

[Company name]  [Company address]

**Présenté par** : Monsieur RANDRIANARIKOTO Alexandre Boris

**Membres de jury** :

* Président : Docteur RAKOTOASIMBAHOAKA Cyprien Robert
* Examinateur : Monsieur MAHATODY Thomas
* Rapporteur :
* Docteur RATIARISON Venot
* Monsieur Pierre Luc GERVAIS

*Année universitaire : 2013-2014*

**CONCEPTION ET REALISATION D’UN SITE E-COMMERCE RESPONSIVE EN UTILISANT LA TECHNOLOGIE NODEJS POUR LE COMPTE DE FLUOO**

**MEMOIRE DE FIN D’ETUDE POUR L’OBTENTION DU DIPLOME D’INGENIEUR EN INFORMATIQUE**

**Option** : Génie logiciel et Base de données

**Intitulé** :

Université de Fianarantsoa/Ecole Nationale d’Informatique Internet Révolution de l’Océan Indien

AVANT-PROPOS

Au terme de deuxième année de la formation d’ingénieur à l’Ecole Nationale d’Informatique, les élèves doivent effectuer un stage de six mois au sein d’une entreprise pour la préparation du mémoire de fin d’études en vue d’obtention du diplôme d’ingénieur en informatique.

Ce stage a pour but d’initier les élèves à s’intégrer dans le domaine du travail ainsi que d’appliquer et de compléter les connaissances acquises à l’Ecole. Un mémoire doit être rédigé par tous les élèves intéressés sur le thème proposé par les enseignants et le responsable de l’entreprise.

Nous avons effectué notre stage de fin d’études au sein de l’entreprise IRE-OI pour la raison que c’est une société experte en architectures Web et solutions Open Source. Notre tâche s’est focalisée autour du thème : « CONCEPTION ET RÉALISATION D’UNSITE E-COMMERCE RESPONSIVE EN UTILISANT LA TECHNOLOGIE NODEJS POUR LE COMPTE DE FLUOO».

REMERCIEMENTS

La présentation ainsi que le contenu de ce présent mémoire furent les fruits d’une collaboration et de coopération étroite entre plusieurs entités sans lesquelles tout cela n’aurait pas été possible. Nous tenons alors à remercier et à saluer tous ceux dont les aides et conseils nous ont été indispensables dans l’accomplissement de ce mémoire.

Nous tenons tout d’abord à remercier Monsieur Fontaine RAFAMATANANTSOA, Docteur et Directeur de l’Ecole Nationale d’Informatique (ENI) de l’Université de Fianarantsoa qui a pu nous donner son consentement d’effectuer ce stage pratique.

Par la même occasion, nous remercions Monsieur Pierre-Luc Gervais, Directeur Général de la société IRE-OI, d’avoir accepté de nous admettre comme stagiaire au sein de son entreprise et d’avoir pu accepter de nous encadrer durant notre stage.

Nous tenons également à adresser nos vifs remerciements à Docteur Cyprien Robert RAKOTOASIMBAHOAKA, Chef de Département pratique et enseignant à l’Ecole Nationale d’Informatique, pour son dévouement à la préparation ainsi qu’à l’organisation du stage.

Nous tenons également à remercier Docteur RATIARISON Venot, chef de département de la formation théorique et enseignant à l’Ecole Nationale d’Informatique, qui a bien voulu accepter de nous encadrer durant ce projet.

Sans oublier également de remercier les membres du corps Enseignant et administratif de l’Ecole Nationale d’Informatique, pour les savoirs et connaissances qu’ils nous ont transmis.

Un grand merci également aux collègues de bureau et à tous les membres de personnel de l’entreprise IRE-OI qui se reconnaîtront, pour leur accueil et la bonne collaboration.

Nos remerciements vont également à nos familles, nos amis, nos proches pour leur soutien dans toutes les situations tout au long de ce travail.

Enfin, nous remercions toutes les personnes qui nous ont aidé tout au long de la mémoire et encouragé à aller dans cette direction.

CURRICULUM VITAE

RANDRIANARIKOTO Alexandre Boris



Né le 21 Janvier 1988 à Fianarantsoa

Célibataire

Adresse: Lot 0307 P / 0040 Soanierana,

Andrainjato-Nord, Fianarantsoa

E-mail : randrianarikoto@gmail.com

GSM : 032 49 834 67 / 034 89 109 82

FORMATIONS ET DIPLOMES OBTENUS

2013 -2014 : Dernière année de formation en Master professionnel en Génie logiciel et Base de données à l’Ecole Nationale d’Informatique de l’Université de Fianarantsoa ;

Diplôme en cours : **Diplôme d’ingénieur en Informatique**.

2012 -2013 : Première année de formation en MASTER à l’Ecole Nationale d’Informatique de l’Université de Fianarantsoa.

2011 -2012 : Troisième année de formation en Licence Professionnelle à l’Ecole Nationale d’Informatique de l’Université de Fianarantsoa, option : Génie logiciel et Base de données ;

Diplôme obtenu : **Licence Professionnelle, option : Génie logiciel et Base de données ;**

Mention : **Excellent** ;

2010 -2011 : Deuxième année de formation en Licence Professionnelle à l’Ecole Nationale d’Informatique de l’Université de Fianarantsoa, option : Génie logiciel et Base de données ;

2009 -2010 : Première année de formation en Licence Professionnelle à l’Ecole Nationale d’Informatique de l’Université de Fianarantsoa, option : Génie logiciel et Base de données ;

2008 - 2009 : Classe de terminale au Lycée RAHERIVELO Ramamonjy à Fianarantsoa

Diplôme obtenu : **Baccalauréat série D**

STAGES EXPERIENCES PROFESSIONNELLES

Novembre – Janvier 20013: Stage pratique au sein de la société de distribution AQUAMAD, Région d’Antananarivo;

Thème : **Mise en place d’une application web dynamique pour la vente en ligne**

Novembre – Février 20012 : Stage pratique au sein de la Direction du Système d`Information du Ministère de l`Intérieur, Région d’Antananarivo ;

Thème :  **Mise en place d’un système d’information pour la gestion de la Carte d’Identité Nationale.**

Février 2011 : Travaux de réalisation à l`Ecole Nationale d`Informatique(ENI) Fianarantsoa ;

Thème : **Gestion bancaire de la Banque Nationale de l’Industriel-Crédit Agricole(BNI).**

Septembre 2005 : Formation en Informatique bureautique à l’ONG Fitiavana, Fianarantsoa.

*\*Grille d’évaluation : 0 : très mauvaise 1 : mauvaise 2 : moyenne 3 : bonne 4 : excellente*

APTITUDES LINGUISTIQUES

|  |  |  |  |
| --- | --- | --- | --- |
| **Langue** | **Compréhension** | **Expression orale** | **Expression écrite** |
| **Français** | 4 | 3 | 3 |
| **Anglais** | 2 | 2 | 3 |

APTITUDES INFORMATIQUES

|  |  |  |
| --- | --- | --- |
| **Discipline** |  | **Niveau** |
| Bureautique | Ms Office, Open Office | 3 |
| Système d’exploitation | Windows | 3 |
| Linux | 3 |
| Développement mobile | Android, Web Mobile | 2 |
| Modélisation et Notation | Merise, UML | 2 |
| Langage de Programmation | C/C++, JAVA, Python | 2 |
| Technologie Web | HTML, XML, JavaScript, AJAX, PHP, JSP | 3 |
| JSF, ASP | 3 |
| JQuery | 2 |
| Jelix | 2 |
| Administration de réseaux | Sous Windows | 3 |
| Sous Linux | 2 |
| Système d’Information Géographique | Base de données spatiales et mise en place d’un SIG | 2 |
| Installation, Maintenance : matérielles et logicielles | | 3 |
| Diagnostique et Dépannage de pannes | | 3 |
| Bases des données | Access, MySQL, PostGreSQL, Oracle | 3 |

APPARTENANCE À UNE ORGANISATION OU UNE ASSOCIATION

Membre du Club Linux et Logiciels Libres de Fianarantsoa (C3LF) ;

Membre du Club Multimédia à L'ENI ;

Président de l’association AENAM (Association des Etudiants Natifs et Avoisinât de Manambondro) ;

Membre de l’association FITEMAFI (Fikambanan’i Teraky Manambondro et Fianarantsoa).

LOISIRS

* Sport: Foot-ball, Basket-ball;
* Jeux PC;
* Internet;
* Music;
* Lecture.

SOMMAIRE

[AVANT-PROPOS i](#_Toc413368185)

[REMERCIEMENTS ii](#_Toc413368186)

[CURRICULUM VITAE iii](#_Toc413368187)

[SOMMAIRE vi](#_Toc413368188)

[LISTE DES FIGURES ix](#_Toc413368189)

[LISTE DES TABLEAUX xi](#_Toc413368190)

[LISTE DES ABEVIATIONS xii](#_Toc413368191)

[LISTE DES ANNEXE xiii](#_Toc413368192)

[INTRODUCTION GENERALE 1](#_Toc413368193)

[**Partie I.** **PRESENTATIONS** 2](file:///F:\Stage\Livre\MEMOIRE-FIN-D'ETUDE-RANDRIANARIKOTO-ALEXANDRE.docx#_Toc413368194)

[Chapitre 1 PRESENTATION DE L’ECOLE NATIONALE DE L’INFORMATIQUE 3](#_Toc413368195)

[**1.1.** **Historique** 3](#_Toc413368196)

[**1.2.** **Organigramme** 4](#_Toc413368197)

[**1.3.** **Domaines de spécialisation** 4](#_Toc413368198)

[**1.4.** **Architecture de la pédagogie** 5](#_Toc413368199)

[**1.5.** **Filières de formation existantes et diplômes délivrés** 6](#_Toc413368200)

[**1.6.** **Partenariat** 7](#_Toc413368201)

[**1.7.** **Projet et perspectives de developpement institutionnel** 9](#_Toc413368202)

[**1.8.** **Ressources humaines** 9](#_Toc413368203)

[**1.9.** **Missions** 9](#_Toc413368204)

[Chapitre 2 PRESENTATION DE L’ENTREPRISE IRE-OI (Internet Révolution Evolution de l’Océan Indien ) 11](#_Toc413368205)

[**2.1.** **Historique** 11](#_Toc413368206)

[**2.2.** **Fiche d’Identification** 11](#_Toc413368207)

[**2.3.** **Organisation (organigramme général)** 12](#_Toc413368208)

[**2.4.** **Objectifs : Mission / Activités** 12](#_Toc413368209)

[**2.5.** **Patrimoine** 14](#_Toc413368210)

[Chapitre 3 PRESENTATION GENERALE DU PROJET 15](#_Toc413368211)

[**Introduction** 15](#_Toc413368212)

[**3.1.** **Objectif et besoins de l’utilisateur** 15](#_Toc413368213)

[**3.2.** **Moyens nécessaires à la réalisation du projet (Humain, matériel et éventuellement financier)** 17](#_Toc413368214)

[**3.3.** **Résultats attendus** 17](#_Toc413368215)

[**3.4.** **Présentation de la société Fluoo** 18](#_Toc413368216)

[**Conclusion** 18](#_Toc413368217)

[**Partie II.** **ANALYSE ET CONCEPTION** 19](file:///F:\Stage\Livre\MEMOIRE-FIN-D'ETUDE-RANDRIANARIKOTO-ALEXANDRE.docx#_Toc413368218)

[Chapitre 4 ANALYSE PREALABLE 20](#_Toc413368219)

[**Introduction** 20](#_Toc413368220)

[**4.1.** **Etude de l’existant** 20](#_Toc413368221)

[**4.2.** **Critique de l’existant** 21](#_Toc413368222)

[**4.3.** **Solution proposée** 21](#_Toc413368223)

[**4.4.** **Modéles et méthodologie adoptées** 21](#_Toc413368224)

[Conclusion 36](#_Toc413368225)

[Chapitre 5 ANALYSE CONCEPTUELLE 37](#_Toc413368226)

[**Introduction** 37](#_Toc413368227)

[**5.1.** **Indentification des fonctionnalités (Features)** 37](#_Toc413368228)

[**5.2.** **Les user stories** 38](#_Toc413368229)

[**5.3.** **Planification des sprints** 44](#_Toc413368230)

[**5.1.** **Diagrammes des cas d’utilisation globale** 51](#_Toc413368231)

[**5.2.** **Etablissement des diagrammes de classe globale utilisé dans les différents sprints** 52](#_Toc413368232)

[**5.3.** **Conception détaillée par sprint** 57](#_Toc413368233)

[Conclusion 100](#_Toc413368234)

[**Partie III.** **REALISATIOIN** 101](file:///F:\Stage\Livre\MEMOIRE-FIN-D'ETUDE-RANDRIANARIKOTO-ALEXANDRE.docx#_Toc413368235)

[Chapitre 6 REALISATION 102](#_Toc413368236)

[**Introduction** 102](#_Toc413368237)

[**6.1.** **Choix des technologies** 102](#_Toc413368238)

[**6.2.** **Architecture finale de l’application** 111](#_Toc413368239)

[**6.3.** **Environnement d’implémentation** 111](#_Toc413368240)

[**6.4.** **Extraits de code** 113](#_Toc413368241)

[**6.5.** **Quelques captures d’écran** 119](#_Toc413368242)

[Conclusion 124](#_Toc413368243)

[CONCLUSION GENERALE 126](#_Toc413368244)

[REFERENCE BIBLIOGRAPHIQUES xiii](#_Toc413368245)

[REFERENCE WEBOGRAPHIQUES xiv](#_Toc413368246)

[ANNEXE : CAHIER DES CHARGES viii](#_Toc413368247)

[**1.** **Présentation du projet** viii](#_Toc413368248)

[**2.** **Objet du document** viii](#_Toc413368249)

[**3.** **Contexte d’élaboration** viii](#_Toc413368250)

[**4.** **Plan du document** ix](#_Toc413368251)

[**5.** **Intevenants** ix](#_Toc413368252)

[**6.** **Déscription des besoins de la société** ix](#_Toc413368253)

[**7.** **Déscription du nouveau systéme** xii](#_Toc413368254)

[**8.** **Etapes du projet, calendrier d’action** xii](#_Toc413368255)

[GLOSSIAIRE xiv](#_Toc413368256)

[TABLE DE MAIERES xvii](#_Toc413368257)

[RESUME xxiv](#_Toc413368258)

[ABSTRACT xxv](#_Toc413368259)

LISTE DES FIGURES

[Figure 1 : Organigramme de l’Ecole Nationale d’Informatique 4](#_Toc413368321)

[Figure 2 : Organigramme de l’IRE-OI 12](#_Toc413368322)

[Figure 3 : Formalisme du diagramme de cas d'utilisation 31](#_Toc413368323)

[Figure 4 : Formalisme du diagramme d’activités 32](#_Toc413368324)

[Figure 5 : Formalisme d’un diagramme d’activité avec transition conditionnelle 33](#_Toc413368325)

[Figure 6 : Formalisme d’un diagramme d’activité avec une barre de synchronisation 33](#_Toc413368326)

[Figure 7 : Formalisme du diagramme de séquence 34](#_Toc413368327)

[Figure 8 : Les associations entre classes 35](#_Toc413368328)

[Figure 9 : Diagramme de cas d’utilisation globale de l’application 51](#_Toc413368329)

[Figure 10 : Diagramme de classe globale de l’application 56](#_Toc413368330)

[Figure 11 : Diagramme de cas d’utilisation « Inscription» 58](#_Toc413368331)

[Figure 12 : Diagramme d’activité « Inscription» 59](#_Toc413368332)

[Figure 13 : Diagramme de séquence « Inscription» 60](#_Toc413368333)

[Figure 14 : Diagramme de cas d’utilisation « Gérer utilisateur» 61](#_Toc413368334)

[Figure 15 : Diagramme d’activité « Gérer utilisateur» 62](#_Toc413368335)

[Figure 16 : Diagramme de séquence « Gérer utilisateur» 63](#_Toc413368336)

[Figure 17 : Diagramme de cas d’utilisation « Gérer abonnement» 64](#_Toc413368337)

[Figure 18 : Diagramme d’activité « Gérer abonnement» 65](#_Toc413368338)

[Figure 19 : Diagramme de séquence « Gérer abonnement» 66](#_Toc413368339)

[Figure 20 : Diagramme de cas d’utilisation « Souscription à un abonnement » 67](#_Toc413368340)

[Figure 21 : Diagramme d’activité « Souscription à un abonnement » 69](#_Toc413368341)

[Figure 22 : Diagramme de séquence « Souscription à un abonnement » 70](#_Toc413368342)

[Figure 23 : Diagramme de cas d’utilisation« Gérer fichier» 70](#_Toc413368343)

[Figure 24 : Diagramme d’activité« Gérer fichier» 71](#_Toc413368344)

[Figure 25 : Diagramme de séquence« Gérer fichier» 72](#_Toc413368345)

[Figure 26 : Diagramme de cas d’utilisation« Acheter fichier à la volet» 73](#_Toc413368346)

[Figure 27 : Diagramme d’activité« Acheter fichier à la volet» 74](#_Toc413368347)

[Figure 28 : Diagramme de séquence« Acheter fichier à la volet» 75](#_Toc413368348)

[Figure 29 : Diagramme de cas d’utilisation« Acheter fichier en utilisant le panier» 76](#_Toc413368349)

[Figure 30 : Diagramme d’activité« Acheter fichier en utilisant le panier» 78](#_Toc413368350)

[Figure 31 : Diagramme de séquence« Acheter fichier en utilisant le panier» 79](#_Toc413368351)

[Figure 32 : Diagramme de cas d’utilisation« Gérer lightbox » 80](#_Toc413368352)

[Figure 33 : Diagramme d’activité« Gérer lightbox» 81](#_Toc413368353)

[Figure 34 : Diagramme de séquence« Gérer lightbox» 81](#_Toc413368354)

[Figure 35 : Diagramme de cas d’utilisation« Gérer lightbox » 82](#_Toc413368355)

[Figure 36 : Diagramme d’activité« Gérer compte de la société» 84](#_Toc413368356)

[Figure 37 : Diagramme de séquence« Gérer compte de la société» 84](#_Toc413368357)

[Figure 38 : Diagramme de cas d’utilisation« Gérer compte de l’utilisateur » 85](#_Toc413368358)

[Figure 39 : Diagramme d’activité« Gérer compte de l’utilisateur» 86](#_Toc413368359)

[Figure 40 : Diagramme de séquence« Gérer compte de l’utilisateur» 87](#_Toc413368360)

[Figure 41 : Diagramme de cas d’utilisation« Gérer le panier » 88](#_Toc413368361)

[Figure 42 : Diagramme d’activité« Gérer le panier» 89](#_Toc413368362)

[Figure 43 : Diagramme de séquence« Gérer le panier» 90](#_Toc413368363)

[Figure 44 : Diagramme de cas d’utilisation« Uploader fichier » 91](#_Toc413368364)

[Figure 45 : Diagramme d’activité« Uploader fichier» 92](#_Toc413368365)

[Figure 46 : Diagramme de séquence« Uploader fichier» 92](#_Toc413368366)

[Figure 47 : Diagramme de cas d’utilisation« Créer sous compte » 93](#_Toc413368367)

[Figure 48 : Diagramme d’activité« Créer sous compte» 94](#_Toc413368368)

[Figure 49 : Diagramme de séquence« Créer sous compte» 95](#_Toc413368369)

[Figure 50 : Diagramme de cas d’utilisation« Mise à jour de l’abonnement » 96](#_Toc413368370)

[Figure 51 : Diagramme d’activité« Mise à jour de l’abonnement» 98](#_Toc413368371)

[Figure 52 : Diagramme de séquence« Mise à jour de l’abonnement» 99](#_Toc413368372)

[Figure 53 : Transmission du code JavaScript au template 106](#_Toc413368373)

[Figure 54 : Architecture finale de l’application 111](#_Toc413368374)

[Figure 55 : Page d’accueil de l’application côté client 119](#_Toc413368375)

[Figure 56 : Page de formulaire de souscription à un abonnement 120](#_Toc413368376)

[Figure 57 : Page de validation de la souscription à un abonnement 120](#_Toc413368377)

[Figure 58 : Achat fichier en utilisant le panier 121](#_Toc413368378)

[Figure 59 : Page d’accueil de l’application côté client 122](#_Toc413368379)

[Figure 60 : Page d’uploade de fichier 123](#_Toc413368380)

[Figure 61 : Formulaire d’information du fichier 124](#_Toc413368381)

LISTE DES TABLEAUX

[Tableau 1 : Liste des personnels et matériels 17](#_Toc413368382)

[Tableau 2 : Liste des fonctionnalités 37](#_Toc413368383)

[Tableau 3 : Liste des user stories 38](#_Toc413368384)

[Tableau 4 : Planification de l’itération 1 44](#_Toc413368385)

[Tableau 5 : Planification de l’itération 2 44](#_Toc413368386)

[Tableau 6 : Planification de l’itération 3 45](#_Toc413368387)

[Tableau 7 : Planification de l’itération 4 45](#_Toc413368388)

[Tableau 8 : Planification de l’itération 5 46](#_Toc413368389)

[Tableau 9 : Planification de l’itération 6 46](#_Toc413368390)

[Tableau 10 : Planification de l’itération 7 47](#_Toc413368391)

[Tableau 11 : Planification de l’itération 8 47](#_Toc413368392)

[Tableau 12 : Planification de l’itération 9 48](#_Toc413368393)

[Tableau 13 : Planification de l’itération 10 48](#_Toc413368394)

[Tableau 14 : Planification de l’itération 11 49](#_Toc413368395)

[Tableau 15 : Planification de l’itération 12 49](#_Toc413368396)

[Tableau 16 : Planification de l’itération 13 50](#_Toc413368397)

[Tableau 17 : Planification de l’itération 14 50](#_Toc413368398)

[Tableau 18 : Dictionnaire de données de l’application 52](#_Toc413368399)

[Tableau 19 : Liste des outils et logiciels 112](#_Toc413368400)

[Tableau 18 : Plan du document ix](#_Toc413368401)

[Tableau 19 : Liste des fonctionnalités au niveau de l’utilisateur x](#_Toc413368402)

[Tableau 20 : Liste des fonctionnalités au niveau de l’administrateur xi](#_Toc413368403)

LISTE DES ABEVIATIONS

**API** : **A**pplication **P**rogramming **I**nterface

**AJAX** : **A**synchronous **J**avaScript **A**nd **X**ML

**BD** : **B**ase de **D**onnées

**CSS** : **C**ascading **S**tyle **S**heets

**ENI**  : **E**cole **N**ationale d’**I**nformatique

**HTML** : **H**yper**T**ext **M**arkup **L**anguage

**HTTP**  : **H**yper**T**ext **T**ransfer **P**rotocol

**INTERNET**  : **INTER**national **NET**work

**IRE-OI** : **I**nternet **R**évolution – **E**volution de l’**O**céan **I**ndien

**JSON** : **J**ava**S**cript **O**bject **N**otation

**LMD** : **L**icence **M**aster **D**octorat

**MVC** : **M**odèle **V**ue **C**ontrôleur

**SGBD**  : **S**ystème de **G**estion de **B**ase de **D**onnées

**UML** : **U**nified **M**odeling **L**anguage

**URL** : **U**niform **R**esource **L**ocator

**WWW**  : **W**orld **W**ide **W**eb

LISTE DES ANNEXE

[ANNEXE - A : CAHIER DES CHARGES **Erreur ! Signet non défini.**](#_Toc314806669)

INTRODUCTION GENERALE

Le développement des sites internet tient une place importante dans le commerce. Des ventes de mains en mains, vers des ventes virtuelles, passeles priorités des opérations de ventes des biens et des services, ce qui nous rend obligés de donner plus d’importance à la vente électronique.

Les sites de vente en ligne permettent aux clients de profiter d’une foire virtuelle disponible est quotidiennement mise à jours sans la moindre contrainte, ce qui leur permettrai de ne jamais rater les coups de cœur, ainsi une foire sans problèmes de distance géographique, ni d’horaire de travail ni de disponibilité de transport. D’une autre part ces sites offrent à la société de profiter de cette espace pour exposer ses produits à une plus large base de clientèle.

Le présente projet réalisé dans le cadre du mémoire de master 2ayant comme objectif principal de développer un site e-commerceresponsive qui permet de vendre des images, vecteurs et vidéos via un ordinateur connecté à internet ou même un appareil mobile.

L’objet du mémoire est donc la réalisation du projet « conception et réalisation d’un site responsive pour les commerces des images, vecteurs et vidéos en utilisant la technologie Nodejs pour le compte de Fluoo» effectuer durant mon stage au sein de l’entreprise IRE-OI.

Pour bien appréhender, voici le plan à suivre pour la réalisation de ce présent projet. La présente mémoire est divisée en trois grandes parties. Tout d’abord nous allons commencer par la première partie, c’est-à-dire la présentation générale ou on va voir la présentation de l’ENI, de la société d’accueil, et du présent projet. Ensuite la deuxième partie concerne l’analyse conceptuelle qui est constitué par l’analyse préalable et la conception. Et pour finir, la troisième partie concerne la réalisation.

1. **PRESENTATIONS**
2. PRESENTATION DE L’ECOLE NATIONALE DE L’INFORMATIQUE
   1. **Historique**

Créée par le Décret N° 83-185 du 24 mai 1983, l’Ecole Nationale d’Informatique, avec 30 ans d’expérience, est une des Ecoles la plus réputée de Madagascar, elle forme les meilleurs informaticiens du pays.

Filière de formation d’Analystes Programmeurs a été créée en 1983, et a été gelée par la suite en 1996.

La filière de formation d’ingénieurs a été ouverte à l’Ecole en 1996.

La filière de formation de Techniciens Supérieurs en Maintenance des Systèmes Informatiques a été mise en place à l’Ecole en 1996 grâce à l’appui matériel et financier de la Mission Française de Coopération dans le cadre du Programme de Renforcement de l’Enseignement Supérieur (PRESUP).

Une formation pour l’obtention de la certification CCNA et/ou Network+, appelée « Cisco Networking Academy à Madagascar », en 2002-2003, a été créé grâce au partenariat avec Cisco System et l’Ecole Supérieure Polytechnique d’Antananarivo (ESPA).

Une formation doctorale a été ouverte depuis l’année universitaire 2003-2004 avec une parfaite coopération entre l’Université de Fianarantsoa (ENI) et celle de Toulouse.

Finalement une formation en licence professionnelle en informatique ayant comme options Systèmes et réseaux, Génie logiciel et base de données a été ouverte pendant l’année universitaire 2007-2008.

La filière de formation de Techniciens Supérieurs en Maintenance des Systèmes Informatiques a été gelée en 2008.

Finalement une formation hybride en informatique ayant a été ouverte pendant l’année

universitaire 2011-2012.

Pa suite d’évènement, la filière de formation d’Ingénieur en Informatiques a été gelée en 2011.

* 1. **Organigramme**

La figure 1 représente l’organigramme de l’Ecole Nationale d’Informatique.

L’Ecole est administrée par un Conseil d’Ecole, et dirigée par un Directeur nommé par un Décret pris en conseil des Ministres.

L’organigramme de l’Ecole comprend trois départements pédagogiques :

* Le département de la Formation théorique ;
* Le département de la Formation pratique ;
* Le département de la Formation doctorale.

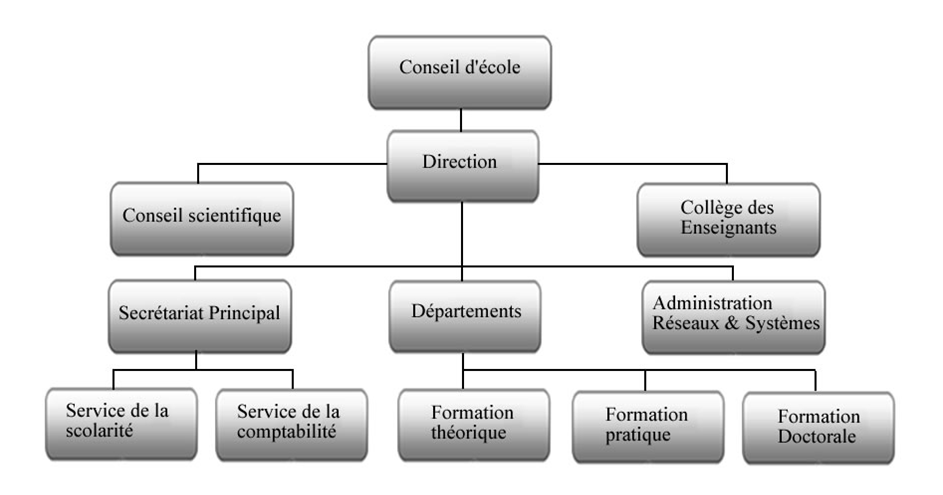


Figure 1 : Organigramme de l’Ecole Nationale d’Informatique

* 1. **Domaines de spécialisation**
* Base de Données et Génie logiciel ;
* Administration des réseaux et systèmes ;
  1. **Architecture de la pédagogie**

L’ENI forme des Techniciens, des Licenciés et des Ingénieurs directement opérationnels au terme de leur formation respective. Ce qui oblige l’Ecole à entretenir des relations de collaboration étroites et permanentes avec les entreprises et le monde professionnel de l’Informatique à Madagascar.

La responsabilité de l’Ecole pour cette professionnalisation des formations dispensées implique de :

* Suivre les progrès technologiques et méthodologiques en Informatique (recherche appliquée, veille technologique, technologies Réseau-Multimédia-Internet,…).
* Prendre en considération dans les programmes de formation les besoins évolutifs des entreprises et des autres utilisateurs effectifs et potentiels de la technologie informatique.

Cependant la professionnalisation ne peut pas se faire en « vase clos », elle exige une «orientation client » et une « orientation marché ». Ce sont les entreprises qui connaissent le mieux leurs besoins en personnel informatique qualifié. Ces entreprises partenaires collaborent avec l’ENI en présentant des pistes et des recommandations pour aménager et réactualiser périodiquement les programmes de formation. Ainsi, dans le cadre de ce partenariat avec les sociétés dans les divers bassins d’emploi en Informatique, l’Ecole offre sur le marché de l’emploi des cadres de bon niveau, directement opérationnels, et avec des connaissances à jour.

L’architecture des programmes pédagogiques à l’Ecole s’appuie sur le couple théorie-pratique :

* Des enseignements théoriques et pratiques de haut niveau sont dispensés intra-muros à l’Ecole,
* Des voyages d’études sont effectués par les étudiants nouvellement inscrits et ayant passé une année d’études à l’Ecole,
* Des stages d’application et d’insertion professionnelle sont pratiqués en entreprise chaque année par les étudiants au terme de chaque formation académique à l’Ecole.

Les stages effectués en entreprise par les étudiants de l’ENI sont principalement des stages de pré- embauche.

Ces stages pratiques font assurer à l’Ecole un taux moyen d’embauche de 97%, six mois après la sortie de chaque promotion de diplômés.

L’Ecole met progressivement en place le système LMD. Ce qui suppose la diversification des filières de formation, ainsi que la réforme et la rénovation des programmes d’enseignement.

* 1. **Filières de formation existantes et diplômes délivrés**
* Cycle licence en informatique spécialisé en administration des systèmes et des réseaux, puis en génie logiciel et base de données, aboutissant au Diplôme Universitaire de licenciés informaticiens. L’effectif des étudiants en année universitaire 2011 – 2012:
* L1 Hybride (Première année de la formation hybride) :150 dont 100 à l’ENI Fianarantsoa et 50 à l’IHM de l’Université de Tuléar,
* L1 (Première année de la formation de Licence) : 100,
* L2 Hybride (Deuxième année de la formation hybride) : 150 dont 100 à !’ENI Fianarantsoa et 50 à l’IHM de l’Université de Tuléar,
* L2 (Deuxième année de la formation de Licence) : 102,
* L3 Hybride(Troisième année de la formation hybride) :150 dont 100 à !’ENI Fianarantsoa et 50 à l’IHM de l’Université de Tuléar,
* L3 (Troisième année de la formation de Licence) : 63.
* Cycle de formation d’Ingénieurs Informaticiens avec de compétences en Gestion, Systèmes et réseaux, de niveaux Baccalauréat + 5 ans. Les effectifs des étudiants en 2010-2011 :
* M1 (Première année de la formation Master) : 100,
* M2 (Deuxième année de la formation Master) : 100.
* La formation en DEA en informatique organisée en partenariat avec l’Université Paul Sabatier de Toulouse. Les trois meilleurs étudiants de la promotion effectuent les travaux de recherche à Toulouse. Cette formation est un élément du système de formation de troisième cycle et d’études doctorales qui sera mise en place progressivement à l’ENI.
* Une formation non diplômant en CISCO ACADEMY, soutenue par les Américains, avec certification CCNA. Les effectifs des étudiants dans le système depuis sa création :
* CISCO Première promotion 2002/2003 : 28,
* CISCO Deuxième promotion 2003/2004 : 5.

Le recrutement d’étudiants à l’ENI se fait chaque année uniquement par voie de concours d’envergure nationale, excepté celui concernant le « Cisco Academy » et celui de la DEA, qui font l’objet de sélections des dossiers de candidature.

Bien qu’il n’existe pas au niveau international de reconnaissance écrite et formelle des diplômes délivrés par l’ENI, les diplômés de l’Ecole sont bien accueillis dans les Institutions universitaires étrangères. Des étudiants diplômés de l’Ecole poursuivent actuellement leurs études supérieures en 3ème cyclé dans plusieurs Universités françaises, notamment à l’IREMIA de l’Université de la Réunion, à l’Université LAVAL au Canada, à l’Ecole Polytechnique Fédérale de Lausanne en SUISSE, à l’Ecole Doctorale STIC (Science de la Technologie de l’Information et de la communication) de l’Ecole Supérieure en Science Informatique de l’Université de Nice Sophia Antipolis.

* 1. **Partenariat**
     1. **Au niveau national**

Les stages pratiqués chaque année par ses étudiants mettent l’Ecole en relation permanente avec plus de 300 entreprises, sociétés et organismes publics et privés nationaux et internationaux.

Parmi ces Etablissements, on peut citer : DTS, TELMA, Banque Centrale, BNI-CL, Agence Universitaire de la Francophonie, Ministère des finances et de l’Economie, Fermic On Line Malagasy, CFSIGE, MIC, Cabinet FTHM, INGENOSYA, INTERCEL Madagascar, IBONIA,JIRAMA, NY HAVANA, COTONA, IOGA, SIMICRO, WWF, International, FID, AFOMA, CNRIT, BULL MADAGASCAR, OMNITEC, ANGAP, SECREN, AIR MADAGASCAR, Société d’Exploitation du port de Toamasina, MICROTEC, GNOSYS, TACTI , COPEFRITO, CNA, QMM ,MFB, Société AIRTEL, Société ORANGE, ACCENTURE Maurice…

L’organisation de stages en entreprise contribue non seulement à assurer une meilleure professionnalisation des formations dispensées, mais elle accroît également de façon exceptionnelle les opportunités d’embauche pour les diplômés.

Les diplômés de l’ENI sont recrutés non seulement par des entreprises et organismes nationaux, mais ils sont aussi embauchés dans des organismes de coopération internationale tels que l’USAID MADAGASCAR, la Délégation de la Commission Européenne, la Banque Africaine de Développement (BAD), la Mission Résidente de la Banque Mondiale, la Commission de l’Océan Indien, etc.

* + 1. **Au niveau international**

Entre 1996 et 1999, l’ENI a bénéficié de l’assistance technique et financière de la Mission Française de Coopération et d’Action Culturelle dans le cadre du PRESUP.

La composante du PRESUP consacré à l’ENI a notamment porté sur :

* Une dotation en logiciels, microordinateurs, équipements de laboratoire de maintenance et de matériels didactiques ;
* La réactualisation des programmes de formation assortie du renouvellement du fond de la bibliothèque ;
* L’appui à la formation des formateurs ;
* L’affectation à l’Ecole d’Assistants techniques français.

Et depuis le mois de mai 2000, l’ENI fait partie des membres de bureau de la Conférence Internationale des Ecoles de formations d’Ingénieurs et Techniciens d’Expression Française (CITEF).

L’ENI a signé un Accord de coopération interuniversitaire avec l’IREMIA de l’Université de la Réunion, l’Université de RENNES 1 et l’Institut National Polytechnique de Grenoble (INPG).

Depuis le mois de juillet 2001, l’ENI abrite le Centre du Réseau Opérationnel (Network Operating Center) du point d’accès à internet de l’Ecole et de l’Université de Fianarantsoa. Grâce à ce projet américain financé par l’USAID Madagascar, l’ENI et l’Université de Fianarantsoa sont maintenant dotées d’une Ligne Spécialisée d’accès permanent à INTERNET.

Par ailleurs, depuis 2002, une nouvelle branche à vocation professionnelle a pu y être mise en place, en partenariat avec Cisco System.

Enfin et non de moindres, l’ENI a noué des relations de coopération avec l’Institut de Recherche pour le Développement (IRD). L’objet de la coopération porte sur la Modélisation environnementale du corridor forestier de Fianarantsoa. Dans le même cadre, un atelier scientifique international sur la modélisation des paysages a été organisé à l’ENI au mois de Septembre 2008.

Comme l’ENI constitue une pépinière incubatrice de technologie de pointe, d’emplois et d’entreprises, elle peut servir d’instrument efficace pour la lutte contre la pauvreté.

De même que l’Ecole permet de renforcer la position concurrentielle de la Grande île sur l’orbite de la mondialisation grâce au développement des nouvelles technologies.

* 1. **Projet et perspectives de developpement institutionnel**
* Restructuration du système pédagogique de l’Ecole selon le schéma LMD (Licence Master Doctorat),
* Mise en place à l’Ecole d’un Département de Formation de 3ème cycle et d’études doctorales en Informatique.
  1. **Ressources humaines**
* Directeur : Docteur RAFAMANTANANTSOA Fontaine,
* Chef de Département  de la Formation Théorique : Docteur  Venot RATIARSON,
* Chef de Département de la Formation Pratique : DocteurCyprien RAKOTOASIMBAHOAKA,
* Responsable de la filière de formation en DEA : Professeur Josvah Paul RAZAFIMANDIMBY,
* Nombre d’Enseignants permanents : 12,
* Nombre d’Enseignants vacataires : 10,
* Personnel administratif et technique : 19.
  1. **Missions**

L’ENI de l’Université de Fianarantsoa constitue à l’heure actuelle la pépinière des élites informaticiennes malgaches. On peut considérer cette Ecole Supérieure comme la vitrine et la mesure de l’avancée technologique du Pays.

Elle se positionne dans le système socio-éducatif malgache comme le plus puissant vecteur de diffusion et de vulgarisation des connaissances et des technologies informatiques. et elle est le seul Etablissement universitaire professionnalisé ayant pour mission de former, des Licenciés en informatique et des Ingénieurs informaticiens de haut niveau, aptes à répondre aux besoins et exigences d’informatisation des entreprises, des sociétés et des organismes implantés à Madagascar. Et avec ses nombreux partenaires, l’étudiant sortant a aussi l’opportunité de travailler à l’étranger.

L’implantation de cette Ecole Supérieure de technologie de pointe dans un pays en voie de développement et dans une province à tissu économique et industriel faiblement développé, ne l’ont pourtant pas empêché de former des spécialistes informaticiens de bons niveaux recherchés par les sociétés et les organismes.

Depuis sa création jusqu’à aujourd’hui, l’ENI placée sous la tutelle de l’Université de Fianarantsoa a formé et déversé sur le marché de l’emploi :

* 13 promotions d’Analystes Programmeurs, soit 447 diplômés,
* 22 promotions d’Ingénieurs Informaticiens, soit 554 diplômés,
* 13 promotions de Techniciens Supérieurs en Maintenance des Systèmes Informatiques, soit 310 diplômés.

Soit en tout 1311 diplômés.

1. PRESENTATION DE L’ENTREPRISE IRE-OI (Internet Révolution Evolution de l’Océan Indien )
   1. **Historique**

La Société a fortement évolué au cours des années. En 2005, les deux associées GERVAIS Pierre Luc et RANDRIAMALALA Hery Nandrianina ont créé à Antananarivo la société IISMadagascar SARL. Une représentation commerciale d’IIS-Madagascar en France a été lancée en 2007. La société a établie différents partenaires notamment avec Vtiger Costumer Relationship Management, la société Vectoris en France dans le domaine des jeux vidéo et Infogrammes aux USA. En 2009, le pôle VOIP (Elastix, Asterisk) fut créé. 2012 est l’année du changement, pour des raisons stratégique. IIS-Madagascar a été rebaptisée IRE-OI (Internet Révolution Evolution de l’Océan Indien).

* 1. **Fiche d’Identification**

IRE-OI est une Société de services informatiques spécialisée dans la création de site web en offshore et le référencement. Evoluant dans un secteur d’activité en plein développement, la société tire son épingle du jeu grâce à une équipe polyvalente disposant d’une expérience considérable dans la gestion des projets internationaux. L’entreprise est actuellement gérée par Monsieur Pierre Luc GERVAIS.

La fiche synthétique d’IRE-OI est:

* Raison sociale : Internet Révolution Evolution de l’Océan Indien,
* Forme juridique : Société à Responsabilité Limitée (SARL),
* Date de création : 14 Février 2012,
* Capital social : 10.000.000 Ariary,
* Registre du commerce : 2012 B 00320 Antananarivo,
* Numéro d’identité statistique : 72309 11 2012 0 10343,
* Numéro d’identité fiscale : 0023180,
* Siège social : Lot 423 Cité Ampefiloha Antananarivo 101,
* Lieu d’exploitation : Lot III G 133 H Ambatolampy Ambohibao Antananarivo,
* Gérants : GERVAIS Pierre Luc et RANDRIAMALALA Hery Nandrianina,
* Téléphone : 034 96 496 95,
* E-mail: info@internet-revolution-oi.com,
* Site web: http://www.internet-revolution-oi.com.
  1. **Organisation (organigramme général)**

IRE-OI emploie une quarantaine personnes incluant des informaticiens, des graphistes, des techniciens, des commerciaux ainsi que le personnel administratif. Des formations dans différents domaines sont régulièrement dispensées au personnel afin de garder un niveau de compétence élevé et maintenir des prestations de qualité.

La figure 2 présente l’organigramme actuel de l’entreprise IREOI.

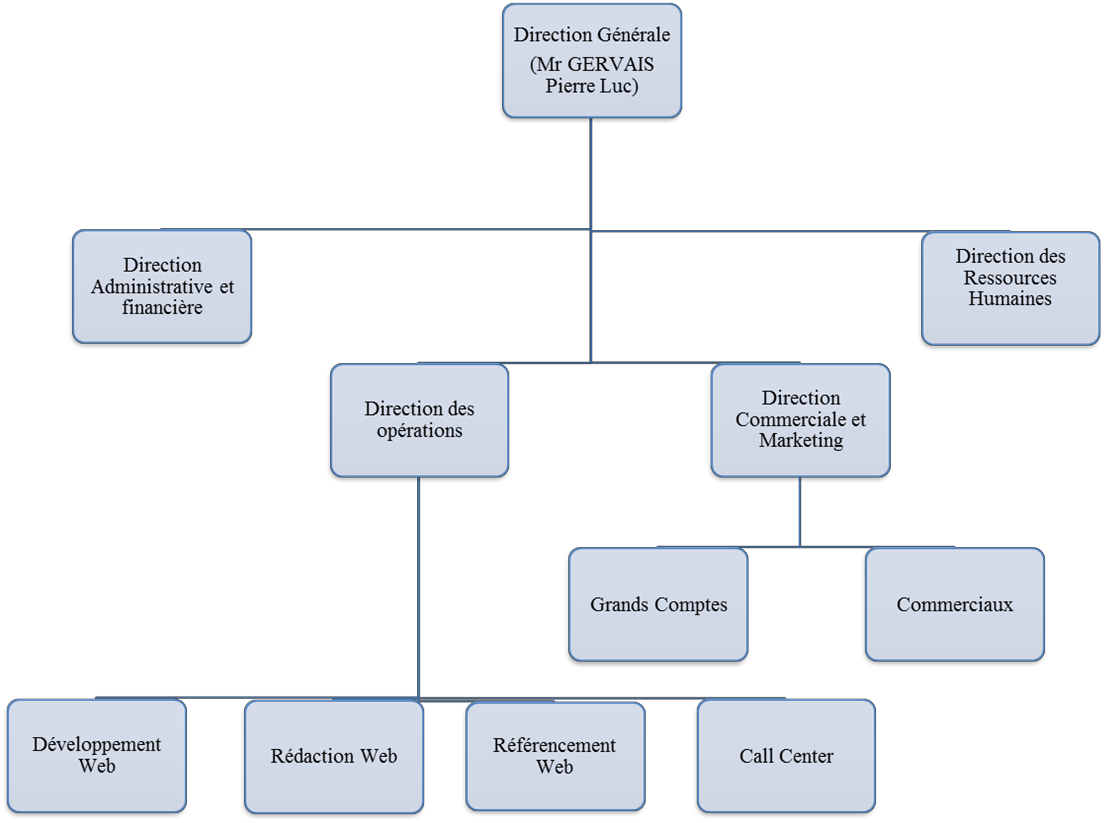


Figure 2 : Organigramme de l’IRE-OI

* 1. **Objectifs : Mission / Activités**
     1. **Développement Web**

La conception des sites web constitue l’activité principale de l’IRE-OI et celle qui fait sa renommée. L’entreprise veille tout particulièrement à préserver l’identité numérique de ses clients avec des conceptions graphiques soignées et un contenu rédactionnel irréprochable. En effet, le développement web est devenu un secteur très concurrentiel à Madagascar grâce à une clientèle de plus en plus importante aussi bien local via développement des NTIC internationale avec des clients attirés par les prix compétitifs résultants d’une main d’œuvre bon marché. Il est donc devenu important de se démarquer des autres entreprises évoluant dans le secteur. Par ailleurs, pour la création de site web, les outils utilisés par IRE-OI s’appuient sur des Content Management System Open Source comme Joomla, Prestashop ou propriétaire comme Miki.

IRE-OI conçoit différents types de site web comme :

* Sites brochures avec formulaire de contact
* Portails collaboratifs
* Sites e-commerce
* Sites catalogues produits
  + 1. **Rédaction web**

La rédaction web est un domaine qui a également connu un essor remarquable à Madagascar, attirant surtout une clientèle à l’étrangère. En effet, les prix à Madagascar sont compétitifs tout en disposant d’une main d’œuvre qualifiée ; Il est donc normal que IRE-OI développe cette activité, en assurant une formation régulière à ses rédacteurs WEB afin que ceux-ci puissent assurer une qualité de rédaction élevée à des clients exigeants et faire face à des concurrents de plus en plus nombreux.

* + 1. **Référencement naturel**

Le référencement est devenu incontournable pour toute entreprise qui veut évoluer sur le WEB de manière efficace mais surtout rentable. Le référencement naturel apporte une bonne visibilité à un site web dans les moteurs de recherche les plus courants (Google, Yahoo…). IRE-OI propose donc une optimisation des sites web de ses clients pour garantir des visites qualifiées.

* + 1. **Social media marketing**

Les réseaux sociaux (Facebook, Twitter, Viadeo…) font maintenant partie intégrante de la vie de tout internaute et constitue un outil marketing des plus efficaces. Le Social Media Marketing est donc devenu un moyen de garantir une identité numérique favorable auprès des clients et d’en identifier de nouveaux. IRE-OI offre ainsi à ses clients des solutions d’optimisation de marketing social et de réputation en ligne.

* + 1. **Call center**

Tout comme la rédaction WEB, les « call centers offshore » se sont beaucoup développés à Madagascar grâce aux tarifs peu élevés qui attirent les clients internationaux. IRE-OI s’est aussi placé sur ce marché à potentiel en se lançant dans le VoIP (Voice over Internet Protocol), qui est une technique qui permet de communiquer par la voix (ou via des flux multimédia : audio et/ou Vidéo) sur Internet ou avec des réseaux compatibles IP. Ainsi, l’entreprise fournit des sociétés étrangères en téléopérateurs pour des services de télémarketing, télé-conseil, etc.

* 1. **Patrimoine**

La Société IRE-OI dispose des matériels et logiciels informatiques pour soutenir son activité. Une quarantaine d’ordinateurs et trois (3) connexions internet à haut débit de 2Mbps souscrites auprès de trois fournisseurs différents sont utilisées pour assurer une disponibilité permanente de la communication, notamment avec l’étranger, car l’outsourcing est le cœur de métier de l’entreprise.

1. PRESENTATION GENERALE DU PROJET

**Introduction**

Dans ce chapitre nous commençons par définir les objectifs et les besoins des utilisateurs, ensuite nous citons les moyens nécessaires à la réalisation du projet et pour clore ce chapitre on va voir les résultats attendus

* 1. **Objectif et besoins de l’utilisateur** 
     1. **Objectif**

Notre projet intitulé « conception et réalisation d’un site e-commerce responsive en utilisant la technologie NodeJS pour le compte de Fluoo» consiste à développer un site web dynamique permettant l’inscription des visiteurs pour devenir membres; aussi de visualiser, acheter et vendre des fichiers telles que des images, vecteurs et vidéos selon les catégories, qualités et tailles spécifiques aux besoins de client. L’application permet aussi à l’administrateur de gérer le site, c’est-à-dire d’effectuer des uploades des fichiers (images, vecteurs, vidéos) en les personnalisant, puis en gérant tous les utilisateurs du site. Aussi l’administrateur peut gérer les demandes effectuées par le contributeur[1]. L’application permette aussi à Fluoo d’administrer le crédit, le moyen de paiement.

* + 1. **Besoins des utilisateurs**

L’application a envisagée va satisfaire les besoins fonctionnels qui seront exécutés par le système et les besoins non fonctionnels perfectionnent la qualité logicielle du système.

* + - 1. **Les besoins fonctionnels**

Les actions que le système doit exécuter sont représentées par les besoins fonctionnels. Mais ces actions ne sont opérationnelles que si l’application le satisfait. Cette application doit couvrir les fonctionnalités suivantes[1] :

* Exposition des fichiers ainsi que leurs prix et caractéristiques,
* Consultation des nouveaux fichiers du Fluoo,
* Voir les informations, prix et descriptions d’un fichier,
* Inscription des clients,
* Souscription à un abonnement,
* Choix du moyen de paiement,
* Paiement en ligne,
* Ajout des fichiers choisis au panier,
* Gérer les fichiers du panier,
* Gérer les fichiers du lightbox,
* Acheter des fichiers à la volet,
* Téléchargement des fichiers,
* Acheter des fichiers en utilisant le panier,
* Recherche par catégorie ou par mots clés les fichiers,
* Gérer les utilisateurs, les abonnements, le fichier et le compte de la société
  + - 1. **Les besoins non fonctionnels**

Les besoins non fonctionnels sont importants car ils agissent de façon indirecte sur le résultat et sur le rendement de l’utilisateur, ce qu’ils ne doivent pas être négligés, pour cela il faut répondre auxexigences suivantes :

* **Fiabilité**: L’application doit fonctionner de façon cohérente sans erreurs et doit être satisfaisante.
* **Les erreurs** : Les ambigüités doivent être signalées par des messages d’erreursbien organisés pour bien guider l’utilisateur et le familiariser avec notresite web.
* **Ergonomie et bonne Interface** :L’application doit être adaptée à l’utilisateur sans qu’il ne fournisse aucun effort (utilisation claire et facile) de point de vue navigation entreles différentes pages, couleurs et mise en textes utilisés.
* **Sécurité** :Notre solution doit respecter surtout la confidentialité des données personnelles des clients qui reste l’une des contraintes les plusimportantes dans les sites web.
* **Aptitude à la maintenance et la réutilisation** :Le système doit être conforme à une architecture standard et claire permettant sa maintenance et sa réutilisation.
* **Compatibilité et portabilité** :Un site web quel que soit son domaine, son éditeur et son langage de programmation ne peut être fiable qu’avec une compatibilité avec tousles navigateurs web et tous les moyens que ce soit PC, IPAD ou Mobiles.
  1. **Moyens nécessaires à la réalisation du projet (Humain, matériel et éventuellement financier)**

La réalisation d’une telle application nécessite des moyens plus sophistiquées pour que ce dernier donne une meilleure satisfaction aux clients.

D’où les moyens utilisés lors de la réalisation de cette présente application ainsi que les personnels qui l’utilisent sont représentés par le table1.

Tableau 1 : Liste des personnels et matériels

|  |  |  |
| --- | --- | --- |
| Personnel | Matériel utilisé | Caractéristique du matériel |
| RANDRIANARIKOTO Alexandre Boris | Ordinateur portable | Intel(R) Core™ i3-2350M CPU @2.30GHz, ~2,30GHz, 4086MB RAM, 250Go DD |
| RAMPIANDRA RAVELONA Masinera | Ordinateur de Bureau | Intel(R)Celeron® CPU J1800 @2.41GHz(2CPUs), ~2,4GHz, 4096MB RAM, 55Go DD |

Pour le financement du projet, Mr Pierre Luc Gervais, le Directeur Générale de la société IRE-OI assure à ce que la date prévue de la livraison du projetet que le financement des besoins soient équitables et que le déroulement du projet respecte la norme proposé selon le cahier de charge.

* 1. **Résultats attendus**

Dés lors, la mise en place d’un site e-commerce devra répondre à ces attentes [1]:

* Gagner de nouveaux clients,
* Augmenter le chiffre d’affaires de l’entreprise,
* Augmenter la notoriété de l’entreprise,
* Fidéliser les clients actuels de l’entreprise,
* Se renforcer face à la concurrence.
  1. **Présentation de la société Fluoo**
     1. **Historique**

Fluoo est une agence de  Communication qui a son siège social à  Rhône-Alpes entre Lyon, Grenoble et Chambéry, Z.A Le PARVIS Rue du Vallon,  38500 Voiron  France.

La société a vu le jour en 2005. Elle est spécialisée dans les stratégies marketing.

* + 1. **Missions et objectifs**

La principale mission de  la société  est orientée vers la création graphique. Dans ce métier,  les prestations proposées par la société sont diverses et larges est apte pour tous projets de création graphique, l’impression  et fabrication de supports imprimés.

 Pour  garantir la réussite des projets que les clients soumettent et que  leur confiance basée sur la qualité du service fournit par l’entreprise ne soit pas effacée, elle offre aux clients  des conseils  et stratégies qui peuvent  être bénéfiques sur différents plans.

* + 1. **Les différentes prestations fournies  par Fluoo**

Comme dit plus haut, la société est compétente premièrement pour la conception graphique de supports de communication.

* Spécialiste de la conception de supports de communication,
* Une société compétente pour l’impression  et  fabrication de supports de communication,
* Des conseils et des stratégies fournis à titre gratuit.

**Conclusion**

Ce chapitre nous a permis d'introduire notre projet et de présenter la société Fluoo, l’analyse préalable sera décrite dans le chapitre suivant.

1. **ANALYSE ET CONCEPTION**
2. ANALYSE PREALABLE

**Introduction**

Dans ce chapitre nous allons mettre le sujet dans son cadre générale, c’est-à-dire une étude qui se focalise sur l’existant suivit des critiques permettant à ce présent projet de se présenter en vue d’une amélioration et de présenter les meilleures solutions appropriés.

* 1. **Etude de l’existant**

L’étude de l’existant trouve son importance dans la manière ou elle mène à comprendre le système actuel pour bien définir les objectifs.

* + 1. **Description de l’existant**

Depuis des années, IRE-OI qui est une entreprise de services informatiques spécialisés dans la création de site web vise à satisfaire les besoins de ces clients. Ceci indique que l’entreprise possède ces propres clients qui lui demandent de service. La société Fluoo est l’un d’eux.

La société Fluoo qui est spécialisée dans la création graphiqueéprouve des soucis sur la manière et moyen pour vendre ces créations. Actuellement, la société possède deux (2) manières pour vendre ces œuvres :

1. Sur le site [www.fotolia.com](http://www.fotolia.com) : Créée en 2004 par deux entrepreneurs français, Fotolia est spécialisée dans la vente d'images et de vidéos d’illustration de qualité professionnelle. Elle est alimentée en continu par des artistes de talent et des agences photo de renommée internationale. La société Fotolia LLC se localise à 41 East, 11th Street, 11th Floor, ville de New York, pays USA, et a pour code postal 10003.

Fluoo est devenue un contributeur de la société Fotolia et c’est pour quoi elle peut vendre ces créations le site [www.fotolia.com](http://www.fotolia.com) .

1. Sur ces clients : Comme tout société, Fluoo possède ces propres clients qui lui commande et achètent ces œuvres.

Malheureusement, toutes ces manières que la société possède ne semblent pas satisfaisantes car les résultats qu’elle veut ne sont pas atteints c’est-à-dire des nombreux fichiers (images) sont encore en stocks.

D‘où la société Fluoo propose au IRE-OI de lui concevoir et réaliser un site web dynamique pour qu’il puisse gérer et vendre ces différents images et, aussi qu’elle puisse bénéficier les avantages de la vente en ligne.

* 1. **Critique de l’existant**

L’étude de l’existant nous a permis de dégager un certain nombre de lacunes :

* Pas de site web pour vendre,
* Insuffisance des clients,
* Des nombreux fichiers sont en stocks.
  1. **Solution proposée**

Afin de pallier aux défaillances, nous proposons de concevoir et réaliser une boutique virtuelle sur Internet.

L'utilisation d'Internet, comme segment de communication de masse, permet également de baisser les coûts marketing et d'autres frais. Les consommateurs et les entreprises s'orientent de plus en plus vers les boutiques en ligne qui permettent la comparaison, la disponibilité des produits et la vérification des prix d’où l’économie considérable du temps.

Il nous est indispensable de préciser à cette étape que notre projetde fin d’étude prendra en considération toutes ces contraintes en essayant de présenter les solutions nécessaires tout en respectant les règles desjeux d’un site web tels que la simplicité de navigation entre les pages, labonne ergonomie et la sécurité des données confidentielles des clients.

* 1. **Modéles et méthodologie adoptées**

Un modèle de développement logiciel désigne toutes les étapes du développement, de sa conception à sa disparition. L'objectif d'un tel découpage est de permettre de définir des jalons intermédiaires admettant la validation du développement logiciel, c'est-à-dire la conformité du logiciel avec les besoins exprimés, et la vérification du processus de développement[2].

L'origine de ce découpage provient du constat que les erreurs ont un coût d'autant plus élevé qu'elles sont détectées tardivement dans le processus de réalisation. Le cycle de vie permet de détecter les erreurs au plus tôt et ainsi de maîtriser la qualité du logiciel, les délais de sa réalisation ainsi que les coûts associés.

Ceci, nous a conduits à adopter l’approche Agile Scrum pour piloter notre projet et pour une question d’uniformité. Nous avons choisi UML comme notation de base dans toute notre modèle de conception .Donc les aspects statiques, dynamiques et physiques du système seront tous notés en UML[2].

* + 1. **Introduction aux approches Agiles**

Les approches agiles sont un ensemble de pratiques de développement itératif et incrémental de logiciel, misant sur des courts cycles de développement, afin de concevoir rapidement des solutions informatiques ayant de la valeur pour le client. Ambler précise qu'il est difficile de trouver une définition spécifique de ce qu'est le développement logiciel   
Agile. Il propose une définition intéressante contenant les éléments principaux qui aident à acquérir une compréhension globale de l'approche[2].

Un processus itératif et incrémental de développement logiciel exécuté dans un esprit de collaboration par des équipes auto organisées, dans un cadre de gouvernance efficace avec suffisamment de cérémonie qui produit des logiciels de haute qualité dans un bon rapport coût-efficacité, et en temps voulu qui répond aux besoins changeants de ses parties prenantes.

Cette définition fait ressortir, entre autres, les quatre valeurs fondamentales que prônent les approches agiles : l’équipe, le logiciel fonctionnel, la collaboration et l’acceptation du changement. Ces valeurs sont d'ailleurs à la base du Manifeste Agile et des principes qui en découlent. Le manifeste a été rédigé par 17 experts du domaine de développement logiciel qui estimaient que les approches traditionnelles n'étaient pas efficaces pour concevoir des logiciels de qualité qui répondent aux besoins des clients. Les approches traditionnelles sont plus axées sur la planification et le contrôle des échéanciers, des dates cibles, du respect des budgets et de la documentation initiale[2].

Les quatre valeurs Agiles sont les suivantes[2] :

* les individus et les interactions priment plutôt que les processus et les outils,
* les logiciels fonctionnels priment plutôt qu’une documentation exhaustive,
* la collaboration avec le client prime plutôt que la négociation de contrat,
* la réponse au changement prime plutôt que le suivi d'un plan.

Les douze principes Agiles sont listés comme suit :

* Notre plus haute priorité est de satisfaire le client en lui livrant rapidement, et ce, de façon continue, un logiciel de qualité.
* Accepter les changements de besoins, même lors du développement. Les processus Agiles exploitent les changements pour augmenter les avantages compétitifs du client.
* Livrer fréquemment un logiciel fonctionnel en visant les délais les plus courts, de quelques semaines à quelques mois.
* Gestionnaires et développeurs doivent travailler ensemble, de façon quotidienne, pour toute la durée du projet.
* Bâtir des projets autour d'individus motivés. Donnez-leur l'environnement et le support dont ils ont besoin, et ayez confiance qu'ils feront le travail.
* La méthode la plus efficace pour transmettre l'information à l'équipe de développement et à l'intérieur de celle-ci est par la conversation de personne à personne.
* Un logiciel fonctionnel est la mesure principale de l'avancement.
* Les processus Agiles favorisent le développement durable. Les responsables, développeurs et utilisateurs devraient pouvoir conserver un rythme constant indéfiniment.
* Une attention continue à l'excellence technique et une bonne conception augmentent l'agilité.
* La simplicité – l'art de minimiser la quantité de travail fait inutilement – est essentiel.
* Les meilleures architectures, exigences et conceptions surgissent des équipes auto-organisées.
* À intervalles réguliers, l'équipe réfléchit à une façon de devenir plus efficace, puis adapte
* et ajuste son comportement en conséquence.
  + 1. **Scrum [2]**

Scrum est une des principales approches agile de gestion et de suivi d’avancement de projets.

* + - 1. **Caractéristiques**

Le terme Scrum est emprunté au rugby à XV et signifie mêlée. Ce processus s'articule en effet autour d'une équipe soudée, qui cherche à atteindre un but, comme c'est le cas dans ce sport pour avancer avec le ballon pendant une mêlée[2].

Le principe de base de Scrum est de focaliser l'équipe sur une partie limitée et maîtrisable des fonctionnalités à réaliser. Ces incréments se réalisent successivement lors de périodes de durée fixe de une à quatre semaines, appelées sprints. Chaque sprint possède, préalablement à son exécution, un but à atteindre, défini par le directeur de produit, à partir duquel sont choisies les fonctionnalités à implémenter dans cet incrément. Un sprint aboutit toujours à la livraison d'un produit partiel fonctionnel. Pendant ce temps, le ScrumMaster a la charge de minimiser les perturbations extérieures et de résoudre les problèmes non techniques de l'équipe[2].

Un principe fort en Scrum est la participation active du client pour définir les priorités dans les fonctionnalités du logiciel et pour choisir celles qui seront réalisées dans chaque sprint. Il peut à tout moment compléter ou modifier la liste des fonctionnalités à produire, mais jamais celles qui sont en cours de réalisation pendant un sprint. Cette interdiction va formellement à l'encontre du principe d'itération (revenir sur les caractéristiques de la fonctionnalité en cours de développement que le concept de "présence de l'utilisateur sur le site" permettrait de favoriser) et d'adaptabilité immédiate, telle la notion de conception émergente basée sur le feedback issu du travail en cours qui représente une des bases de l'Extrême Programming. En résumé, la notion d'incrément est une valeur facilitant le contrôle de pilotage du projet et la notion d'itération (Méthode itérative) est une valeur conduisant à l'adaptabilité ainsi qu'à la qualité fonctionnelle ou technique. Ces points représentent les principes fondamentaux d'une Méthode Agile appliquée à la complexité de l'Ingénierie du logiciel.

* + - 1. **Idées clé**[2]
* Le client au cœur du projet ;
* Esprit d’équipe ;
* La communication est la clé ;
* Simplicité, efficacité et qualité ;
* Flexibilité aux changements ;
* Avancement basé sur le concret.
  + - 1. **Rôles**

Scrum définit trois rôles principaux : le directeur de produit, le facilitateur / animateur et l'équipe. Des intervenants peuvent s'intégrer également au projet de façon plus ponctuelle.

1. **Directeur de produit**

Le directeur de produit (Product Owner) est le représentant des clients ainsi que utilisateurs. C'est lui qui définit l'ordre dans lequel les fonctionnalités seront développées et qui prend les décisions importantes concernant l'orientation du projet. Le terme directeur n'est d'ailleurs pas à prendre au sens hiérarchique du terme, mais dans le sens de l'orientation.

Dans l'idéal, le directeur de produit travaille dans la même pièce que l'équipe. Il est important qu'il reste très disponible pour répondre aux questions de l'équipe et pour lui donner son avis sur divers aspects du logiciel (interface par exemple).

1. **Équipe**

L'équipe ne comporte pas de rôles prédéfinis, elle est auto-gérée. Il n'y a pas non plus de notion de hiérarchie interne : toutes les décisions sont prises ensemble et personne ne donne d'ordre à l'équipe sur sa façon de procéder. Contrairement à ce que l'on pourrait croire, les équipes auto-gérées sont celles qui sont les plus efficaces et qui produisent le meilleur niveau de qualité de façon spontanée.

L'équipe s'adresse directement au directeur de produit. Il est conseillé qu'elle lui montre le plus souvent possible le logiciel développé pour qu'il puisse ajuster les détails d'ergonomie et d'interface par exemple.

1. **Facilitateur / Animateur**

Le facilitateur / animateur (ScrumMaster) joue un rôle capital : c'est lui qui est chargé de protéger l'équipe de tous les éléments perturbateurs extérieurs à l'équipe et de résoudre ses problèmes non techniques (administratifs par exemple). Il doit aussi veiller à ce que les valeurs de Scrum soient appliquées, mais il n'est pas un chef de projet ni un intermédiaire de communication avec les clients.

On parle parfois d'équipe étendue, qui intègre en plus le ScrumMaster et le directeur de produit. Ce concept renforce l'idée que client et fournisseur travaillent d'un commun effort vers le succès du projet.

1. **Intervenants**

Les intervenants (Stakeholders) sont les personnes qui souhaitent avoir une vue sur le projet sans réellement s'y investir dedans. Il peut s'agir par exemple d'experts techniques ou d'agents de direction qui souhaitent avoir une vue très éloignée de l'avancement du projet.

1. **Planification**

Scrum propose une planification opérationnelle à trois niveaux : release/projet, sprint et quotidien.

1. **Sprints**

Scrum est un processus itératif : les itérations sont appelées des sprints et durent en théorie 30 jours calendaires. En pratique, les itérations durent généralement entre 2 et 4 semaines. Chaque sprint possède un but et on lui associe une liste d'items de backlog de produit (fonctionnalités) à réaliser. Ces items sont décomposés par l'équipe en tâches élémentaires de quelques heures, les items de backlog de sprint[2].

Pendant un sprint, les items de backlog de sprint à réaliser ne peuvent pas être changés. Les changements éventuels sont pris en compte dans le backlog de produit et seront éventuellement réalisés dans les sprints suivants[site 1].

Il y a une exception à cela : il se peut que l'équipe se rende compte en cours du sprint qu'elle n'aura pas le temps de terminer un item du backlog de sprint ou au contraire qu'elle aura fini en avance. Dans ce cas, et seulement d'un commun accord entre l'équipe et le directeur du produit, on peut enlever ou ajouter un item à ce qui a été prévu.

1. **Releases**

Pour améliorer la lisibilité du projet, on regroupe généralement des itérations en releases. Bien que ce concept ne fasse pas explicitement partie de Scrum, il est utilisé pour mieux identifier les versions. En effet, comme chaque sprint doit aboutir à la livraison d'un produit partiel, un release permet de marquer la livraison d'une version aboutie, susceptible d'être mise en exploitation[site 1].

Il est intéressant de planifier à l'échelle d’un release, en répartissant les items du backlog de produit sur les sprints, en respectant leur priorité. Bien entendu, ce qui est planifié au-delà du sprint courant peut changer à tout moment, rien n'est figé à l'avance.

1. **Quotidien**

Au quotidien, une réunion, le ScrumMeeting (appelé également réunion Post-it), permet à l'équipe et au ScrumMaster de faire un point d'avancement sur les tâches et sur les difficultés rencontrées.

* + - 1. **Gestion des besoins**

1. **Backlog de produit**

Scrum utilise une approche fonctionnelle pour récolter les besoins des utilisateurs. L'objectif est d'établir une liste de fonctionnalités à réaliser, que l'on appelle backlog de produit (NDT : Le terme backlog peut être traduit par cahier, liste ou carnet de commandes, qui ne collent pas bien avec l'esprit du terme anglais qui évoque aussi une réserve, un retard accumulé ; aussi ce terme a été gardé tel quel).

À chaque item de backlog sont associés deux attributs : une estimation en points arbitraires (voir Estimation) et une valeur client, qui est définie par le directeur de produit (retour sur investissement par exemple). Ce dernier définit dans quel ordre devront être réalisés ces items. Il peut changer cet ordre en cours de projet et même ajouter, modifier ou supprimer des items dans le backlog.

La somme des points des items du backlog de produit constitue le reste à faire total du projet. Cela permet de produire un release burndown chart, qui montre les points restant à réaliser au fur et à mesure des sprints.

**Remarque** : Il arrive souvent qu'on utilise dans Scrum les User Stories de la méthode Extreme Programming, qui propose des pratiques et des techniques intéressantes (le Planning poker pour les estimer par exemple).

1. **Backlog de sprint**

Lorsqu'on démarre un sprint, on choisit quels items du backlog de produit seront réalisés dans ce sprint. L'équipe décompose ensuite chaque item en liste de tâches élémentaires (techniques ou non), chaque tâche étant estimée en heures et ne devant pas durer plus de 2 jours. On constitue ainsi le backlog de sprint.

Pendant le déroulement du sprint, chaque équipier s'affecte des tâches du backlog de sprint et les réalise. Il met à jour régulièrement dans le backlog du sprint le reste à faire de chaque tâche. Les tâches ne sont pas réparties initialement entre tous les équipiers, elles sont prises au fur et à mesure que les précédentes sont terminées.

La somme des heures des items du backlog de sprint constitue le reste à faire total du sprint. Cela permet de produire un sprint burndown chart qui montre les heures restantes à réaliser au fur et à mesure du sprint.

* + - 1. **Estimations**

Scrum ne définit pas spécialement les unités pour les items des backlogs. Néanmoins, certaines techniques se sont imposées de fait.

1. **Items de backlog de produit**

Les items de backlog de produit sont souvent des User Stories empruntées à Extreme Programming. Ces User Stories sont estimées en points relatifs, sans unité. L'équipe prend un item représentatif et lui affecte un nombre de points arbitraire. Cela devient un référentiel pour estimer les autres items. Par exemple, un item qui vaut 2 points représente deux fois plus de travail qu'un item qui en vaut 1. Pour les valeurs, on utilise souvent les premières valeurs de la suite de Fibonacci (1, 2,3,5,8,13), qui évitent les difficultés entre valeurs proches (8 et 9 par exemple).

L'intérêt de cette démarche est d'avoir une idée du travail requis pour réaliser chaque fonctionnalité sans pour autant lui donner une valeur en jours que le directeur de produit serait tenté de considérer comme définitivement acquise. En revanche, on utilise la vélocité pour planifier le projet à l'échelle macroscopique de façon fiable et précise.

1. **Calcul de vélocité**

Une fois que tous les items de backlog de produit ont été estimés, on attribue un certain nombre d'items à réaliser aux sprints successifs. Ainsi, une fois un sprint terminé, on sait combien de points ont été réalisés et on définit alors la vélocité de l'équipe, c'est-à-dire le nombre de points qu'elle peut réaliser en un sprint[2].

En partant de cette vélocité et du total de points à réaliser, on peut déterminer le nombre de sprints qui seront nécessaires pour terminer le projet (ou la release en cours). L'intérêt, c'est qu'on a une vision de plus en plus fiable (retours d'expérience de sprint en sprint) de la date d'aboutissement du projet, tout en permettant d'aménager les items de backlog du produit en cours de route[2].

1. **Items de backlog de sprint**

Les items de backlog de sprint sont généralement exprimés en heures et ne doivent pas dépasser 2 journées de travail, sinon il convient de les décomposer en plusieurs items. Par abus de langage, on emploie le terme de tâches, les concepts étant très proches.

* + - 1. **Documentation de projet**

Scrum n'impose aucune documentation particulière pour les projets. Des documents sont implicitement produits (backlogs, burndown charts), mais ils ont une vocation avant tout utilitaire.

Produire de la documentation est souvent utile mais aussi souvent inutile. En plus, il faut la maintenir à jour, quelque chose qui est rarement fait sur place. Pour savoir s'il faut rédiger un document, on peut se poser une question très simple : Est-ce que ce document va m'être vraiment utile et tout de suite ?

Voici quelques exemples de documents utiles et dans quels cas[2] :

* diagrammes métiers (processus, objets, etc.), associé au backlog de produit : uniquement si le logique métier du client qui concerne l'application est vraiment complexe. Dans ce cas, l'équipe devrait produire ce document avec lui ;
* diagramme de séquence, associé à un item du backlog du produit : uniquement si la fonctionnalité aura une utilisation complexe, tant au niveau métier qu'applicatif ;
* diagrammes d'architecture du logiciel (classes, modules, composants, etc.), pour le projet : indispensable pour avoir toujours sous les yeux une vue de l'architecture et s'assurer ainsi qu'elle est de qualité ;
* les manuels utilisateur, à chaque sprint : les manuels sont produits à chaque sprint et pas en fin de projet. Utiliser des vidéos de démonstrations commentées est une solution efficace ;
* les FAQ pour le centre d'appel : des cas classiques où les utilisateurs ne vont pas comprendre un comportement métier. Cela permet de traiter un maximum de problèmes au niveau du centre d'appel, avant que cela n'arrive aux équipes de développement/maintenance.
  + 1. **UML**

Le plus grand avantage d’une méthode orientée objet est qu’elle permet de structurer un système sans centrer uniquement l’analyse sur les données sur les traitements mais peut se faire sur les deux à la fois. Une telle approche a pour but de modéliser les propriétés statiques et dynamiques de l’environnement du système. Elle met en correspondance le problème et la solution, en préservant la structure et le comportement du système analysé.

UML est un langage qui permet de représenter des modèles, mais il ne définit pas le processus d'élaboration de ces modèles. Ce langage possède les caractéristiques suivantes[2] :

* UML n’est pas une méthode ou un processus ;
* UML est fondé sur un méta modèle ;
* UML propose aussi une notation ;
* UML cadre l'analyse objet ;
* UML est un support de communication.

**Les points forts d’UML**

* UML est un langage formel et normalisé
* gain de précision ;
* gage de stabilité ;
* encourage l'utilisation d'outils ;
* UML est un support de communication performant
* Il cadre l'analyse.
* Il facilite la compréhension de représentations abstraites complexes.
* Son caractère polyvalent et sa souplesse en font un langage universel.

UML 2 contient 13 diagrammes mais nous n’en retiendront que 4 seulement dont :

* + - 1. **Les diagrammes de cas d’utilisation**

Les diagrammes de cas d’utilisation qui sont des visions orientées utilisateur, donc capturent les besoins des utilisateurs et scindent les fonctionnalités du système en unité cohérente.



Figure 3 : Formalisme du diagramme de cas d'utilisation

* **Acteur :** rôle joué par un utilisateur humain ou un autre système qui interagit directement avec le système étudié. Un acteur participe à au moins un cas d’utilisation.
* **Cas d’utilisation** (use case) : ensemble de séquences d’actions réalisées par le système produisant un résultat observable intéressant pour un acteur particulier. Collection de scénarios reliés par un objectif utilisateur commun.
* **Association** : utilisée dans ce type de diagramme pour relier les acteurs et les cas d’utilisation par une relation qui signifie simplement « participe à ».
* **Inclusion** : le cas d’utilisation de base en incorpore explicitement un autre, de façon obligatoire, à un endroit spécifié dans ses enchaînements.
* **Extension** : le cas d’utilisation de base en incorpore implicitement un autre, de façon optionnelle, à un endroit spécifié indirectement dans celui qui procède à l’extension
* **Généralisation** : les cas d’utilisation descendants héritent de la description de leur parent commun. Chacun d’entre eux peut néanmoins comprendre des relations spécifiques supplémentaires avec d’autres acteurs ou cas d’utilisation.
  + - 1. **Les diagrammes d’activité**

Les diagrammes d’activitéqui permettent de mettre l’accent sur les traitements. Ils sont très adaptés pour modéliser les processus métier et le déroulement des opérations (workflow).

début

activité

decision

synchronisation

fin

transition



Activite\_1



Decision\_1

Figure 4 : Formalisme du diagramme d’activités

* **Transition :** Le passage d'une activité vers une autre est matérialisé par une transition.

Les transitions sont déclenchées par la fin d'une activité et provoquent le début immédiat d'une autre (elles sont automatiques).

En théorie, tous les mécanismes dynamiques pourraient être décrits par un diagramme d'activités, mais seuls les mécanismes complexes ou intéressants méritent d'être représentés.

* [**Notation des transitions conditionnelles**](#_Toc19055780)

Le formalisme d’une transition conditionnelle est comme l’instruction si ..alors..si non… et il est représenté comme suit :



Figure 5 : Formalisme d’un diagramme d’activité avec transition conditionnelle

* [**Notation des transitions synchrones**](#_Toc19055781)

Il est possible de synchroniser les transitions à l'aide des "barres de synchronisation".

Une barre de synchronisation permet d'ouvrir et de fermer des branches parallèles au sein d'un flot d'exécution :

Les transitions qui partent d'une barre de synchronisation ont lieu en même temps.

On ne franchit une barre de synchronisation qu'après la réalisation de toutes les transitions qui s'y rattachent.

La figure ci-après représente les activités synchrones à l’aide des barres de synchronisations :



Figure 6 : Formalisme d’un diagramme d’activité avec une barre de synchronisation

* + - 1. **Les diagrammes de séquence**

Les diagrammes de séquence qui montrent comment les messages sont émis et reçus entre les lignes de vie des différents objets et leurs ordres chronologiques pour accomplir une tache spécifiée.



Figure 7 : Formalisme du diagramme de séquence

* **Ligne de vie :** représentation de l’existence d’un élément participant dans un diagramme de séquence. Cela peut être un acteur ou le système en modélisation d’exigences, des objets logiciels en conception préliminaire ou conception détaillée.
* **Message :** élément de communication unidirectionnel entre objets qui déclenche une activité dans l’objet destinataire. La réception d’un message provoque un événement dans l’objet récepteur.
* **La flèche pointillée**représente un retour au sens UML. Cela signifie que le message en question est le résultat direct du message précédent.
* **Spécification d’activation :** bande blanche qui représente une période d’activité sur une ligne de vie.
* **Message synchrone :** envoi de message pour lequel l’émetteur se bloque en attente du retour et qui est représenté par une flèche pleine.
* **Un message asynchrone**, au contraire, est représenté par une flèche ouverte.
* **Occurrence d’interaction :** une interaction peut faire référence explicitement à une autre interaction grâce à un cadre avec le mot-clé ref et indiquant le nom de l’autre interaction.

UML 2 a ajouté une nouvelle notation très utile : les cadres d’interaction. Chaque cadre possède un opérateur et peut être divisé en fragments.

Les principaux opérateurs sont :

* **loop :** boucle. Le fragment peut s’exécuter plusieurs fois, et la condition de garde explicite l’itération.
* **opt :** optionnel. Le fragment ne s’exécute que si la condition fournie est vraie.
* **alt :** fragments alternatifs. Seul le fragment possédant la condition vraie s’exécutera.
  + - 1. **Les diagrammes de classe**

Les diagrammes de classe qui sont des vues statiques du système en termes de classes et qui montrent aussi les associations entre elles.

* **Classe:** description abstraite d’un ensemble d’objets qui partagent les mêmes propriétés (attributs et associations) et comportements (opérations et états).
* **Attribut :** donnée déclarée au niveau d’une classe, éventuellement typée, à laquelle chacun des objets de cette classe donne une valeur. Un attribut peut posséder une multiplicité et une valeur initiale. Un attribut dérivé (« / ») est un attribut dont la valeur peut être déduite d’autres informations disponibles dans le modèle.
* **Opération :** élément de comportement des objets, défini de manière globale dans leur classe. Une opération peut déclarer des paramètres (eux-mêmes typés) ainsi qu’un type de retour.



Figure 8 : Les associations entre classes

* **Association :** relation sémantique durable entre deux classes, qui décrit un ensemble de liens entre instances. Une association est bidirectionnelle par défaut, sauf si l’on restreint sa navigabilité en ajoutant une flèche.
* **Rôle :** nom donné à une extrémité d’une association ; par extension, manière dont les instances d’une classe voient les instances d’une autre classe au travers d’une association.
* **Multiplicité :** le nombre d’objets (min..max) qui peuvent participer à une relation avec un autre objet dans le cadre d’une association. Multiplicités fréquentes :
* 0..1 = optionnel (mais pas multiple)
* 1 = exactement 1
* 0..\* = \* = quelconque
* 1..\* = au moins 1

## Conclusion

Après avoir mis le projet dans son cadre et mis en place une démarche de développement qui nous aidera tout au long du projet, nous pouvons ainsi entamer la prochaine étape, c'est-à-dire la phase d’analyse conceptuelle.

1. ANALYSE CONCEPTUELLE

**Introduction**

Dans le cycle de vie de notre projet, l’analyse conceptuelle représente unephase primordiale et déterminante pour produire une application de hautequalité car il permet de traduire les besoins fonctionnels et les contraintes issues du cahier des charges et de la spécification des exigences dans un langage plus professionnel et compréhensible par tous les individus intervenants dans la réalisation et l’utilisation de l’application.

* 1. **Indentification des fonctionnalités (Features)**

Nous allons donc, dans un premier temps définir les fonctionnalités. Cette première étape est à réaliser avec le client. C’est à lui de déterminé les fonctionnalités de l’application.

Dans le cadre de la création d’un site e-commerce, nous allons citer sur le tableau 2 ci-après les listes des fonctionnalités qui sont aux nombres de 14.

Tableau 2 : Liste des fonctionnalités

|  |  |  |
| --- | --- | --- |
| Nom | Rang | Description |
| L’inscription au site | 1 | L’utilisateur doit pouvoir s’inscrire au site |
| La gestion des utilisateurs | 2 | L’administrateur du système doit pouvoir gérer les utilisateurs inscrits dans le site |
| La gestion des abonnements | 3 | L’administrateur du système doit pouvoir gérer les quotas de téléchargement et aussi éditer les durées des abonnements |
| La souscription à un abonnement | 4 | L’utilisateur doit pouvoir effectuer un abonnement |
| La gestion des fichiers | 5 | L’administrateur doit pouvoir gérer les fichiers (images, vecteurs, vidéos) |
| L’achat de fichier à la volet | 6 | L’utilisateur doit pouvoir acheter un fichier directement |
| L’achat des fichiers utilisant le panier | 7 | L’utilisateur doit pouvoir acheter le(s) fichier(s) en utilisant le panier |
| La gestion du lightbox | 8 | L’utilisateur doit pouvoir gérer le lightbox |
| La gestion le compte de la société | 9 | L’administrateur doit pouvoir gérer le compte de la société |
| La gestion de compte de l’utilisateur | 10 | L’utilisateur doit pouvoir gérer son compte |
| La gestion de panier | 11 | L’utilisateur doit pouvoir gérer son panier |
| L’uploade de fichier | 12 | L’administrateur doit pouvoir uploader de(s) fichier(s) |
| La création de sous compte | 13 | L’utilisateur doit pouvoir créer des sous comptes |
| La mise à jour de l’abonnement | 14 | L’utilisateur doit pouvoir effectuer une mise à jour de son abonnement |

* 1. **Les user stories**

Une fois les *features* recensées, nous allons pouvoir créer des *user stories* pour chacune d’entre elle dont la liste est fournie dans le tableau 2. Cette phase s’effectue souvent avec le client et l’ensemble de l’équipe de développement. Lorsque l’état des lieux des *features* est effectué, le *product owner* à la possibilité d’accepter ou de refuser les différentes *user stories*.

Tableau 3 : Liste des user stories

|  |  |  |  |
| --- | --- | --- | --- |
| Nom | Description | Estimation | Fonctionnalités |
| S’inscrire au site | L’utilisateur doit pouvoir s’inscrire au site | 3 | L’inscription au site |
| Lister les utilisateurs | L’administrateur doit pouvoir lister les utilisateurs inscrits afin qu’il puisse les supprimer, modifier et ajouter un nouveau utilisateur | 2 | La gestion des Utilisateurs |
| Gérer les utilisateurs | L’administrateur doit pouvoir gérer les utilisateurs inscrits | 3 |
| Choisir le type d’abonnement | L’administrateur doit pouvoir choisir le type d’abonnement | 2 | La gestion des abonnements |
| Lister les abonnements disponibles en vente | L’administrateur doit pouvoir lister les abonnements les abonnements disponibles en vente | 1 |
| Gérer les abonnements | L’administrateur doit pouvoir gérer (ajouter, supprimer, modifier) les valeurs de quota de téléchargement, aussi, l’administrateur doit pouvoir éditer les durées des abonnements | 3 |
| Choisir le type d’abonnement | L’utilisateur doit pouvoir choisir le type d’abonnement | 1 | La souscription à un abonnement |
| Choisir les options d’abonnements | L’utilisateur doit pouvoir choisir les options d’abonnement | 3 |
| Choisir le moyen de paiement | L’utilisateur doit pouvoir choisir le moyen de paiement | 2 |
| Valider le paiement | L’utilisateur doit pouvoir valider le paiement | 3 |
| Choisir le type de fichier | L’administrateur doit pouvoir choisir le type de fichier | 1 | La gestion des fichiers |
| Lister les fichiers | L’administrateur doit pouvoir lister les fichiers existants | 2 |
| Gérer les fichiers | L’administrateur doit pouvoir gérer les fichiers en effectuant des mises à jour des informations du fichier ou une suppression du fichier, aussi il doit pouvoir ajouter des nouveaux fichiers en les uploadant | 5 |
| Lister les fichiers disponibles selon le type choisi | L’utilisateur doit pouvoir lister les fichiers disponibles selon le type choisi | 2 | L’achat de fichier à la volet |
| Voir le détail d’un fichier | L’utilisateur doit pouvoir consulter le détail d’un fichier | 2 |
| Choisir le mode de paiement | L’utilisateur doit pouvoir choisir le mode paiement | 2 |
| Lister le(s) fichier(s) dans le panier | L’utilisateur doit pouvoir choir le mode d’achat | L’achat de fichier direct | L’achat des fichiers utilisant le panier |
| Choisir le mode d’achat | L’utilisateur doit pouvoir choisir la (les) taille(s) du fichier(s) |  |
| Choisir la (les) taille(s) de(s) fichier(s) | L’utilisateur doit pouvoir lister les fichiers dans le lightbox |  |
| Lister les fichiers dans le lightbox | L’utilisateur doit pouvoir gérer les fichiers dans le lightbox |  | La gestion de lightbox |
| Gérer le lightbox | L’administrateur doit pouvoir consulter le compte de la société | 2 |
| Consulter le compte de la société | L’administrateur doit pouvoir gérer le compte de la société (mettre à jour des informations générales de la société, lire des nouveaux messages, répondre aux messages, mettre à jour les informations du compte paypal) | 3 | La gestion du compte de la société |
| Gérer le compte de la société | L’utilisateur doit pouvoir consulter son tableau de bord | 3 |
| Consulter le tableau de bord | L’utilisateur doit pouvoir gères ses compte (modifier le profil et le compte, consulter les nouveaux messages dans le boite de réception, voir les crédits et les abonnements disponibles) | 5 | La gestion du compte utilisateur |
| Gérer le tableau le compte utilisateur | L’utilisateur doit pouvoir lister le(s) fichier(s) dans le panier | 3 |
| Afficher le(s) fichier(s) dans le panier | L’utilisateur doit pouvoir supprimer le(s) fichier(s) dans le panier | 3 | La gestion du panier |
| Supprimer le(s) fichier(s) dans le panier | L’utilisateur doit pouvoir annuler la(les) sélection(s) du (des) fichiers | 3 |
| Annuler la (les) sélection(s) du (des) fichier(s) | L’utilisateur doit pouvoir annuler les sélections du (des) fichier(s) | 1 |
| Uploader de fichier | L’administrateur doit pouvoir uploader de fichier | 5 | L’uploade de fichier |
| Créer un (des) sous compte(s) | L’utilisateur doit pouvoir créer un (des) sous compte(s) | 3 | La création de sous compte |
| Choisir le type d’abonnement | L’utilisateur doit pouvoir choisir le type d’abonnement qu’il souhaite mettre à jour | 2 | La mise à jour de l’abonnement |
| Choisir les options de la mise à jour de l’abonnement | L’utilisateur doit pouvoir choisir les options de la mise à jour de l’abonnement | 3 |
| Choisir le moyen de paiement | L’utilisateur doit pouvoir choisir le moyen de paiement | 2 |
| Valider le paiement | L’utilisateur doit pouvoir valider le paiement | 3 |

Les User stories dans le *backlog* sont donc prêtes à être réparties dans les itérations.

* 1. **Planification des sprints**
     1. **Planification de l’itération 1**

Tableau 4 : Planification de l’itération 1

|  |  |
| --- | --- |
| Itération 1 : début le : 10 /08/2014 fin le : 20/08/2014  But  : L’inscription au site | |
| User Stories | Tâches |
| Inscription au site | Créer le formulaire d’inscription |
| Créer une fonction pour gérer l’inscription |

* + 1. **Planification de l’itération 2**

Tableau 5 : Planification de l’itération 2

|  |  |
| --- | --- |
| Itération2 : début le : 24 /08/2014 fin le : 18/09/2014  But  : Gestion des utilisateurs | |
| User Stories | Tâches |
| Lister les utilisateurs | Créer une page pour lister les utilisateurs |
| Créer une fonction pour afficher la page qui liste les utilisateurs |
| Gérer les utilisateurs | Créer un formulaire pour éditer l’utilisateur |
| Créer une fonction qui permet de valider l’édition de l’utilisateur |
| Créer une fonction qui permet de supprimer l’utilisateur |

* + 1. **Planification de l’itération 3**

Tableau 6 : Planification de l’itération 3

|  |  |
| --- | --- |
| Itération 3 : début le : 21 /09/2014 fin le : 05/10/2014  But  : Gestion des abonnements | |
| User Stories | Tâches |
| Choisir le type d’abonnement à gérer | Créer une page qui permet choisir le type d’abonnement |
| Créer une fonction permettant de choisir le type d’abonnement |
| Créer une page qui permet lister les options d’un type d’abonnement |
| Créer une fonction qui permet de lister les options d’un type d’abonnement |
| Gérer les abonnements | Créer un formulaire pour éditer le quota de téléchargement d’un type abonnement |
| Créer une fonction qui permet de valider l’édition d’un quota de téléchargement d’un type d’abonnement |
| Créer une fonction qui permet de supprimer un quota de téléchargement d’un type d’abonnement |
| Créer un formulaire qui permet l’ajout d’un quota de téléchargement d’un type d’abonnement |
| Créer une fonction qui permet l’ajout d’un quota de téléchargement d’un type d’abonnement |

* + 1. **Planification de l’itération 4**

Tableau 7 : Planification de l’itération 4

|  |  |
| --- | --- |
| Itération 4 : début le : 06 /10/2014 fin le : 20/10/2014  But  : Souscription à un abonnement | |
| User Stories | Tâches |
| Choisir le type d’abonnement | Créer une page qui permet choisir le type d’abonnement |
| Créer une fonction permettant de choisir le type d’abonnement |
| Créer une page qui permet lister les options d’un type d’abonnement |
| Créer une fonction qui permet de lister les options d’un type d’abonnement |
| Choisir les options d’abonnement | Créer un formulaire qui permet de choisir les options d’abonnement |
| Créer une fonction qui permet de choisir les options d’abonnement |
| Créer une fonction qui permet de valider l’achat |
| Choisir le moyen de paiement | Créer une page qui permet de lister les moyens de paiement existants |
| Créer une fonction qui permet de lister les moyens de paiement existants |
| Créer une fonction qui permet de valider le paiement |
| Valider le paiement | Créer une fonction qui permet de valider le paiement |
| Créer une fonction qui permet de Controller le retour de validation du paiement |

* + 1. **Planification de l’itération 5**

Tableau 8 : Planification de l’itération 5

|  |  |
| --- | --- |
| Itération 5 : début le : 21 /10/2014 fin le : 30/10/2014  But  : Gestion des fichiers | |
| User Stories | Tâches |
| Choisir le type de fichier | Créer un menu qui contient les différents type de fichier |
| Créer une fonction qui permet de lister les fichiers d’un type |
| Lister les fichiers | Créer une page qui permet de lister les fichiers |
| Créer une fonction qui permet de lister les fichiers |
| Gérer les fichiers | Créer une page qui permet de mettre à jour les informations d’un fichier |
| Créer une fonction qui permet de valider la mise à jour des informations d’un fichier |

* + 1. **Planification de l’itération 6**

Tableau 9 : Planification de l’itération 6

|  |  |
| --- | --- |
| Itération 6 : début le : 01 /11/2014 fin le : 10/11/2014  But  : Achat de fichier à la volet | |
| User Stories | Tâches |
| Lister les fichiers disponibles selon le type | Créer un page permettant de lister tous les fichiers disponibles selon ses types |
| Créer une fonction qui permet de lister les fichiers selon ses types |
| Détaillé un fichier | Créer une page qui permet d’afficher les détails d’un fichier |
| Créer une fonction qui permet d’obtenir les détails d’un fichier |
| Choisir le mode de paiement | Créer une fonction permettant de choisir le mode paiement |
| Créer une fonction permettant de valider l’achat direct |

* + 1. **Planification de l’itération 7**

Tableau 10 : Planification de l’itération 7

|  |  |
| --- | --- |
| Itération 7 : début le : 12 /11/2014 fin le : 30/11/2014  But  : Achat des fichiers utilisant le panier | |
| User Stories | Tâches |
| Lister le(s) fichier(s) dans le panier | Créer une page qui permet de lister les fichiers dans le panier |
| Créer une fonction qui permet de lister les fichiers dans le panier |
| Choisir le mode d’achat | Créer une page permettant de voir les disponibilités des fichiers selon le mode d’achat |
| Créer une fonction permettant de voir les disponibilités des fichiers selon le mode d’achat |
| Choisir la (les) taille(s) des fichiers | Créer une fonction permettant de choisir le(s) taille(s) du (des) fichier(s) |
| Créer une fonction permettant de valider l’achat du (des) fichier(s) |

* + 1. **Planification de l’itération 8**

Tableau 11 : Planification de l’itération 8

|  |  |
| --- | --- |
| Itération 8 : début le : 01 /12/2014 fin le : 08/12/2014  But  : Gestion de lightbox | |
| User Stories | Tâches |
| Lister les fichiers dans le lightbox | Créer une page qui permet de lister les fichiers dans le lightbox |
| Créer une fonction qui permet de lister les fichiers dans le lightbox |
| Gérer le lightbox | Créer une fonction qui permet d’ajouter le fichier dans le lightbox dans le panier |
| Créer une fonction qui permet de supprimer le fichier le fichier dans le lightbox |

* + 1. **Planification de l’itération 9**

Tableau 12 : Planification de l’itération 9

|  |  |
| --- | --- |
| Itération 9 : début le : 09 /12/2014 fin le : 16/12/2014  But  : Gestion du compte de la société | |
| User Stories | Tâches |
| Consulter le compte de la société | Créer une page qui permet de consulter le compte de la société |
| Créer une fonction qui permet d’avoir les informations du compte de la société |
| Gérer le compte de la société | Créer un formulaire qui permet de mettre à jour les informations générales de la société |
| Créer une fonction qui permet de valider les mises à jour des informations générales de la société |
| Créer une page permettant de lire le nouveau message |
| Créer une fonction permettant d’avoir les informations du nouveau message |
| Créer une fonction qui permet de supprimer un message |
| Créer un formulaire permettant de mettre à jour les informations du compte paypal de la société |
| Créer une fonction permettant de valider la mise à jour des informations du compte paypal de la société |

* + 1. **Planification de l’itération 10**

Tableau 13 : Planification de l’itération 10

|  |  |
| --- | --- |
| Itération10 : début le : 16 /12/2014 fin le : 27/12/2014  But  : Gestion du compte utilisateur | |
| User Stories | Tâches |
| Consulter le tableau de bord | Créer une page qui permet de consulter le tableau de bord de l’utilisateur |
| Créer une fonction qui permet d’avoir les informations nécessaires au tableau de bord d’un utilisateur |
| Gérer le compte utilisateur | Créer un formulaire qui permet d’effectuer la mise à jour du compte utilisateur |
| Créer une fonction qui permet de valider les mises à jour du compte utilisateur |
| Créer une page qui permet de lire le nouveau message |
| Créer une fonction qui permet d’avoir les informations du nouveau message |
| Créer un formulaire qui permet de mettre à jour le mot de passe utilisateur |
| Créer une fonction qui permet de valider la mise à jour du mot de passe utilisateur |

* + 1. **Planification de l’itération 11**

Tableau 14 : Planification de l’itération 11

|  |  |
| --- | --- |
| Itération 11 : début le : 28 /12/2014 fin le : 08/01/2015  But  : Gestion de panier | |
| User Stories | Tâches |
| Afficher le(s) fichier(s) dans le panier | Créer une page qui permet de lister les fichiers dans le panier |
| Créer une fonction qui permet de lister les fichiers dans le panier |
| Supprimer le fichier dans le panier | Créer une fonction permettant de supprimer un fichier dans le panier |
| Annuler la (les) sélection(s) de(s) fichier(s) | Créer une fonction permettant d’annuler la (les) sélection(s) de(s) fichier(s) dans le panier |

* + 1. **Planification de l’itération 12**

Tableau 15 : Planification de l’itération 12

|  |  |
| --- | --- |
| Itération 12 : début le : 9/01/2015 fin le : 20/02/2015  But  : Uploade des fichiers | |
| User Stories | Tâches |
| Uploader fichier | Créer une page qui permet de faire l’uploade d’un nouveau fichier |
| Créer une fonction qui permet d’uploader un nouveau fichier |
| Créer un formulaire qui permet de remplir les informations du fichier à uploader |
| Créer une fonction qui permet de valider l’uploade d’un fichier |

* + 1. **Planification de l’itération 13**

Tableau 16 : Planification de l’itération 13

|  |  |
| --- | --- |
| Itération 13 : début le : 21 /02/2015 fin le : 27/02/2015  But  : Création de sous compte | |
| User Stories | Tâches |
| Création de sous compte | Créer une page permettant de lister les sous comptes de l’utilisateur |
| Créer un formulaire permettant de remplir les informations du sous compte |
| Créer une fonction permettant de valider le formulaire d’informations du sous compte |

* + 1. **Planification de l’itération 14**

Tableau 17 : Planification de l’itération 14

|  |  |
| --- | --- |
| Itération 14 : début le : 28 /02/2015 fin le : 04/03/2015  But  : Mise à jour de l’abonnement | |
| User Stories | Tâches |
| Choisir le type d’abonnement | Créer une page qui permet choisir le type d’abonnement |
| Créer une fonction permettant de promouvoir le type d’abonnement disponible pour la mise à jour |
| Créer une page qui permet lister les options de mise à jour d’un type d’abonnement |
| Créer une fonction qui permet de lister les options d’un type d’abonnement |
| Choisir les options de mise à jour de l’abonnement | Créer un formulaire qui permet de choisir les options d’abonnement |
| Créer une fonction qui permet de choisir les options d’abonnement |
| Créer une fonction qui permet de valider l’achat |
| Choisir le moyen de paiement | Créer une page qui permet de lister les moyens de paiement existants |
| Créer une fonction qui permet de lister les moyens de paiement existants |
| Créer une fonction qui permet de valider le paiement |
| Valider le paiement | Créer une fonction qui permet de valider le paiement |
| Créer une fonction qui permet de Controller le retour de validation du paiement |

* 1. **Diagrammes des cas d’utilisation globale**
     1. **Identification des acteurs**

Un acteur est l’idéalisation d’un rôle joué par une personne externe, un processus ou une chose qui interagit avec un système. C’est-à-dire un acteur modélise :

* un rôle joué par un utilisateur du système ;
* ce qui existe en dehors du système et qui interagit avec le système.

**Le visiteur** : c’est un individu qui est entrain de fouiller sur le net, cherchant un produit pour l’acheter ou pour avoir une idée sur les modèleset les prix. Jusqu’au ce stade c’est un utilisateur inconnu donc il n’est pasencore un client.

**L’utilisateur** : cette acteur est un visiteur ayant déjà créé un comptesur notre site, il peut donc suivre le processus d’achat entoute sécurité sachant que notre système doit être l’unique responsable dela confidentialité des données personnelles de ses clients.

**Système :** c’est l’application client qui va s’interagir avec l’application serveur

**L’administrateur (admin)** : pour les sites web on l’appelle généralement« *le webmaster* ». C’est celui qui assure le dynamisme du site et veille surles mises à jour des produits, de leurs prix, de leurs disponibilités, de la gestion des payements.

* + 1. **Diagramme de cas d’utilisation globalede l’application**

Nous allons présenter dans la figure 9, le diagramme de cas d’utilisation globale de l’application :

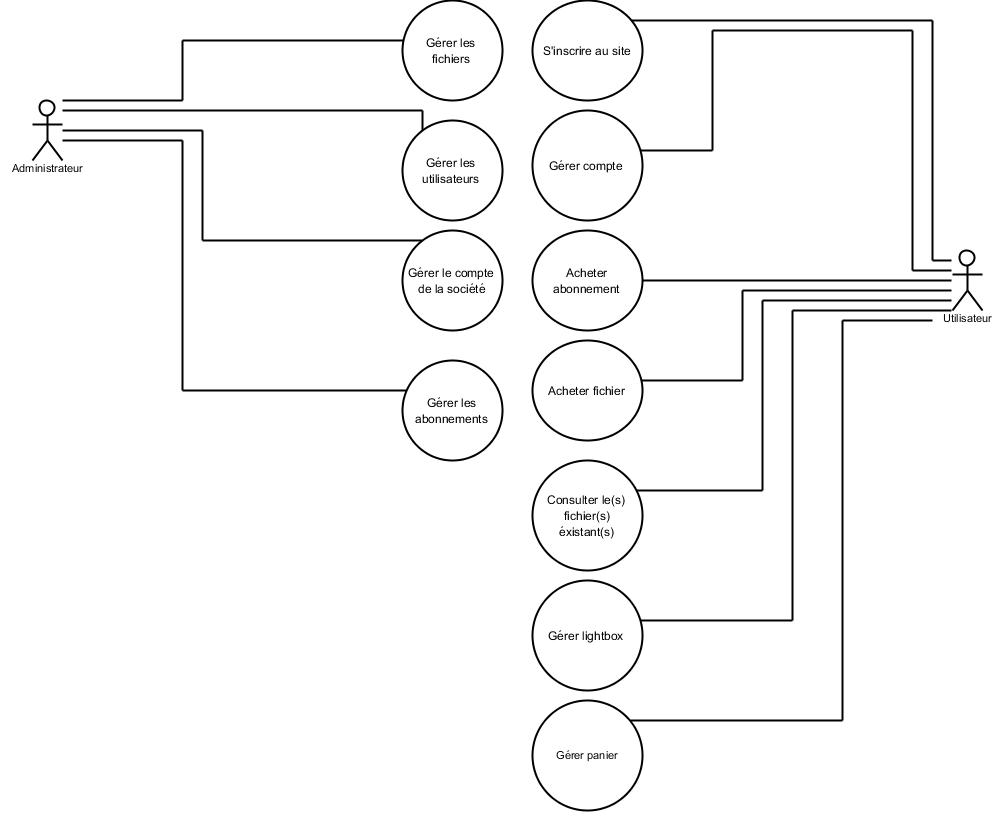


Figure 9 : Diagramme de cas d’utilisation globale de l’application

* 1. **Etablissement des diagrammes de classe globale utilisé dans les différents sprints**
     1. **Régles de gestion**

Voici donc les règles de gestion qui sont liées au système :

* RG 01 : Un administrateur peut gérer un ou plusieurs utilisateurs
* RG 02 : Un administrateur peut gérer un ou plusieurs fichiers
* RG 03 : Un utilisateur ne peut être gérer que par un seul administrateur
* RG 04 : Un fichier ne peut être gérer que par un seul administrateur
* RG 05 : Un compte ne peut être appartient que par un seul utilisateur
* RG 06 : Un utilisateur peut avoir un compte
* RG 07 : Un utilisateur peut consulter un ou plusieurs fichiers achetés
* RG 08 : Un utilisateur peut acheter un ou plusieurs fichiers
* RG 09 : Un fichier peut être acheté par plusieurs utilisateurs
* RG 10 : Un utilisateur peut effectuer un ou plusieurs achats d’abonnement
* RG 11 : Un abonnement peut être acheté par plusieurs utilisateurs
* RG 12 : Un utilisateur peut gérer un et un seul panier
* RG 13 : Un panier n’est peut être gérer que par un seul utilisateur
* RG 14 : Un utilisateur peut gérer un et un seul lightbox
* RG 15 : Un lightbox ne peut être gérer que par un seul utilisateur
* RG 16 : Un abonnement peut avoir un ou plusieurs quotas de téléchargement
* RG 17 : Un abonnement peut avoir un ou plusieurs durées
* RG 18 : Un administrateur peut gérer un et un seul compte société
* RG 19 : Un compte société peut avoir un ou plusieurs comptes paypal
* RG 20 : Un compte paypal ne peut être appartenir qu’a un seul compte société
  + 1. [**Dictionnaire des données**](#_Toc19055773)**de l’application**

Le tableau 16 présente le dictionnaire des données de l’application.

Tableau 18 : Dictionnaire de données de l’application

|  |  |  |
| --- | --- | --- |
| **Nom** | **Type de données** | **Description** |
| Adresse | Alphabétique | Adresse du compte de l’utilisateur |
| adresse\_societe | Alphabétique | Adresse de la société |
| code\_postal | Alphabétique | Code postal du compte de l’utilisateur |
| code\_postal\_societe | Alphabétique | Code postal de la société |
| date\_achat\_abnmt | Alphabétique | Date d’achat de l’abonnement |
| date\_achat\_fichier | Alphabétique | Date d’achat du fichier |
| date\_facturation | Alphabétique | Date de facturation |
| emailUtilisateur | Alphabétique | e-mail de l’utilisateur |
| etat\_fichier | Alphabétique | Etat du fichier |
| email\_societe | Alphabétique | e-mail de la société |
| email\_facturation | Alphabétique | e-mail de la facturation |
| etat\_societe | Alphabétique | Etat de la société |
| id\_abonnement | Numérique | identification de l’abonnement |
| id\_abnmt\_acheter | Numérique | Indentification de l’abonnement acheté |
| id\_compte | Numérique | Identification du compte |
| id\_compte\_paypal | Numérique | Indentification du compte paypal |
| id\_duree | Numérique | Identification de la durée |
| id\_facture | Numérique | Identification de la facture |
| id\_fichier | Numérique | Identification du fichier |
| id\_fichier\_acheter | Numérique | Identification du fichier acheté |
| id\_lightbox | Numérique | Identification du lightbox |
| id\_panier | Numérique | Identification du panier |
| id\_quota\_telechargement | Numérique | Indentification du quota de téléchargement |
| id\_societe | Numérique | Identification de la société |
| id\_utilisateur | Numérique | Identification de l’utilisateur |
| Loginutilisateur | Alphabétique | Login de l’utilisateur |
| mot\_clet\_fichier | Alphabétique | Mot clés du fichier |
| moyen\_paiement | Alphabétique | Moyen de paiement |
| Nom | Alphabétique | Nom de l’utilisateur |
| nom\_fichier | Alphabétique | Nom du fichier |
| nom\_societe | Alphabétique | Nom de la société |
| num\_tel | Alphabétique | Numéro téléphone |
| num\_tva | Alphabétique | Numéro TVA |
| Password | Alphabétique | Mot de passe de l’utilisateur |
| password\_admin | Alphabétique | Mot de passe de l’administrateur |
| paypal\_client\_id | Alphabétique | Identification du compte paypal |
| paypal\_client\_secret | Alphabétique | Mot de passe du compte paypal |
| paypal\_business\_mail | Alphabétique | Adresse e-mail du compte paypal |
| Pays | Alphabétique | Pays dans laquelle se trouve l’utilisateur |
| pays\_fichier | Alphabétique | Pays dans laquelle se trouve le fichier |
| Prenom | Alphabétique | Prénom de l’utilisateur |
| prix\_abnmt | Nuérique | Prix de l’abonnement |
| prix\_achat\_abnmt | Numérique | Prix d’achat de l’abonnement |
| prix\_facture | Numérique | Prix de la facturation |
| Profilutilisateur | Alphabétique | Profil de l’utilisateur |
| Qte | Numerique | Quantité de l’achat effectué |
| Region | Alphabétique | Région ou se situe l’utilisateur |
| Service | Alphabétique | Le type de service d’achat pour la facturation |
| taille\_fichier | Numérique | Tailles du fichier |
| Temps | Alphabétique | Temps de la facturation |
| tmp\_abnmt | Alphabétique | Temps de l’achat d’abonnement |
| tel\_societe | Alphabétique | Numéro de téléphone de la société |
| titre\_fichier | Alphabétique | Titre du fichier |
| tva\_societe | Alphabétique | Numéro TVA da la société |
| type\_compte | Alphabétique | Type du compte de l’utilisateur |
| type\_fichier | Alphabétique | Type du fichier |
| type\_transaction | Alphabétique | Type de la transaction |
| valeur\_quota\_telechargement | Numérique | Valeur du quota de téléchargement |
| ville\_societe | Alphabétique | Ville ou se situe la société |

* + 1. **Diagramme de classe globale de l’application**

Nous allons représenter dans la figure 10, le diagramme de classe globale de l’application:

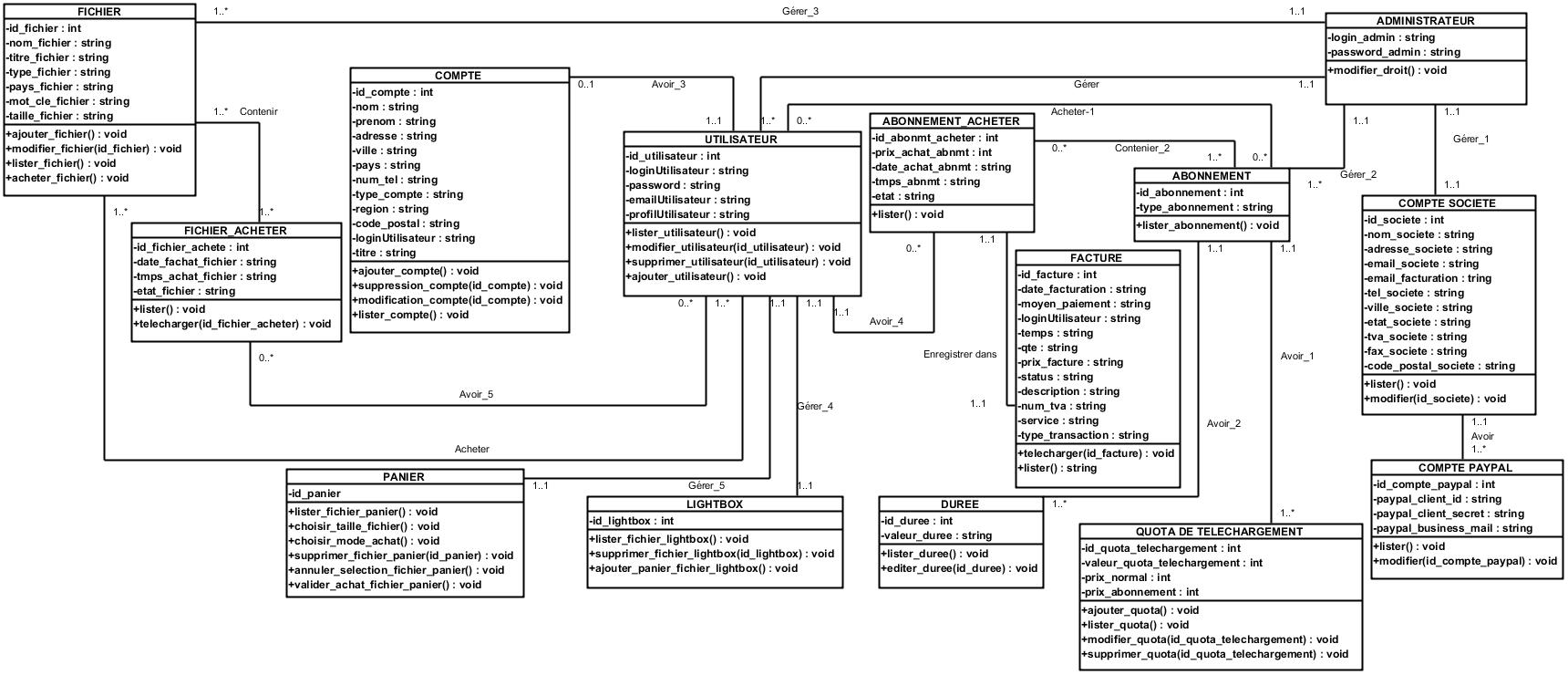


Figure 10 : Diagramme de classe globale de l’application

* + 1. **Liste des Tables sur l’application serveur**

Table « compte »

Table « utilisateur »

Table « quota\_telechargement »

Table « fichier »

Table « lightbox »

Table « panier »

Table « duree »

Table « facture »

Table  « abonnement »

Table « abonnement\_achete »

Table « fichier\_achete »

Table « compte\_paypal »

Table « compte\_societe »

* 1. **Conception détaillée par sprint**
     1. **Itération 1 «L’inscription au site»**
        1. **Diagramme de cas d’utilisation «Inscription»**

La figure 11 représente le diagramme de cas d’utilisation de « Inscription ».

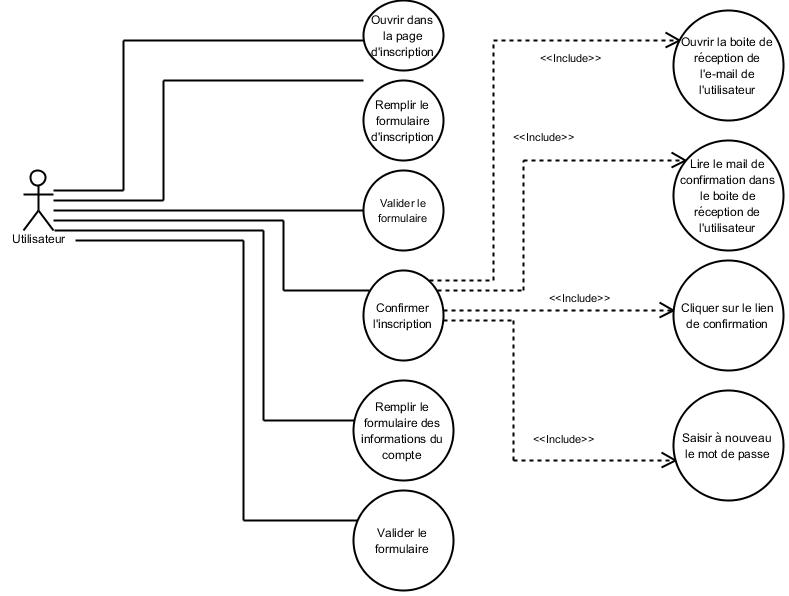


Figure 11 : Diagramme de cas d’utilisation « Inscription»

**Description textuelle**

* Cas d’utilisation **: Inscription au site**
* Acteur primaire: Utilisateur
* Description : après avoir rempli le formulaire d’inscription, l’utilisateur devra ouvrir la boite de réception de l’e-mail qu’il avait saisi sur le formulaire et de lire le message de confirmation envoyé par le site. Ensuite l’utilisateur doit cliquer sur le lien de confirmation de l’inscription, ensuite après avoir rediriger par le lien sur le site l’utilisateur doit saisir à nouveau son mot de passe. Ensuite, l’utilisateur doit remplir le formulaire qui indique les compléments des informations du compte et le valider.
* **Analyse**

Une réalisation de ce cas d’utilisation se fait comme suit :

* L’utilisateur :
  + - accède à l’application ;
    - remplit et valide le formulaire d’inscription ;
    - confirme l’inscription en cliquant sur le lien de confirmation qui se trouve dans la boite de réception de l’adresse e-mail qu’il avait saisi sur le formulaire d’inscription ;
    - Saisi de nouveau le mot de passe ;
    - Remplit et valide le formulaire de complément des informations du compte.
      1. **Diagramme d’activité «Inscription»**

La figure montre le diagramme d’activité de « Inscription ».

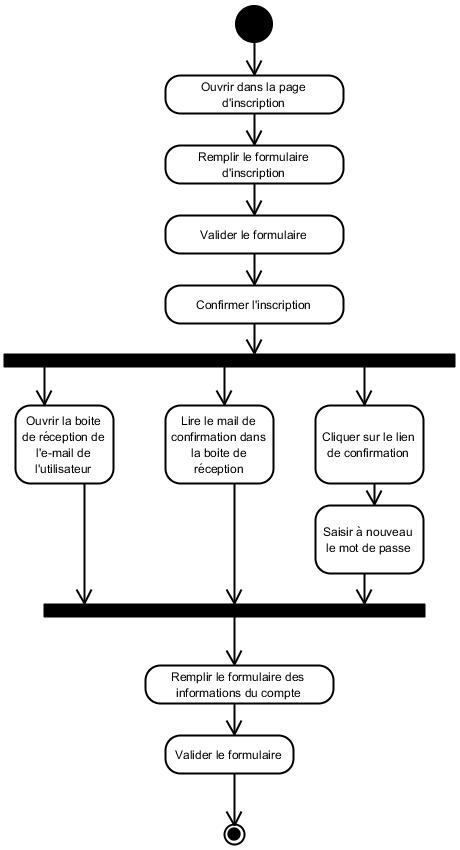


Figure 12 : Diagramme d’activité « Inscription»

* + - 1. **Diagramme de séquence «Inscription»**

Ce diagramme montre les étapes à suivre lorsque les visiteurs du site veulent s’inscrire:

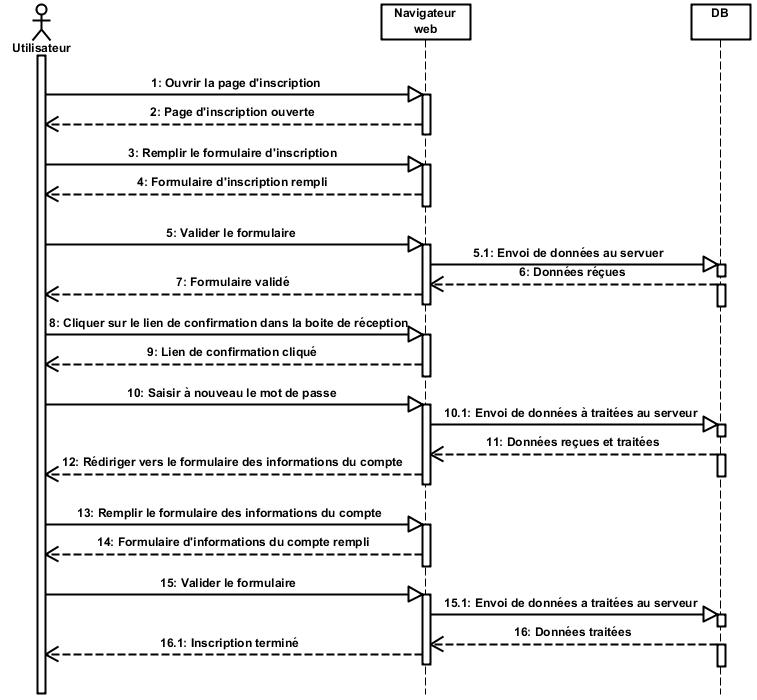


Figure 13 : Diagramme de séquence « Inscription»

* + 1. **Itération 2 «La gestion des utilisateurs»**
       1. **Diagramme de cas d’utilisation «Gérer utilisateur»**

La figure 14 représente le diagramme de cas d’utilisation de la gestion des utilisateurs.

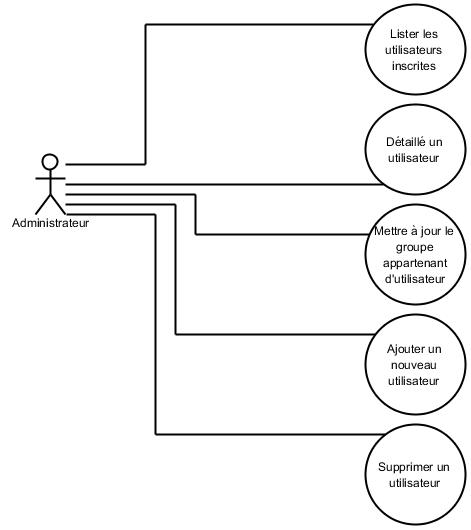


Figure 14 : Diagramme de cas d’utilisation « Gérer utilisateur»

**Description textuelle**

* Cas d’utilisation **: Gestion des utilisateurs**
* Acteur primaire: Administrateur
* Description : après l’authentification, l’administrateur peut gérer les utilisateurs inscrits, modifier ses droits, supprimer et même ajouter d’autres utilisateurs.
* **Analyse**

Une réalisation de ce cas d’utilisation se fait comme suit :

* L’administrateur liste les utilisateurs inscrits, sélectionne un utilisateur pour le modifier ou le supprimer ;
* L’administrateur ajoute un autre utilisateur en remplissant les formules d’inscriptions de l’application.
  + - 1. **Diagramme d’activité «Gérer utilisateur»**

La figure 15 montre le diagramme d’activité de la gestion des utilisateurs.

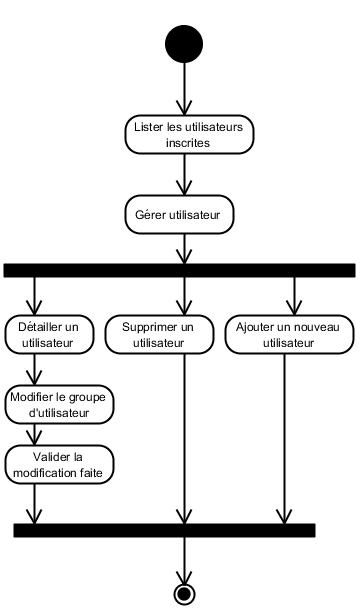


Figure 15 : Diagrammed’activité « Gérer utilisateur»

* + - 1. **Diagramme de séquence «Gérer utilisateur»**

La figure 16 représente le diagramme de séquence de la gestion des utilisateurs.

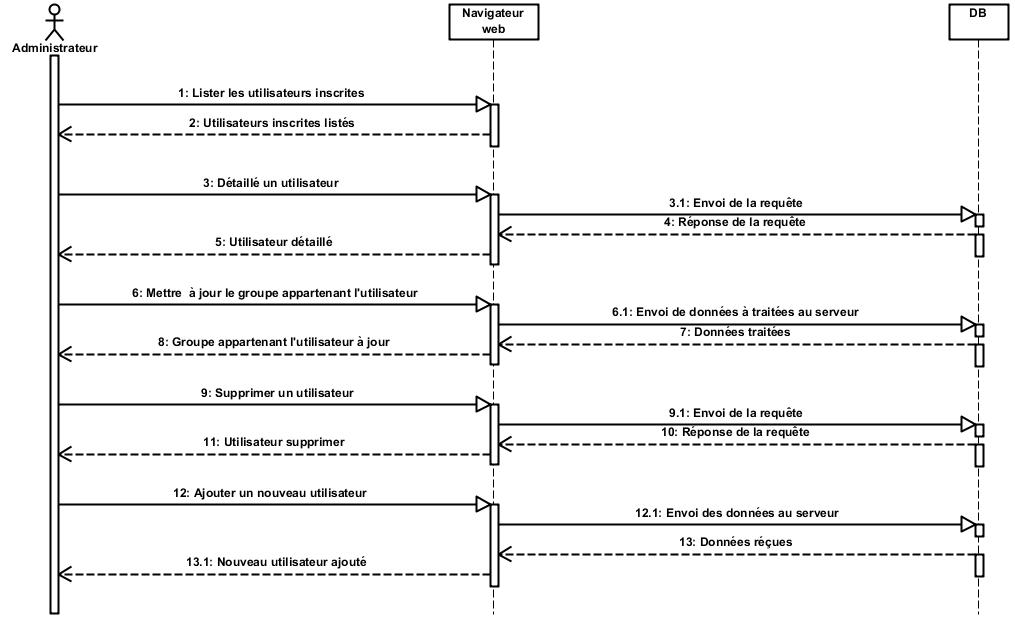


Figure 16 : Diagramme de séquence « Gérer utilisateur»

* + 1. **Itération 3 «La gestion des abonnements»**
       1. **Diagramme de cas d’utilisation «Gérer abonnement»**

La figure 17 représente le diagramme de cas d’utilisation de la gestion des abonnements.

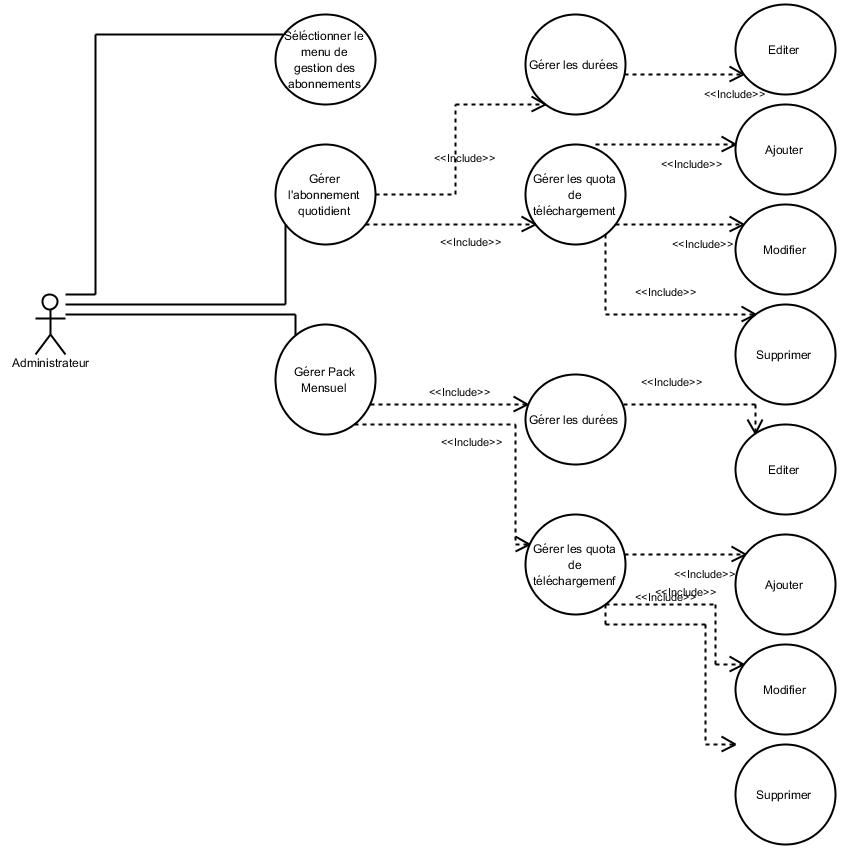


Figure 17 : Diagramme de cas d’utilisation « Gérer abonnement»

**Description textuelle**

* Cas d’utilisation **: Gestion des abonnements**
* Acteur primaire: Administrateur
* Description : après l’authentification, l’administrateur peut gérer les options d’abonnements, modifier, supprimer les valeurs de quota de téléchargement, éditer les valeurs des durées, aussi il peut ajouter un nouveau quota de téléchargement.
* **Analyse**

Une réalisation de ce cas d’utilisation se fait comme suit :

* L’administrateur liste les abonnements disponibles sur l’application;
* L’administrateur ajoute un nouveau quota de téléchargement selon le type d’abonnement choisi;
* L’administrateur modifie, supprime le quota de téléchargement selon le type d’abonnement choisi ;
* L’administrateur édite les valeurs de durées selon le type d’abonnement choisi.
  + - 1. **Diagramme d’activité «Gérer abonnement»**

La figure 18 montre le diagramme d’activité de la gestion d’abonnement.

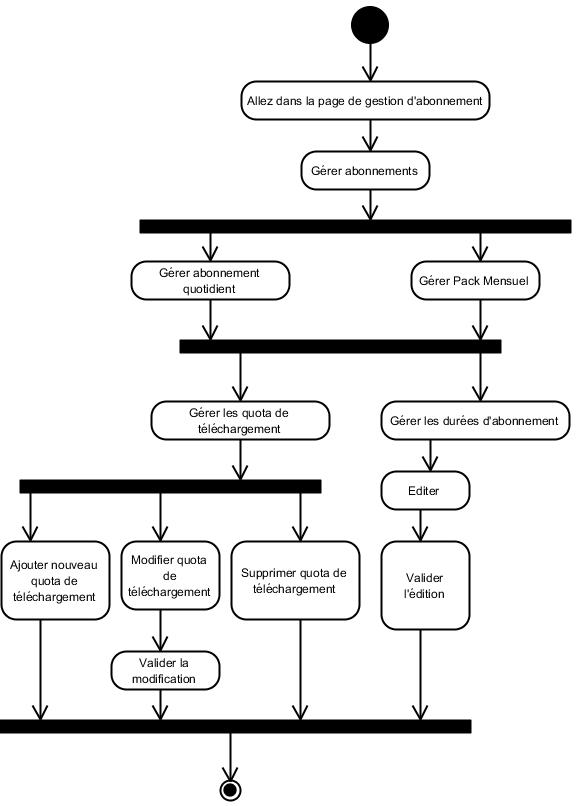


Figure 18 : Diagramme d’activité « Gérer abonnement»

* + - 1. **Diagramme de séquence «Gérer abonnement»**

La figure 19 montre le diagramme de séquence de la gestion d’abonnement.

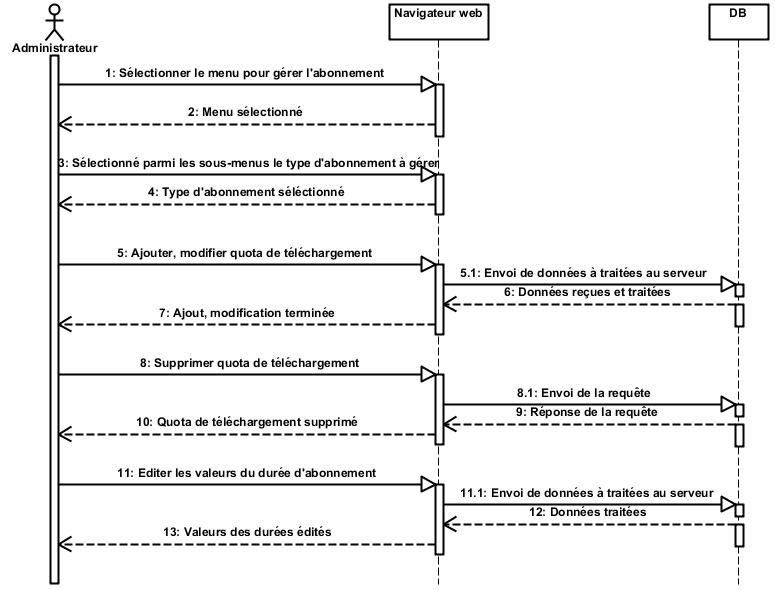


Figure 19 : Diagramme de séquence « Gérer abonnement»

* + 1. **Itération 4 «La souscriptionà un abonnement»**
       1. **Diagramme de cas d’utilisation «Souscription à un abonnement»**

La figure 20 représente le diagramme de cas d’utilisation de « Souscription à un abonnement ».

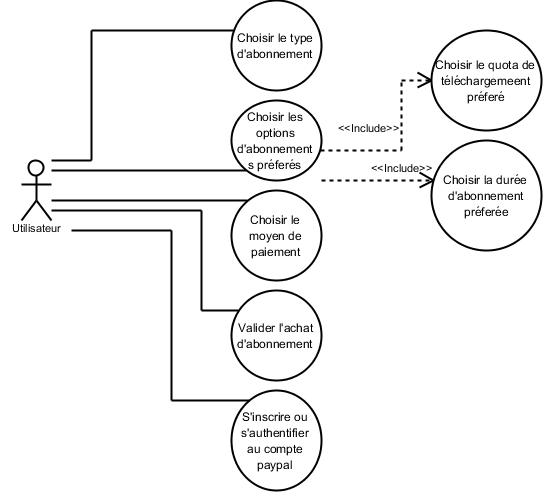


Figure 20 : Diagramme de cas d’utilisation« Souscription à un abonnement »

**Description textuelle**

* Cas d’utilisation **: Souscrire à un abonnement**
* Acteur primaire: Utilisateur
* Description : après l’authentification, l’utilisateur peut effectuer l’achat d’abonnement en choisissant le type et les options d’abonnements qu’il préféré. Après validation des options choisies, l’utilisateur doit choisir son moyen de payement (paypal) et valide à son tour. Après la validation du moyen de paiement, l’utilisateur doit s’authentifié ou s’inscrire au compte paypal et confirme ensuite le paiement.
* **Analyse**

Une réalisation de ce cas d’utilisation se fait comme suit :

* L’utilisateur choisi le type et les options d’abonnement qu’il préfère ;
* L’utilisateur valide les options choisies ;
* L’utilisateur choisi et valide le moyen de paiement (paypal) ;
* L’utilisateur s’inscrit ou s’authentifie au compte paypal ;
* L’utilisateur confirme son paiement.
  + - 1. **Diagramme d’activité «Acheter abonnement»**

La figure 21 montre le diagramme d’activité « Souscription à un abonnement ».

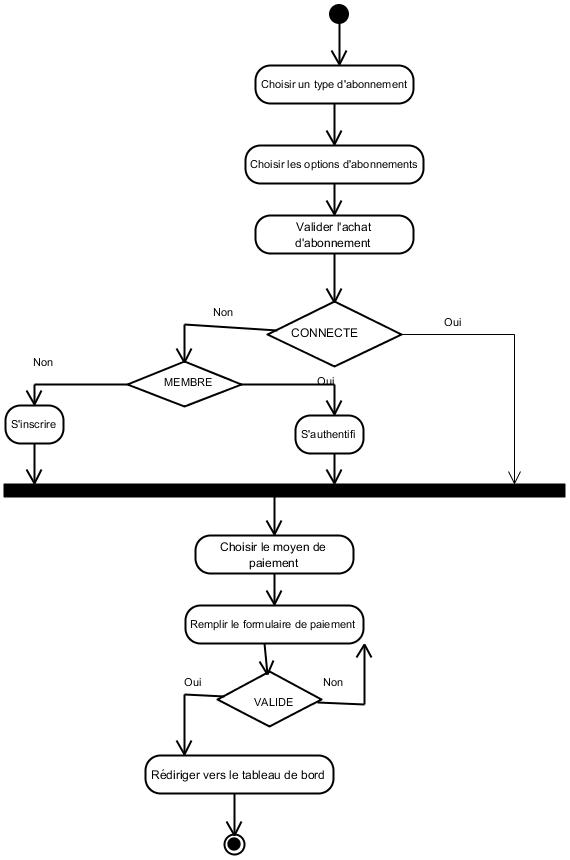


Figure 21 : Diagramme d’activité« Souscription à un abonnement »

* + - 1. **Diagramme de séquence «Souscrire à un abonnement»**

La figure 22 représente le diagramme de séquence « Souscription à un abonnement ».

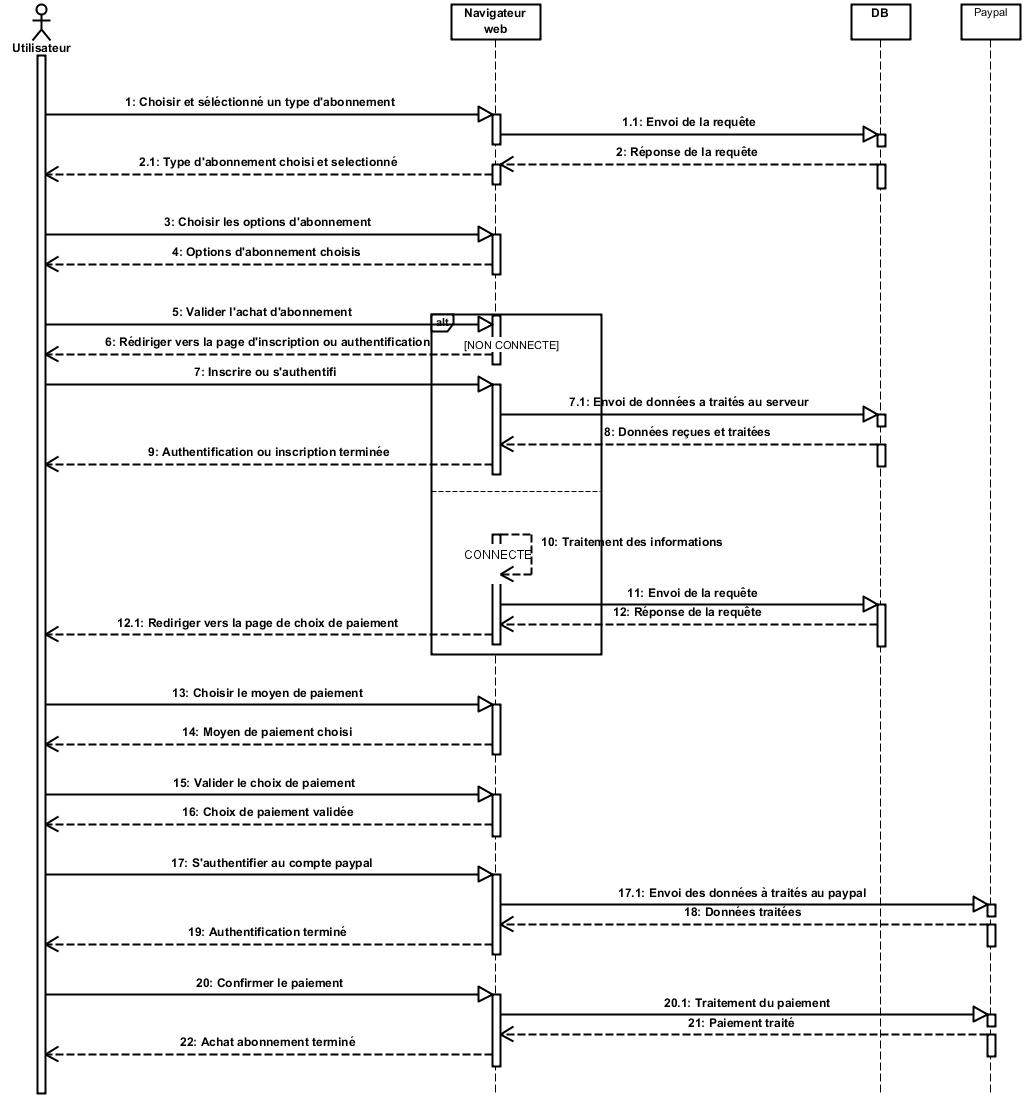


Figure 22 : Diagramme de séquence« Souscription à un abonnement »

* + 1. **Itération 5 «La gestion des fichiers»**
       1. **Diagramme de cas d’utilisation «Gérer fichier»**

La figure 23 représente le diagramme de cas d’utilisation de « Gérer fichier »

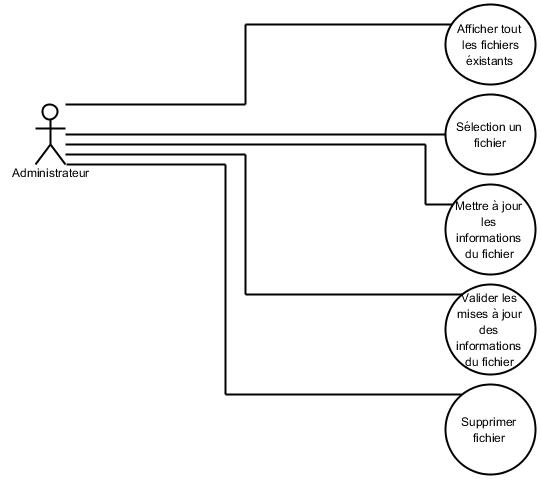


Figure 23 : Diagramme de cas d’utilisation« Gérer fichier»

**Description textuelle**

* Cas d’utilisation **: Gestion de fichier**
* Acteur primaire: Administrateur
* Description : après l’authentification, l’administrateur peut ouvrir la page de gestion de fichier, il peut mettre à jour les informations du (des) fichier(s) et il peut aussi supprimer quelques fichiers selon son choix.
* **Analyse**

Une réalisation de ce cas d’utilisation se fait comme suit :

* L’administrateuraffiche tous les fichiers existants ;
* L’administrateur sélectionne un fichier pour voir ses détails et ses informations ;
* L’administrateur met à jour les informations du fichier ;
* L’administrateur supprime le fichier selon son choix ;
  + - 1. **Diagramme d’activité «Gérer fichier»**

La figure 24 montre le diagramme d’activité de « Gérer fichier ».

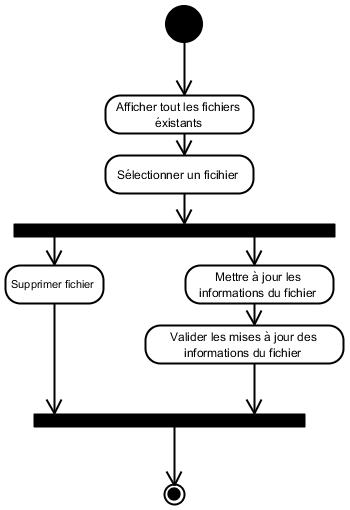


Figure 24 : Diagramme d’activité« Gérer fichier»

* + - 1. **Diagramme de séquence «Gérer fichier»**

La figure 25 représente le diagramme de séquence de « Gérer fichier ».

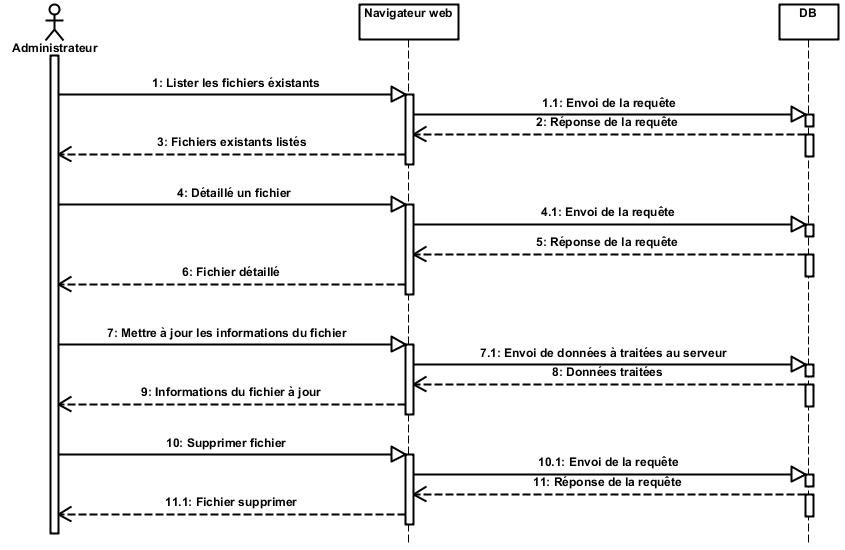


Figure 25 : Diagramme de séquence« Gérer fichier»

* + 1. **Itération 6 «L’achat de fichier à la volet»**
       1. **Diagramme de cas d’utilisation «Acheter fichier à la volet»**

La figure 26 représente le diagramme de cas d’utilisation de « Acheter fichier à la volet »

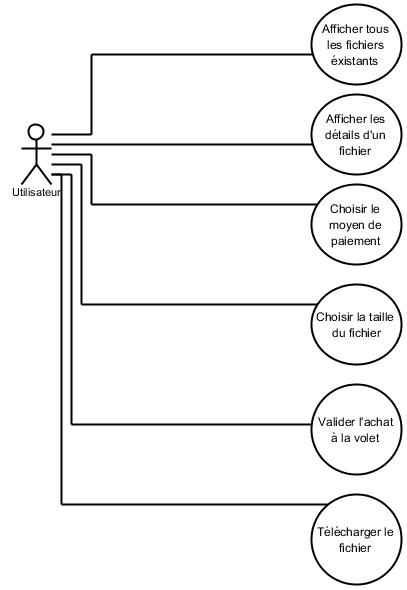


Figure 26 : Diagramme de cas d’utilisation« Acheter fichier à la volet»

**Description textuelle**

* Cas d’utilisation **: Achat fichier à la volet**
* Acteur primaire: Utilisateur
* Description : après l’authentification, l’utilisateur peut effectuer l’achat de fichier à la volet c’est-à-dire acheter du fichier sans utilisant le panier mais. Après la sélection d’un fichier, l’utilisateur choisi le moyen de paiement ainsi que la taille du fichier qu’il préfère, ensuite il l’achète directement. En fin l’utilisateur peut télécharger le fichier.
* **Analyse**

Une réalisation de ce cas d’utilisation se fait comme suit :

* L’utilisateur choisi et sélectionne un fichier selon son choix sur les différents fichiers existants au site ;
* L’utilisateur choisi la taille et le moyen de paiement qu’il préfère ;
* L’utilisateur valide l’achat ;
* L’utilisateur télécharge le fichier
  + - 1. **Diagramme d’activité «Acheter fichier à la volet»**

La figure 27 représente le diagramme d’activité de « Acheter fichier à la volet ».

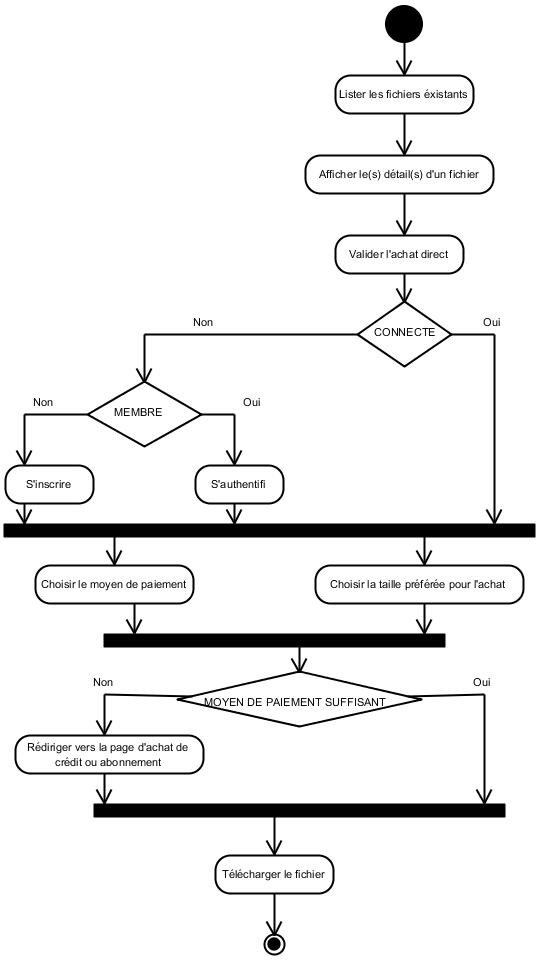


Figure 27 : Diagramme d’activité« Acheter fichier à la volet»

* + - 1. **Diagramme de séquence «Acheter fichier à la volet»**

La figure 28 montre le diagramme de séquence de « Acheter fichier à la volet ».

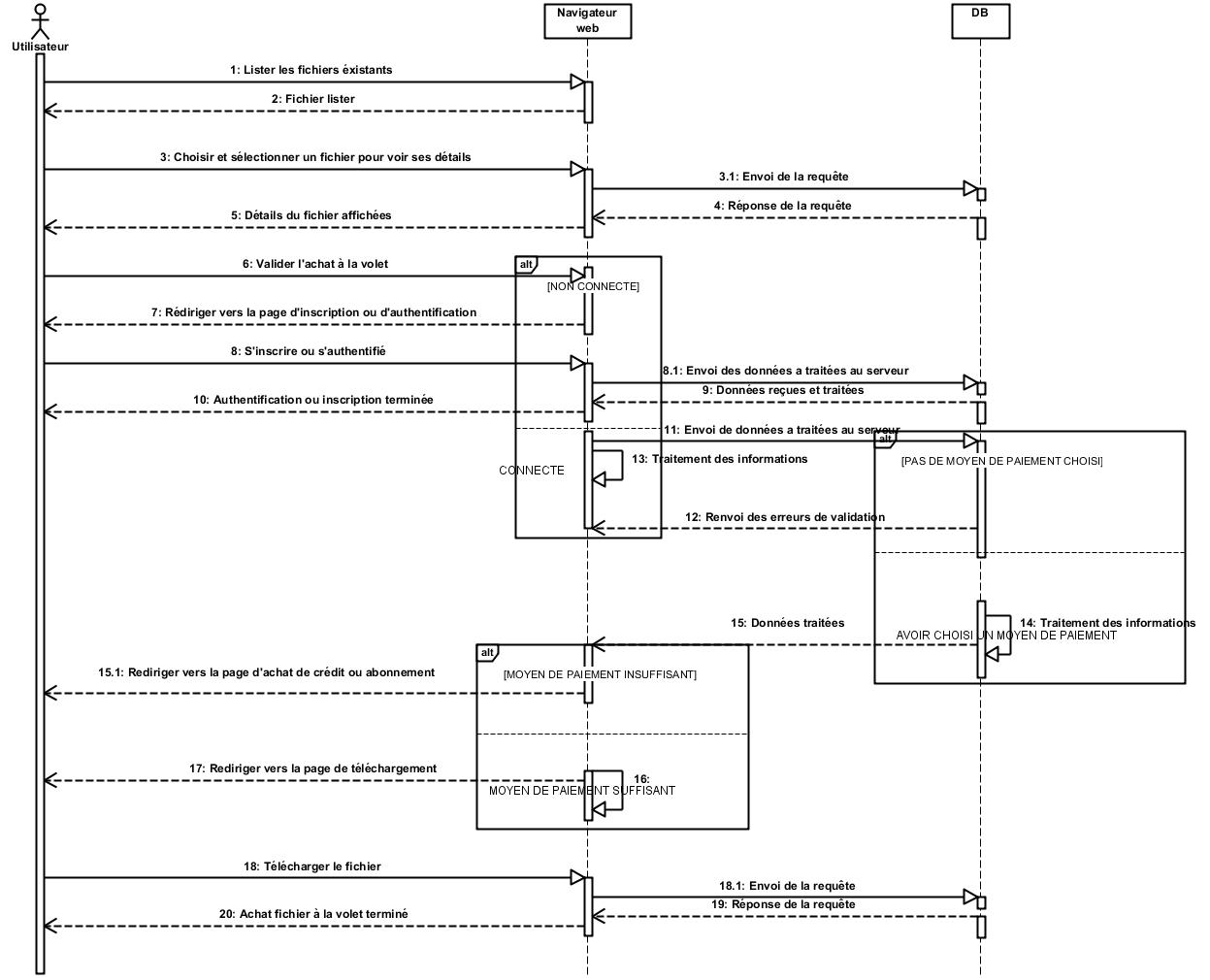


Figure 28 : Diagramme de séquence« Acheter fichier à la volet»

* + 1. **Itération 7 «L’achat de fichier utilisant le panier»**
       1. **Diagramme de cas d’utilisation «Acheter du fichier en utilisant le panier»**

La figure 29 représente le diagramme de cas d’utilisation de « Acheter fichier en utilisant le panier ».



Figure 29 : Diagramme de cas d’utilisation« Acheter fichier en utilisant le panier»

**Description textuelle**

* Cas d’utilisation **: L’achat de fichier en utilisant le panier**
* Acteur primaire: Utilisateur
* Description : après l’authentification, l’utilisateur peut effectuer l’achat de fichier en utilisant le panier. L’utilisateur peut ajouté des fichiers dans son panier. Après que l’utilisateur liste les fichiers qui se trouvent dans son panier, il peut choisi le mode d’achat (par abonnement ou par crédit). Ensuite il sélectionne les tailles des fichiers qui sont proportionnels au prix. En fin, après validation de l’achat, l’utilisateur peut télécharger le fichier.
* **Analyse**

Une réalisation de ce cas d’utilisation se fait comme suit :

* L’utilisateur ajoute des fichiers dans son panier ;
* L’utilisateur Liste les fichiers qui se trouvent dans son panier ;
* L’utilisateur choisi le mode d’achat ;
* L’utilisateur sélectionne les tailles préférés qui sont proportionnels au prix ;
* L’utilisateur valide l’achat ;
* L’utilisateur télécharge les fichiers.
  + - 1. **Diagramme d’activité «Acheter du fichier en utilisant le panier»**

La figure 30montre le diagramme d’activité « Acheter fichier en utilisant le panier ».

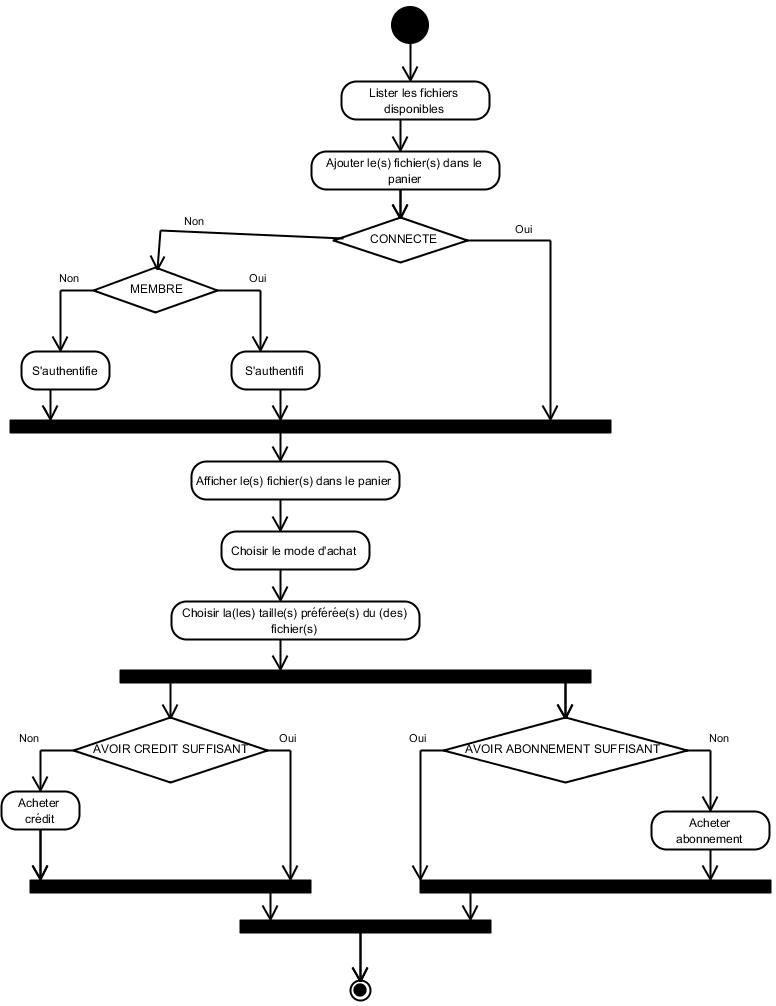


Figure : Diagramme d’activité« Acheter fichier en utilisant le panier»

* + - 1. **Diagramme de séquence «Acheter du fichier en utilisant le panier»**

La figure 31 représente le diagramme de séquence de « Acheter fichier en utilisant le panier ».

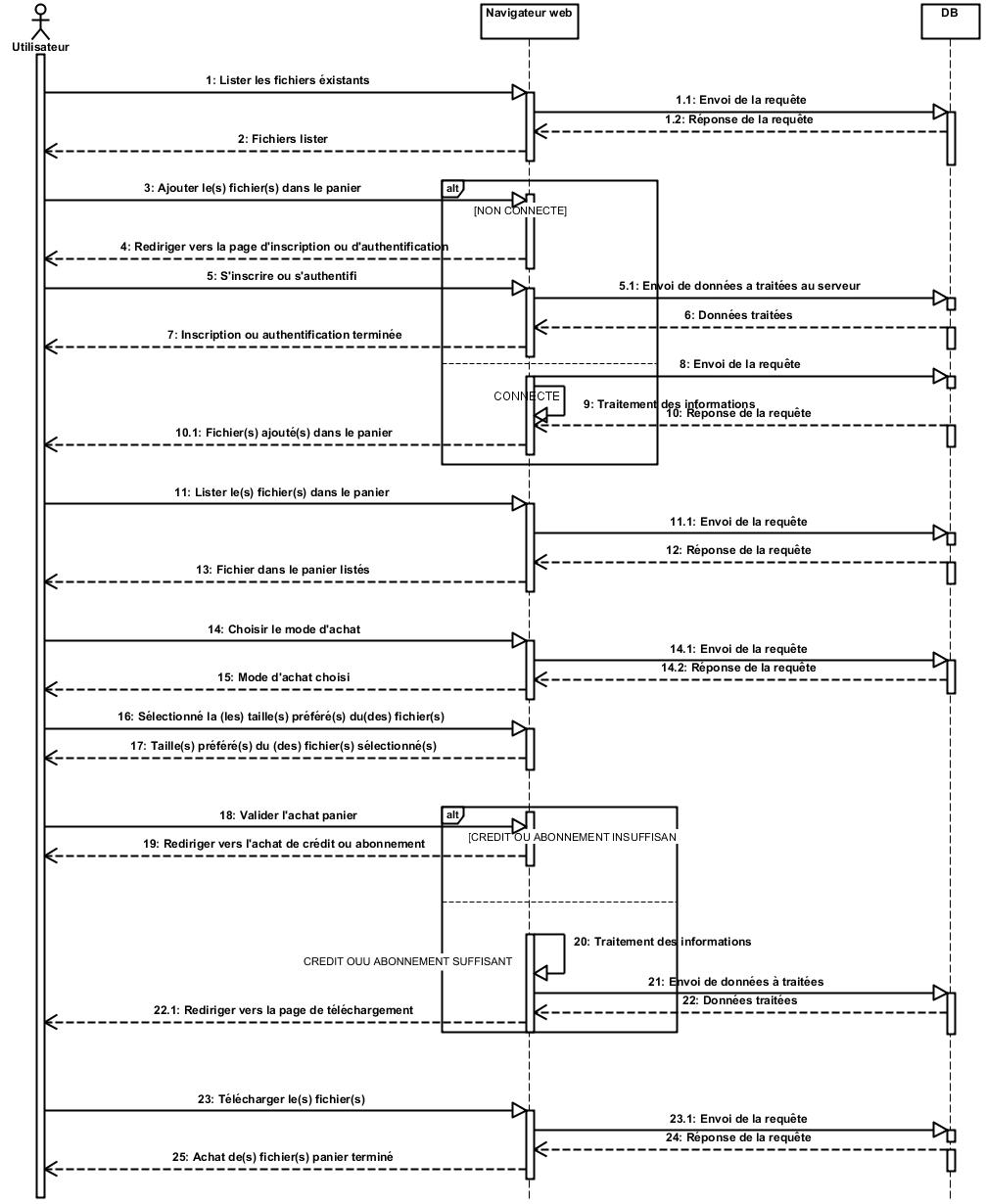


Figure : Diagramme de séquence« Acheter fichier en utilisant le panier»

* + 1. **Itération 8 «La gestion du lightbox»**
       1. **Diagramme de cas d’utilisation «Gérer lightbox»**

La figure 32 représente le diagramme de cas d’utilisation de « Gérer lightbox ».

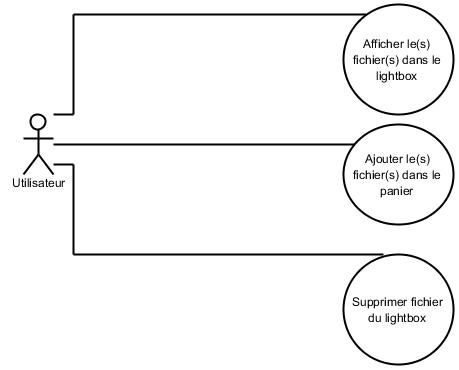


Figure 32 : Diagramme de cas d’utilisation« Gérer lightbox »

**Description textuelle**

* Cas d’utilisation **: La gestion du lightbox**
* Acteur primaire: Utilisateur
* Description : après l’authentification, l’utilisateur peut gérer son lightbox. Après avoir listé les fichiers que l’utilisateur a mis dans son lightbox, l’utilisateur peut ajouter ces fichiers dans son panier ou même il peut les supprimés.
* **Analyse**

Une réalisation de ce cas d’utilisation se fait comme suit :

* L’utilisateur ajoute des fichiers dans son lightbox ;
* L’utilisateur liste les fichiers qui se trouvent dans son lightbox ;
* L’utilisateur ajoute les fichiers du lightbox dans le panier ;
* L’utilisateur supprime les fichiers.
  + - 1. **Diagramme d’activité «Gérer lightbox»**

La figure 33 montre le diagramme d’activité « Gérer lightbox ».

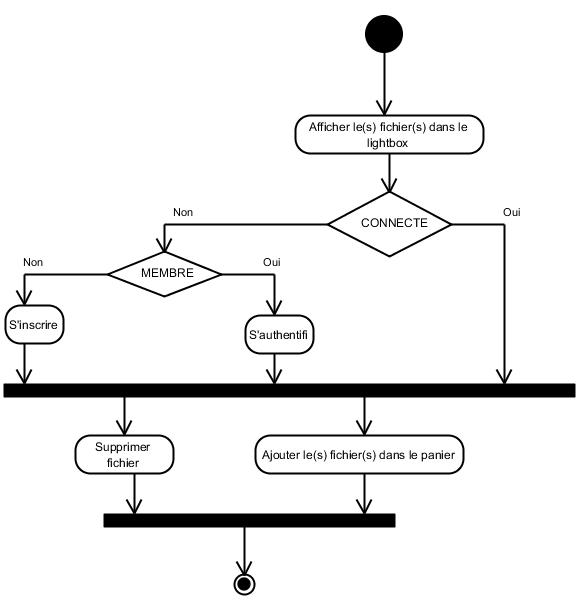


Figure 33 : Diagramme d’activité« Gérer lightbox»

* + - 1. **Diagramme de séquence «Gérer lightbox»**

La figure 34 montre le diagramme de séquence « Gérer lightbox ».

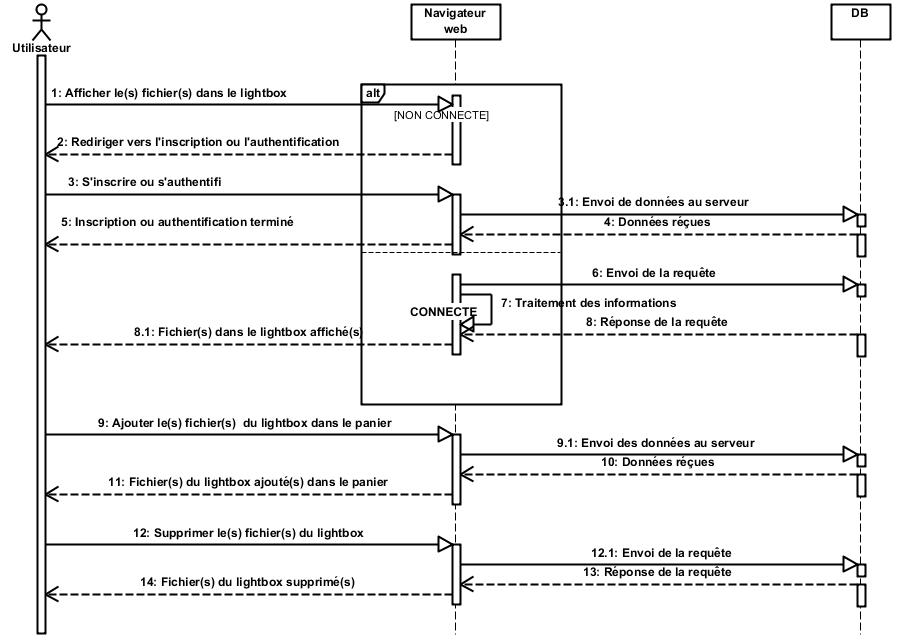


Figure 34 : Diagramme de séquence« Gérer lightbox»

* + 1. **Itération 9 «La gestion du compte de la société»**
       1. **Diagramme de cas d’utilisation «Gérer le compte de la société»**

La figure 35 représente le diagramme de cas d’utilisation de « Gérer compte de la société ».

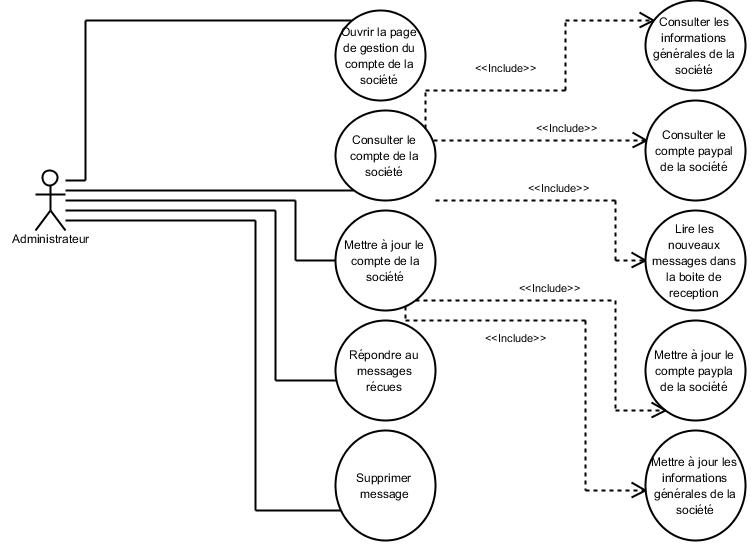


Figure 35 : Diagramme de cas d’utilisation« Gérer lightbox »

**Description textuelle**

* Cas d’utilisation **: La gestion du compte de la société**
* Acteur primaire: Administrateur
* Description : après l’authentification, l’administrateur peut gérer le compte de la société. Il peut modifier ou en d’autres termes mettre à jour les informations générales et le compte paypal de la société. Après avoir lu les nouveaux messages, il peut les répondre ou les supprimés.
* **Analyse**

Une réalisation de ce cas d’utilisation se fait comme suit :

* L’administrateur consulte les informations générales et le compte paypal de la société ;
* L’administrateur met à jour les informations générales et le compte paypal de la société ;
* L’administrateur lit les nouveau messages dans la boite de réception de la société ;
* L’administrateur réponde ou supprime les messages dans la boite de réception.
  + - 1. **Diagramme d’activité «Gérer le compte de la société»**

La figure 36 montre le diagramme d’activité « Gérer compte de la société ».

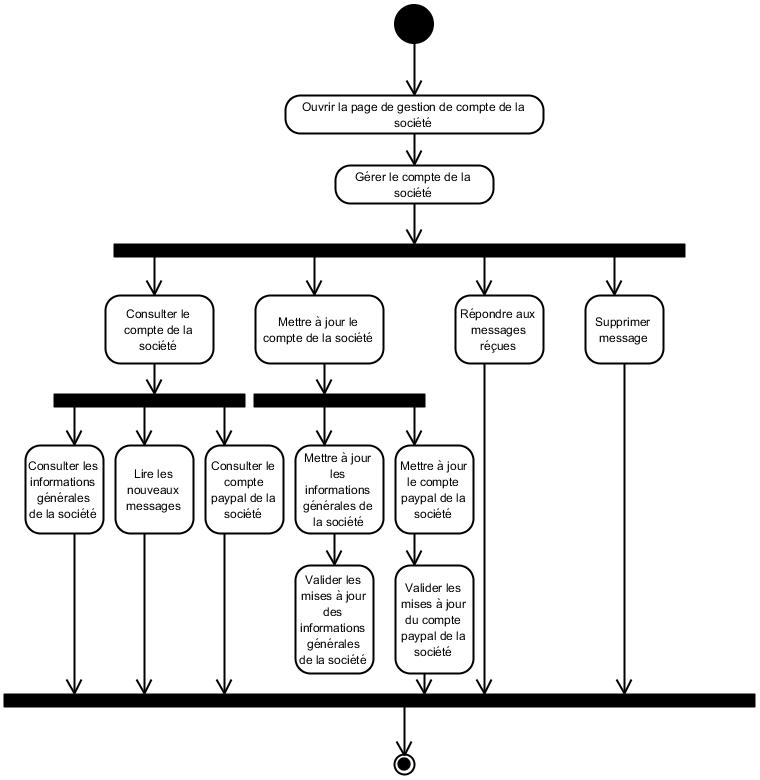


Figure 36 : Diagramme d’activité« Gérer compte de la société»

* + - 1. **Diagramme de séquence «Gérer le compte de la société»**

La figure 37 montre le diagramme de séquence« Gérer compte de la société ».

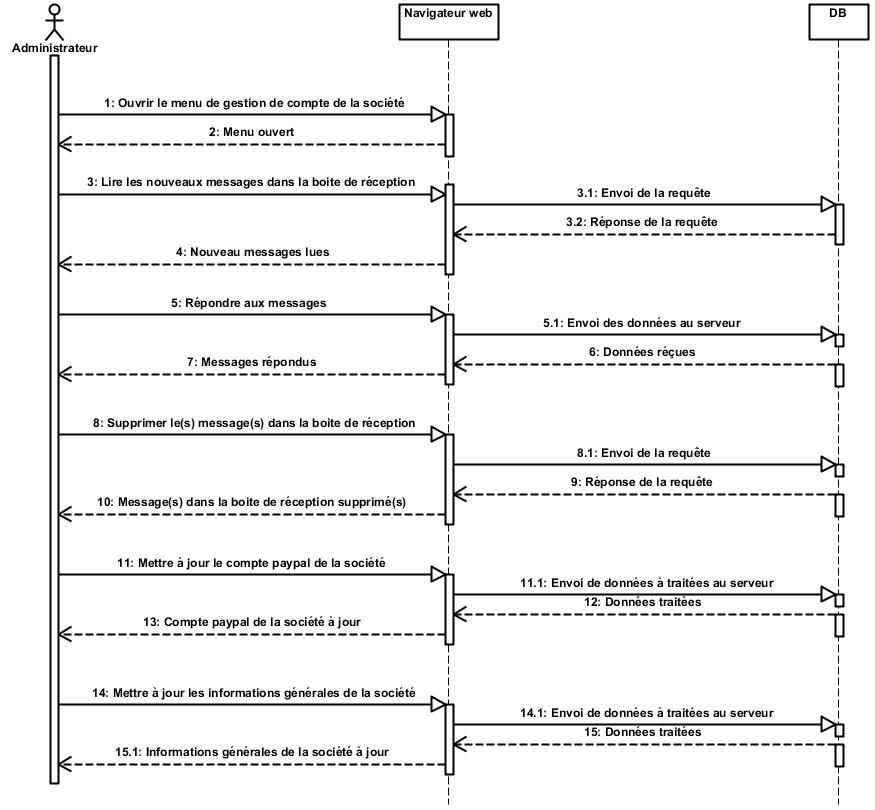


Figure 37 : Diagramme de séquence« Gérer compte de la société»

* + 1. **Itération 10 «La gestion du compte de l’utilisateur»**
       1. **Diagramme de cas d’utilisation «Gérer le compte de l’utilisateur»**

La figure 38 représente le diagramme de cas d’utilisation de « Gérer compte de l’utilisateur».

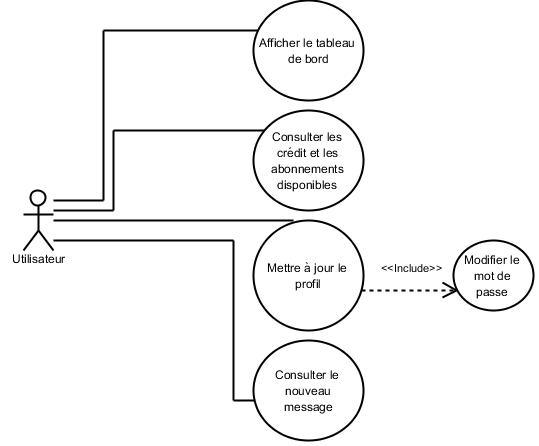


Figure 38 : Diagramme de cas d’utilisation« Gérer compte de l’utilisateur »

**Description textuelle**

* Cas d’utilisation **: La gestion de compte de l’utilisateur**
* Acteur primaire: Utilisateur
* Description : après l’authentification, l’utilisateur peut gérer son compte. En affichant son tableau de bord, l’utilisateur peut consulter ces nouveaux messages, ces abonnements ou crédits disponibles ou même mettre à jour son profil.
* **Analyse**

Une réalisation de ce cas d’utilisation se fait comme suit :

* L’utilisateur affiche son tableau de bord ;
* L’utilisateur consulte et lit ces nouveaux messages dans sa boite de réception ;
* L’utilisateur met à jour son profil.
  + - 1. **Diagramme d’activité «Gérer le compte de l’utilisateur»**

La figure 39représente le diagramme d’activité « Gérer compte de l’utilisateur».

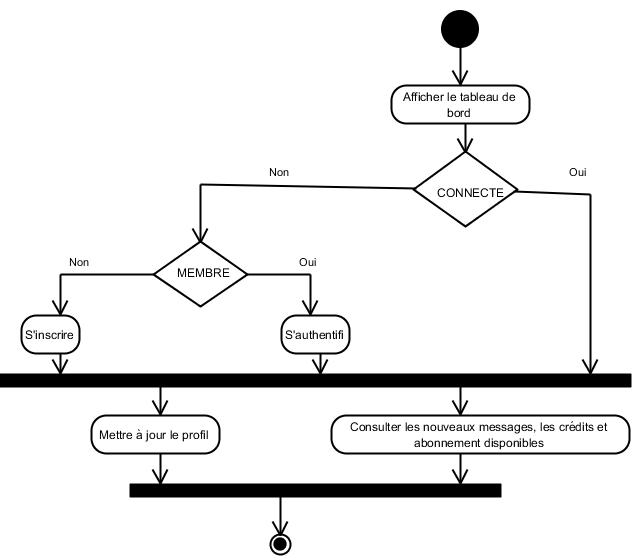


Figure 39 : Diagramme d’activité« Gérer compte de l’utilisateur»

* + - 1. **Diagramme de séquence «Gérer le compte de l’utilisateur»**

La figure 40 représente le diagramme de séquence « Gérer compte de l’utilisateur».

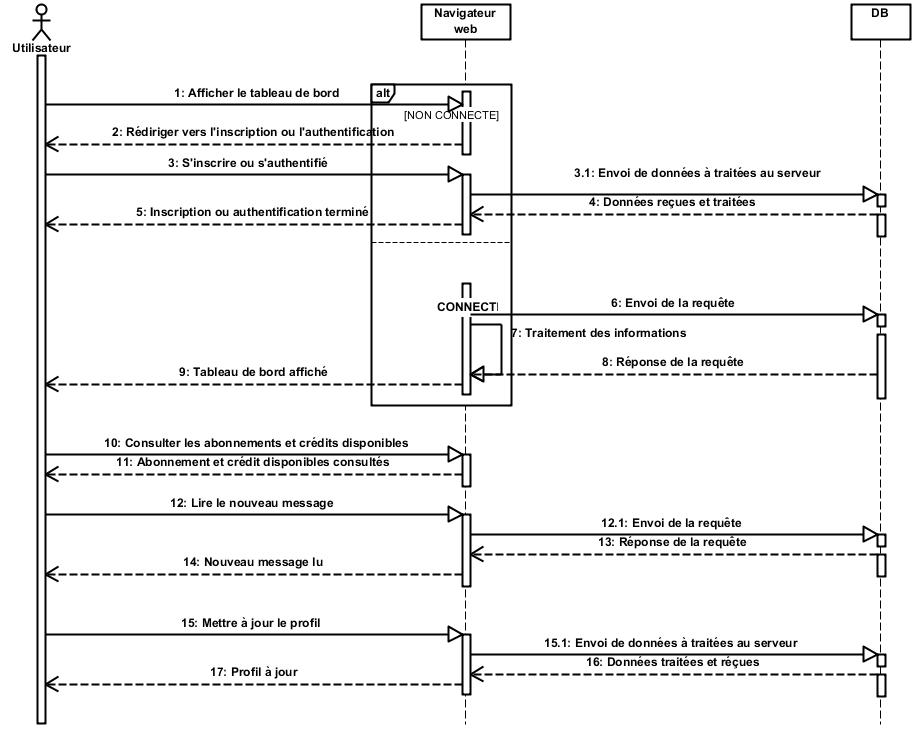


Figure 40 : Diagramme de séquence« Gérer compte de l’utilisateur»

* + 1. **Itération 11 «La gestion de panier»**
       1. **Diagramme de cas d’utilisation «Gérer le panier»**

La figure 41 représente le diagramme de cas d’utilisation de « Gérer le panier».

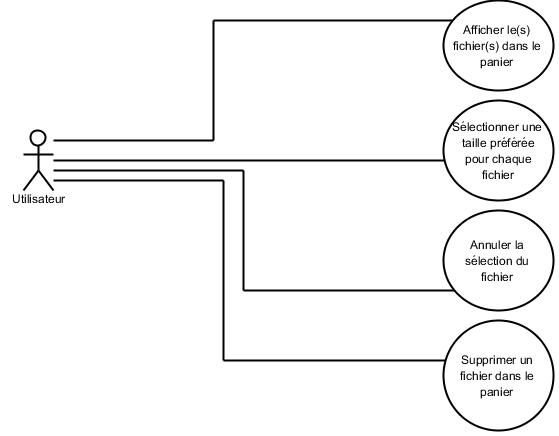


Figure 41 : Diagramme de cas d’utilisation« Gérer le panier »

**Description textuelle**

* Cas d’utilisation **: La gestion du panier**
* Acteur primaire: Utilisateur
* Description : après l’authentification, l’utilisateur peut gérer son panier. En listant les fichiers qui se trouvent dans son panier, l’utilisateur peut choisir la(les) taille(s) de préférences qui sont proportionnels au prix de(s) fichier(s). L’utilisateur peut annuler la (les) sélection(s) de(s) fichier(s) et peut aussi supprimer le(s) fichier(s).
* **Analyse**

Une réalisation de ce cas d’utilisation se fait comme suit :

* L’utilisateur liste les fichiers qui se trouvent dans le panier ;
* L’utilisateur sélectionne les tailles de préférences qui sont proportionnels au prix selon son choix ;
* L’utilisateur annule les sélections des fichiers ;
* L’utilisateur supprime les fichiers.
  + - 1. **Diagramme d’activité «Gérer le panier»**

La figure 42montre le diagramme d’activité « Gérer le panier».

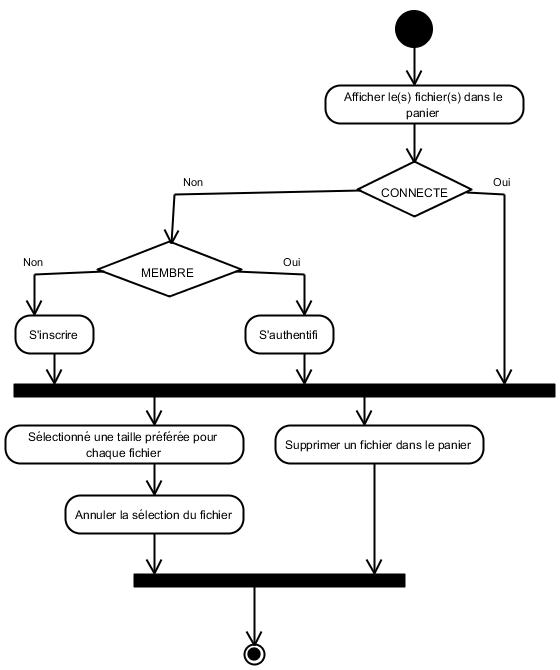


Figure 42 : Diagramme d’activité« Gérer le panier»

* + - 1. **Diagramme de séquence «Gérer le panier»**

La figure 43 montre le diagramme de séquence « Gérer le panier».

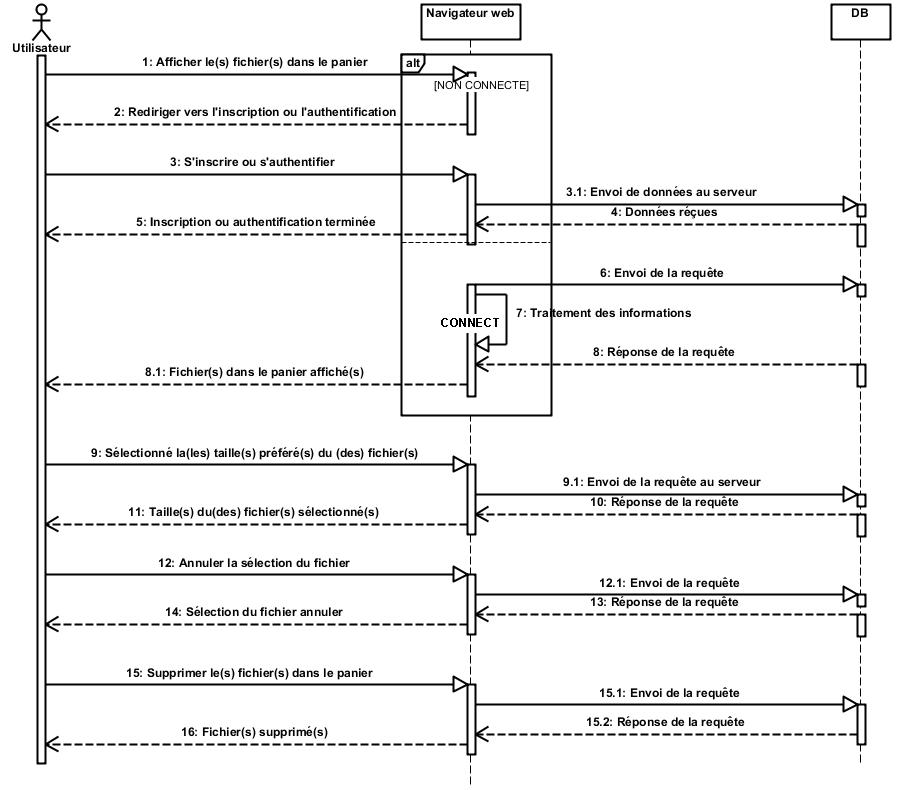


Figure 43 : Diagramme de séquence« Gérer le panier»

* + 1. **Itération 12 «L’uploade des fichiers»**
       1. **Diagramme de cas d’utilisation «Uploader fichiers»**

La figure 44 représente le diagramme de cas d’utilisation de « Uploader fichier».

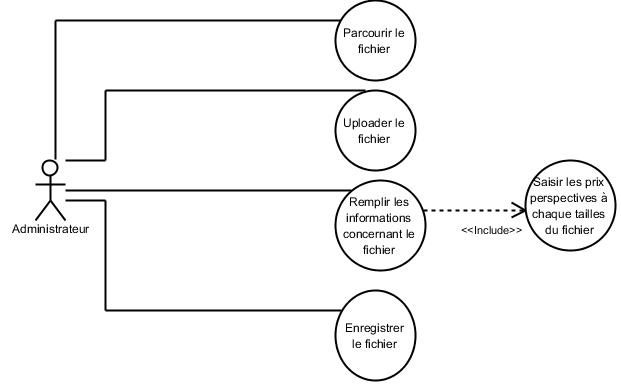


Figure 44 : Diagramme de cas d’utilisation« Uploader fichier »

**Description textuelle**

* Cas d’utilisation **: Uploade de fichier**
* Acteur primaire: Administrateur
* Description : après l’authentification, l’administrateur peut uploader de fichier. Après le parcours de fichier, l’administrateur peut remplir les informations concernant le fichier et aussi de saisir les prix de fichiers selon les tailles qu’il possède.
* **Analyse**

Une réalisation de ce cas d’utilisation se fait comme suit :

* L’administrateur parcours le fichier à uploader ;
* L’administrateur remplit les informations générales du fichier ;
* L’administrateur remplit les prix proportionnels aux tailles que le fichier possèdent ;
* L’administrateur enregistre le fichier.
  + - 1. **Diagramme d’activité «Uploader fichiers»**

La figure 45 montre le diagramme d’activité « Uploader fichier».

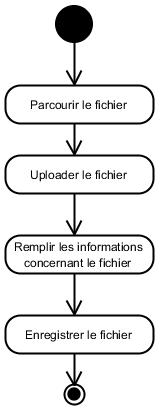


Figure 45 : Diagramme d’activité« Uploader fichier»

* + - 1. **Diagramme de séquence «Uploader fichiers»**

La figure 46représente le diagramme de séquence de « Uploader fichier».

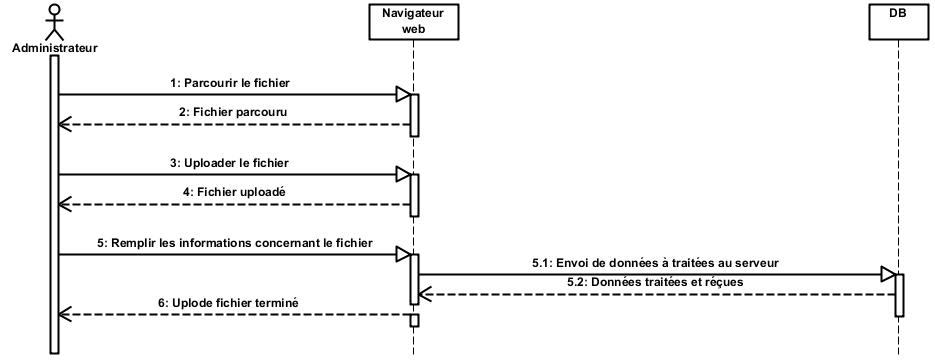


Figure 46 : Diagramme de séquence« Uploader fichier»

* + 1. **Itération 13 «La création de sous compte»**
       1. **Diagramme de cas d’utilisation «Créer sous compte»**

La figure 47 représente le diagramme de cas d’utilisation de « Créer sous compte».

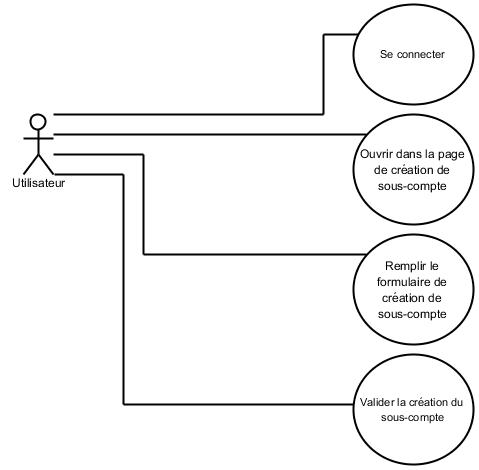


Figure 47 : Diagramme de cas d’utilisation« Créer sous compte »

**Description textuelle**

* Cas d’utilisation **: Créer sous compte**
* Acteur primaire: Utilisateur
* Description : après l’authentification, l’utilisateur peut créer un ou plusieurs sous comptes selon son choix. En allant dans la page de création de sous compte, l’utilisateur peut remplir le formulaire de création de sous comptes et en le validant par suite.
* **Analyse**

Une réalisation de ce cas d’utilisation se fait comme suit :

* L’utilisateur remplis le formulaire de création de sous compte ;
* L’utilisateur valide et enregistre les informations sur le formulaire qu’il a remplis.
  + - 1. **Diagramme d’activité «Créer sous compte»**

La figure 48 montre le diagramme d’activité « Créer sous compte».

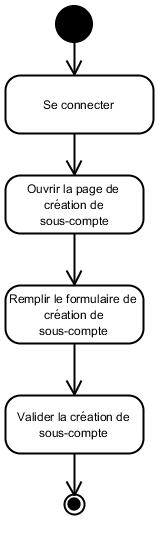


Figure 48 : Diagramme d’activité« Créer sous compte»

* + - 1. **Diagramme de séquence «Créer sous compte»**

La figure 49représente le diagramme de séquence de « Créer sous compte».

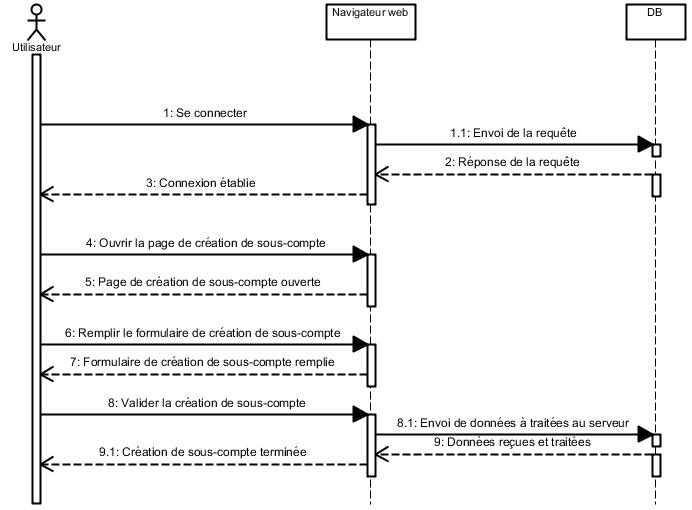


Figure 49 : Diagramme de séquence« Créer sous compte»

* + 1. **Itération 14 «La mise à jour de l’abonnement»**
       1. **Diagramme de cas d’utilisation «Mise à jour de l’abonnement»**

La figure 50 représente le diagramme de cas d’utilisation de « Mise à jour de l’abonnement».

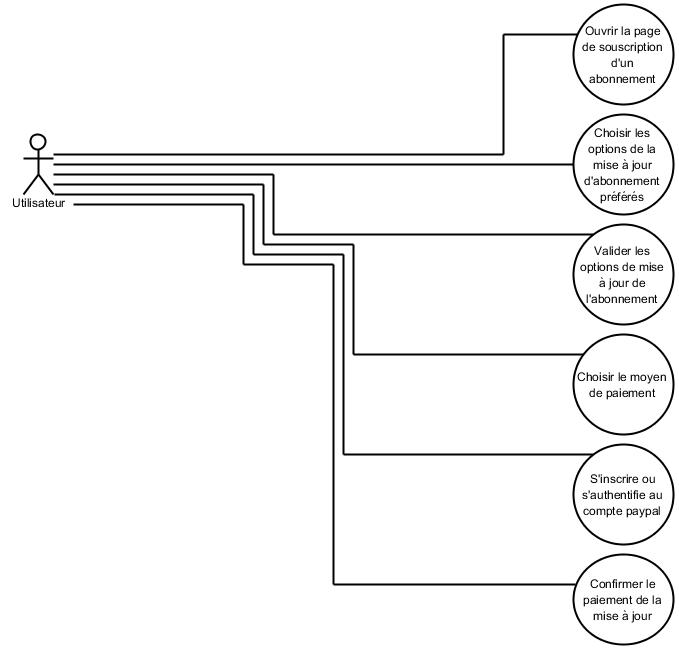


Figure 50 : Diagramme de cas d’utilisation« Mise à jour de l’abonnement »

**Description textuelle**

* Cas d’utilisation **: Mise à jour de l’abonnement**
* Acteur primaire: Utilisateur
* Description : après authentification, si l’utilisateur est déjà souscrit à un abonnement, en allant dans la page de souscription, l’utilisateur sera rediriger vers une page ou il peut effectuer une mise à jour de son abonnement. L’utilisateur peut choisir le type d’abonnement qu’il préfère mettre à jour selon sa possession et aussi, il peut choisir les options de mise à jour qu’il préfère. Après la validation de ses préférences, il peut effectuer un paiement de la mise à jour qu’il a fait selon le mode de paiement qu’il a choisi. A la fin, l’utilisateur confirme son paiement.
* **Analyse**

Une réalisation de ce cas d’utilisation se fait comme suit :

* L’utilisateur choisi le type d’abonnement qu’il préfère mettre à jour selon sa possession,
* L’utilisateur choisi les options de mise à jour de son abonnement,
* L’utilisateur valide les options qu’il a choisies,
* L’utilisateur choisi le moyen de paiement qu’il préfère,
* L’utilisateur valide son choix pour le moyen de paiement,
* L’utilisateur effectue le paiement,
* L’utilisateur confirme le paiement.
  + - 1. **Diagramme d’activité «Créer sous compte»**

La figure 51 montre le diagramme d’activité « Mise à jour de l’abonnement».

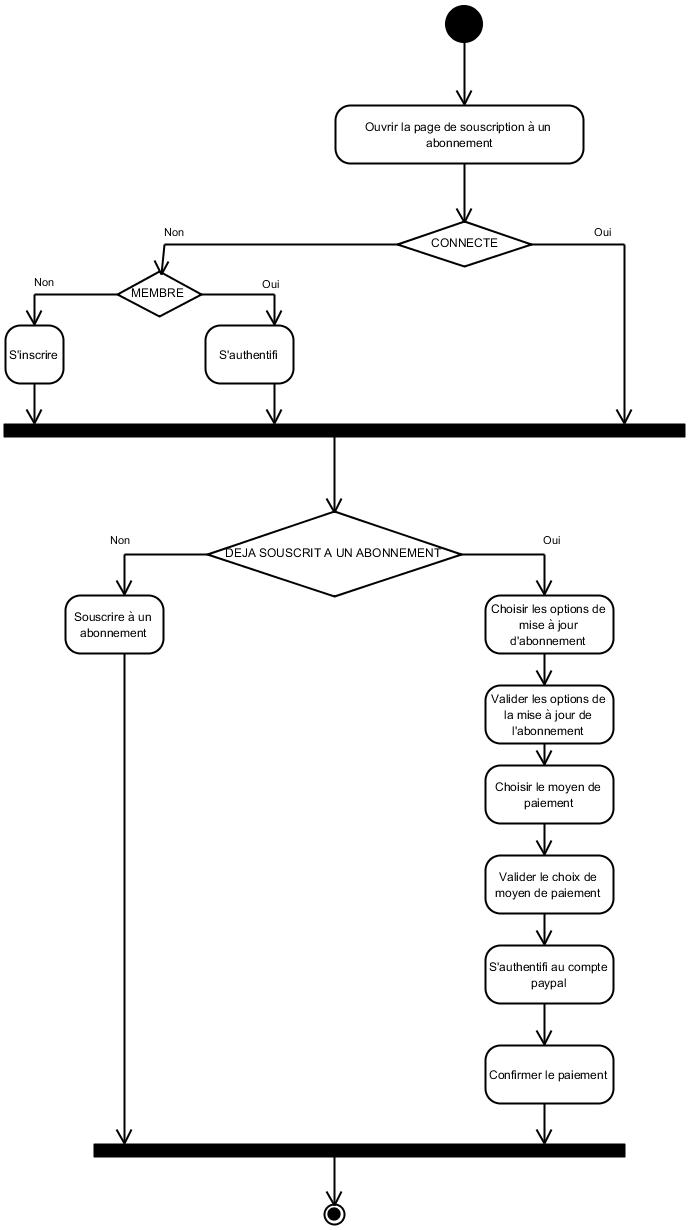


Figure 51 : Diagramme d’activité« Mise à jour de l’abonnement»

* + - 1. **Diagramme de séquence «Mise à jour de l’abonnement»**

La figure 52représente le diagramme de séquence de « Mise à jour de l’abonnement».

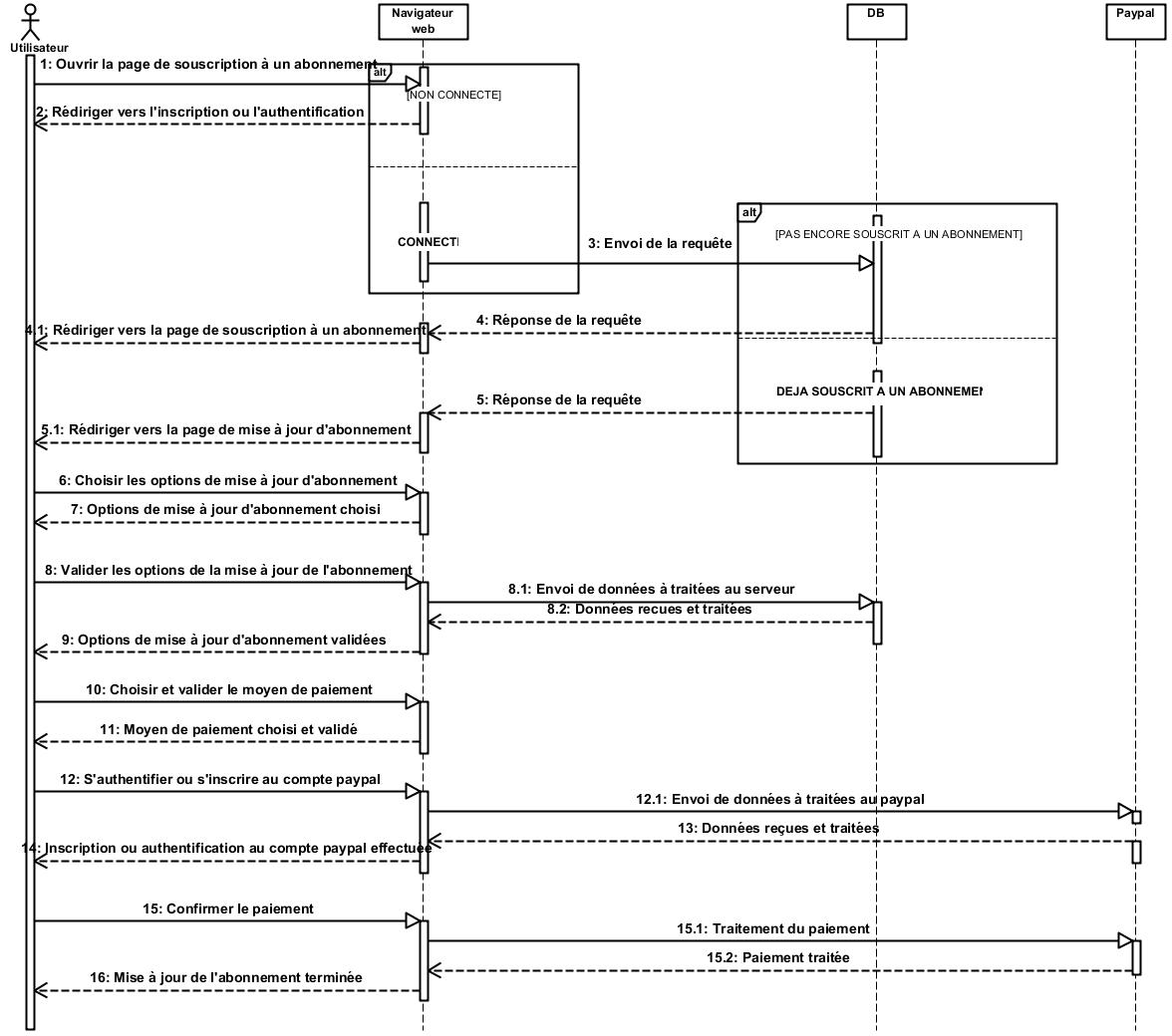


Figure 52 : Diagramme de séquence« Mise à jour de l’abonnement»

## Conclusion

Dans ce chapitre, nous avons conçu et documenté le code que nous devons produire. Dans cette phase, toutes les questions concernant la manière de réaliser le système à développer ont été élucidées. Le produit obtenu est un modèle graphique (ensemble de diagrammes) prêt à être codé. Dans le chapitre suivant nous allons étudier en détails les outils et les langages utilisés durant la phase de réalisation.

1. **REALISATIOIN**
2. REALISATION

**Introduction**

Après avoir achevé l’étape d’analyse conceptuelle, on va entamer dans ce chapitre la partie réalisation et implémentation dans laquelle on s’assure que le système est prêt pour être exploité par les utilisateurs finaux.

A la fin de ce chapitre, les objectifs doivent avoir été atteints et le projet doit être clos.

1. **Choix des technologies**
2. **Technologies Client**
3. **Choix de la technologie NodeJS**
   * + - 1. **Historique**

Node.js a été créé par [Ryan Lienhart Dahl](http://fr.wikipedia.org/w/index.php?title=Ryan_Lienhart_Dahl&action=edit&redlink=1) en 2009. Son développement et sa maintenance sont effectués par l'entreprise [Joyent](http://fr.wikipedia.org/w/index.php?title=Joyent&action=edit&redlink=1)[site 7]. Dahl a eu l'idée de créer *Node.js* après avoir vu la barre de progression d'un chargement de fichier sous [Flickr](http://fr.wikipedia.org/wiki/Flickr). Le navigateur ne savait pas quel pourcentage du fichier était chargé et devait adresser une requête au serveur web. Dahl voulait développer une méthode plus simple. Le [Serveur web Mongrel](http://fr.wikipedia.org/wiki/Mongrel_%28serveur_HTTP%29) de Ruby a été l'autre source d'inspiration pour Dahl. Dahl avait échoué dans plusieurs projets en C, Lua et Haskell, mais, quand [le moteur V8](http://fr.wikipedia.org/wiki/V8_%28moteur_JavaScript%29) fut diffusé, il commença à s'intéresser à JavaScript. Même si son idée d'origine était d'utiliser des [E/S non bloquantes](http://fr.wikipedia.org/w/index.php?title=E/S_Asynchrones&action=edit&redlink=1), il en reviendra un peu dans son module système car cela provoquait des problèmes de chargement de bibliothèques externes[site 7].

* + - * 1. **Quelques avantages de l’utilisation de la technologie NodeJS**
* Open source, sous licence MIT [site 8].
* Projet actif et soutenu par des grands concepts (Microsoft, Google)[site 8].
* Un seul langage Fronted et Backend
* Gestionnaire de modules performant et très simples
* Langage connu de tous, pas de réapprentissage (ou juste une mise à niveau)
* Pas besoins d’une infrastructure « Lourde » (Apache) pour développer l’application[site 8].
* Simplicité de mise en œuvre (installation et test)
* Simplicité de mise en place d’une répartition des traitements sur plusieurs nœuds (distants ou non)
  + - * 1. **Inconvénients**
* Projet jeune et encore très[site 1].
* Documentation peu clair [site 1].
* Pas de communauté française
  + - * 1. **Ses qualités**

Node.js : JavaScript côté serveur permettrait des gains en productivité et performances[site 7]

* Performance élevée,
* Mode asynchrone,
* Gestion native du http,
* Multi-plateformes (mobile, desktop, tv),
* Portabilité du code JavaScript aussi bien côté serveur que client,
* Une communauté très active.
  + - * 1. **Le moteur V8**[site 7].

Node.js utilise le moteur d'exécution ultrarapide V8 de Google Chrome. Ce moteur V8 avait fait beaucoup parler de lui à la sortie de Google Chrome, car c'est un outil open source créé par Google qui analyse et exécute du code JavaScript très rapidement.

Jusqu'à la sortie de Chrome, la plupart des navigateurs lisaient le code JavaScript de façon peu efficace : le code était lu et interprété au fur et à mesure. Le navigateur mettait beaucoup de temps à lire le JavaScript et à le transformer en code machine compréhensible pour le processeur.

Le moteur V8 de Google Chrome, qui est réutilisé ici par Node.js, fonctionne complètement différent. Très optimisé, il fait ce qu'on appelle de la compilation JIT (Just In Time). Il transforme le code JavaScript très rapidement en code machine et l'optimise même grâce à des procédés complexes : *code inlining*, *copy elision*.

* + - * 1. **Installation**

Pour installer NodeJS, il faut d’abord qu’on a les sources d’installation. Selon le système d’exploitation utilisé, la technologie est disponible en open source, d’où le téléchargement de source d’installation sur le site officiel <http://w.w.w.Nodejs.org> est nécessaire[site 1].

1. **Technologies Serveur**
2. **Choix du Framework Express**
   * + - 1. **Généralité**

NodeJS propose plusieurs frameworks comme Socket.io, TotalJS, GeddyJS, TowerJS, SailsJS, Flatiron, [Meteor](http://fr.wikipedia.org/wiki/Meteor_%28framework%29) mais le plus connu de tous est Express[site 9].

La façon d'organiser son code, en divers répertoires et fichiers, est en bonne partie une question de préférence personnelle. Le cadre d'application Web (*Web framework*) [Express](http://expressjs.com/) n'impose aucune structure particulière, laissant le développeur totalement libre d'arranger le tout selon son inspiration du moment[site 9]. Cette liberté peut parfois donner le vertige et provoquer un effet pervers: le syndrome de la page blanche.

* + - * 1. **Structure du répertoire**

Bien qu'il n'y ait pas de *bonne* structure canonique, les bonnes pratiques reconnues et un peu d'expérimentation permettent d'avancer une proposition réfléchie.

Express n'impose aucune structure, mais ce n'est pas toute la vérité. Il suggère une structure, lorsqu'on utilise l'outil de ligne de commande pour créer un cadre de départ (ex.: express struct\_express). Ça donne ceci [site 9]:

struct\_express

public

images

javascripts

stylesheets

routes

views

app.js

On comprend aisément que « public » contient les contenus non sécurisés, utilisés par les pages html produites par l'application et pris en charge par « [Connect](http://www.senchalabs.org/connect/) » (le cadre applicatif plus générique sur lequel est construit Express) via l'intergiciel (middleware) de fichiers statiques[site 7].

Le répertoire « *routes »* porte à confusion, car à l'analyse du code qu'il contient (dans *index.js*), on n'y retrouve non pas la définition des URLs et des verbes HTTP supportés par l'application (la *route*), mais seulement l'implémentation, la logique applicative rattachée à cette route:

// Fichier /routes/index.js

exports.index=function(req,res){

res.render('index',{title:'Express'})

};

La définition de la route comme telle, elle se retrouve à la racine de l'application, dans le fichier maître « app.js »:

// Fichier /app.js

varexpress=require('express')

,routes=require('./routes')

// ...

// Routes

app.get('/',routes.index);

C'est probablement acceptable pour de petits projets ou des tests rapides, mais pour une application d'une certaine taille, une meilleure organisation du code est nécessaire.

Le répertoire *« views »*, on retrouve les modèles de vues (*templates*) servant à produire les pages html. Personnellement nous avons utilisé jade, mais Express supporte d'autres engins. Chacun de ces templates ayant son lot d'avantages et d'inconvénients. En général ils gèrent tous l'essentiel, à savoir : les variables, les conditions, les boucles[site 8].

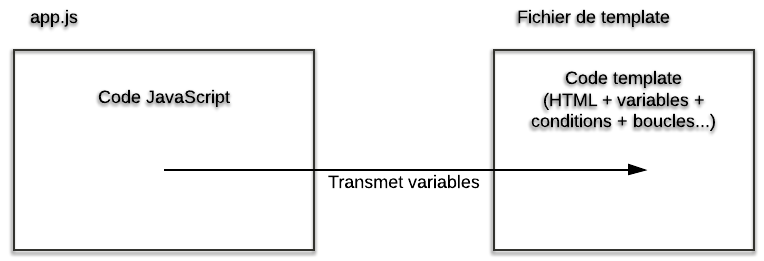
La figure 53 représente le principe de transmission des codes javascripts dans le template.

Figure 53 : Transmission du code JavaScript au template

1. **Choix du Framework Bootstrap**
   * + - 1. **Généralité**

Vous connaissez forcément Twitter, un des principaux réseaux sociaux qui inondent la planète de liens virtuels entre les humainsdevenus des noyaux cybernétiques[site 4].

Le projet Bootstrap a été publié en août 2011 pour répondre à des besoins internes de développement de cette entrepris e. Il a été mis à dis position du public sous licence Apache. La vers ion 2.1 est sorti en août 2012.En août 2013, soit deux ans après le lancement de la première version, est sorti Bootstrap 3. Cette dernière version offre un tout nouveau design, une meilleure gestion des erreurs, des options de personnalisation avancées et un tas d’autre goodies qui font frétiller d’envie les développeurs[5].

Bootstrap est une compilation de plusieurs éléments et fonctions web-design personnalisables, le tout emballé dans un seul et même outil. Les développeurs qui utilisent Bootstrap pour la création de leur site web choisissent les éléments qu’ils veulent utiliser avec la certitude qu’ils ne seront pas incompatibles entre eux. En fait, c’est comme un puzzle. Sauf que dans ce puzzle, chaque pièce s’imbrique parfaitement dans les autres, quelle qu’elle soit[5].

* + - * 1. **Contenu du kit**

Bootstrap propose :

* une mise en page basée sur une grille de 12 colonnes. Vous avez donc une grille à disposition bien pratique. Bien sûr, si vous avez besoin de plus de 12 colonnes, ou de moins, il est toujours possible de modifier la configuration[site 9].
* un reset CSS bas é sur Normalize.css
* du code fondé sur HTML 5 et CSS 3
* une bibliothèque totalement open source sous licence Apache
* du code qui tient compte du format d'affichage des principaux outils de navigation (responsive design) : smartphones, tablettes ...
* des plugins jQuery de qualité
* un résultat cross -browser, en prenant même en compte IE7, donc une garantie de compatibilité maximale
* une bonne documentation sur le site de Twitter
* la garantie d'une évolution permanente
* une architecture basée sur LESS, un outil bien pratique qui étend les possibilités de CSS

**C'est quoi un reset CSS ?**

Les navigateurs n'adoptent pas tous les mêmes valeurs par défaut pour les styles des éléments **HTML**. Cela peut générer quelques surprises au rendu des pages WEB selon le navigateur utilisé. Le reset consiste à mettre à zéro toutes les valeurs pour ainsi uniformiser le rendu. Le reset de **Bootstrap**est basé sur [celui d'Eric Meyer](http://meyerweb.com/eric/tools/css/reset/index.html), allégé et adapté [site 9].

1. **LESS : Compilateur CSSS**
   * + - 1. **Généralité**

LESS est un langage dynamique de génération de [feuilles de style](http://fr.wikipedia.org/wiki/Feuilles_de_style_en_cascade) conçu par Alexis Sellier. Il est influencé par [Sass](http://fr.wikipedia.org/wiki/Sass_%28langage%29) et a influencé "SCSS" (la syntaxe plus récente de Sass) : la syntaxe de LESS est plus proche de CSS1 et un code CSS est aussi un code LESS valide qui a la même sémantique [site 9].

LESS ajoute à CSS les mécanismes suivants : variables, imbrication, [mixins](http://fr.wikipedia.org/wiki/Mixin), [opérateurs](http://fr.wikipedia.org/wiki/Op%C3%A9rateur_%28informatique%29) et [fonctions](http://fr.wikipedia.org/wiki/Fonction_informatique).

LESS est implémenté en [open source](http://fr.wikipedia.org/wiki/Open_source). Sa première version a été écrite en Ruby. Dans les versions ultérieures, Ruby a été remplacé par [Javascript](http://fr.wikipedia.org/wiki/Javascript). Une particularité de LESS par rapport aux autres préprocesseurs CSS est qu'il peut être compilé à la volée, soit lors du rendu par le navigateur, soit côté serveur. Il peut également être compilé à l'avance en un simple fichier CSS [site 9].

* + - * 1. **Comparaison avec Sass**

[Sass](http://fr.wikipedia.org/w/index.php?title=Sass_%28stylesheet_language%29&action=edit&redlink=1) et LESS sont tous deux des préprocesseurs CSS, qui permettent d'écrire des feuilles CSS propres au lieu de règles statiques[2](file:///C:\Users\RANDRIANARIKOTO\Documents\boris\Downloads\Documents\LESS%20%28langage%29%20%E2%80%94%20Wikip%C3%A9dia.htm#cite_note-whats-wrong-with-css-2).

À partir de la version 1.4, LESS supporte les règles imbriquées et héritées via les pseudo-sélecteurs &:extends et @extends. Auparavant, la principale différence entre LESS et les autres préprocesseurs comme SASS était le manque d'une directive @extends, qui permettrait d'étendre une classe existante pour lui ajouter plus de règles, ce qui conduirait à un code CSS plus propre, sans règles dupliquées [site 9].

LESS est inspiré par Sass[3](file:///C:\Users\RANDRIANARIKOTO\Documents\boris\Downloads\Documents\LESS%20%28langage%29%20%E2%80%94%20Wikip%C3%A9dia.htm#cite_note-about-3). Sass a été conçu à la fois pour simplifier et étendre CSS, ainsi des éléments comme les accolades ont été enlevés de la syntaxe. LESS a été conçu pour être aussi proche que possible de CSS, ainsi la syntaxe est identique à un code CSS. En conséquence, un code CSS peut être utilisé comme un code LESS valide [site 9].

Des versions plus récentes de Sass ont également introduit une syntaxe à la CSS appelée SCSS (Sassy CSS).

* + - * 1. **Utilisation**

LESS peut être utilisé sur un site de différentes façons. Une possibilité est d'inclure le fichier [JavaScript](http://fr.wikipedia.org/wiki/JavaScript) less.js pour convertir le code à la volée. Le navigateur compilera le fichier destination CSS. Une autre possibilité est de compiler le fichier LESS en pur CSS avant le l'uploader sur le site. Avec cette option, il n'y a pas besoin d'uploader de fichier .less et le site n'a pas besoin du convertisseur javascript less.js.

1. **Serveur de bases de données : MySQL**

**Remarque**

Dans un premier temps du de la réalisation du projet, nous avons choisi comme base de données MongoDB**.**

MongoDB est l'une des bases de données composant le mouvement NoSQL au même titre qu'Apache CouchDB pour ne citer qu'elle. L'intérêt de ce genre de bases de données se ressent dans la manipulation de très grosses bases de données où le temps de réponse compte plus que l'intégrité des données.

En effet, les bases NoSQL sont des SGBD et laissent ainsi tomber le R de relationnel au profit d'une plus grande simplicité, performance et montée en charge.

Mais par suite des événements, les clients ont eu des soucis sur la maintenance et la mise en place d’un serveur de base de données MongoDB. Comme c’est une base plus récente, nombreux utilisateurs ont encore des difficultés sur sa maintenance.

C’est qui nous a poussés à choisir MySQL comme serveur de base données, étant donné qu’elle est flexible et performante.

* + - * 1. **Quelques avantages de MySQL**

Les principaux concurrents de MySQL sont PostgreSQL, Microsoft SQL Server et Oracle. Par rapport à tous ces produits, MySQL possède plusieurs avantages majeurs, dont :

* les performances élevées ;
* le coût réduit ;
* la simplicité de configuration ;
* la portabilité ;
* l'accessibilité de son code source.
  + - * 1. **Performances de MySQL**

MySQL est indéniablement un système rapide. Nous pouvons consulter les statistiques des performances sur le site **www.mysql.com**.

* **Coût** : MySQL est disponible gratuitement, sous une licence Open Source, ou pour un prix très raisonnable pour les licences commerciales si nous en avons besoin pour développer une application.
* **Portabilité** : MySQL peut être utilisé sur un grand nombre de systèmes Unix, ainsi qu'avec Windows.

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

La figure 54illustre l’architecture finale de l’application après les divers choix technologiques

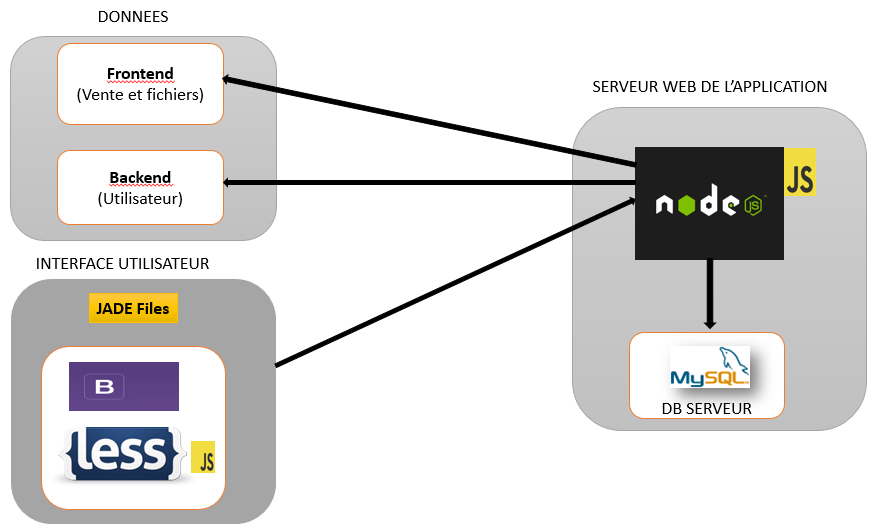


Figure 54 : Architecture finale de l’application

1. **Environnement d’implémentation**
2. **Moyen matériel**

Une connexion Internet est mise à la disposition du stagiaire dans le but d’offrir une condition de travail convenable, des opportunités d’élargir les connaissances et de faciliter la documentation. En outre, à cause de manque de matériel informatique à l’entreprise, nous avons utilisé un ordinateur personnel. Ci-dessous les caractéristiques de la machine:

Ordinateur portable

Intel(R) Core™ i3-2350M CPU

Processeur : 2.30GHz

Mémoire RAM : 4086MB

Disque dur 250 Go

1. **Outils et logiciels utilisés**

Tableau 19 : Liste des outils et logiciels

|  |  |  |
| --- | --- | --- |
| **PHASE** | **LOGICIEL** | **DESCRIPTION** |
| **ANALYSE ET CONCEPTION** | **Visual Paradigm** | Visual Paradigm est un [logiciel](http://fr.wikipedia.org/wiki/Logiciel) de modélisation. Il permet de modéliser les traitements informatiques et leurs [bases de données](http://fr.wikipedia.org/wiki/Base_de_donn%C3%A9es) associées [site 4]. |
| **IceScrum** | C’est un outil d'aide à la gestion de projet Scrum. Il est particulièrement apprécié pour son interface aboutie et originale. |
| **REALISATION** | **Sublime Text 2** | Sublime Text est un outil permettant d’éditer des fonctions JavaScripts et de manipuler les codes NodeJS |
| **Notepad++ v5.8.5** | Notepad++ est un éditeur de texte générique codé en C++, qui intègre la coloration syntaxique de code source |
| **Crunch** | Crunch est outil permettant de compiler et programmer en LESS |
| **Firefox 30** | Navigateurs web basé |
| **WampServer 2.0** | WampServer (anciennement WAMP5) est une plateforme de développement Web de type [WAMP](http://fr.wikipedia.org/wiki/WAMP), permettant de faire fonctionner localement (sans se connecter à un serveur externe) des scripts [PHP](http://fr.wikipedia.org/wiki/PHP). |

1. **Extraits de code**
   * 1. **Point d’entrée et dépendance de l’application**

Cet extrait de code fait partie du point d’entrée et de la configuration de la dépendance de l’application qui se trouve dans le fichier « package. json »

{

"name": "samFluoo",

"version": "0.0.1",

"private": true,

"scripts": {

"start": "node ./bin/www"

},

"dependencies": {

"express": "~4.2.0",

"static-favicon": "~1.0.0",

"morgan": "~1.0.0",

"cookie-parser": "~1.0.1",

"body-parser": "~1.0.0",

"debug": "~0.7.4",

"jade": "~1.3.0",

"rimraf": "2.1",

"monk": "\*"

}

}

* + 1. **Définition de routes utilisées par l’application, configuration de la connexion au base de données et appellation des modules nécessaires**

La définition de la route comme telle, elle se retrouve à la racine de l'application, dans le fichier maître « app.js », et « app.js » comporte aussi l’appellation des différents modules qui sont nécessaires à l’exécution de l’application. Voici un extrait de code.

//Exportation des modules necessaries

var express = require('express');

var path = require('path');

...

var mysql = require('mysql');

var session = require('express-session')

var SessionStore = require('express-mysql-session')

var less = require('less');

var exec = require('child\_process').exec;

…

//Definition des routes utlisés dans l’application

var routes = require('./routes/index');

var usersAdmin = require('./routes/forAdmin/users');

var pyAdmin = require('./routes/python');

var membres = require('./routes/membres');

var imgClient = require('./routes/images');

var imgVectClient = require('./routes/vecteurs');

var videoClient = require('./routes/videos');

var panier = require('./routes/panier');

var lb = require('./routes/lightbox');

var paiement\_paypal= require('./routes/paiement\_paypal');

var abonnement = require('./routes/abonnement');

var app = express();

// view engine setup

app.set('views', path.join(\_\_dirname, 'views'));

app.set('view engine', 'jade');

…

//se souvien du login et de mot de passe

app.use(function(req, res,next){

var expire\_authentification = Date.now() \* 1000;

if(!req.cookies.remember\_login){

res.cookie('remember\_login', "", { maxAge: expire\_authentification });

res.cookie('remember\_pwd', "", { maxAge: expire\_authentification });

}

next();

});

app.use(function(req, res,next){

res.locals.user = req.session.user;

req.session.mailSite = 'masinera.ravelona@gmail.com';

req.session.mailSitePass = 'ravelonamasinera';

var err = req.session.error;

// delete req.sesion.error;

res.locals.message = '';

if (err) {

res.locals.message = err;

}

var sqlPanier = '(SELECT '+

'panier.\*,fichier.nom\_fichier,fichier.type\_fichier,fichier.XS, fichier.S,fichier.M,fichier.L,fichier.XL,fichier.XXL,fichier.X ' +

'FROM panier, fichier, categorie\_2 where panier.id\_fichier = fichier.id\_fichier and fichier.id\_categorie = categorie\_2.id\_categorie\_2 and panier.loginUtilisateur = ?) ' +

'UNION ' +

'(SELECT ' +

'panier.\*,videos.nom\_fichier,videos.type\_fichier,videos.titre,videos.CGA,videos.EGA,videos.XGA,videos.U\_XGA,videos.HD\_1080,videos.X ' +

'FROM panier, categorie\_2,videos WHERE panier.id\_video = videos.id\_video AND videos.id\_categorie = categorie\_2.id\_categorie\_2 and panier.loginUtilisateur = ?)';

connection.query(sqlPanier, [req.session.user,req.session.user], function(e, docsPanier){

if(e) req.session.nbPaniers = 0;

else req.session.nbPaniers = docsPanier.length;

});

var sqlLightbox = '(SELECT '+

'lightbox.\*,fichier.nom\_fichier,fichier.type\_fichier,fichier.XS, fichier.S,fichier.M,fichier.L,fichier.XL,fichier.XXL,fichier.X ' +

'FROM lightbox, fichier, categorie\_2 where lightbox.id\_fichier = fichier.id\_fichier and fichier.id\_categorie = categorie\_2.id\_categorie\_2 and lightbox.loginUtilisateur = ?) ' +

'UNION ' +

'(SELECT ' +

'lightbox.\*,videos.nom\_fichier,videos.type\_fichier,videos.titre,videos.CGA,videos.EGA,videos.XGA,videos.U\_XGA,videos.HD\_1080,videos.X ' +

'FROM lightbox, categorie\_2,videos WHERE lightbox.id\_video = videos.id\_video AND videos.id\_categorie = categorie\_2.id\_categorie\_2 and lightbox.loginUtilisateur = ?)';

connection.query(sqlLightbox, [req.session.user,req.session.user], function(e, docsLightbox){

if(e) req.session.nbLightboxs = 0;

else req.session.nbLightboxs = docsLightbox.length;

});

connection.query('SELECT \* FROM credit\_disponible WHERE loginUtilisateur= ?', req.session.user, function(e, docsCreditDispo){

if(e) req.session.nbCreditDispo = 0;

else{

if(docsCreditDispo.length > 0){//si le login entré existe

for (var i = 0, len = docsCreditDispo.length; i < len; ++i) {

var sqlResult = docsCreditDispo[i];

nbCreditDispo = sqlResult['creditDispo'];

req.session.nbCreditDispo = nbCreditDispo;

}

}

else{

req.session.nbCreditDispo = 0;

}

}

});

next();

});

app.use(function(req, res,next){

connection.query('SELECT \* FROM comptesociete LIMIT 0, 1', function(err, results) {

if (err){

info = "Vous devez ajouter un compte de la societe";

alert\_class = "alert alert-danger";

}

else{

if(results.length > 0){//si le login entré existe

for (var i = 0, len = results.length; i < len; ++i) {

var sqlResult = results[i];

nomSociete = sqlResult['nom\_societe'],

tel = sqlResult['tel'],

businessMail = sqlResult['paypal\_mailBusiness'],

paypal\_client\_id = sqlResult['paypal\_client\_id'],

paypal\_client\_secret = sqlResult['paypal\_client\_secret'];

req.session.nom\_societe = nomSociete;

req.session.tel\_societe = tel;

req.session.paypal\_client\_id = paypal\_client\_id;

req.session.paypal\_client\_secret = paypal\_client\_secret;

}

}

}

});

next();

});

//Faire la base accessible au route

app.use(function(req, res,next){

req.db\_connection = connection;

next();

});

app.use('/', routes);

app.use('/login', routes);

app.use('/logout', routes);

app.use('/forAdmin/users', usersAdmin);

app.use('/forAdmin/contributeurs', contributeursAdmin);

app.use('/forAdmin/images', imgAdmin);

app.use('/forAdmin/vecteurs', vecteursAdmin);

app.use('/forAdmin/videos', videoAdmin);

app.use('/forAdmin/categorie', categoriesAdmin);

app.use('/forAdmin/python', pyAdmin);

app.use('/forAdmin/credit', creditAdmin);

app.use('/forAdmin/abonnement', abonnementAdmin);

app.use('/forAdmin/moyenPaiement', moyenPaiement);

app.use('/forAdmin/compte', compte);

app.use('/forAdmin/info', infoAdmin);

app.use('/forAdmin/faq', faqAdmin);

app.use('/forAdmin/dmdConvertCredit', dmdConvertCredit);

app.use('/membres', membres);

app.use('/images', imgClient);

app.use('/vecteurs', imgVectClient);

app.use('/videos', videoClient);

app.use('/panier', panier);

app.use('/Lightbox', lb);

app.use('/credit', credit);

app.use('/moyen\_paiement',moyen\_paiement);

app.use('/paiement\_paypal',paiement\_paypal);

app.use('/info', info);

app.use('/MesFichiers', mesFichiers);

app.use('/Categories', categories);

app.use('/Recherche', recherche);

//app.use('/twitter', twitter);

app.use('/abonnement', abonnement);

/// catch 404 and forward to error handler

app.use(function(req, res, next) {

var err = new Error('Not Found');

err.status = 404;

next(err);

});

/// error handlers

/\*

// development error handler

if (app.get('env') === 'development') {

app.use(function(err, req, res, next) {

res.status(err.status || 404);

res.render('pageNotFound', {

title: "Acheter des images et videos sur "+ req.session.nom\_societe,

"tel\_societe":req.session.tel\_societe,"nom\_societe": req.session.nom\_societe

});

});

}

\*/

// will print stacktrace

if (app.get('env') === 'development') {

app.use(function(err, req, res, next) {

res.status(err.status || 500);

res.render('error', {

message: err.message,

error: err

});

});

}

// production error handler

// no stacktraces leaked to user

app.use(function(err, req, res, next) {

res.status(err.status || 500);

res.render('error', {

message: err.message,

error: {}

});

});

module.exports = app;

//option de configuration de la connexion à la base de données

var options = {

host: 'localhost',

port: 3306,

user:"root",

password:'alex',

database :'photodb'

}

app.use(session({

key: 'session\_admin',

secret: 'admin',

store: new SessionStore(options)

}));

var connection = mysql.createConnection({

host : options.host,

user : options.user,

password : options.password

});

connection.query("use "+ options.database);

app.use(favicon());

…

app.use(express.static(path.join(\_\_dirname, 'node\_modules')));

app.use(busboy());

//authentification .htaccess

var auth = require('http-auth');

var basic = auth.basic({

realm: "Page d'adinistration du site de photos.",

file: \_\_dirname + "/conf/users.htpasswd" // gevorg:gpass, Sarah:testpass ...

});

…

app.use(function(req, res,next){

var expire\_authentification = Date.now() \* 1000;

if(!req.cookies.remember\_login){

res.cookie('remember\_login', "", { maxAge: expire\_authentification });

res.cookie('remember\_pwd', "", { maxAge: expire\_authentification });

}

next();

});

app.use(function(req, res,next){

res.locals.user = req.session.user;

req.session.mailSite = 'masinera.ravelona@gmail.com';

req.session.mailSitePass = 'ravelonamasinera';

var err = req.session.error;

// delete req.sesion.error;

res.locals.message = '';

if (err) {

res.locals.message = err;

}

var sqlPanier = '(SELECT '+

'panier.\*,fichier.nom\_fichier,fichier.type\_fichier,fichier.XS, fichier.S,fichier.M,fichier.L,fichier.XL,fichier.XXL,fichier.X ' +

'FROM panier, fichier, categorie\_2 where panier.id\_fichier = fichier.id\_fichier and fichier.id\_categorie = categorie\_2.id\_categorie\_2 and panier.loginUtilisateur = ?) ' +

'UNION ' +

'(SELECT ' +

'panier.\*,videos.nom\_fichier,videos.type\_fichier,videos.titre,videos.CGA,videos.EGA,videos.XGA,videos.U\_XGA,videos.HD\_1080,videos.X ' +

'FROM panier, categorie\_2,videos WHERE panier.id\_video = videos.id\_video AND videos.id\_categorie = categorie\_2.id\_categorie\_2 and panier.loginUtilisateur = ?)';

connection.query(sqlPanier, [req.session.user,req.session.user], function(e, docsPanier){

if(e) req.session.nbPaniers = 0;

else req.session.nbPaniers = docsPanier.length;

});

var sqlLightbox = '(SELECT '+

'lightbox.\*,fichier.nom\_fichier,fichier.type\_fichier,fichier.XS, fichier.S,fichier.M,fichier.L,fichier.XL,fichier.XXL,fichier.X ' +

'FROM lightbox, fichier, categorie\_2 where lightbox.id\_fichier = fichier.id\_fichier and fichier.id\_categorie = categorie\_2.id\_categorie\_2 and lightbox.loginUtilisateur = ?) ' +

'UNION ' +

'(SELECT ' +

'lightbox.\*,videos.nom\_fichier,videos.type\_fichier,videos.titre,videos.CGA,videos.EGA,videos.XGA,videos.U\_XGA,videos.HD\_1080,videos.X ' +

'FROM lightbox, categorie\_2,videos WHERE lightbox.id\_video = videos.id\_video AND videos.id\_categorie = categorie\_2.id\_categorie\_2 and lightbox.loginUtilisateur = ?)';

connection.query(sqlLightbox, [req.session.user,req.session.user], function(e, docsLightbox){

if(e) req.session.nbLightboxs = 0;

else req.session.nbLightboxs = docsLightbox.length;

});

connection.query('SELECT \* FROM credit\_disponible WHERE loginUtilisateur= ?', req.session.user, function(e, docsCreditDispo){

if(e) req.session.nbCreditDispo = 0;

else{

if(docsCreditDispo.length > 0){//si le login entré existe

for (var i = 0, len = docsCreditDispo.length; i < len; ++i) {

var sqlResult = docsCreditDispo[i];

nbCreditDispo = sqlResult['creditDispo'];

req.session.nbCreditDispo = nbCreditDispo;

}

}

else{

req.session.nbCreditDispo = 0;

}

}

});

next();

});

app.use(function(req, res,next){

connection.query('SELECT \* FROM comptesociete LIMIT 0, 1', function(err, results) {

if (err){

info = "Vous devez ajouter un compte de la societe";

alert\_class = "alert alert-danger";

}

else{

if(results.length > 0){//si le login entré existe

for (var i = 0, len = results.length; i < len; ++i) {

var sqlResult = results[i];

nomSociete = sqlResult['nom\_societe'],

tel = sqlResult['tel'],

businessMail = sqlResult['paypal\_mailBusiness'],

paypal\_client\_id = sqlResult['paypal\_client\_id'],

paypal\_client\_secret = sqlResult['paypal\_client\_secret'];

req.session.nom\_societe = nomSociete;

req.session.tel\_societe = tel;

req.session.paypal\_client\_id = paypal\_client\_id;

req.session.paypal\_client\_secret = paypal\_client\_secret;

}

}

}

});

next();

});

//Faire la base accessible au route

app.use(function(req, res,next){

req.db\_connection = connection;

next();

});

app.use('/', routes);

app.use('/login', routes);

app.use('/logout', routes);

app.use('/forAdmin/users', usersAdmin);

app.use('/forAdmin/contributeurs', contributeursAdmin);

app.use('/forAdmin/images', imgAdmin);

app.use('/forAdmin/vecteurs', vecteursAdmin);

app.use('/forAdmin/videos', videoAdmin);

app.use('/forAdmin/categorie', categoriesAdmin);

app.use('/forAdmin/python', pyAdmin);

app.use('/forAdmin/credit', creditAdmin);

app.use('/forAdmin/abonnement', abonnementAdmin);

app.use('/forAdmin/moyenPaiement', moyenPaiement);

app.use('/forAdmin/compte', compte);

app.use('/forAdmin/info', infoAdmin);

app.use('/forAdmin/faq', faqAdmin);

app.use('/forAdmin/dmdConvertCredit', dmdConvertCredit);

app.use('/membres', membres);

app.use('/images', imgClient);

app.use('/vecteurs', imgVectClient);

app.use('/videos', videoClient);

app.use('/panier', panier);

app.use('/Lightbox', lb);

app.use('/credit', credit);

app.use('/moyen\_paiement',moyen\_paiement);

app.use('/paiement\_paypal',paiement\_paypal);

app.use('/info', info);

app.use('/MesFichiers', mesFichiers);

app.use('/Categories', categories);

app.use('/Recherche', recherche);

//app.use('/twitter', twitter);

app.use('/abonnement', abonnement);

/// catch 404 and forward to error handler

app.use(function(req, res, next) {

var err = new Error('Not Found');

err.status = 404;

next(err);

});

/// error handlers

/\*

// development error handler

if (app.get('env') === 'development') {

app.use(function(err, req, res, next) {

res.status(err.status || 404);

res.render('pageNotFound', {

title: "Acheter des images et videos sur "+ req.session.nom\_societe,

"tel\_societe":req.session.tel\_societe,"nom\_societe": req.session.nom\_societe

});

});

}

\*/

// will print stacktrace

if (app.get('env') === 'development') {

app.use(function(err, req, res, next) {

res.status(err.status || 500);

res.render('error', {

message: err.message,

error: err

});

});

}

// production error handler

// no stacktraces leaked to user

app.use(function(err, req, res, next) {

res.status(err.status || 500);

res.render('error', {

message: err.message,

error: {}

});

});

module.exports = app;

…

// production error handler

// no stacktraces leaked to user

app.use(function(err, req, res, next) {

res.status(err.status || 500);

res.render('error', {

message: err.message,

error: {}

});

});

module.exports = app;

* + 1. **Validation de l’abonnement et envoi des données à traités au paypal**

Cet extrait de code se trouve dans le fichier route « abonnement.js ».

/\*valide du formulaire d’abonnement\*/

router.post('/valideAbnmt', function(req, res){

//Declaration des variables

var con = req.db\_connection;

var method = req.body.methodPayment;

var currency = req.body.currency;

var amount = req.body.amount;

var shipping = req.body.shipping;

var total = parseInt(amount)+parseInt(shipping);

var login = req.session.user;

…

var payment = {

"intent": "sale",

"payer": {

},

"transactions": [{

"amount": {

"currency": currency,

"total": total,

"details": {

"subtotal": amount,

// "tax": tax,

"shipping": shipping

}

},

"description": req.body.item\_name

}]

};

if (method === 'paypal') {

payment.payer.payment\_method = 'paypal';

paypal.configure({

"host" : "api.sandbox.paypal.com",

"port" : "",

'mode': 'sandbox', //sandbox or live

'client\_id': 'EBWKjlELKMYqRNQ6sYvFo64FtaRLRR5BdHEESmha49TM',

'client\_secret': 'EO422dn3gQLgDbuwqTjzrFgFtaRLRR5BdHEESmha49TM'

});

//http:localhost:3000/credit/saveAbnmt?nouveau\_abnmt= valeur?durre=valeur

http:

payment.redirect\_urls = {

//"return\_url": "http://localhost:3000/credit/saveAbnmt/"+amount+"?durreAbnmt="+durreAbnmt,

"return\_url": "http://"+nom\_host+"/abonnement/saveAbnmt?loginUser="+login,

//"return\_url": "http://"+nom\_host+"/abonnement/saveAbnmt/"+login+"?idAbonnement="+idAbnmt,

//"return\_url": "http://localhost:3000/credit/saveCreditAchete?nouveau\_credit="+amount,

"cancel\_url": "http://"+nom\_host+"/abonnement/CancelPaiementAbnmt?loginUser="+login

};

} else if (method === 'credit\_card') {

var type = req.body.typeCB.toLowerCase(),

numero = req.body.number;

console.log(numero);

var funding\_instruments = [

{

"credit\_card": {

"type": type,

"number": numero,

"expire\_month": req.body.expire\_month,

"expire\_year": req.body.expire\_year,

"first\_name": req.body.first\_name,

"last\_name": req.body.last\_name

}

}

];

payment.payer.payment\_method = 'credit\_card';

payment.payer.funding\_instruments = funding\_instruments;

}

paypal.payment.create(payment, function (error, payment) {

if (error) {

console.log(error);

res.render('error', { 'error': error });

} else {

req.session.paymentId = payment.id;

var redirectUrl;

if(payment.payer.payment\_method == 'paypal') {

for(var i=0; i < 3; i++) {

var link = payment.links[i];

if (link.method === 'REDIRECT') {

redirectUrl = link.href;

}

}

}

res.redirect(redirectUrl);

// res.render('create', { 'payment': payment });

}

});

});

//declaration du variable du paiement paypal

var payment = {

"intent": "sale",

"payer": { },

"transactions": [{

"amount": {

"currency": currency,

"total": total,

"details": {

"subtotal": amount,

"shipping": shipping

}

},

"description": req.body.item\_name

}]

};

/\*Si le myen de payement est paypal\*/

if (method === 'paypal') {

payment.payer.payment\_method = 'paypal';

paypal.configure({

"host" : "api.sandbox.paypal.com",

"port" : "",

'mode': 'sandbox', //sandbox or live

'client\_id': 'EBWKjlELKMYqRNQ6sYvFo64FtaRLRR5BdHEESmha49TM',

'client\_secret': 'EO422dn3gQLgDbuwqTjzrFgFtaRLRR5BdHEESmha49TM'

});

payment.redirect\_urls ={"http://localhost:3000/credit/saveAbnmt/"+amount+"?durreAbnmt="+durreAbnmt,

"return\_url": "http://"+nom\_host+"/abonnement/saveAbnmt?loginUser="+login,

};

"http://localhost:3000/credit/saveCreditAchete?nouveau\_credit="+amount,

"cancel\_url": "http://"+nom\_host+"/abonnement/CancelPaiementAbnmt?loginUser="+login

};

} else if (method === 'credit\_card') {

var type = req.body.typeCB.toLowerCase(),

numero = req.body.number;

console.log(numero);

var funding\_instruments = [

{

"credit\_card": {

"type": type,

"number": numero,

"expire\_month": req.body.expire\_month,

"expire\_year": req.body.expire\_year,

"first\_name": req.body.first\_name,

"last\_name": req.body.last\_name

}

}

];

payment.payer.payment\_method = 'credit\_card';

payment.payer.funding\_instruments = funding\_instruments;

}

paypal.payment.create(payment, function (error, payment) {

if (error) {

console.log(error);

res.render('error', { 'error': error });

} else {

req.session.paymentId = payment.id;

var redirectUrl;

if(payment.payer.payment\_method == 'paypal') {

for(var i=0; i < 3; i++) {

var link = payment.links[i];

if (link.method === 'REDIRECT') {

redirectUrl = link.href;

}

}

}

res.redirect(redirectUrl);

// res.render('create', { 'payment': payment });

}

});

});

//Si le client paye par carte de crédit

} else if (method === 'credit\_card') {

var type = req.body.typeCB.toLowerCase(),

numero = req.body.number;

console.log(numero);

var funding\_instruments = [{

"credit\_card": {

"type": type,

"number": numero,

"expire\_month": req.body.expire\_month,

"expire\_year": req.body.expire\_year,

"first\_name": req.body.first\_name,

"last\_name": req.body.last\_name

} }];

payment.payer.payment\_method = 'credit\_card';

payment.payer.funding\_instruments = funding\_instruments;

}

paypal.payment.create(payment, function (error, payment) {

if (error) {

console.log(error);

res.render('error', { 'error': error });

} else {

req.session.paymentId = payment.id;

if(payment.payer.payment\_method == 'paypal') {

for(var i=0; i < 3; i++) {

var link = payment.links[i];

if (link.method === 'REDIRECT') {

redirectUrl = link.href;

}}}

…

});

});

1. **Quelques captures d’écran**

Dans cette partie, nous allons présenter quelques interfaces de l’application, répondant aux recommandations ergonomiques de compatibilité, de guidage, de clarté, d’homogénéité et de souplesse.

* + 1. **Page d’accueil de l’application**

La figure 55présente la page d’accueil de l’application du côté client.

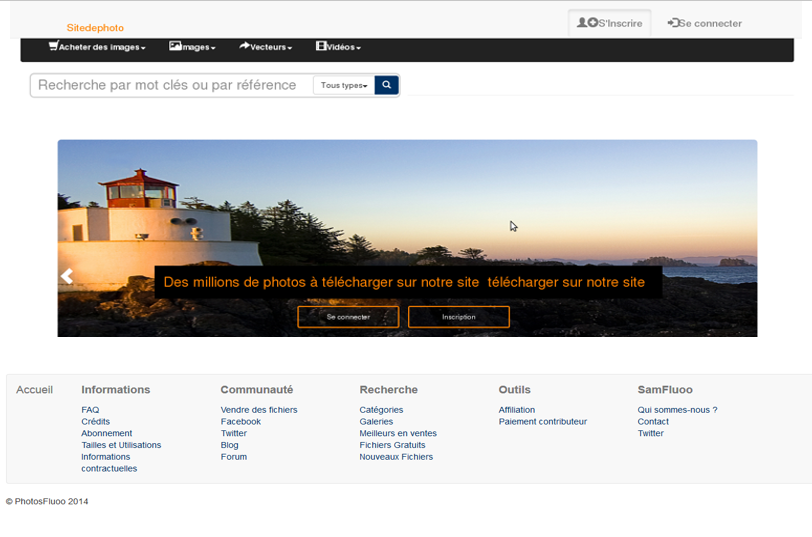


Figure 55 : Page d’accueil de l’application côté client

Cette page est affichée lors de l’ouverture du site, c’est la page d’accueil côté client. Elle donne les aperçues de tous les menus que le site propose aux internautes qui visitent l’application.

* + 1. **Page de souscription à un abonnement**

Les figures56 et 57présentent respectivement le formulaire de souscription à un abonnement et la page de validation de la souscription pour le paiement.

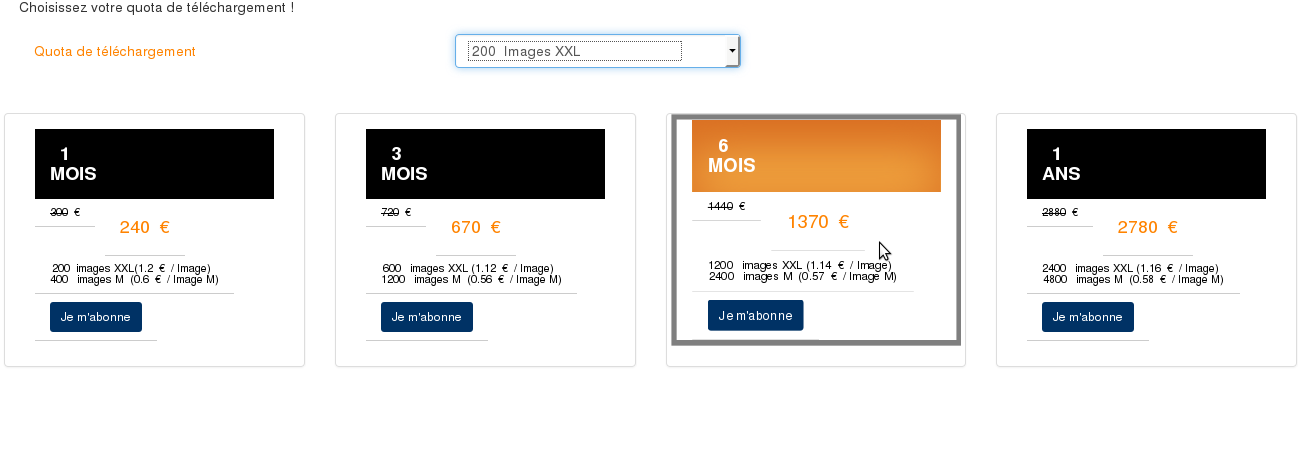


Figure 56 : Page de formulaire de souscription à un abonnement

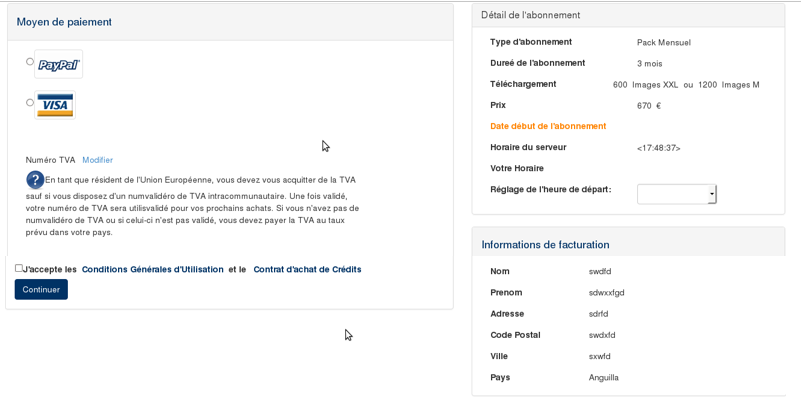


Figure 57 : Page de validation de la souscription à un abonnement

L’accès au formulaire de souscription représenté par la figure 56 ne nécessite pas d’être connecté. En outre pour souscrire à un abonnement, il faut d’abord se connecté sur son compte et puis valider les choix des différents options d’abonnement.

La validation de la souscription présentée par la figure 57 nécessite que l’utilisateur soit connecté au site et que toutes les informations de son compte soient bien remplies. En validant ses choix d’abonnement en cliquant sur le bouton « je m’abonne » de la figure 56, il va accéder au page de validation de la souscription ou il pourra choisir son mode paiement et voir d’autres côtés les différents options qu’il a choisi. D’où il pourra ensuite continuer son souscription en acceptant les conditions générales d’utilisations et en cliquant sur le bouton « continuer ».

* + 1. **Page d’achat de fichier en utilisant le panier**

La figure 58 présente l’achat de fichier en utilisant le panier.

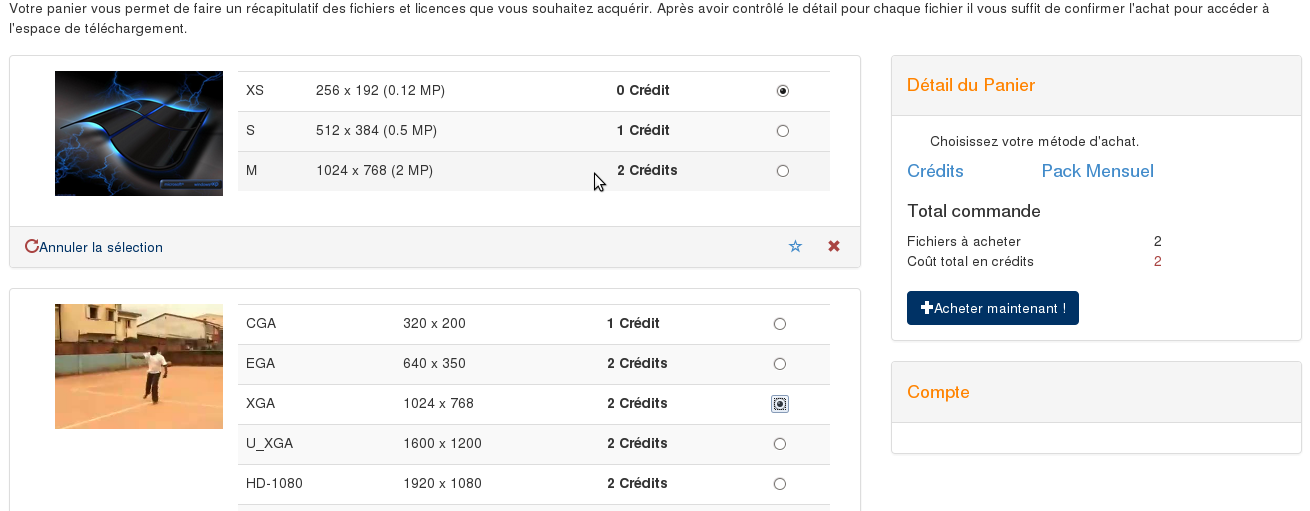


Figure 58 : Achat fichier en utilisant le panier

L’accès à cette page d’achat de fichier en utilisant le panier nécessite que l’utilisateur soit connecté.

Après avoir ajouté de fichier dans son panier, il pourra ensuite accéder à cette page. Sur cette page, on trouve une liste de fichier que l’utilisateur à ajoutés dans son panier.

En accédant à cette page, l’utilisateur peut :

* Choisir son mode d’achat ;
* L’utilisateur peut choisir la taille et le prix de fichier qu’il préféré acheter ;
* L’utilisateur peut valider son achat ;
* L’utilisateur peut retirer de fichier de son panier ;
* L’utilisateur peut annuler la sélection de son fichier.
  + 1. **Page d’achat de fichier à la volet**

La figure 59 présente la page d’achat de fichier à la volet.

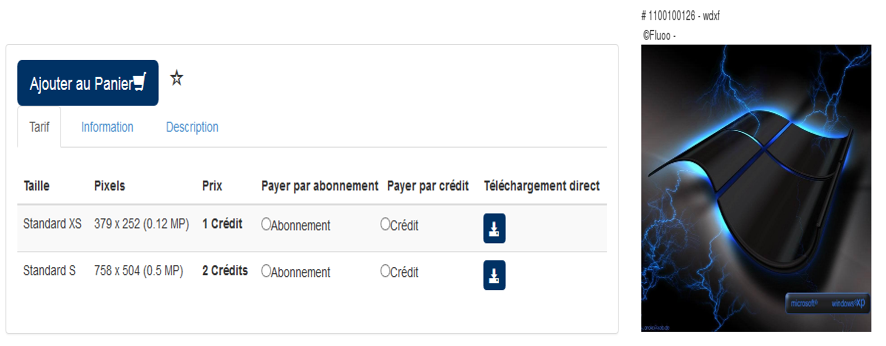


Figure 59 : Page d’accueil de l’application côté client

L’accès à cette page nécessite que l’utilisateur soit connecté sur son compte et qu’il possédé déjà des moyens de paiement (crédit ou abonnement).

Après avoir choisi un de fichier sur les listes des fichiers de fichiers existants sur le site, l’utilisateur accède à cette page.

L’utilisateur peut :

* Voir les tarifs, informations et description du fichier ;
* Choisir le moyen de paiement pour son achat à la volet ;
* Télécharger le fichier.
  + 1. **Page d’uploade de fichier effectuer par l’administrateur**

La figure 60 et 61 présentent respectivement la page d’uploade de fichier et le formulaire qui permet de remplir les informations concernant le fichier à uploader par l’administrateur du site.



Figure 60 : Page d’uploade de fichier

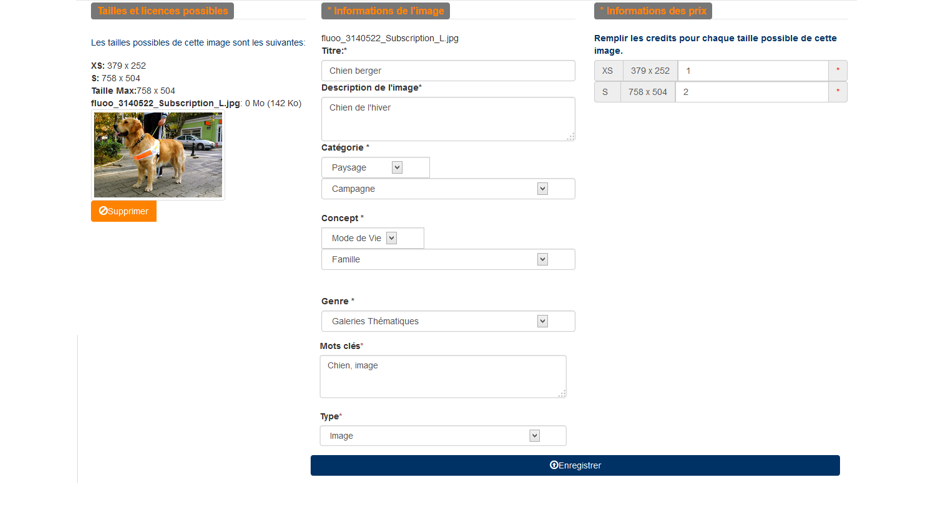


Figure 61 : Formulaire d’information du fichier

Après avoir connecté en tant qu’administrateur et après avoir choisi et parcourir le fichier, on accède directement à la page représenté par la figure 60. Un script calculer et affiche tout de suite les tailles possédés par le fichier. L’administrateur peur supprimer en cliquant sur le bouton « supprimer » le fichier ou continuer son uploade en cliquant sur le bouton « upload ».

En continuant l’uploade de fichier par l’administrateur, on accède directement sur le formulaire d’information de fichier représenté par la figure 61. Sur cette formulaire, l’administrateur peut remplir les différents informations nécessaires pour l’uploade de fichier et ensuite l’enregistrer en cliquant sur le bouton « Enregistrer ».

## Conclusion

A travers ce chapitre, nous avons présenté la réalisation de l’application en justifiant nos choix technologiques, en représentant quelques interfaces graphiques que nous avons jugé les plus importantes et en décrivant brièvement comment nous avons planifié notre projet.

CONCLUSION GENERALE

En guise de conclusion, la réalisation d’un projet nécessite avant tout une analyse bien approfondie, suivie d’une bonne conception et enfin une organisation du travail.

Notre but, tout au long de ce stage, était de concevoir et réaliser un site e-commerce responsive en utilisant la technologie NodeJS pour les clients, qui leur permettent en général de consulter et d’acheter les fichiers de la société Fluoo.

Pour atteindre ce but, la méthode d’approche agile SCRUM a été utilisée, en raison de sa performance sur l’amélioration de la productivité des équipes auparavant ralenties par des méthodologies plus lourdes. D'autre part, UML représente une réelle avancée dans le monde méthodologique de l'approche objet, et elle a été utilisée pour la notation.

Le projet a donc, abouti aux conceptions et réalisation d’un site e-commerce responsive en utilisant la technologie NodeJS sous le framework Express et Bootstrap et en utilisant aussi un compilateur CSS appelé LESS.

Pour le moment le site e-commerce est presque terminé mais fonctionnel c’est à dire à cause des besoins changeant de jour en jour du client, l’application est encore en maintenance, nous souhaitons qu’il trouvera les conditions nécessaires pour entrer en vigueur.

Ce mémoire de fin d’études est un objet de recherche au niveau de l’entreprise et aussi renforcement de connaissance que l’on a acquis de l’école.

REFERENCE BIBLIOGRAPHIQUES

Adel RAISSI, « Conception et développement d’un site web de e-commerce pour le compte de LSAT\_Nokia », Edition 2013 [1]

Claude AUBRY, « SCRUM Le guide pratique de la méthode Agile la plus populaire », Edition DUNOD, Février 2012 [2]

Danny Goodman, « JavaScript Bible », Edition EYROLLES, 2004 [3]

Michel Martin, « Simplifier vos développement javascript avec JQuery », Edition OPENCLASSROOMS, 07 janvier 2013 [4]

Par Bestmomo, « Bootstrap de Twitter : un kit CSS et plus !», Edition OPENCLASSROOMS, 12 janvier 2013 [5]

REFERENCE WEBOGRAPHIQUES

<http://fr.openclassrooms.com> [site 1]

<http://www.grafikart.fr> [site 2]

<https://www.youtube.com> [site 3]

<http://bootstrapdocs.com> [site 4]

<http://www.lafermeduweb.net> [site 5]

<https://developer.paypal.com> [site 6]

<http://www-igm.univ-mlv.fr> [site 7]

<http://www.nodejs.org> [site 8]

<http://www.wikipedia.org>[site 9]

ANNEXE : CAHIER DES CHARGES

1. **Présentation du projet**

* L’application e-commerce permettra aux utilisateurs de :
* S’inscrire au site,
* Gérer son compte,
* Consulter les fichiers existants qui sont uploadés par l’administrateur,
* Consulter les différents types d’abonnement existants,
* Souscrire à un abonnement,
* Acheter des fichiers en utilisant le panier,
* Acheter de fichier à la volet.
* Aussi, l’application va permettre à l’administrateur de :
* Gérer le compte de la société,
* Gérer les abonnements,
* Gérer les fichiers,
* Gérer les utilisateurs,
* Uploader de fichier.

1. **Objet du document**

Ce document constitue le cahier de charge pour la mise en place du site e-commerce Sam Fluoo pour la vente des fichiers en ligne.

Ce document va décrire les fonctionnalités de l’application. Il est complété par les aides d’utilisation du système.

1. **Contexte d’élaboration**

Ce présent document est basé sur les échanges entre IRE-OI et Fluoo ainsi que sur un documentation données par le client pour la réalisation du prensent projet.

Cette réalisation devra permettre :

* Au client de consulter, acheter des fichiers au sein du site e-commerce Sam Fluoo,
* A l’administrateur de charger la mise à jour de base de données, c’est-à-dire gérer les utilisateurs inscrites dans le site ; gérer les abonnements, le compte de la société et les fichiers, etc…

1. **Plan du document**

Le tableau 12 présente le plan du document.

Tableau 20 : Plan du document

|  |  |
| --- | --- |
| **Chapitre** | **Contenu** |
| Description du projet | Précise les principes directeurs du projet |
| Synthèse des besoins de Fluoo | Présente la synthèse des besoins fonctionnels et techniques. |
| Description détaillée des besoins  Fonctionnels | Décrit la liste des besoins fonctionnels sur l’ensemble des domaines d’activité du site: Visualisation des fichiers, inscription au site, souscription à un abonnement, achat des fichiers. |
| Architecture technique et matérielle | Décrit la méthode de vente des fichiers actuellement utilisé par Fluoo.  Propose les différentes solutions possibles du site e-commerce.  Décrit la cartographie applicative et les principes directeurs de l’architecture cible souhaitée. |

1. **Intevenants**

Mr Jonathan Product owner

[J.Fluoo@fr.yahoo](mailto:J.Fluoo@fr.yahoo)

Mr Pierre Luc Gervais Directeur et chef de Projet

[referenecementmadagascar@gmail](mailto:referenecementmadagascar@gmail).com

Mr Rémy Directeur technique

[Remy.fluoo@gmail.com](mailto:Remy.fluoo@gmail.com)

Mr RANDRIANARIKOTO Alexandre Boris Developpeur

[randrianarikoto@gmail.com](mailto:randrianarikoto@gmail.com)

Mr RAMPIANDRA RAVELONA Masinera Développeur

[masinera.ravelona@gmail.com](mailto:masinera.ravelona@gmail.com)

1. **Déscription des besoins de la société**

Le tableau 19 et 20 présentent respectivement les listes des bésions de la société au niveau de l’utilisateur et au niveau de l’administrateur.

Tableau 21 : Liste des fonctionnalités au niveau de l’utilisateur

|  |  |
| --- | --- |
| **Thème** | **Besoins** |
| Gestion de l’inscription | Le système permet deà l’utilisateur des’inscrire au site |
| Souscription à un abonnement | Le système permet à l’utilisateur de   * Voir le différent type d’abonnement * Choisir un type d’abonnement * Choisir les options d’abonnement préféré * Valider la souscription à un abonnement * Choisir et valider le moyen de paiement de la souscription |
| Achat de fichier à la volet | Le système permet à l’utilisateur de   * Consulter les fichiers existants dans l’application * Voir le détail d’un fichier * Choisir le moyen de paiement * Acheter à la volet le fichier * Télécharger le fichier acheté |
| Achat des fichiers utilisant le panier | Le système permet à l’utilisateur de   * Lister les fichiers existants * Ajouter des fichiers dans le panier * Lister les fichiers dans le panier * Choisir le mode d’achat * Choisir et sélectionner les tailles et les prix de préférences de chaque fichier à achetés * Annuler les sélections des fichiers * Valider l’achat des fichiers panier |
| Gestion du lightbox | Le système permet à l’utilisateur de   * D’ajouter des fichiers dans le lightbox * Lister les fichiers du lightbox * Supprimer un fichier du lightbox * Ajouter dans le panier le fichier du lightbox |
| Gestion de compte de l’utilisateur | Le système permet de   * Consulter le tableau de bord de l’utilisateur * Modifier le profil de l’utilisateur * Consulter les nouveaux messages de l’utilisateur |
| Gestion de panier | Le système permet à l’utilisateur de   * Ajouter des fichiers dans le panier * Lister les fichiers du panier * Supprimer les fichiers du panier |

Tableau 22 : Liste des fonctionnalités au niveau de l’administrateur

|  |  |
| --- | --- |
| **Thème** | **Besoins** |
| Gestion des utilisateurs | Le système permet à l’administrateur de   * Lister les utilisateurs inscrits * Voir les détails d’un utilisateur * Mettre à jour les informations concernant l’utilisateur * Supprimer un utilisateur * Ajouter un nouvel utilisateur |
| Gestion des abonnements | Le système permet à l’administrateur de   * Voir les lister les abonnements disponibles * Consulter les options d’abonnement disponibles * Editer les options d’abonnement * Ajouter un nouveau quota de téléchargement * Modifier le les bénéfices de chaque durée d’abonnement * Supprimer un quota de téléchargement |
| Gestion des fichiers | Le système permet à l’administrateur de   * Lister les fichiers enregistrés dans la base de données * Voir les détails d’un fichier * Mettre à jour les informations du fichier * Supprimer un fichier |
| Gestion le compte de la société | Le système permet à l’administrateur de   * Voir le compte de la société * Modifier les informations générales de la société * Modifier les informations du compte paypal de la société * Consulter les nouveaux messages de la société |
| Uploade de fichier | Le système permet à l’administrateur vde :   * parcourir un fichier * Remplir un formulaire permettant de remplir les informations concernant le fichier * Enregistrer les informations du fichier |

1. **Déscription du nouveau systéme**

La mise en place d’un site e-commerce devra répondre aux exigences ci-après:

* **Qualités sur le fonctionnement** : pour rendre le site plus aisément par les internautes, on devra tourner rapidement en utilisant un système pour la transformation du site en service.
* **Maquette et design**: Le futur site devra être très facile à utiliser et à maintenir.

D’où le nouveau système permet

* une gestion facile des informations,
* une possibilité de consolider les informations avec un degré de fiabilité élevé,
* une possibilité d’accès par les bénéficiaires aux informations consolidées

1. **Etapes du projet, calendrier d’action**

Pour la réalisation du site e-commerce Sam Fluoo, IREOI propose une collaboration en mode « AGILE ». Ce mode de fonctionnement sous-entend un engagementde l’IREOI par itération. Les itérations sont basées sur une liste de fonctionnalités prédéfinies.

L’offre comprendra:

* Une première phase de spécification. Ce qui permettra de définir et lister à travers un document Excel l’ensemble des fonctionnels à réaliser pour la version du produit attendu. Chaque fonctionnalité sera scindée en tâches élémentaire de développement.
* La réalisation de la prestation attendue en méthodologie agile. C'est-à-dire développement d’un certain nombre de tâches élémentaires choisis d’un commun accord avec le client. Et ceci dans des itérations qui seront de 2 semaines (10 jours de travail). Au début de chaque itération, la complexité (donc la charge) de chaque tâche élémentaire est définie par l’équipe de réalisation.
* Conseils et accompagnement au changement pendant la réalisation du produit
* Formation sur le produit final

IREOI offre en plus de cela au client un contrôle et une visibilité sur le projet et sa mise en œuvre tout au long du contrat. Et ceci grâce à des outils collaboratifs :

* Confluence ou Google docs pour la mise en place d’espace d’échange avec le client
* Skype pour la communication avec le client
* Les locaux d’IREOI pour les séances de travail

GLOSSIAIRE

**Analyse** : étude des différentes manières de mettre en œuvre un besoin informatique au sein d’un système d’information.

**Backlog de produit (Product Backlog)**: liste des fonctionnalités qui devront être réalisées par le logiciel.

**Backlog de sprint (Sprint Backlog)** : liste des tâches à accomplir pendant un sprint. Elles correspondent à la réalisation des items de backlog du produit affectés au sprint.

**Base de données** : Usuellement abrégée en BD ou BDD, est un ensemble structuré et organisé permettant le stockage de grandes quantités d’informations afin d’en faciliter l’exploitation.

**Conception** : Phase de processus du développement d’un logiciel dont l’objet principal est de décider la façon d’implémenter le système. Pendant la phase de conception, des décisions tactiques et stratégiques sont à prendre afin de satisfaire aux exigences fonctionnelles et de qualité voulues pour le système.

**Design** : (la stylique en français) est un domaine visant à la création d’objets, d’environnements ou d’œuvres graphiques, à la fois fonctionnels, esthétiques et conformes aux impératifs d’une production industrielle.

**Directeur de produit (Product Owner)** :personne responsable de produire et maintenir à jour le backlog de produit. C'est lui qui en détermine les priorités et qui prend les décisions concernant l'orientation du projet.

**E-commerce** :Commercialisation des produits sur internet.

**Chef de mélée (ScrumMaster)**: membre de l'équipe dont l'objectif principal est de la protéger des perturbations extérieures. Il est complètement transparent pour la communication entre l'équipe et les clients et n'a aucun pouvoir hiérarchique sur l'équipe. C'est en revanche un facilitateur pour les problèmes non techniques de l'équipe.

**Framework** : Un framework est un espace de travail modulaire. C’est un ensemble de bibliothèques et de conventions permettant le développement rapide d’applications. Il fournit suffisamment de briques logicielles et impose suffisamment de rigueur pour pouvoir produire une application aboutie et facile à maintenir. Ces composants sont organisés pourêtre utilisés en interaction les uns avec les autres.

**FTP** : Le File Transfer Protocol (protocole de transfert de fichiers), ou FTP, est un protocole de communication dédié à l’échange informatique de fichiers sur un réseau TCP/IP. Il permet, depuis un ordinateur, de copier des fichiers depuis ou vers un autre ordinateur du réseau, d’administrer un site web, ou encore de supprimer ou modifier des fichiers sur cet ordinateur.

**Graphique d'avancement (Burndown Chart)** : graphique qui montre la tendance du reste à faire total de jour en jour (pour les sprints) ou de sprint en sprint (pour les releases).

**HTML** : Un langage informatique de balisage conçu pour écrire les pages Web, et notamment pour créer de l’hypertexte, d’où son nom. HTML permet aussi de structurer sémantiquement et de mettre en page le contenu des pages, d’inclure des ressources multimédias dont des images, des formulaires de saisie, Ingénierie Informatique :] (Ou Génie informatique), est une discipline qui traite de la conception, du développement et de la fabrication de systèmes informatiques, à la fois matériels et logiciels.

**Internet** : Internet est un réseau mondial qui relie des milliers de réseaux indépendants. Tous les réseaux sont interconnectés entre eux de la même manière et peuvent s’échanger des informations.

**Mêlée quotidienne (Daily Scrum)**: réunion quotidienne de 15 minutes qui a pour but de faire le point sur ce qui a été fait depuis la dernière mêlée, ce qui est prévu de faire jusqu'à la prochaine et quelles sont les embûches rencontrées durant le travail.

**Modèle** : Représentation simplifiée de la réalité étudiée. Quel que soit son niveau de complexité, un bon modèle doit être aussi fidèle que possible à l’objet ou au système modéliser.

**Scrum**:terme emprunté au rugby à XV et signifie mêlée

**Sprint**: nom d'une itération dans Scrum. Cette itération dure 30 jours calendaires en théorie, mais en pratique on utilise plutôt entre 2 et 4 semaines. Pendant une itération, l'équipe doit développer une liste d'items du backlog de produit qui a été définie au début de ce sprint.

**Quota** : Quantité allouée pour une période ou temps de déterminé.

**Userstory** : Un point particulier d'une feature. Cela pourrait correspondre à "Gérer l'authentification" qui est une partie intégrée à "Gérer les utilisateurs et leurs droits sur l'application", ladite feature.

**La roadmap ou timeline** : Un planning découpé par des releases. Contrairement aux plannings des démarches prédictives, les charges de travail ne rentrent pas en compte.

**La release** : La release est un laps de temps de plusieurs mois, défini initialement avec le client. Il y en a donc plusieurs dans un projet. A chaque fin de release, une livraison d'une partie du projet pleinement fonctionnelle est à effectuer. Il est nécessaire de déterminer avant chaque début de release, les features qui devront être livrées.

TABLE DE MAIERES

[AVANT-PROPOS i](#_Toc413368404)

[REMERCIEMENTS ii](#_Toc413368405)

[CURRICULUM VITAE iii](#_Toc413368406)

[SOMMAIRE vi](#_Toc413368407)

[LISTE DES FIGURES ix](#_Toc413368408)

[LISTE DES TABLEAUX xi](#_Toc413368409)

[LISTE DES ABEVIATIONS xii](#_Toc413368410)

[LISTE DES ANNEXE xiii](#_Toc413368411)

[INTRODUCTION GENERALE 1](#_Toc413368412)

[**Partie I.** **PRESENTATIONS** 2](file:///F:\Stage\Livre\MEMOIRE-FIN-D'ETUDE-RANDRIANARIKOTO-ALEXANDRE.docx#_Toc413368413)

[Chapitre 1 PRESENTATION DE L’ECOLE NATIONALE DE L’INFORMATIQUE 3](#_Toc413368414)

[**1.1.** **Historique** 3](#_Toc413368415)

[**1.2.** **Organigramme** 4](#_Toc413368416)

[**1.3.** **Domaines de spécialisation** 4](#_Toc413368417)

[**1.4.** **Architecture de la pédagogie** 5](#_Toc413368418)

[**1.5.** **Filières de formation existantes et diplômes délivrés** 6](#_Toc413368419)

[**1.6.** **Partenariat** 7](#_Toc413368420)

[**1.6.1.** **Au niveau national** 7](#_Toc413368421)

[**1.6.2.** **Au niveau international** 8](#_Toc413368422)

[**1.7.** **Projet et perspectives de developpement institutionnel** 9](#_Toc413368423)

[**1.8.** **Ressources humaines** 9](#_Toc413368424)

[**1.9.** **Missions** 9](#_Toc413368425)

[Chapitre 2 PRESENTATION DE L’ENTREPRISE IRE-OI (Internet Révolution Evolution de l’Océan Indien ) 11](#_Toc413368426)

[**2.1.** **Historique** 11](#_Toc413368427)

[**2.2.** **Fiche d’Identification** 11](#_Toc413368428)

[**2.3.** **Organisation (organigramme général)** 12](#_Toc413368429)

[**2.4.** **Objectifs : Mission / Activités** 12](#_Toc413368430)

[**2.4.1.** **Développement Web** 12](#_Toc413368431)

[**2.4.2.** **Rédaction web** 13](#_Toc413368432)

[**2.4.3.** **Référencement naturel** 13](#_Toc413368433)

[**2.4.4.** **Social media marketing** 13](#_Toc413368434)

[**2.4.5.** **Call center** 14](#_Toc413368435)

[**2.5.** **Patrimoine** 14](#_Toc413368436)

[Chapitre 3 PRESENTATION GENERALE DU PROJET 15](#_Toc413368437)

[**Introduction** 15](#_Toc413368438)

[**3.1.** **Objectif et besoins de l’utilisateur** 15](#_Toc413368439)

[**3.1.1.** **Objectif** 15](#_Toc413368440)

[**3.1.2.** **Besoins des utilisateurs** 15](#_Toc413368441)

[**3.2.2.1.** **Les besoins fonctionnels** 15](#_Toc413368442)

[**3.2.2.2.** **Les besoins non fonctionnels** 16](#_Toc413368443)

[**3.2.** **Moyens nécessaires à la réalisation du projet (Humain, matériel et éventuellement financier)** 17](#_Toc413368444)

[**3.3.** **Résultats attendus** 17](#_Toc413368445)

[**3.4.** **Présentation de la société Fluoo** 18](#_Toc413368446)

[**3.4.1.** **Historique** 18](#_Toc413368447)

[**3.4.2.** **Missions et objectifs** 18](#_Toc413368448)

[**3.4.3.** **Les différentes prestations fournies  par Fluoo** 18](#_Toc413368449)

[**Conclusion** 18](#_Toc413368450)

[**Partie II.** **ANALYSE ET CONCEPTION** 19](file:///F:\Stage\Livre\MEMOIRE-FIN-D'ETUDE-RANDRIANARIKOTO-ALEXANDRE.docx#_Toc413368451)

[Chapitre 4 ANALYSE PREALABLE 20](#_Toc413368452)

[**Introduction** 20](#_Toc413368453)

[**4.1.** **Etude de l’existant** 20](#_Toc413368454)

[**4.1.1.** **Description de l’existant** 20](#_Toc413368455)

[**4.2.** **Critique de l’existant** 21](#_Toc413368456)

[**4.3.** **Solution proposée** 21](#_Toc413368457)

[**4.4.** **Modéles et méthodologie adoptées** 21](#_Toc413368458)

[**4.4.1.** **Introduction aux approches Agiles** 22](#_Toc413368459)

[**4.4.2.** **Scrum [2]** 24](#_Toc413368460)

[**4.4.2.1.** **Caractéristiques** 24](#_Toc413368461)

[**4.4.2.2.** **Idées clé** [2] 25](#_Toc413368462)

[**4.4.2.3.** **Rôles** 25](#_Toc413368463)

[**a.** **Directeur de produit** 25](#_Toc413368464)

[**b.** **Équipe** 25](#_Toc413368465)

[**c.** **Facilitateur / Animateur** 26](#_Toc413368466)

[**d.** **Intervenants** 26](#_Toc413368467)

[**4.5.2.1.** **Planification** 26](#_Toc413368468)

[**a.** **Sprints** 26](#_Toc413368469)

[**b.** **Releases** 27](#_Toc413368470)

[**c.** **Quotidien** 27](#_Toc413368471)

[**4.4.2.4.** **Gestion des besoins** 27](#_Toc413368472)

[**a.** **Backlog de produit** 27](#_Toc413368473)

[**b.** **Backlog de sprint** 28](#_Toc413368474)

[**4.4.2.5.** **Estimations** 28](#_Toc413368475)

[**a.** **Items de backlog de produit** 28](#_Toc413368476)

[**b.** **Calcul de vélocité** 29](#_Toc413368477)

[**c.** **Items de backlog de sprint** 29](#_Toc413368478)

[**4.4.2.6.** **Documentation de projet** 29](#_Toc413368479)

[**4.4.3.** **UML** 30](#_Toc413368480)

[**4.4.3.1.** **Les diagrammes de cas d’utilisation** 31](#_Toc413368481)

[**4.4.3.2.** **Les diagrammes d’activité** 32](#_Toc413368482)

[**4.4.3.3.** **Les diagrammes de séquence** 33](#_Toc413368483)

[**4.4.3.4.** **Les diagrammes de classe** 35](#_Toc413368484)

[Conclusion 36](#_Toc413368485)

[Chapitre 5 ANALYSE CONCEPTUELLE 37](#_Toc413368486)

[**Introduction** 37](#_Toc413368487)

[**5.1.** **Indentification des fonctionnalités (Features)** 37](#_Toc413368488)

[**5.2.** **Les user stories** 38](#_Toc413368489)

[**5.3.** **Planification des sprints** 44](#_Toc413368490)

[**5.3.1.** **Planification de l’itération 1** 44](#_Toc413368491)

[**5.3.2.** **Planification de l’itération 2** 44](#_Toc413368492)

[**5.3.3.** **Planification de l’itération 3** 45](#_Toc413368493)

[**5.3.4.** **Planification de l’itération 4** 45](#_Toc413368494)

[**5.3.5.** **Planification de l’itération 5** 46](#_Toc413368495)

[**5.3.6.** **Planification de l’itération 6** 46](#_Toc413368496)

[**5.3.7.** **Planification de l’itération 7** 47](#_Toc413368497)

[**5.3.8.** **Planification de l’itération 8** 47](#_Toc413368498)

[**5.3.9.** **Planification de l’itération 9** 48](#_Toc413368499)

[**5.3.10.** **Planification de l’itération 10** 48](#_Toc413368500)

[**5.3.11.** **Planification de l’itération 11** 49](#_Toc413368501)

[**5.3.12.** **Planification de l’itération 12** 49](#_Toc413368502)

[**5.3.13.** **Planification de l’itération 13** 50](#_Toc413368503)

[**5.3.14.** **Planification de l’itération 14** 50](#_Toc413368504)

[**5.1.** **Diagrammes des cas d’utilisation globale** 51](#_Toc413368505)

[**5.1.1.** **Identification des acteurs** 51](#_Toc413368506)

[**5.1.2.** **Diagramme de cas d’utilisation globale de l’application** 51](#_Toc413368507)

[**5.2.** **Etablissement des diagrammes de classe globale utilisé dans les différents sprints** 52](#_Toc413368508)

[**5.2.1.** **Régles de gestion** 52](#_Toc413368509)

[**5.2.2.** **Dictionnaire des données de l’application** 52](#_Toc413368510)

[**5.2.3.** **Diagramme de classe globale de l’application** 55](#_Toc413368511)

[**5.2.4.** **Liste des Tables sur l’application serveur** 57](#_Toc413368512)

[**5.3.** **Conception détaillée par sprint** 57](#_Toc413368513)

[**5.3.1.** **Itération 1 «L’inscription au site»** 57](#_Toc413368514)

[**5.3.1.1.** **Diagramme de cas d’utilisation «Inscription»** 57](#_Toc413368515)

[**5.3.1.2.** **Diagramme d’activité «Inscription»** 59](#_Toc413368516)

[**5.3.1.3.** **Diagramme de séquence «Inscription»** 60](#_Toc413368517)

[**5.3.2.** **Itération 2 «La gestion des utilisateurs»** 60](#_Toc413368518)

[**5.3.2.1.** **Diagramme de cas d’utilisation «Gérer utilisateur»** 60](#_Toc413368519)

[**5.3.2.2.** **Diagramme d’activité «Gérer utilisateur»** 61](#_Toc413368520)

[**5.3.2.3.** **Diagramme de séquence «Gérer utilisateur»** 62](#_Toc413368521)

[**5.3.3.** **Itération 3 «La gestion des abonnements»** 63](#_Toc413368522)

[**5.3.3.1.** **Diagramme de cas d’utilisation «Gérer abonnement»** 63](#_Toc413368523)

[**5.3.3.2.** **Diagramme d’activité «Gérer abonnement»** 65](#_Toc413368524)

[**5.3.3.3.** **Diagramme de séquence «Gérer abonnement»** 66](#_Toc413368525)

[**5.3.4.** **Itération 4 «La souscriptionà un abonnement»** 66](#_Toc413368526)

[**5.3.4.1.** **Diagramme de cas d’utilisation «Souscription à un abonnement»** 66](#_Toc413368527)

[**5.3.4.2.** **Diagramme d’activité «Acheter abonnement»** 68](#_Toc413368528)

[**5.3.4.3.** **Diagramme de séquence «Souscrire à un abonnement»** 69](#_Toc413368529)

[**5.3.5.** **Itération 5 «La gestion des fichiers»** 70](#_Toc413368530)

[**5.3.5.1.** **Diagramme de cas d’utilisation «Gérer fichier»** 70](#_Toc413368531)

[**5.3.5.2.** **Diagramme d’activité «Gérer fichier»** 71](#_Toc413368532)

[**5.3.5.3.** **Diagramme de séquence «Gérer fichier»** 71](#_Toc413368533)

[**5.3.6.** **Itération 6 «L’achat de fichier à la volet»** 72](#_Toc413368534)

[**5.3.6.1.** **Diagramme de cas d’utilisation «Acheter fichier à la volet»** 72](#_Toc413368535)

[**5.3.6.2.** **Diagramme d’activité «Acheter fichier à la volet»** 73](#_Toc413368536)

[**5.3.6.3.** **Diagramme de séquence «Acheter fichier à la volet»** 75](#_Toc413368537)

[**5.3.7.** **Itération 7 «L’achat de fichier utilisant le panier»** 76](#_Toc413368538)

[**5.3.7.1.** **Diagramme de cas d’utilisation «Acheter du fichier en utilisant le panier»** 76](#_Toc413368539)

[**5.3.7.2.** **Diagramme d’activité «Acheter du fichier en utilisant le panier»** 77](#_Toc413368540)

[**5.3.7.3.** **Diagramme de séquence «Acheter du fichier en utilisant le panier»** 79](#_Toc413368541)

[**5.3.8.** **Itération 8 «La gestion du lightbox»** 80](#_Toc413368542)

[**5.3.8.1.** **Diagramme de cas d’utilisation «Gérer lightbox»** 80](#_Toc413368543)

[**5.3.8.2.** **Diagramme d’activité «Gérer lightbox»** 80](#_Toc413368544)

[**5.3.8.3.** **Diagramme de séquence «Gérer lightbox»** 81](#_Toc413368545)

[**5.3.9.** **Itération 9 «La gestion du compte de la société»** 82](#_Toc413368546)

[**5.3.9.1.** **Diagramme de cas d’utilisation «Gérer le compte de la société»** 82](#_Toc413368547)

[**5.3.9.2.** **Diagramme d’activité «Gérer le compte de la société»** 83](#_Toc413368548)

[**5.3.9.3.** **Diagramme de séquence «Gérer le compte de la société»** 84](#_Toc413368549)

[**5.3.10.** **Itération 10 «La gestion du compte de l’utilisateur»** 84](#_Toc413368550)

[**5.3.10.1.** **Diagramme de cas d’utilisation «Gérer le compte de l’utilisateur»** 84](#_Toc413368551)

[**5.3.10.2.** **Diagramme d’activité «Gérer le compte de l’utilisateur»** 85](#_Toc413368552)

[**5.3.10.3.** **Diagramme de séquence «Gérer le compte de l’utilisateur»** 86](#_Toc413368553)

[**5.3.11.** **Itération 11 «La gestion de panier»** 88](#_Toc413368554)

[**5.3.11.1.** **Diagramme de cas d’utilisation «Gérer le panier»** 88](#_Toc413368555)

[**5.3.11.2.** **Diagramme d’activité «Gérer le panier»** 89](#_Toc413368556)

[**5.3.11.3.** **Diagramme de séquence «Gérer le panier»** 90](#_Toc413368557)

[**5.3.12.** **Itération 12 «L’uploade des fichiers»** 91](#_Toc413368558)

[**5.3.12.1.** **Diagramme de cas d’utilisation «Uploader fichiers»** 91](#_Toc413368559)

[**5.3.12.2.** **Diagramme d’activité «Uploader fichiers»** 92](#_Toc413368560)

[**5.3.12.3.** **Diagramme de séquence «Uploader fichiers»** 92](#_Toc413368561)

[**5.3.13.** **Itération 13 «La création de sous compte»** 93](#_Toc413368562)

[**5.3.13.1.** **Diagramme de cas d’utilisation «Créer sous compte»** 93](#_Toc413368563)

[**5.3.13.2.** **Diagramme d’activité «Créer sous compte»** 94](#_Toc413368564)

[**5.3.13.3.** **Diagramme de séquence «Créer sous compte»** 94](#_Toc413368565)

[**5.3.14.** **Itération 14 «La mise à jour de l’abonnement»** 95](#_Toc413368566)

[**5.3.14.1.** **Diagramme de cas d’utilisation «Mise à jour de l’abonnement»** 95](#_Toc413368567)

[**5.3.14.2.** **Diagramme d’activité «Créer sous compte»** 97](#_Toc413368568)

[**5.3.14.3.** **Diagramme de séquence «Mise à jour de l’abonnement»** 99](#_Toc413368569)

[Conclusion 100](#_Toc413368570)

[**Partie III.** **REALISATIOIN** 101](file:///F:\Stage\Livre\MEMOIRE-FIN-D'ETUDE-RANDRIANARIKOTO-ALEXANDRE.docx#_Toc413368571)

[Chapitre 6 REALISATION 102](#_Toc413368572)

[**Introduction** 102](#_Toc413368573)

[**6.1.** **Choix des technologies** 102](#_Toc413368574)

[**6.1.1.** **Technologies Client** 102](#_Toc413368575)

[**6.1.1.1.** **Choix de la technologie NodeJS** 102](#_Toc413368576)

[**6.1.1.1.2.** **Historique** 102](#_Toc413368577)

[**6.1.1.1.3.** **Quelques avantages de l’utilisation de la technologie NodeJS** 102](#_Toc413368578)

[**6.1.1.1.4.** **Inconvénients** 103](#_Toc413368579)

[**6.1.1.1.5.** **Ses qualités** 103](#_Toc413368580)

[**6.1.1.1.6.** **Le moteur V8** [site 7]. 103](#_Toc413368581)

[**6.1.1.1.7.** **Installation** 104](#_Toc413368582)

[**6.1.2.** **Technologies Serveur** 104](#_Toc413368583)

[**6.1.2.1.** **Choix du Framework Express** 104](#_Toc413368584)

[**6.1.2.1.1.** **Généralité** 104](#_Toc413368585)

[**6.1.2.1.2.** **Structure du répertoire** 104](#_Toc413368586)

[**6.1.2.2.** **Choix du Framework Bootstrap** 106](#_Toc413368587)

[**6.1.2.2.1.** **Généralité** 106](#_Toc413368588)

[**6.1.2.2.2.** **Contenu du kit** 106](#_Toc413368589)

[**6.1.2.3.** **LESS : Compilateur CSSS** 107](#_Toc413368590)

[**6.1.2.3.1.** **Généralité** 107](#_Toc413368591)

[**6.1.2.3.2.** **Comparaison avec Sass** 108](#_Toc413368592)

[**6.1.2.3.3.** **Utilisation** 108](#_Toc413368593)

[**6.1.2.4.** **Serveur de bases de données : MySQL** 108](#_Toc413368594)

[**Remarque** 108](#_Toc413368595)

[**6.1.2.4.1.** **Quelques avantages de MySQL** 109](#_Toc413368596)

[**6.1.2.4.2.** **Performances de MySQL** 109](#_Toc413368597)

[**6.2.** **Architecture finale de l’application** 111](#_Toc413368598)

[**6.3.** **Environnement d’implémentation** 111](#_Toc413368599)

[**6.3.1.** **Moyen matériel** 111](#_Toc413368600)

[**6.3.2.** **Outils et logiciels utilisés** 112](#_Toc413368601)

[**6.4.** **Extraits de code** 113](#_Toc413368602)

[**6.4.1.** **Point d’entrée et dépendance de l’application** 113](#_Toc413368603)

[**6.4.2.** **Définition de routes utilisées par l’application, configuration de la connexion au base de données et appellation des modules nécessaires** 114](#_Toc413368604)

[**6.4.3.** **Validation de l’abonnement et envoi des données à traités au paypal** 116](#_Toc413368605)

[**6.5.** **Quelques captures d’écran** 119](#_Toc413368606)

[**6.5.1.** **Page d’accueil de l’application** 119](#_Toc413368607)

[**6.5.2.** **Page de souscription à un abonnement** 119](#_Toc413368608)

[**6.5.3.** **Page d’achat de fichier en utilisant le panier** 121](#_Toc413368609)

[**6.5.4.** **Page d’achat de fichier à la volet** 122](#_Toc413368610)

[**6.5.5.** **Page d’uploade de fichier effectuer par l’administrateur** 123](#_Toc413368611)

[Conclusion 124](#_Toc413368612)

[CONCLUSION GENERALE 126](#_Toc413368613)

[REFERENCE BIBLIOGRAPHIQUES xiii](#_Toc413368614)

[REFERENCE WEBOGRAPHIQUES xiv](#_Toc413368615)

[ANNEXE : CAHIER DES CHARGES viii](#_Toc413368616)

[**1.** **Présentation du projet** viii](#_Toc413368617)

[**2.** **Objet du document** viii](#_Toc413368618)

[**3.** **Contexte d’élaboration** viii](#_Toc413368619)

[**4.** **Plan du document** ix](#_Toc413368620)

[**5.** **Intevenants** ix](#_Toc413368621)

[**6.** **Déscription des besoins de la société** ix](#_Toc413368622)

[**7.** **Déscription du nouveau systéme** xii](#_Toc413368623)

[**8.** **Etapes du projet, calendrier d’action** xii](#_Toc413368624)

[GLOSSIAIRE xiv](#_Toc413368625)

[TABLE DE MAIERES xvii](#_Toc413368626)

[RESUME xxiv](#_Toc413368627)

[ABSTRACT xxv](#_Toc413368628)

RESUME

Actuellement la technologie web s’évolue de jours en jours, par exemple, le concept Web 2, l’arrivé de l’HTML5 et CSS3 a pris l’intention des informaticiens. La technologie à fait un départ sur le site statique et par la suite vers le site dynamique en architecture client-serveur et base de données et par suite des événements qui se produisent une nouvelle technologie est apparu : NodeJS.

Notre projet intitulé « Conception et réalisation d’un site -commerce responsive en utilisant la technologie NodeJS pour le compte de Fluoo » a pour objectif de développer un site web dynamique qui permet de visualiser et acheter les fichiers existants dans le site via n’importe quel diapositives qui utilisent des navigateurs web.

Pour ce faire, dans un premier temps, nous avons effectué une étude préalable du système qui nous a permis d’avancer quelques critiques avec des solutions envisageables.

Par la suite, nous sommes passés à l’étude détaillée au cours de laquelle nous avons élaboré les dossiers de conception du projet en faisant appel à la méthode SCRUM couplé avec la notation UML.

Finalement, nous avons effectué une technique relative à la réalisation de ce projet en utilisant leFrameworkExpress et Bootstrap qui est couplé avec le compilateur CCS appelé LESS et MySQL comme serveur de base de données.

Motsclès:NodeJS, SCRUM, MYSQL, Web 2, LESS, Bootstrap

ABSTRACT

Currently technology Web is evolved of days in days, for example, the concept Web 2, arrived of the HTML5 and CSS3 took the intention of the data processing specialists. Technology with fact a departure on the static site and thereafter towards the dynamic site in customer-server architecture and bases data and in consequence of the events which occur a new technology appeared: NodeJS.

Our project entitled " Design and realization of a site - trade in reply by using NodeJS technology for the account of Fluoo " aims to develop a dynamic Web site which makes it possible to visualize and buy the existing files in the site via any slides which use navigators Web.

With this intention, initially, we carried out a preliminary study of the system which enabled us to advance some criticisms with possible solutions.

Thereafter, we passed being studied detailed during which we worked out the files of design of the project by calling upon method SCRUM coupled with notation UML.

Finally, we carried out a technique relating to the realization of this project by using Framework Express and Bootstrap which is coupled with compiler CCS called LESS and MySQL like database server.

Key words: NodeJS, Scrum, Mysql, Web 2, Less, Bootstrap