ce dernier peut gérer les articles dans son magasin.

Ce centre veut héberger la technologie dans ces transactions en développant une application web qui facilite le processus d'achat pour les clients, et améliore les revenus des magasins et donc du centre commercial, et le développement de ce site va atteindre ces objectifs.

## Conclusion :

Dans ce chapitre, nous avons introduit le processus 2TUP, et rappelé quelques notions sur les applications web, Puis nous avons expliqué le framework qu'on va l'utiliser qui est laravel et pourquoi l'utiliser et le modèle MVC, et enfin nous avons présenté le centre commercial et plus exactement 'Beldi'.

# Chapitre 02

---

## Etude préliminaire

## Introduction

Dans ce chapitre, nous présentons la première étape du modèle en y, qui est l'étude préliminaire, qui sert à recueillir les besoins des utilisateurs et de situer le contexte du système. Nous allons décrire les besoins et les acteurs qui vont interagir avec notre système.

La figure 01 résume la phase de l'étude préliminaire.

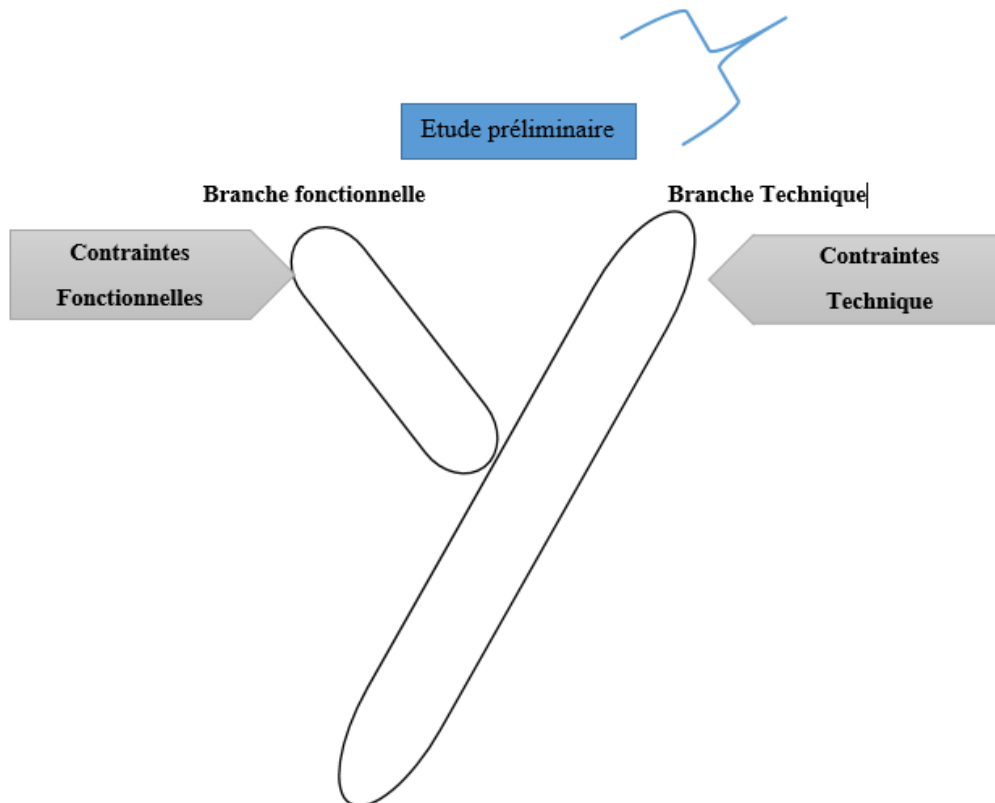


Figure 01 : phase d'étude préliminaire

## 1. Etude préliminaire

L'étude préliminaire (ou pré étude) est la toute première étape de notre processus de développement. Elle survient à la suite d'une décision de démarrage du projet, et consiste à effectuer un premier repérage des besoins fonctionnels et opérationnels, en utilisant principalement le texte, ou des diagrammes très simples. [01]

La figure 02 résume les activités de l'étude préliminaire :

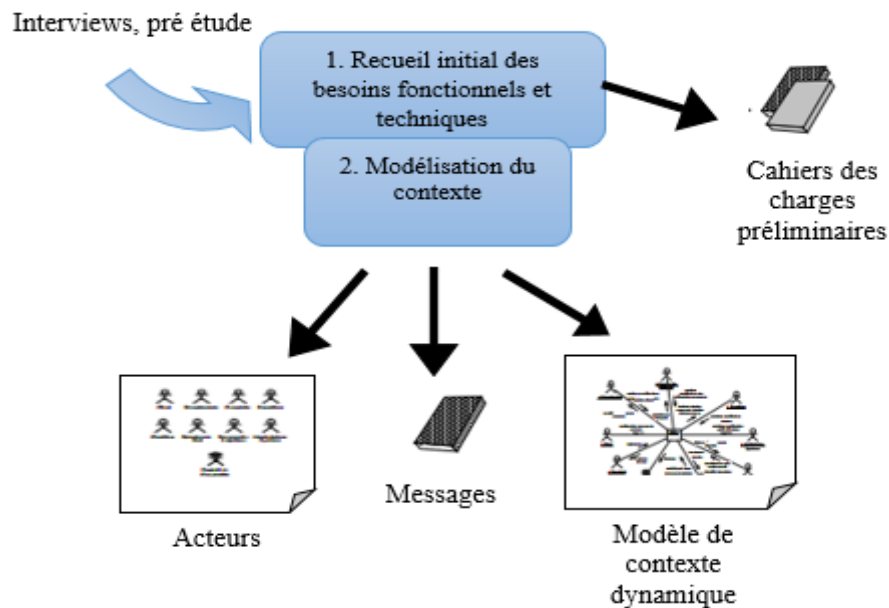


Figure 02 : Résumé des activités et des produits de l'étude préliminaire

## 2. Elaboration du cahier des charges

### 2.1. Présentation du projet

C'est une application web qui aide l'internaute dans son poste de travail d'acheter un article en ligne, c'est à dire de sélectionner le produit voulu pour l'ajouter dans son panier d'achat virtuel, de modifier la quantité des produits dans le panier si il veut ou de retirer carrément un article, après la validation de sa commande et consultation de sa facture il arrive à l'étape de paiement pour conclure le processus d'achat.

### 2.2. Grands choix techniques

On va utiliser une approche itérative et incrémentale, fondée sur le processus en Y.

Nous allons utiliser d'un certain nombre de techniques-clés qui sont principalement :

- Le langage de modélisation UML.
- Le processus de développement en Y (2TUP).
- Le serveur Apache.
- Le Framework "Laravel" (langage de programmation PHP).
- Le système de gestion de bases de données relationnelles MYSQL.
- Internet.

### 2.3. Recueil des besoins fonctionnels

Pour pouvoir établir l'ensemble des besoins fonctionnels de notre logiciel, On peut établir le cahier des charges préliminaire suivant :

- ✓ **Gestion des articles** : ces articles sont mis à jour par le vendeur (ajout ou suppression des articles, modification des prix de la quantité, etc.).
- ✓ **Gestion des commandes des clients** : le client passe sa commande après le choix des articles à acheter, après le vendeur s'occupe de la livraison de ces commandes.
- ✓ **Gestion des informations des clients** : un client peut modifier le mot de passe de son compte.
- ✓ **Gestion des informations des vendeurs** : un vendeur peut modifier les informations de son compte (mot de passe, numéro de sa carte visa, etc.).
- ✓ **Gestion des comptes** : l'administrateur gère les comptes des vendeurs et des clients (ajouter compte vendeur ou supprimer les comptes des vendeurs ou des clients).

### 2.4. Recueil des besoins opérationnels

- ✓ **Sécurité** :
  - Un client connecté via Internet doit être identifié par un mot de passe et n'accède qu'aux informations qui lui concernent.
  - Un administrateur système est chargé de gérer les comptes des utilisateurs.
- ✓ **Volume de données** :
  - Les données du système doivent être traitées en temps réel (la modification des informations du compte, la commande des articles, etc.).

## 3. Description de contexte du système

Une fois ce premier recueil des besoins effectué, la description de contexte du système peut commencer.

Elle consiste en trois activités successives :

- l'identification des acteurs.
- l'identification des messages.
- la réalisation des diagrammes de contexte. [1]

## 3.1. Identification des acteurs

Un acteur représente l'abstraction d'un rôle joué par des entités externes (utilisateur, dispositif matériel ou autre système) qui interagissent directement avec le système étudié. Un acteur peut consulter et/ou modifier directement l'état du système, en émettant et/ou en recevant des messages éventuellement porteurs de données. [1]

On peut distinguer deux types des acteurs :

- **Acteur principal** : nous appelons acteur principal celui pour qui le cas d'utilisation produit un résultat observable.
- **Acteur secondaire** : les acteurs secondaires sont souvent sollicités pour des informations complémentaires, ils peuvent uniquement consulter ou informer le système lors de l'exécution du cas d'utilisation. [2]

Les acteurs de notre système sont :

- **Acteurs principaux** :
  - **Visiteur** : une personne qui cherche sur internet un article pour l'acheter ou juste avoir une idée sur les prix par exemple, ou qui veut créer un compte pour pouvoir acheter en ligne.
  - **Client** : une personne qui a plus d'options par rapport à un visiteur c'est-à-dire elle a déjà créé un compte donc elle peut passer une commande.
  - **Vendeur** : une personne qui possède un compte et gère les articles.
  - **Administrateur** : une personne qui gère les comptes des vendeurs et des clients et voir les statistiques du site.
- **Acteurs secondaires** :
  - **Système de paiement** : Le système de paiement sécurisé est nécessaire au moment du paiement en ligne. Il assure le traitement des flux de paiement du secteur économique et financier.

Les acteurs	Description des besoins fonctionnels
<b>Visiteur</b>	<ul style="list-style-type: none"> <li>• Créer un compte.</li> <li>• Rechercher un article.</li> <li>• Consulter un article.</li> <li>• Gérer le panier.</li> </ul>
<b>Client</b>	<ul style="list-style-type: none"> <li>• Rechercher un article.</li> </ul>

	<ul style="list-style-type: none"> <li>• Consulter un article.</li> <li>• Gérer le panier.</li> <li>• S’authentifier.</li> <li>• Passer commande</li> <li>• Consulter la facture.</li> <li>• Paiement en ligne.</li> <li>• Modifier les informations du compte.</li> </ul>
<b>Vendeur</b>	<ul style="list-style-type: none"> <li>• Modifier les informations du compte.</li> <li>• Gérer les articles.</li> <li>• Traiter commande.</li> </ul>
<b>Administrateur</b>	<ul style="list-style-type: none"> <li>• Gérer les comptes.</li> <li>• Statistique.</li> </ul>

**Tableau 01 : Identification des acteurs**

### 3.2 Identification des messages

Un message représente la spécification d’une communication entre objets qui transporte de l’information avec l’intention de déclencher une activité chez le récepteur. [01]

Le tableau 02 représente l’identification des messages.

<b>Le système émet les messages suivants :</b>	<ul style="list-style-type: none"> <li>• Le formulaire d’inscription.</li> <li>• La liste des articles disponibles.</li> <li>• Le formulaire d’authentification.</li> <li>• La liste des articles recherchés.</li> <li>• La liste des commandes.</li> <li>• le formulaire de la modification du compte.</li> <li>• le formulaire de mise à jour des articles.</li> <li>• la liste des comptes.</li> </ul>
<b>Le système reçoit les messages suivants :</b>	<ul style="list-style-type: none"> <li>• la saisie des informations d’inscription.</li> <li>• la saisie des informations d’authentification.</li> <li>• la saisie des articles à rechercher.</li> <li>• La saisie des informations d’une commande.</li> </ul>

	<ul style="list-style-type: none"> <li>• La saisie des informations de modification d'un compte.</li> <li>• La saisie des informations de modification d'un article.</li> </ul>
--	---

Tableau 02 : Identification des messages

### 3.3. Modélisation du contexte

Tous les messages (système <-> acteurs) identifiés précédemment peuvent être représentés de façon synthétique sur un diagramme, que l'on peut qualifier de diagramme de contexte dynamique en utilisant un diagramme de collaboration de la façon suivante :

- le système étudié est représenté par un objet central ;
- cet objet central est entouré par d'autres objets symbolisant les différents acteurs ;
- des liens relient le système à chacun des acteurs ;
- sur chaque lien sont montrés les messages en entrée et en sortie du système, sans numérotation.

[01]

#### ➤ Diagramme de contexte dynamique

La figure 03 représente le diagramme de contexte dynamique qui montre les messages entre le système et les acteurs.

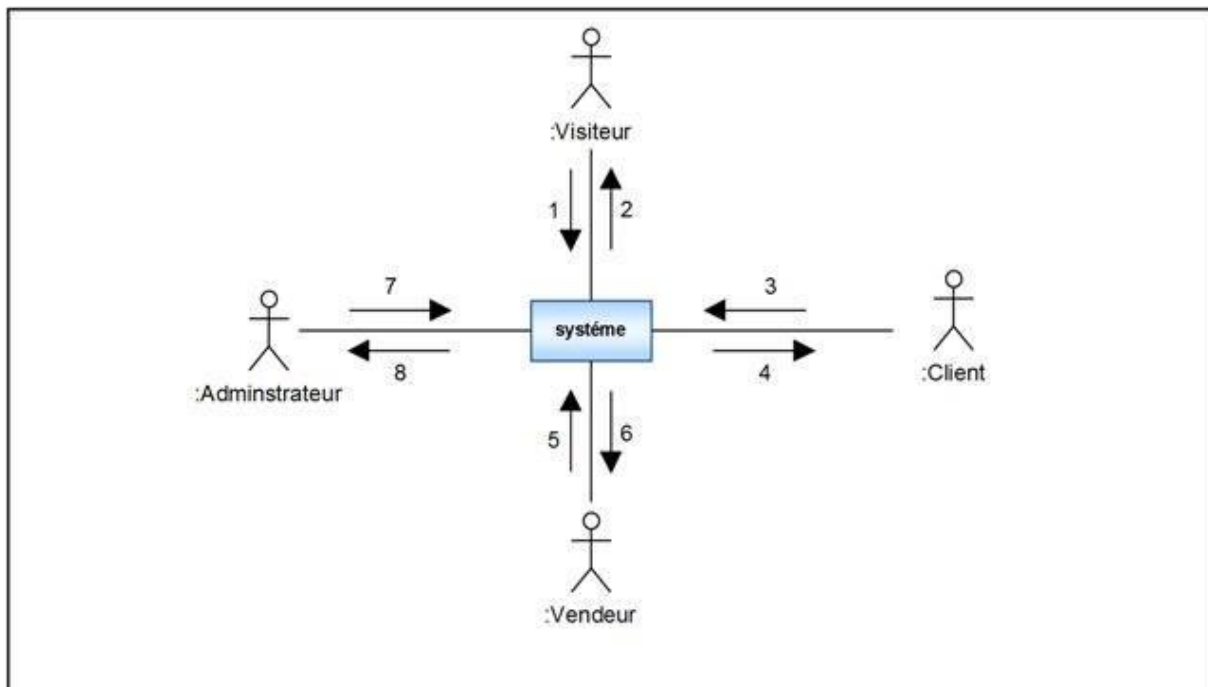


Figure 03 : diagramme de contexte dynamique

#### ➤ Description détaillée des messages



Le tableau suivant montre la description détaillée des messages entre le système et les acteurs.

<b>1 : Visiteur ► Système</b>
<ul style="list-style-type: none"> <li>• Les informations de la recherche.</li> <li>• Les informations de création d'un compte.</li> </ul>
<b>2 : Système ► Visiteur</b>
<ul style="list-style-type: none"> <li>• Le formulaire d'inscription.</li> <li>• Liste des articles disponibles.</li> <li>• Les informations concernant l'état du panier.</li> </ul>
<b>3 : Client ► Système</b>
<ul style="list-style-type: none"> <li>• Les informations de la recherche.</li> <li>• Les informations concernant une commande.</li> </ul>
<b>4 : Système ► Client</b>
<ul style="list-style-type: none"> <li>• Liste des articles disponibles.</li> <li>• Les informations concernant l'état du panier.</li> <li>• Les informations de la facture.</li> </ul>
<b>5 : Vendeur ► Système</b>
<ul style="list-style-type: none"> <li>• La mise à jour des informations sur les articles (suppression, modification, ajout).</li> </ul>
<b>6 : Système ► Vendeur</b>
<ul style="list-style-type: none"> <li>• Les informations concernant les commandes.</li> <li>• Confirmation des mises à jour concernant les articles.</li> </ul>
<b>7 : Administrateur ► Système</b>
<ul style="list-style-type: none"> <li>• Les informations de création d'un compte d'un vendeur.</li> <li>• L'ajout d'un compte vendeur.</li> <li>• Le blocage des comptes des clients ou des vendeurs.</li> </ul>
<b>8 : Système ► Administrateur</b>
<ul style="list-style-type: none"> <li>• Confirmation des informations de création d'un compte d'un vendeur.</li> <li>• Confirmation de la suppression d'un compte.</li> </ul>

**Tableau 03 : Description détaillée des messages**

## Conclusion

Dans cette phase qui est la première étape de notre processus 2TUP, nous avons dégagé les besoins fonctionnels et opérationnels ainsi que l'identification des messages entre objets.

Dans le chapitre suivant nous allons poursuivre la formalisation de ces besoins.

# Chapitre 03

---

## **Capture des besoins fonctionnels**

## Introduction

Dans ce chapitre, nous présentons la deuxième étape du modèle en y, qui est "capture des besoins fonctionnels" pour présenter le point de vue fonctionnel de l'architecture système.

La figure 01 représente la phase de capture des besoins fonctionnels.

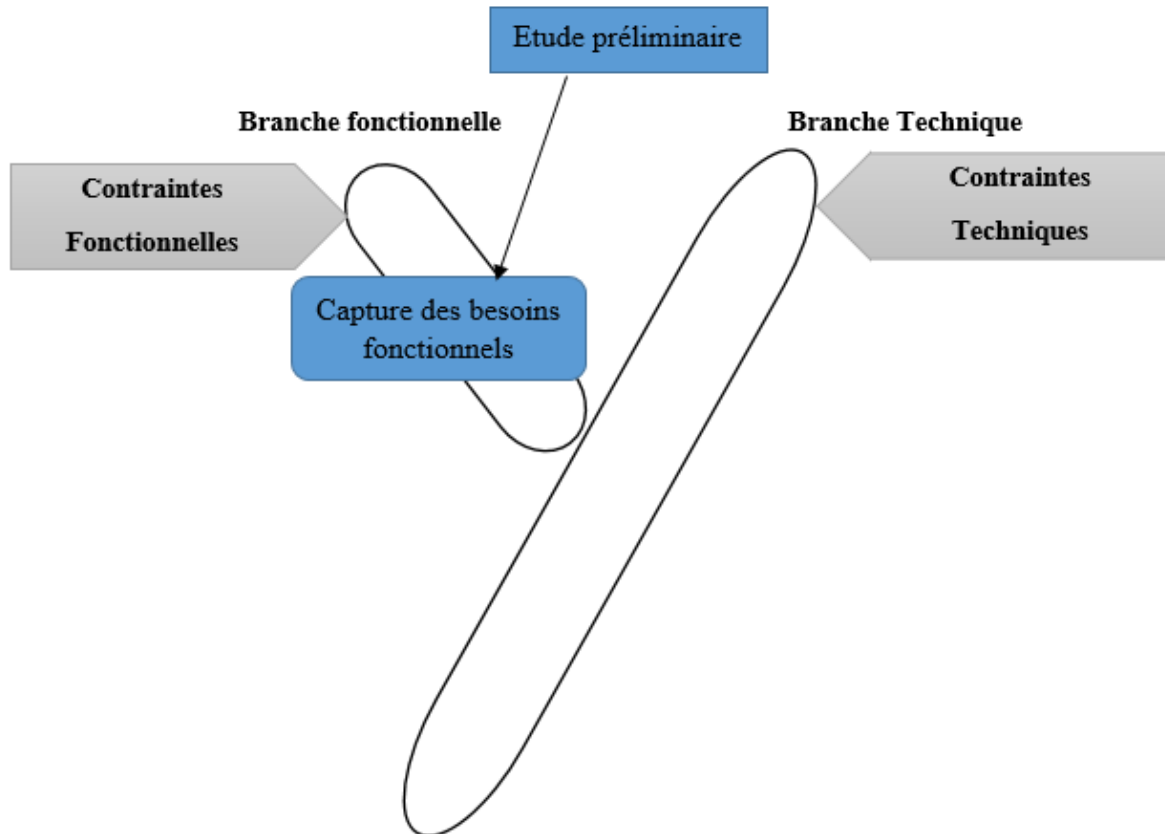


Figure 01 : Phase de capture des besoins fonctionnels

### 1. Capture des besoins fonctionnels

La capture des besoins fonctionnels est la première étape de la branche gauche du cycle en Y.

Elle formalise et détaille ce qui a été ébauché au cours de l'étude préliminaire. [01]

La figure 02 représente la démarche de la phase de capture des besoins fonctionnels.

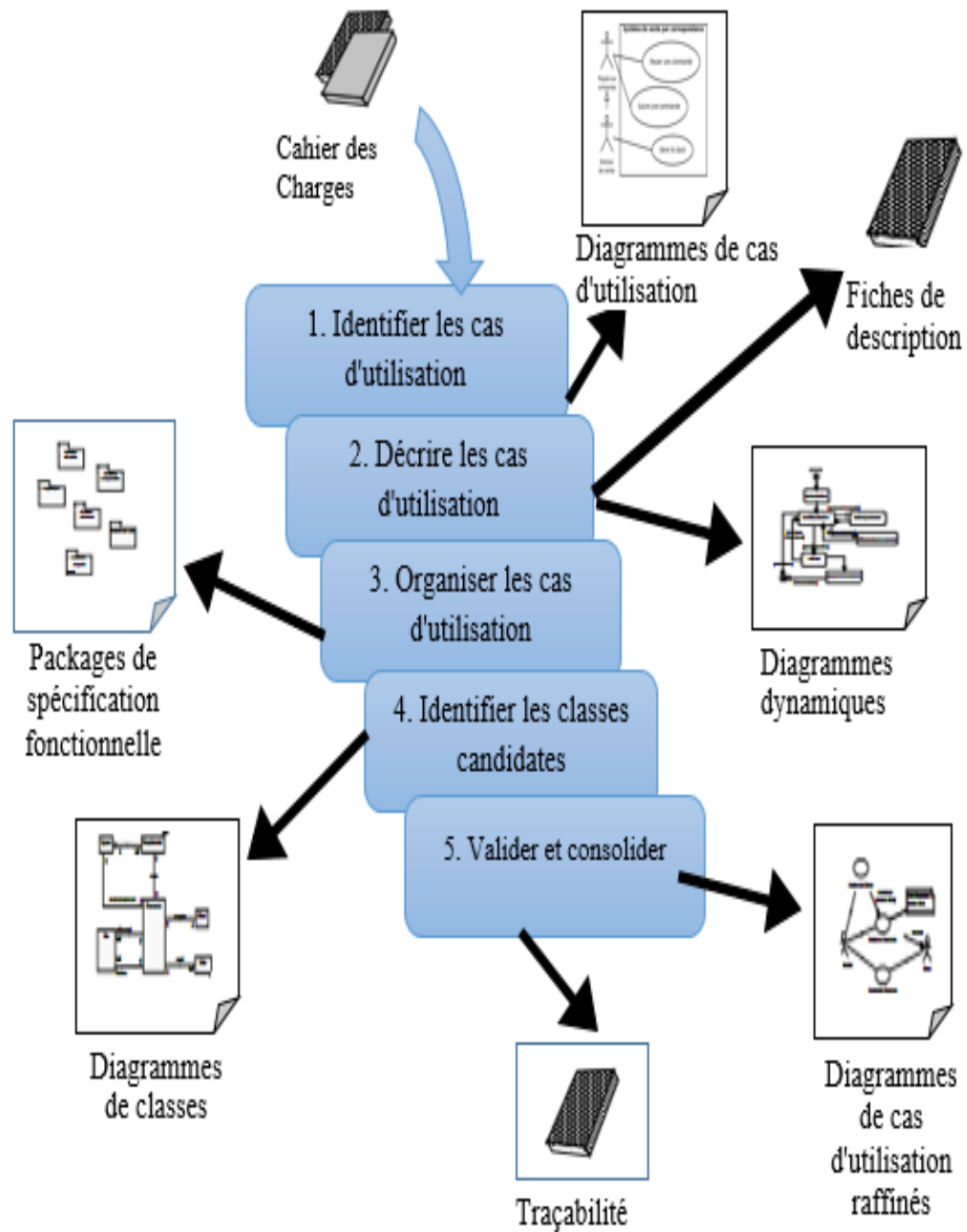


Figure 02 : Démarche de capture des besoins fonctionnels

## 2. Identifier les cas d'utilisation

Un cas d'utilisation (use case) représente un ensemble de séquences d'actions réalisées par le système et produisant un résultat observable intéressant pour un acteur particulier. [01]

### 2.1. Liste préliminaire des cas d'utilisation

Le tableau 01 représente la liste préliminaire des cas d'utilisation, et montre les acteurs pour chaque cas, et les messages émis et reçus par les acteurs.

Cas d'utilisation	Acteur principal / secondaires	Message(s) émis / reçus par les acteurs
créer compte	Visiteur	émet : les informations personnelles. reçoit : un compte client.
Consulter article	Visiteur/client	reçoit : la liste des articles disponible.
Rechercher article	Visiteur/client	émet : mot clé. reçoit : liste d'article recherché.
Gérer panier	Visiteur/ Client	reçoit : l'état du panier.
S'authentifier	Client/ Vendeur/ Administrateur	émet : email et le mot de passe. reçoit : un compte client / vendeur / administrateur.
Passer commande	Client	émet : les informations de la commande. reçoit : message de confirmation.
Consulter facture	Client	reçoit : les informations de la facture.
Paiement en ligne	Client	émet : les informations du compte bancaire. reçoit : message de confirmation.
Modifier les informations du compte	Client / Vendeur	émet : les informations à modifier. reçoit : compte mis à jour.
Gérer les articles	Vendeur	émet : mise à jour des articles. reçoit : confirmation de la mise à jour.
Traiter commande	Vendeur	reçoit : les commandes des clients.
Gérer les comptes	Administrateur	émet : mise à jour des comptes reçoit : confirmation de la mise à jour.
Statistique	Administrateur	reçoit : les statistiques.

Tableau 01 : Liste préliminaire des cas d'utilisation

2.2. Diagramme de cas d'utilisation

Nous présentons ci-dessous le diagramme de cas d'utilisation

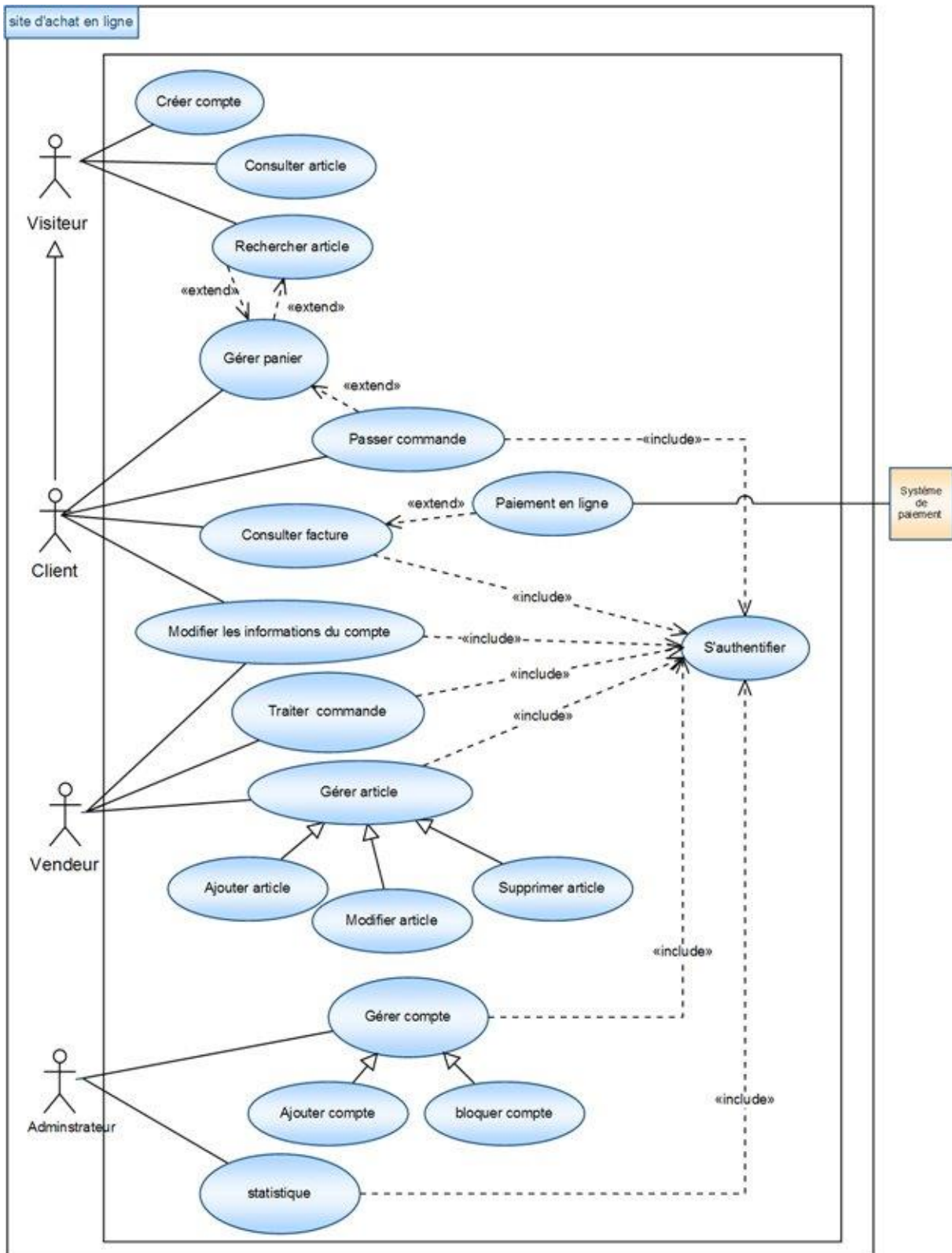


Figure 03 : diagramme de cas d'utilisation

### 3. Description détaillée des cas d'utilisation

Nous allons détailler les cas d'utilisation en utilisant la description textuelle, les diagrammes de séquences et les diagrammes d'activité.

#### 3.1. Créer compte

- Description textuelle du cas d'utilisation

Cas d'utilisation	Créer compte
Objectif	S'inscrire sur le site
Acteur	Visiteur
Précondition	Néant
Post condition	Un compte client est créé
Scénario nominal	<ol style="list-style-type: none"> <li>1. Le visiteur demande de créer un compte.</li> <li>2. Le système affiche le formulaire d'inscription.</li> <li>3. Le visiteur saisit les informations (nom, prénom, email, mot de passe.)</li> <li>4. Le visiteur valide les informations saisies.</li> <li>5. Le système vérifie la validité des informations, et affiche un message de confirmation</li> </ol>
Scénario alternatif	<p><b>5.1</b> S'il y a des erreurs :</p> <p>Un message d'erreur est affiché pour informer le visiteur que les informations saisies ne sont pas valide (champ vide, informations erronées, etc.).</p>

Tableau 02 : Description textuelle du cas d'utilisation « Créer compte ».

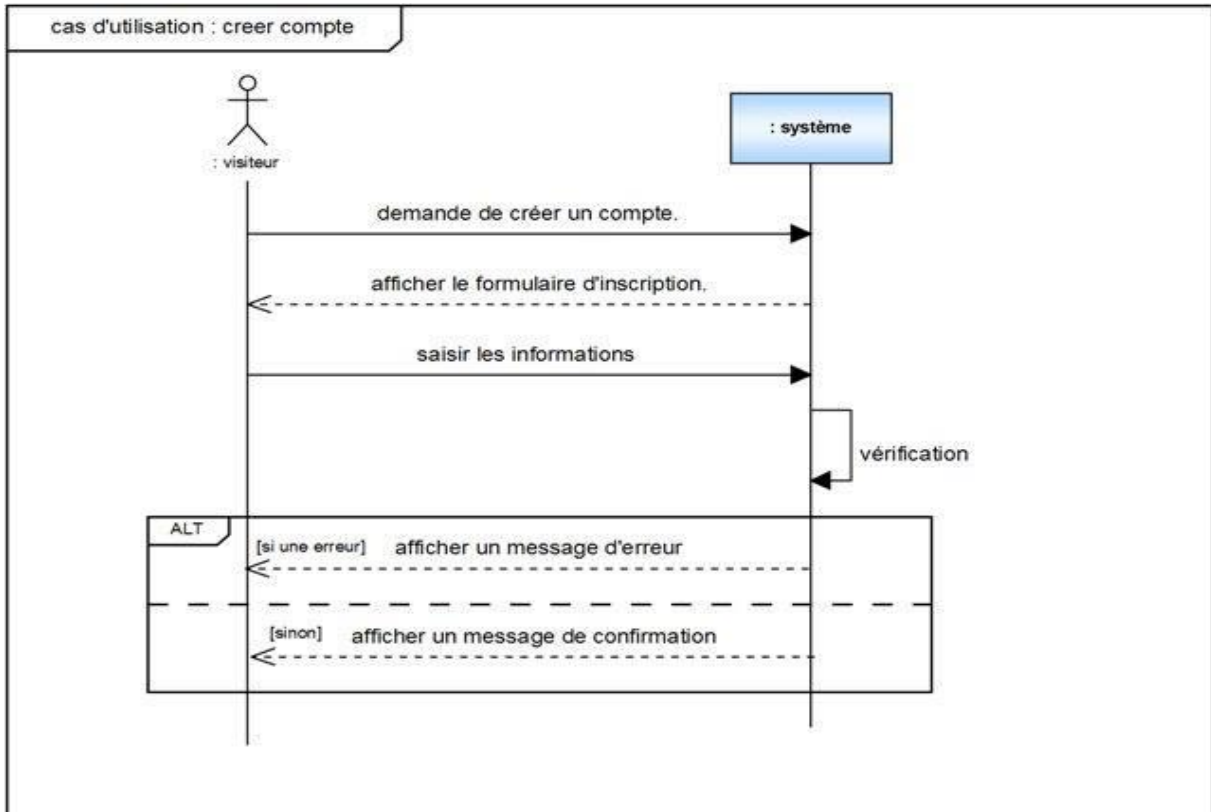


Figure 04 : Diagramme de séquence du cas d'utilisation « créer compte »

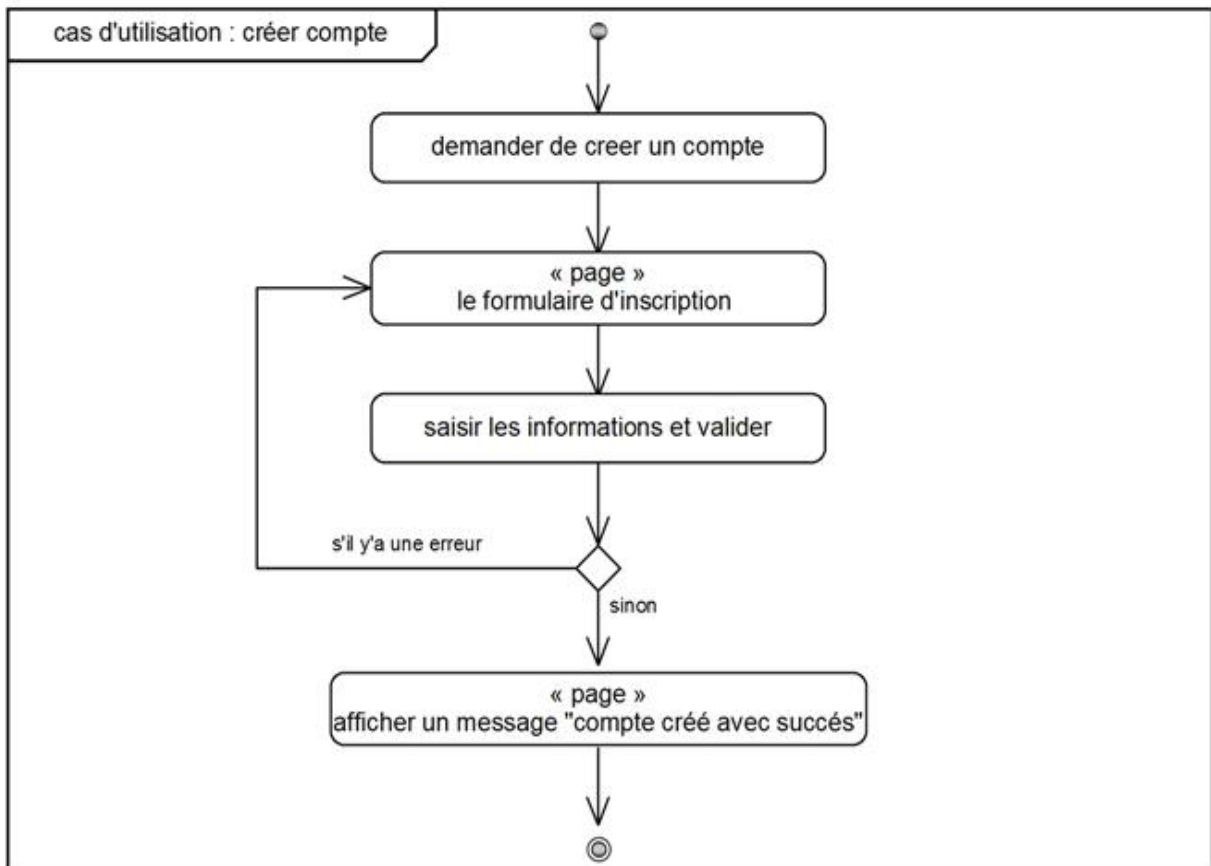


Figure 05 : Diagramme d'activité du cas d'utilisations « créer compte »



3.2. Consulter article

- Description textuelle du cas d'utilisation

Cas d'utilisation	Consulter article
objectif	Permet de consulter le contenu du site.
acteur	Visiteur, client
Précondition	Néant
Post-condition	L'acteur consulte une liste d'articles.
Scénario nominal	<ol style="list-style-type: none"> <li>1. L'acteur demande de consulter un article.</li> <li>2. Le système affiche la page demandée.</li> </ol>
Scénario alternatif	Néant

Tableau 03 : Description textuelle du cas d'utilisation « Consulter article ».

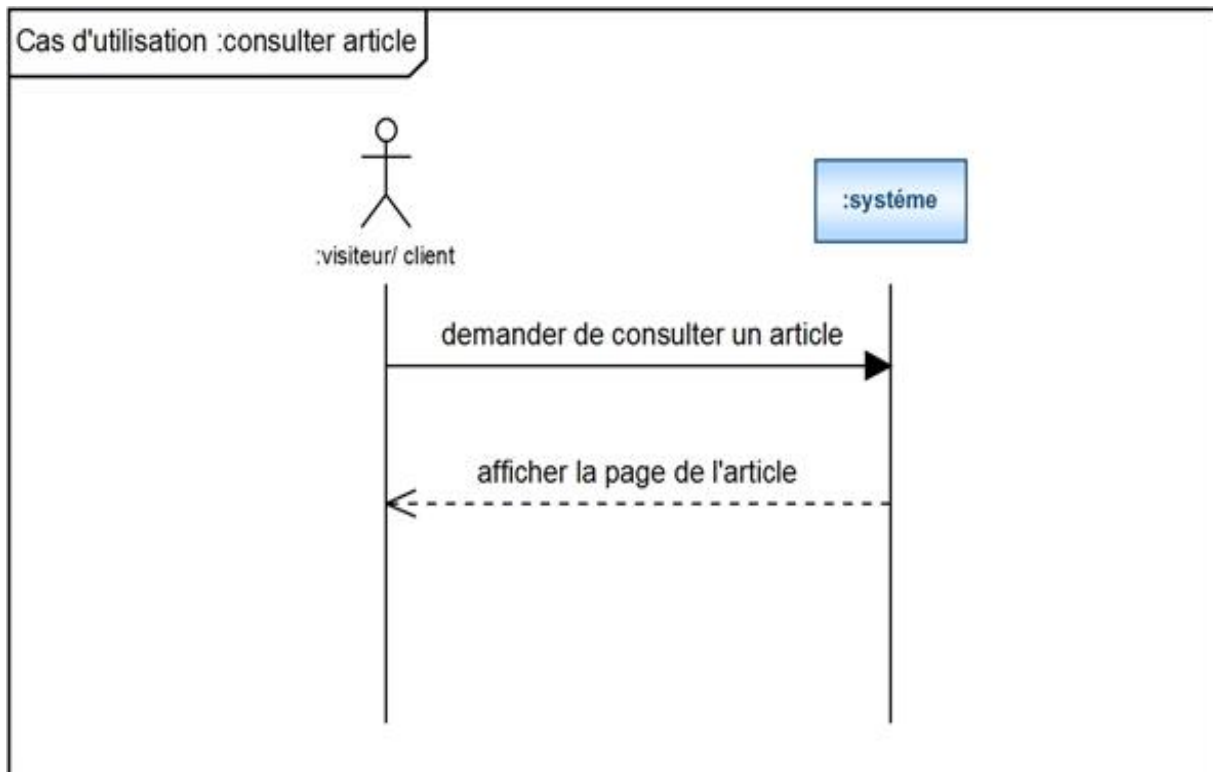


Figure 06 : Diagramme de séquence du cas d'utilisation « Consulter article »

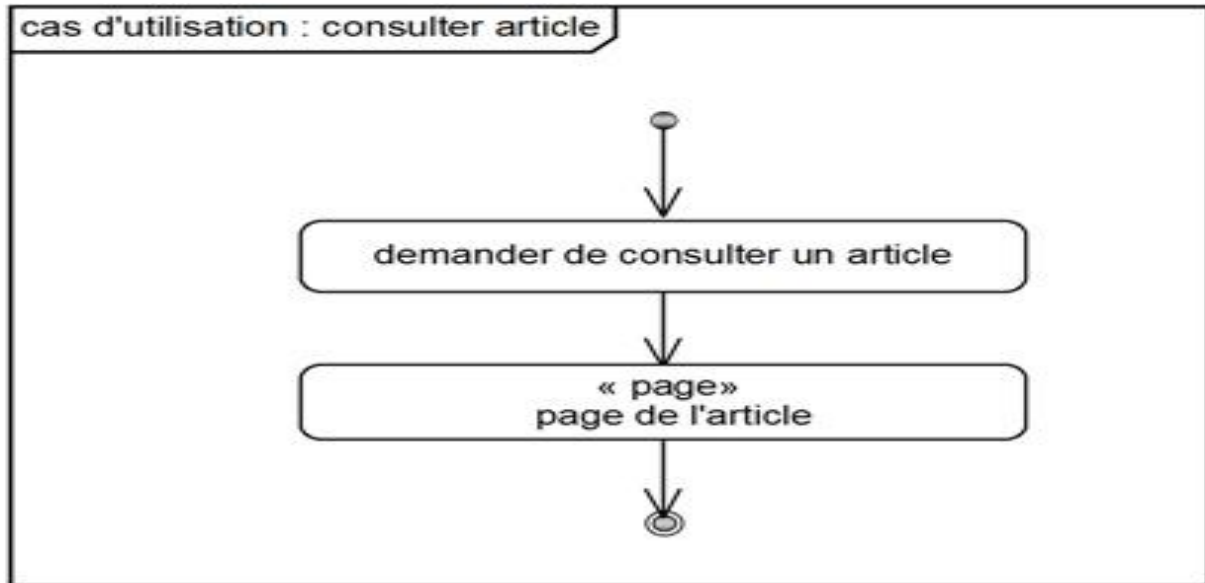


Figure 07 : Diagramme d'activité du cas d'utilisation « Consulter article »

### 3.3. Rechercher article

➤ Description textuelle du cas d'utilisation

Cas d'utilisation	Rechercher article.
Objectif	Lancer une recherche pour trouver un article.
Acteur	Visiteur, Client.
Précondition	Néant.
Post condition	La liste des articles recherchés.
Scénario nominal	<ol style="list-style-type: none"> <li>1. L'acteur saisit l'article à rechercher.</li> <li>2. L'acteur valide les informations saisies.</li> <li>3. Le système vérifie la validité des informations.</li> <li>4. Le système affiche les articles recherchés.</li> </ol>
Scénario alternatif	<p><b>3.1</b> S'il y'a un article non disponible :</p> <p>Afficher un message que l'article recherché par l'acteur est non disponible.</p>

Tableau 04 : Description textuelle du cas d'utilisation « Rechercher article ».

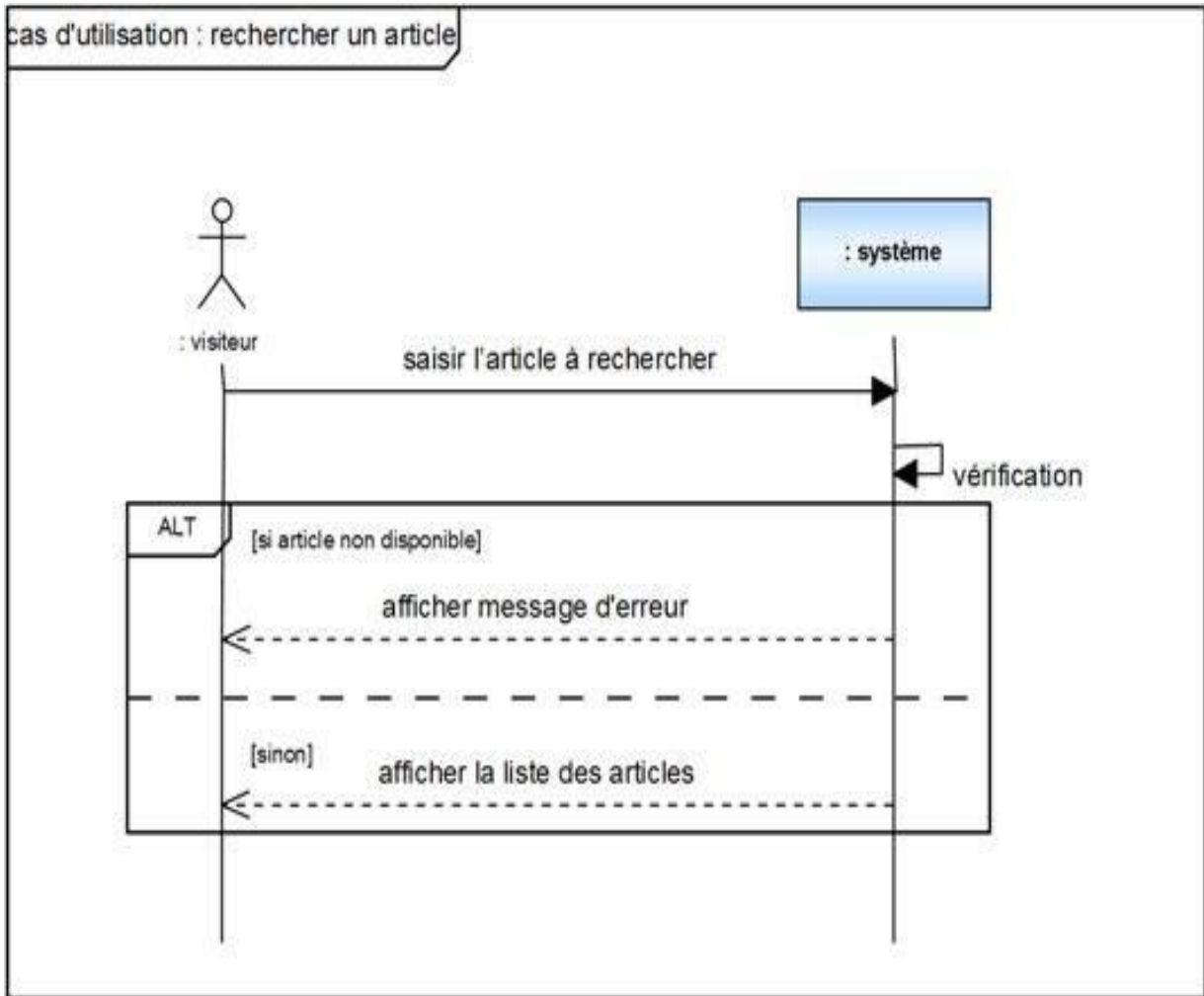


Figure 08 : Diagramme de séquence du cas d'utilisation « Rechercher article »

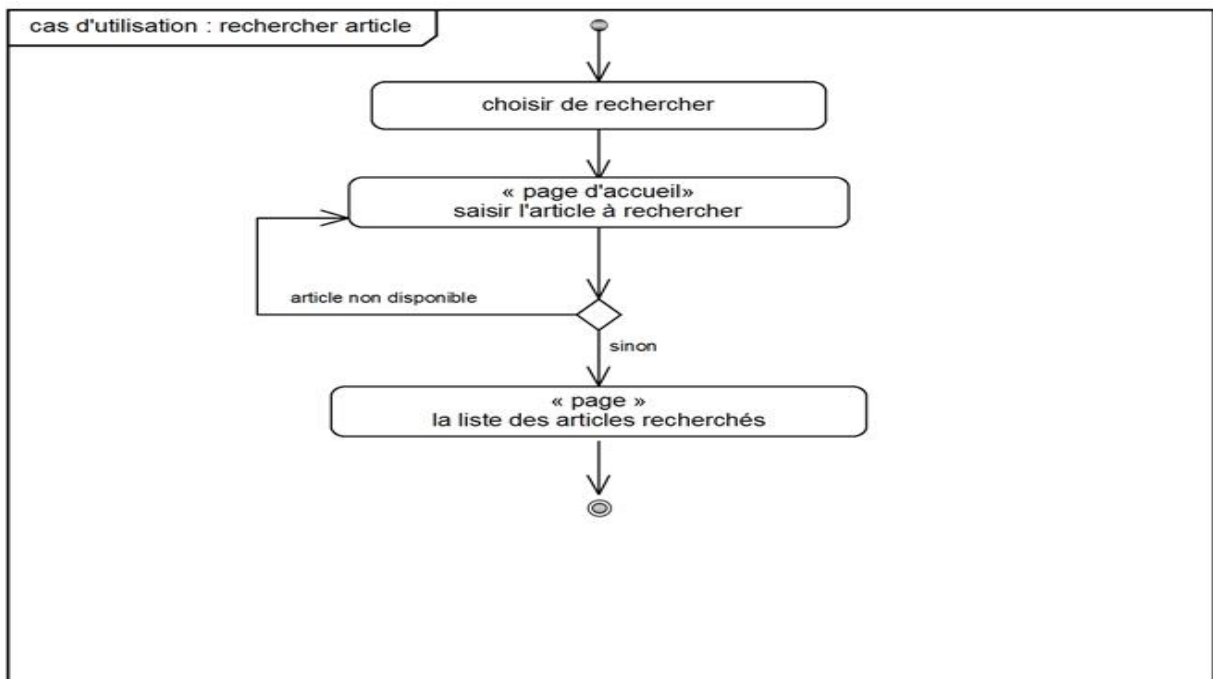


Figure 09 : Diagramme d'activité du cas d'utilisation « Rechercher article »

3.4. Gérer panier

➤ Description textuelle du cas d'utilisation

Cas d'utilisation	Gérer panier
Objectif	La mise à jour des articles dans le panier.
Acteur	Visiteur, client
Pré- condition	Néant
Post-condition	Mise à jour sur les articles (modifier/retirer).
Scénario nominal	<ol style="list-style-type: none"> <li>1. Le client demande l'accès au panier.</li> <li>2. Le système affiche les articles dans son panier.</li> <li>3. Le client met à jour son panier (modifier, retirer, ou ajouter des articles.) et valide.</li> <li>4. Le système confirme cette mise à jour.</li> </ol>
Scénario alternatif	<ol style="list-style-type: none"> <li>3.1 S'il y'a une erreur</li> <li>3.2 Un message d'erreur est affiché sur l'écran.</li> </ol>

Tableau 05 : Description textuelle du cas d'utilisation « Gérer panier ».

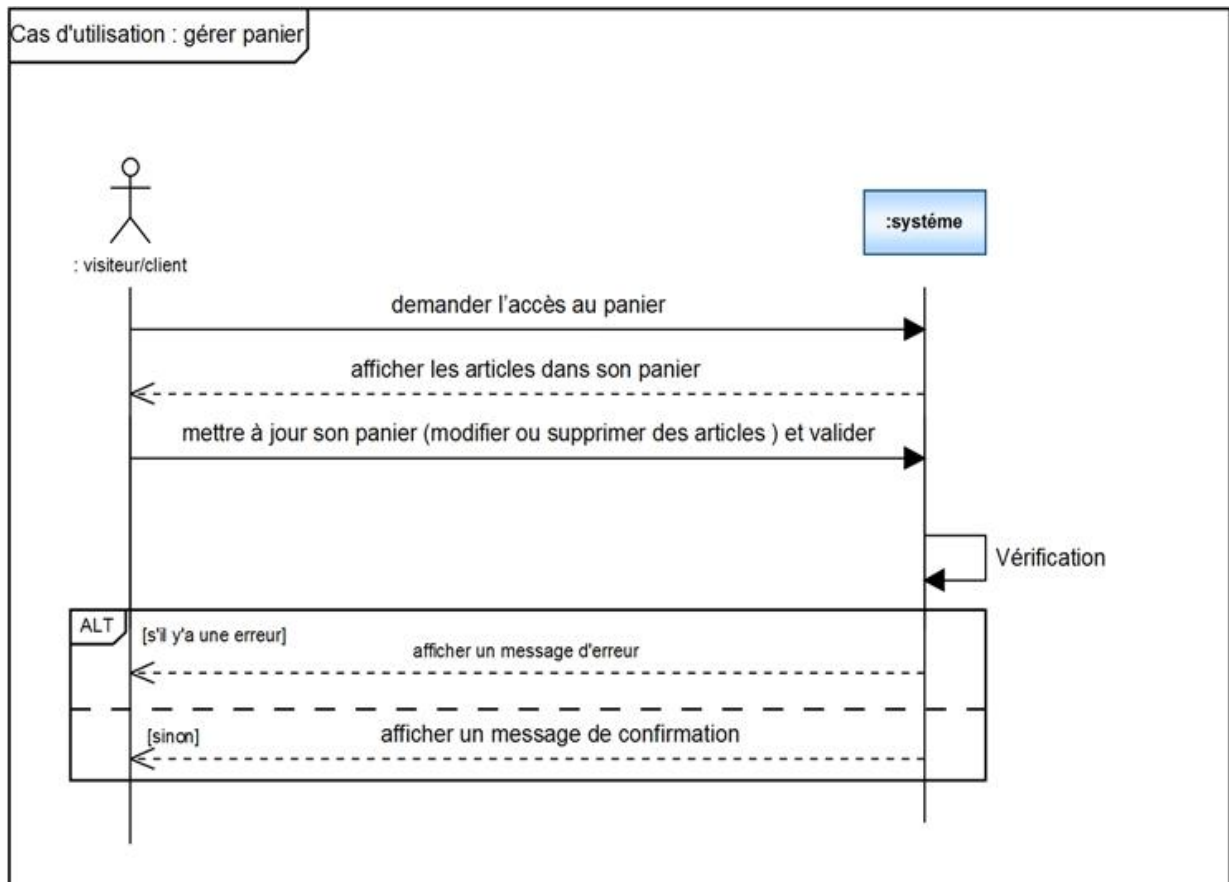


Figure 10 : Diagramme de séquence du cas d'utilisation « Gérer panier »

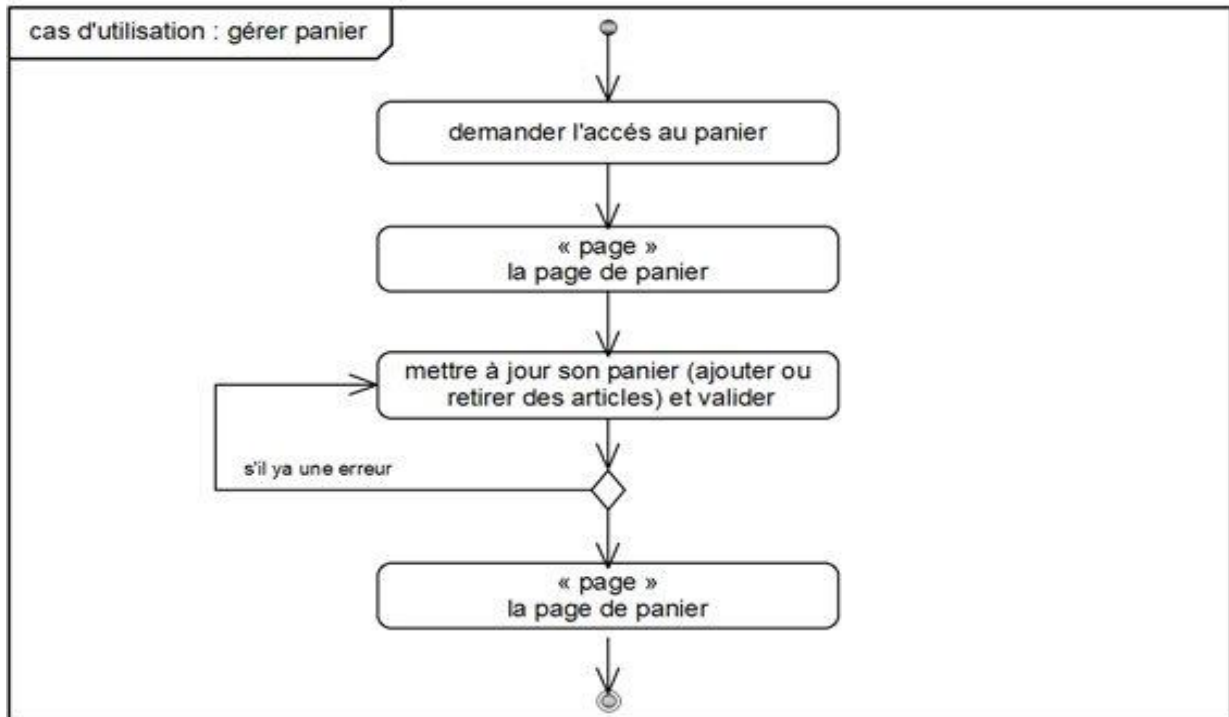


Figure 11 : Diagramme d’activité du cas d’utilisation « Gérer panier »

### 3.5. S’authentifier

➤ Description textuelle du cas d’utilisation

Cas d’utilisation	S’authentifier
objectif	Vérifier la validité du compte
acteur	Client, Vendeur, Administrateur.
Pré- condition	L’acteur possède un compte.
Post condition	Néant
Scénario nominal	<ol style="list-style-type: none"> <li>1. L’acteur demande de s’authentifier.</li> <li>2. Le système lui affiche la page d’authentification.</li> <li>3. L’acteur remplit le formulaire (email, mot de passe).</li> <li>4. L’acteur valide les informations saisies.</li> <li>5. Le système vérifie les informations saisies par l’utilisateur.</li> <li>6. Afficher la page d’accueil.</li> </ol>
Scénario alternatif	<p><b>5.1</b> S’il y a des erreurs</p> <p>Un message d’erreur est affiché sur l’écran que les informations saisies non valides.</p>

Tableau 06 : Description textuelle du cas d’utilisation « S’authentifier ».

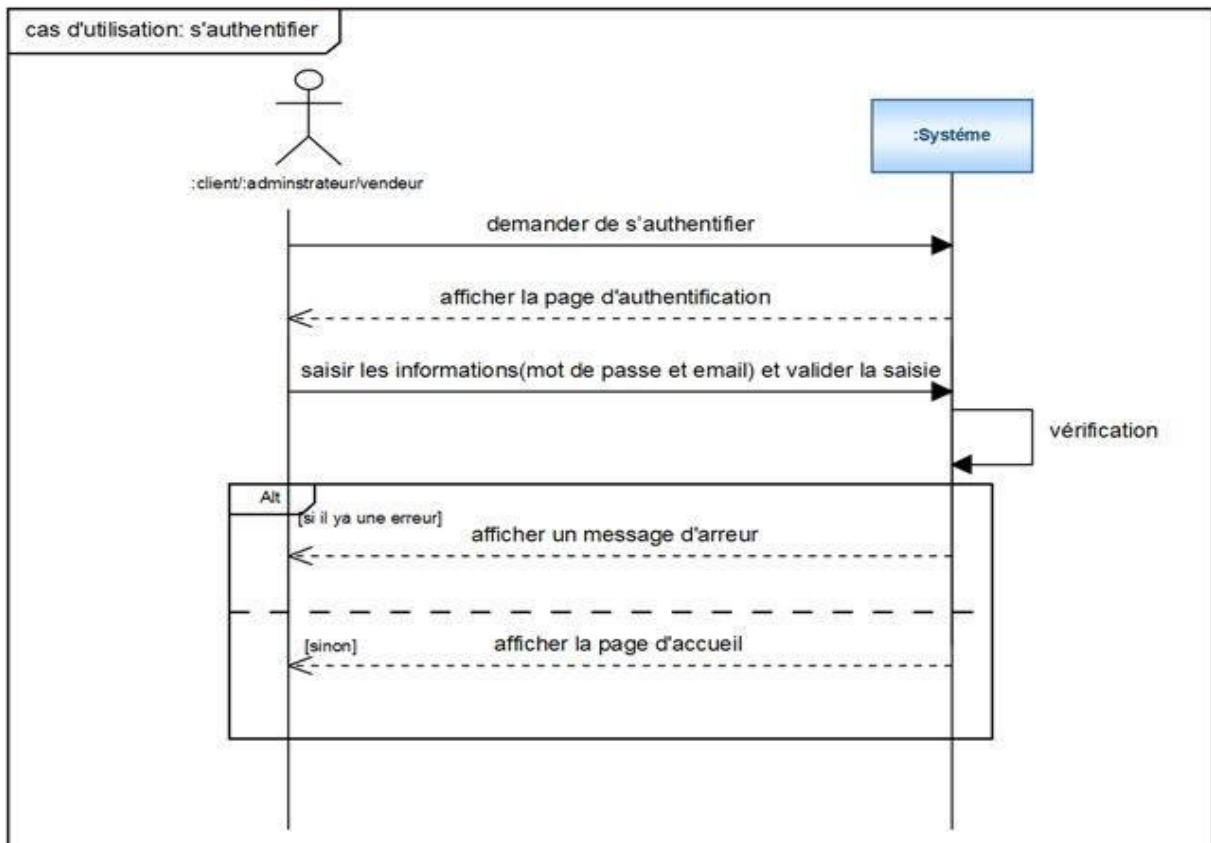


Figure 12 : Diagramme de séquence du cas d'utilisation « s'authentifier »

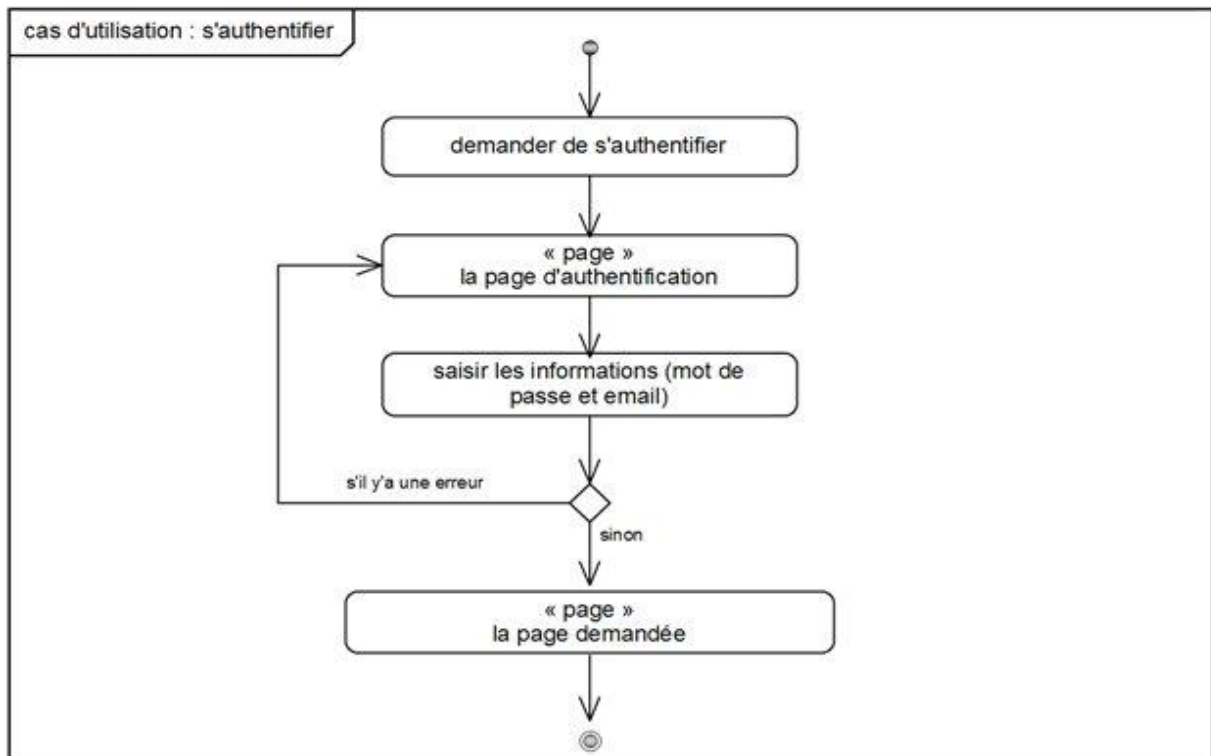


Figure 13 : Diagramme d'activité du cas d'utilisation « S'authentifier »

### 3.6. Passer commande

- Description textuelle du cas d'utilisation

Cas d'utilisation	Passer commande
Objectif	Acheter un article
Acteur	Client
Précondition	Client s'authentifie. Client à gérer son panier.
Post condition	Une commande est ajoutée.
Scénario nominal	<ol style="list-style-type: none"> <li>1. Le client demande de passer une commande.</li> <li>2. Le système affiche le formulaire d'une commande.</li> <li>3. Le client saisit les informations nécessaires.</li> <li>4. Le client valide les informations saisies.</li> <li>5. Le système vérifie la validité des informations.</li> <li>6. Afficher la liste des commandes.</li> </ol>
Scénario alternatif	<p><b>5.1</b> s'il y'a une erreur un message d'erreur est affiché pour le client que les informations saisies ne sont pas valide (champ vide, informations erronées, etc.).</p>

Tableau 07 : Description textuelle du cas d'utilisation « Passer commande ».

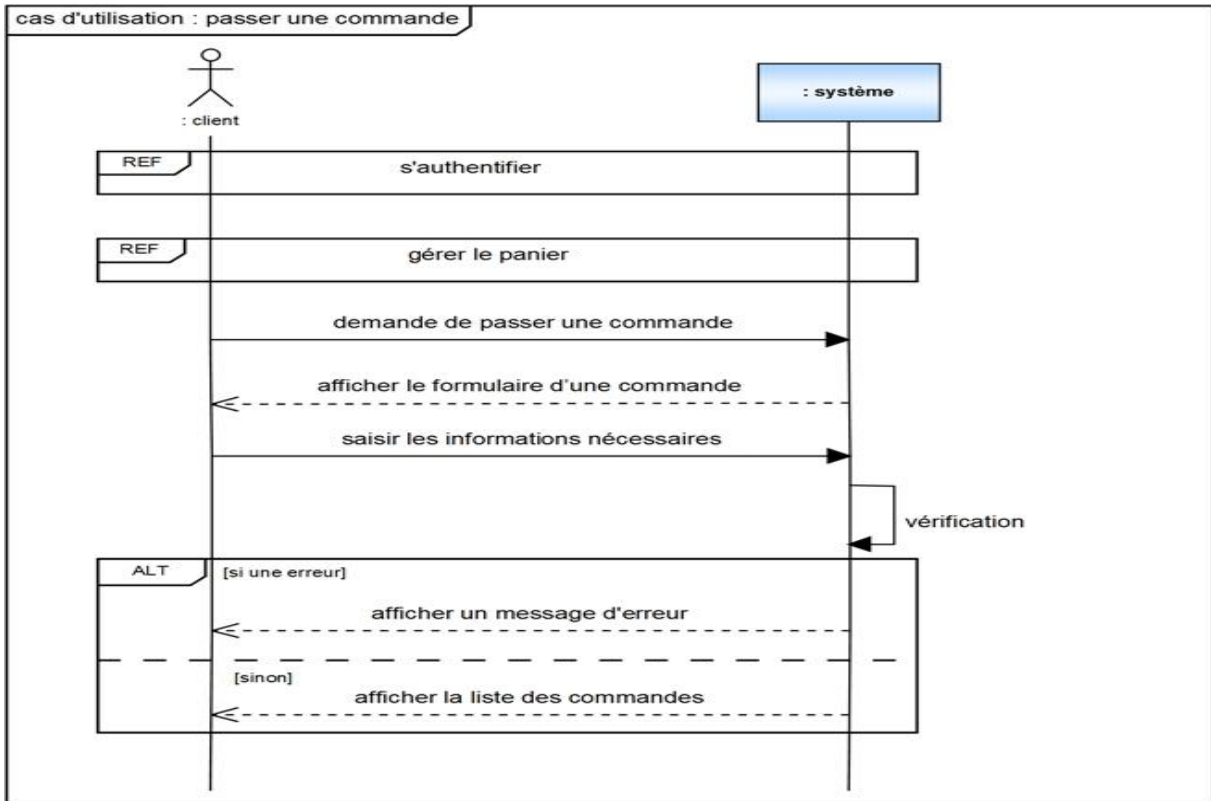


Figure 14 : Diagramme de séquence du cas d'utilisation « Passer commande »

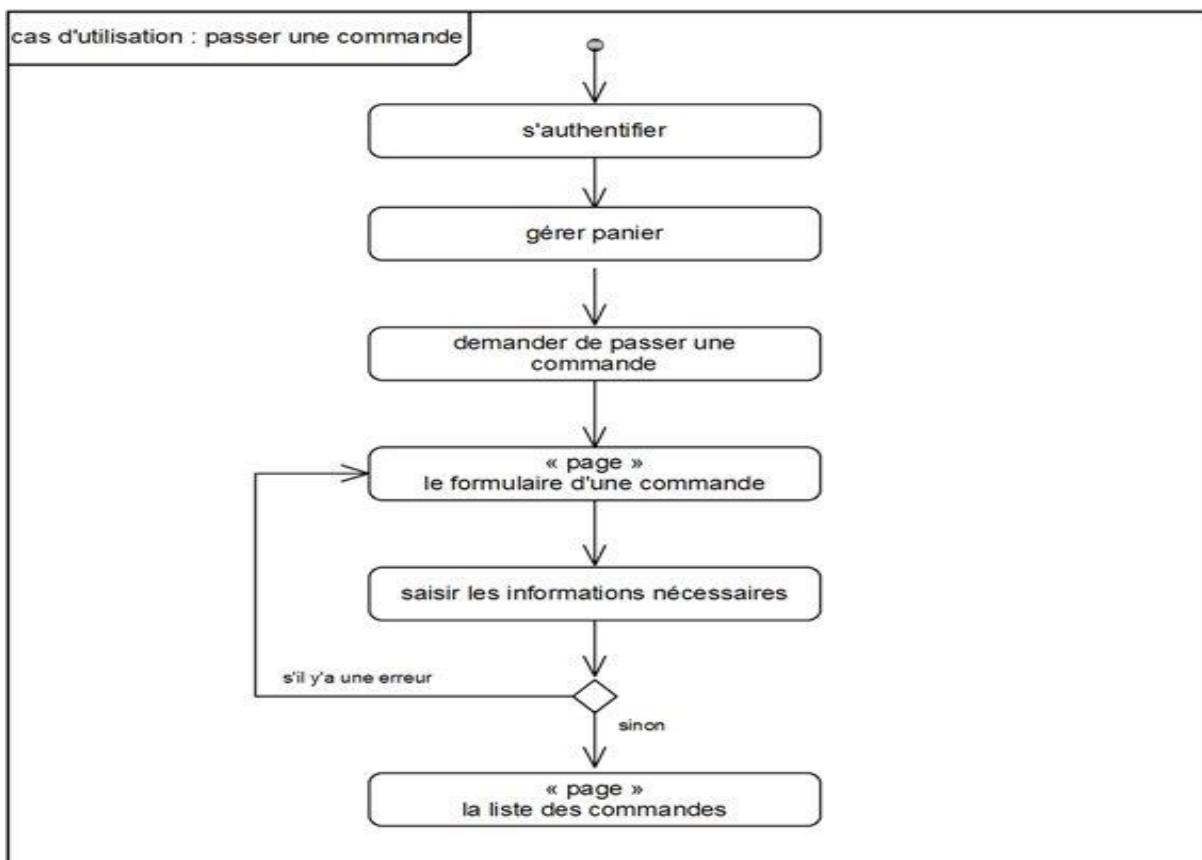


Figure 15 : Diagramme d'activité du cas d'utilisation « Passer commande »



3.7. Consulter facture

- Description textuelle du cas d'utilisation

Cas d'utilisation	Consulter facture
objectif	Permet de consulter la facture
Acteur	Client
Pré- condition	-Le client s'authentifie.
Post condition	Néant
Scénario nominal	<ol style="list-style-type: none"> <li>1. Le client demande de consulter la facture.</li> <li>2. Le système affiche la facture avec les différentes informations concernant (les articles, les prix unitaires et le montant total etc.).</li> </ol>
Scénario alternatif	Néant

Tableau 08 : Description textuelle du cas d'utilisation « Consulter facture».

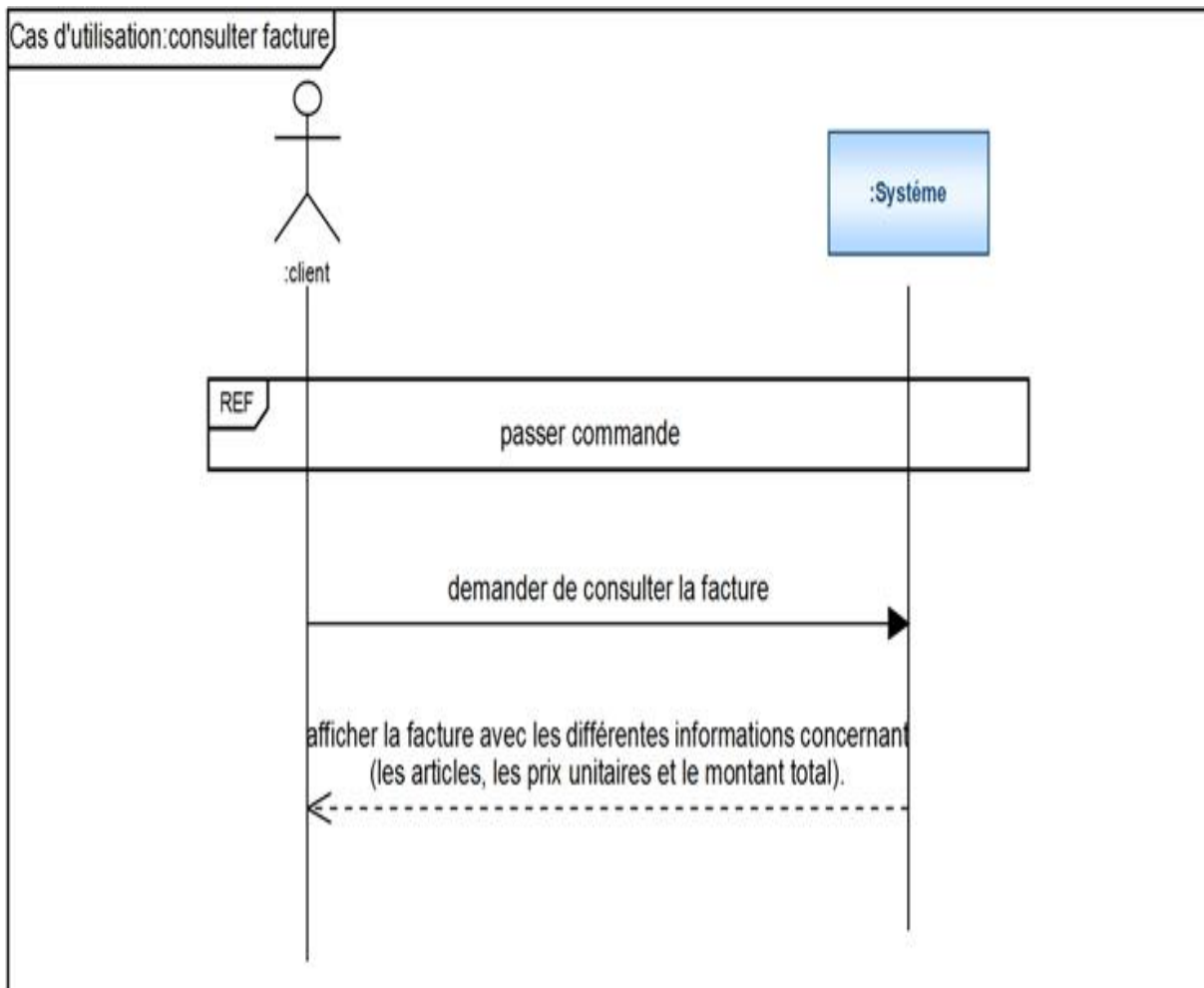


Figure 16 : Diagramme de séquence du cas d'utilisation « Consulter facture »

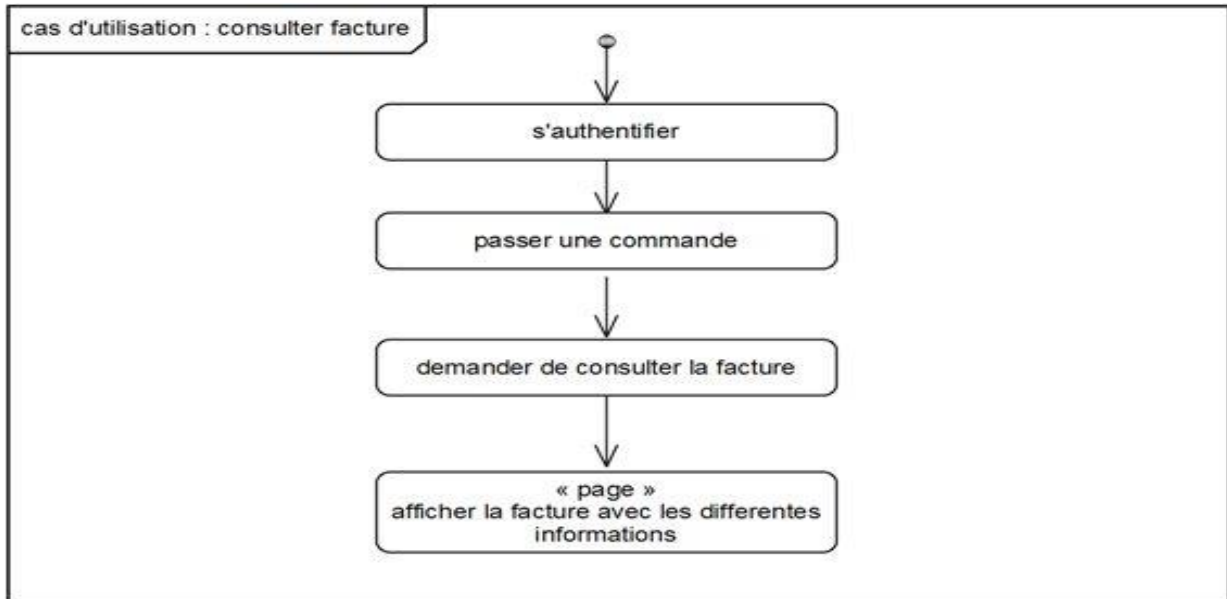


Figure 17 : Diagramme d’activité du cas d’utilisation « Consulter facture »

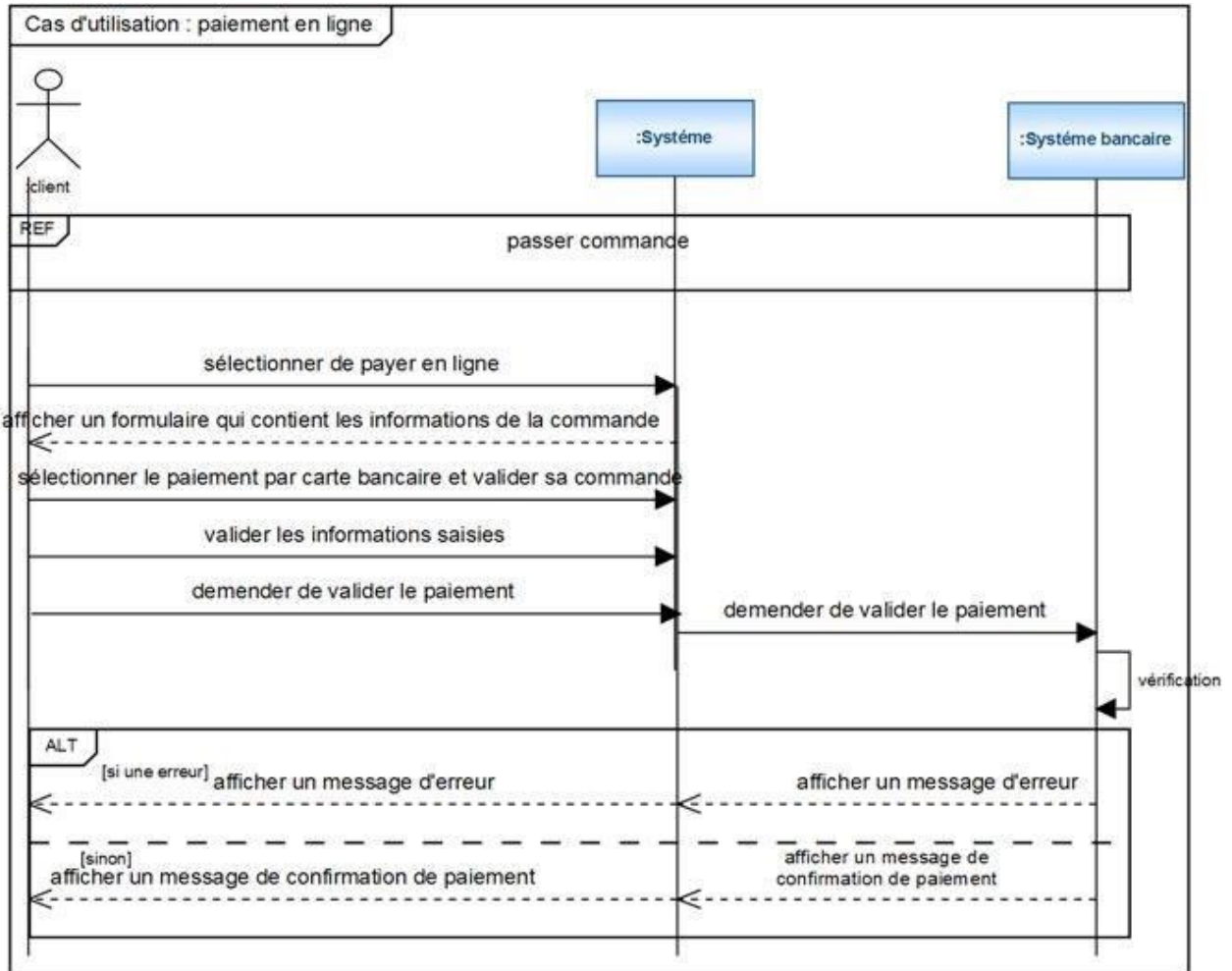
### 3.8. Paiement en ligne

➤ Description textuelle du cas d’utilisation

Cas d’utilisation	Paiement en ligne
objectif	Permet de faire le paiement.
Acteur	Client
Pré- condition	-Le client s’authentifie. -La commande est déjà passée.
Post condition	Néant
Scénario nominal	<ol style="list-style-type: none"> <li>1. Le client sélectionne l’option de paiement en ligne.</li> <li>2. Le système affiche un formulaire qui contient les informations de la commande. « les informations personnelles »</li> <li>3. Le client sélectionne le paiement par carte bancaire et valide sa commande.</li> <li>4. Le client valide les informations saisies.</li> <li>5. Le système envoie les informations au système externe de paiement.</li> <li>6. Le système de paiement vérifie la validité de la carte.</li> <li>7. Le système envoi une confirmation.</li> </ol>

Scénario alternatif	<p><b>6.1</b> S'il y a des erreurs</p> <p>Un message d'erreur est affiché le système de paiement détecte que les informations sur la carte sont incomplètes ou erronées</p>
---------------------	---

**Tableau 09 : Description textuelle du cas d'utilisation « Paiement en ligne ».**



**Figure 18 : Diagramme de séquence du cas d'utilisation « Paiement en ligne »**

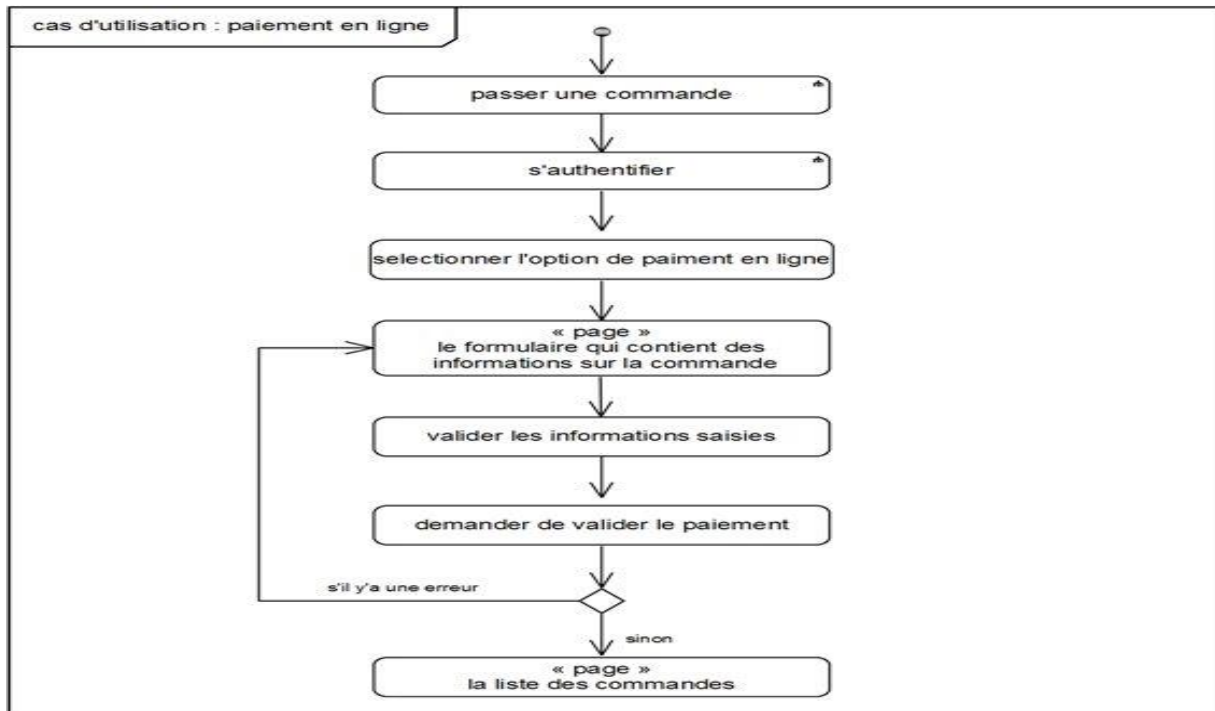


Figure 19 : Diagramme d’activité du cas d’utilisation « Paiement en ligne »

### 3.9. Modifier les informations du compte

➤ Description textuelle du cas d’utilisation

Cas d’utilisation	Modifier les informations du compte.
Objectif	Modifier les informations personnelles de l’acteur.
Acteur	Client, vendeur.
Précondition	Client s’authentifie.
Post condition	Le compte est modifié.
Scénario nominal	<ol style="list-style-type: none"> <li>1. L’acteur demande de modifier son compte.</li> <li>2. Le système affiche le formulaire de la modification du compte.</li> <li>3. L’acteur fait la modification du compte.</li> <li>4. L’acteur valide les informations modifiées.</li> <li>5. Le système vérifie la validité des informations et fait la mise à jour.</li> <li>6. Le système confirme la modification.</li> </ol>
Scénario alternatif	<p><b>5.1</b> S’il y a des erreurs :</p> <p>Un message d’erreur est affiché pour le client que les informations saisies ne sont pas valide (champ vide, informations erroné, etc.).</p>

Tableau 10 : Description textuelle du cas d’utilisation « Modifier les informations du compte ».

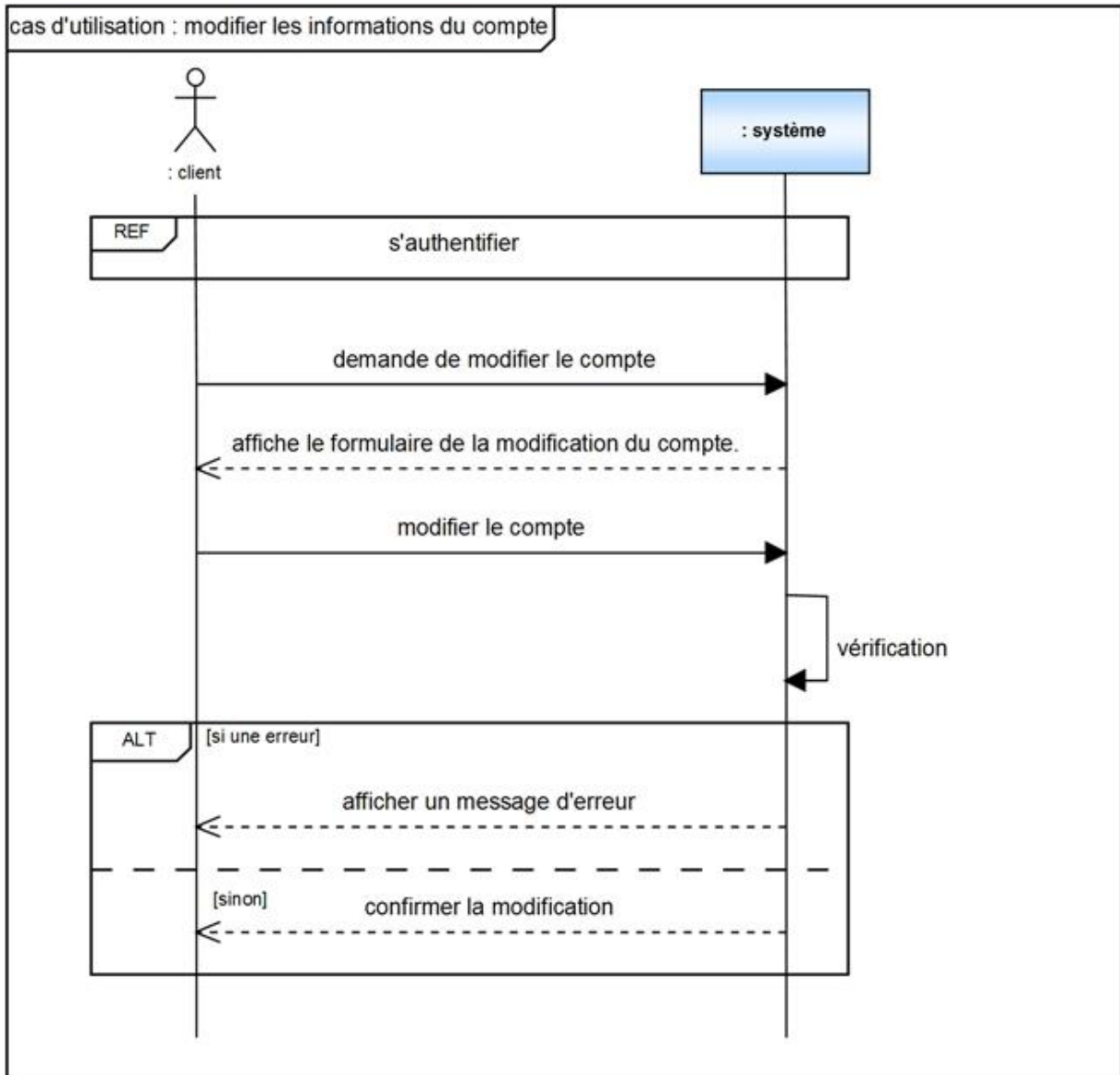


Figure 20 : Diagramme de séquence du cas d'utilisation « Modifier les informations du compte »

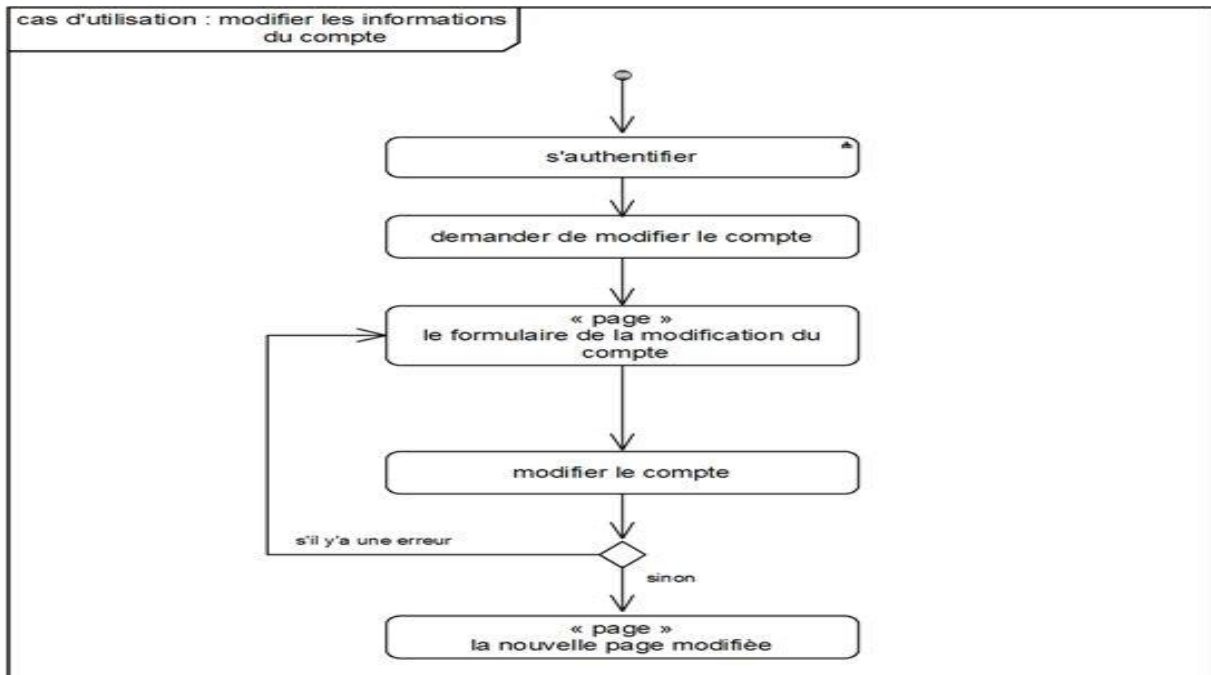


Figure 21 : Diagramme d’activité du cas d’utilisation « Modifier les informations du compte »

### 3.10. Gérer article

➤ Description textuelle du cas d’utilisation

Cas d’utilisation	Gérer article
objectif	Permet de gérer les articles d’une Catégorie.
acteur	Vendeur
Pré- condition	Le vendeur s’authentifie
Post condition	Un nouvel article est ajouté, modifié ou supprimé
Scénario nominal	<ol style="list-style-type: none"> <li>1. Le vendeur demande la mise à jour des articles.</li> <li>2. Le système affiche la page de mise à jour des articles.</li> <li>3. Le vendeur met à jour les informations (ajouter, modifier ou supprimer article).</li> <li>4. Le vendeur confirme la mise à jour.</li> <li>5. le système confirmer la mise à jour.</li> </ol>
Scénario alternatif	<p><b>5.1</b> S’il y a des erreurs</p> <p>Un message d’erreur est affiché sur l’écran que les informations saisies sont non valides.</p>

Tableau 11 : Description textuelle du cas d’utilisation « Gérer article ».

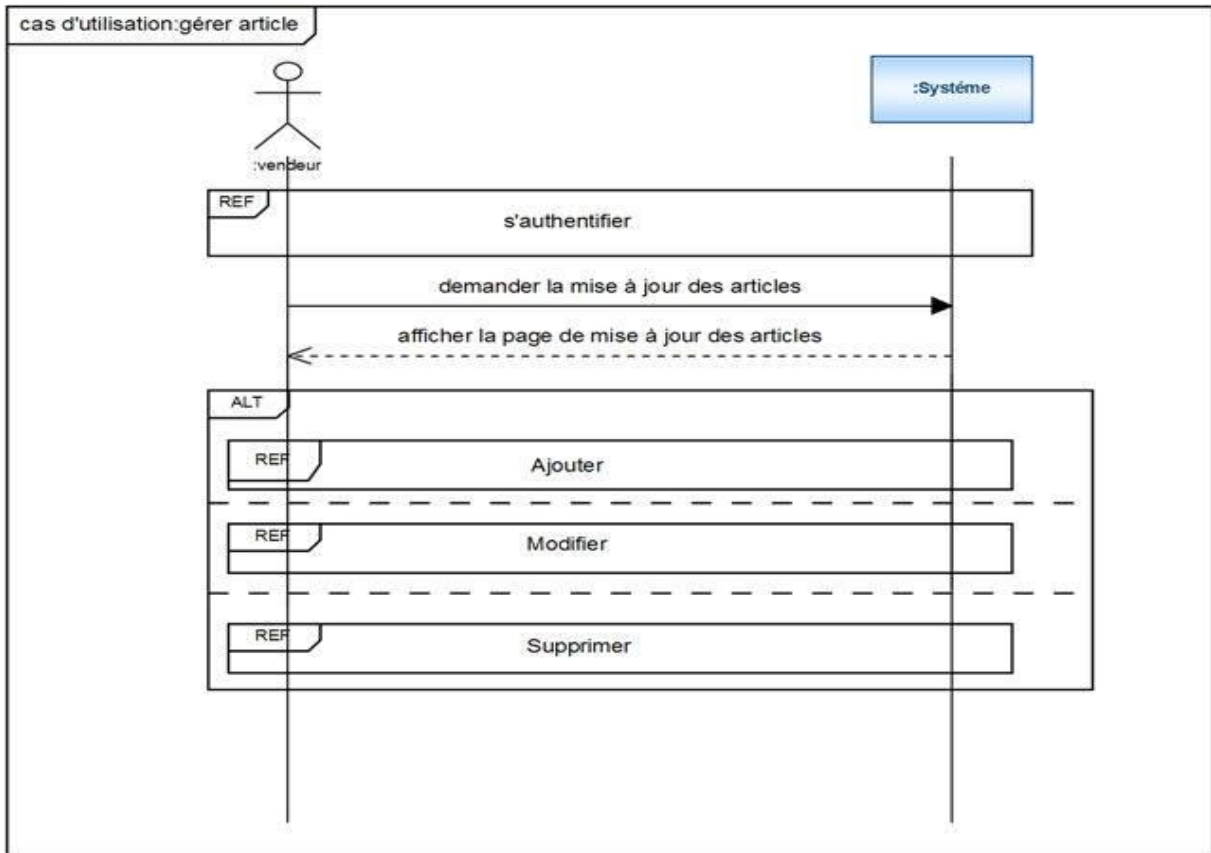


Figure 22 : Diagramme de séquence du cas d'utilisation « Gérer article »

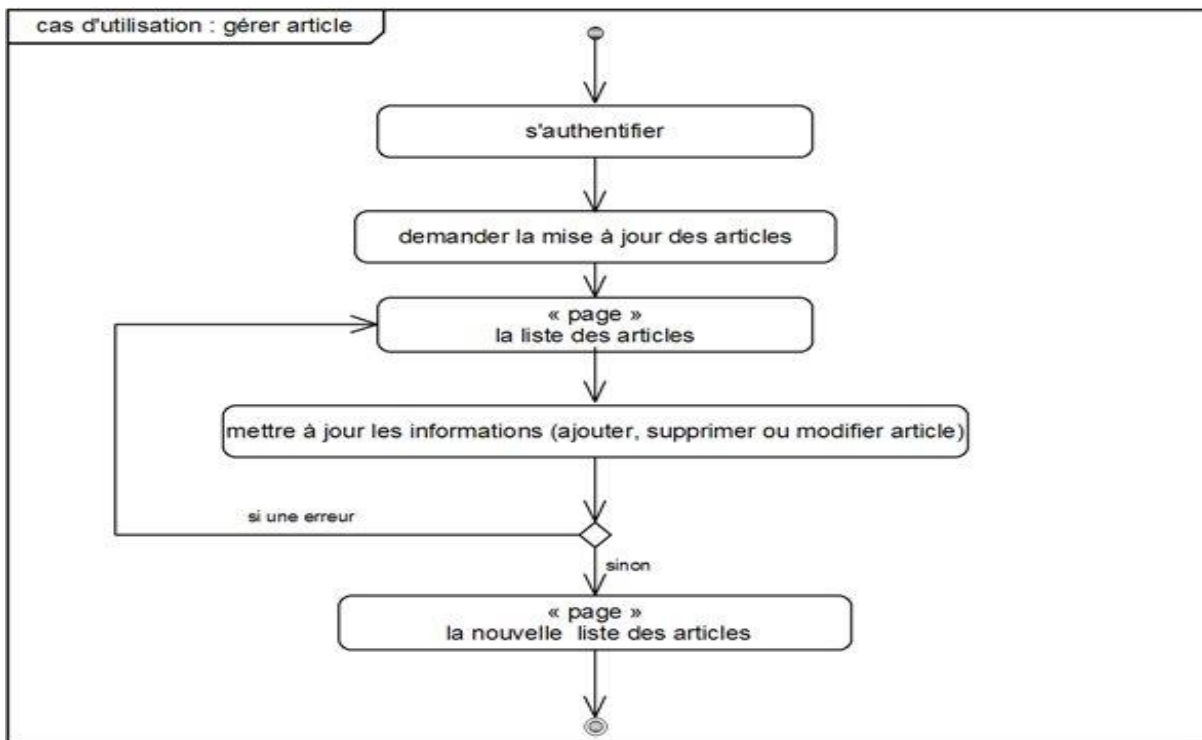


Figure 23 : Diagramme d'activité du cas d'utilisation « Gérer article »

### 3.11. Ajouter article

- Description textuelle du cas d'utilisation

Cas d'utilisation	Ajouter article
objectif	Ajouter des articles à la base de données du site.
acteur	Vendeur
Pré- condition	Le vendeur s'authentifie
Post condition	Un nouvel article est ajouté
Scénario nominal	<ol style="list-style-type: none"> <li>1. Le vendeur demande d'ajouter un article.</li> <li>2. Le système lui affiche la page d'ajout d'un article.</li> <li>3. Le vendeur saisit les informations de ce nouvel article.</li> <li>4. Le vendeur valide les informations saisies.</li> <li>5. Le système vérifie la validité des informations, et affiche un message de confirmation</li> </ol>
Scénario alternatif	<p><b>5.1</b> S'il y a des erreurs</p> <p>Un message d'erreur est affiché sur l'écran que les informations saisies non valides, article existe déjà.</p>

**Tableau 12 : Description textuelle du cas d'utilisation « Ajouter article ».**





## 3.12. Modifier article

- Description textuelle du cas d'utilisation

Cas d'utilisation	Modifier article.
Objectif	Modifier les informations sur les articles (la quantité, le prix, etc.).
Acteur	Vendeur.
Précondition	Vendeur s'authentifie. Article existe déjà.
Post condition	L'article est modifié (prix, quantité, stock, etc.).
Scénario nominal	<ol style="list-style-type: none"> <li>1. Le vendeur demande de modifier un article.</li> <li>2. Le système affiche la liste des articles</li> <li>3. Le vendeur choisit et sélectionne un article pour le modifier.</li> <li>4. Le système lui affiche la page demandée.</li> <li>5. Le vendeur modifie les informations souhaitées.</li> <li>6. le vendeur valide et confirme les modifications saisies.</li> <li>7. Le vendeur demande de valider sa confirmation.</li> <li>8. Le système vérifie la validité des informations, et affiche message de confirmation.</li> </ol>
Scénario alternatif	<p><b>8.1</b> S'il y a des erreurs :</p> <p>Un message d'erreur est affiché pour le vendeur.</p>

Tableau 13 : Description textuelle du cas d'utilisation « Modifier article ».

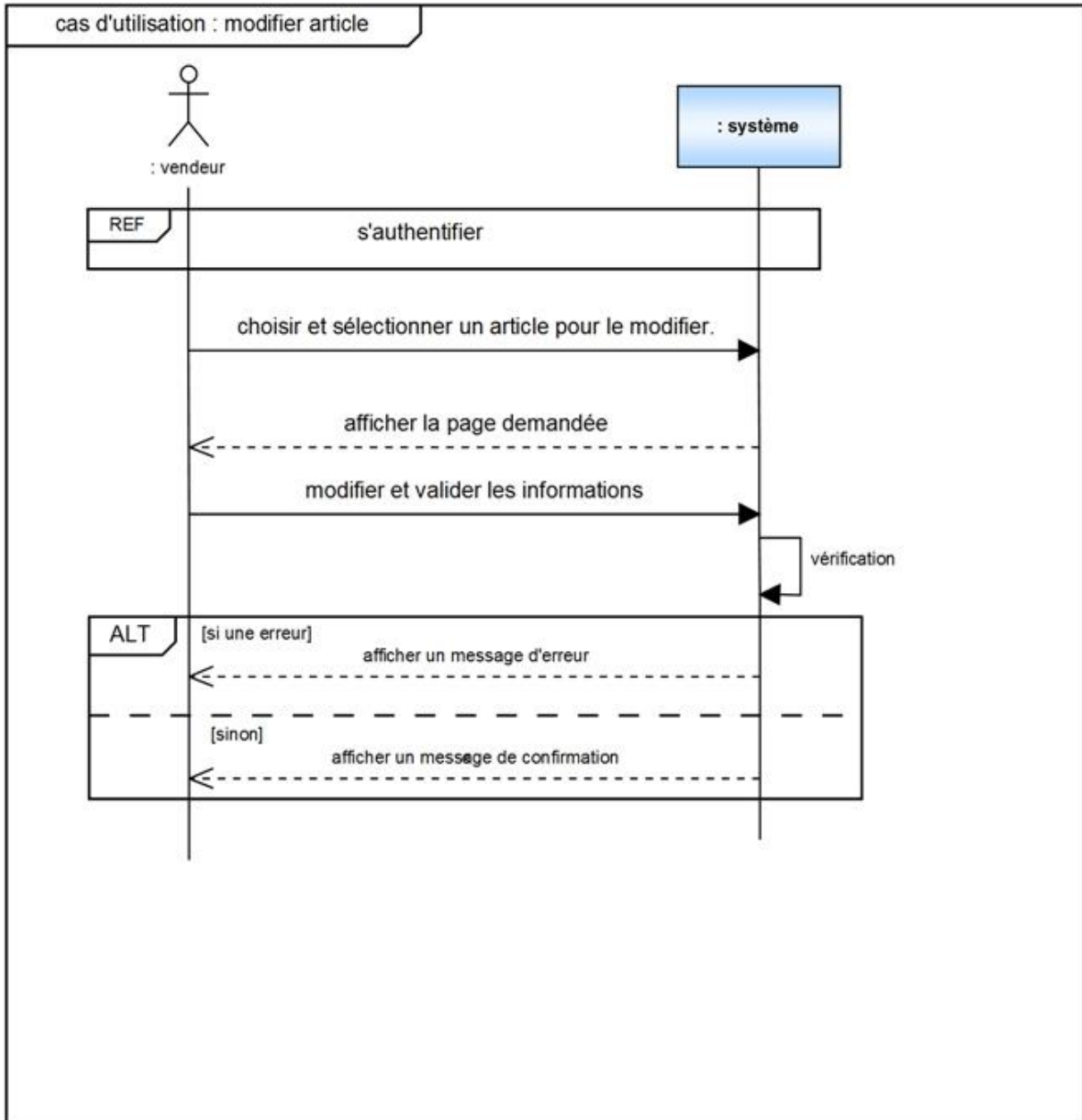


Figure 26 : Diagramme de séquence du cas d'utilisation « Modifier article »

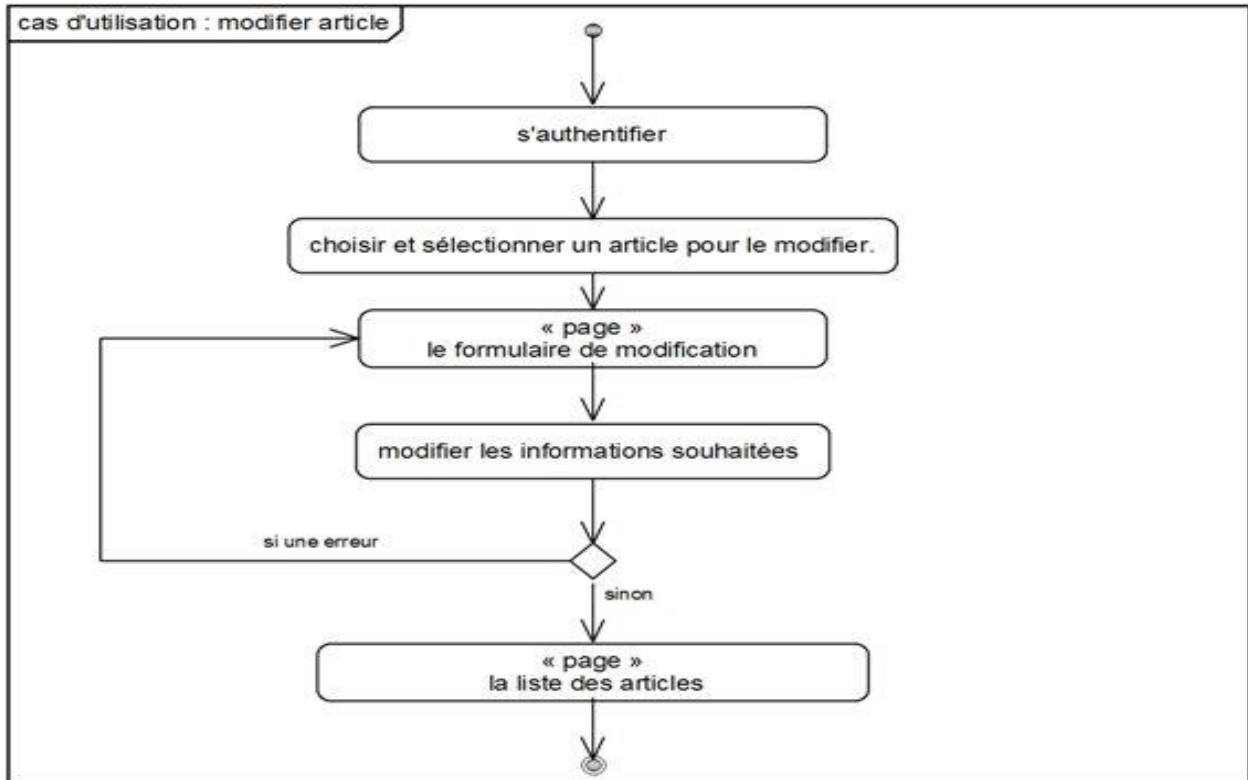


Figure 27 : Diagramme d’activité du cas d’utilisation « Modifier article »

### 3.13. Supprimer article

➤ Description textuelle du cas d’utilisation

Cas d’utilisation	Supprimer articles
objectif	Supprimer un article de la base de donné du site.
acteur	Vendeur
Pré- condition	Le vendeur s’authentifie
Post condition	Un article a été supprimé.
Scénario nominal	<ol style="list-style-type: none"> <li>1. Le vendeur demande de supprimer un article.</li> <li>2. Le système affiche la liste des articles.</li> <li>3. Le vendeur choisit un article pour le supprimer.</li> <li>4. Le système affiche un message pour confirmer la suppression.</li> <li>5. Le vendeur valide la suppression de l'article.</li> <li>6. Le system affiche un message de confirmation.</li> </ol>
Scénario alternatif	<ol style="list-style-type: none"> <li>4.1. Si le vendeur annule la suppression. le système affiche un message d’annulation</li> </ol>

Tableau 14 : Description textuelle du cas d’utilisation « Supprimer article ».

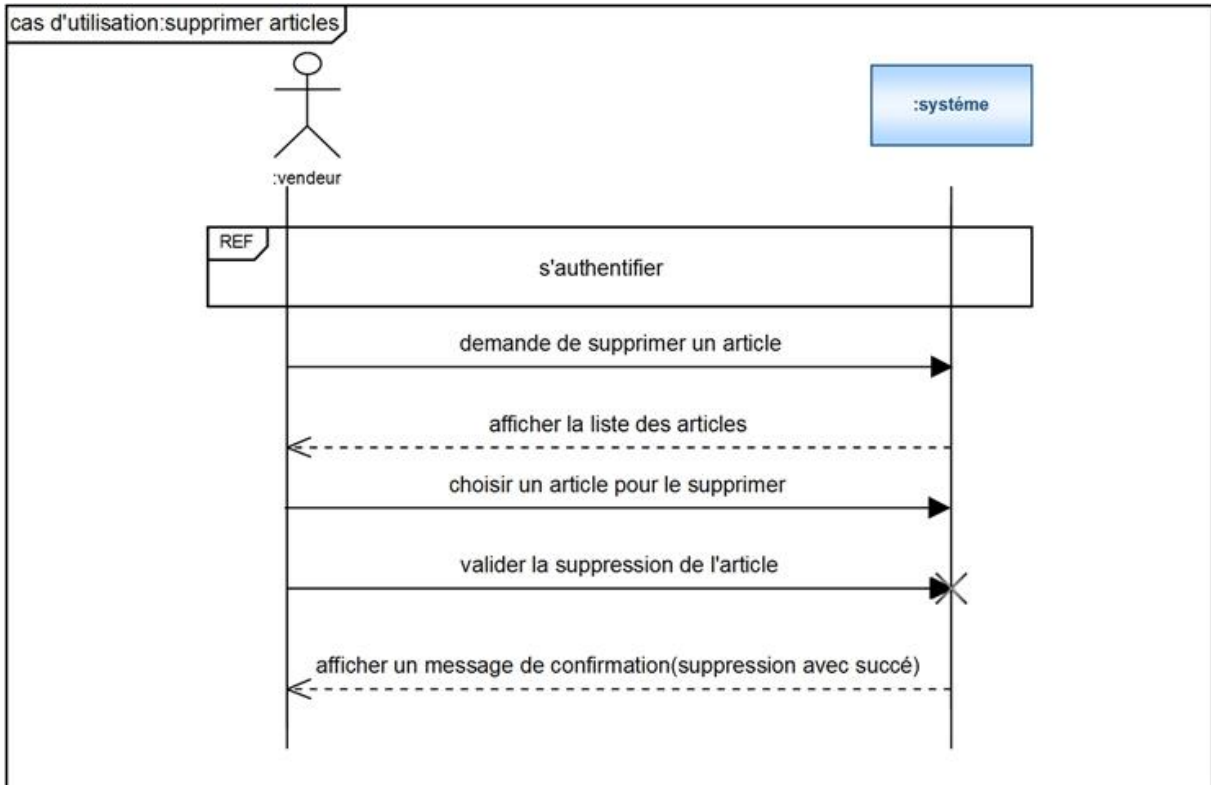


Figure 28 : Diagramme de séquence du cas d'utilisation « Supprimer article »

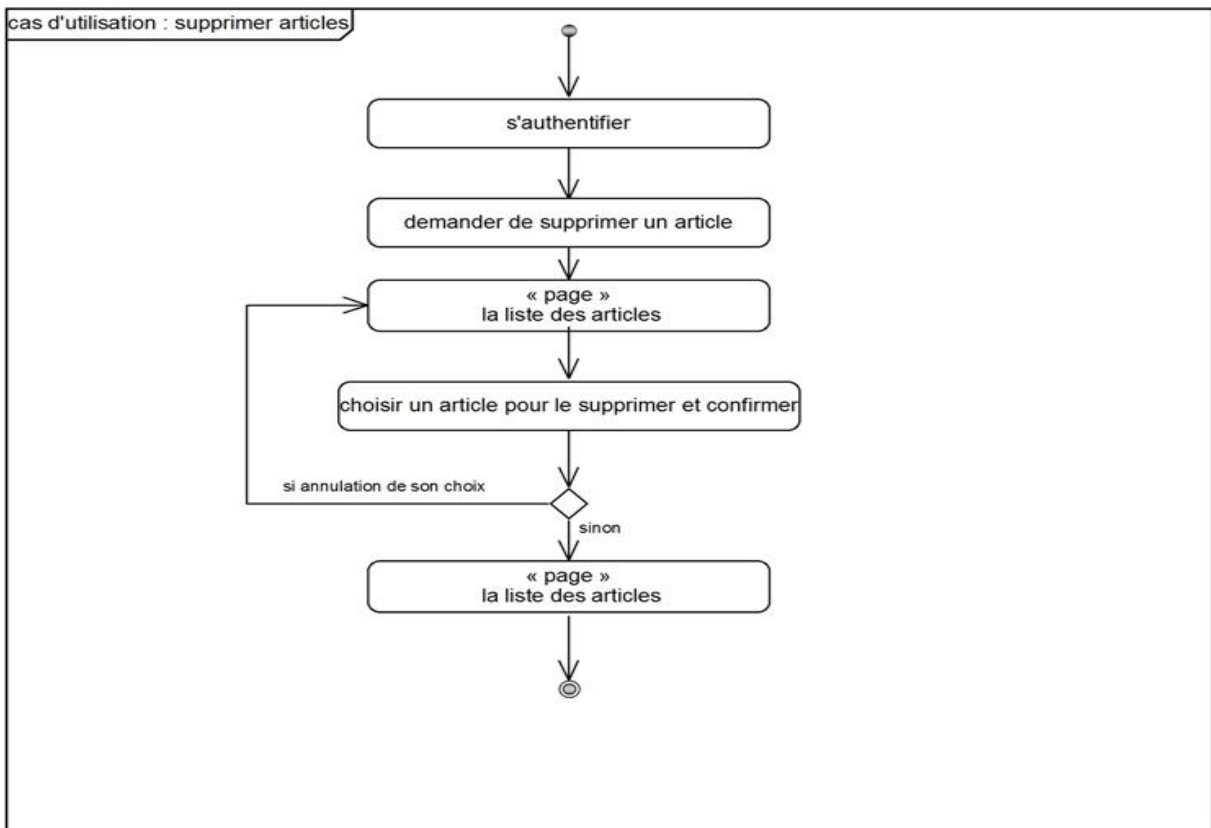


Figure 29 : Diagramme d'activité du cas d'utilisation « supprimer article »

3.14. Traiter commande

➤ Description textuelle du cas d'utilisation

Cas d'utilisation	Traiter commande
Objectif	Recevoir les commandes des clients et les traiter.
Acteur	Vendeur.
Précondition	Vendeur s'authentifie. La commande est déjà payée par le client.
Post condition	Les commandes sont traitées.
Scénario nominal	<ol style="list-style-type: none"> <li>1. Le vendeur demande les commandes existées.</li> <li>2. Le système affiche les commandes.</li> <li>3. Le vendeur vérifie les commandes.</li> <li>4. Le vendeur s'occupe de la livraison.</li> </ol>
Scénario alternatif	Néant

Tableau 15 : Description textuelle du cas d'utilisation « Traiter commande».

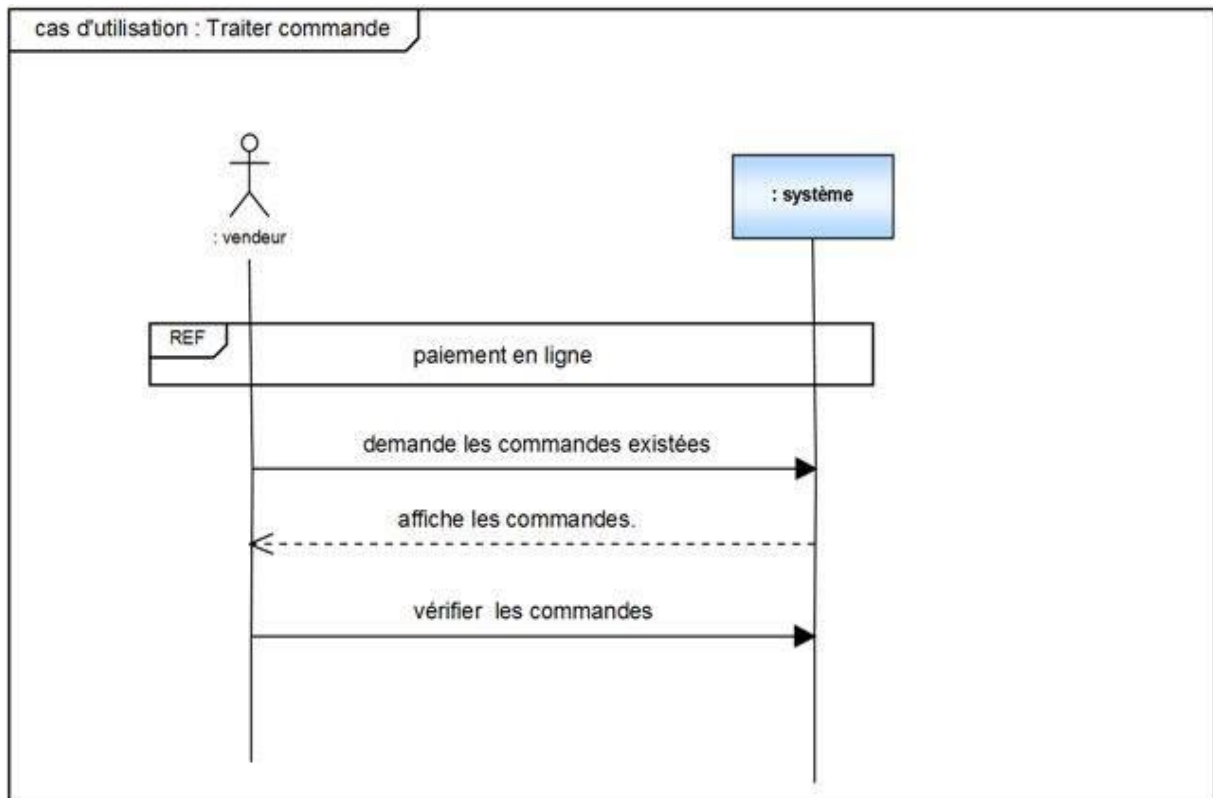


Figure 30 : Diagramme de séquence du cas d'utilisation «Traiter commande»

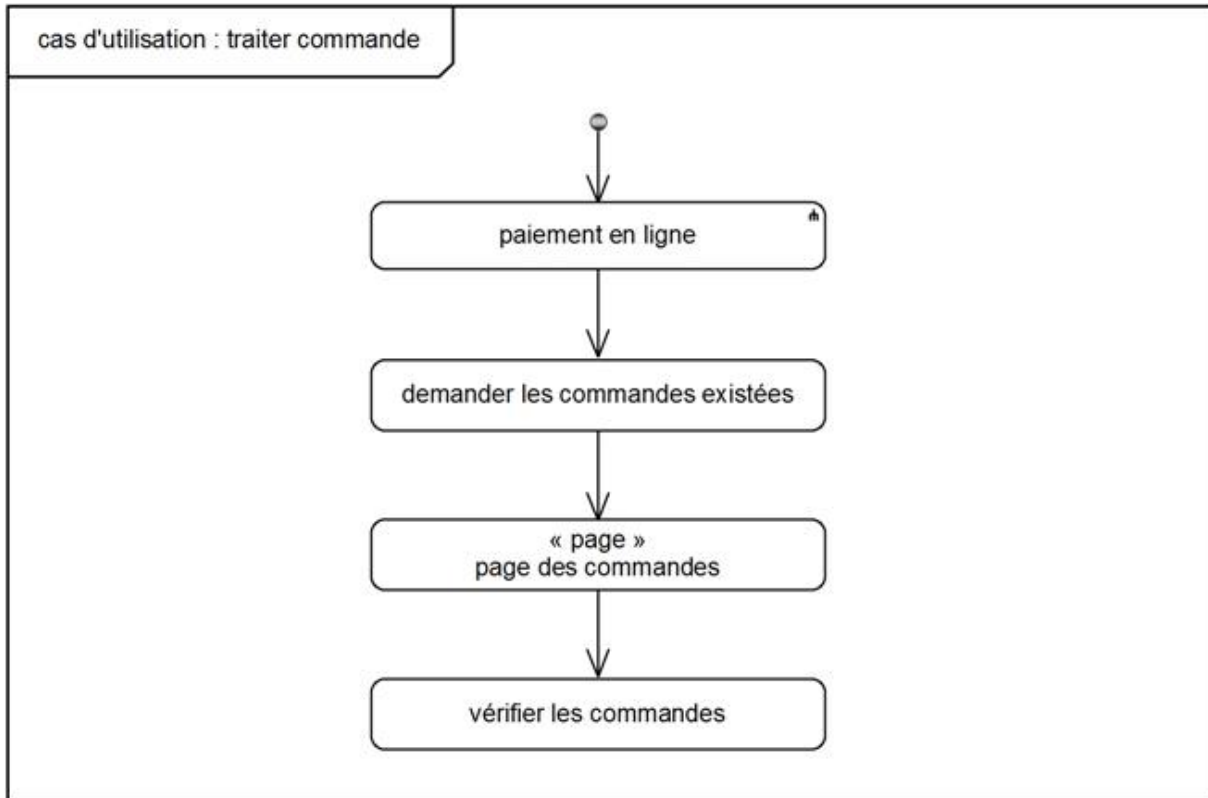


Figure 31 : Diagramme d'activité du cas d'utilisation « Traiter commande »

### 3.15. Gérer compte

➤ Description textuelle du cas d'utilisation

Cas d'utilisation	Gérer compte
Objectif	Permet à l'administrateur d'ajouter ou bloquer un compte.
Acteur	Administrateur.
Précondition	Administrateur s'authentifie.
Post condition	Un compte est ajouté / bloqué
Scénario nominal	<ol style="list-style-type: none"> <li>1. L'administrateur demande au système de gérer les comptes.</li> <li>2. Le système affiche la liste des comptes.</li> <li>3. L'administrateur choisit l'opération à sélectionner (ajout, blocage).</li> <li>4. l'administrateur demande de valider les modifications.</li> <li>5. Le système confirme l'opération.</li> </ol>
Scénario alternatif	Néant

Tableau 16 : Description textuelle du cas d'utilisation « Gérer compte ».

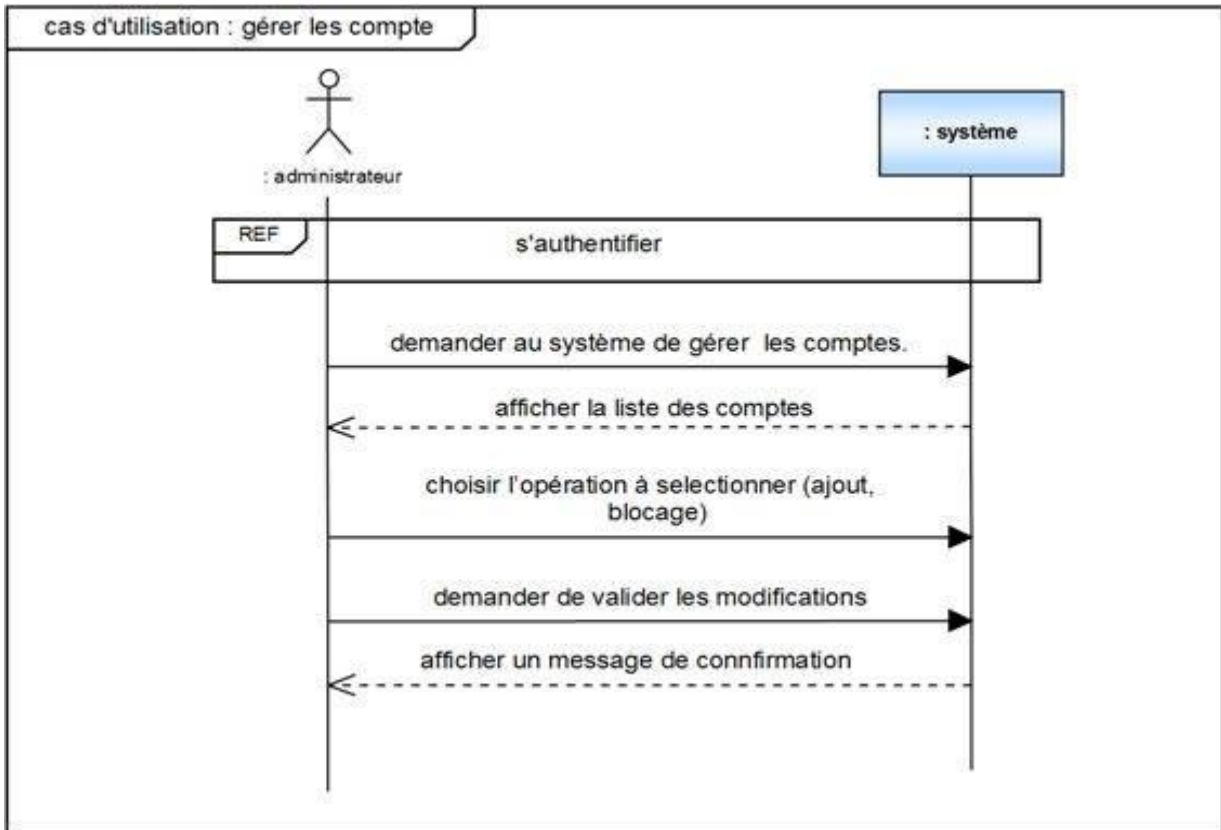


Figure 32 : Diagramme de séquence du cas d'utilisation « Gérer compte »

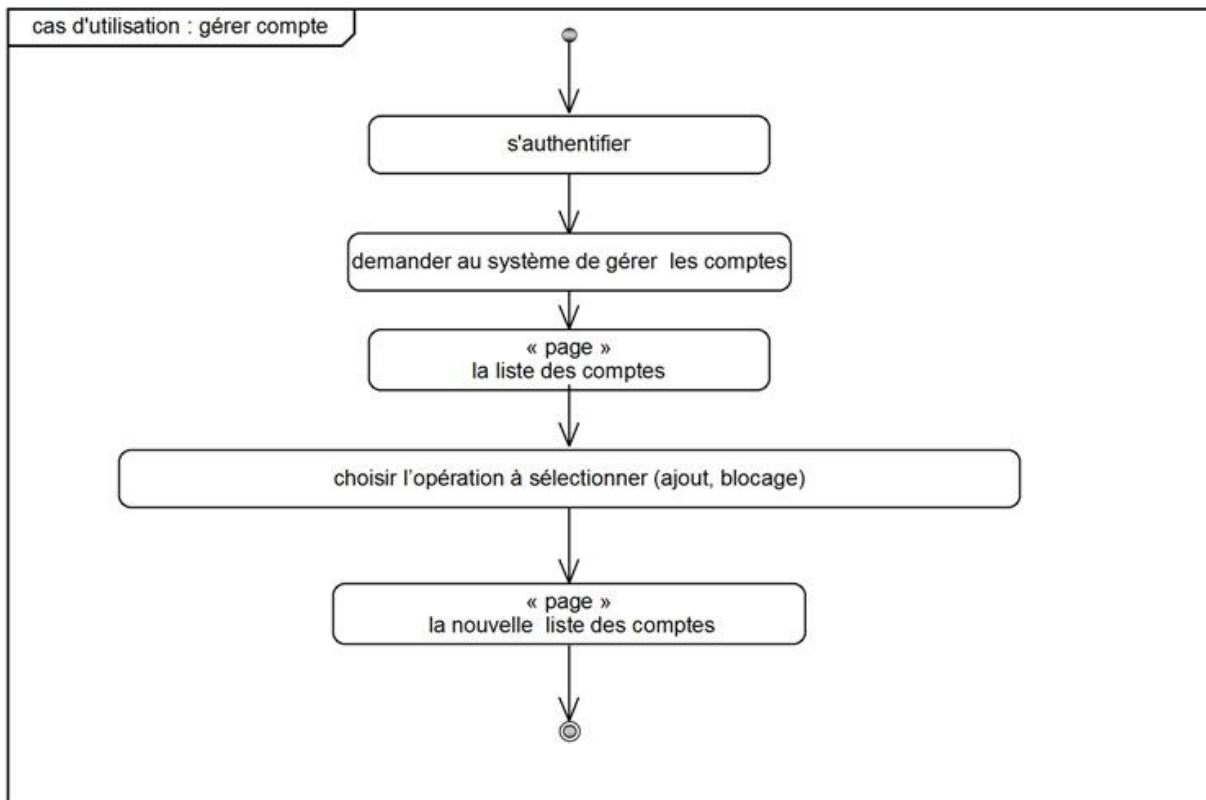


Figure 33 : Diagramme d'activité du cas d'utilisation « Gérer compte »



### 3.16. Ajouter compte

- Description textuelle du cas d'utilisation

Cas d'utilisation	Ajouter compte
Objectif	Ajouter des comptes vendeur a la base de données du site.
Acteur	Administrateur
Pré- condition	L'administrateur s'authentifier
Post condition	Un nouveau compte est ajouté
Scénario nominal	<ol style="list-style-type: none"> <li>1. L'administrateur demande d'ajouter un compte.</li> <li>2. Le système lui affiche la page d'ajout un compte.</li> <li>3. L'administrateur saisit les informations de ce nouvel compte.</li> <li>4. L'administrateur valide les informations saisies.</li> <li>5. Le système vérifie la validité des informations, et affiche un message de confirmation.</li> </ol>
Scénario alternatif	<p><b>5.1</b> S'il y a des erreurs</p> <p>Un message d'erreur est affiché sur l'écran que les informations saisies sont non valides.</p>

**Tableau 17 : Description textuelle du cas d'utilisation « Ajouter compte ».**

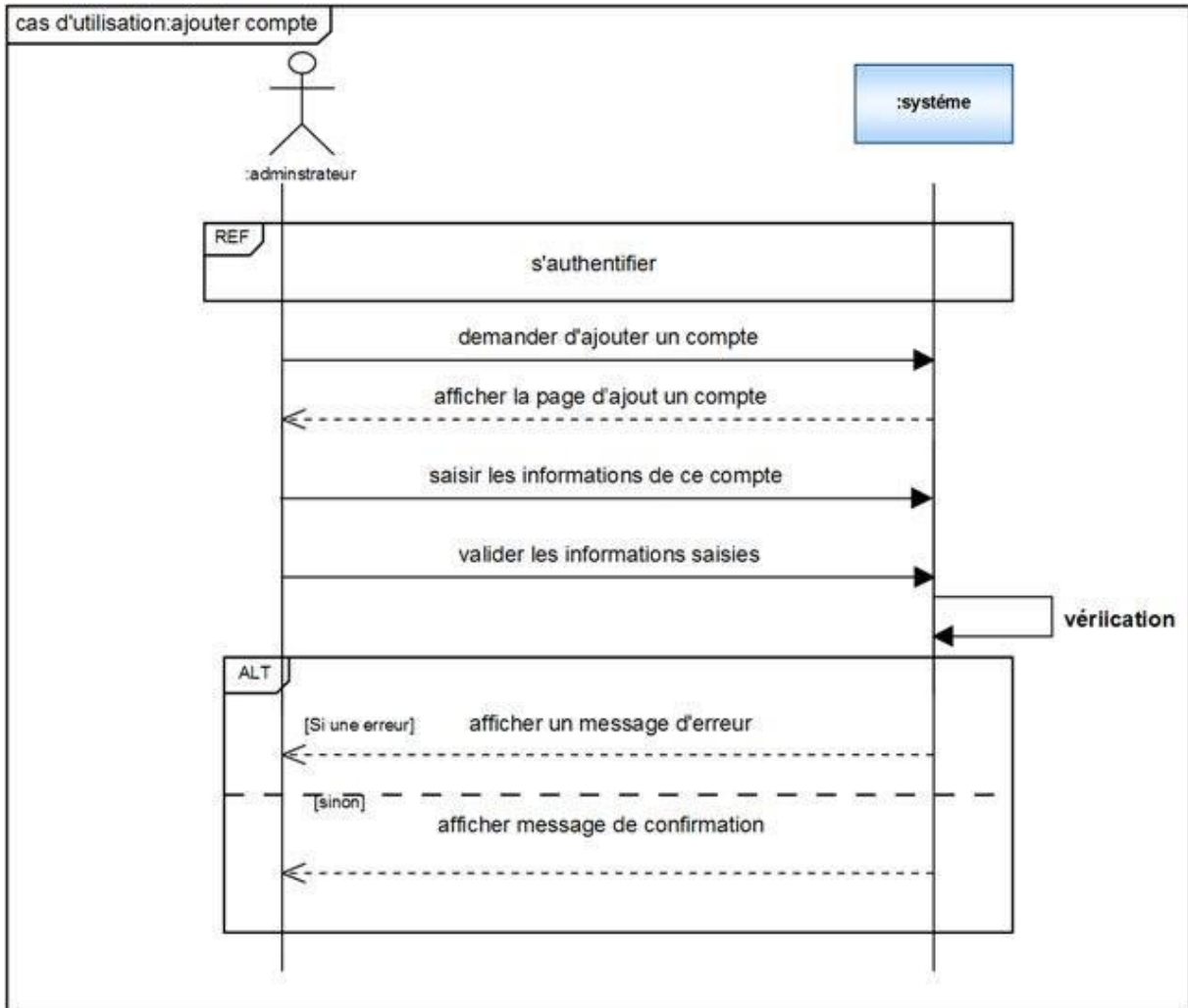


Figure 34 : Diagramme de séquence du cas d'utilisation « Ajouter compte »

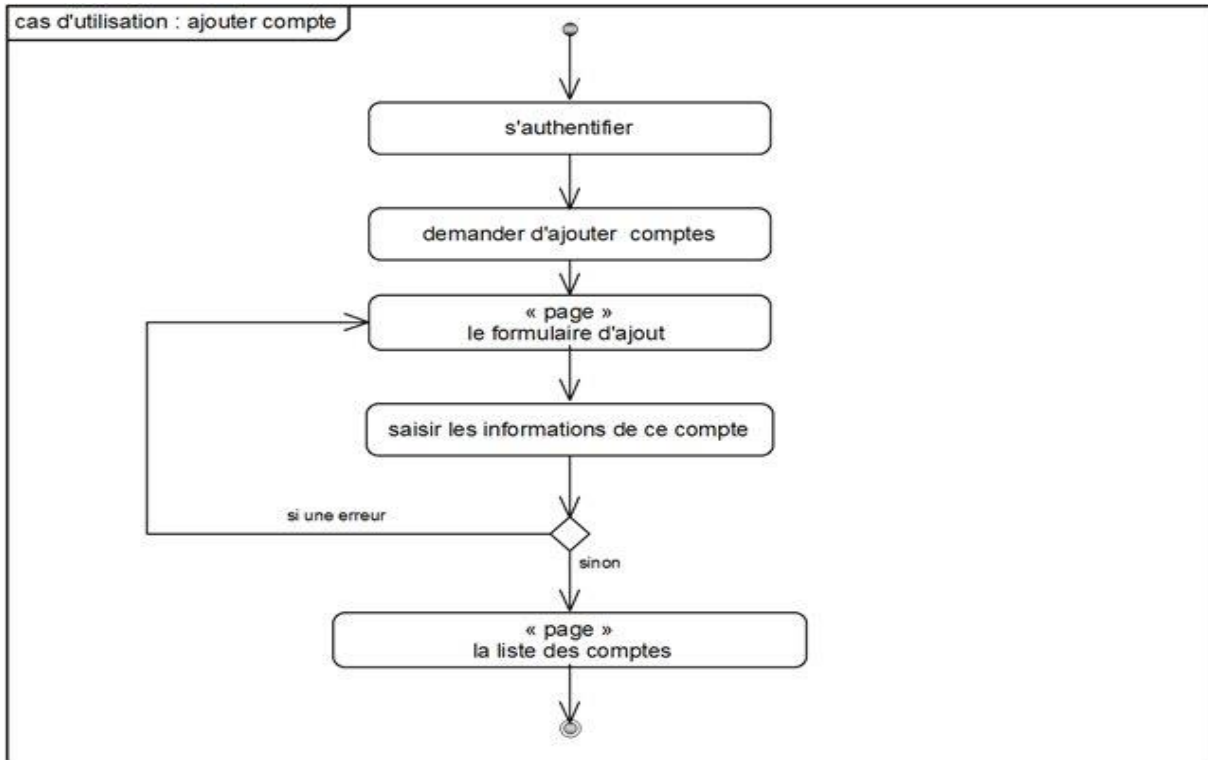


Figure 35 : Diagramme d'activité du cas d'utilisation « Ajouter compte »

### 3.17. Bloquer comptes

- Description textuelle du cas d'utilisation bloquer compte (le blocage des comptes vendeurs).

Cas d'utilisation	Bloquer compte
Objectif	Bloquer des comptes vendeur ou client.
Acteur	Administrateur
Pré- condition	L'administrateur s'authentifie
Post condition	Un compte a été bloqué.
Scénario nominal	<ol style="list-style-type: none"> <li>1. L'administrateur demande de bloquer un compte.</li> <li>2. Le système affiche la liste des comptes.</li> <li>3. L'administrateur choisit un compte pour le bloquer.</li> <li>4. système affiche un message pour confirmer le blocage.</li> <li>5. L'administrateur valide le blocage du compte.</li> <li>6. Le système affiche un message de confirmation.</li> </ol>
Scénario alternatif	<ol style="list-style-type: none"> <li>5.1 Si l'administrateur annule son choix le système affiche un message d'annulation</li> </ol>

Tableau 18 : Description textuelle du cas d'utilisation « Bloquer compte ».

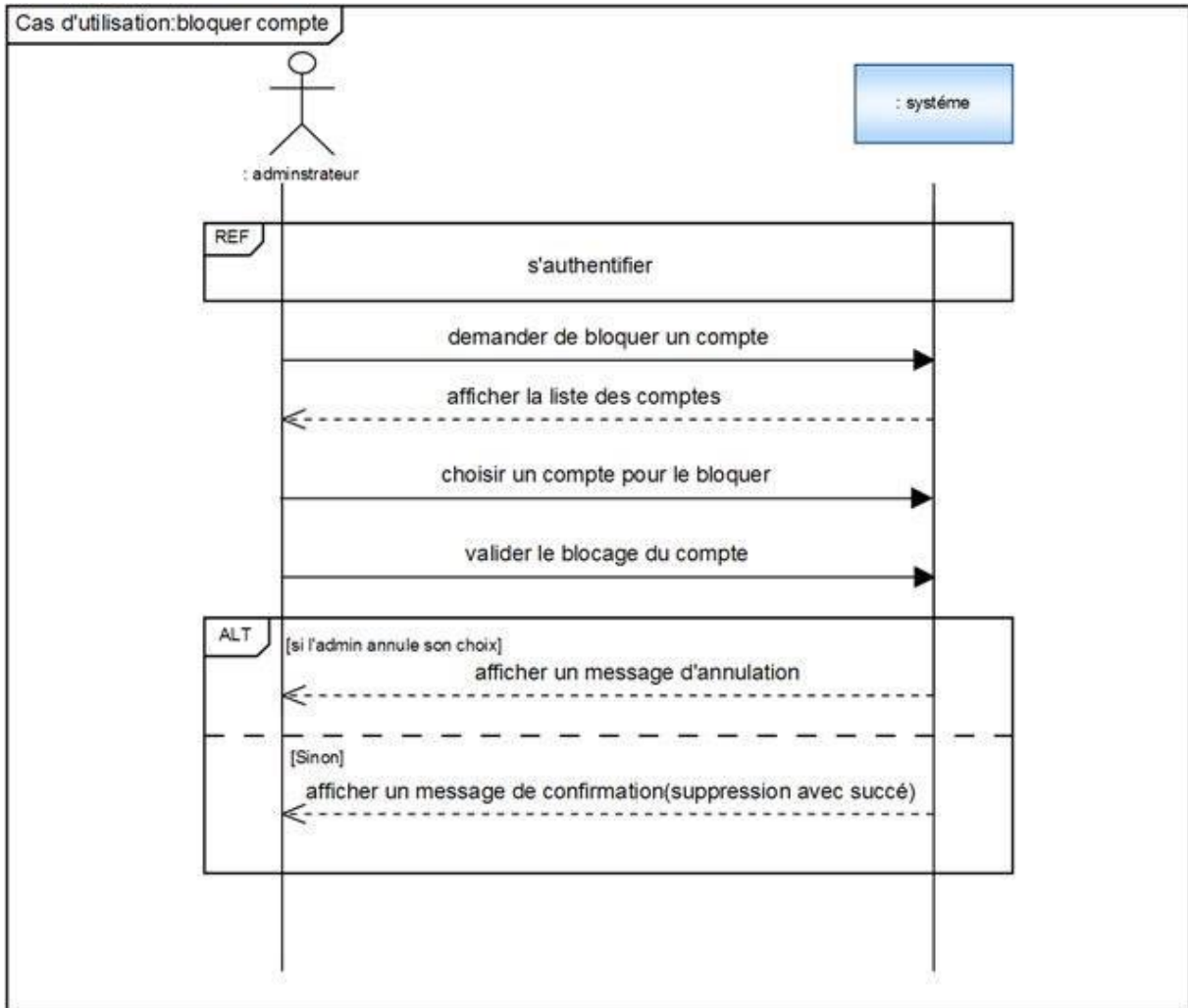


Figure 36 : Diagramme de séquence du cas d'utilisation « Bloquer compte »

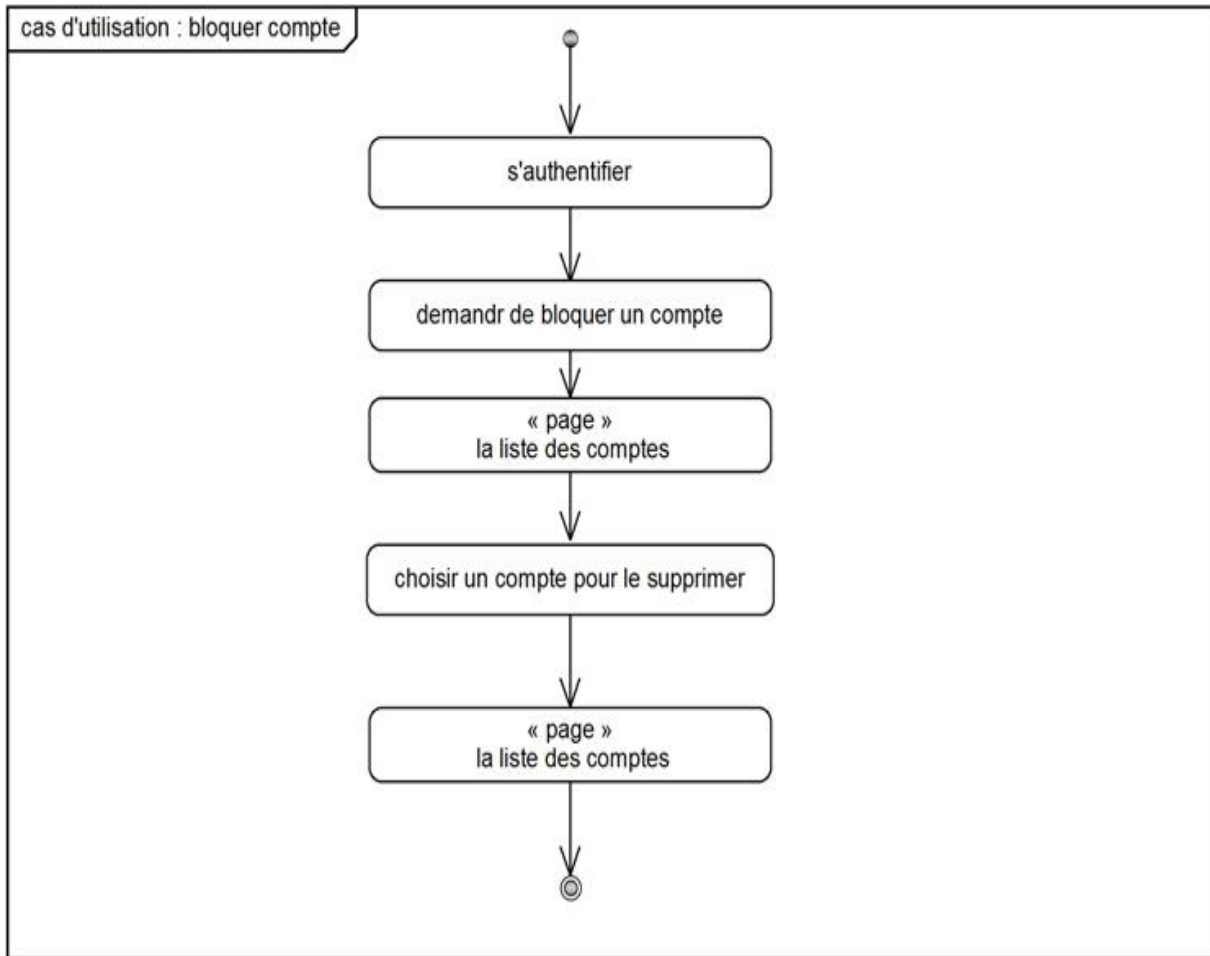


Figure 37 : Diagramme d’activité du cas d’utilisation « Bloquer compte »

### 3.18. Statistiques

- Description textuelle du cas d’utilisation

Cas d’utilisation	Statistique
Objectif	Voir les différentes statistiques concernant les ventes et achats.
Acteur	Administrateur.
Précondition	administrateur s’authentifie.
Post condition	Néant
Scénario nominal	<ol style="list-style-type: none"> <li>1. L’administrateur demande de voir les statistiques.</li> <li>2. Le système affiche les statistiques.</li> </ol>
Scénario alternatif	néant

Tableau 19 : Description textuelle du cas d’utilisation « Statistiques ».

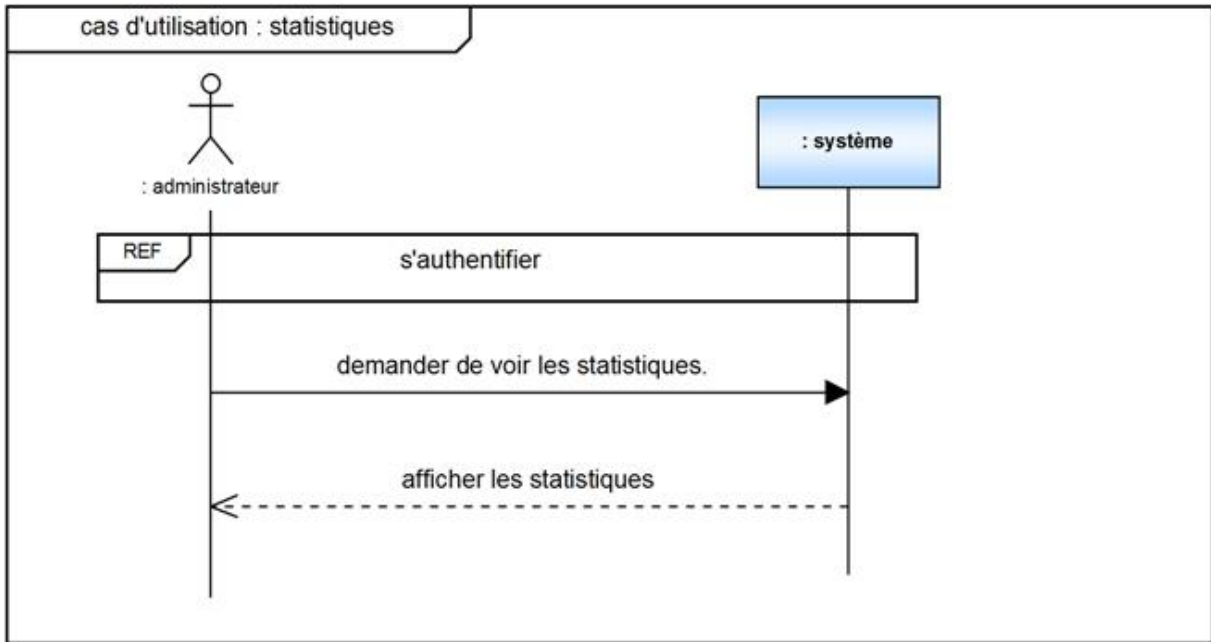


Figure 38 : Diagramme de séquence du cas d'utilisation « Statistiques »

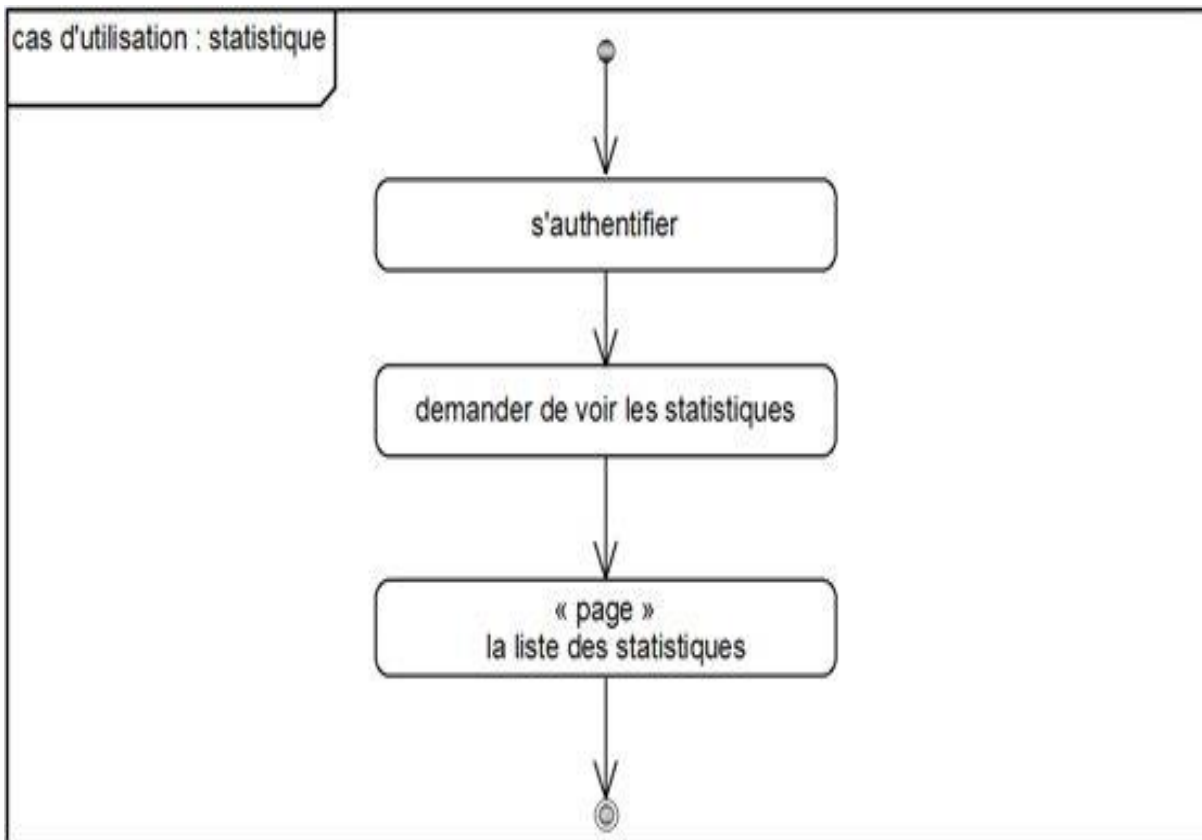


Figure 39 : Diagramme d'activité du cas d'utilisation « Statistique »

## 4. Identification des classes candidats :

Cette phase nous aider à trouver les classes principales du futur modèle statique d'analyse, la technique utilisée pour identifier les classes candidates est :

- Cherchez les noms communs importants dans les descriptions textuelles des cas d'utilisation.
- Vérifiez les propriétés «objet» de chaque concept (identité, propriétés, Comportement), puis définissez ses responsabilités. [1]

### ➤ Responsabilité :

Une responsabilité est une sorte de contrat, ou d'obligation, pour une classe. Elle se place à un niveau d'abstraction plus élevé que les attributs ou les opérations. En fait, on peut dire que les attributs, les opérations, et les associations représentent les propriétés élémentaires qui contribueront à remplir les responsabilités de la classe. [1]

Ces diagrammes préliminaires, que nous appelons « diagramme de classes participantes », n'ont pas d'objectif d'exhaustivité. Ils servent uniquement à démarrer la découverte des classes du modèle d'analyse pour la partie de l'application délimitée par un cas d'utilisation. [1]

### 4.1 La liste des classes candidates :

<b>La liste des classes candidates</b>	<ul style="list-style-type: none"><li>• Compte</li><li>• Administrateur</li><li>• Client</li><li>• Vendeur</li><li>• Catégorie</li><li>• Article</li><li>• Panier</li><li>• Commande</li><li>• Facture</li></ul>
--	--

4.2 Responsabilités des classes :

- la classe compte

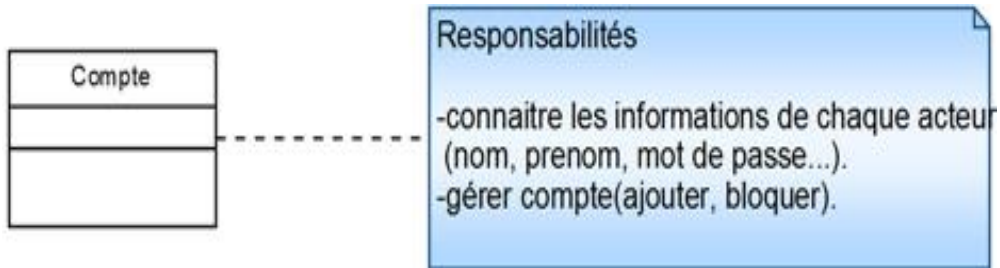


Figure 40 : Les responsabilités de la classe candidat « Compte »

- la classe administrateur

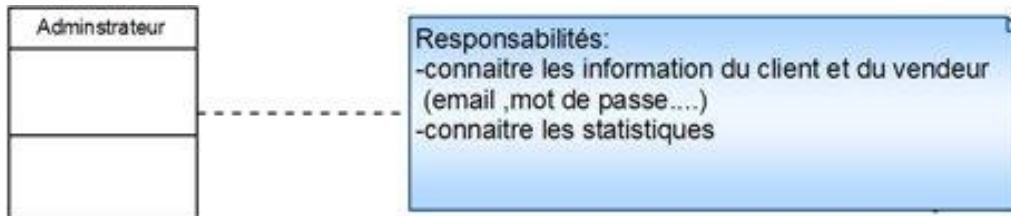


Figure 41 : Les responsabilités de la classe candidat « Administrateur »

- la classe client



Figure 42 : Les responsabilités de la classe candidat « Client »

- la classe vendeur



Figure 43 : Les responsabilités de la classe candidat « Vendeur »



- la classe catégorie



Figure 44 : Les responsabilités de la classe candidat « Catégorie »

- la classe article



Figure 45 : Les responsabilités de la classe candidat « Article »

- la classe panier

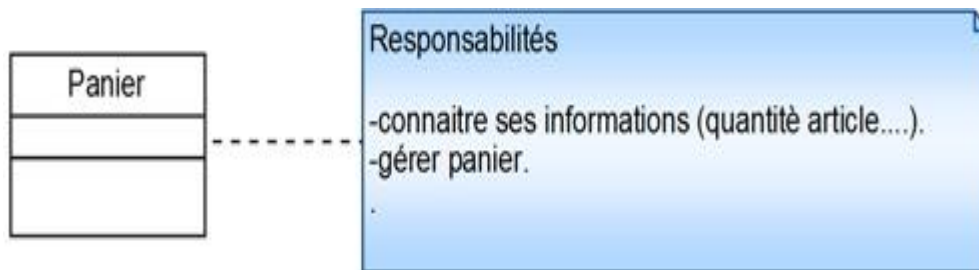


Figure 46 : Les responsabilités de la classe candidat « Panier »

- la classe commande



Figure 47 : Les responsabilités de la classe candidat « Commande »

- la classe facture



Figure 48 : Les responsabilités de la classe candidat « Facture»

### 5 .Diagramme de classes participantes

Les figures qui vont suivre représentent les diagrammes de classes participantes pour chaque cas d'utilisation.

#### 5.1 Cas d'utilisation : Créer compte

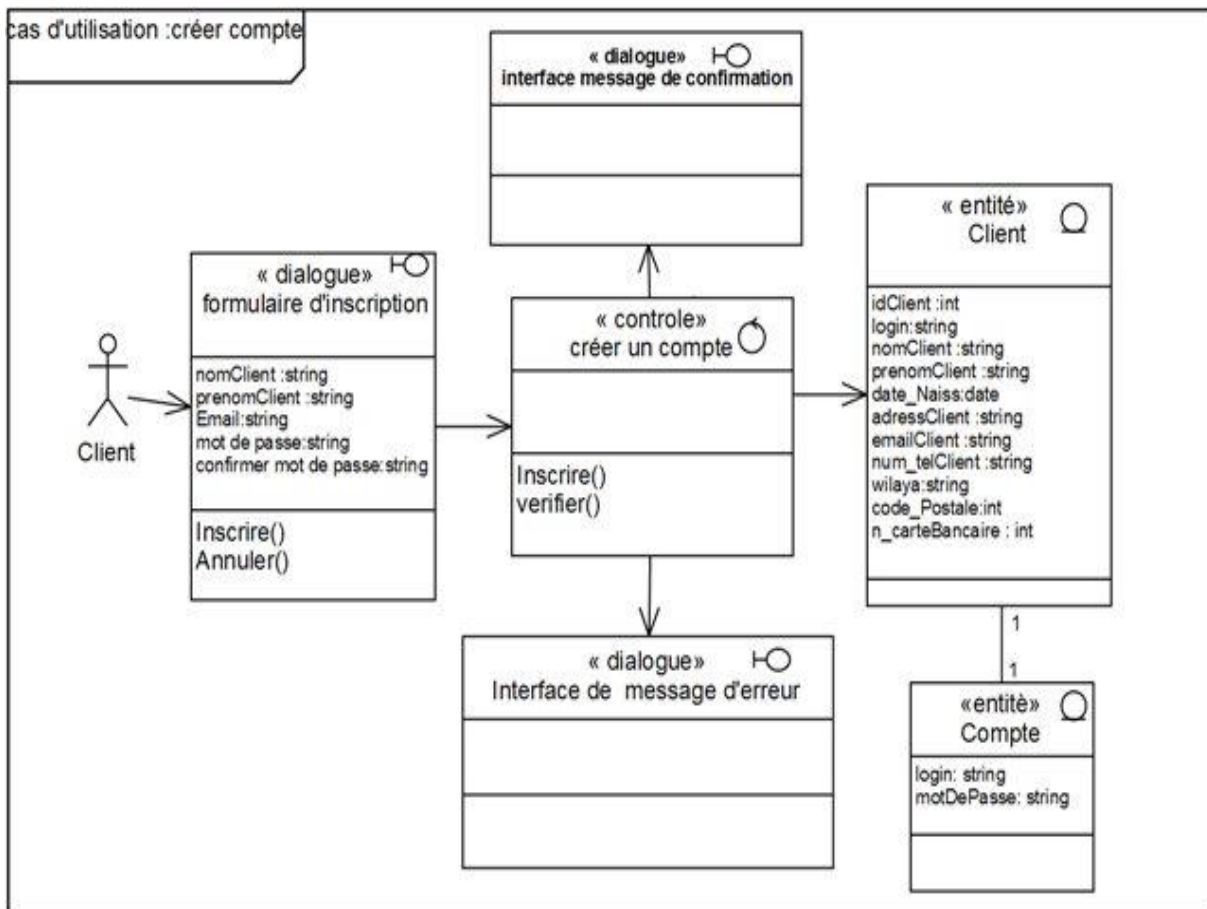


Figure 49 : Diagramme de classe participantes de «créer compte»

5.2. Cas d'utilisation : consulter article

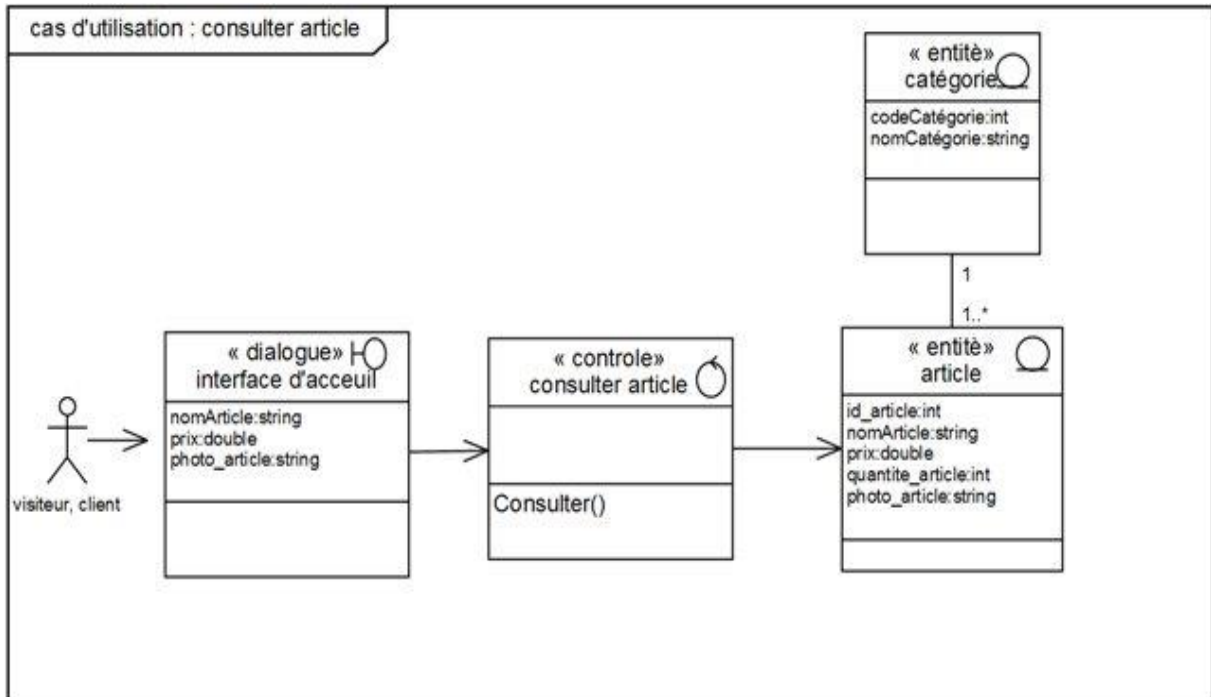


Figure 50 : Diagramme de classe participantes de «consulter article»

5.3. Cas d'utilisation : Rechercher article

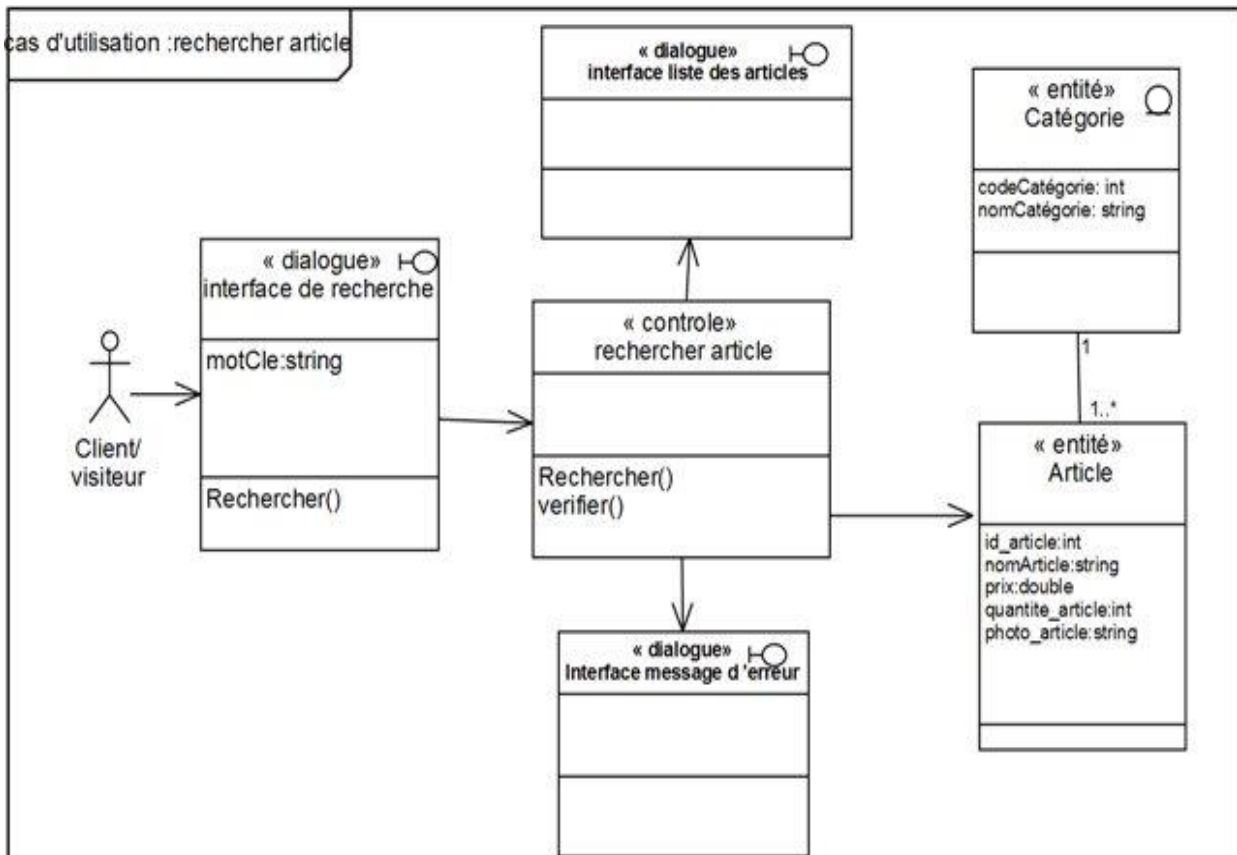


Figure 51 : Diagramme de classe participantes de «rechercher article»

5.4. Cas d'utilisation : Gérer panier

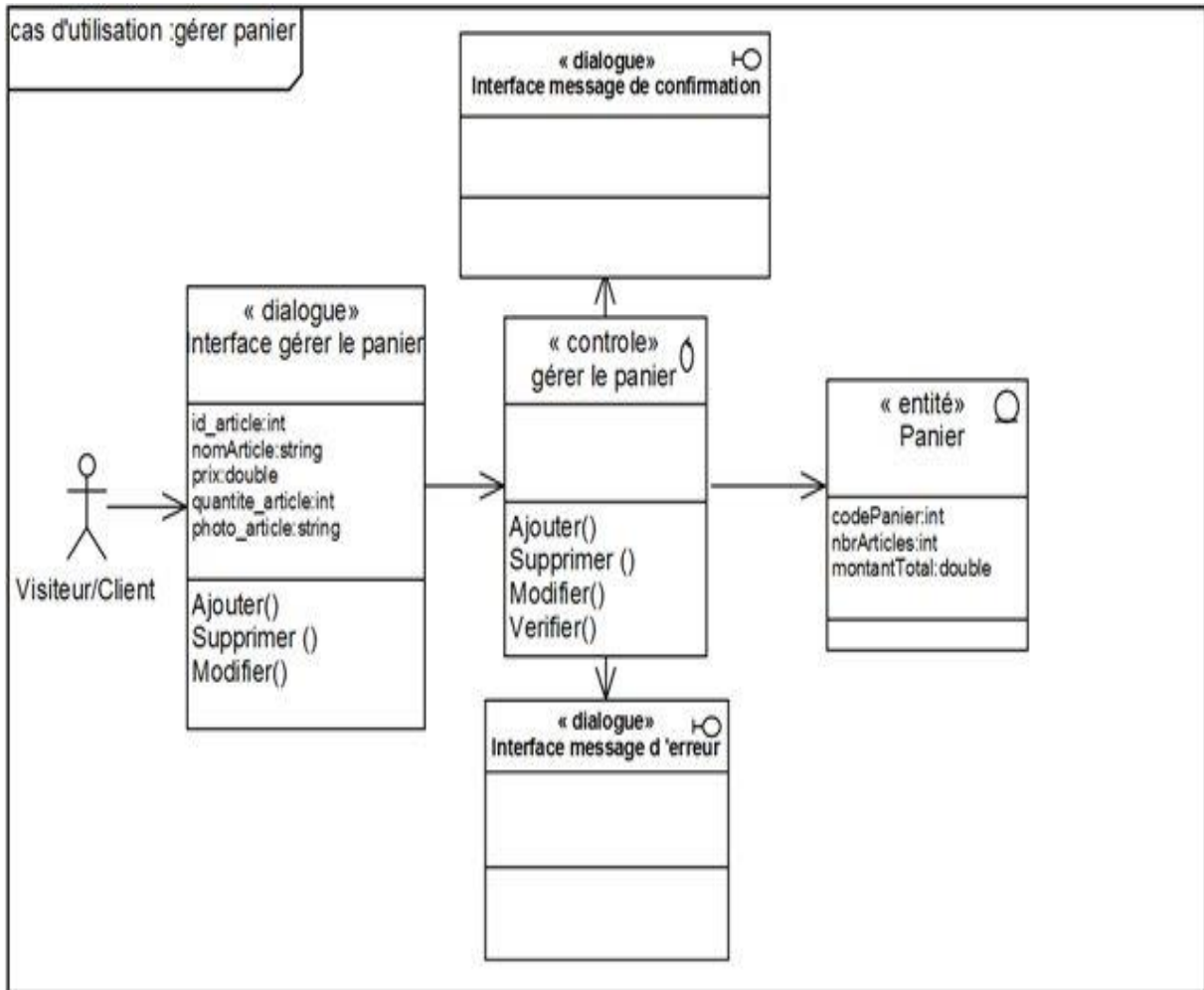


Figure 52 : Diagramme de classe participantes de « gérer panier »

5.5. Cas d'utilisation s'authentifier

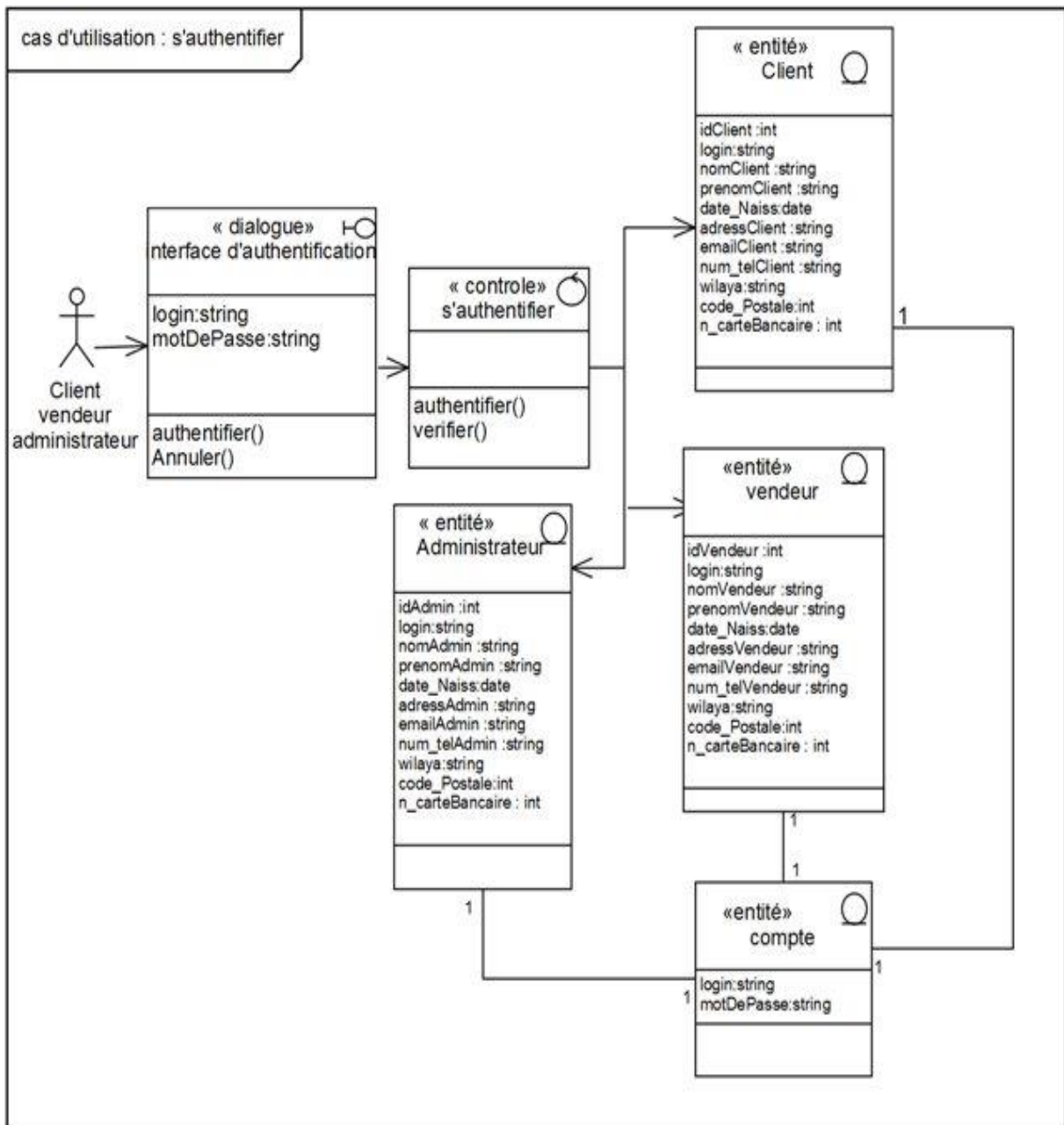


Figure 53 : Diagramme de classe participantes de « s'authentifier »

5.6. Cas d'utilisation : Passer commande

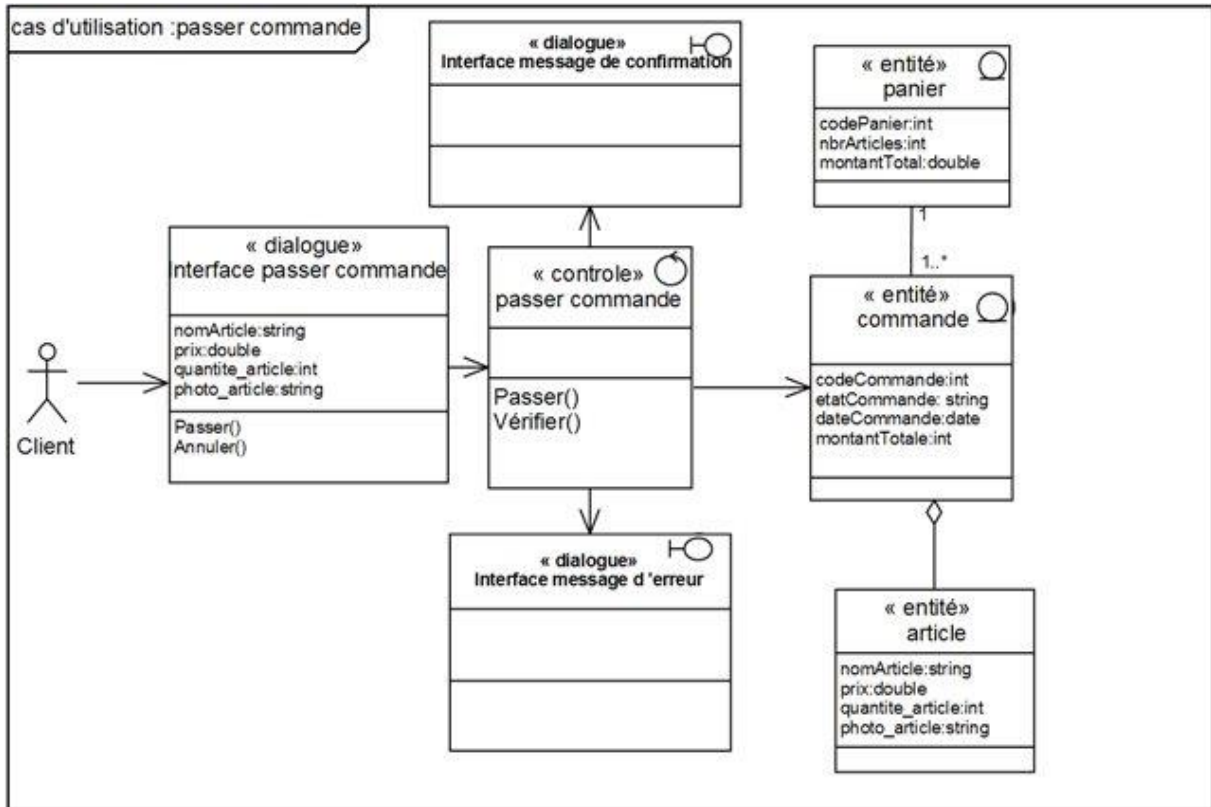


Figure 54 : Diagramme de classe participantes de « passer commande »

5.7. Cas d'utilisation : Consulter facture

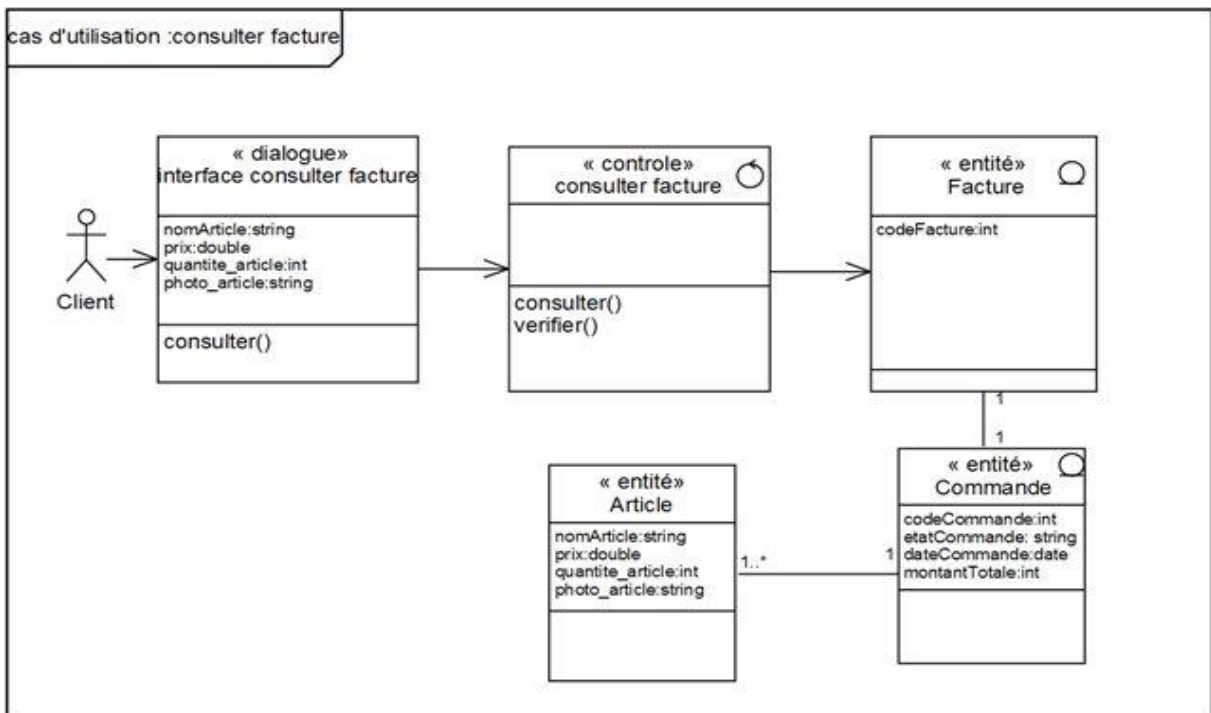


Figure 55 : Diagramme de classe participantes de « consulter facture »

5.8. Cas d'utilisation : Paiement en ligne

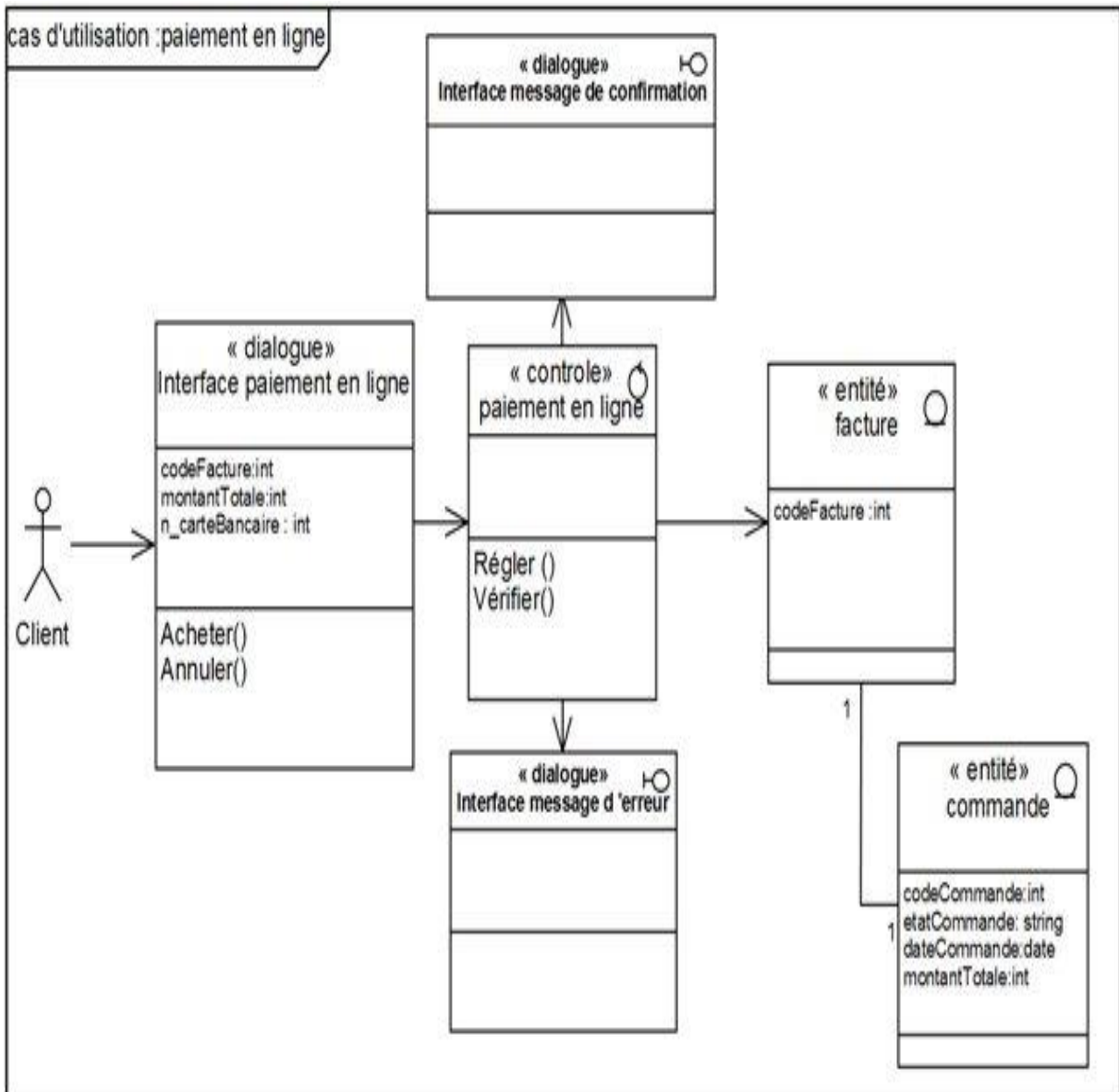


Figure 56 : Diagramme de classe participantes de « paiement en ligne »

5.9. Cas d'utilisation : Modifier les informations du compte

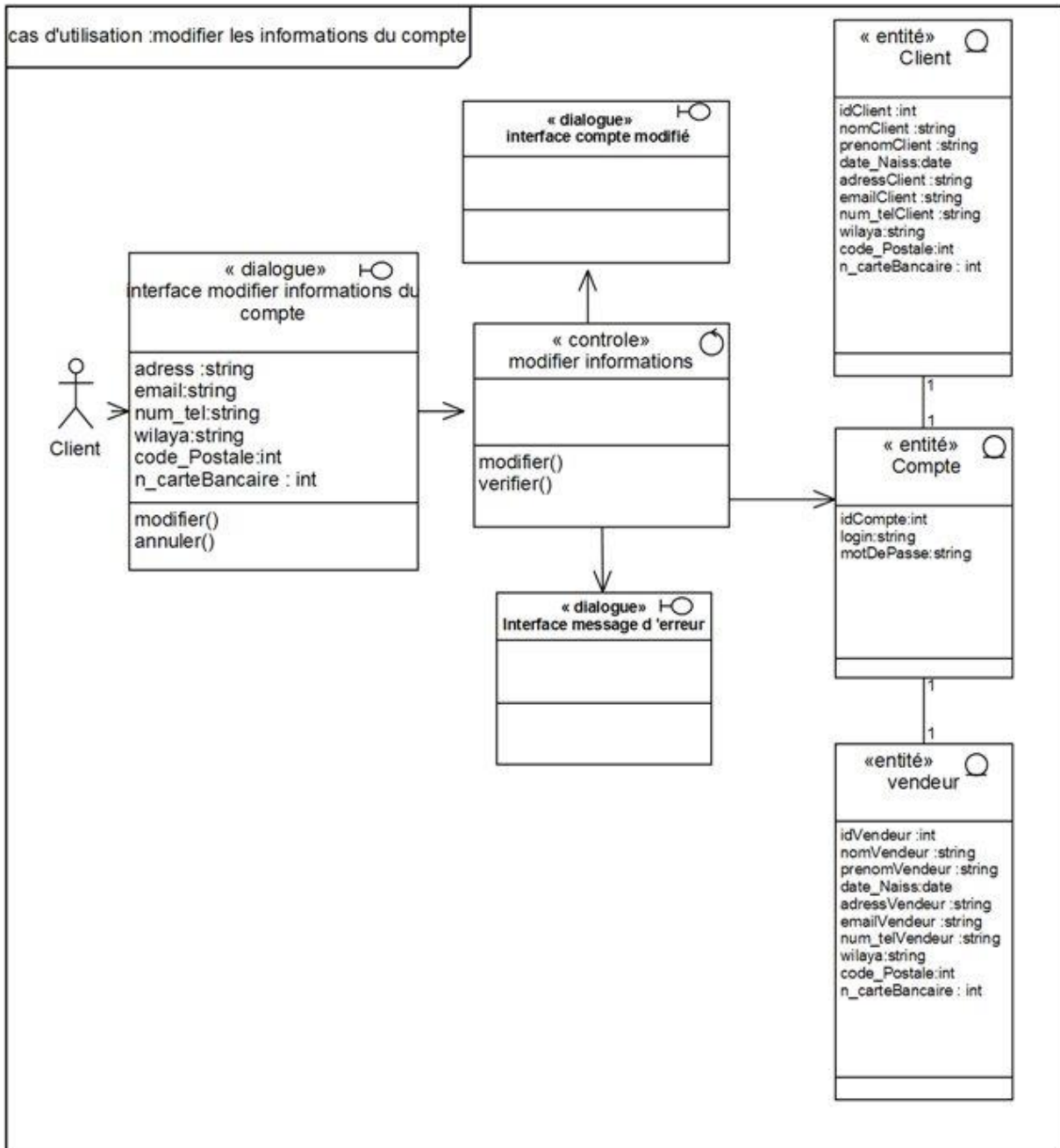


Figure 57 : Diagramme de classe participantes de « modifier les informations du compte »



5.10. Cas d'utilisation : Ajouter article

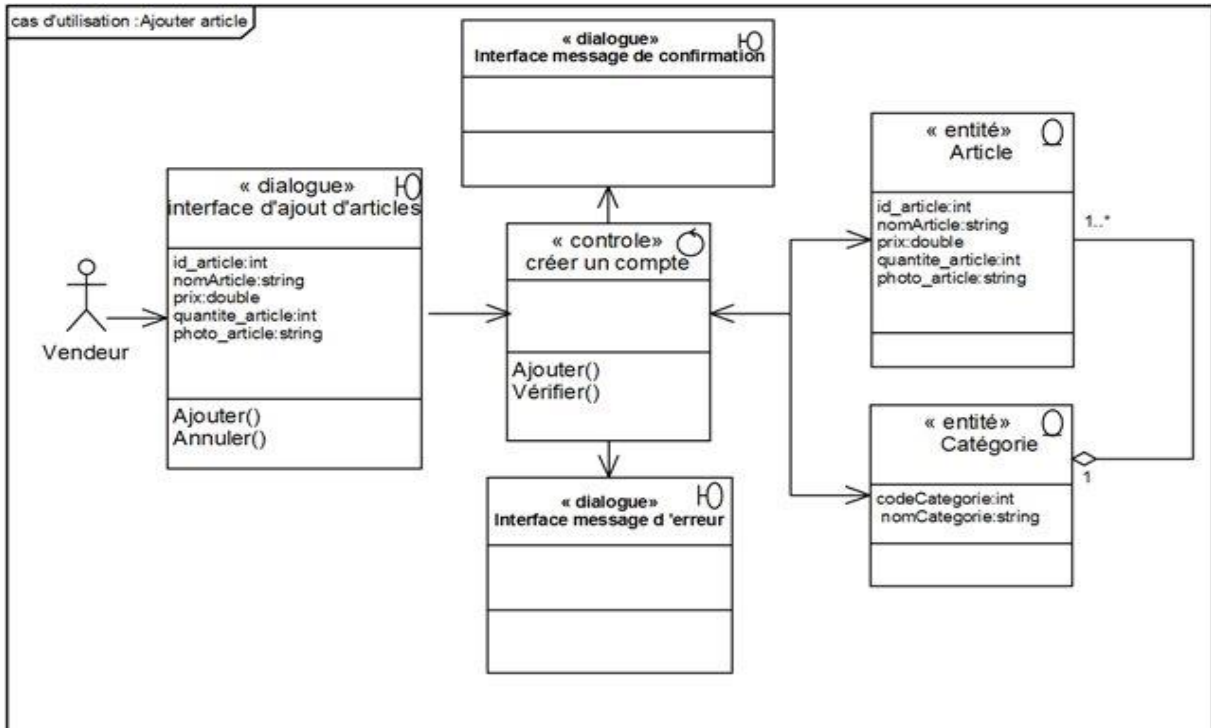


Figure 58 : Diagramme de classe participantes de « ajouter article »

5.11. Cas d'utilisation : Modifier article

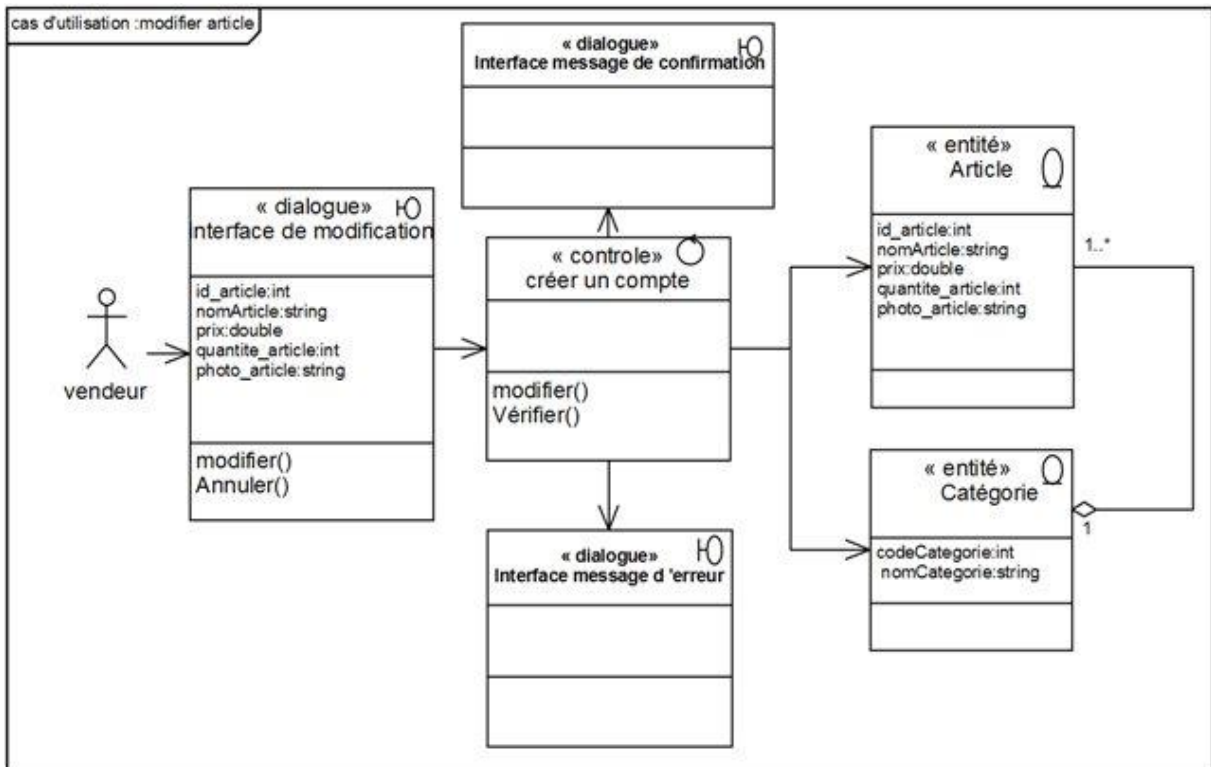


Figure 59 : Diagramme de classe participantes de « modifier article »

5.12. Cas d'utilisation : Supprimer article

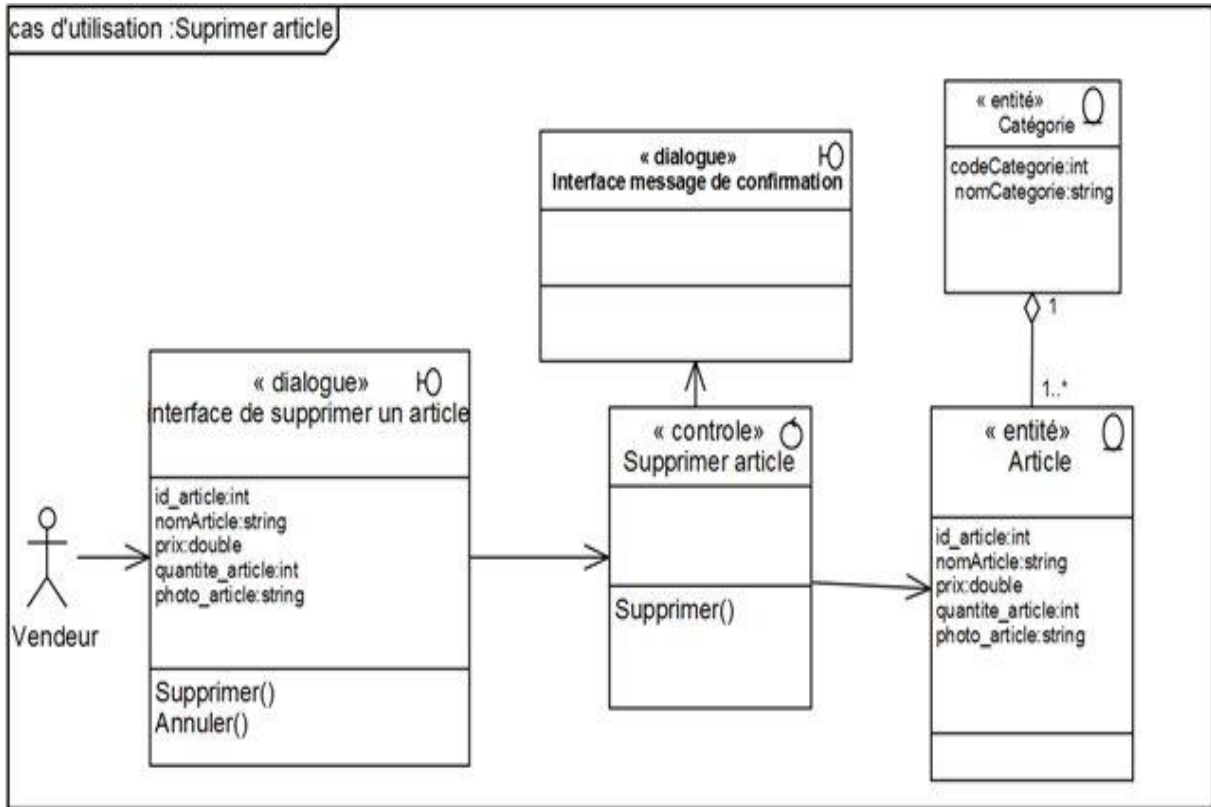


Figure 60 : Diagramme de classe participantes de « supprimer article »

5.13. Cas d'utilisation : Traiter commande

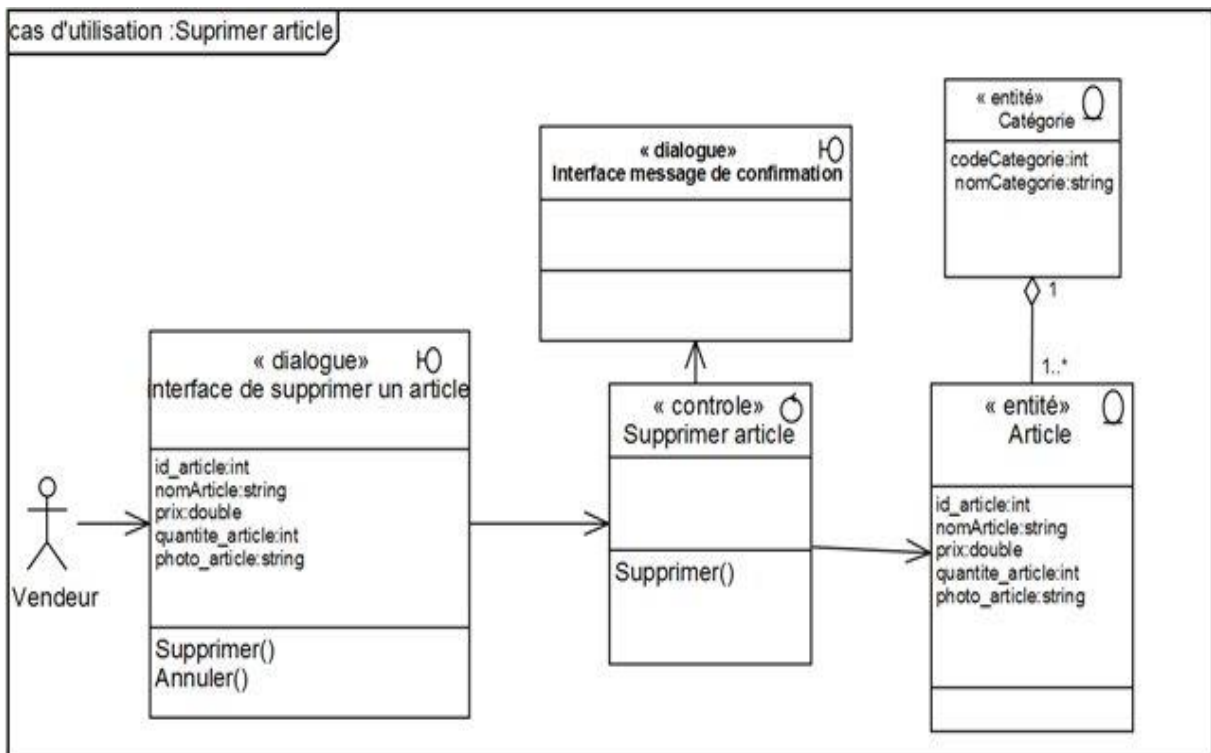


Figure 61 : Diagramme de classe participantes de « Traiter commande »

5.14. Cas d'utilisation : Ajouter compte

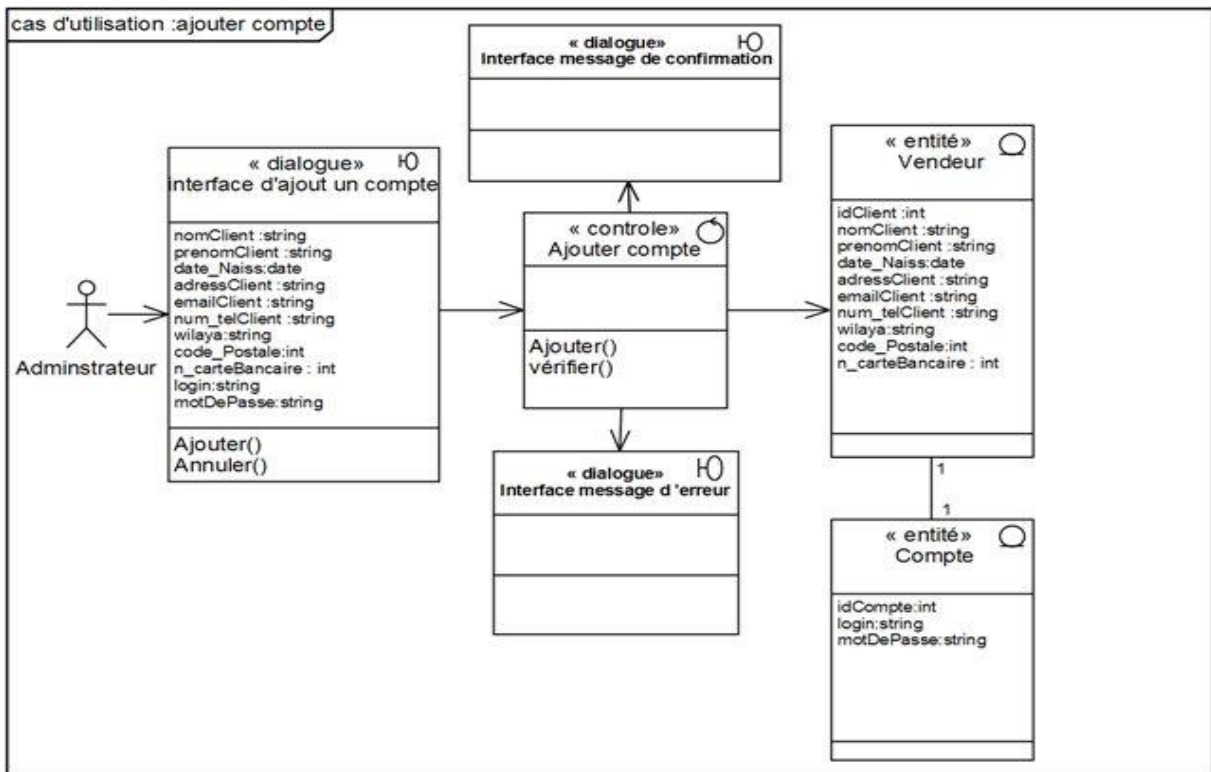


Figure 62 : Diagramme de classe participantes de « ajouter compte »

5.15. Cas d'utilisation : Bloquer compte

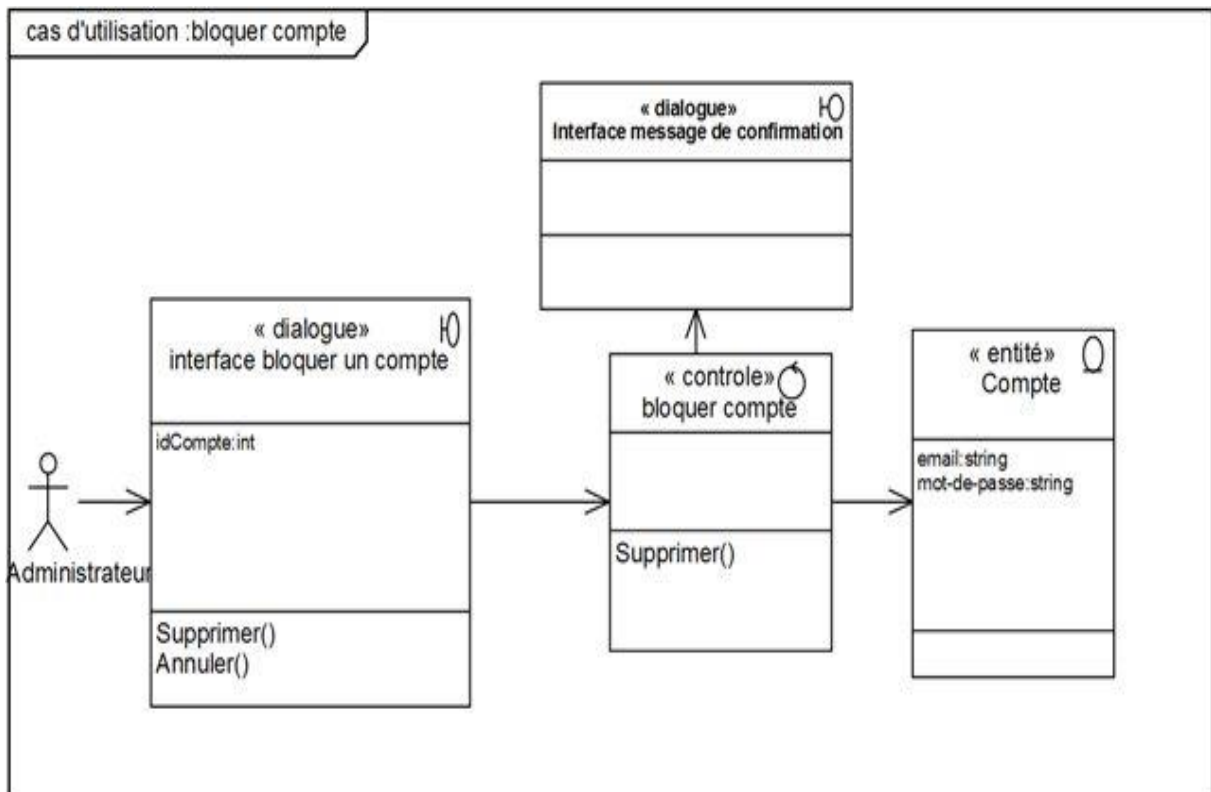


Figure 63 : Diagramme de classe participantes de « Bloquer compte »

5.16. Cas d'utilisation : Statistique

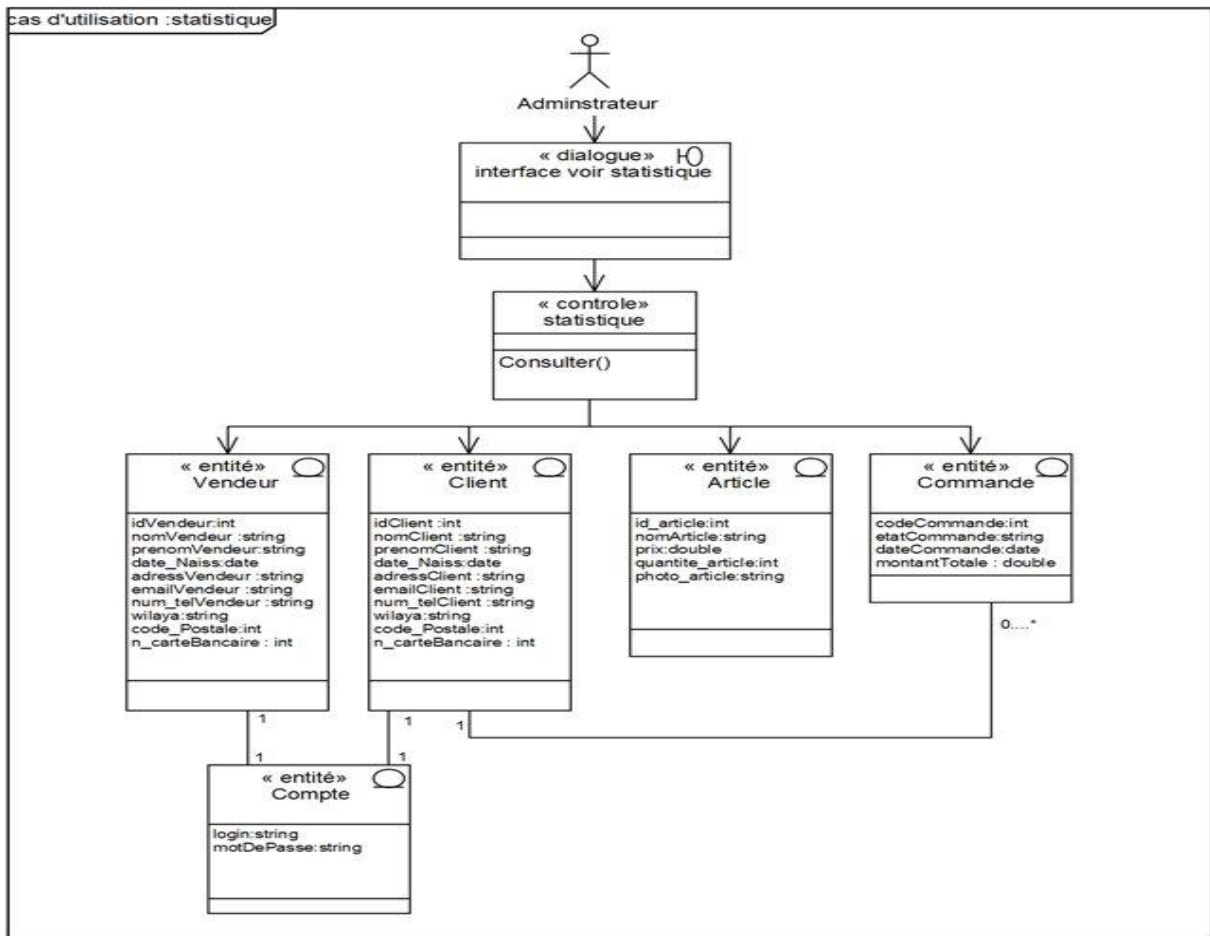


Figure 64 : Diagramme de classe participantes de « statistique »

Conclusion

Dans ce chapitre nous avons présenté les besoins fonctionnels et tous les critères qu'on doit prendre en considération dans la réalisation de ce projet et nous avons recueilli toutes les informations d'aspect fonctionnel du modèle Y.

# Chapitre 04

---

## **Capture des besoins techniques**

## Introduction

Dans ce chapitre, nous présentons la troisième étape du modèle en y, qui est "capture des besoins techniques" elle couvre par complémentarité avec celle des besoins fonctionnels. Et nous présentons l'architecture globale utilisée par notre système sur les deux points de vue matériel et logiciel.

La figure 01 représente la phase de capture des besoins techniques

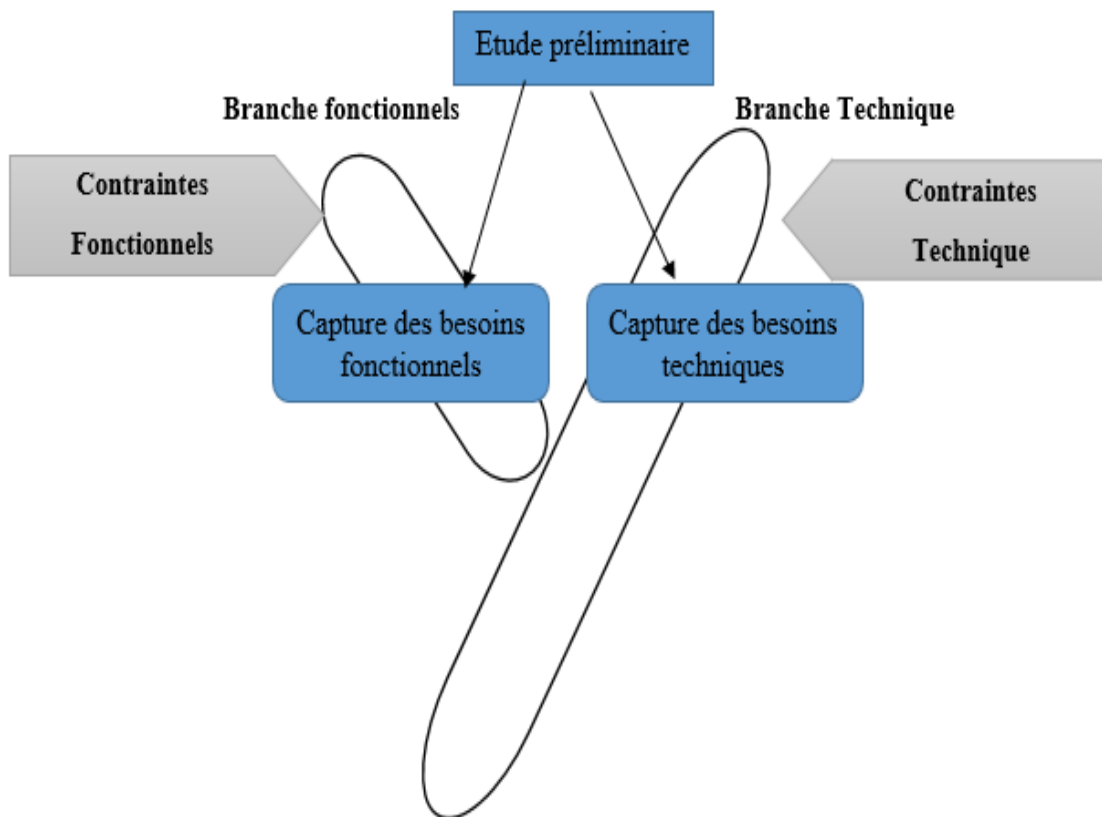


Figure 01 : Phase capture des besoins techniques

## 1. Capture des besoins techniques

La spécification technique est une activité de la branche droite du Y ; elle est primordiale pour la conception d'architecture.

Le modèle s'y exprime suivant les deux points de vue que sont la spécification logicielle et la structure du matériel à exploiter. [01]

La figure suivante représente la démarche de la phase de capture des besoins techniques.

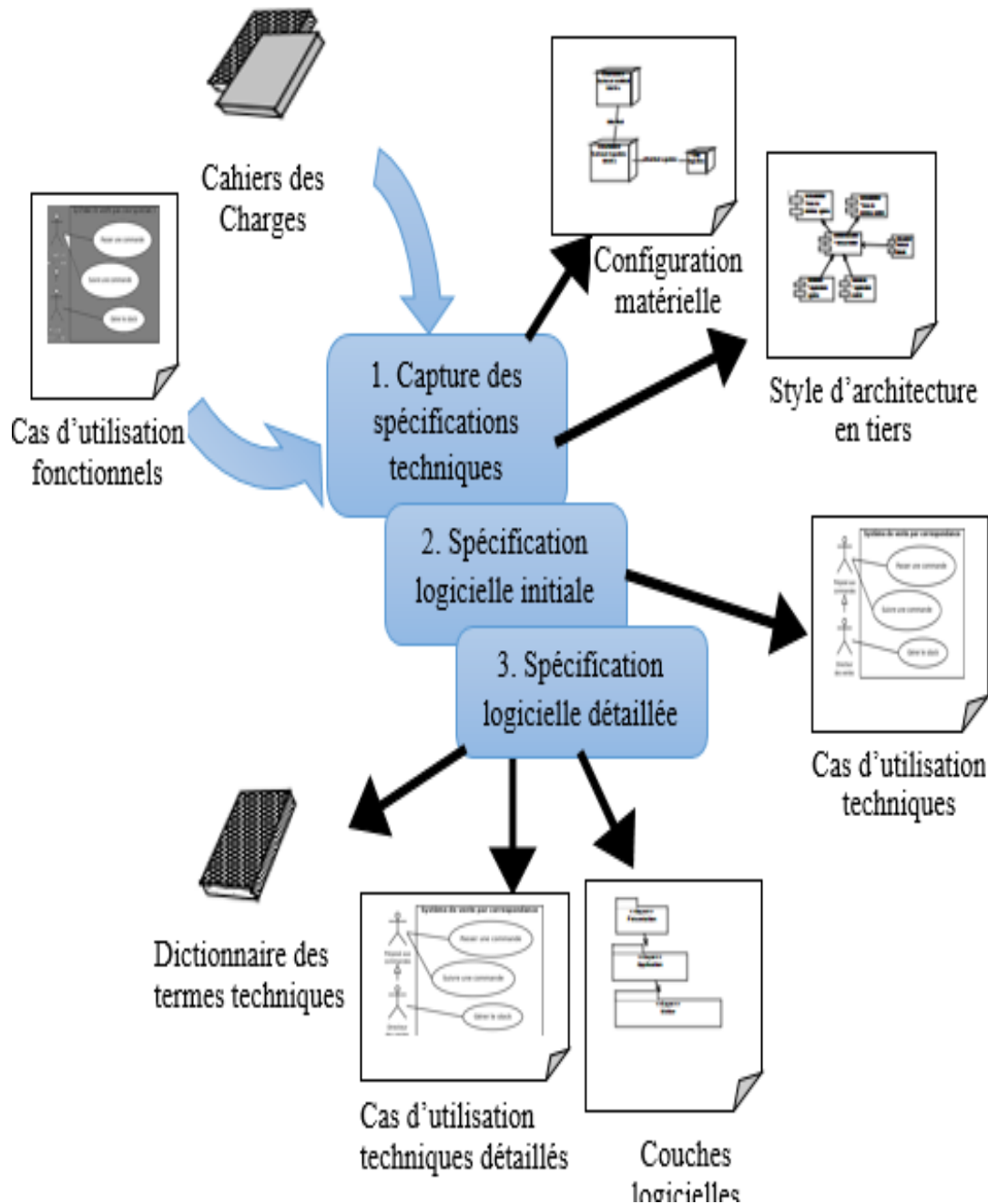


Figure 02 : Construction de l'étape de capture des besoins technique

### 1.1 Spécification technique du point de vue matériel

Les prérequis techniques ont été exprimés dans l'étude préliminaire, lors de l'expression des besoins opérationnels et de celle des choix stratégiques de développement. Ces choix impliquent des contraintes relatives à la configuration du réseau matériel. Elles sont de nature

géographique, organisationnelle, et technique. Elles concernent les performances d'accès aux données, la sécurité du système, la volumétrie. [01]

➤ **Style d'architecture en niveaux :**

Le style d'architecture en niveaux spécifie le nombre de niveaux géographiques et organisationnels où vont se situer les environnements d'exécution du système.

- ✓ L'architecture à deux niveaux met en œuvre un environnement de travail de niveau départemental et local.
- ✓ L'architecture à trois niveaux met en œuvre le système informatique d'une entreprise
- ✓ La conjonction des contraintes géographique et organisationnelle conduit donc à des systèmes complexes dotés d'une architecture multiniveau. [01]

➤ **Les contraintes techniques :**

- ✓ Performances
- ✓ Sécurité
- ✓ Disponibilités. [01]

➤ **La configuration matérielle**

Dans cette partie nous allons présenter l'utilisation de l'infrastructure physique par le système et la manière dont les composants du système sont répartis. [06]



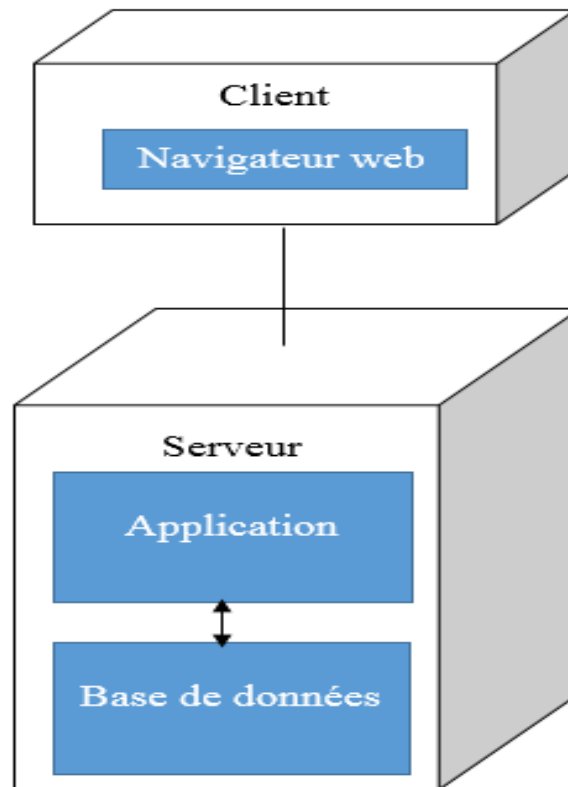


Figure 03 : Configuration matérielle du système

## 1.2 Spécification d'architecture et influence sur le modèle de déploiement

L'expression des prérequis techniques implique également le choix d'un style d'architecture client/serveur. Ce choix conditionne la façon dont seront organisés et déployés les composants d'exploitation du système. [01]

- **Composant d'exploitation**

Un composant d'exploitation est une partie du système logiciel qui doit être connue, installée, déclarée et manipulée par les exploitants du système.

Un composant d'exploitation doit être interchangeable entre différentes versions et peut être arrêté ou démarré séparément. [01]

- **Style d'architecture en tiers**

Le style d'architecture en tiers (tiers signifie « partie » en anglais) spécifie l'organisation des composants d'exploitation mis en œuvre pour réaliser le système. Chaque partie indique une responsabilité technique à laquelle souscrivent les différents composants d'exploitation d'un système. [01]

- **Composant métier**

Un composant métier est un composant d'exploitation dont la fonction est de distribuer les services d'un ou de plusieurs objets métier de l'entreprise. L'intégration de composant métier implique le recours à un style d'architecture 3-tiers. [01]

### 1.3 Élaboration du modèle de spécification logicielle

- **Exploitant**

L'exploitant est un acteur au sens d'UML, si ce n'est qu'il ne bénéficie que des fonctionnalités techniques du système. [01]

- **Cas d'utilisation technique**

Un cas d'utilisation technique est destiné à l'exploitant, c'est une séquence d'actions produisant une valeur ajoutée opérationnelle ou purement technique. [01]

#### 1.3.1 Identification des cas d'utilisation techniques

Les exploitants du système sont :

- **L'utilisateur**, qui utilise l'application du système.
- **L'ingénieur d'exploitation**, qui est chargé de déployer et de dépanner le système. [01]

Les cas d'utilisation techniques du système sont :

- ✓ Gérer l'intégrité.
- ✓ Gérer les erreurs.
- ✓ Gérer la sécurité.

- **Modèle de spécification logicielle**

Dans la figure suivante nous allons montrer le modèle de spécification logicielle du système.

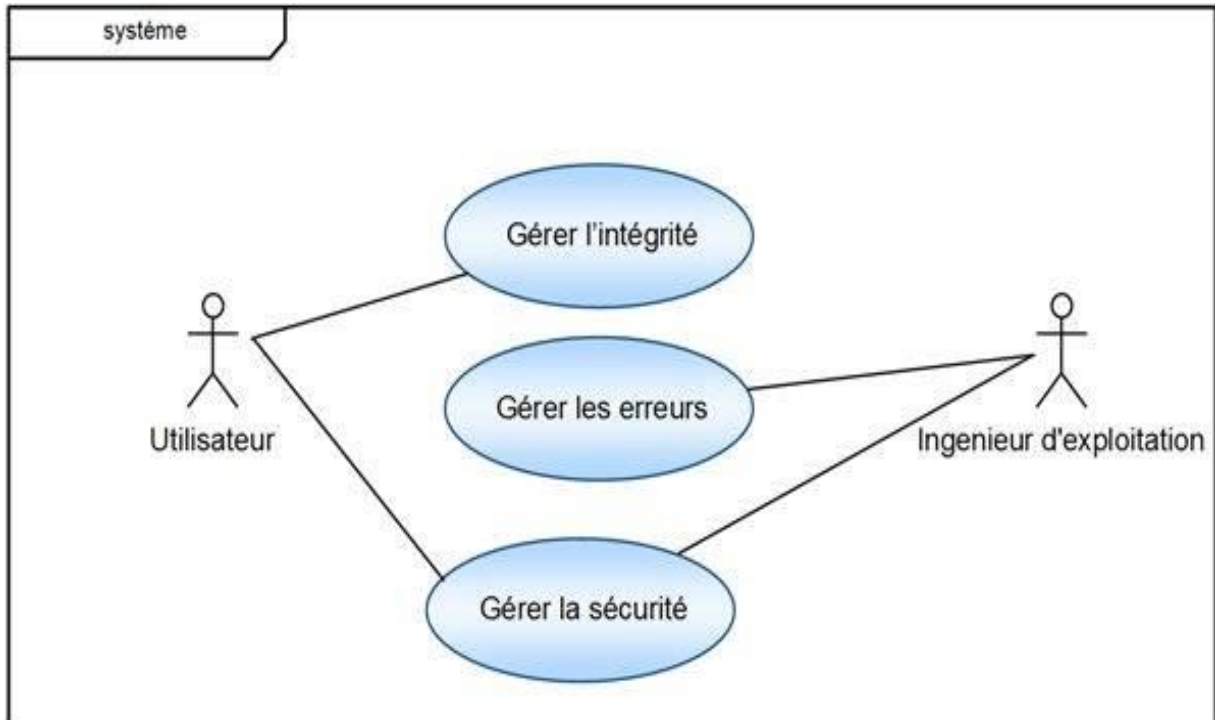


Figure 04 : Modèle de spécification logicielle du système

#### 1.4 Organisation du modèle de spécification logicielle

Le modèle de spécification logicielle obtenu est cependant trop sommaire pour permettre une spécification technique détaillée. [1]

- **Couche logicielle**

Une couche logicielle représente un ensemble de spécifications ou de réalisations qui respectivement expriment ou mettent en œuvre un ensemble de responsabilités techniques et homogènes pour un système logiciel.

Cette organisation correspond au style d'architecture en couches préconisé pour le développement d'une solution client/serveur. [1]

Dans la figure qui suit nous montrons le style d'architecture de notre système.

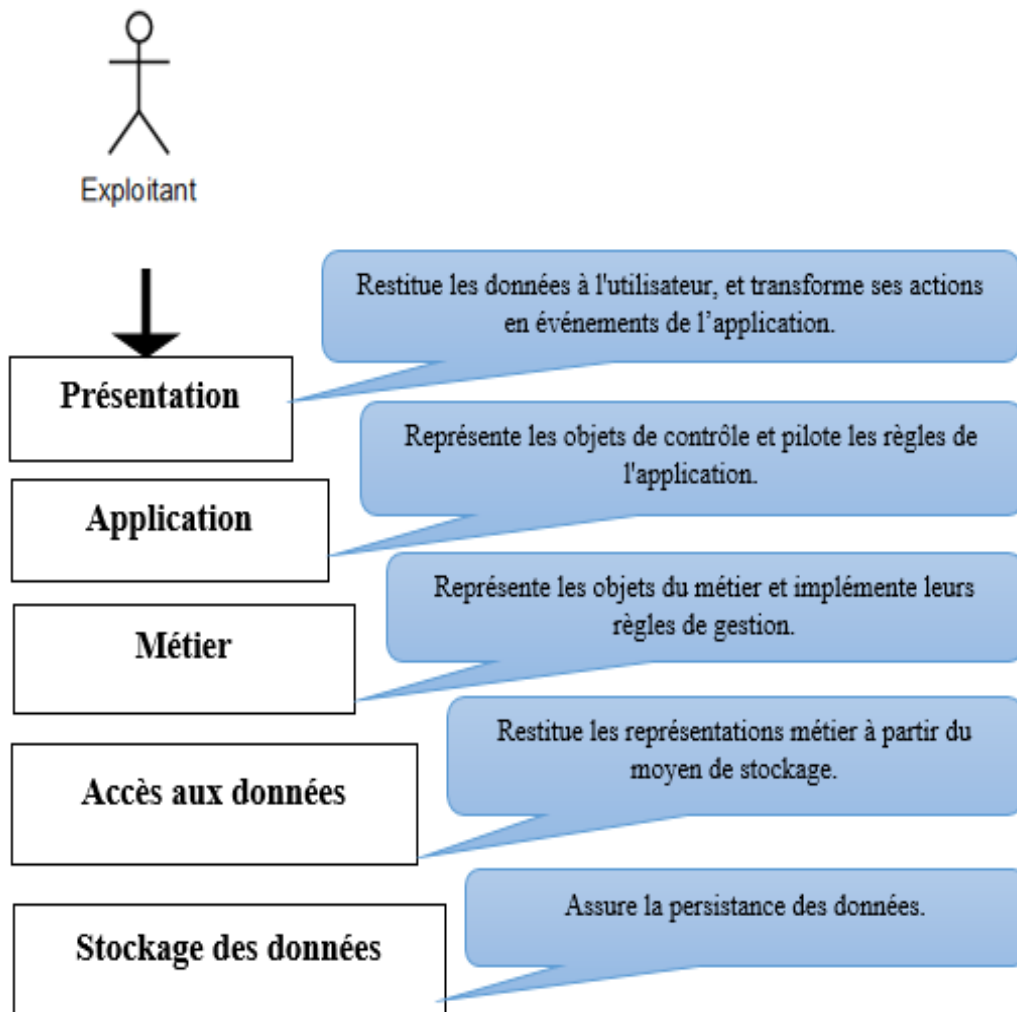


Figure 05 : Style d'architecture en 5 couches

#### 1.4.1 Organisation en couches du modèle de spécification

Le style d'architecture en cinq couches structure le modèle de spécification logicielle. Les couches s'organisent suivant les dépendances qui s'établissent entre elles. [1]

Dans le modèle UML, les couches logicielles correspondent à des packages, la figure suivante représente l'organisation du modèle de spécification logicielle.

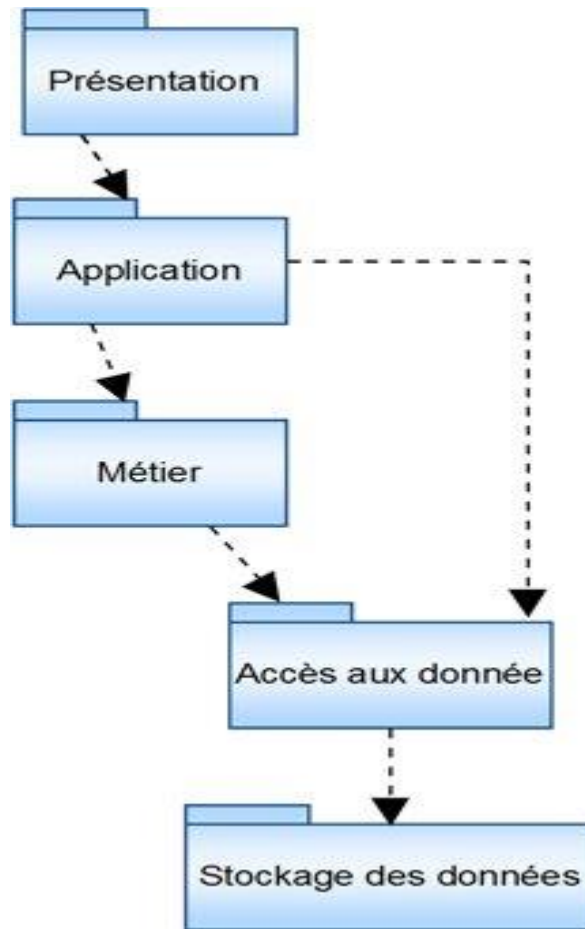
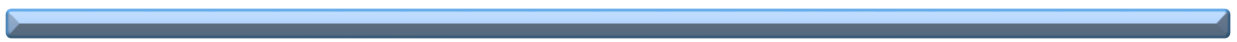


Figure 06 : Organisation du modèle de spécification logicielle

## Conclusion

Dans ce chapitre nous avons présenté les besoins technique et tous les critères qu'on doit prendre en considération dans la réalisation de ce projet et nous avons recueilli toutes les informations d'aspect techniques du modèle Y.

# Chapitre 05



## Analyse

## Introduction

Dans ce chapitre, et après les phases de l'étude préliminaire, capture des besoins fonctionnels, et capture des besoins techniques étudiés dans les chapitres 2 et 3, nous présentons trois activités de la phase d'analyse, qui sont découpage en catégorie, le développement du modèle statique et développement du modèle dynamique.

La figure suivante montre la phase d'analyse.

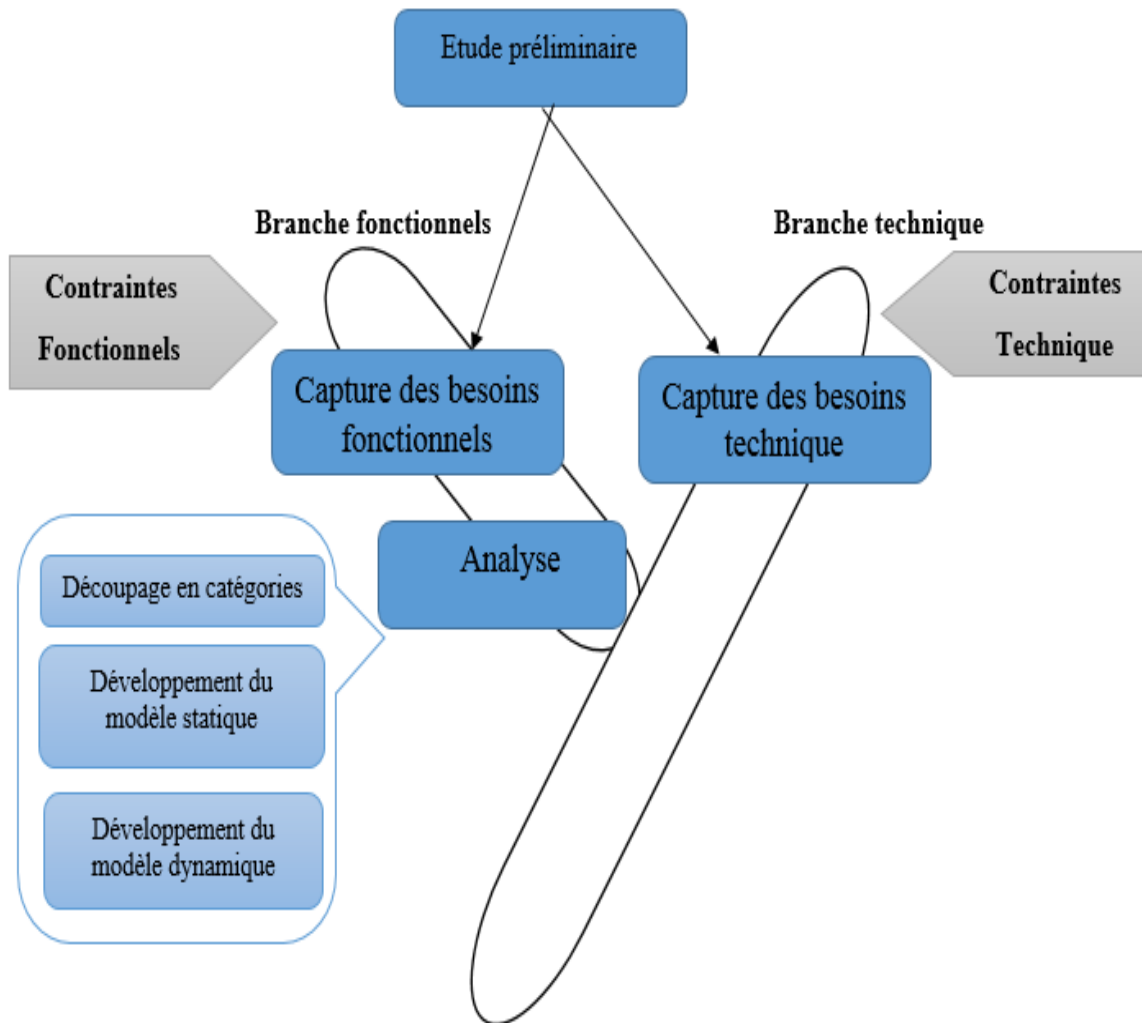


Figure 01 : Phase d'analyse

### 1. Découpage en catégorie

Le découpage en catégories constitue la première activité de l'étape d'analyse. Il se situe sur la branche gauche du cycle en Y et succède à la capture des besoins fonctionnels. [01]

La figure 02 représente la démarche de la première activité de la phase d'analyse qui est le découpage en catégorie.

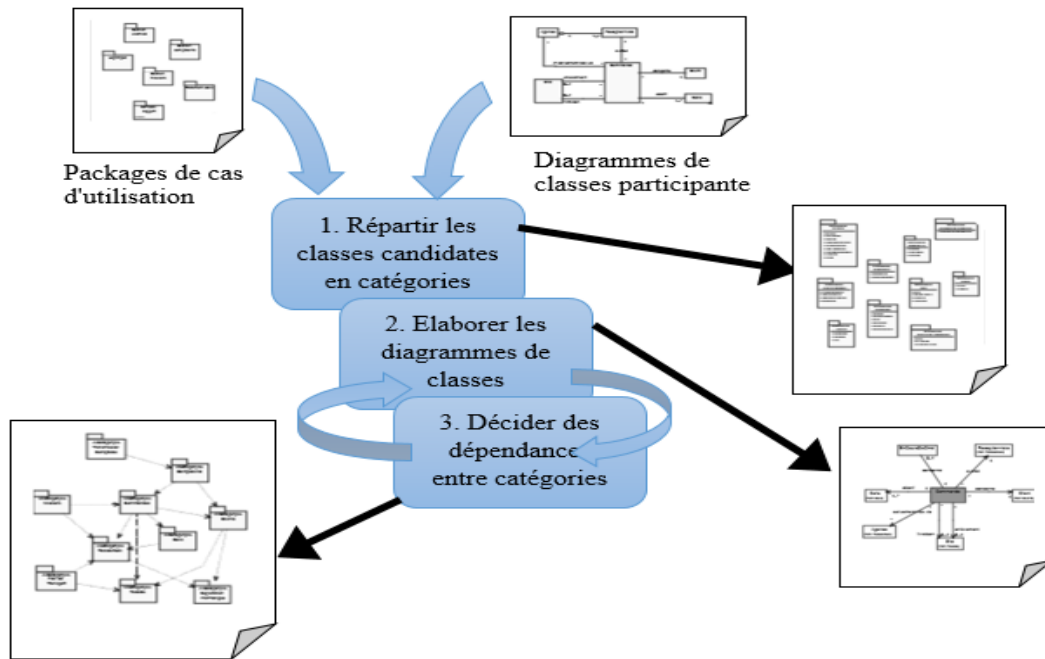


Figure 02 : Démarche du découpage en catégorie

- **Catégorie**

Une catégorie consiste en un regroupement logique de classes à forte cohérence interne et faible couplage externe. [01]

### 1.1 Répartition des classes candidates en catégories

La figure suivante montre l'activité du découpage en catégorie.

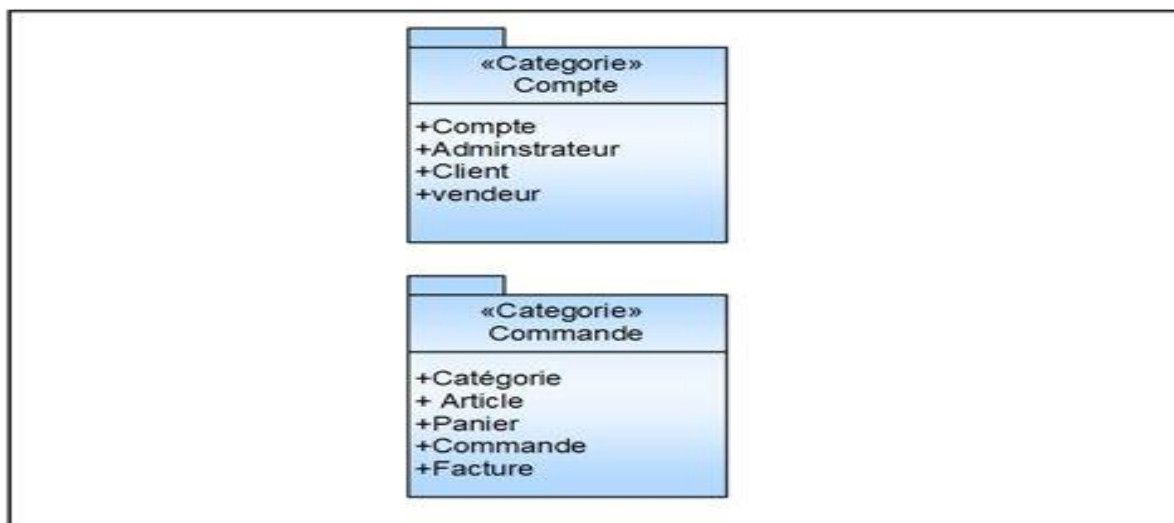


Figure 03 : Le découpage en catégorie



1.2 Elaboration des diagrammes de classes préliminaires par catégorie

La figure 04 représente le diagramme de classes préliminaires par catégorie.

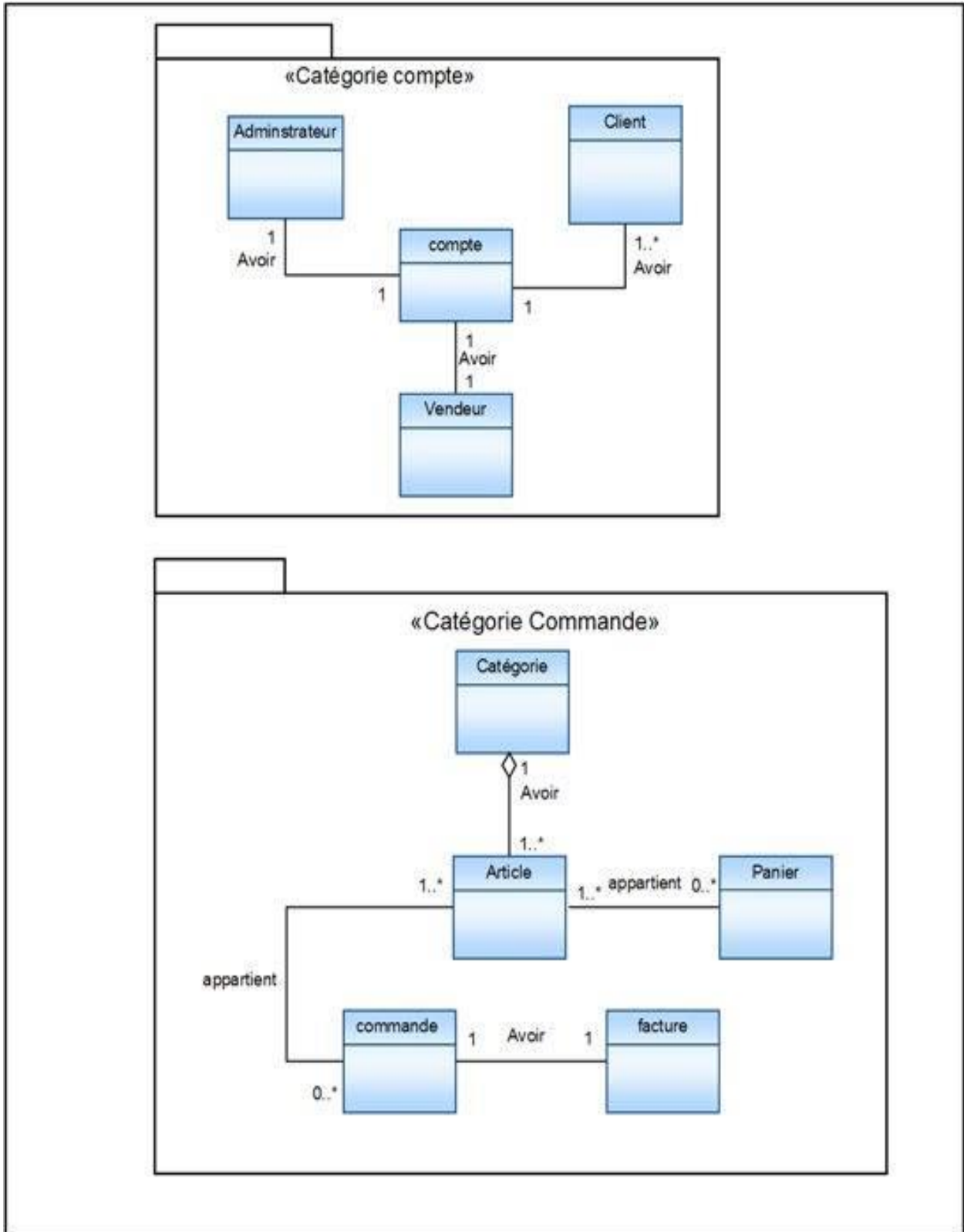


Figure 04 : Diagramme de classes préliminaires par catégorie.

1.3. Associations entre catégories

La figure 05 représente les associations entre les catégories.

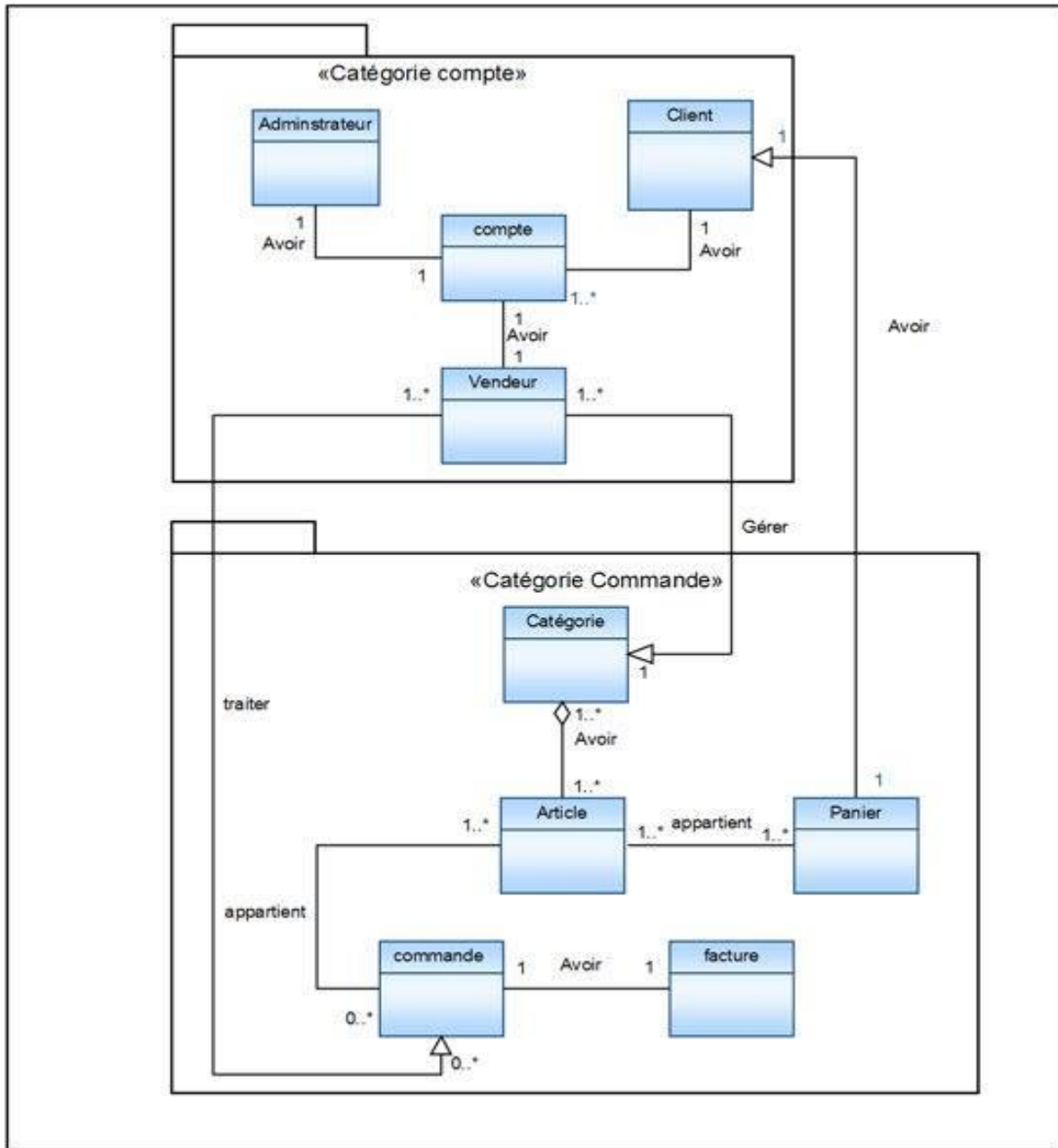


Figure 05 : Dépendance entre catégorie

2. Développement du modèle statique

Le développement du modèle statique constitue la deuxième activité de l'étape d'analyse. Elle se situe sur la branche gauche du cycle en Y et succède au découpage en catégories. [01]

La figure suivante illustre la démarche d'élaboration du modèle statique.

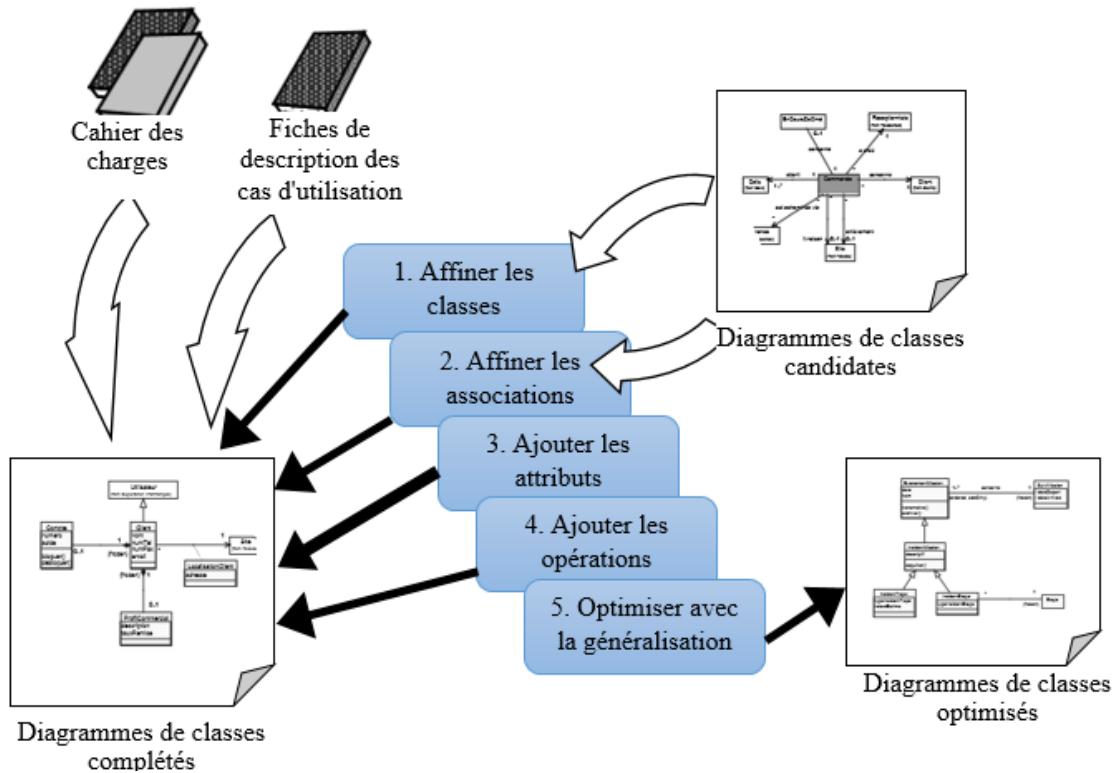


Figure 06 : Démarche d’élaboration du modèle statique

### 2.1. Diagramme de classes pour la catégorie compte

Les deux figures qui suivent représentent le diagramme de classe préliminaires pour les deux catégories « compte » et « commande ».

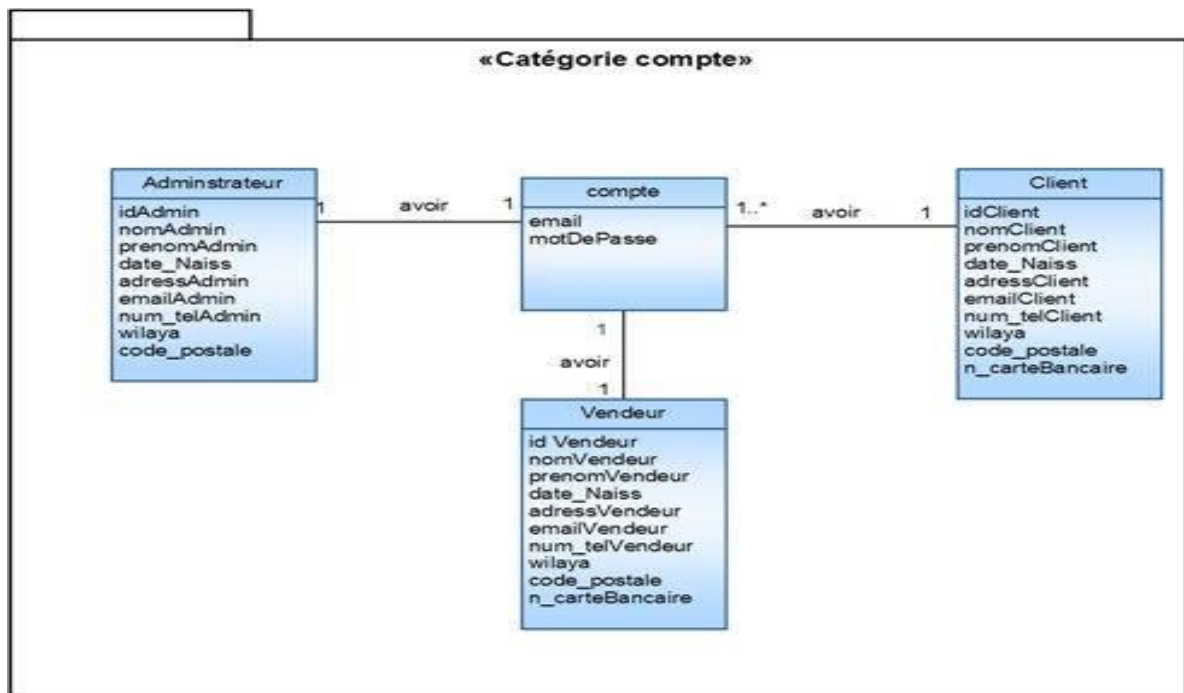


Figure 07 : diagramme de classe pour la catégorie compte

2.2 Diagramme de classes pour la catégorie commande

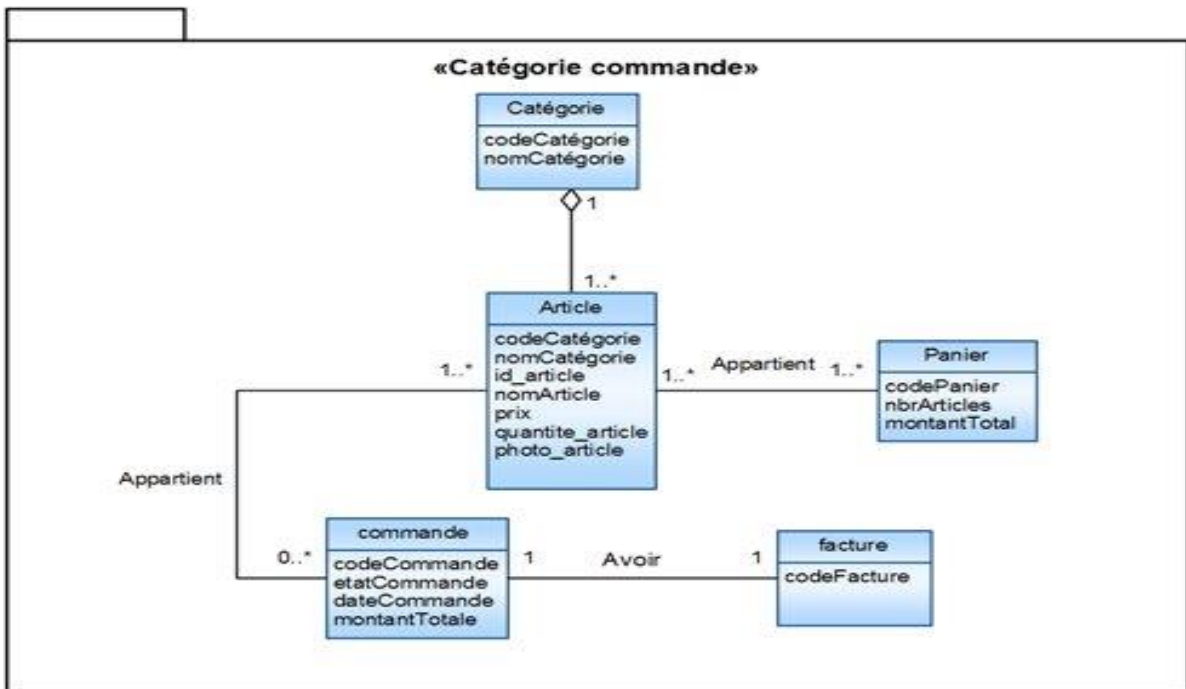


Figure 08 : diagramme de classe pour la catégorie commande

3. Développement du modèle dynamique

Le développement du modèle dynamique constitue la troisième activité de l'étape d'analyse. Elle se situe sur la branche gauche du cycle en Y. Il s'agit d'une activité itérative, fortement couplée avec l'activité de modélisation statique. [01]

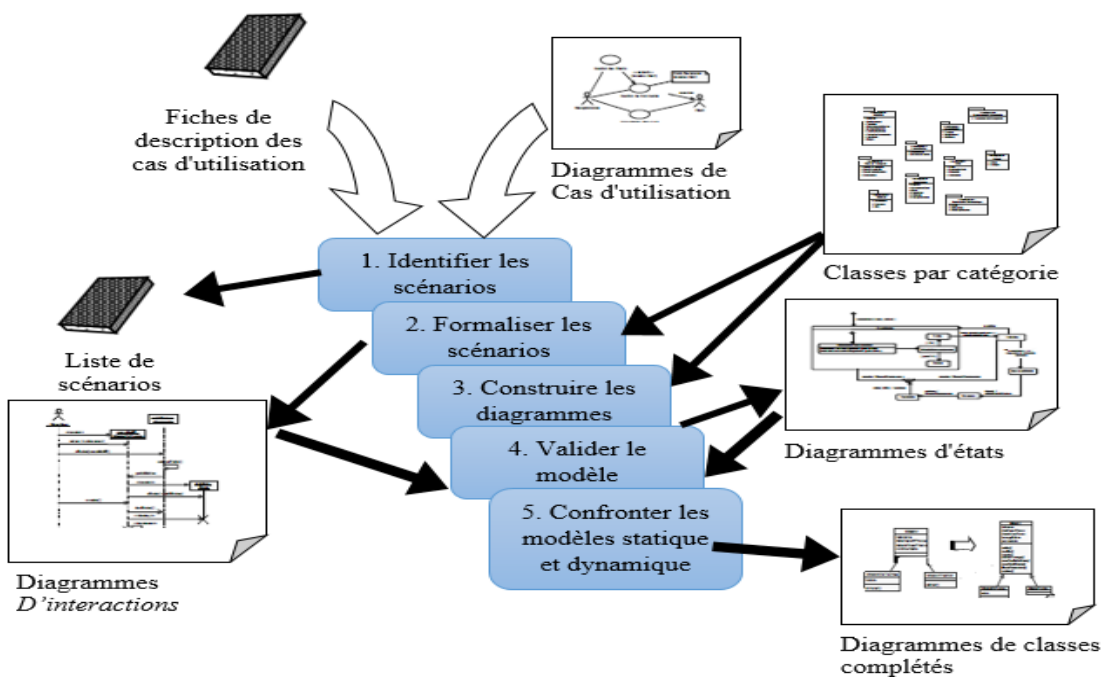


Figure 09 : Démarche d'élaboration du modèle dynamique

### 3.1. Les diagrammes d'interaction

Les figures qui suivent représentent le diagramme d'interaction pour chaque cas d'utilisation.

➤ Diagramme d'interaction du cas «créer compte»

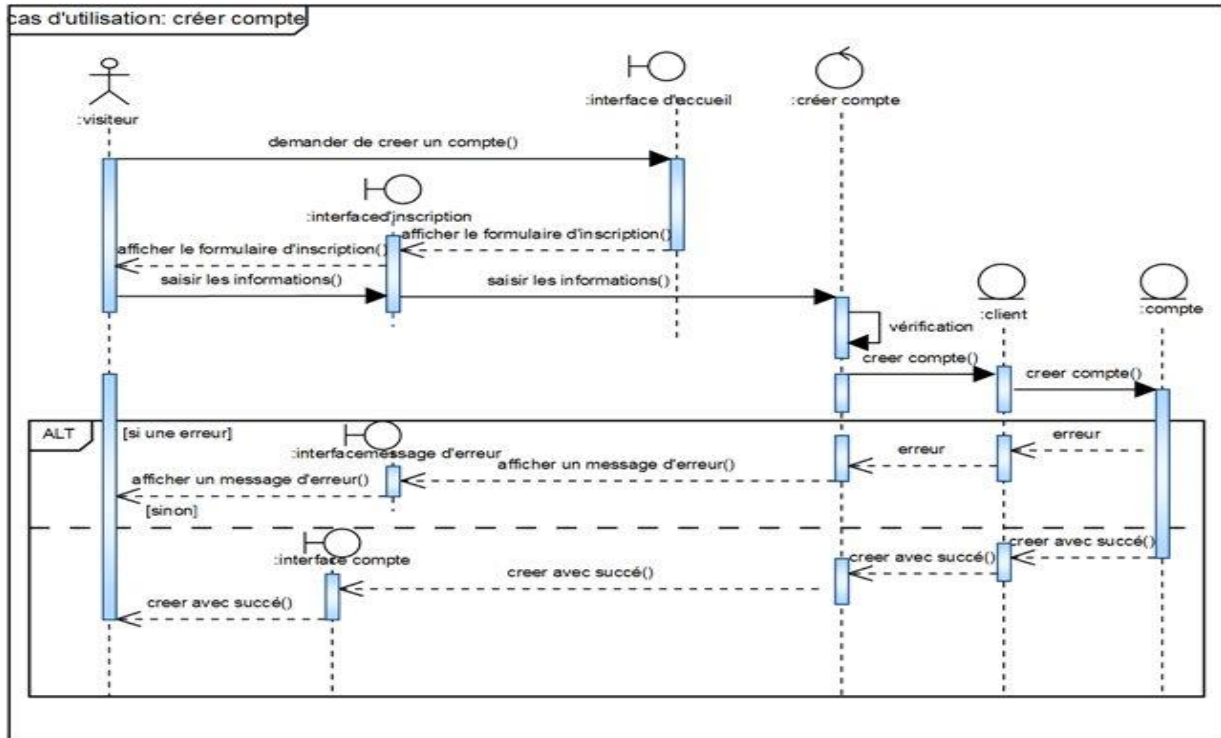


Figure 10 : Le diagramme d'interaction de cas d'utilisation « créer compte »

➤ Diagramme d'interaction du cas «consulter article»

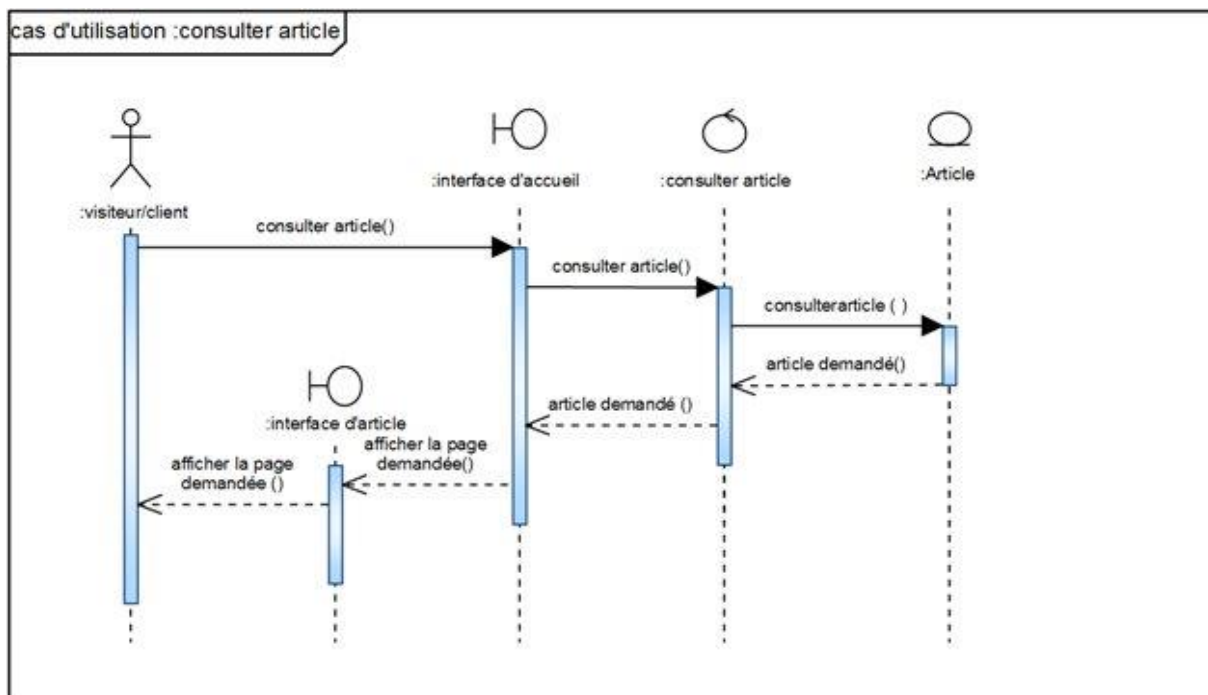


Figure 11 : Le diagramme d'interaction de cas d'utilisation « consulter article »

➤ Diagramme d'interaction du cas «rechercher article»

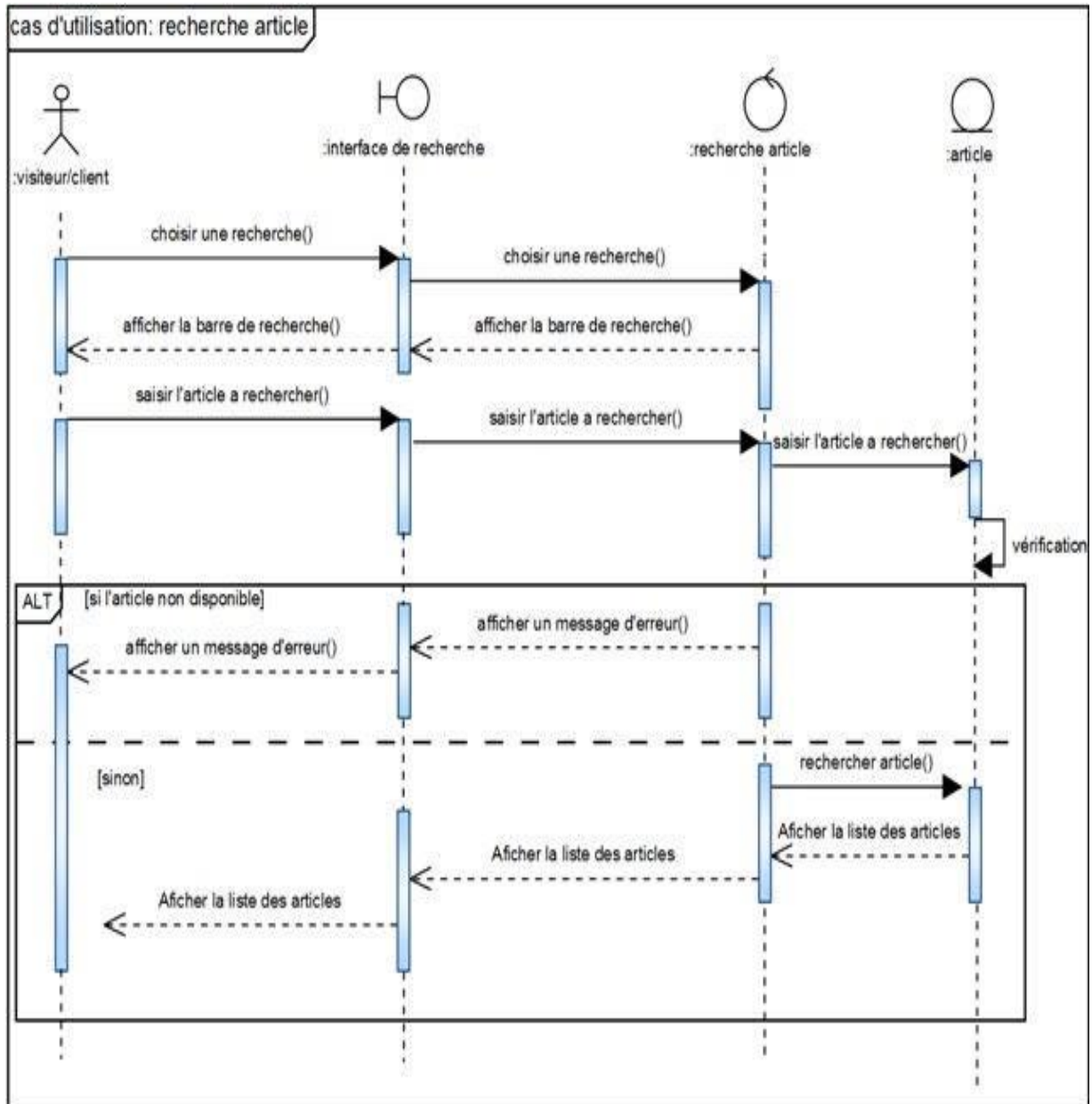


Figure 12 : Le diagramme d'interaction de cas d'utilisation « rechercher article »

➤ Diagramme d'interaction du cas «gérer panier»

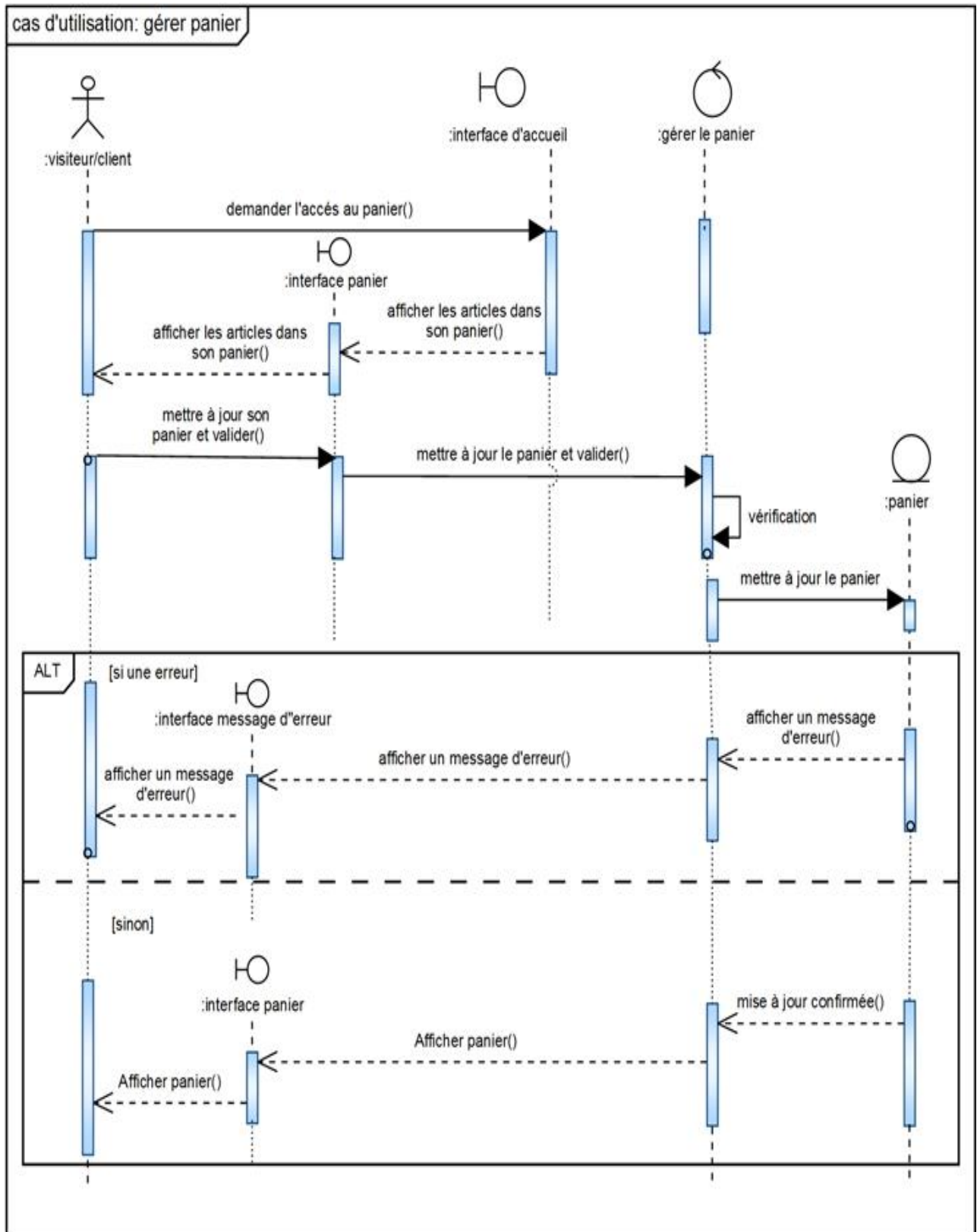


Figure 13 : Le diagramme d'interaction de cas d'utilisation « gérer panier »

➤ Diagramme d'interaction du cas «s'authentifier»

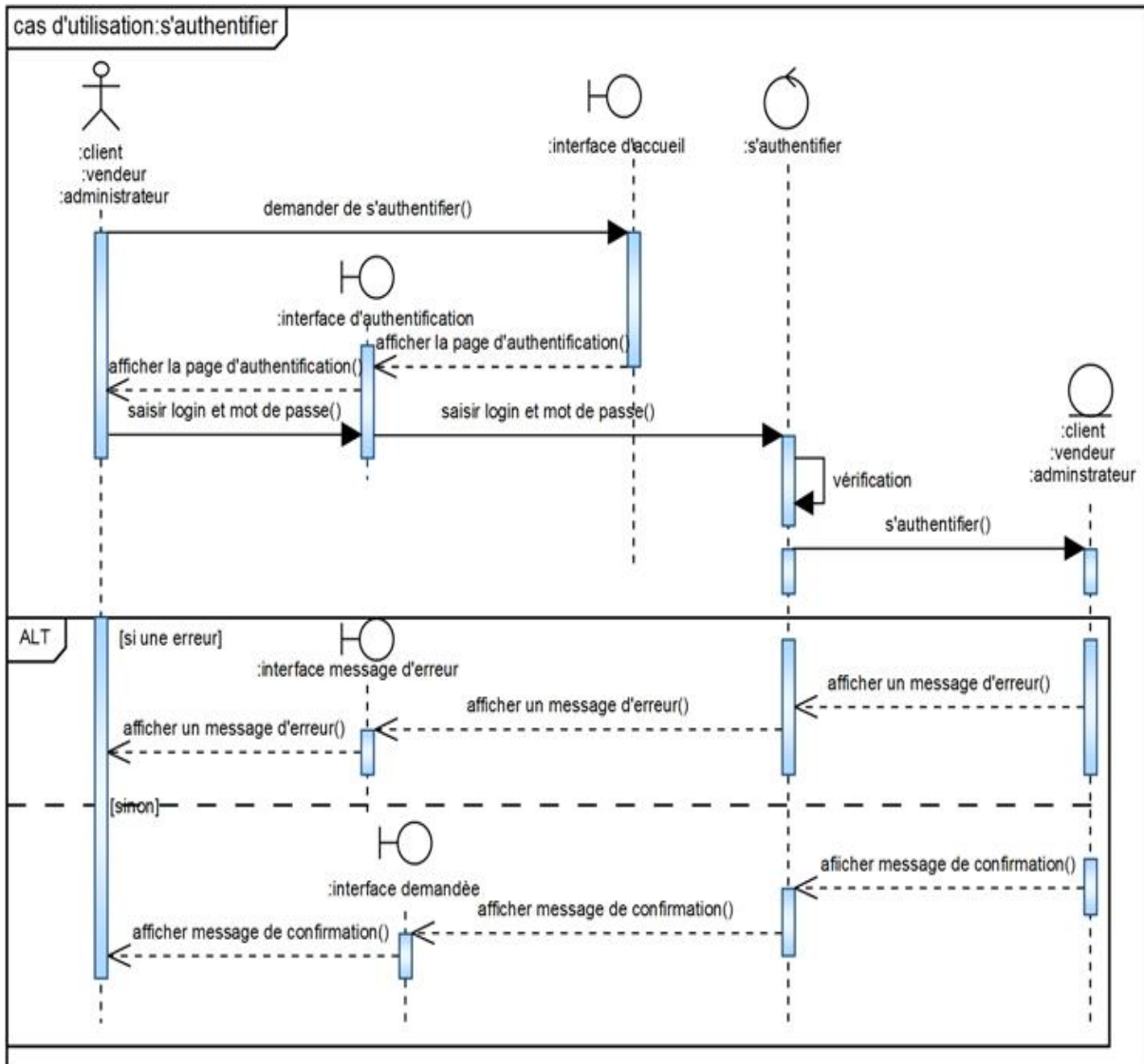


Figure 14 : Le diagramme d'interaction de cas d'utilisation « s'authentifier »



➤ Diagramme d'interaction du cas «passer commande»

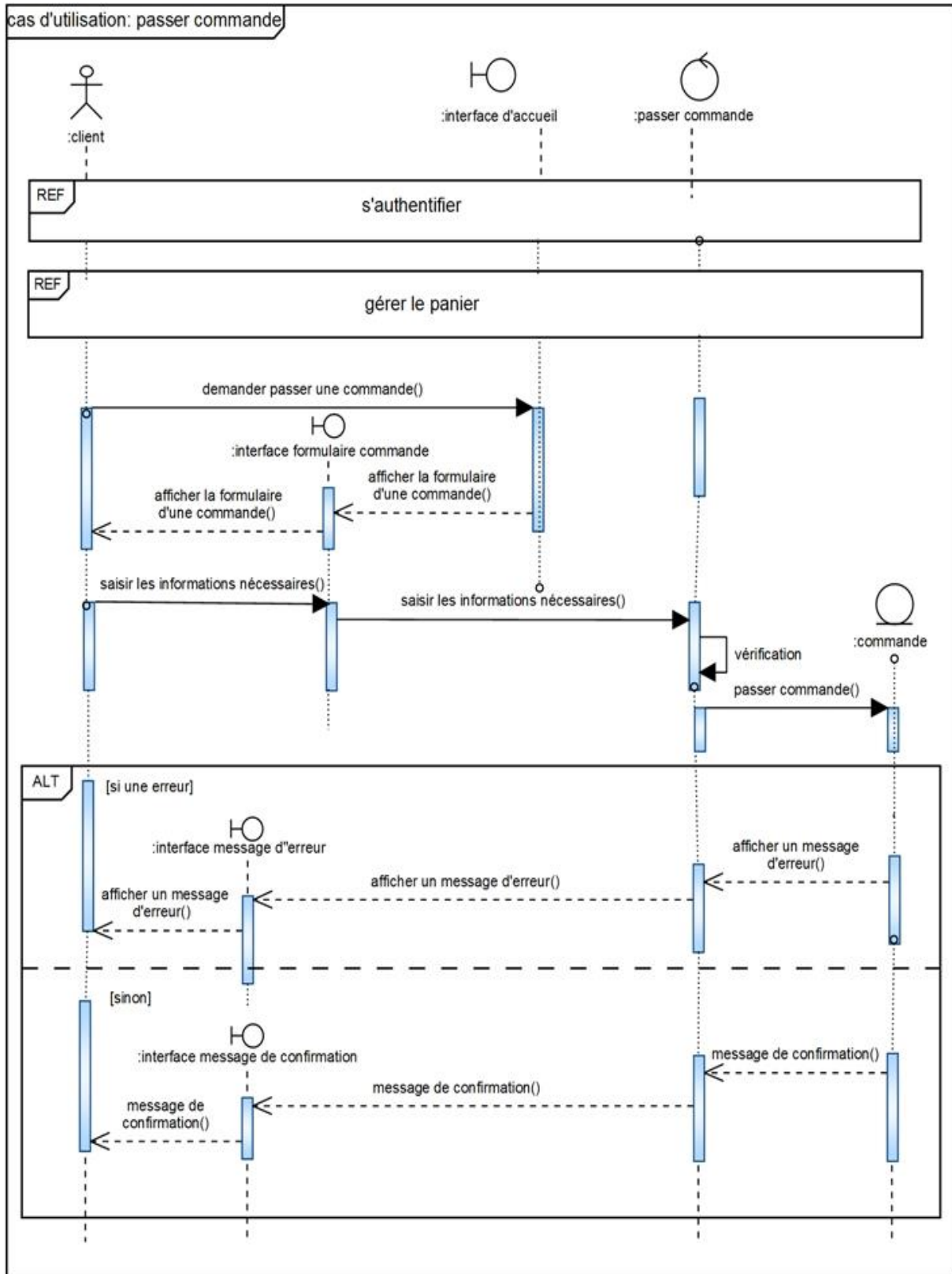


Figure 15 : Le diagramme d'interaction de cas d'utilisation « passer commande »

➤ Diagramme d'interaction du cas «consulter facture»

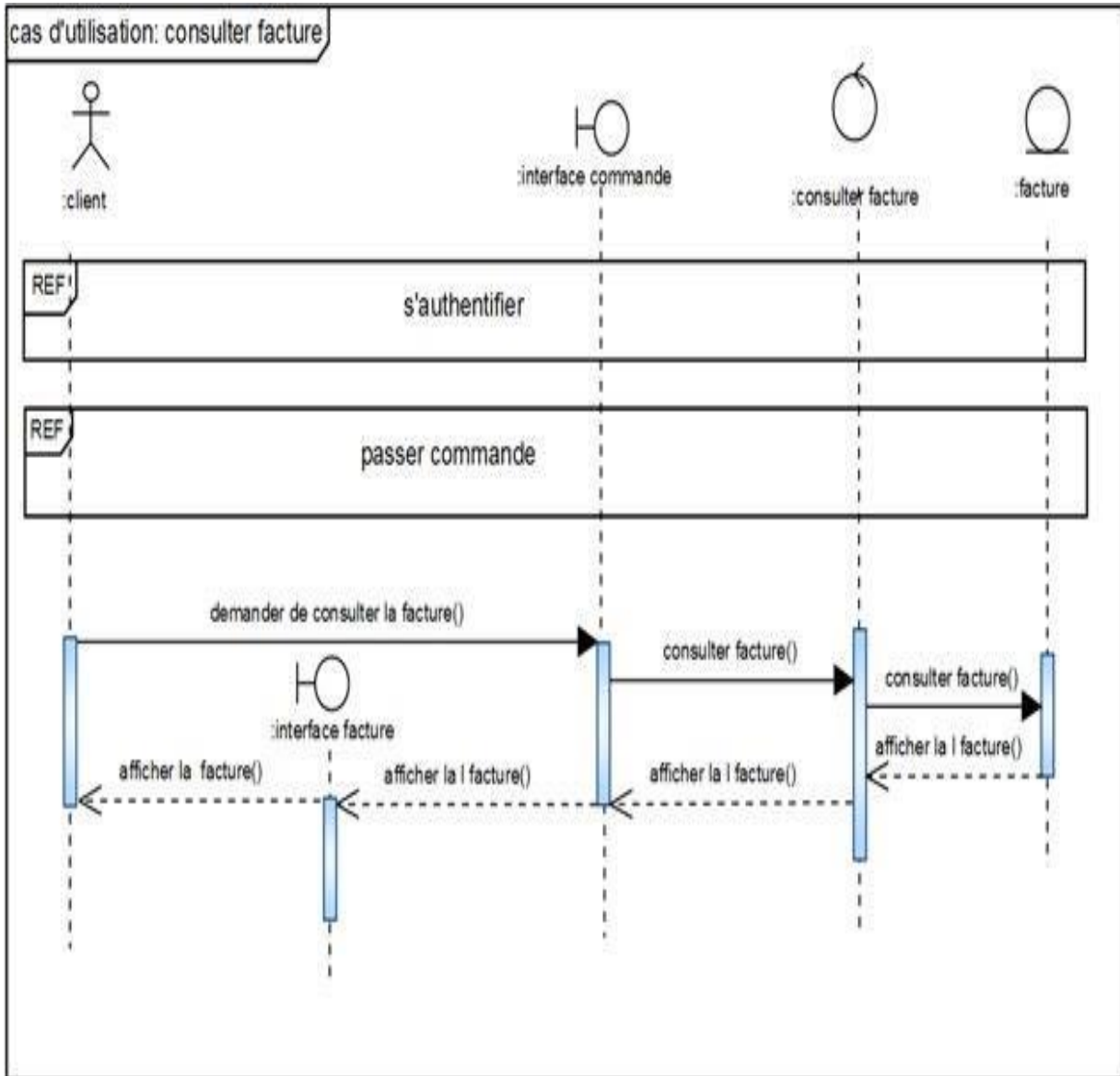


Figure 16 : Le diagramme d'interaction de cas d'utilisation « consulter facture »

➤ Diagramme d'interaction du cas «paiement en ligne»

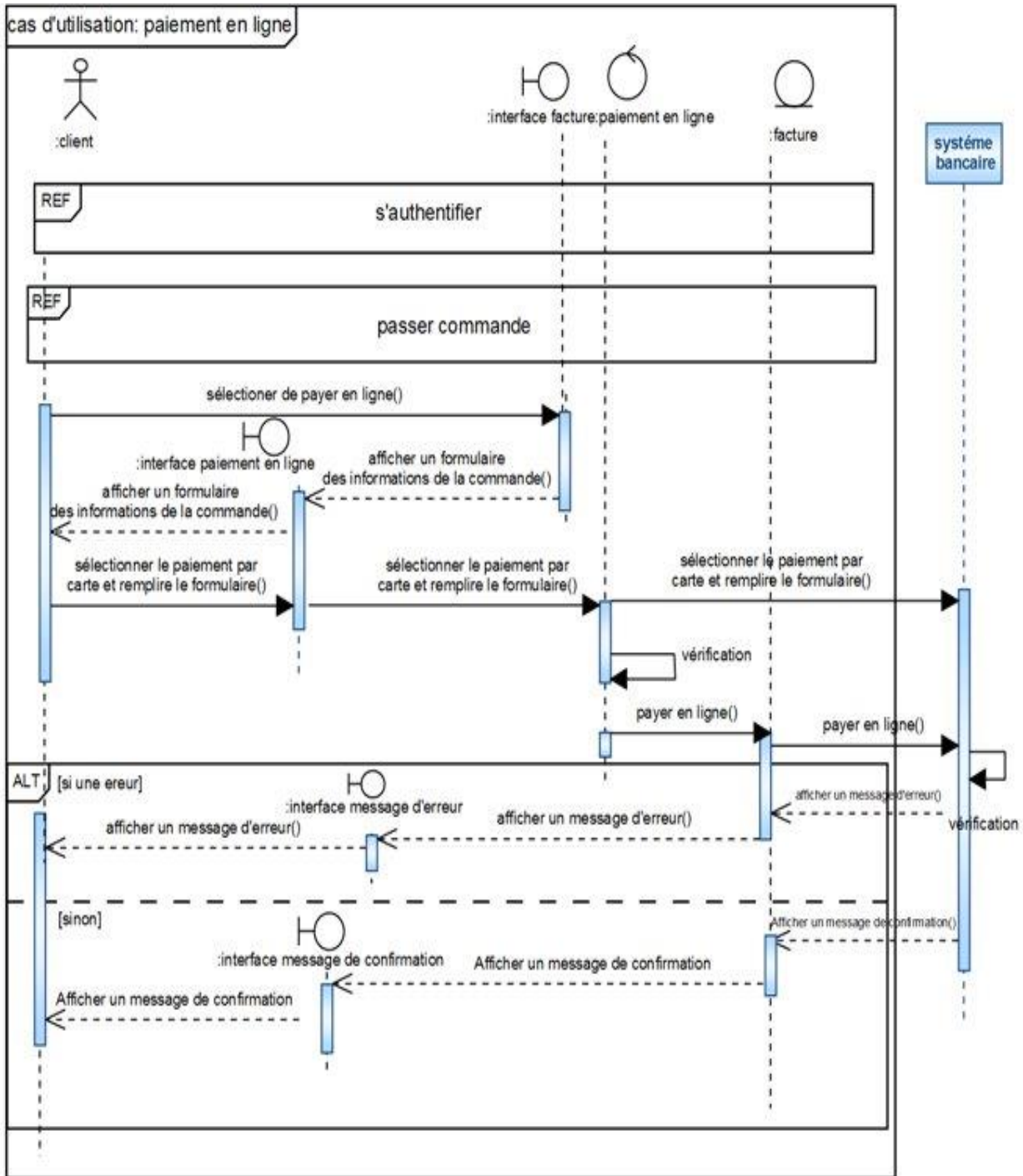


Figure 17 : Le diagramme d'interaction de cas d'utilisation «paiement en ligne»

➤ Diagramme d'interaction du cas «modifier les informations du compte»

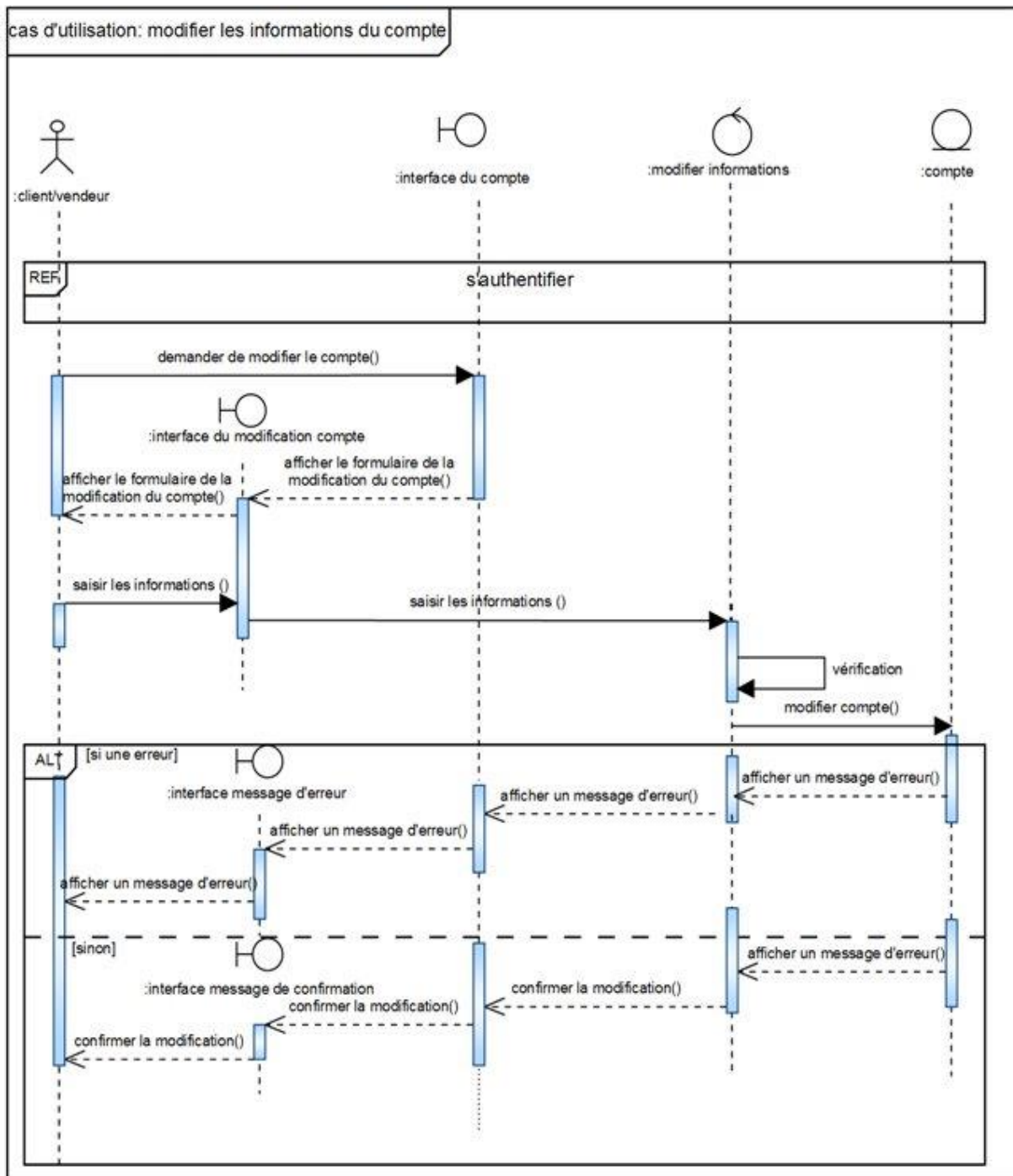


Figure 18 : Le diagramme d'interaction de cas d'utilisation

« Modifier les information du compte »

➤ Diagramme d'interaction du cas «ajouter article»

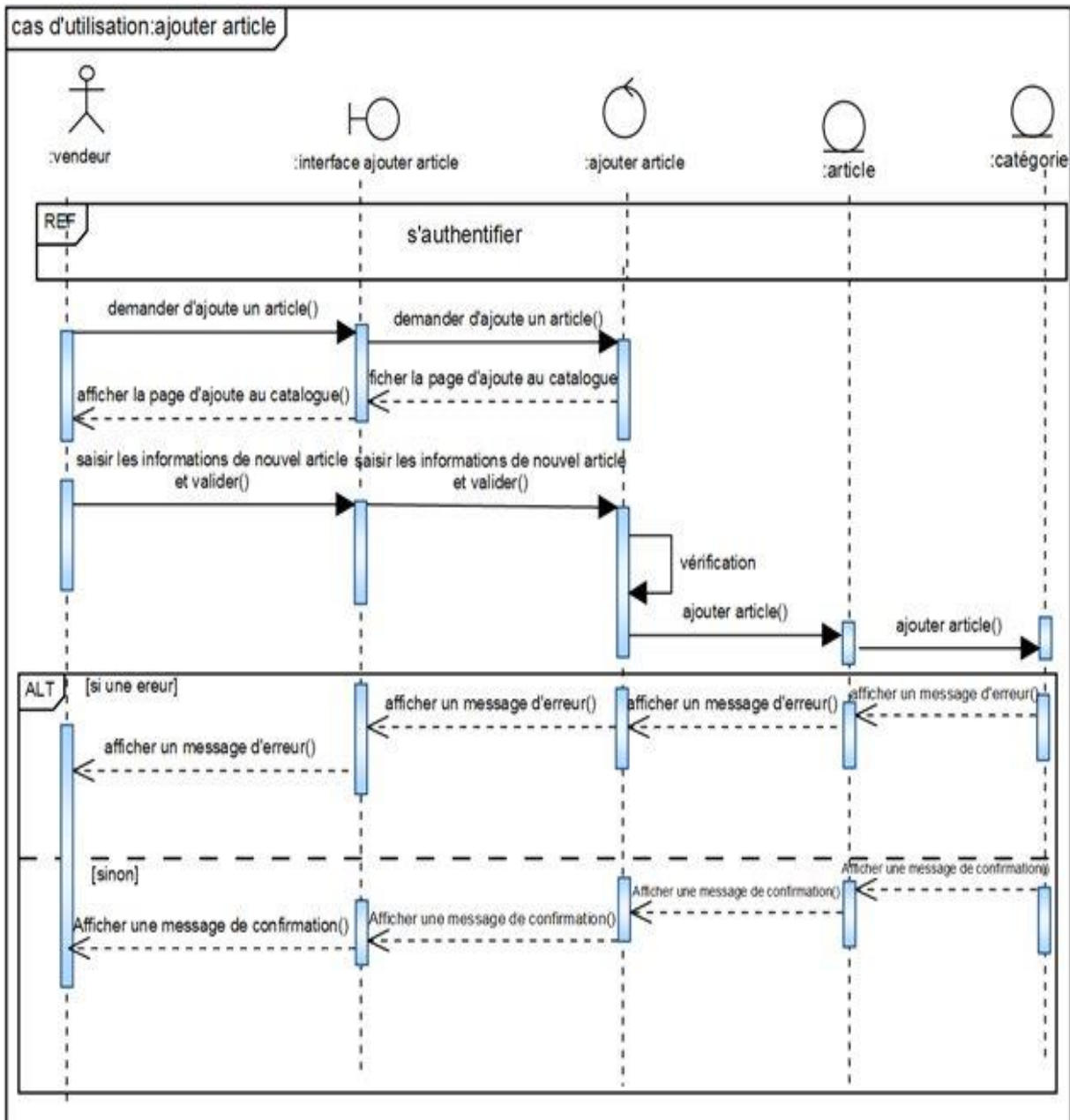


Figure 19 : Le diagramme d'interaction de cas d'utilisation « ajouter article »

➤ Diagramme d'interaction du cas «modifier articles»

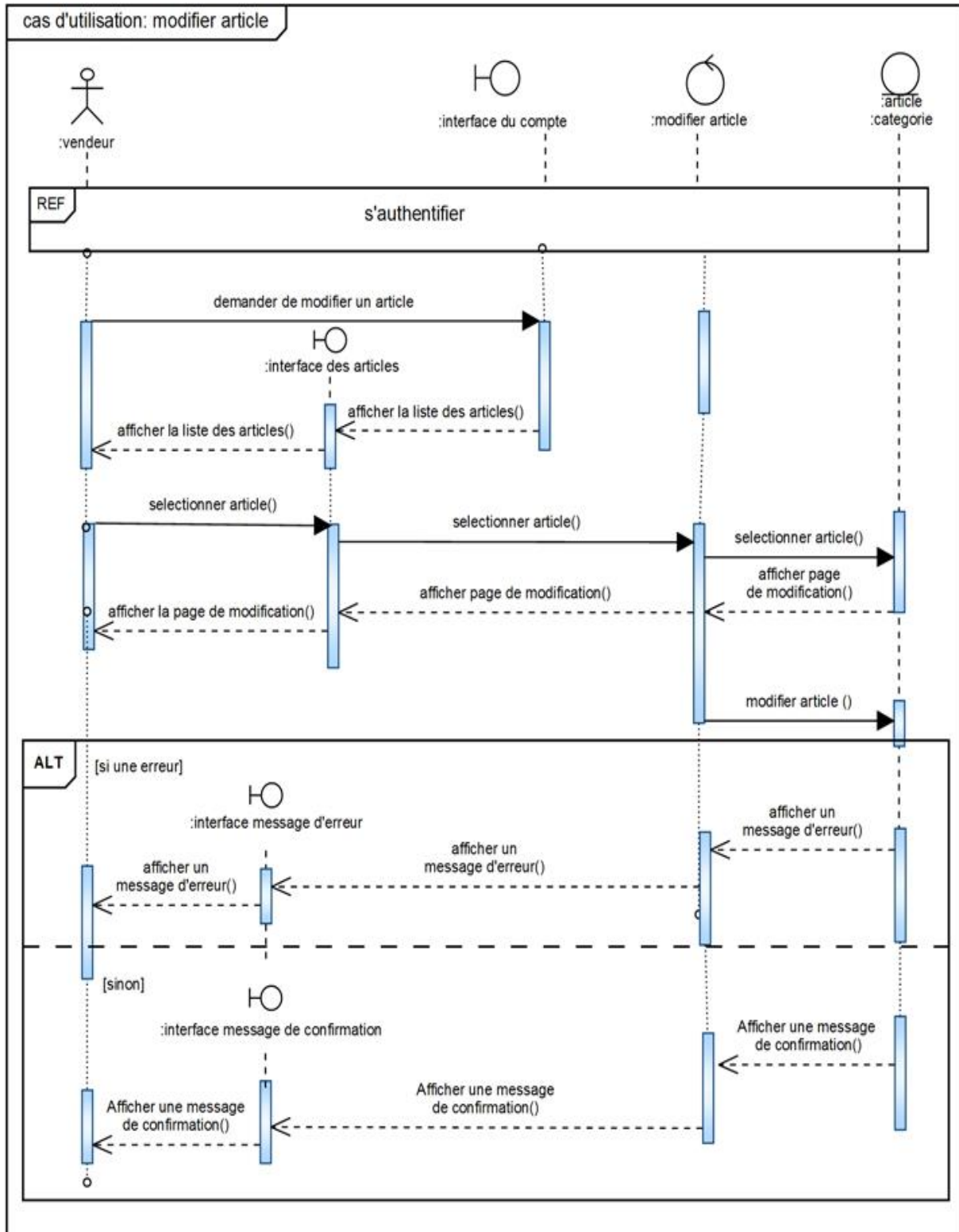


Figure 20 : Le diagramme d'interaction de cas d'utilisation « modifier article »

➤ Diagramme d'interaction du cas «supprimer article»

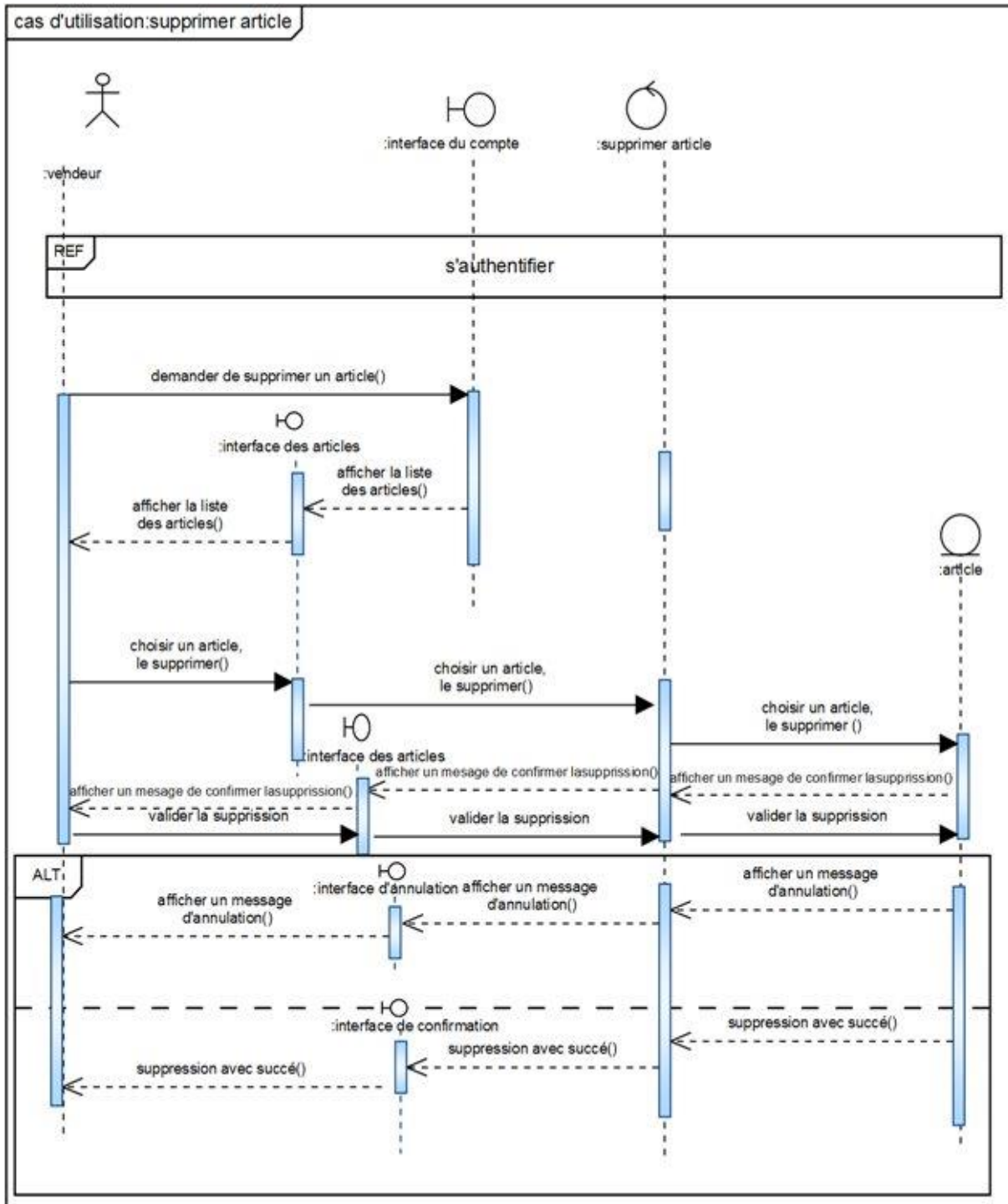


Figure 21 : Le diagramme d'interaction de cas d'utilisation « supprimer article »

➤ Diagramme d'interaction du cas «traiter commande»

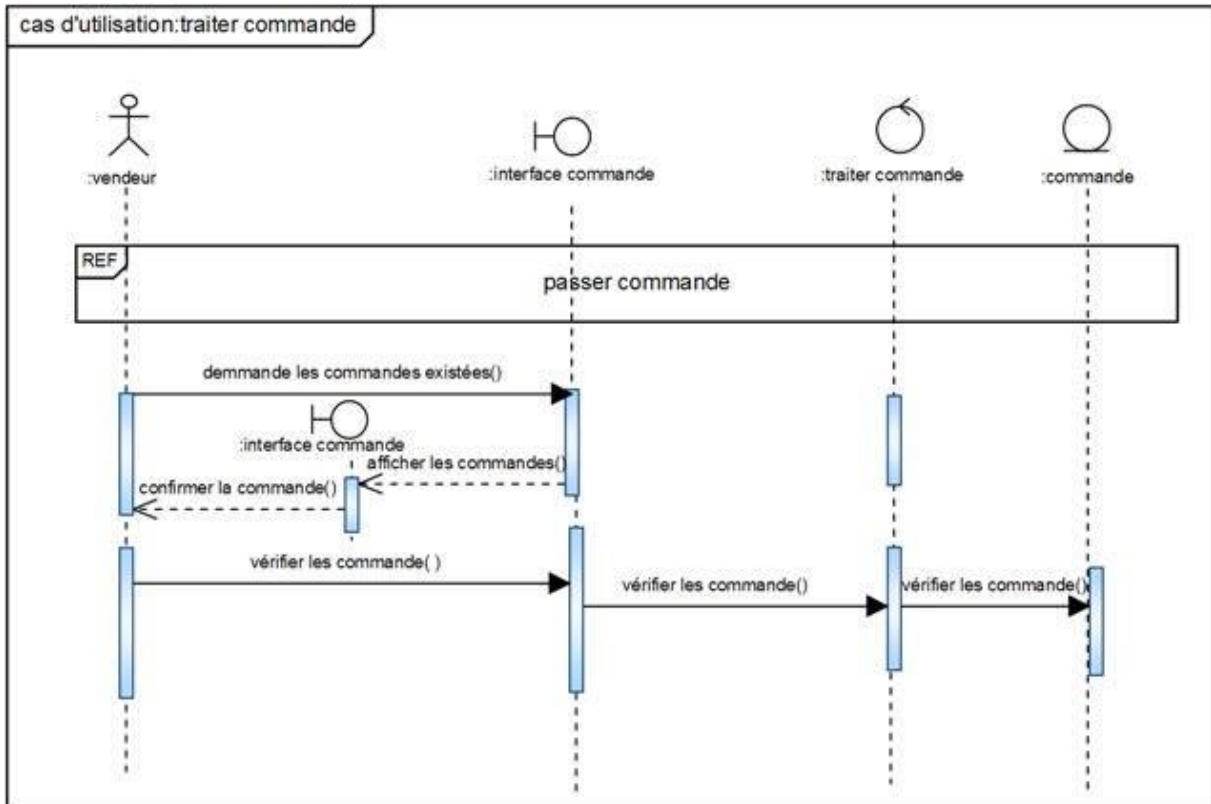


Figure 22 : Le diagramme d'interaction de cas d'utilisation « traiter commande »



➤ Diagramme d'interaction du cas «ajouter compte»

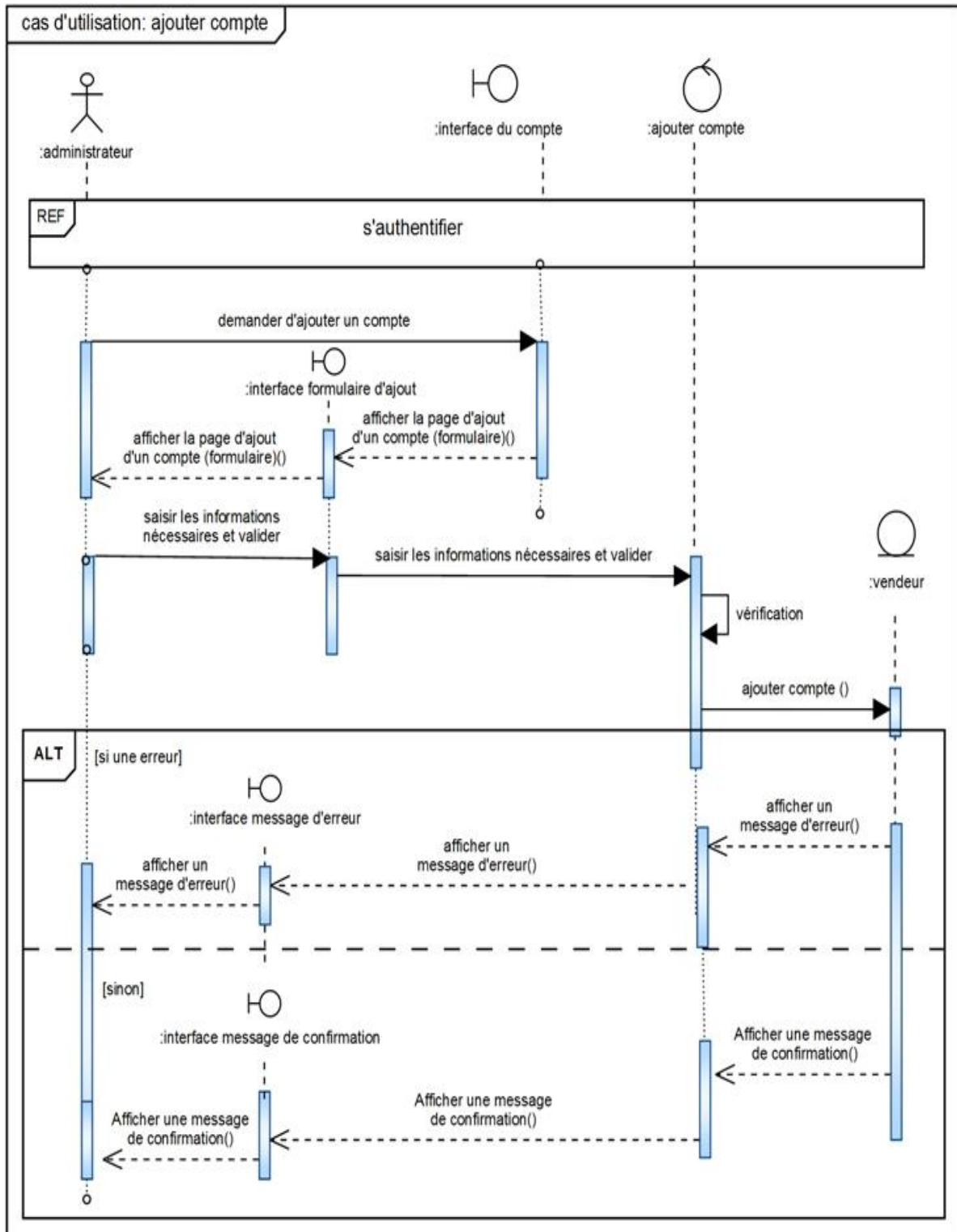


Figure 23 : Le diagramme d'interaction de cas d'utilisation « ajouter compte »

➤ Diagramme d'interaction du cas «bloquer compte»

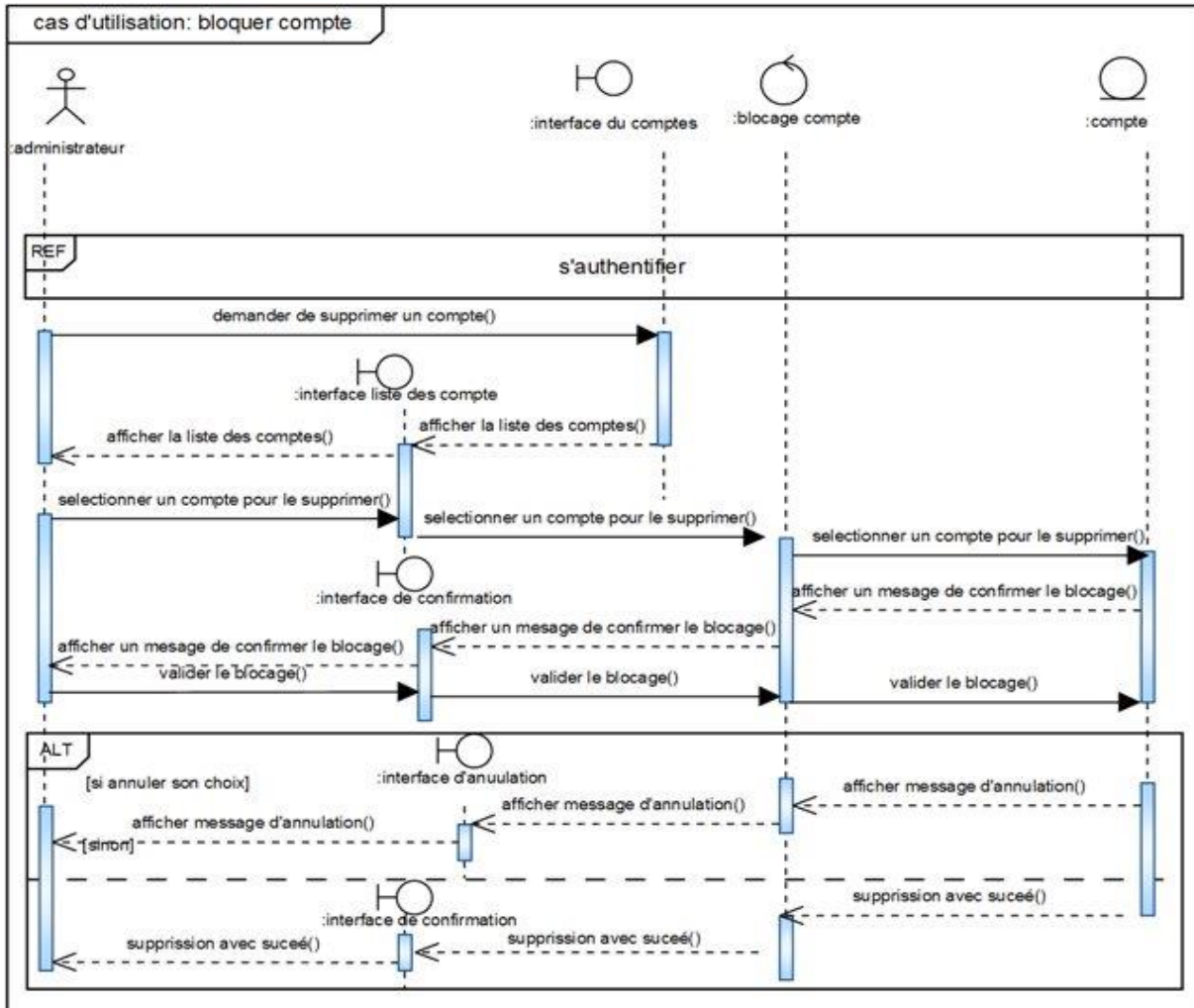


Figure 24 : Le diagramme d'interaction de cas d'utilisation «bloquer compte»

➤ Diagramme d'interaction du cas «statistique»

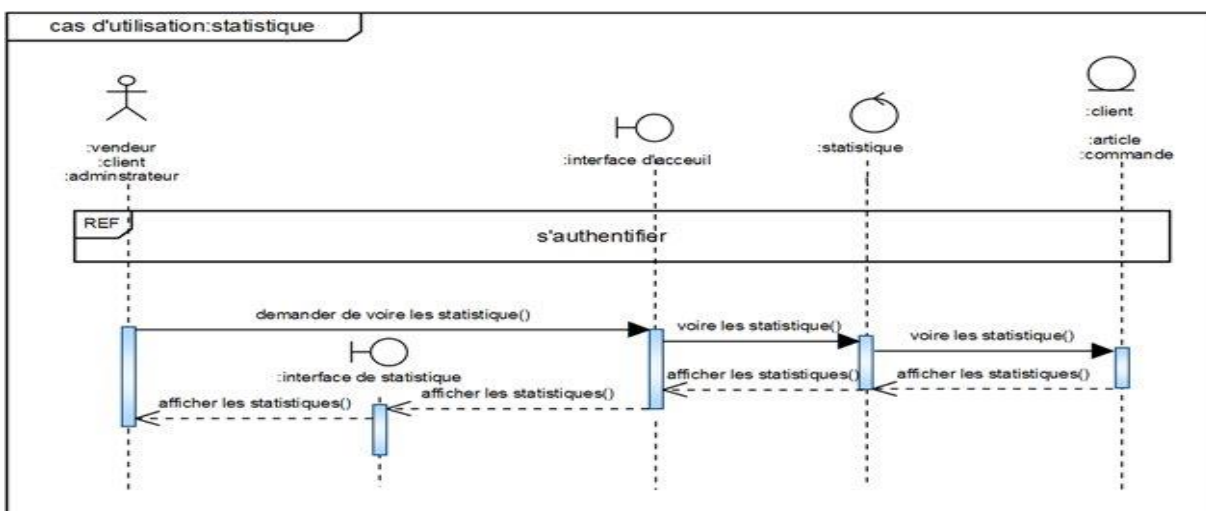


Figure 25 : Le diagramme d'interaction de cas d'utilisation «statistique»

3.2 ajouter les opérations

Dans les deux figures qui suivent nous allons ajouter les opérations pour chaque catégorie.

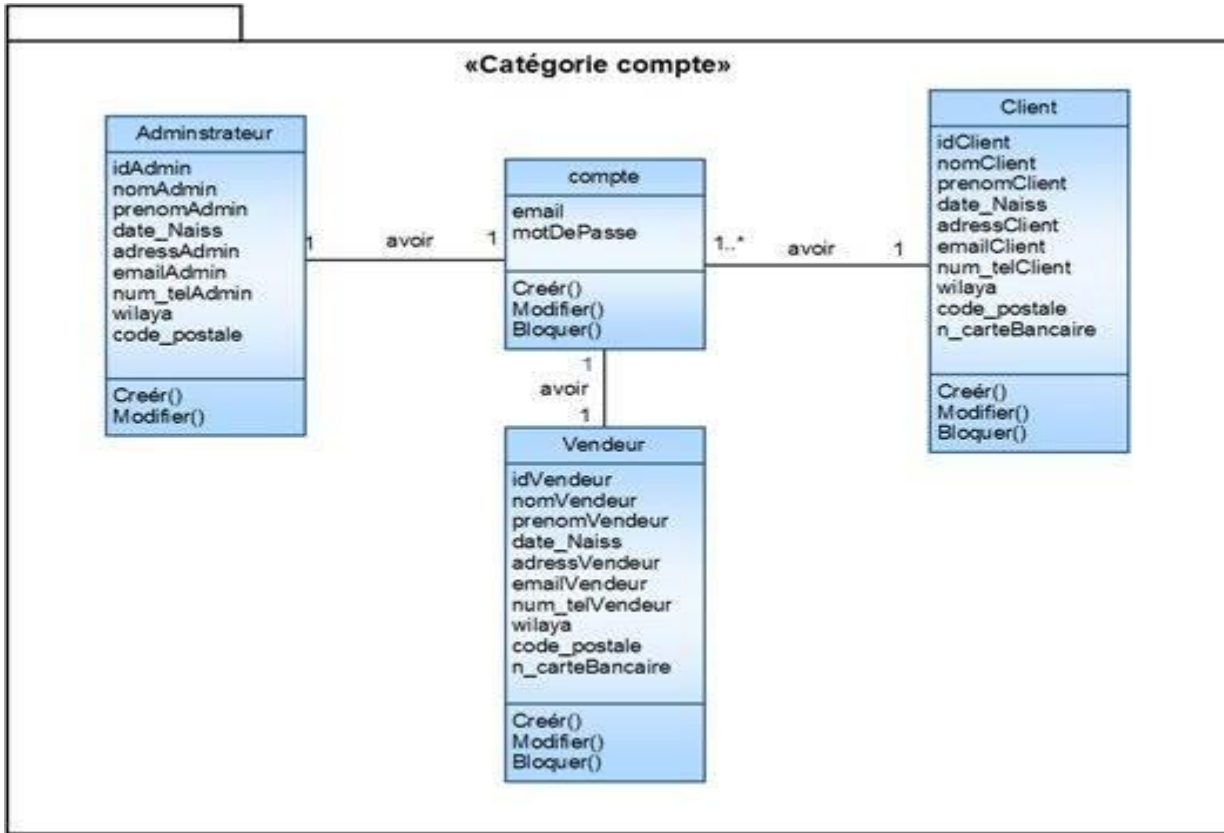


Figure 26 : Ajouter les opérations pour la catégorie compte

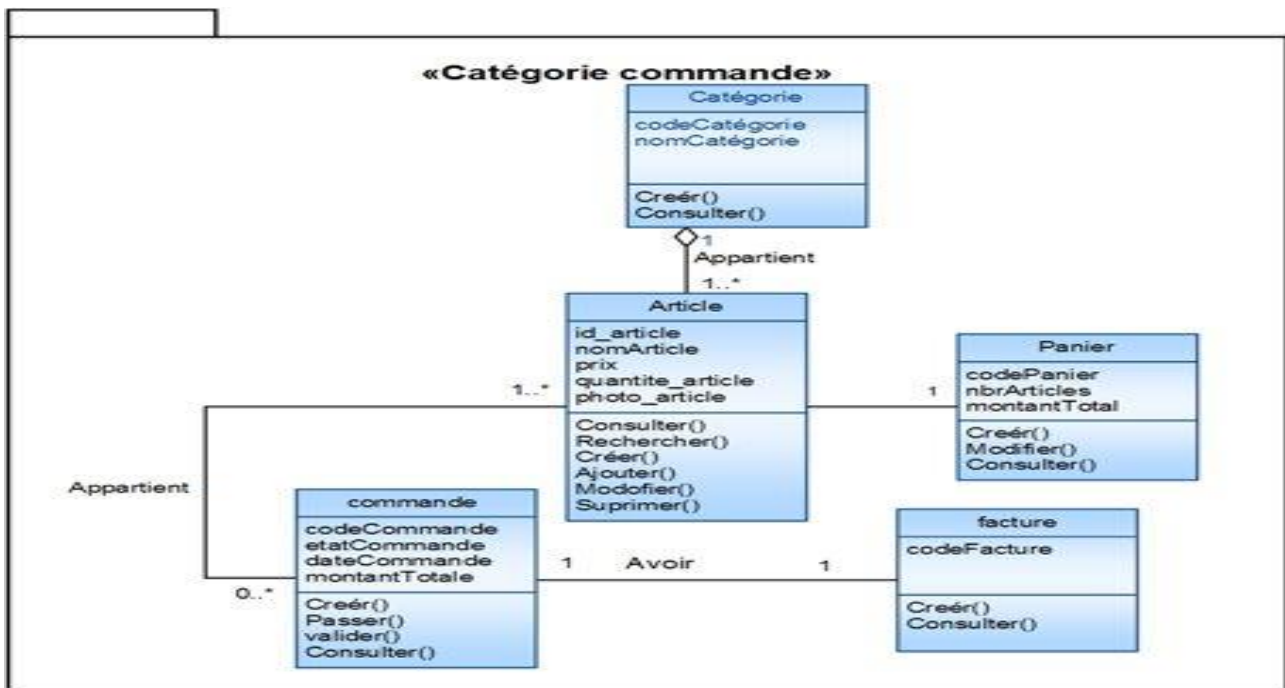


Figure 27 : ajouter les opérations pour la catégorie commande

### Conclusion

Dans ce chapitre nous avons présenté l'étape d'analyse, Ce chapitre nous a permis de passer de la structuration fonctionnelle à la structuration objet, Nous allons décrire dans le chapitre suivant la conception de notre système.

# Chapitre 06

---

## Conception

## Introduction

Dans ce chapitre, nous entamons la phase de la conception qui contient trois parties : la conception générique, préliminaire et détaillée. Nous expliquons brièvement les notions de la conception générique et préliminaire, ensuite dans la partie de conception détaillée nous élaborons le diagramme de classe détaillée et le model relationnel.

La figure suivante illustre la phase de conception.

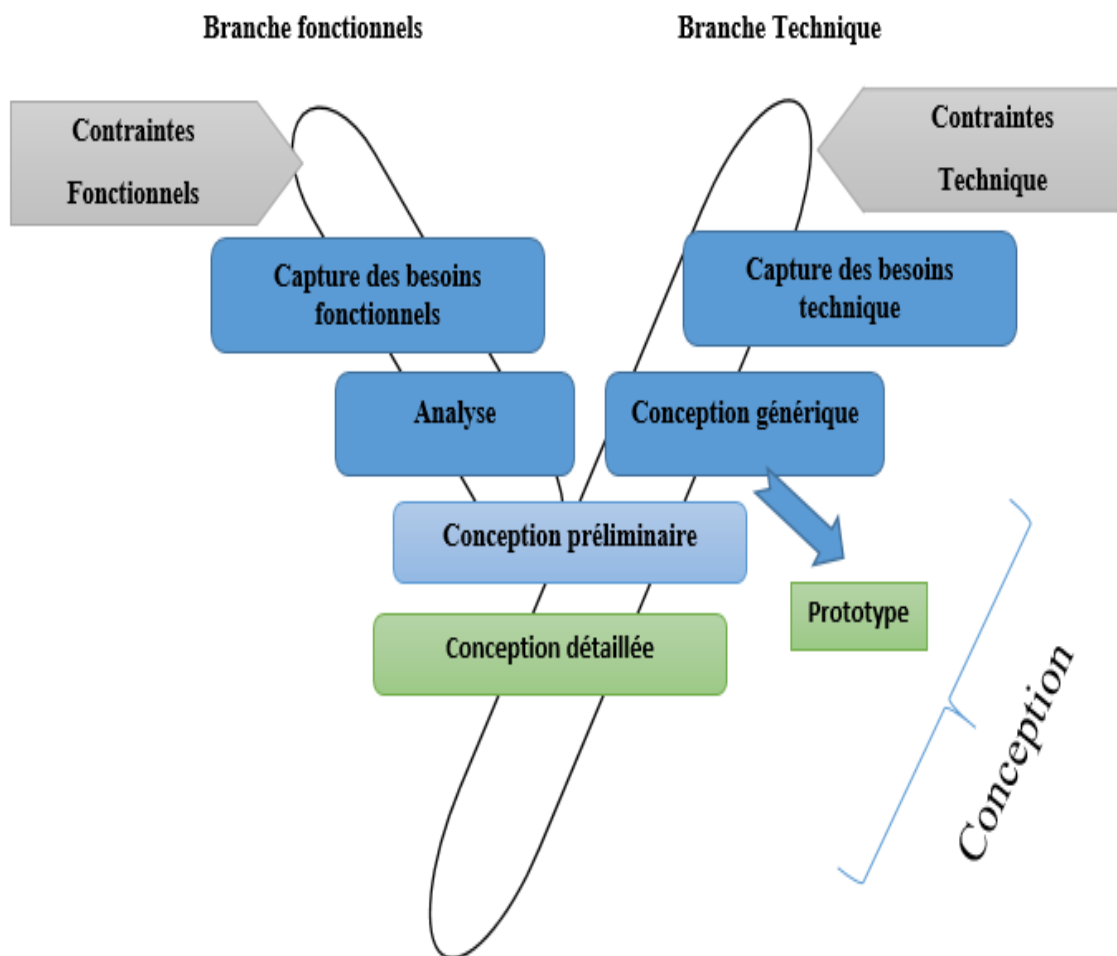


Figure 01 : Phase de conception.

### 1. Conception générique

La conception générique consiste à développer la solution qui répond aux spécifications techniques. Cette conception est qualifiée de générique car elle est entièrement indépendante des aspects fonctionnels spécifiés en branche gauche. [1]

### **1.1 Notion de classes**

L'intégrité de conception s'exprime sous la forme d'un ensemble de classes techniques que les concepteurs du projet vont par la suite réutiliser pour développer les différentes composantes fonctionnelles du système. [1]

### **1.2. Définition d'un framework technique**

Un framework est un réseau de classes qui collaborent à la réalisation d'une responsabilité qui dépasse celle de chacune des classes qui y participent. Un framework technique ne concerne que les responsabilités de la branche droite du processus. [1]

### **1.3. Interface**

Une interface est un ensemble d'opérations utilisé pour spécifier le service d'une classe ou d'un composant. [1]

### **1.4. Élaboration du modèle logique de conception**

Les classes, les interfaces et les frameworks techniques représentent les briques de construction d'un modèle logique de conception générique. Les diagrammes de classes en constituent la trame centrale ; autour de celle-ci viennent se greffer différents diagrammes dynamiques en complément de l'étude du fonctionnement de la structure.

La modélisation des classes de conception avec UML ne nécessite pas de reproduire exactement la structure du code qui doit être développé à terme. La conception est avant tout un travail de réflexion et de communication. Il s'agit donc de s'appuyer sur un ensemble de schémas suffisamment précis pour exploiter les possibilités techniques d'une solution et d'en explorer les avantages et inconvénients. [1]

#### **1.4.2 Organisation des frameworks techniques**

L'organisation du modèle logique reprend les couches logicielles. A chaque couche correspond un framework technique, en partie abstrait, qui définit des interfaces génériques de réalisation des responsabilités logicielles. [1]

La figure 01 représente L'organisation du framework technique de notre système, l'organisation du modèle logique reprend les couches logicielles ou chaque couche correspond un framework technique.

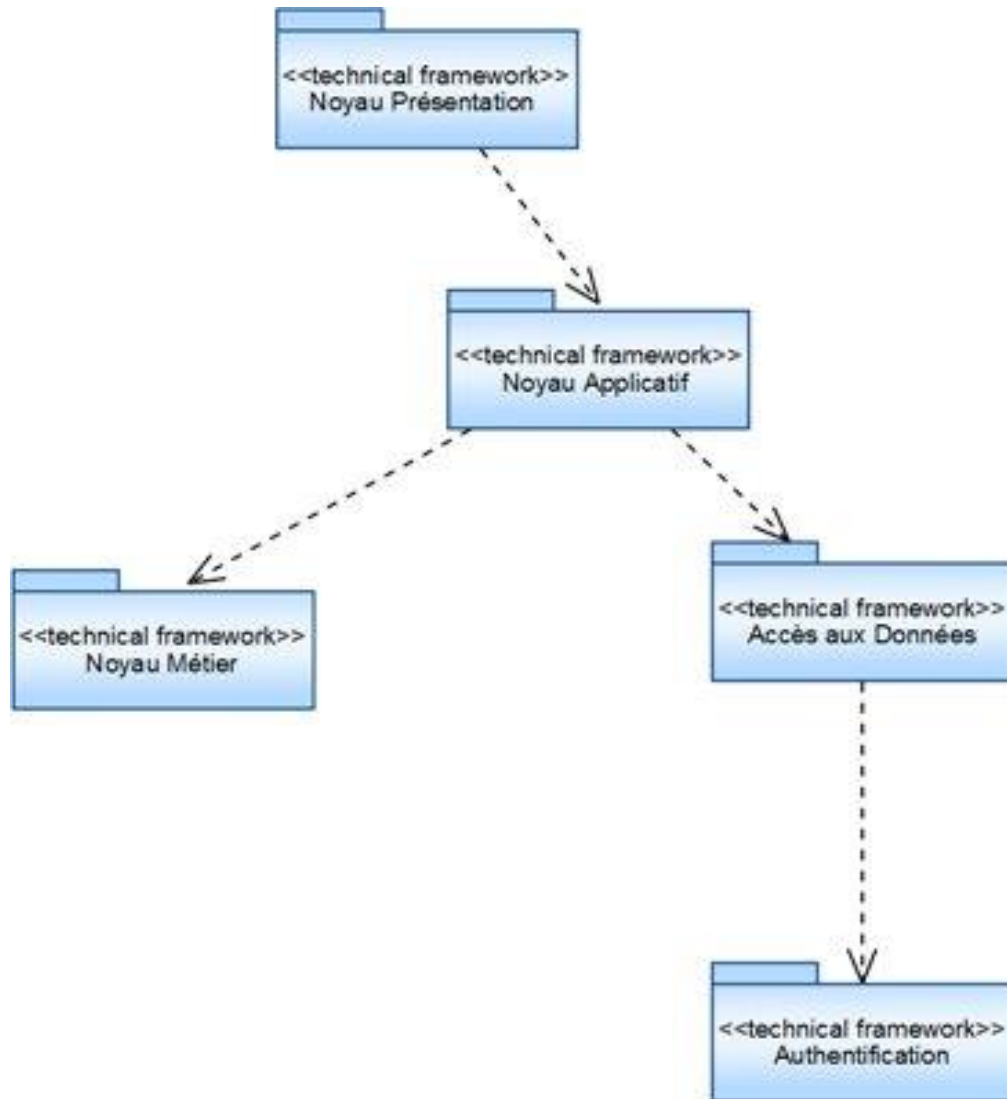


Figure 02 : Organisation du modèle logique

- **le noyau présentation** : définit les classes, les interfaces et les mécanismes de base pour réaliser l’affichage d’objets.
- **le noyau applicatif** : définit ces mêmes éléments pour rafraîchir les vues, charger les modèles de fonctionnement et contrôler les commandes d’une application.
- **le noyau métier** : définit les éléments permettant d’identifier les objets métier et de définir leurs attributs caractéristiques.
- **le noyau d’accès aux données** : définit les mécanismes de chargement, de sauvegarde, de mise à jour et de recherche des objets persistants. [1]



## **2. Conception préliminaire**

La conception préliminaire est certainement l'étape la plus délicate du processus 2TUP car elle en représente le cœur. C'est en effet à cette occasion que s'effectue la fusion des études fonctionnelles et techniques. En conséquence, plusieurs activités doivent coexister.

Il convient de :

- passer de l'analyse objet à la conception,
- intégrer les fonctions métier et applicatives du système dans l'architecture technique,
- adapter la conception générique aux spécifications fournies par l'analyse. [1]

### **2.1 Développement du modèle de déploiement**

#### **2.1.1. Poste de travail**

Le poste de travail représente un ou plusieurs acteurs pouvant être localisés sur une machine d'un type particulier et remplissant une fonction identifiée dans l'entreprise. Le poste de travail ne représente pas forcément une machine physique, mais peut consister en plusieurs machines, à condition qu'elles donnent lieu au même type de déploiement. [1]

Dans cette activité on va donner le schéma suivant qui représente son digramme de déploiement. Ce diagramme exprime la répartition physique des fonctions métier du système et permet de justifier la localisation des bases de données et des environnements de travail.

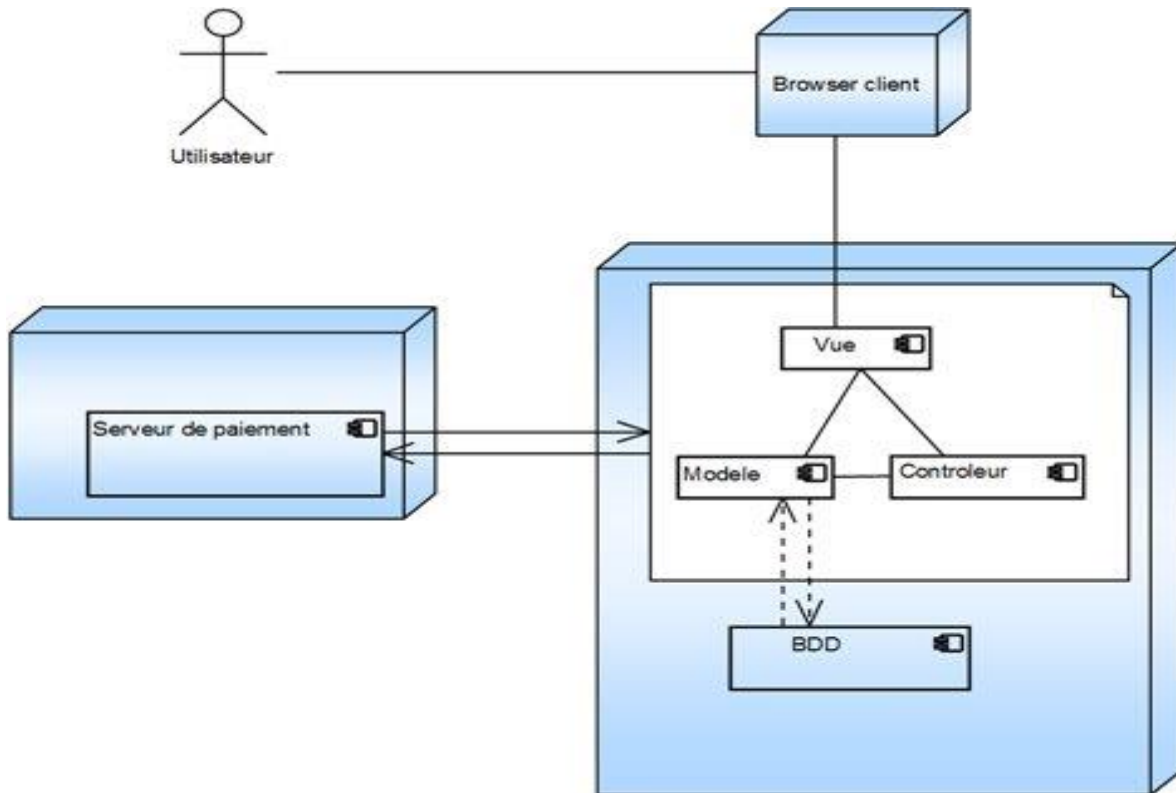


Figure 03 : Le modèle de déploiement

### 3. Conception détaillée

La conception détaillée est une activité qui s’inscrit dans l’organisation définie par la conception préliminaire. Le modèle logique y est particulièrement important dans la mesure où c’est en conception détaillée que l’on génère le plus gros volume d’informations. Il est ainsi possible de confier les catégories à des personnes différentes, qui pourront travailler indépendamment les unes des autres. [1]

#### 3.1. Dictionnaire des données

##### 3.1.1. Les classes et les attributs

Classe	Code	Désignation	type	taille
Client	-idClient	-Identifiant client	Int	7
	-nomClient	-Nom client	String	30
	-prenomClient	-Prénom client	String	30
	-date_Naiss:date	-Date de naissance client	Date	20
	-adressClient	-Adresse client	String	100
	emailClient	-Email client	String	40
	num_telClient	-Numéro de téléphone du client	Int	15

	-wilaya -code_Postale -n_carteBancaire	-Wilaya client -Code postale de la wilaya -Numéro de la carte bancaire	String Int int	30 10 20
Vendeur	-idVendeur -nomVendeur -prenomVendeur -date_Naiss:date -adressVendeur -emailVendeur -num_telVendeur -wilaya -code_Postale -n_carteBancaire	-Identifiant vendeur -Nom vendeur -Prénom vendeur -Date de naissance vendeur -Adresse vendeur -Email vendeur -Numéro de téléphone du vendeur -Wilaya vendeur -Code postale de la wilaya -Numéro de la carte bancaire	Int String String Date String String Int String Int Int	7 30 30 20 100 40 15 30 10 20
Administrateur	-idAdmin -nomAdmin -prenomAdmin -date_Naiss -adressAdmin -emailAdmin -num_telAdmin -wilaya -code_Postale -n_carteBancaire	-Identifiant administrateur -Nom administrateur -Prénom administrateur -Date de naissance administrateur -Adresse administrateur -Email administrateur -Numéro de téléphone de l'administrateur -Wilaya de l'administrateur -Code postale de l'administrateur -Numéro de la carte bancaire	Int String String Date String String Int String Int Int	7 30 30 20 100 40 15 30 10 20
Compte	-email -motDePasse	-Email d'authentification -Mot de passe d'authentification	String String	40 20
Article	-id_article -nomArticle -prix -quantite_article -photo_article	-l'identifiant de l'article -le nom de l'article -le prix de l'article -la quantité des articles -la photo de l'article	Int String Int String Double	10 20 10 20 10

			Int	10
			String	20
Catégorie	-codeCatégorie -nomCatégorie	-Code de la catégorie -Nom de la catégorie	Int string	6 40
Commande	-codeCommande -etatCommande -dateCommande -montantTotale	-Code de la commande -Etat de la commande -La date de la commande -Montant total de la commande	Int String Date double	10 20 20 10
Panier	-codePanier -nbrArticle  -montantTotale	-Le code de panier -Le nombre des articles dans le panier -Le montant total de la commande	Int Int  double	10 10  10
facture	-codeFacture	Code de la facture	Int	10

Tableau 1 : Conception des classes.

## 3.1.2. Les opérations

Classe	Operations	Description
Client	créer	Créer un compte client
	modifier	Modifier les informations de son compte
	bloquer	Bloquer un compte client
Vendeur	créer	Créer un compte vendeur
	modifier	Modifier les informations de son compte
	bloquer	Bloquer un compte vendeur
Administrateur	Créer	Créer un compte administrateur
	Modifier	Modifier un compte administrateur
Compte	Créer	Créer un compte

	Modifier	Modifier les informations de son compte
	Bloquer	Bloquer un compte vendeur
Article	consulter	Consulter la disponibilité des articles
	rechercher	Rechercher des articles particuliers
	Créer	Créer des nouveaux articles
	ajouter	Ajouter des articles
	supprimer	Supprimer des articles
	modifier	Modifier des articles
Catégorie	Créer	Créer une nouvelle catégorie
	Consulter	Consulter les catégories
Commande	Créer	Créer une nouvelle commande
	passer	Passer la commande
	traiter	Traiter la commande
	consulter	Consulter la commande
Panier	Créer	Créer un nouvel panier
	Modifier	Modifier le contenu de son panier
	retirer	Supprimer des articles
Facture	Créer	Créer la facture
	Consulter	Consulter la facture

Tableau 2 : Tableau des opérations.

3.2. Diagramme de classe détaillé

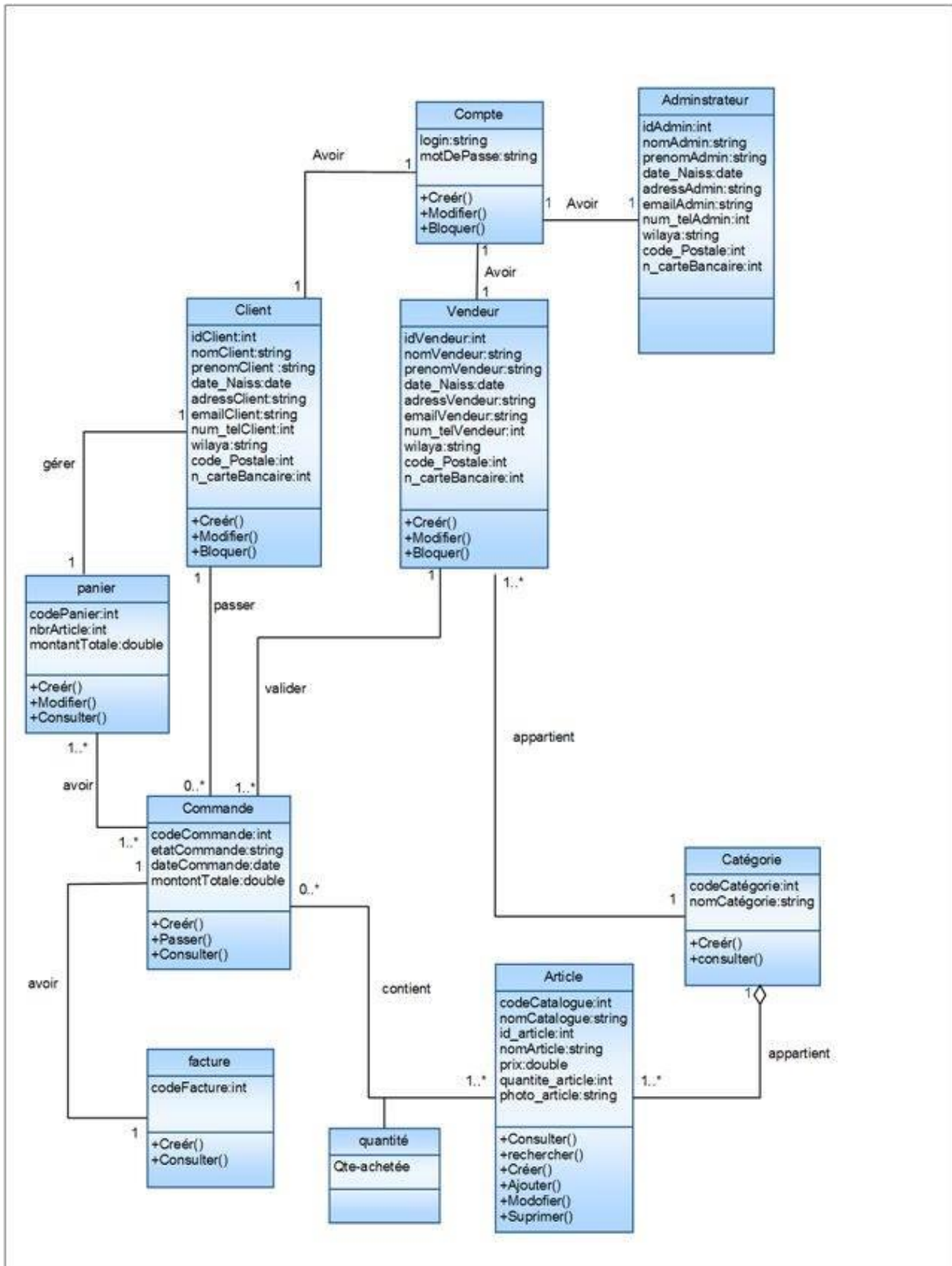


Figure 04 : diagramme de classe détaillé

### 3.3. Passage vers le model relationnel

Le modèle relationnel représente la base de données comme un ensemble de tables, sans préjuger de la façon dont les informations sont stockées dans la machine. Les tables constituent donc la structure logique du modèle relationnel. [8]

#### 3.3.1 Les règles de passages

La réalisation du modèle relationnel peut être faite à partir de la description conceptuelle. Ce passage de modèle doit respecter un certain nombre de règle, nous avons utilisé ces règles qui sont les plus simples et les plus opérationnelles :

- **Règle1** : Toute classe devient une relation, les attributs de la classe deviennent des attributs de la relation .Si la classe possède un identifiant il devient la clé primaire de la relation, sinon, il faut ajouter une clé primaire arbitraire.
- **Règle2** : pour représenter une association 1 vers 1 (1...1) entre deux relations, la clé primaire de l'une des relations doit figurer comme clé étrangère dans l'autre relation.
- **Règle3** : pour représenter une association 1 vers plusieurs (1...\*), on procède comme une association 1 vers 1, excepté que c'est forcément la relation du côté plusieurs qui reçoit comme clé étrangère la clé primaire de la relation du côté 1.
- **Règle4** : pour représenter une association plusieurs vers plusieurs (\*...\*), il faut introduire une nouvelle relation dont les attributs sont les clés primaires des relations en association, et dont la clé primaire est la concaténation de ces deux attributs. Si l'association possède des attributs, ils deviennent des attributs de la relation correspondante.
- **Règle5** : cas de héritage, transformer chaque sous classe en une relation, la clé primaire de la super classe devient clé primaire de chaque sous classe.
- **Règle6** : cas de composition, la clé primaire de la classe composée devient clé étrangère de la classe composant.
- **Règle7** : cas d'agrégation, le même principe que la Règle 3. [7]

### 3.3.2. Les tables de la base de données

-**Client** (+idClient, nomClient, prenomClient, date\_Naiss, adressClient, emailClient, num\_telClient, wilaya, code\_Postale, n\_carteBancaire, #**idCompte**).

-**Administrateur** (+idAdmin, nomAdmin, prenomAdmin, date\_Naiss, adressAdmin, emailAdmin, num\_telAdmin, wilaya, code\_Postale, n\_carteBancaire, #**idCompte**).

-**Vendeur** (+idVendeur, nomVendeur, prenomVendeur, date\_Naiss, adressVendeur, emailVendeur, num\_telVendeur, wilaya, code\_Postale, n\_carteBancaire, #**idCompte**).

-**Compte** (+idCompte, login, motDePasse).

-**Panier** (+codePanier, nbrArticle, montantTotale, #**idClient**).

-**Catégorie** (+codeCatégorie, nom Catégorie).

-**Article** (+id article, codeCatalogue, nomCatalogue, nomArticle, prix, quantite\_article, photo\_article, #**codeCatégorie**).

-**Commande** (+codeCommande, etatCommande, dateCommande, montontTotal, #**idClient** )

-**Facture** (+codeFacture,# codeCommande).

-**CommanderArticle** (# codeCommande , id article, Qte\_achetée).

### Conclusion :

Au cours de ce chapitre, nous avons présenté les trois étapes : la conception générique, la conception détaillée et la conception préliminaire pour passer à la conception de la base de données.



# Chapitre 07

---

## Implémentation

## Introduction

Ce chapitre est consacré à la réalisation et la mise en œuvre de notre application, nous allons présenter les outils de développement adopté et montrons quelques interfaces principales de notre application.

## 1. Langages et outils de développement

### 1.1. Le Framework Laravel



C'est un framework PHP open sources basé sur l'architecture MVC. Sortie en juin 2011, il est actuellement l'un des frameworks PHP les plus populaires grâce à sa documentation bien fournie. Son plus gros point fort est sa simplicité, le code que l'on écrit avec Laravel est toujours simple, lisible, concis et agréable à lire. On y retrouve les standards de programmation ainsi que la possibilité d'ajouter facilement des extensions. [15]

### 1.2 Le serveur XAMPP



XAMPP est un ensemble de logiciels qui permet de mettre en place facilement un serveur Web confidentiel, un serveur FTP et un serveur de messagerie électronique. Simple d'utilisation, il est au porté d'un grand nombre de personnes puisqu'il ne demande aucune connaissance particulière. [10]

- ✓ Xampp contient principalement les utilitaires suivant :
  - serveur Apache comme serveur d'application.
  - MYSQL comme serveur de base de données.
  - PHP pour gérer les contenus dynamiques.

### 1.2.1 Apache



Le logiciel Apache est un serveur HTTP en Open Source utilisé principalement sur les hébergements Internet en Linux, Unix ou OS X. C'est actuellement le plus utilisé sur le WEB. Différentes fonctionnalités sont implantées comme la possibilité d'utiliser un seul serveur Internet pour héberger plusieurs sites, l'utilisation des langages interprétés Perl, PHP et Python. [9]

### 1.2.2 PHP



Le PHP est un utilisé sur l'internet. Le terme langage informatique PHP est un acronyme récursif de "*PHP: Hypertexte Preprocessor*"

Ce langage est principalement utilisé pour produire un site web dynamique. Il est courant que ce langage soit associé à une base de données.[14]

### 1.2.3 MySQL



**MySQL** est une **base de données** relationnelle libre qui a vu le jour en 1995 et très employée sur le Web, souvent en association avec *PHP* (langage) et *Apache* (serveur web). MySQL fonctionne indifféremment sur tous les systèmes d'exploitation (*Windows, Linux, Mac OS* notamment).

Le principe d'une base de données relationnelle est d'enregistrer les informations dans des **tables**, qui représentent des regroupements de données par **sujets** (table des clients, table des fournisseurs, table des produits, par exemple). Les tables sont reliées entre elles par des **relations**. [13]

### 1.3. Les standards web

✓ **HTML** (HyperText Markup Language)

Est le langage de programmation utilisé pour créer des pages web invisible pour l'internaute, mais interprété par le navigateur. [11]

✓ **CSS** (*Cascading Style Sheets*)

Il permet de décrire la présentation des documents, il est utilisé avec HTML (XHTML) et XML. CSS est un le langage de décoration d'une page web. [5]

✓ **JavaScript**

C'est un langage de programmation de scripts qui est exécuté principalement par le navigateur Web sur les machines clientes. JavaScript permet de donner du dynamisme aux pages Web. [5]

✓ **Bootstrap**

Est un framework utilise les principaux langages de développement web (HTML, CSS & JavaScripts). Il s'agit d'un code qui raccourcit différentes fonctionnalités, permettant au développeur de gagner du temps et de réaliser simplement des codes complexes et tout en réduisant la quantité de caractères requis, et donc le poids du site web. [12]

## 2. Description de l'application

Dans cette partie nous allons présenter les interfaces principales de notre application.

### ➤ L'interface d'accueil

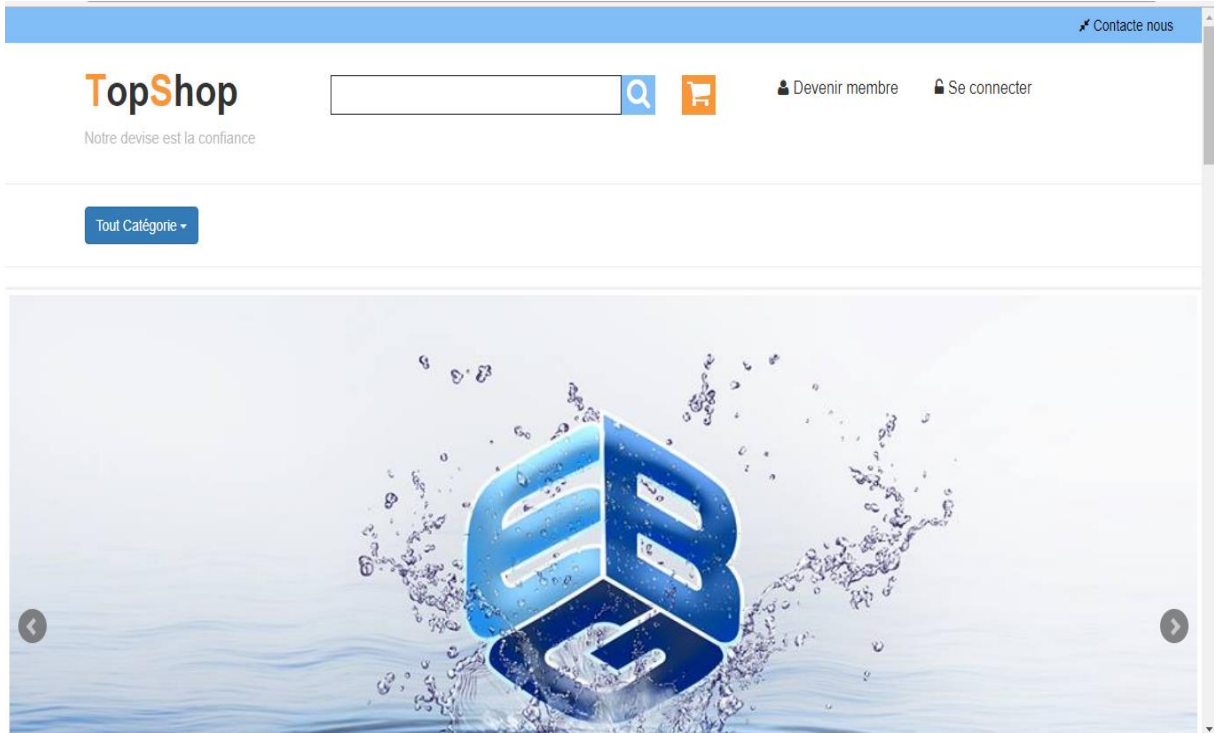


Figure 01 : Fenêtre d'accueil

### ➤ L'interface d'inscription

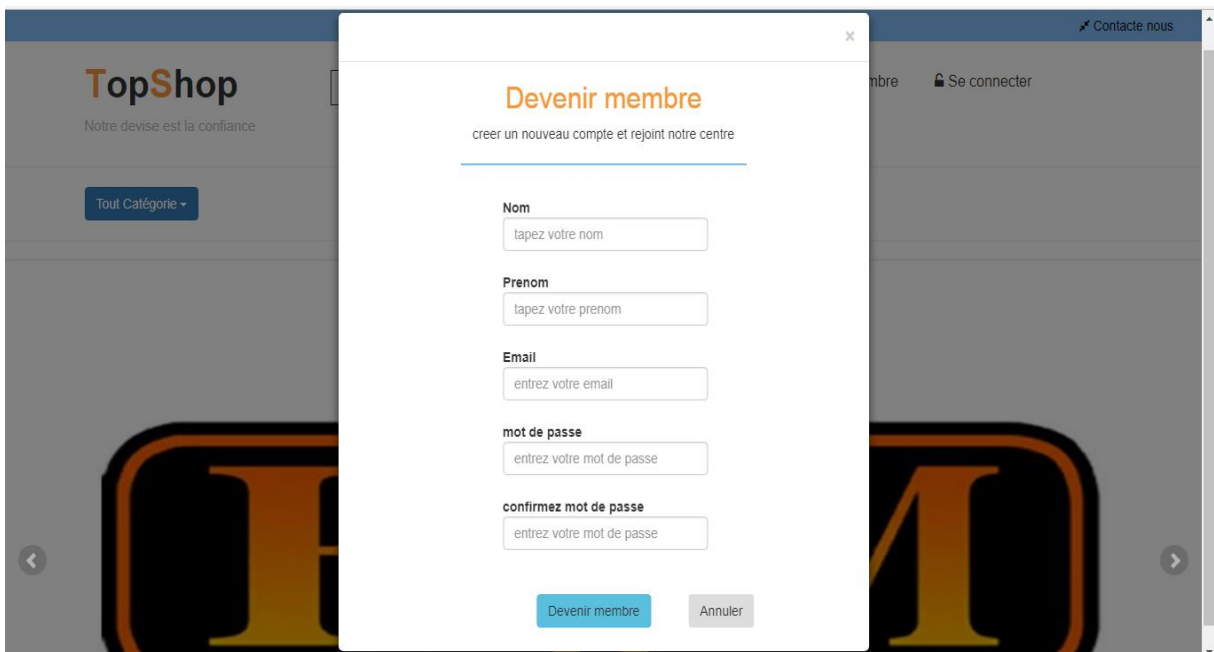


Figure 02 : Fenêtre d'inscription

➤ L'interface d'authentification

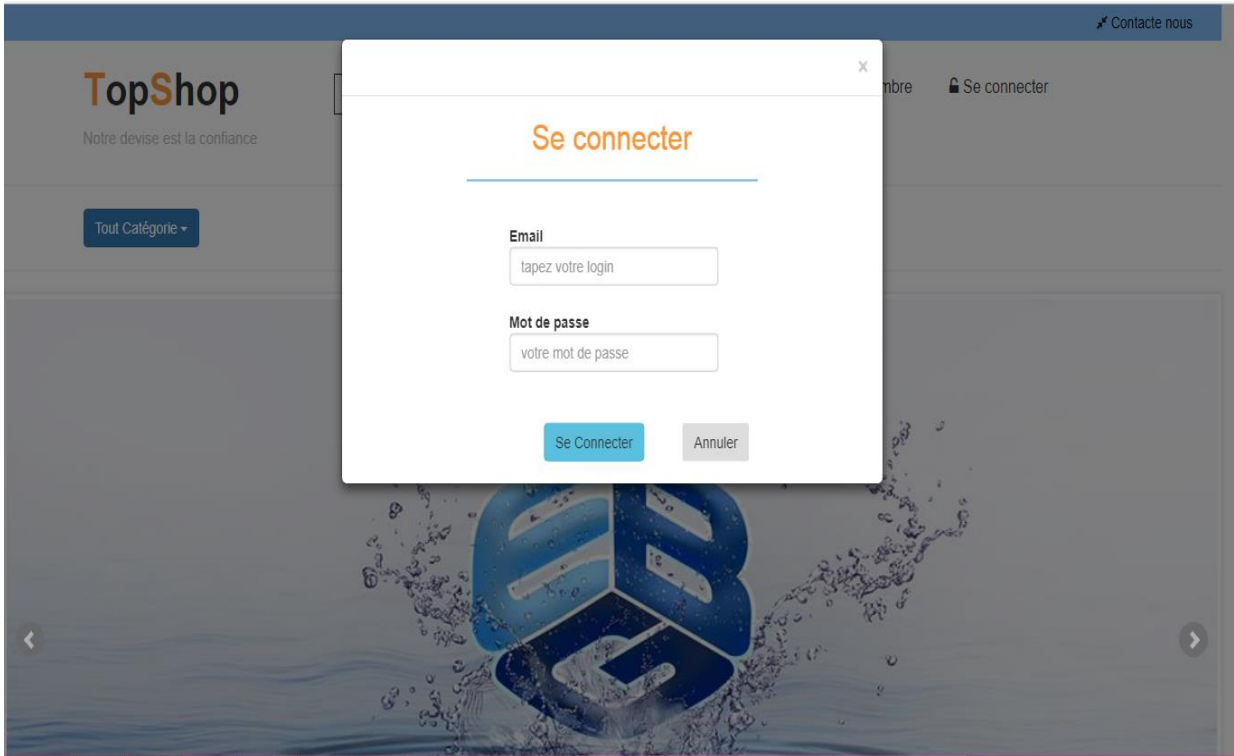


Figure 03 : Fenêtre d'authentification

➤ L'interface panier

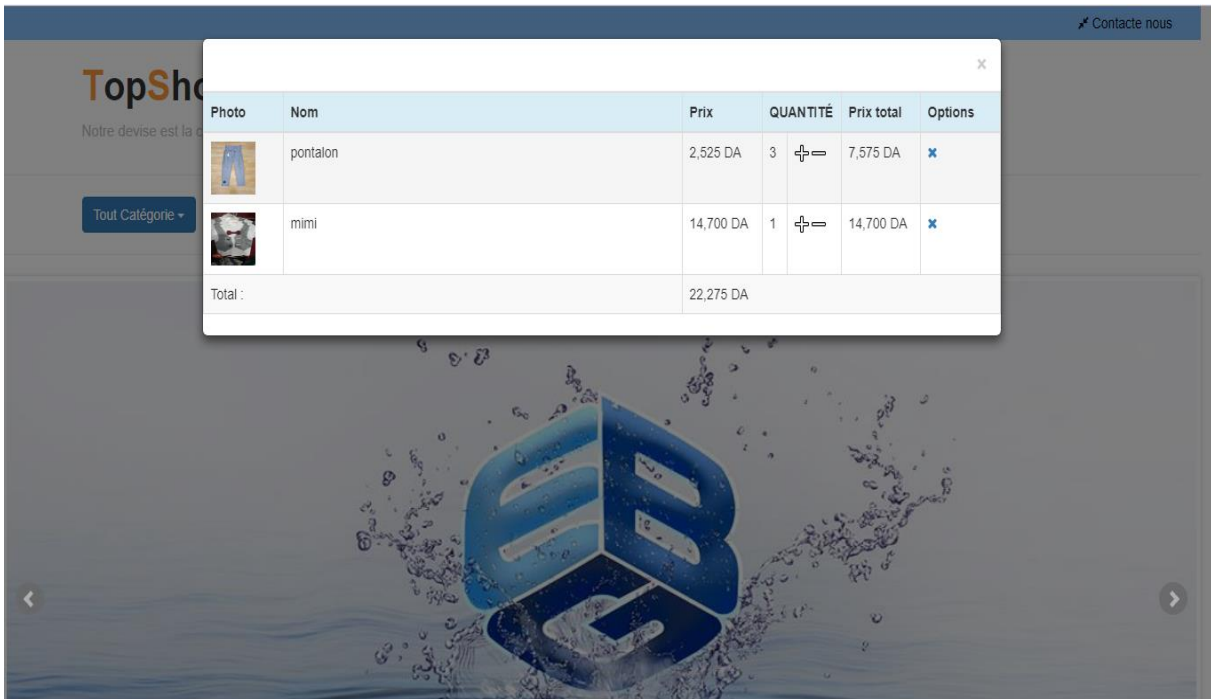


Figure 04 : Fenêtre panier

➤ L'interface Gérer la liste des articles

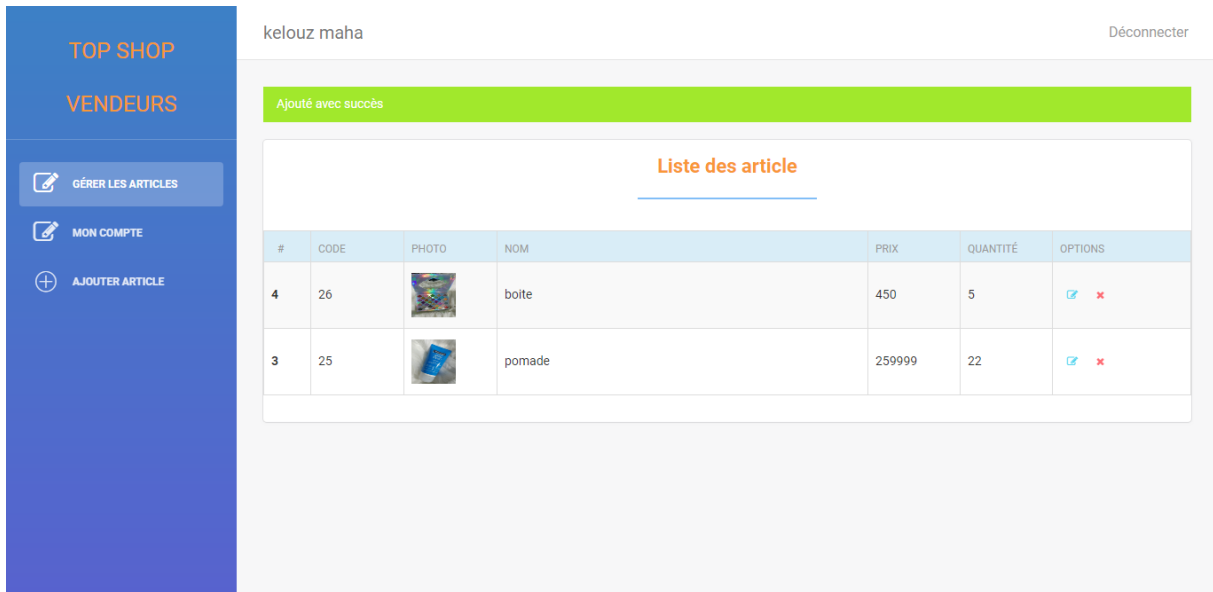


Figure 05 : Fenêtre de la gestion des articles.

➤ L'interface d'ajout d'un article

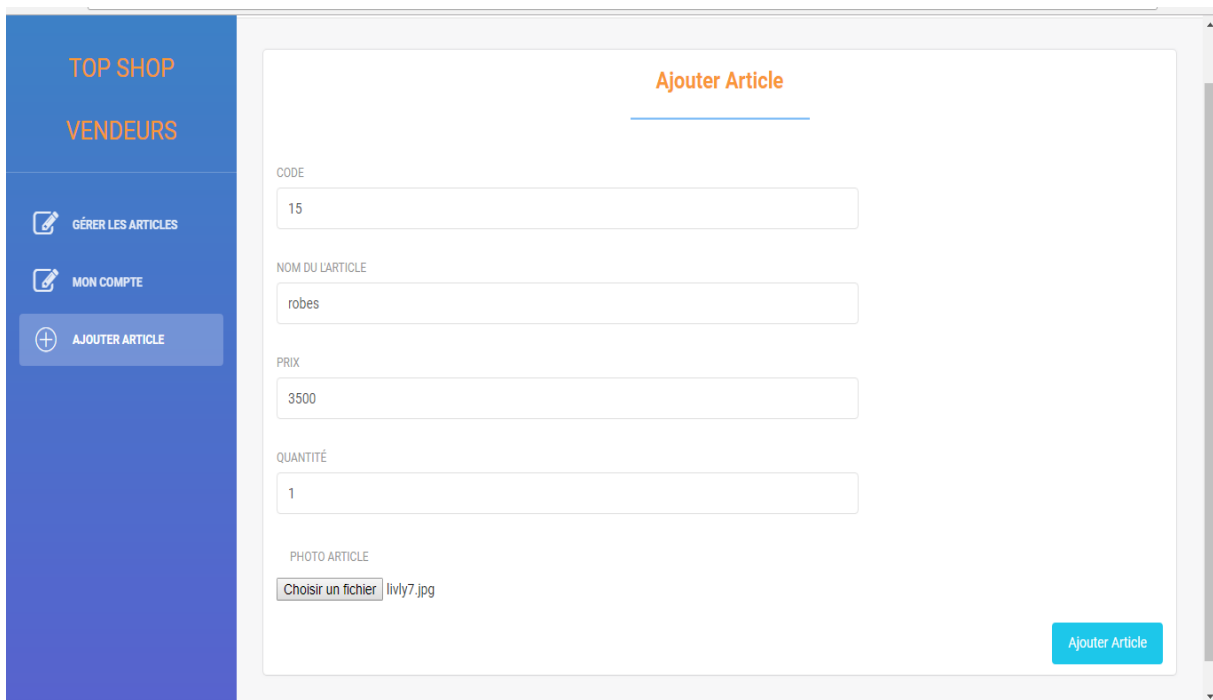
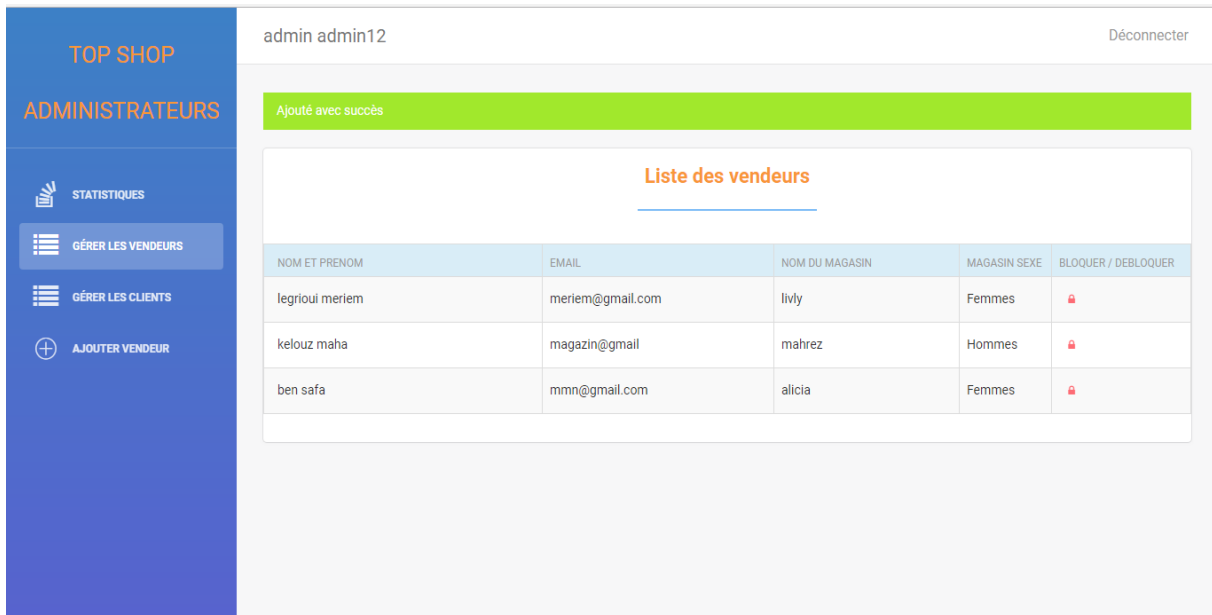


Figure 06 : Fenêtre d'ajout d'un article

➤ L'interface Gérer la liste des comptes



The screenshot shows a web application interface for managing accounts. On the left is a blue sidebar with navigation options: 'TOP SHOP', 'ADMINISTRATEURS', 'STATISTIQUES', 'GÉRER LES VENDEURS' (highlighted), 'GÉRER LES CLIENTS', and 'AJOUTER VENDEUR'. The main content area shows a user logged in as 'admin admin12' with a 'Déconnecter' link. A green banner indicates 'Ajouté avec succès'. Below is a table titled 'Liste des vendeurs' with the following data:

NOM ET PRENOM	EMAIL	NOM DU MAGASIN	MAGASIN SEXE	BLOQUER / DEBLOQUER
legrioui meriem	meriem@gmail.com	livly	Femmes	
kelouz maha	magazin@gmail	mahrez	Hommes	
ben safa	mmn@gmail.com	alicia	Femmes	

Figure 07 : Fenêtre de la gestion des comptes

## Conclusion :

Dans ce chapitre nous avons présenté la réalisation et la mise en œuvre de notre application web, le langage de programmation et les outils de développement, et enfin nous avons montré les principales interfaces et fenêtres de l'application.



# Conclusion générale

---

## Conclusion générale

---

Notre travail a été réalisé dans le cadre de projets de fin d'études. L'objectif de ce travail est la conception d'une application web d'achat en ligne dans le centre commercial « Beldi », cette application web va simplifier le processus d'achat pour les clients comme pour les vendeurs, et améliorer les revenus du centre.

Pour la conception de notre application, nous avons choisi un processus nommé « 2TUP » qui signifie « 2 Track Unified Process », « 2 Track » signifie littéralement que le processus suit deux chemins. Il s'agit des chemins « fonctionnels » et des chemins « techniques ».

Pour la réalisation, nous avons utilisé le Framework « Laravel » qui est un Framework PHP basé et sur l'architecture MVC et c'est l'un des Frameworks PHP les plus populaires.

Malgré les efforts déployés, la réalisation de l'application n'a pas été entièrement achevée, certaines fonctionnalités ne sont pas encore implémentées.

Comme dernier mot nous tenons à dire que notre parcours durant ce projet nous a retourné plus d'avantages que d'inconvénients, et une opportunité pour aborder le domaine de développement web.

# Bibliographie

## Ouvrages

- ✚ [1] Roques, P. Vallée, F. Troisième tirage 2001.UML en action de l'analyse des besoins à la conception en Java. Groupe Eyrolles.
- ✚ [2] Roques, P. 4eme édition. les Cahiers du Programmeur UML2 Modéliser une application web. Paris : Groupe Eyrolles. 43p.
- ✚ [3] Roques, P., Vallée, F. 2003.UML en action de l'analyse des besoins à la conception en Java. Deuxième édition. Groupe Eyrolles. [12.13.14.15] p 1
- ✚ [4] chavelli, M. (2016). Découvrez le framework php laravel.Paris : ÉDITIONS EYROLLES. [4.5.6] p

## Support de cour

- ✚ [5] Boukhechem, Nadir. «Développement Web Avancé» Centre universitaire de Mila, MILA, 3 mars 2016.

## Mémoires

- ✚ [6] Mahboub, O., Majdoul, A. (2012-2013). Conception Orienté Objet et Réalisation du cas Agri selon le processus 2tup. mémoire de projet de fin de 2ème année : Ecole nationale supérieure d'informatique et d'analyse des systèmes. Université mohammed v – souissi. 55pages.
- ✚ [7] BOUMAAD, S. , CHETIBI, H. (2016/2017). Conception et réalisation d'une application client/serveur pour le suivi des projets au sein de la Direction des Travaux Publics (DTP Mila). Mémoire : Sciences et Technologies de l'Information et de la Communication (STIC).Mila : centre universitaire abdlhafid bousouf, 167page.

## Site web

- ✚ [8] Chaabane, R. Le modèle de données relationnel[en ligne]. (Page consultée le 15/03/2018). < <http://www.ai.univ-paris8.fr/~lysop/bd/seance4-ModeleRel.pdf>>.

- ✚ [9] Apache Software Foundation. Matériel informatique.be[en ligne]. (Page consultée le 18/04/2018). < <http://www.materiel-informatique.be/apache.php> >.
  
- ✚ [10] Pack Logiciels Libres de l'Entreprise[en ligne]. (Page consultée le 18/04/2018). < <http://www.pack-logiciels-libres.fr/spip.php?logiciel44>>.
  
- ✚ [11] Bathelot, B. définition marketing[en ligne]. (Page consultée le 18/04/2018). < <https://www.definitions-marketing.com/definition/html/>>.
  
- ✚ [12] Agency Inside. Bootstrap Framework [en ligne]. (Page consultée le 18/04/2018).<<https://agency-inside.com/2016/06/definition-webmarketing-bootstrap/>>.
  
- ✚ [13] mosaïque informatique-formation-site web[en ligne]. (Page consultée le 18/04/2018).<<http://www.mosaique-info.fr/glossaire-web-referencement-infographie-multimedia-informatique/m-glossaire-informatique-et-multimedia/448-mysql-definition.html> >.
  
- ✚ [14] infoWebmaster[en ligne]. (Page consultée le 18/04/2018). <<http://glossaire.infowebmaster.fr/php/> >.
  
- ✚ [15] Agence web Compiègne .Le PHP oui, mais avec Laravel [en ligne]. (Page consultée le 18/12/2017). <<https://www.anthedesign.fr/developpement-web/php-avec-laravel/>>.
  
- ✚ [16] Dr.CHBLOUNE Habib. U.M.L - Méthode 2TUP. [en ligne]. (Page consultée le 18/12/2017). <<https://fr.scribd.com/doc/57496887/U-M-L-Methode-2TUP>>.
  
- ✚ [17] Entrepreneurs d'avenir. Une application web : qu'est-ce que c'est. [en ligne]. (Page consultée le 18/12/2017). <<https://fr.scribd.com/doc/57496887/U-M-L-Methode-2TUP>>.
  
- ✚ [18] Agence web Compiègne .Le PHP oui, mais avec Laravel [en ligne]. (Page consultée le 18/12/2017). <<https://www.anthedesign.fr/developpement-web/php-avec-laravel/>>.

- ✚ [19] OC. Apprenez ASP.NET MVC. [en ligne]. (Page consultée le 18/12/2017). <<https://openclassrooms.com/courses/apprendre-asp-net-mvc/le-pattern-mvc>>.
- ✚ [20] Définition marketing. Définition : E-commerce. [en ligne]. (Page consultée le 18/12/2017). <<https://www.definitions-marketing.com/definition/e-commerce/>>.
- ✚ [21] création site web. Agence webmarketing. [en ligne]. (Page consultée le 18/12/2017). <http://www.creationsitesinternet.net/site-e-commerce.html>