

Université Abderrahmane Mira Bejaia

Faculté Des Sciences Exactes

Département d'Informatique

Soutenance de Licence Académique



Conception et réalisation d'une logithèque pour l'université de Bejaïa

Proposé et dirigé par :

❖ BOUZIDI L'hadi

Présenté par :

❖ BOULEHBAL Mahmoud

❖ MAMMERI Souhila

❖ MAZARI Ahmed

❖ MAZOUZ Asma

Promotion
2013

Introduction

- ❖ Objectif du projet.
- ❖ la méthode de développement proposée par Pascal Roques.
- ❖ Le projet est décomposer en chapitres comme suit:
 - Concepts théoriques abordés.
 - L'expression initiale des besoins.
 - La spécification des besoins –Cas d'utilisation et IHM.
 - La spécification détaillée des besoins-DSS.
 - Analyse.
 - Conception objet préliminaire - DSD.
 - Conception objet détaillée et réalisation

Chapitre.1 – Concepts théoriques abordés

PFE - Logithèque

Introduction

Chap.1 - Concepts théoriques : UML + Méthode de développement d'app. Web

Spécification fonctionnelle

Chap.2 - Expression initiale des besoins

Chap.3 - Spécification des besoins - Cas d'utilisation et IHM

Chap.4 - Spécification détaillée des besoins - DSS

Spécification structurelle

Chap.5 - Analyse

Chap.6 - Conception objet préliminaire : DSD

Chap.7 - Conception objet détaillée et réalisation

Conclusion (bilan et perspectives)

Concepts théoriques abordés

■ UML

Pascal Roques dans son ouvrage intitulé « UML2, modéliser une application web » définit UML comme un langage de modélisation graphique et textuel, destiné à comprendre et décrire des besoins, concevoir des solutions et communiquer des points de vue.

■ Les diagrammes qu'on a utiliser sont :

- Diagrammes de cas d'utilisation
- Diagrammes de classe
- Diagrammes d'interaction
- Diagramme de déploiement

Concepts théoriques abordés

■ Processus de développement d'application Web

Un processus de développement est un enchainement d'actions que devrait suivre un concepteur afin de répondre le mieux aux exigences d'un client.

■ Le processus de développement qu'on a utiliser est :

- Conduit par les cas d'utilisation.
- Léger, mais ne néglige pas les activités d'analyse et de conception
- Fondé sur l'utilisation d'un sous-ensemble des diagrammes du langage UML.

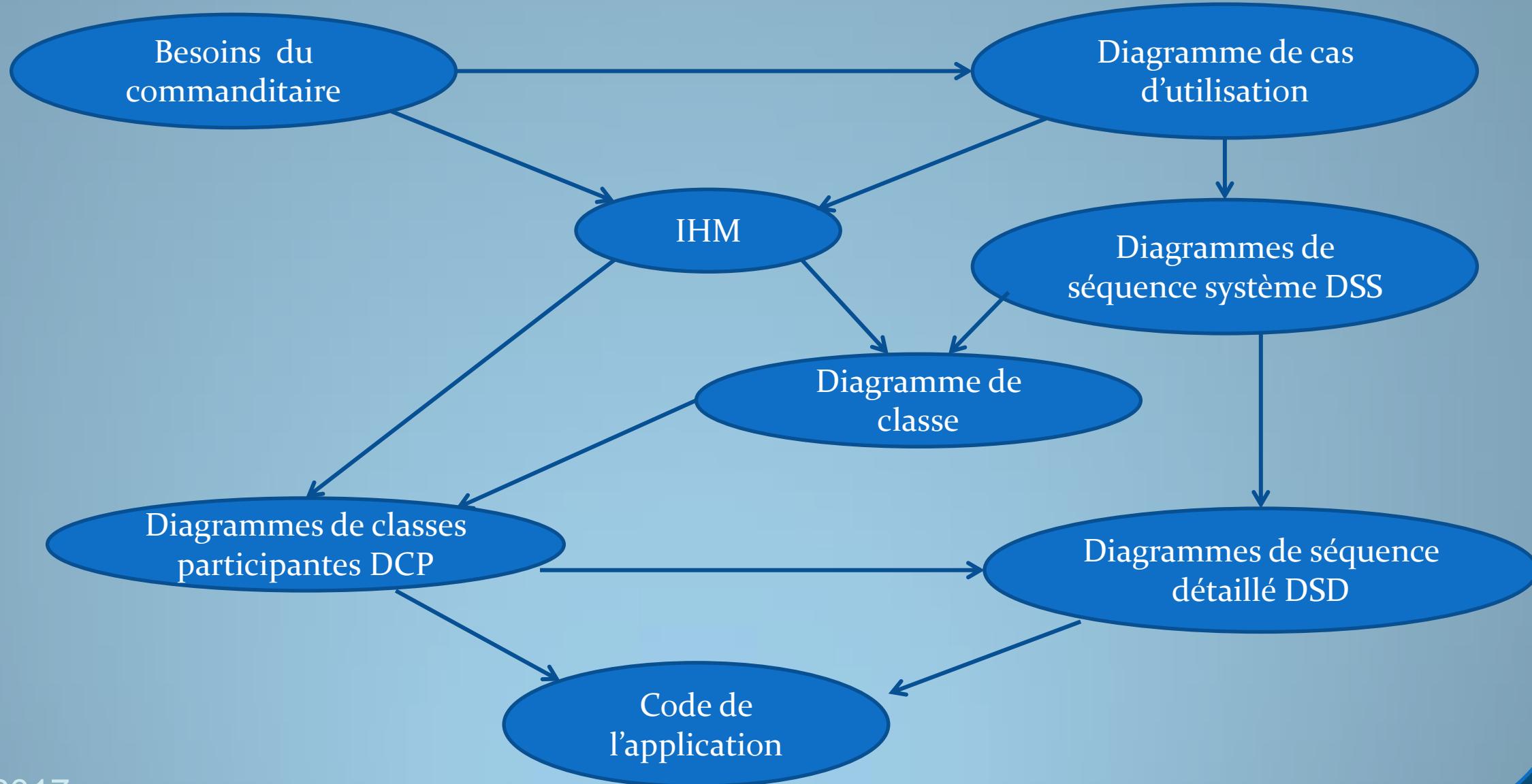
Concepts théoriques abordés

- Processus de développement d'application Web

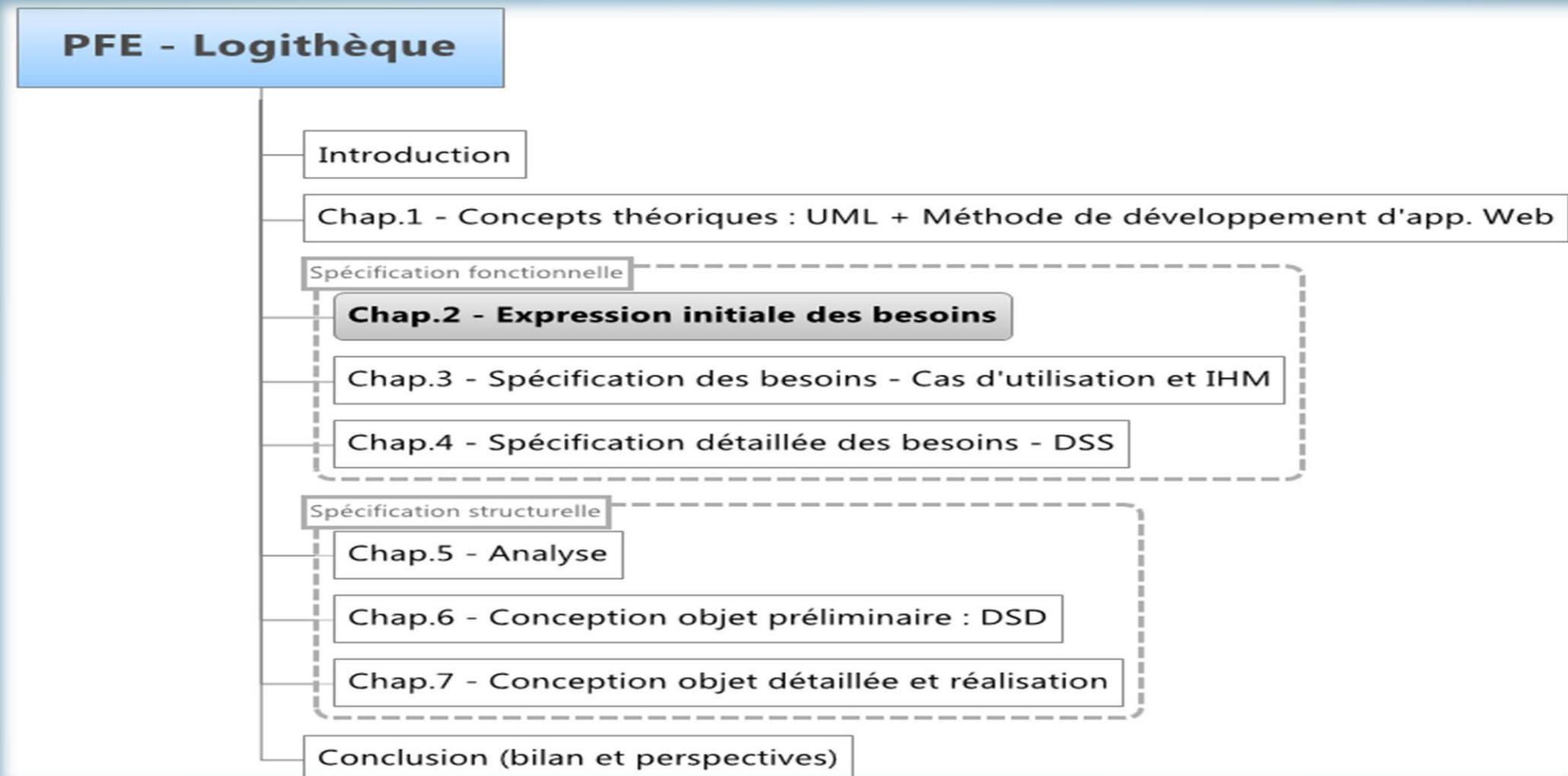


Comment passer des besoins des utilisateurs au code de l'application?

Concepts théoriques abordés



Chapitre.2 – Expression initiale des besoins



Expression initiale des besoins

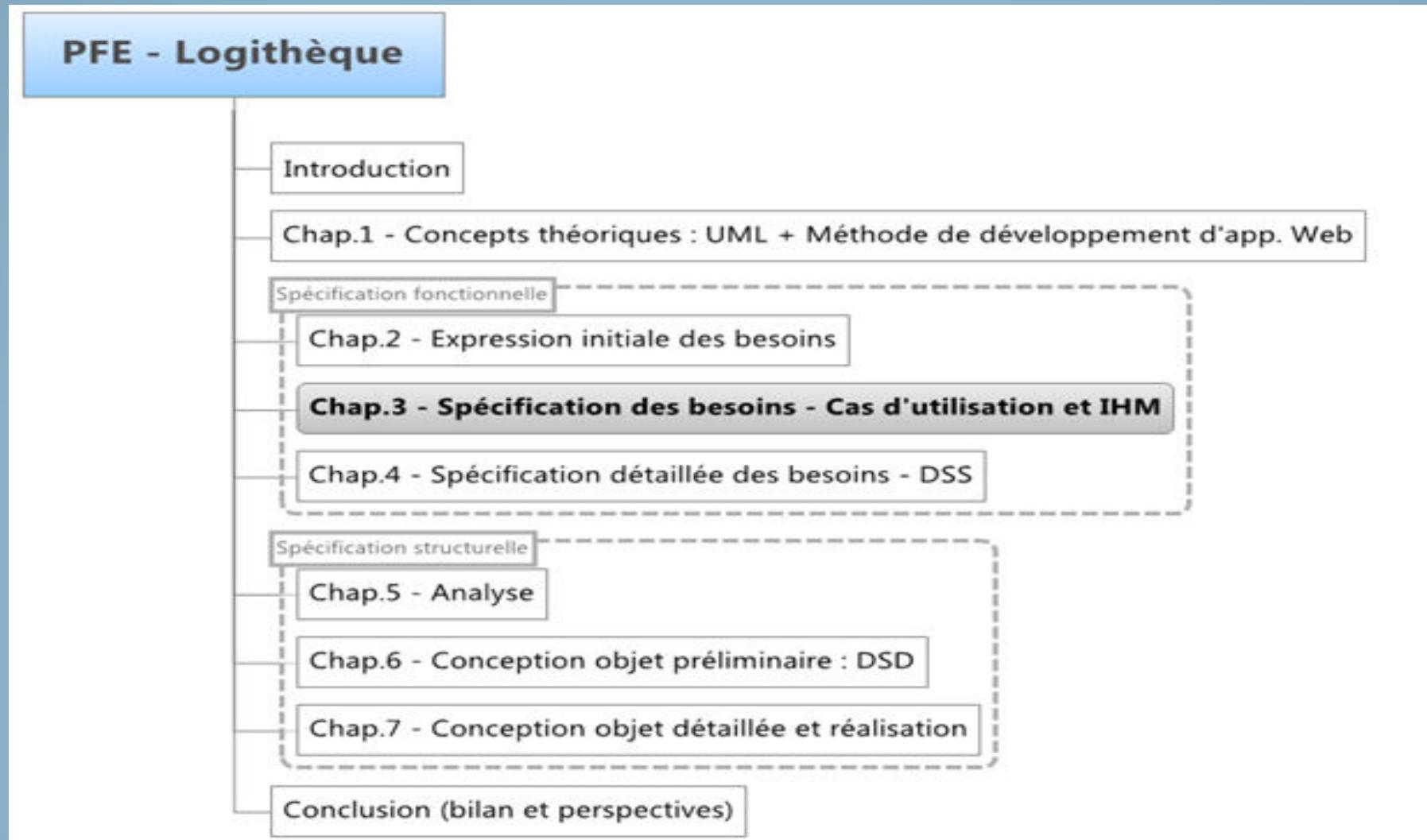
❖ Besoins fonctionnels

- Les utilisateurs de la logithèque.
- L'organisation de logiciels, catégories.
- La newsletter et les articles.

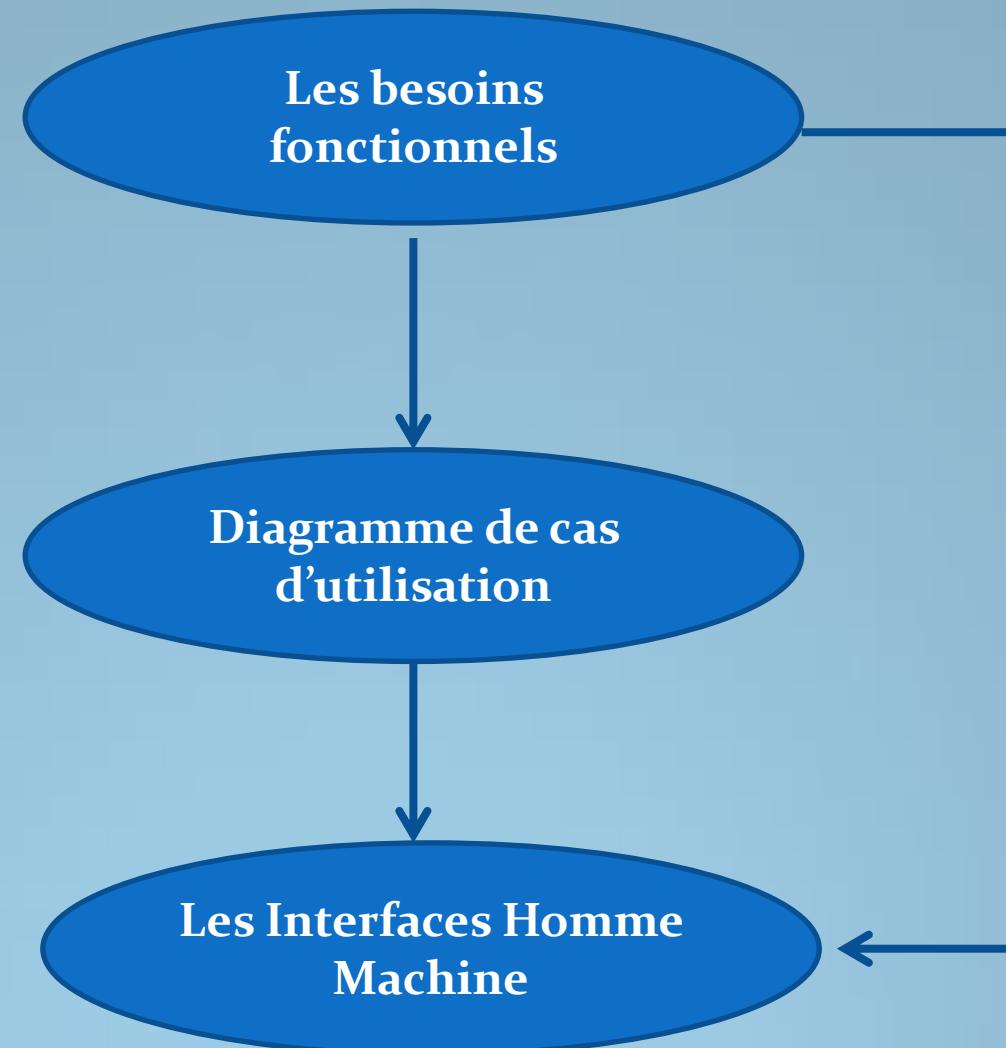
❖ Besoins non fonctionnels

- Simplicité du code source.
- Convivialité et facilité d'utilisation de la logithèque.
- Cohérence de données.

Chapitre 3 Spécification des besoins : Cas d'utilisation et IHM



Spécification des besoins : Cas d'utilisation- IHM



Spécification des besoins

Cas d'utilisation



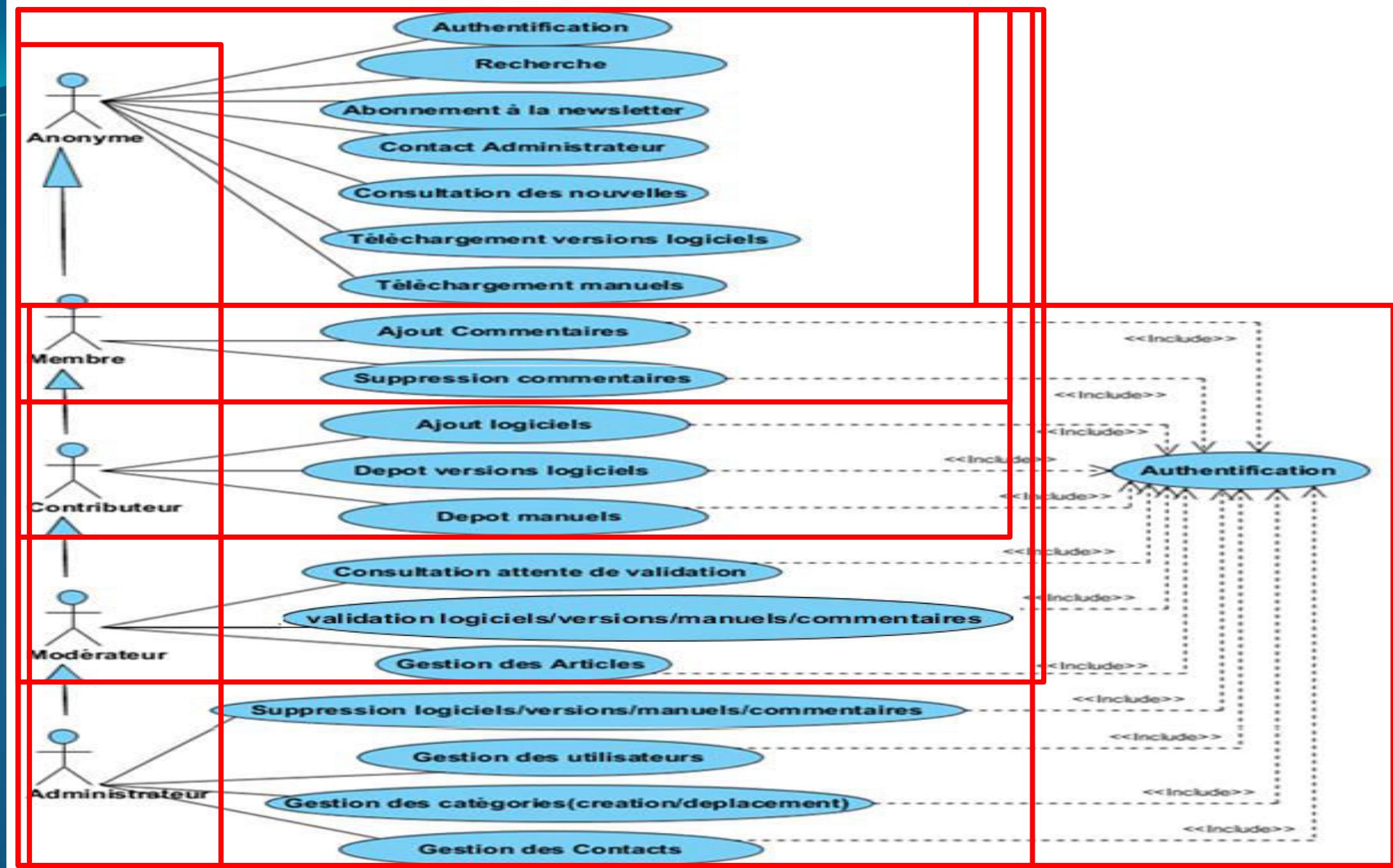
Notre application possède 5 types d'acteurs :

- Administrateur
- Modérateurs
- Contributeurs
- Membres
- Et les utilisateurs anonymes



Cas d'utilisation :

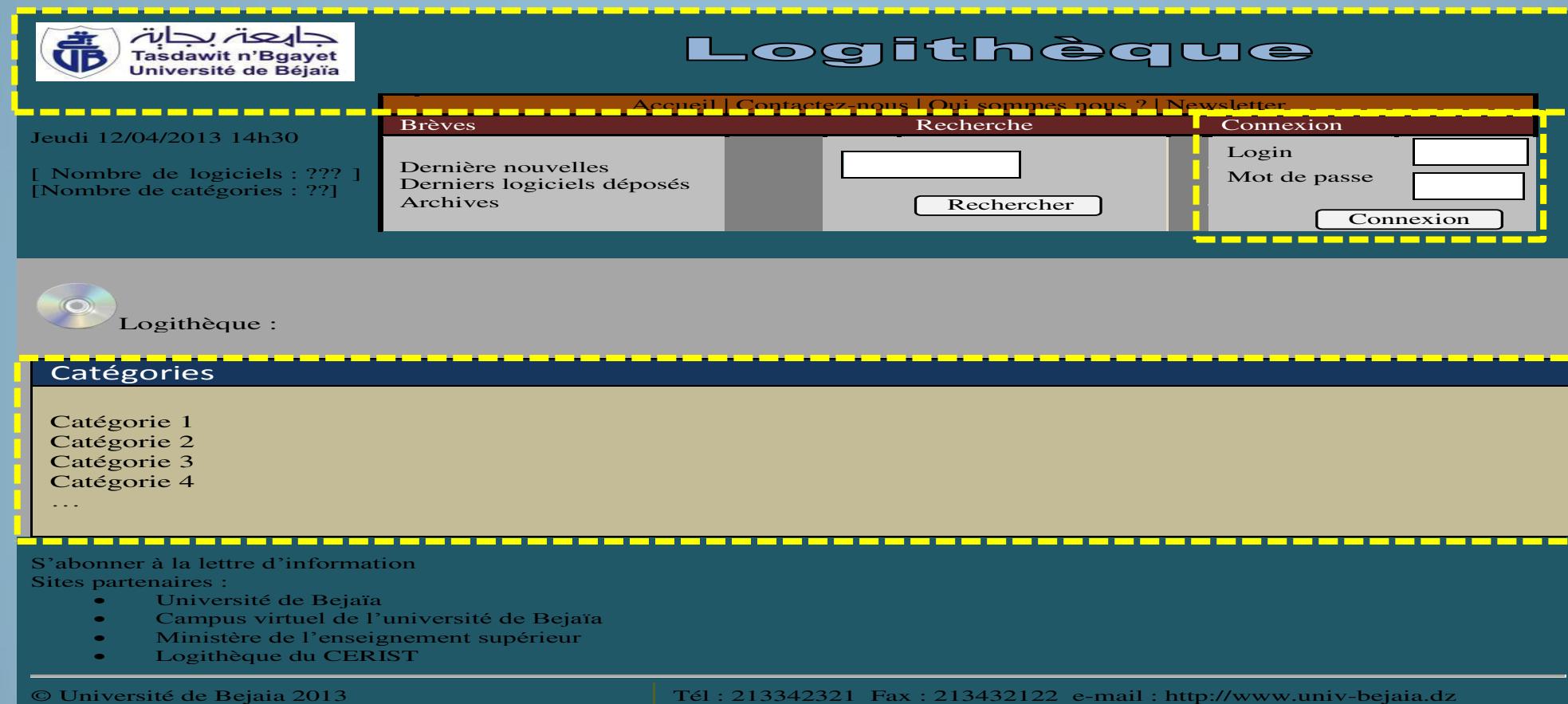
- Diagramme de cas d'utilisation général



Spécification des besoins

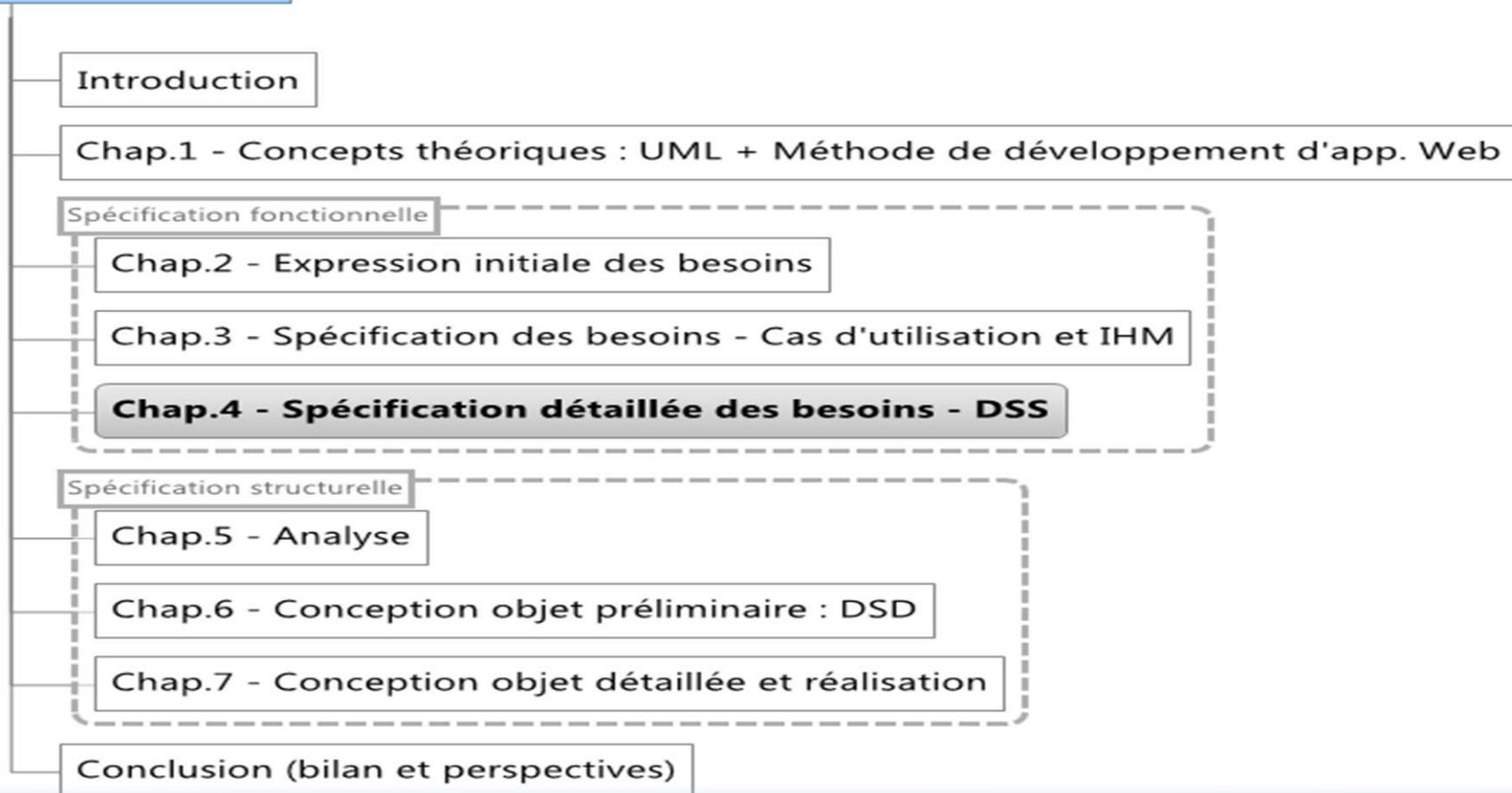
□ Maquettage IHM

➤ Modèle pour l'écran d'accueil de la logithèque



Chapitre.4 – Spécification détaillé des besoins - DSS

PFE - Logithèque



Spécification détaillée des besoins : DSS

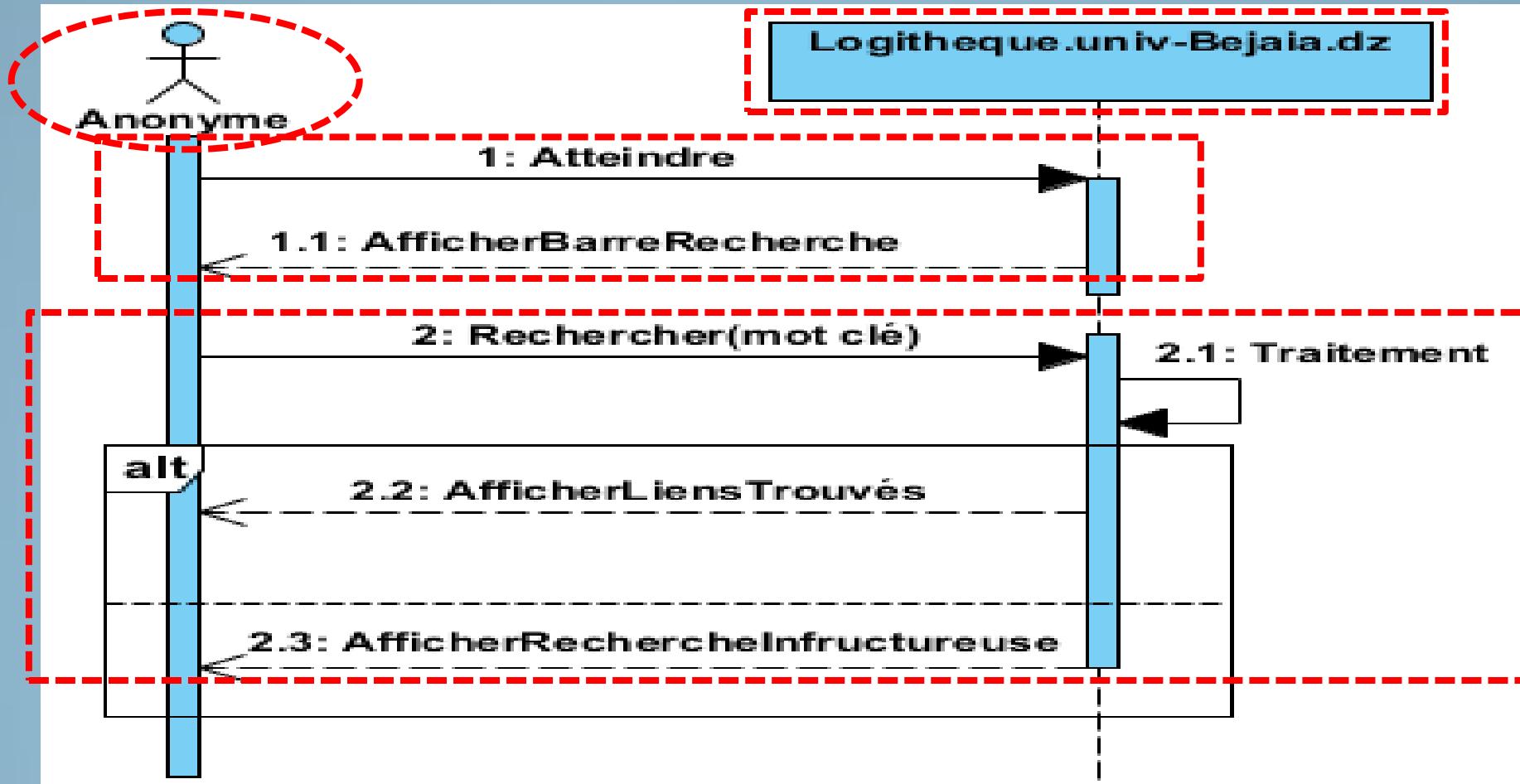


Diagramme de séquence système du cas d'utilisation : Rechercher

Spécification détaillée des besoins : DSS

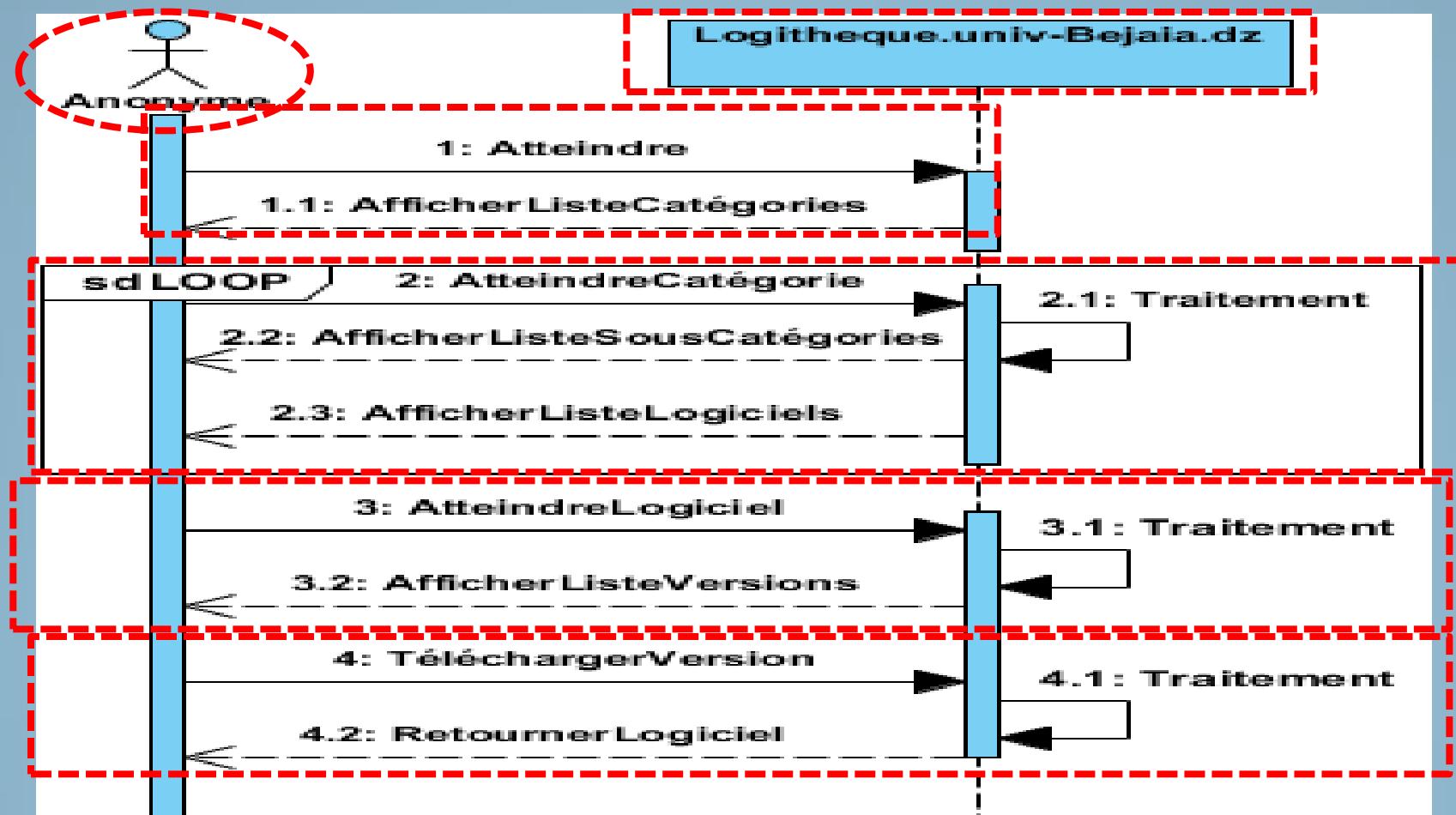


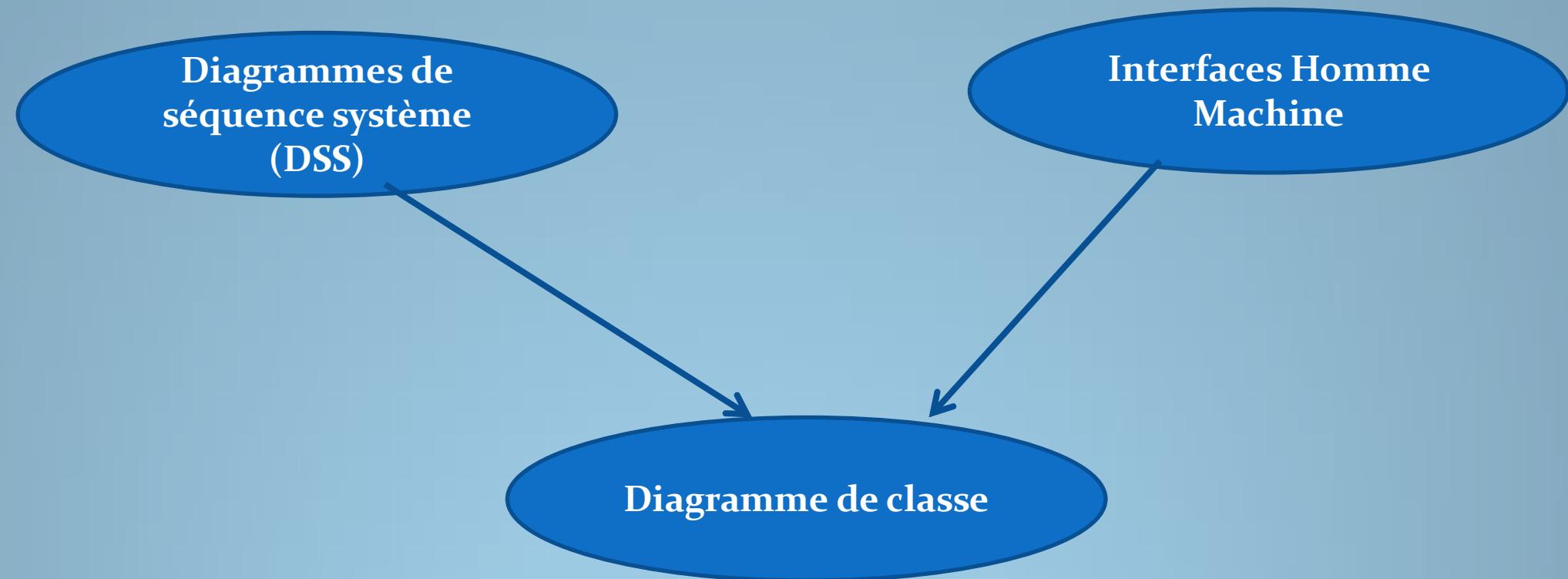
Diagramme de séquence système du cas d'utilisation : Télécharger Version Logiciel

Chapitre.5 – Analyse

PFE - Logithèque



Analyse



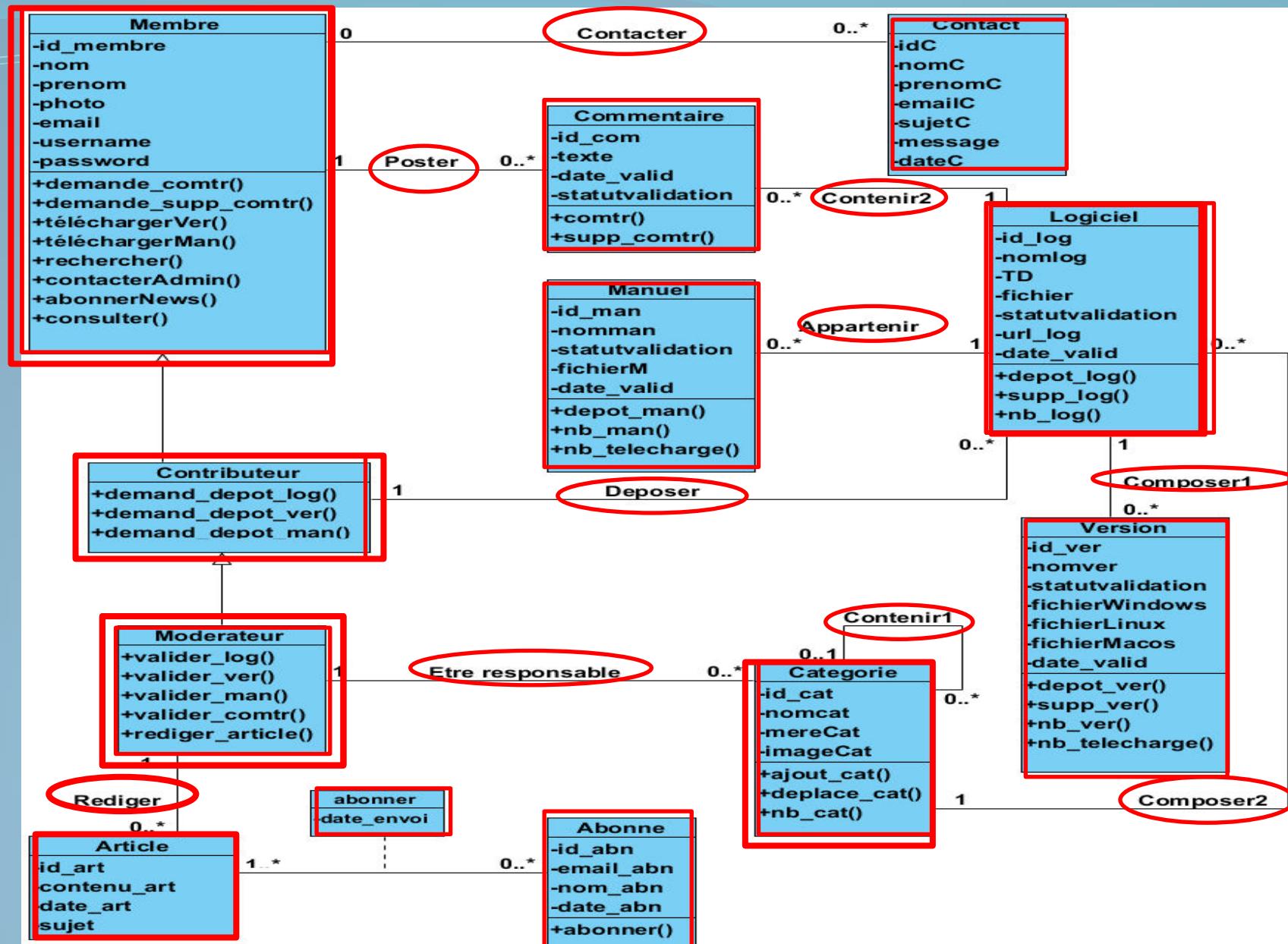


Diagramme de classe

Analyse : Modèle relationnel

Membre (id_membre, nom, prenom, photo, email, username, password).

Contributeur (id_membre#).

Moderateur (id_membre#).

Categorie (id_cat, nomcat, mereCat, imageCat, id_membre#).

Logiciel(id_log, nomlog, TD-fichier, statutvalidation, url_log, date_valid, id_cat#, id_membre#).

Version(id_ver, nomver, statutvalidation, fichierWindows, fichierLinux, fichierMacos, date_valide, id_log#).

Manuel (id_man, nomman, statutvalidation, fichierM, date_valide, id_log#).

Commentaire (id_com, texte, date_valid, statutvalidation, id_log#, id_membre#).

Article (id_art, contenu_art, date_art, sujet, id_membre#).

Contact (idC, nomC, prenomC, emailC, sujetC, message, dateC, id_membre#).

Abonne (id_abn, nom_abn, email_abn, date_abn).

abonner (id_abn#, id_art#, date_envoi).

Analyse : Diagrammes de classes participantes

pour chaque cas d'utilisation, nous identifions les classes de :

- Contrôle : permettant d'assurer les traitements de l'application et de jouer l'intermédiaire entre les classes de dialogue et les classes d'entité.
- Entité (modèles au sens du concept MVC) : permettant de jouer l'intermédiaire entre les classes de contrôle et la base de données.
- Dialogues : Ecrans.

Analyse : Diagrammes de classes participantes

- **Règles d'établissement des diagrammes de classes participantes**
- **Les entités vont seulement posséder des attributs.**
- **Les contrôles vont posséder des attributs et des opérations.**
- **Les dialogues vont posséder des attributs et des opérations**

Analyse : Diagrammes de classes participantes

□ Les règles à respecter entre les trois types de classes :

- Les dialogues ne peuvent être reliés qu'aux contrôles ou à d'autres dialogues, mais pas directement aux entités.
- Les entités ne peuvent être reliées qu'aux contrôles ou à d'autres entités.
- Les contrôles ont accès à tous les types de classes, y compris d'autres contrôles.
- Enfin, nous rajouterons les acteurs qui ne doivent être reliés qu'aux dialogues.

Analyse : Diagrammes de classes participantes

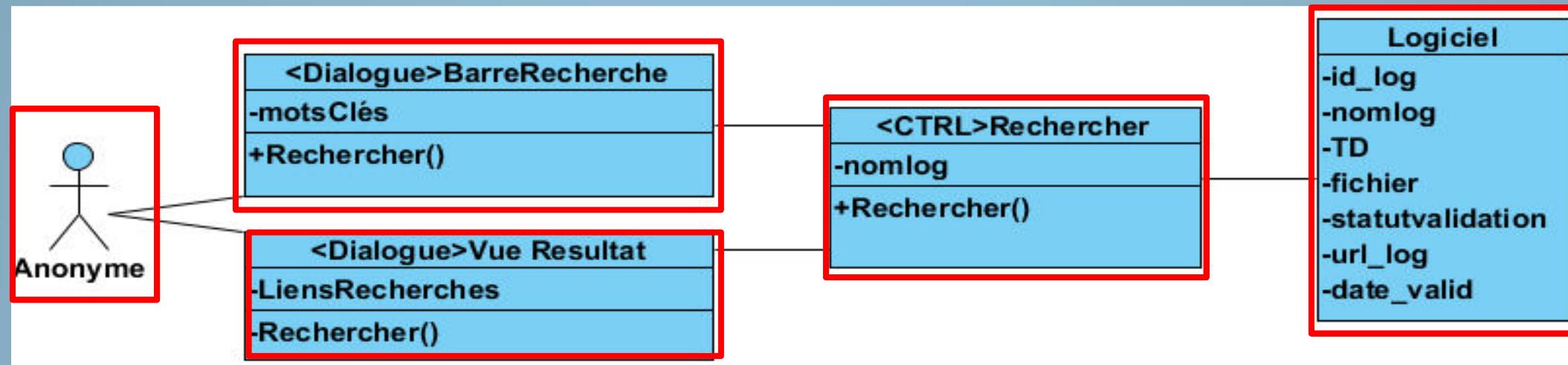


Diagramme de classe participant du cas d'utilisation : Recherche

Analyse : Diagrammes de classes participantes

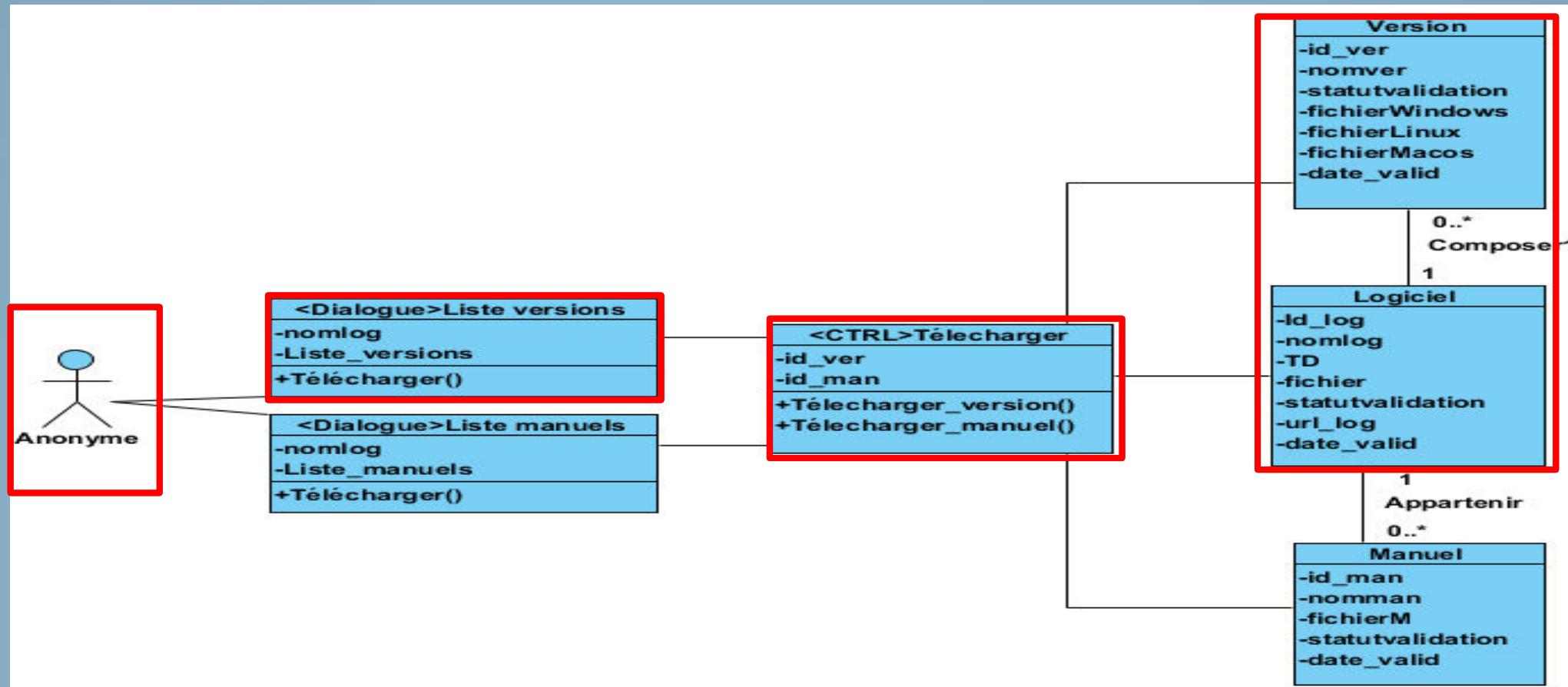
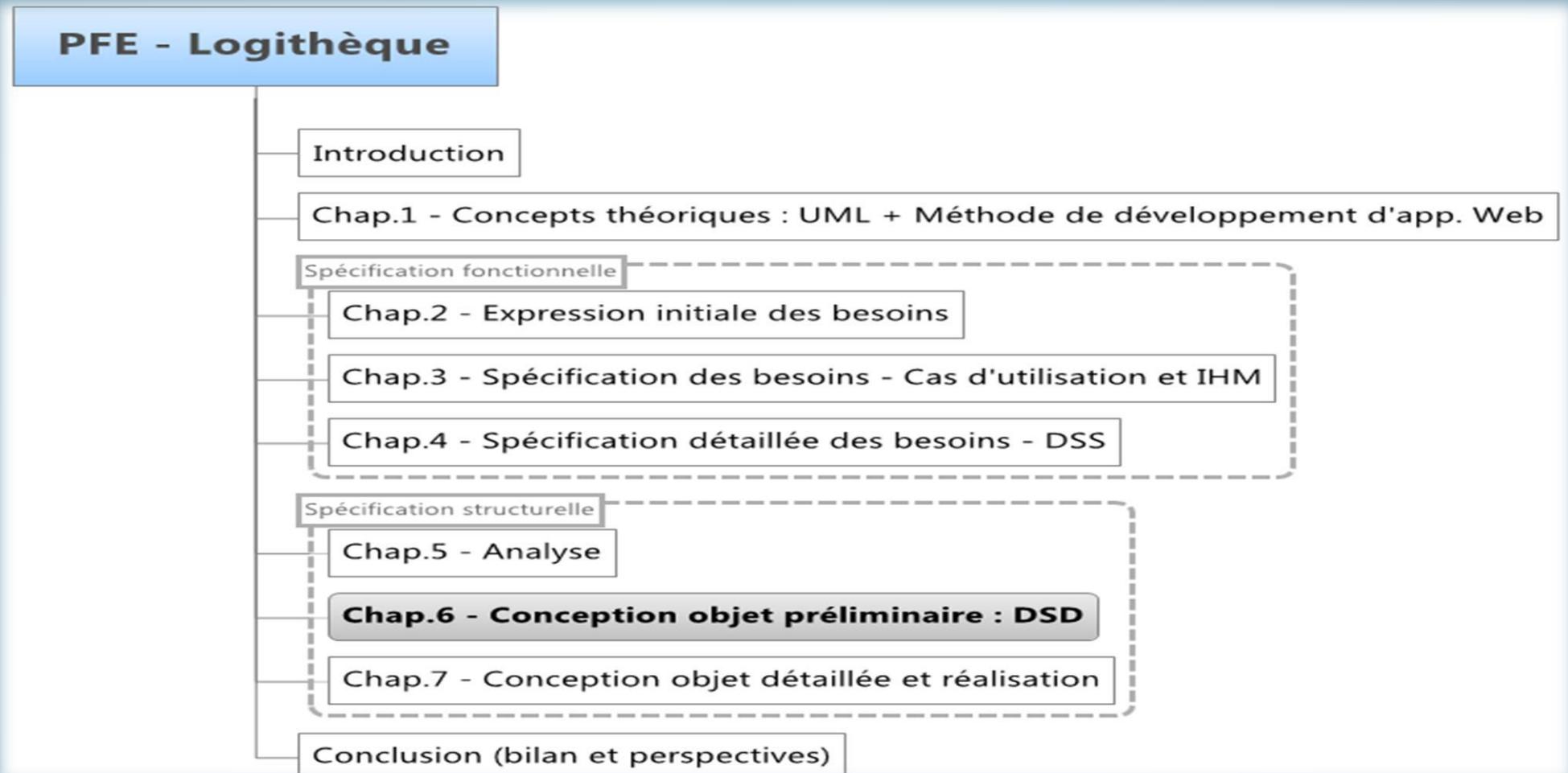
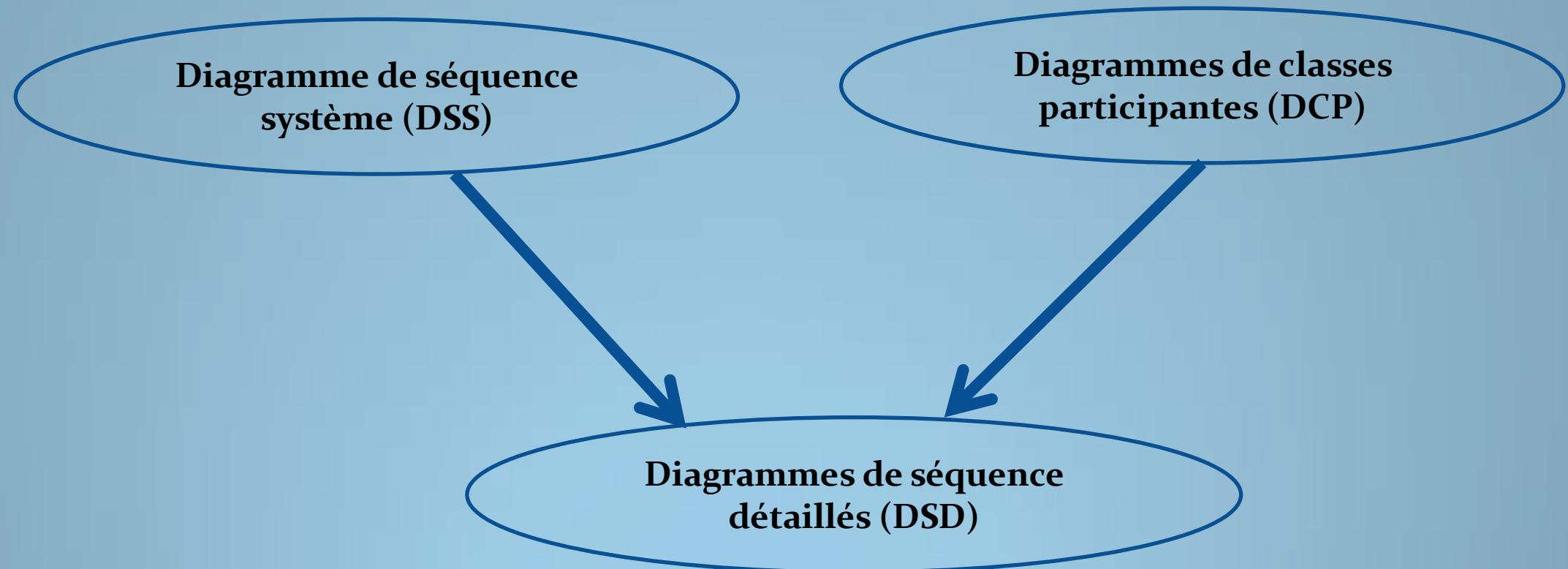


Diagramme de classes participantes du cas d'utilisation : Téléchargement d'une version / un manuel

Chapitre.6 – Conception objet préliminaire : DSD

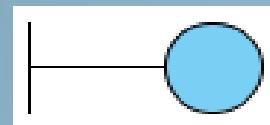


Chapitre 6 - Conception objet préliminaire : DSD

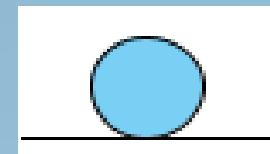


Chapitre 6 - Conception objet préliminaire : DSD

➤ Les stéréotypes utilisés



Vue



Modèle



Contrôleur

Chapitre 6 - Conception objet préliminaire : DSD

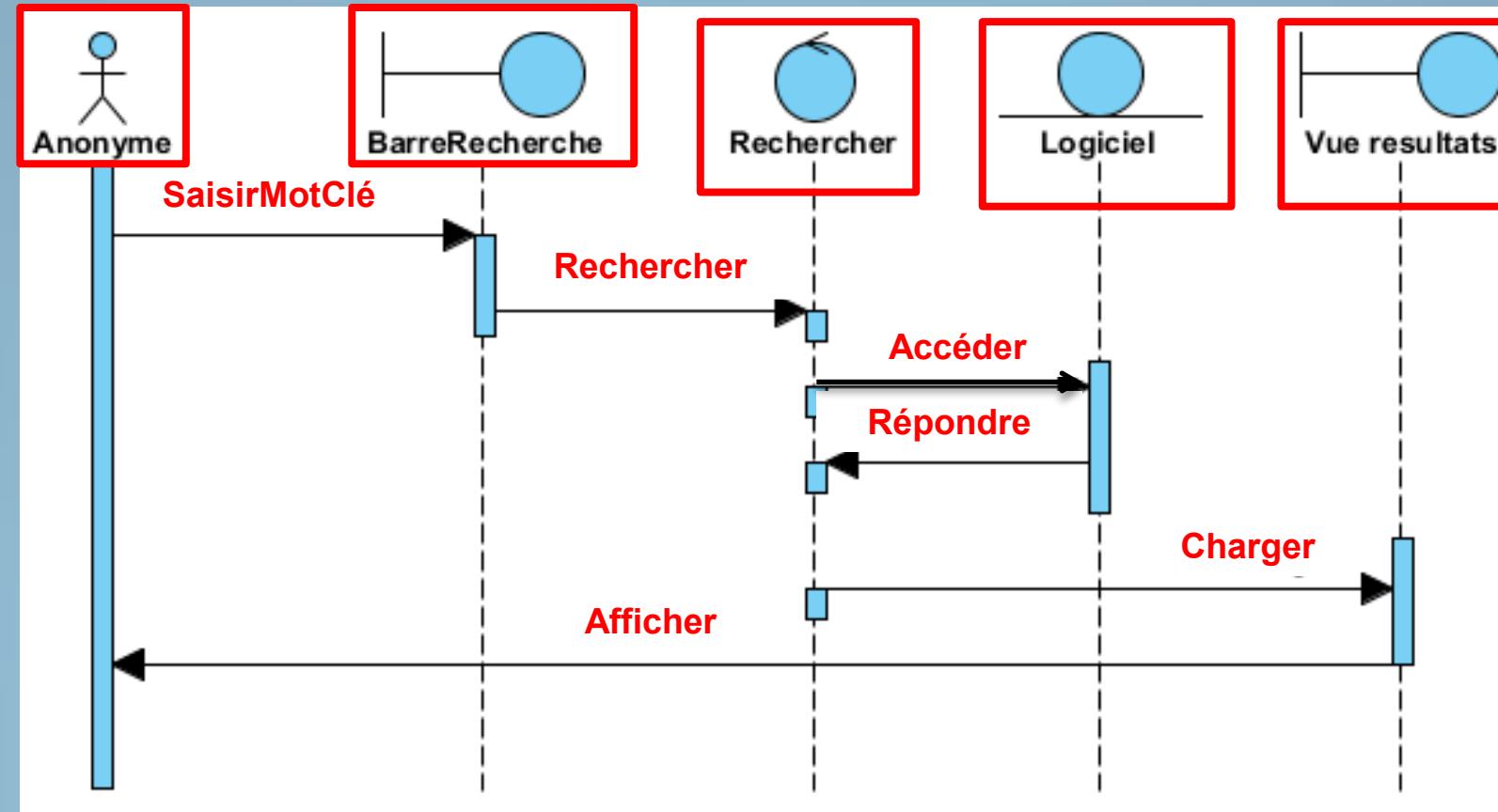


Diagramme de séquence détaillé du cas d'utilisation : Recherche

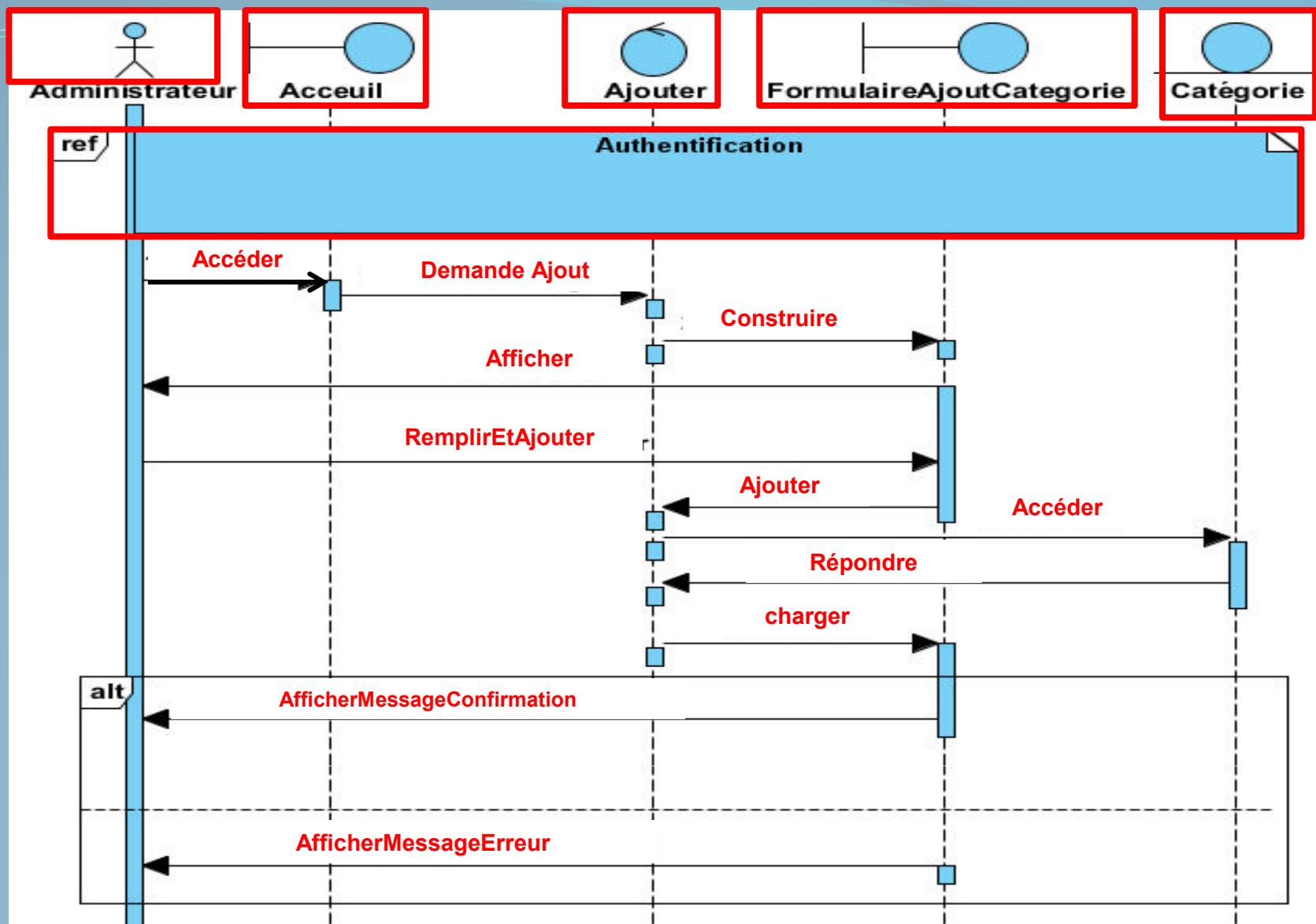
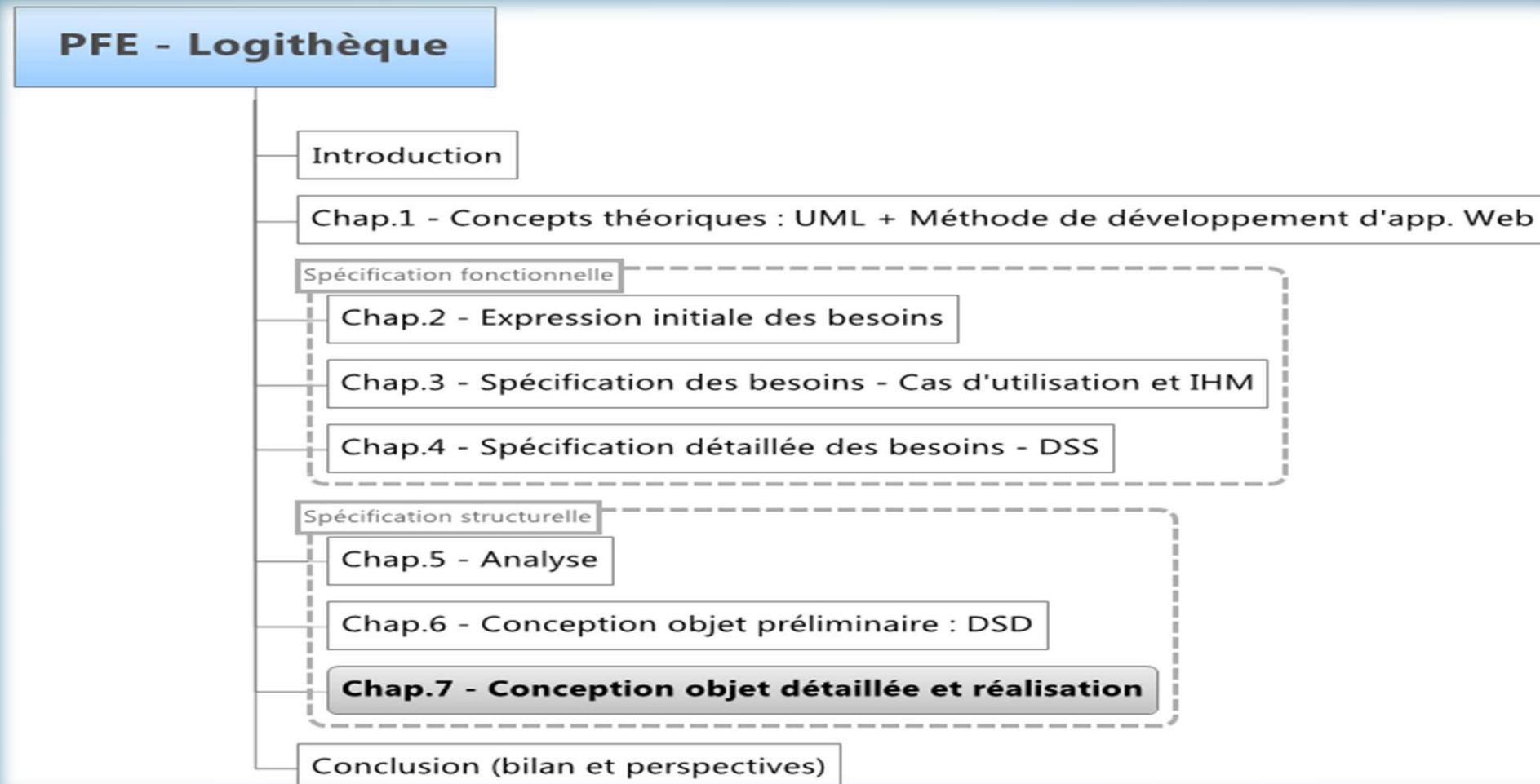
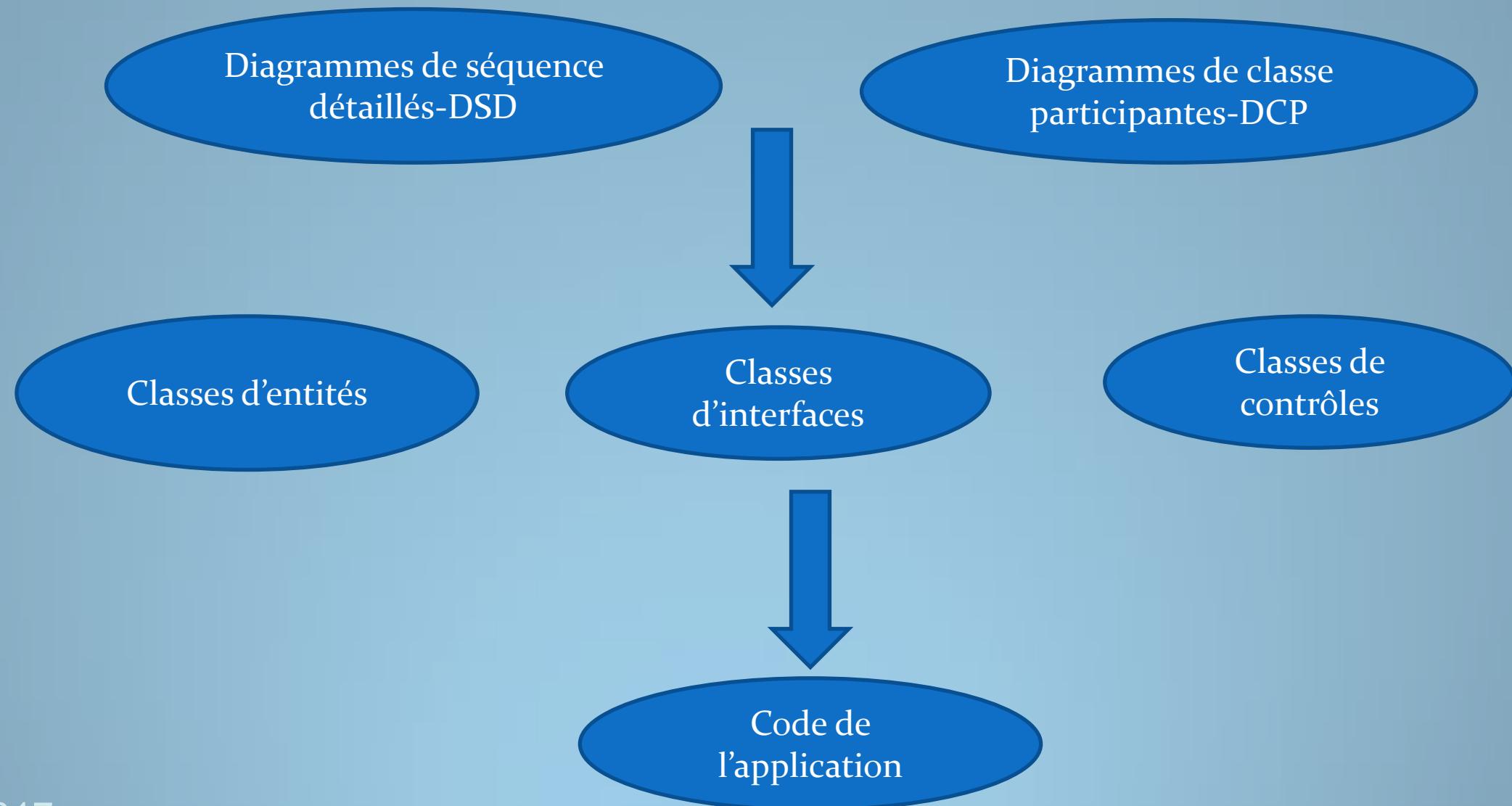


Diagramme de séquence détaillé du cas d'utilisation :Ajout catégorie

Chapitre.7 – Conception objet détaillée et réalisation



Résumé de la conception



Solution technologique

La logithèque est accessible depuis un navigateur Web et son code est hébergé sur un serveur, qui est doté :

- **d'un interpréteur de Script.**
- **d'applications serveur :**
 - **serveur Web (on a opté pour Apache).**
 - **serveur BDD (on a opté pour MySQL).**

Comment pouvons-nous réaliser le code ?

Nous avons choisi de programmer avec PHP **en** utilisant un Framework reposant sur une architecture MVC.

C'est quoi MVC?

MVC est une architecture qui sépare entre les données ,les traitements et la présentation, ce qui devise l'application en trois parties:

Modèle, Vue et Contrôleur

C'est quoi un Framework et pourquoi l'utiliser ?

Un Framework est un cadre de travail offrant un ensemble d'outils permettant d'automatiser certaines parties de développement d'application, il donne comme avantages :

- Un code homogène.
- Application plus facilement sécurisées.
- Robustesse .

Quel Framework PHP adoptant la structure MVC à choisir ?

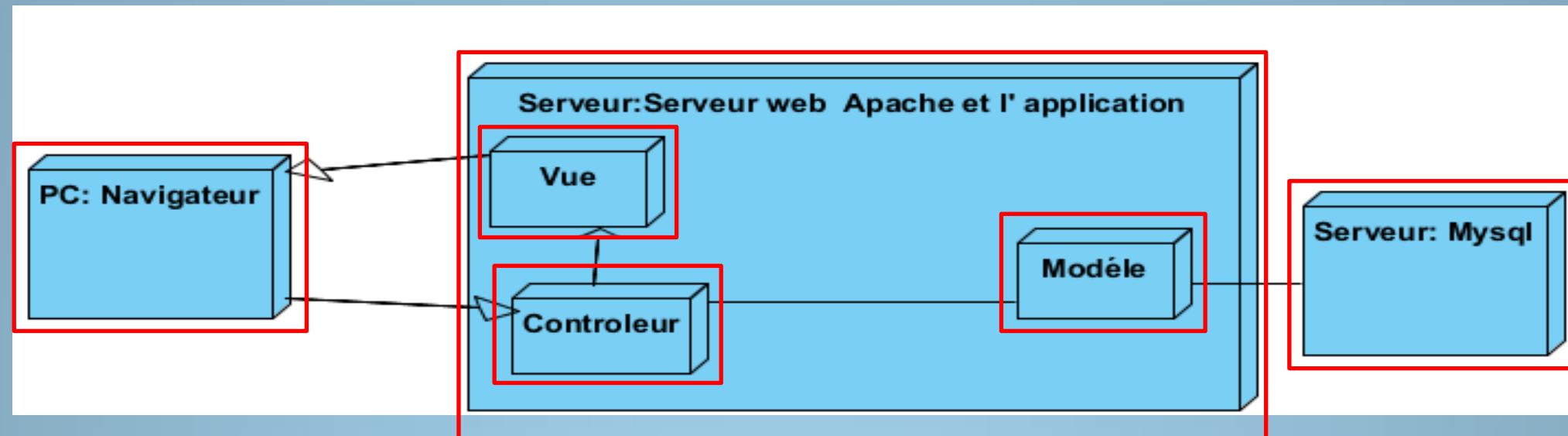
Nous avons choisi CodeIgniter

Pourquoi CodeIgniter ?

- ✿ Puissant et nécessite peu de ressources.
- ✿ En évolution et il a une importante collection de librairies .
- ✿ Principaux atouts : légèreté , robustesse, fiable, facile à installé et accepte de multiples environnements.

Diagramme de déploiement

Le diagramme de déploiement montre la disponibilité physique des matériels du système.



Implémentation de notre solution se base sur une architecture à 3 niveaux :

- Niveau présentatif: représentant le navigateur web.
- Niveau applicatif : pris en charge par le serveur Apache.
- Niveau qui permet de faire les liens entre contrôleurs et BDD.

Sécurité de la logithèque

Fonctionnalités de sécurité de CodeIgniter :

- ☛ Utilisation du fichier Index.html et limitation des caractères autorisés dans une URI.
- ☛ Utilisation d'un filtre pour chaque appel d'un contrôleur.
- ☛ Utilisation d'un filtre XSS.
- ☛ Utilisation de certains paramètres pour se prémunir contre les attaques CSRF.

Autre moyen pour sécuriser l'application :

- ◆ Utilisation du chiffrement SHA1 pour sécuriser les mots de passes.
- ◆ Utilisation des fonctions pour éviter les injections SQL.
- ◆ Utilisation d'expressions régulières JavaScript afin de sécuriser les formulaires.

Exemple du code : cas d'utilisation <<Télécharger une version>>

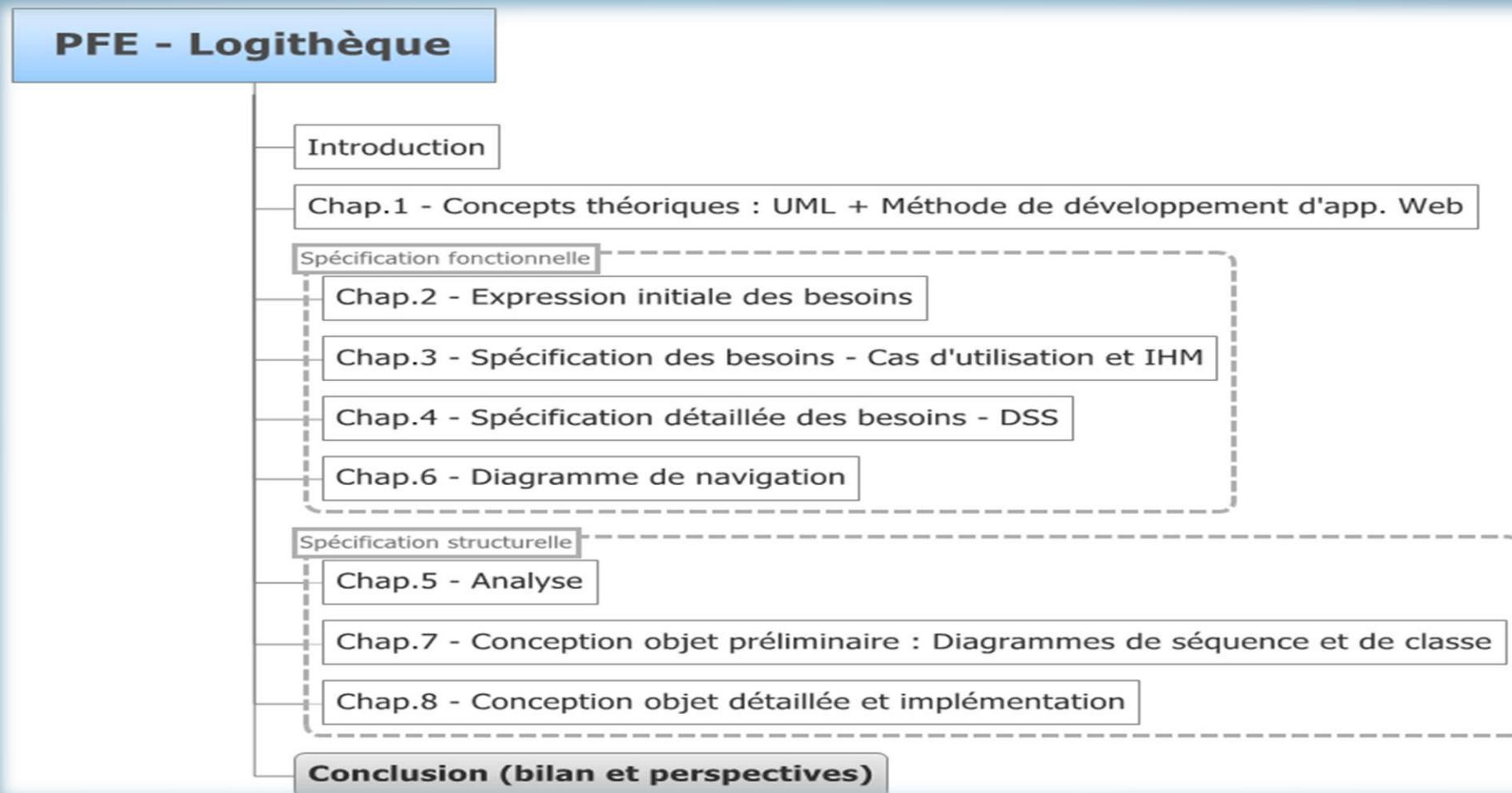
Vue

```
<?php $log= mysql_real_escape_string($_GET['id_log']); ?>
    <tr><td>
<?php if ($records>0) :?>
    <div id="thread"> Les versions (<?php echo $nb_versions;?>)</div>
    <?php foreach($records as $row):?>
        <tr>
            <td>input->get(mysql_real_escape_string('id_ver'));
            $this->load->model('versions_model');
            $this->versions_model->down($i);
        }
    }
}
```

Modèle

```
<?php
class Versions_model extends CI_Model
{
private $table = 'version';
function down($i)
{
    $this->db->protect_identifiers('version');
    $sql=$this->db->query("SELECT * FROM version WHERE id_ver = '". mysql_real_escape_string($i) . "'");
    if($sql->num_rows()>0)
    {
        $row=$sql->row();
        $this->load->helper('download');
        $name = $row->nomver ;
        redirect('http://localhost/cill/image_logitheque/' . $name);
    }
}
```

Conclusion (bilan et perspectives)



Déroulement d'un exemple



TOP5

- Abiword
- xamp
- Inventory Serveur
- Avira
- Java

Nouveaux Articles

- Exemple de formulaire
- Exigences fonctionnelles
- controles entites
- Manifeste Agile
- unified process

Nouveaux Logiciels

- WAMPSERVER
- Abiword
- Inventory Serveur
- Ubuntu
- Avira

[Accueil](#)[Newsletter](#)[Contact](#)[Nous](#)

Rechercher

search

[Voir tous les logiciels](#)

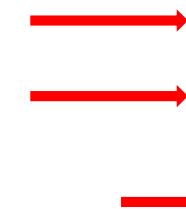
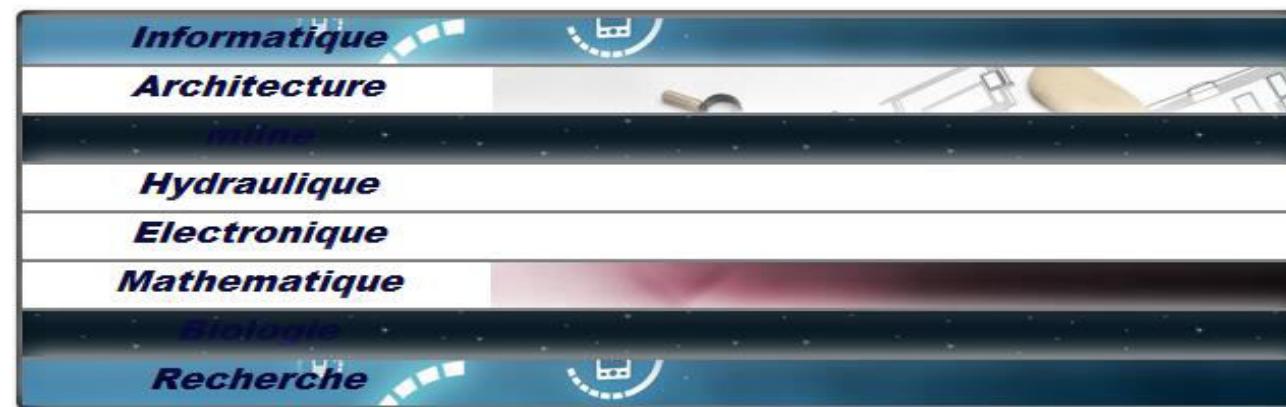
04:30:1

Espace Membre

identifiant

mot de passe

se connecter



Sites partenaires

- Universite Bejaia
- Rectorat Bejaia
- ServicesCommunautaires
- Statistique

- Vous etes le visiteur: 1434
- Nombre Categories: 1
- Nombre Logiciels: 1



TOP5

- Abiword
- xamp
- Inventory Serveur
- Avira
- Java

Nouveaux Articles

- Exemple de formulaire
- Exigences fonctionnelles
- controles entites
- Manifeste Agile
- unified process

Nouveaux Logiciels

- WAMPSERVER
- Abiword
- Inventory Serveur
- Ubuntu
- Avira

[Accueil](#)[Newsletter](#)[Contact](#)[Nous](#)

Rechercher

search

[Voir tous les logiciels](#)

05:23:30

Espace Membre



- Bouzidi El hadi
 - Role: administrateur
- Gestion d'utilisateurs
► Gestion des contacts
► AttenteValidation
► gestion des articles

Deconnexion

Sites partenaires

- Universite Bejaia
- Rectorat Bejaia
- ServicesCommuns

Statistique

- Vous etes le visiteur 1450 .
- Nombre Categories: 23
- Nombre Logiciels: 14



A large central image shows a man in a suit pointing at a digital screen. The screen displays several academic subjects: Informatique, Architecture, Hydraulique, Electronique, Mathematique, Biologie, and Recherche. The word 'Informatique' is prominently displayed at the top left of the screen. A red arrow points from the left margin towards this central image.

[Ajouter categorie](#)



TOP5

- Abiword
- xamp
- Inventory Serveur
- Avira
- Java

Nouveaux Articles

- Exemple de formulaire
- Exigences fonctionnelles
- controles entites
- Manifeste Agile
- unified process

Nouveaux Logiciels

- WAMP SERVER
- Abiword
- Inventory Serveur
- Ubuntu
- Avira

Accueil

Newsletter

Contact

Nous

Rechercher

search

Voir tous les logiciels

05:23:30

Espace Membre



- Bouzidi El hadi
- Role: administrateur
- Gestion d'utilisateurs
- Gestion des contacts
- AttenteValidation
- gestion des articles

Deconnexion

Sites partenaires

- Universite Bejaia
- Rectorat Bejaia
- ServicesCommuns

Statistique

- Vous etes le visiteur 1450 .
- Nombre Categories: 23
- Nombre Logiciels: 14

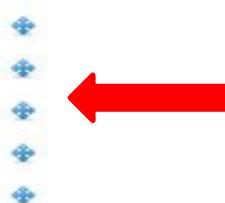
Les logiciels(2) +

- LibreOffice
- acrobat



Les sous categories(5)

- Bureautique
- Developpement
- Securite
- Systeme
- Reseaux





TOP5

- CodeIgniter
- Abiword
- PDFcreator
- Avira
- Java

[Accueil](#)[Contact](#)[Nous](#)[Newsletter](#)

nomlog

search

[Voir tous les logiciels](#)

Espace Membre



- Bouzidi El hadi
- Role:administrateur
- Gestion d'utilisateurs
- Gestion des contacts
- AttenteValidation

[Deconnexion](#)

Sites partenaires



Nouveaux Logiciels

- school
- samira
- Abiword
- Inventory Serveur
- Ubuntu

Nouveaux Articles

- Avira Free Antivirus
- Open Computers
- logiciels libres7
- Parc informatique
- Image et Texte

Les logiciels(3) +

- ▶ CodeIgniter
- ▶ Java
- ▶ Komposer



Les sous categories(0) +

Statistique

Nombre visiteurs:
Nombre Categories:23
Nombre Logiciels: 17

ma page d'accueil categorie

localhost/ci11/index.php/ouvrir/pageLogici.php?id_log=47

Google

Rechercher

search

Voir tous les logiciels

05:23:30

TOP5

- Abiword
- xamp
- Inventory Serveur
- Avira
- Java

Nouveaux Articles

- Exemple de formulaire
- Exigences fonctionnelles
- contrôles entités
- Manifeste Agile
- unified process

Nouveaux Logiciels

- WAMPSERVER
- Abiword
- Inventory Serveur
- Ubuntu
- Avira

CodeIgniter



CodeIgniter est un framework libre écrit en PHP. Il suit le motif de conception MVC et s'inspire du fonctionnement de Ruby on Rails. Les versions inférieures à 2.0.0 sont compatibles avec PHP 4 et 5, tandis que celles supérieures à 2.0.0 ne sont pas compatibles qu'avec PHP 5.1.6 ou plus.

Site de l'auteur

NB download: 2

Les versions(1)+

Codeigniter → Télécharger pour Windows

Les manuels(2)

codeigniter.pdf →
CodeIgniter_User_Guide.pdf →

Espace Membre

Bouzidi El hadi

Role: administrateur

Gestion d'utilisateurs

Gestion des contacts

AttenteValidation

gestion des articles

Déconnexion

Sites partenaires

- Université Bejaia
- Rectorat Bejaia
- ServicesCommuns

Statistique

- Vous êtes le visiteur 1450 .
- Nombre Categories: 23
- Nombre Logiciels: 14

Conclusion

La réalisation de la logithèque a été une véritable expérience pédagogique qui nous a permis de découvrir le monde du développement d'applications logicielles. Nous nous sommes confronté dès le début au choix d'une méthode de conception, qui soit cohérente. Cependant, la découverte de l'ouvrage de Pascal Roques nous a aidé énormément sur ce point. Ce processus permet d'aller des besoins jusqu'au code de l'application. Arrivé à l'implémentation, le choix était fait sur l'utilisation du Framework CodeIgniter qui répond au souci de cohérence avec la conception ou nous avons respecté le principe MVC.