

RAPPORT DE STAGE CDA

"La plus grande attention doit être portée à la compréhension du problème, faute de quoi l'algorithme n'a aucune chance d'être correct". Denis Lapoire

"C'est toujours l'impatience de gagner qui fait perdre", Louis XIV cité dans "L'immortel" de FOG.

J'écoute et j'oublie.
Je lis et je retiens.
Je fais et j'apprends.
(Proverbe chinois)

Version du document : 1.0.10
Date de création : 3 octobre 2018
Date de dernière mise à jour : 30 février 2019

SOMMAIRE

| | |
|--|----|
| Chapitre 1 - Présentation générale..... | 4 |
| 1.1 - Avant-propos..... | 5 |
| 1.2 - Abstract..... | 6 |
| 1.3 - Remerciements..... | 7 |
| 1.4 - La démarche générale..... | 8 |
| 1.5 - Architecture basique d'une application multi-couches..... | 9 |
| Chapitre 2 - Cahier des charges..... | 10 |
| 2.1 - Expression des besoins..... | 11 |
| 2.2 - Cahier des charges..... | 12 |
| Chapitre 3 - Le planning..... | 13 |
| 3.1 - Diagramme de GANTT..... | 14 |
| Chapitre 4 - Analyse..... | 15 |
| 4.1 - Le Diagramme de Cas d'Utilisation..... | 16 |
| 4.1.1 - Représentation graphique..... | 16 |
| 4.1.2 - Fiche de description textuelle d'un cas d'utilisation..... | 17 |
| 4.2 - Les maquettes..... | 18 |
| 4.2.1 - Maquette 2 : Authentification..... | 18 |
| 4.2.2 - Modification d'un produit..... | 19 |
| 4.3 - Le Diagramme de Navigation..... | 20 |
| 4.4 - Les Diagrammes de Séquence Système..... | 21 |
| 4.4.1 - Définition..... | 21 |
| 4.5 - Les Diagrammes d'Activité..... | 22 |
| Chapitre 5 - Conception de la Base de Données..... | 23 |
| 5.1 - La démarche utilisée..... | 24 |
| 5.2 - Le Diagramme de Classes (UML) ou le MCD (Merise)..... | 25 |
| 5.3 - Le MLD (Merise ou pas)..... | 26 |
| 5.4 - Le MPD (Schéma de la BD)..... | 27 |
| 5.5 - SQL : le LDD..... | 28 |
| 5.6 - Les procédures stockées..... | 30 |
| Chapitre 6 - Conception de l'application..... | 32 |
| 6.1 - Les Diagrammes de Séquence Détaillés..... | 33 |
| 6.2 - Les Diagrammes d'Activité..... | 36 |
| 6.3 - Les Diagrammes de Classes Participantes..... | 37 |
| Chapitre 7 - Développement..... | 38 |
| 7.1 - Technologies utilisées..... | 39 |
| 7.2 - Commencez par les interfaces..... | 40 |
| 7.2.1 - Copies d'écrans..... | 41 |
| 7.2.2 - Codes statiques des écrans..... | 42 |
| 7.2.3 - Codes dynamiques des écrans..... | 43 |
| 7.3 - Continuez avec la partie Entities/Models et DAO..... | 44 |
| 7.4 - Terminez avec la « glue » : les controls..... | 45 |

| | |
|---|----|
| Chapitre 8 - Déploiement..... | 46 |
| 8.1 - Le Diagramme de Déploiement..... | 47 |
| 8.2 - Le déploiement..... | 48 |
| Chapitre 9 - La gestion de projet..... | 49 |
| 9.1.1 - UP..... | 50 |
| 9.1.2 - XP..... | 51 |
| 9.1.3 - SCRUM..... | 52 |
| 9.1.4 - GIT..... | 53 |
| 9.1.5 - MAVEN..... | 54 |
| Chapitre 10 - La sécurité..... | 55 |
| 10.1 - Injection SQL..... | 56 |
| 10.2 - Injection XSS..... | 57 |
| Chapitre 11 - Formation des utilisateurs..... | 58 |
| Chapitre 12 - Conclusion..... | 59 |
| Chapitre 13 - Annexes..... | 60 |
| 13.1 - Correspondances Projet/Reac..... | 61 |
| 13.2 - Gestion de projet..... | 63 |
| 13.2.1 - git..... | 63 |
| 13.2.2 - maven..... | 63 |
| 13.3 - Outils utilisés ... pour quels objectifs ?..... | 64 |
| 13.4 - Bibliographie et Webographie..... | 65 |
| 13.4.1 - Bibliographie..... | 65 |
| 13.4.2 - Webographie..... | 66 |
| 13.4.2.1 - git..... | 66 |
| 13.4.2.2 - maven..... | 66 |
| 13.4.2.3 - HTML..... | 66 |
| 13.4.2.4 - Java..... | 66 |
| 13.4.2.5 - PHP..... | 66 |
| 13.4.2.6 - SQL..... | 66 |
| 13.4.2.7 - MySQL..... | 66 |
| 13.4.2.8 - JavaScript..... | 67 |
| 13.4.2.9 - jQuery..... | 67 |
| 13.5 - Glossaire/Lexique ou/et Liste de mots-clés et sigles..... | 68 |
| 13.6 - Autres codes..... | 69 |
| 13.6.1.1 - Code pour l'insertion de données de test dans la BD..... | 69 |
| 13.6.1.2 - POJO complet..... | 70 |
| 13.6.1.3 - DAO complet..... | 71 |
| 13.6.1.4 - Autres..... | 72 |
| Chapitre 14 - Tables et Index..... | 73 |
| 14.1 - Table des illustrations..... | 74 |
| 14.2 - Index..... | 75 |

CHAPITRE 1 - PRÉSENTATION GÉNÉRALE

1.1 - AVANT-PROPOS

1.2 - ABSTRACT

1.3 - REMERCIEMENTS

1.4 - LA DÉMARCHE GÉNÉRALE

Cette démarche a été adoptée, enrichie par le cours et adaptée par rapport à l'équipe dans laquelle nous avons travaillé, notre vision et notre projet.

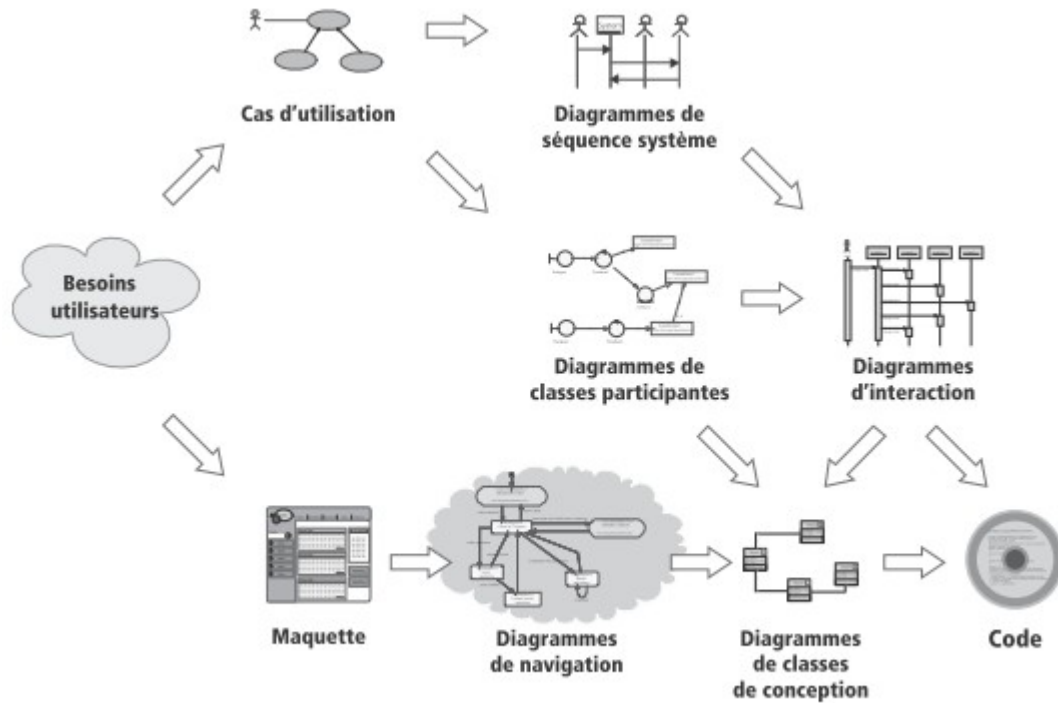
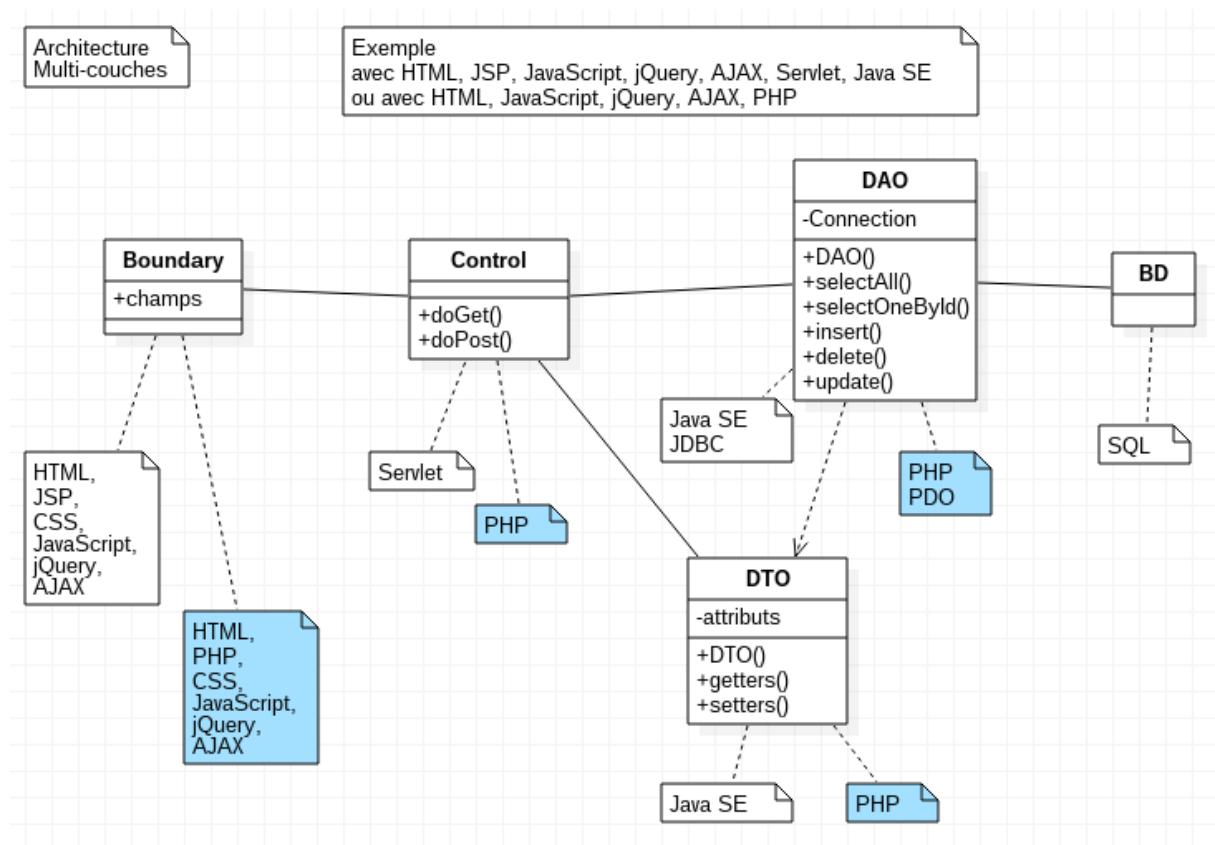


Figure 1-20 Schéma complet du processus de modélisation d'une application web

1.5 - ARCHITECTURE BASIQUE D'UNE APPLICATION MULTI-COUCHES



CHAPITRE 2 - CAHIER DES CHARGES

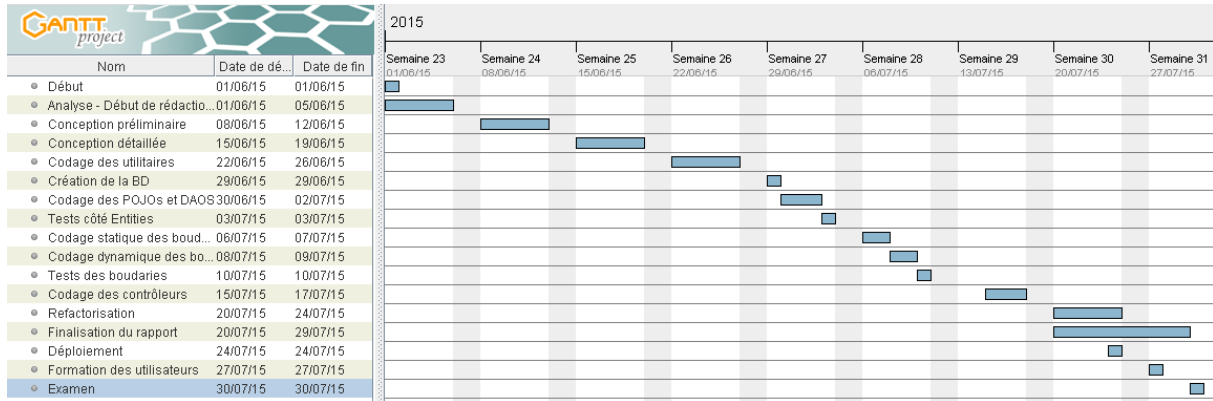
2.1 - EXPRESSION DES BESOINS

2.2 - CAHIER DES CHARGES

CHAPITRE 3 - LE PLANNING

3.1 - DIAGRAMME DE GANTT

Réalisé avec GANTT Project.

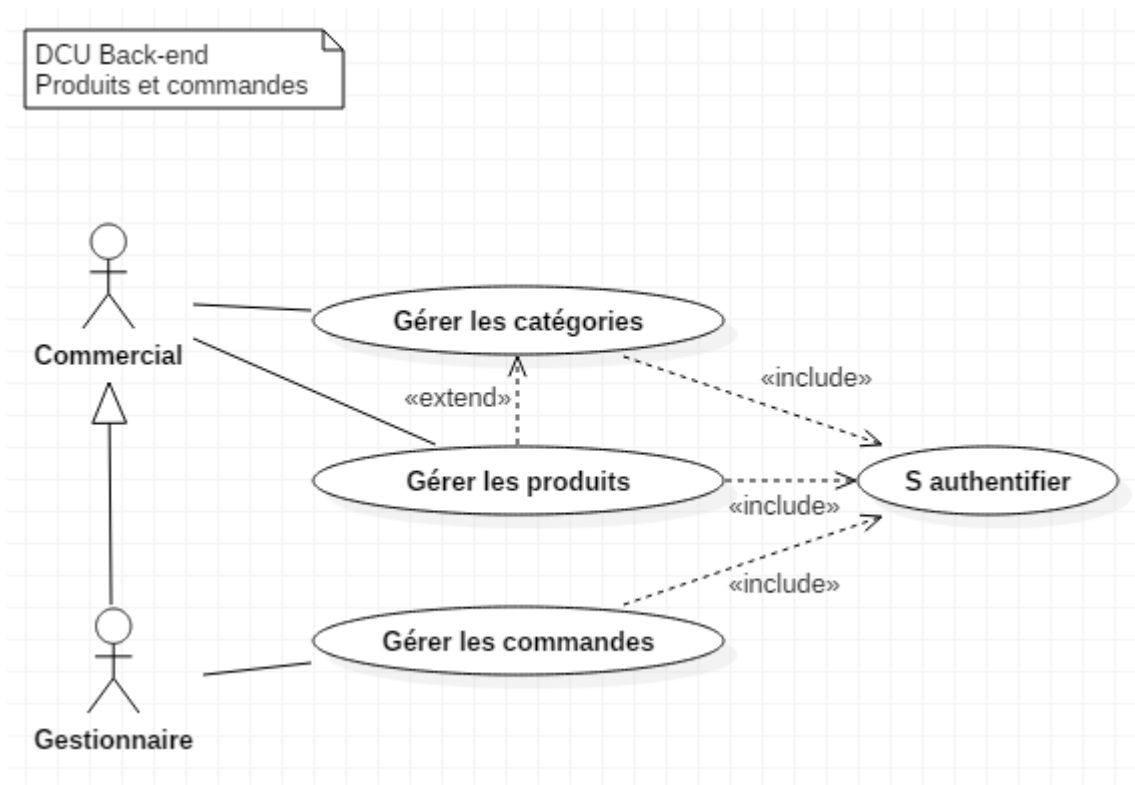


CHAPITRE 4 - ANALYSE

4.1 - LE DIAGRAMME DE CAS D'UTILISATION

4.1.1 - Représentation graphique

Un diagramme de cas d'utilisation (DCU) est un ensemble de cas d'utilisation (CU). Il formalise graphiquement les besoins des utilisateurs.



Réalisé avec StarUML

Mots clés :

Diagramme de Cas d'Utilisation,
 Cas d'utilisation,
 Acteur,
 Association simple,
 Généralisation,
 Extension,
 Inclusion,
 Association orientée.

4.1.2 - Fiche de description textuelle d'un cas d'utilisation

| Étapes | Description |
|--|---|
| Identification du CU | Titre : Résumé : Acteur : Date de création : Date de dernière modification : Version : Auteur : |
| Pré-conditions | |
| Scenario nominal | |
| Scenarii d'erreurs du cas nominal | |
| Scenarii alternatifs | |
| Scenarii d'erreurs des cas alternatifs | |
| Post-conditions | |
| | |
| Exigences non fonctionnelles | |
| Besoins d'IHM | |

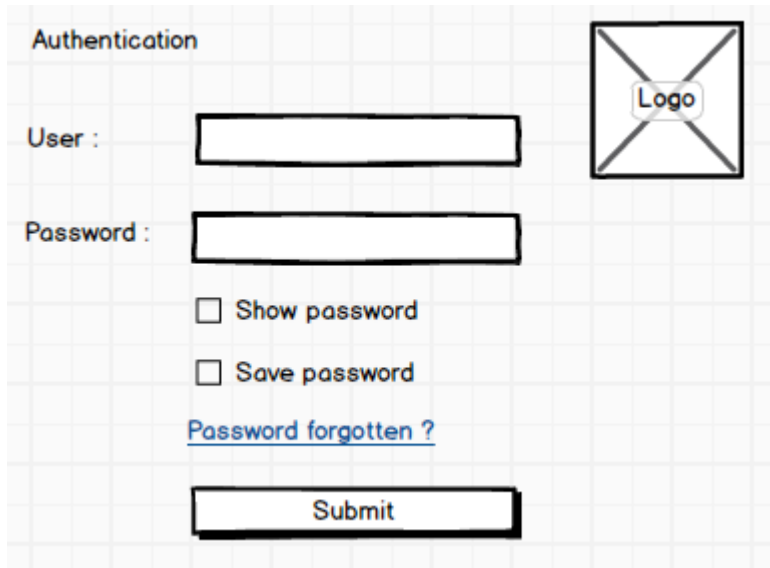
Les scenarii sont détaillés avec des DSS.

Le CU est éventuellement représenté avec un DAC.

4.2 - LES MAQUETTES

Avec un outil de maquettage (Pencil Evolus, Balsamiq, OpenOffice Writer, ...) qui potentiellement produit du HTML (surtout si vous produisez un site web). No comment !

4.2.1 - Maquette 2 : Authentification



The wireframe shows an authentication form on a light gray grid background. At the top left is the title 'Authentication'. To the right is a square placeholder for a logo, containing the word 'Logo' and a large 'X'. Below the title are two input fields: 'User :' followed by a text box, and 'Password :' followed by a text box. Under the password field are two checkboxes: 'Show password' and 'Save password'. Below these is a blue underlined link 'Password forgotten ?'. At the bottom is a 'Submit' button.

Réalisée avec « Balsamiq Mockups ».

4.2.2 - Modification d'un produit

The mockup shows a form for modifying a product. It starts with the title 'Modification d'un produit'. Below it is the instruction 'Sélectionnez un produit'. There is a dropdown menu labeled 'Produits' and a button labeled 'valider la sélection'. Below these are four input fields: 'ID', 'Désignation', 'Prix', and 'Stock'. At the bottom is a button labeled 'Validez les modifications'.

Modification d'un produit

Sélectionnez un produit

Produits ▼ valider la sélection

ID

Désignation

Prix

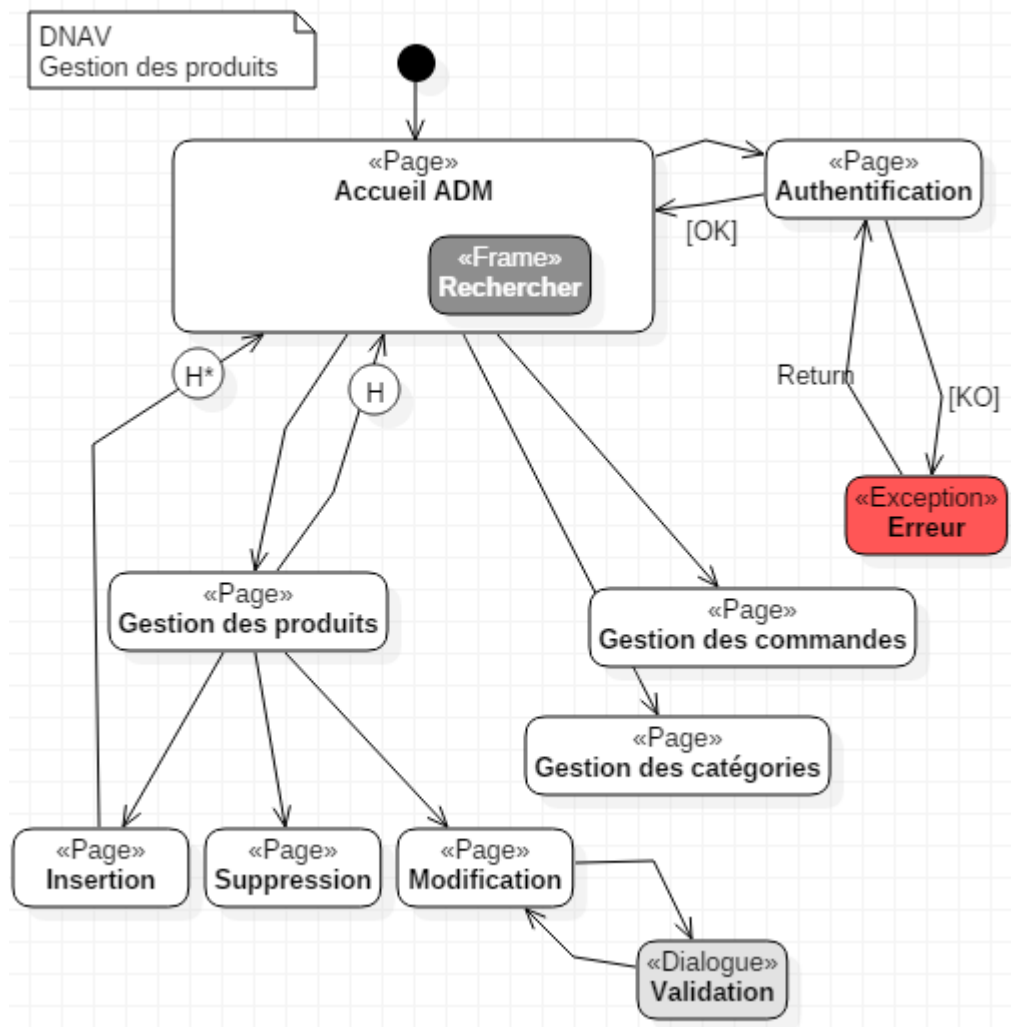
Stock

Validez les modifications

Réalisée avec « Balsamiq Mockups ».

4.3 - LE DIAGRAMME DE NAVIGATION

Le DNAV ...



Mots clés :

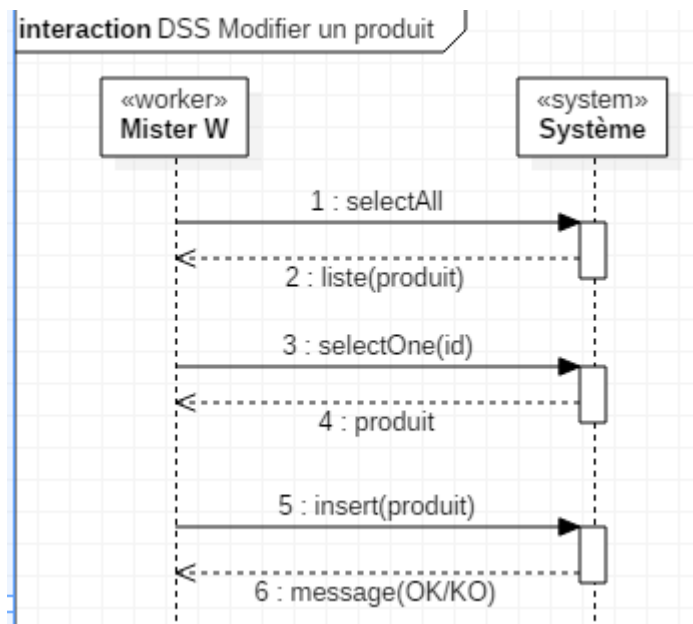
A vous ...

4.4 - LES DIAGRAMMES DE SÉQUENCE SYSTÈME

4.4.1 - Définition

Un Diagramme de Séquence Système (DSS) est une représentation graphique d'un CU ou même d'un scénario d'un CU.

Il est préférable de représenter chaque scénario (nominal, alternatif(s), erreur(s)) par autant de DSS.



Réalisé avec StarUML.

Mots clés :

A vous ...

4.5 - LES DIAGRAMMES D'ACTIVITÉ

Si un DSS ne représente qu'un scénario, il est possible de représenter un CU par un DAC.

Mots clés :

A vous ...

CHAPITRE 5 - CONCEPTION DE LA BASE DE DONNÉES

5.1 - LA DÉMARCHE UTILISÉE

Rappel :

Pour les CDI cette section compte pour 1/3 du diplôme !

Soit la démarche par l'analyse du discours (Méthode Chen) : un nom correspond à une classe (ou un Individu ou entité dans le monde Merisien) ou à un attribut et un verbe à une association.

Il faut passer ensuite le modèle à la « moulinette » des formes normales.

Soit la méthode des dépendances fonctionnelles. Et certains « compléments » sont obtenus avec la méthode précédente.

Dans tous les cas insistez sur les Formes Normales.

5.2 - LE DIAGRAMME DE CLASSES (UML) OU LE MCD (MERISE)

Note : MCD : Modèle Conceptuel des Données.

Comment le MCD ou le DCL est-il obtenu ?

Grâce aux règles de gestion ?

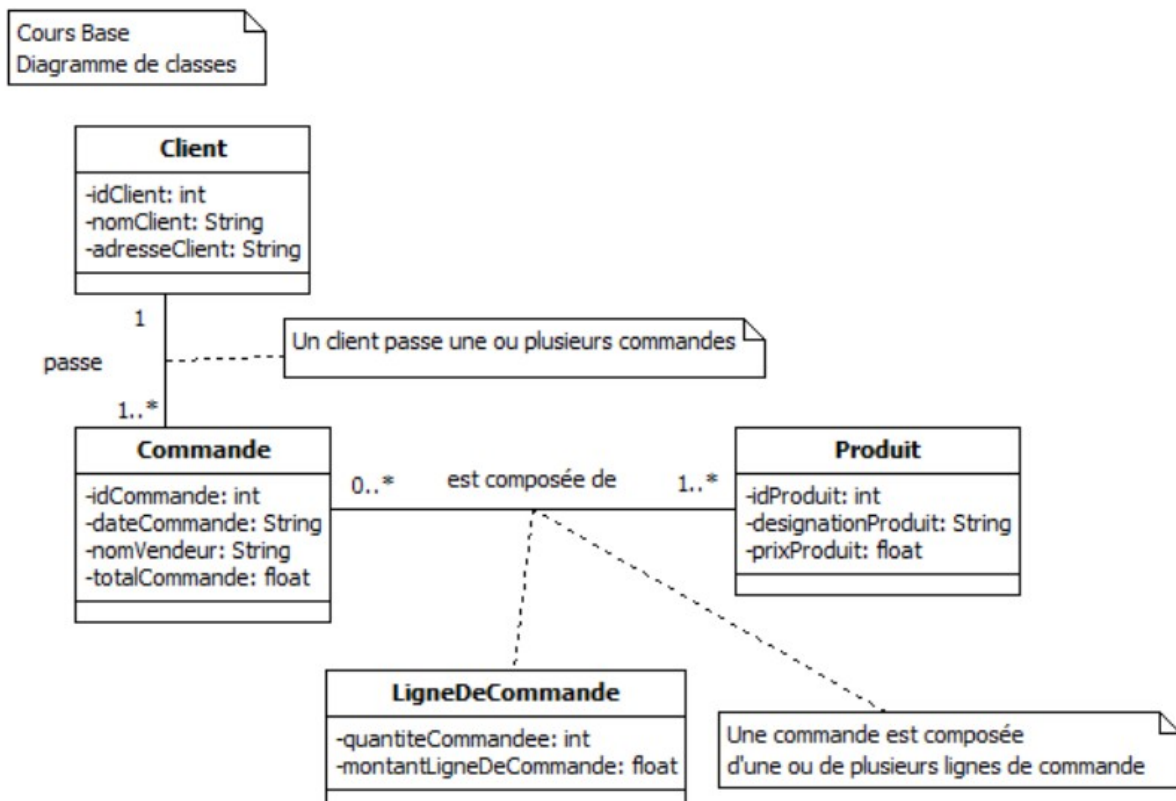
Par l'analyse des DF ?

Par les deux ?

Les dépendances fonctionnelles sont utilisées dans les 2 cas majeurs ?

Dans le 1^{er} cas en bout de chaîne dans le 2^{ème} cas en début de chaîne.

Dans le 1^{er} cas on utilisera la normalisation jusqu'à la 3^{ème} forme normale. A quoi servent-elles ?



Mots clés :

A vous ...

5.3 - LE MLD (MERISE OU PAS)

MLD : Modèle Logique des Données.

Comment est-il obtenu ?

Quelles sont les règles de transformation (les algorithmes) qui permettent de passer d'un DCL ou d'un MCD à un MLD ?

Pourquoi ce niveau ? Que signifie niveau logique ? S'agit-il d'une question de famille de produits ?

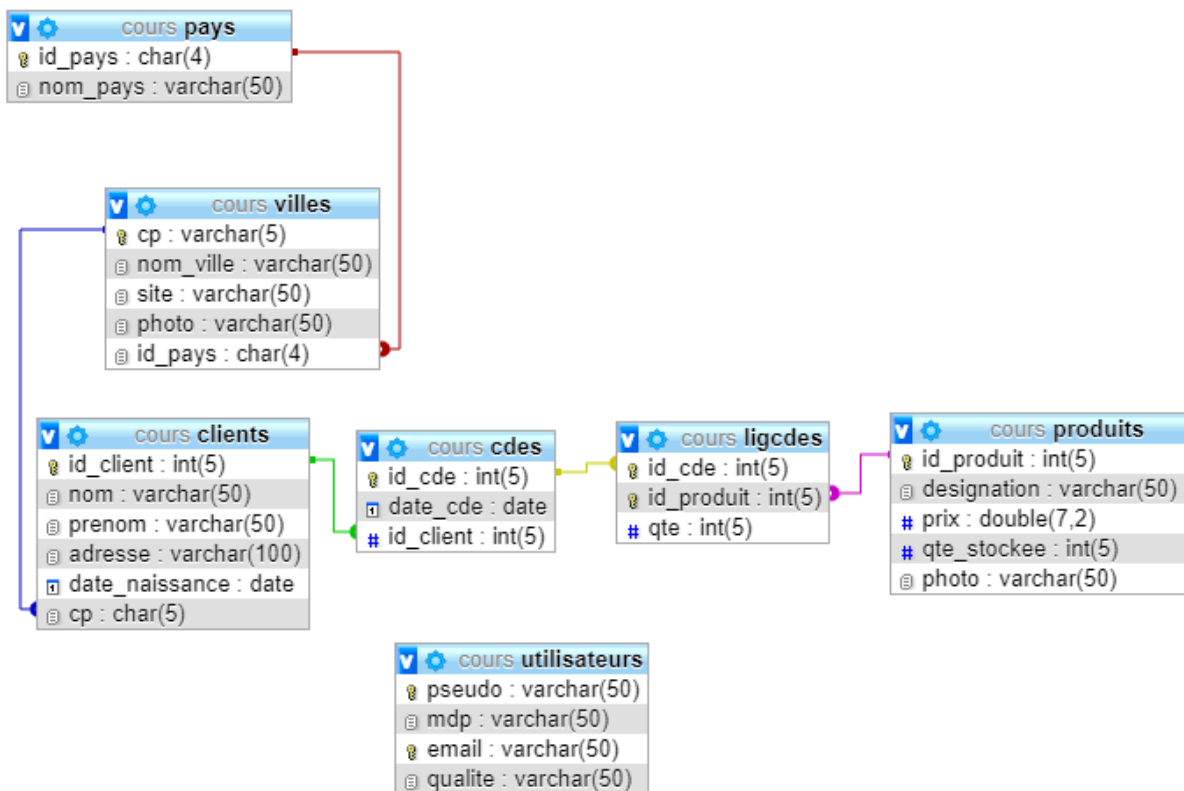
Mots clés :

A vous ...

5.4 - LE MPD (SCHÉMA DE LA BD)

MLD : Modèle physique des Données.

Pourquoi ce niveau ? Que signifie niveau physique ? S'agit-il d'une question de produit de la famille ?



Mots clés :

A vous ...

5.5 - SQL : LE LDD

Le code complet de création de la BD est dans les annexes.

Quelques exemples significatifs :

CREATE USER ...

CREATE DATABASE ...

```
| CREATE DATABASE IF NOT EXISTS nom_de_la_bd  
| DEFAULT CHARACTER SET utf8  
| COLLATE utf8_general_ci;
```

```
| USE nom_de_la_bd;
```

CREATE TABLE ...

CREATE UNIQUE INDEX ...

CREATE INDEX ...

```
| CREATE TABLE produits ...
```

```
CREATE TABLE IF NOT EXISTS villes (  
    cp varchar(5) NOT NULL,  
    nom_ville varchar(50) NOT NULL,  
    site varchar(50) default NULL,  
    photo varchar(50) default NULL,  
    id_pays char(4) NOT NULL,  
    PRIMARY KEY (cp),  
    KEY Index_id_pays (id_pays)  
) ENGINE=InnoDB;
```

```
ALTER TABLE villes  
    ADD CONSTRAINT FK_villes_pays FOREIGN KEY (id_pays) REFERENCES  
    pays (id_pays)  
    ON DELETE CASCADE ON UPDATE CASCADE;
```

CREATE PROCEDURE ...
CREATE TRIGGER ...

Mettez en relief les instructions et la puissance de SQL !

Avec les types, les PK, les FK, les index uniques, ..., éventuellement les procédures stockées, éventuellement les triggers.

5.6 - LES PROCÉDURES STOCKÉES

Les procédures stockées du CRUD+ (selectAll, selectOne, insert, update, delete).
3 procédures stockées implémentant des jointures et des fonctions agrégats.

CRUD+

```

DELIMITER $$

USE xxx $$

DROP PROCEDURE IF EXISTS produits_select_all $$
DROP PROCEDURE IF EXISTS produits_select_one $$
DROP PROCEDURE IF EXISTS produits_insert $$
DROP PROCEDURE IF EXISTS produits_update $$
DROP PROCEDURE IF EXISTS produits_delete $$

-- CREATE select_all POUR produits
CREATE PROCEDURE produits_select_all()
BEGIN
    SELECT * FROM produits ;
END $$

-- CREATE select_one POUR produits
CREATE PROCEDURE produits_select_one(paramID_PRODUIT INT)
BEGIN
    SELECT *
    FROM produits
    WHERE id_produit = paramID_PRODUIT ;
END $$

-- CREATE insert POUR produits
CREATE PROCEDURE produits_insert(paramID_PRODUIT
INT,paramDESIGNATION VARCHAR(50) ,paramPRIX DOUBLE,paramQTE_STOCKEE
INT,paramPHOTO VARCHAR(50) )
BEGIN
    INSERT INTO
produits(id_produit,designation,prix,qte_stockee,photo)

VALUES (paramID_PRODUIT,paramDESIGNATION,paramPRIX,paramQTE_STOCKEE,
paramPHOTO) ;
END $$

-- CREATE update POUR produits
CREATE PROCEDURE produits_update(paramID_PRODUIT
INT,paramDESIGNATION VARCHAR(50) ,paramPRIX DOUBLE,paramQTE_STOCKEE
INT,paramPHOTO VARCHAR(50) )
BEGIN
    UPDATE produits
    SET designation = paramDESIGNATION,prix = paramPRIX,qte_stockee
= paramQTE_STOCKEE,photo = paramPHOTO
    WHERE id_produit = paramID_PRODUIT ;
END $$

```

```
-- CREATE delete POUR produits
CREATE PROCEDURE produits_delete(paramID_PRODUIT INT)
BEGIN
    DELETE FROM produits
    WHERE id_produit = paramID_PRODUIT ;
END $$

DELIMITER ;
```

Autres

```
DELIMITER $$
USE xxx $$

CREATE PROCEDURE `pays_et_villes` ()
BEGIN
    SELECT nom_ville, nom
    FROM villes v LEFT OUTER JOIN clients c
    ON v.cp = c.cp
    ORDER BY nom;
END $$

CREATE PROCEDURE `produits_et_quantite_commandee` ()
BEGIN
    SELECT p.designation "Désignation", l.qte "Quantité"
    FROM produits p INNER JOIN ligcdes l
    ON p.id_produit = l.id_produit
    WHERE l.qte >= 5;
END $$

CREATE PROCEDURE `ca_par_pays_par_produit` ()
BEGIN
    SELECT pays.nom_pays, produits.designation,
           SUM(ligcdes.qte * produits.prix) AS CA
    FROM (((cours.cdes cdes
            INNER JOIN cours.clients clients
            ON (cdes.id_client = clients.id_client))
          INNER JOIN cours.villes villes ON (clients.cp =
villes.cp))
          INNER JOIN cours.pays pays ON (villes.id_pays =
pays.id_pays))
          INNER JOIN cours.ligcdes ligcdes ON (ligcdes.id_cde =
cdes.id_cde))
          INNER JOIN cours.produits produits
          ON (ligcdes.id_produit = produits.id_produit)
    GROUP BY pays.nom_pays, produits.designation;
END $$

DELIMITER ;
```

CHAPITRE 6 - CONCEPTION DE L'APPLICATION

6.1 - LES DIAGRAMMES DE SÉQUENCE DÉTAILLÉS

Les Boundaries (Pages Web, Fenêtres Swing, Activités Android, ...).

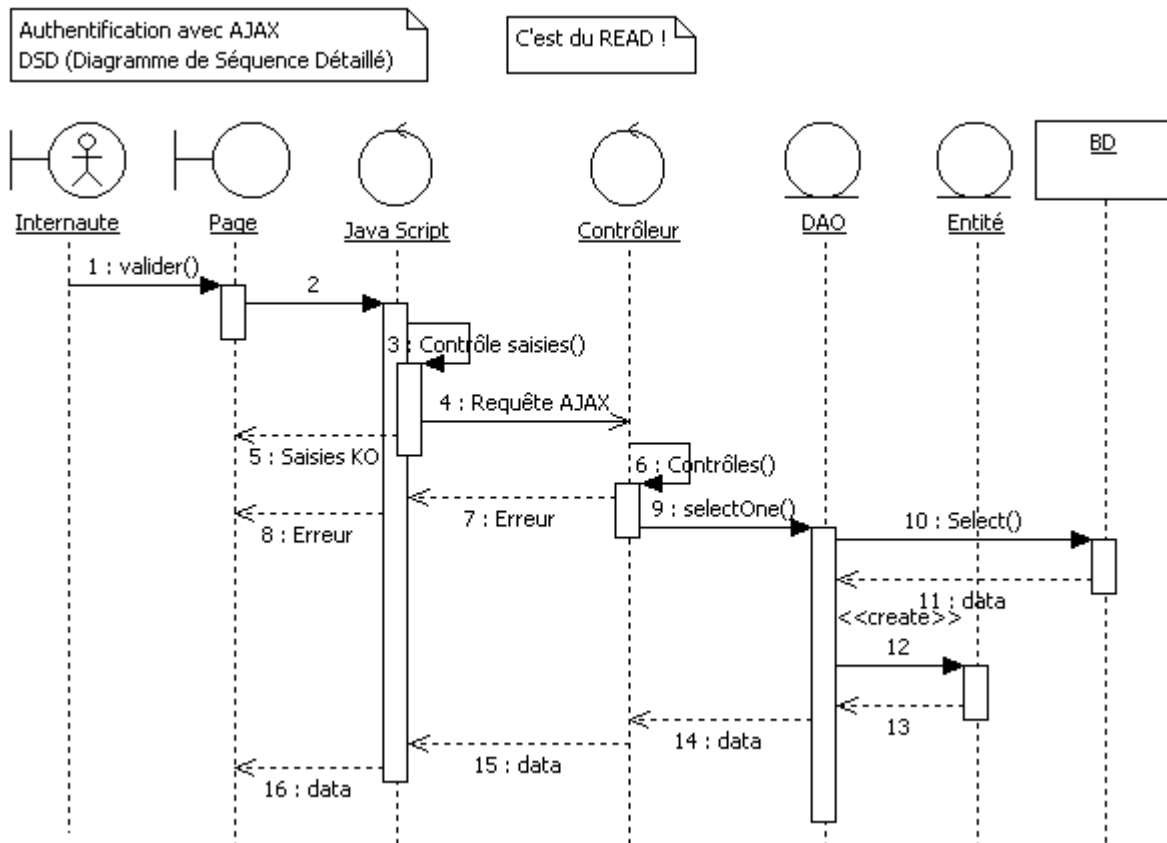
Les controls (les classes contrôleurs).

Les Entities (classes DTO) et les DAOs.

Mots clés :

A vous ...

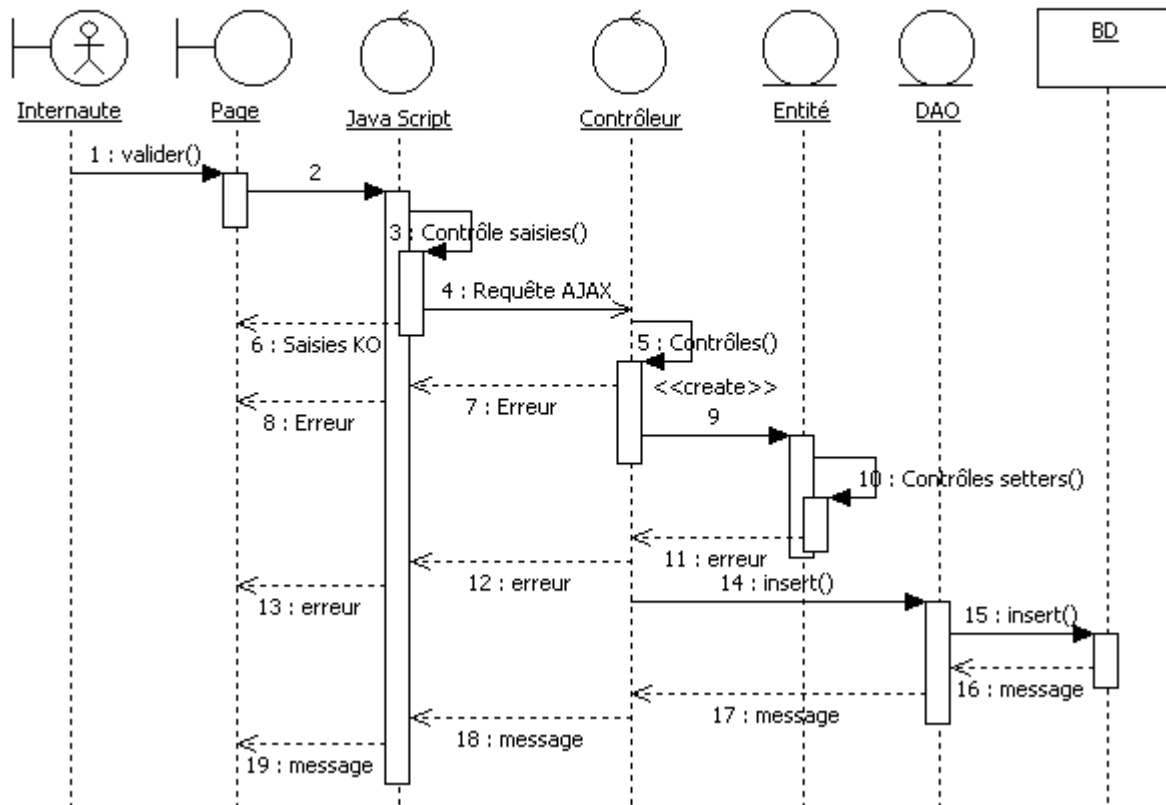
READ ...



CUD ...

Création d'un compte avec AJAX
DSD (Diagramme de Séquence Détaillé)

C'est du CREATE, UPDATE, DELETE !



6.2 - LES DIAGRAMMES D'ACTIVITÉ

Mots clés :

A vous ...

6.3 - LES DIAGRAMMES DE CLASSES PARTICIPANTES

Facultatif.

C'est un diagramme de classes où toutes les classes des DSEQ détaillés sont représentées.

Mots clés :

A vous ...

CHAPITRE 7 - DÉVELOPPEMENT

Une bonne démarche – simple (eXtreme Programming), évolutive (Scrum/UP) – est essentielle.

Ne faites pas tout en même temps.
Soyez patient !

Le postulat est que les data (BD, JSON, XML, CSV, ...) sont disponibles.

Vous avez 2 possibilités :

- ✓ Interfaces (Boundaries/Views) – Entities/Models – Controls/Controllers.
- ✓ Entities/Models – Interfaces (Boundaries/Views) - Controls/Controllers.

7.1 - TECHNOLOGIES UTILISÉES

Par exemple :

MySQL,
Hibernate,
JSF

ou

MySQL,
Symfony2.

7.2 - COMMENCEZ PAR LES INTERFACES

Statiques d'abord (La couche présentation) :

HTML/Swing controls/FX controls/Layout Android/...

Pour HTML chaque page HTML doit « montrer » la structure sémantique du code.
Pas de morcellement intempestif. Le contrôleur (et c'est encore mieux avec un template – JSF, Twig, ...) s'occupera de générer les Boundaries/Vues à partir des datas.

Testez ! Testez ! Testez !

Dynamiques ensuite (La couche dialogue) :

JavaScript et/ou jQuery/Events/Activities/...

Testez ! Testez ! Testez !

Profitez-en pour créer des bibliothèques !

7.2.1 - Copies d'écrans

7.2.2 - Codes statiques des écrans

|

7.2.3 - Codes dynamiques des écrans

|

7.3 - CONTINUEZ AVEC LA PARTIE ENTITIES/MODELS ET DAO

Fichiers properties/Connexion

Testez ! Testez ! Testez !

Beans entities/Entities PHP/...

Testez ! Testez ! Testez !

Interface DAO/DAOs/Factory de DAOs/...

Testez ! Testez ! Testez !

Du générique ? C'est anti-SUN/Oracle. Mais pas interdit !

7.4 - TERMINEZ AVEC LA « GLUE » : LES CONTROLS

AJAX.

Servlets/Codes PHP/...

Testez ! Testez ! Testez !

CHAPITRE 8 - DÉPLOIEMENT

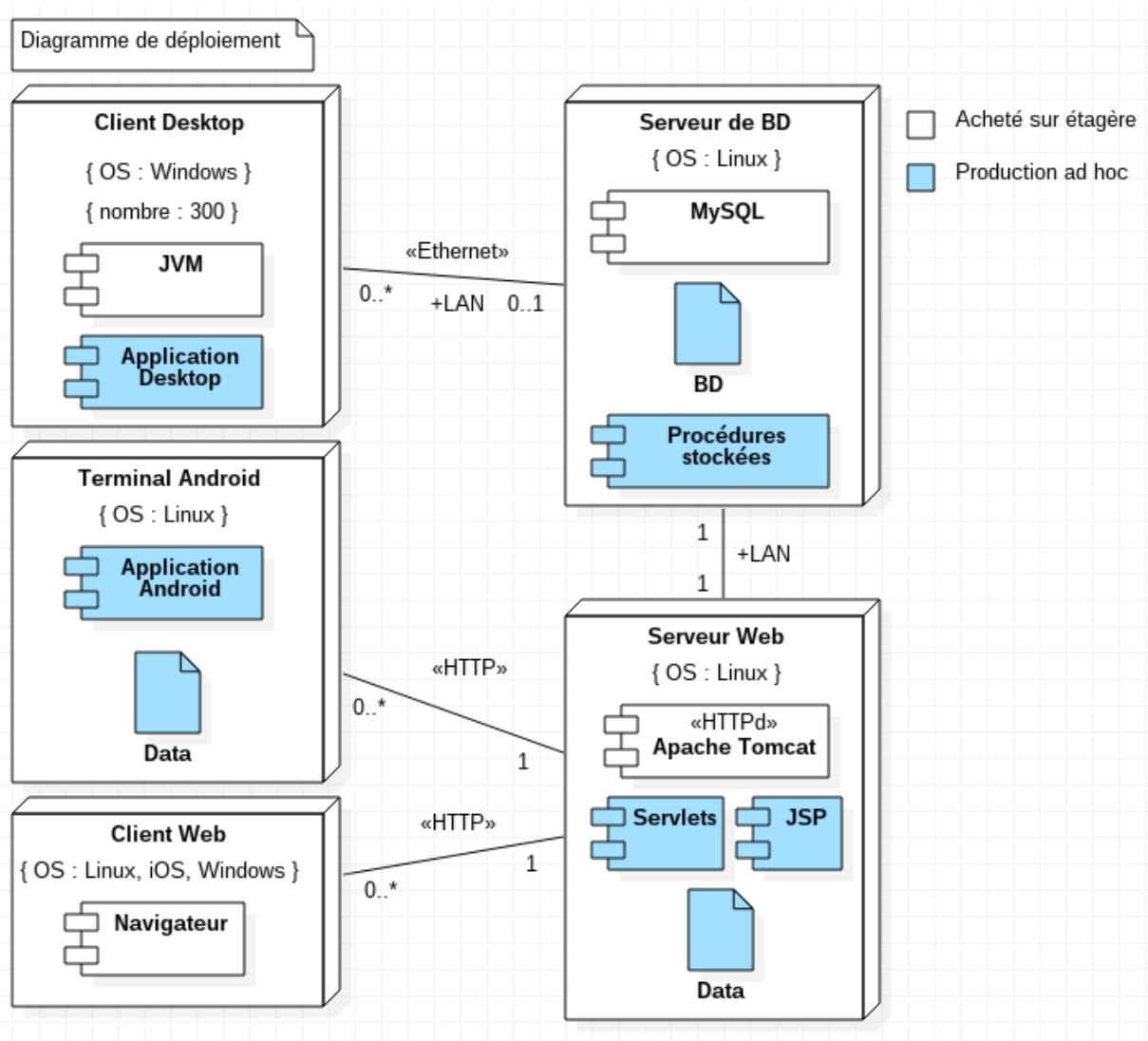
8.1 - LE DIAGRAMME DE DÉPLOIEMENT

Principalement ce diagramme doit être utile à l'administrateur Système et à l'administrateur BD.

Codes couleurs :

blanc : acheté sur étagère

bleu : production ad hoc



Mots clés :

A vous ...

8.2 - LE DÉPLOIEMENT

De la BD (exécution du code LDD - CREATE USER ..., CREATE TABLE ..., ALTER TABLE ..., CREATE INDEX ..., etc - sans oublier le CREATE DATABASE et le USE nomDeLaBD, ...).

De l'application (serveur Web, jar, war, apk, ftp, ...)

CHAPITRE 9 - LA GESTION DE PROJET

9.1.1 - UP

9.1.2 - XP

9.1.3 - SCRUM

9.1.4 - GIT

9.1.5 - MAVEN

CHAPITRE 10 - LA SÉCURITÉ

10.1 - INJECTION SQL

10.2 - INJECTION XSS

CHAPITRE 11 - FORMATION DES UTILISATEURS

CHAPITRE 12 - CONCLUSION

CHAPITRE 13 - ANNEXES

13.1 - CORRESPONDANCES PROJET/REAC

A vous de mettre en gras dans la colonne de gauche ce que vous avez réalisé dans votre projet.

| Activité | Compétence | Correspondance |
|---|--|---|
| Concevoir et développer des composants d'interface utilisateur en intégrant les recommandations de sécurité | | |
| | Maquetter une application | UML (DCU, Maquettes, DSEQ, DNAV, DAC, DET) AGL StarUML, PowerAMC |
| | Développer une interface utilisateur de type desktop | Swing ou FX |
| | Développer des composants d'accès aux données | Langage serveur (PHP, Java, Servlets) |
| | Développer la partie front-end d'une interface utilisateur web | HTML, CSS, JavaScript pour les contrôles des données entrantes, Angular, React, AJAX. Langage serveur (PHP, JSP, JSF, ...) |
| | Développer la partie back-end d'une interface utilisateur web | !!! |
| | | |
| Concevoir et développer la persistance des données en intégrant les recommandations de sécurité | | |
| | Concevoir une base de données (Schéma entité-Association, modèle physique normalisé) | UML (DCL) et/ou Merise (MCD, MOD, MLD, MPD) AGL jMerise, StarUML, PowerAMC, etc |
| | Mettre en place une base de données (intégrité des données, ...) | SQL côté administrateur Création de la BD, création des tables (les types, la PK, les index UNIQUE, ...) mise en place des contraintes (FK) BD NoSQL |
| | Développer des composants dans le langage d'une base de données | Procédures stockées avec P/SQL le langage procédural de MySQL |
| | | |

| | | |
|---|--|--|
| Concevoir et développer une application multicouche répartie en intégrant les recommandations de sécurité | | |
| | Concevoir une application | UML (DCU, DNAV, DSEQ, ...) |
| | Collaborer à la gestion d'un projet informatique et à l'organisation de l'environnement de développement | Gestion de projet. SCRUM, XP, ... |
| | Développer des composants métier | Classes « Entities » |
| | Construire une application organisée en couches | ECB (Diagrammes de séquence détaillés) |
| | Développer une application de mobilité numérique | Android Natif, ionic |
| | Préparer et exécuter les plans de tests d'une application | JUnit |
| | Préparer et exécuter le déploiement d'une application | Diagramme de déploiement (UML) Déploiement. Installation de la BD. Installation des codes HTML, CSS et JavaScript. Installation des codes PHP. |
| | | |
| Compétences transversales | | |
| | Français | |
| | Anglais | |
| | Sécurité | |

13.2 - GESTION DE PROJET

13.2.1 - git

13.2.2 - maven

<http://maven-guide-fr.erwan-alliaume.com/maven-guide-fr/site/reference/introduction.html>

13.3 - OUTILS UTILISÉS ... POUR QUELS OBJECTIFS ?

| Outil | Objectif | Commentaire |
|--------------------|-------------------------------|---|
| phpMyAdmin | Gestion de la BD. | C/S « léger ». Simple, rapide, parfois lourd (l'export par exemple) et l'interface est lourde. Bon schéma du modèle physique. Création aisée des FK. A connaître absolument car utilisé chez tous les hébergeurs. |
| WorkBench | Gestion de la BD | C/S « lourd ». Lourd, mais des résultats meilleurs pour l'export. |
| ToadForMySQL | Gestion de la BD | C/S « lourd ». Lourd mais puissant pour son QBE |
| NetBeans | IDE | Idéale pour HTML, CSS, JavaScript, PHP et Java. |
| Visual Studio Code | IDE | |
| StarUML | Modéliser en UML | Gratuit, design propre. Ne génère pas de code directement exploitable en PHP ou Java. Et pas de SQL dans tous les cas. La version 2 est disponible sur Linux, Mac et Windows. |
| PowerAMC | Modéliser en UML ou en Merise | Payant mais version d'essai assez puissante. Permet de générer du SQL, du Java, du PHP. |
| PencilEvolus | Outil de maquettage | Windows, Mac, Linux |
| Balsamiq Mockups | Outil de maquettage | |
| JustInMind | Outil de maquettage | Windows, Mac |
| GANTT Project | Diagramme de GANTT | Windows, Mac, Linux |

13.4 - BIBLIOGRAPHIE ET WEBOGRAPHIE

13.4.1 - Bibliographie

Christian Soutou, « Programmer avec MySQL », Eyrolles, 2015

Pascal Roques, « UML 2, Modéliser une application web », Eyrolles, 2008.

Christian Soutou avec la contribution de Frédéric Brouard, « UML 2 pour les bases de données », Eyrolles, 2012.

Rich Cannings, Himanshu Dwivedi, Zane Lackey , « Hacking sur le Web 2.0: Vulnérabilité du Web 2.0 et sécurisation » , Pearson, 2008.

Eric Daspet, Cyril Pierre De Geyer, « PHP 5 avancé », Eyrolles, 2012

Maxime Gréau, « Apache Maven », Eyrolles, 2011

13.4.2 - Webographie

13.4.2.1 - git

<https://git-scm.com/>

13.4.2.2 - maven

<http://maven-guide-fr.erwan-alliaume.com/maven-guide-fr/site/reference/introduction.html>

13.4.2.3 - HTML

<http://www.w3.org/>

13.4.2.4 - Java

<http://docs.oracle.com/javase/7/docs/>

<http://www.oracle.com/technetwork/java/javaee/documentation/index.html>

13.4.2.5 - PHP

<http://www.php.net/>

13.4.2.6 - SQL

<http://sqlpro.developpez.com/>

13.4.2.7 - MySQL

<http://www.mysql.fr/>

13.4.2.8 - JavaScript

<https://developer.mozilla.org/fr/docs/Web/JavaScript>

13.4.2.9 - jQuery

<https://jquery.com/>

<https://jqueryui.com/>

13.5 - GLOSSAIRE/LEXIQUE OU/ET LISTE DE MOTS-CLÉS ET SIGLES

| Mot clé | Description |
|---------|---|
| CRUD | Concerne le LMD (Langage de manipulation de données). C : create R : read U : update D : delete |
| HTTP | Protocole de communication entre 2 logiciels (un client et un serveur) |
| etc | |

13.6 - AUTRES CODES

13.6.1.1 - Code pour l'insertion de données de test dans la BD

|

13.6.1.2 - POJO complet

|

13.6.1.3 - DAO complet

|

13.6.1.4 - Autres

CHAPITRE 14 - TABLES ET INDEX

14.1 - TABLE DES ILLUSTRATIONS

Index personnalisé

| | |
|--|----|
| Démarche Roques..... | 8 |
| Diagramme de GANTT..... | 15 |
| Diagramme de séquence détaillé (READ)..... | 32 |
| Diagramme de séquence détaillé (CUD)..... | 33 |

14.2 - INDEX