

# Projet Médiathèque

---

## Rapport de travail



Master 1 TIIL

2013/2014

Grégory Harzo  
Mamadou Oury Diallo  
Mohammed Akar  
Ali Al Kaissouni

## Sommaire

I.	Présentation .....	3
II.	Travail effectué.....	3
	Explications :.....	4
III.	Bilan .....	6

## **I. Présentation**

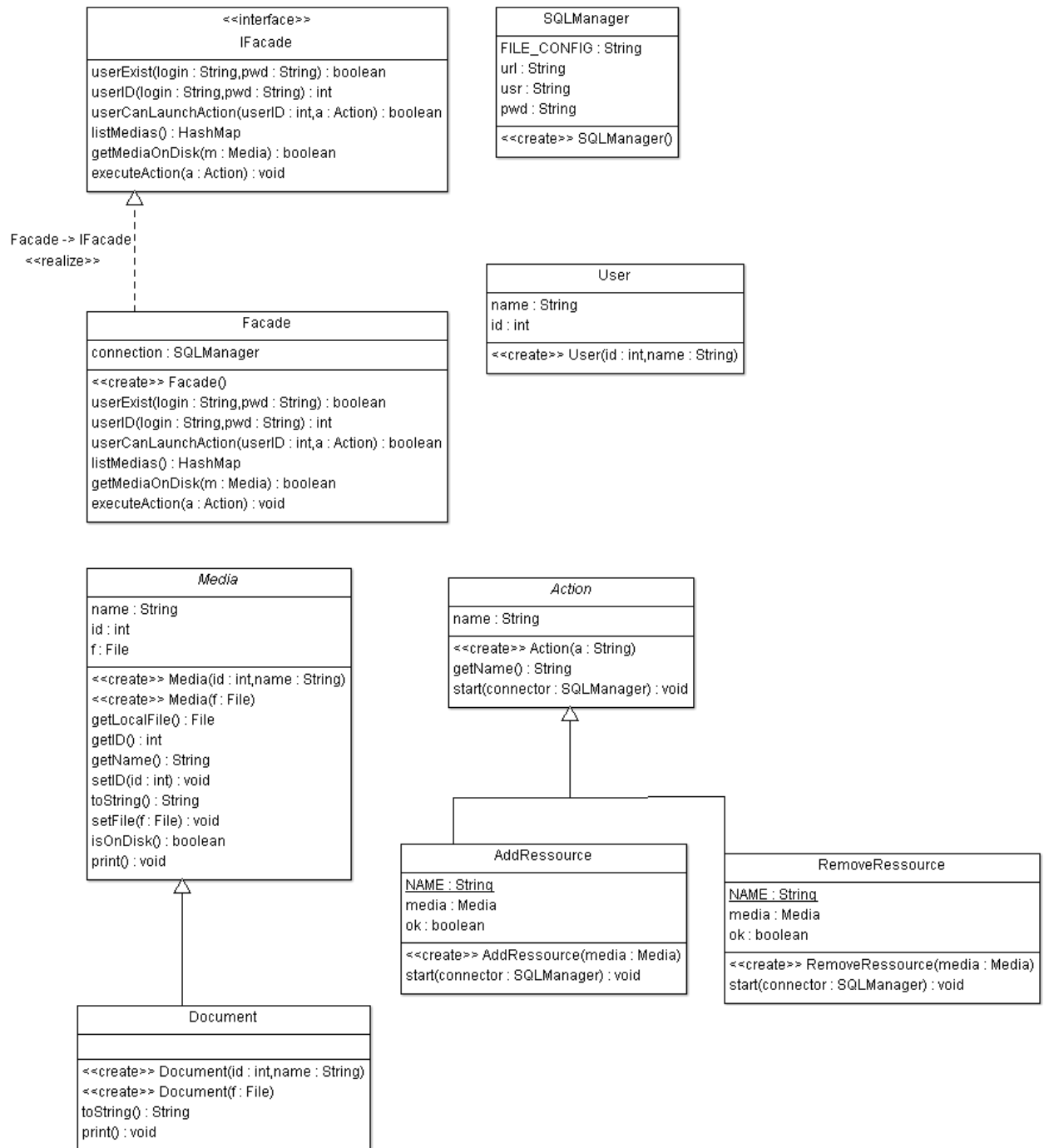
Ce projet de création d'une médiathèque s'inscrit dans le cadre de notre UE IDL enseigné à l'UBO de Brest en Master 1 TIIL. Le but est de mettre en pratique toutes les connaissances acquises sur la modélisation UML, gestion de projet (Ant, GIT), création de tests JUnit à travers ce projet. Voici un petit récapitulatif des éléments attendus :

- Modélisation UML à partir du cahier des charges avec une architecture en Facade
- Mise en place de la solution à travers une maquette
- Des tests unitaires afin de valider le fonctionnement de la solution proposée
- Un fichier de configuration Ant
- Un dépôt Git/SVN

Passons maintenant à la partie suivante qui détail le travail effectué.

## **II. Travail effectué**

Pour commencer, voici le diagramme UML de notre modèle (page suivante).



## Explications :

Les documents sont représentés par la classe « Document » qui hérite de la classe abstraite « Media ». En effet, bon nombre de médias partagent des attributs communs (id, name, file...). Rien ne nous empêche d'ajouter d'autres types de documents très simplement.

Dans les instructions qui nous ont été données, une des exigences était de proposer une architecture en façade. Nous avons donc créé une interface « IFacade » contenant l'ensemble des services proposés par la façade. La classe « Facade » implémente ces services.

Etant donné que notre client Java communique avec une Base de données, nous avons décidé de créer une classe pour toutes ces opérations de transferts. Cette classe se nomme « SQLManager ». De plus, un fichier de configuration est présent dans notre projet (configBDD.properties) afin de renseigner les champs login, mot de passe...

Un utilisateur sera représenté par la classe « User ».

Aussi, la médiathèque doit permettre l'administration et la gestion des droits pour les administrateurs, utilisateurs abonnés, utilisateurs enregistrés... Pour répondre à cette demande, nous avons décidé de créer une classe abstraite « Action ». Chaque action possible devra étendre cette dernière et avoir un nom unique. Par exemple, la classe « AddRessource » permet d'ajouter une ressource dans la base de données. Notre base de données contient pour chaque utilisateur, la liste des actions possibles. Cette solution permet de garder une liberté totale sur la gestion des droits. On peut ainsi créer un nombre de profils illimités.

La mise en place des profils pourra se faire avec des fichiers XML contenant la liste des actions possibles pour chaque profil d'utilisateur.

Voici les définitions de nos tables présentes dans notre base de données :

```
CREATE TABLE IF NOT EXISTS `t_actions` (
  `id` int(5) unsigned NOT NULL AUTO_INCREMENT,
  `name` varchar(20) NOT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8 AUTO_INCREMENT=3 ;

CREATE TABLE IF NOT EXISTS `t_medias` (
  `id` int(5) NOT NULL AUTO_INCREMENT,
  `name` varchar(20) NOT NULL,
  `binaryBlob` longblob NOT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=6 ;

CREATE TABLE IF NOT EXISTS `t_rights` (
  `id_user` int(5) unsigned NOT NULL,
  `id_action` int(5) unsigned NOT NULL,
  KEY `id_user` (`id_user`),
  KEY `id_action` (`id_action`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

CREATE TABLE IF NOT EXISTS `t_users` (
  `id` int(5) unsigned NOT NULL AUTO_INCREMENT,
  `name` varchar(12) NOT NULL,
  `password` varchar(12) NOT NULL,
  PRIMARY KEY (`id`),
  UNIQUE KEY `name` (`name`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8 AUTO_INCREMENT=2 ;

ALTER TABLE `t_rights`
  ADD CONSTRAINT `t_rights_ibfk_1` FOREIGN KEY (`id_user`) REFERENCES
```

```
`t_users` (`id`) ON DELETE CASCADE ON UPDATE CASCADE,  
  ADD CONSTRAINT `t_rights_ibfk_2` FOREIGN KEY (`id_action`) REFERENCES  
  `t_actions` (`id`) ON DELETE CASCADE ON UPDATE CASCADE;
```

Les médias sont stockés réellement dans la table « t\_medias ». Le type « long blob » permet de stocker le flux binaire des fichiers.

Un fichier « localhost.sql » est inclus dans le dossier « doc » sur GIT. Ce fichier crée la base de données « idl » avec toutes les tables et insertions de valeurs pour les tests unitaires.

Le dossier « tests » du projet contient une classe « TestFacade » qui permet de tester la facade. Un fichier de configuration ant « build.xml » permet de compiler cette classe de tests. Un fichier « TEST-testUnit.TestFacade » est généré à chaque lancement du « build.xml ». Il permet de visualiser les traces d'exécutions. Un fichier « exempleTracesExecutions.xml » contient les résultats après avoir lancé les tests de notre côté.

Un serveur GIT contient tous les documents du projet à l'adresse suivante :

<https://github.com/MasterGregg/Mediatech.git>

### III. Bilan

Voici un bref résumé des tâches terminées avec les éléments attendus :

- Un serveur GIT hébergé sur Github contenant tous les documents du projet (<https://github.com/MasterGregg/Mediatech.git>)
- Un modèle fonctionnel avec l'ensemble des classes dans le dossier « src »
- Une classe de test JUnit présente dans le dossier « tests » permettant de valider le bon fonctionnement de la facade
- Un fichier de configuration Ant « build.xml » qui compile et lance les tests unitaires. Elle génère aussi un fichier de traces « TEST-testUnit.TestFacade »
- Un fichier « localhost.sql » contenant l'ensemble des instructions SQL permettant la création complète de la base avec des insertions pour les tests JUnit (le fichier peut être importé directement depuis PhpMyAdmin dans le menu « importation base »)
- Un fichier de configuration « configBDD.properties » présent dans notre dossier src définissant les paramètres d'accès à la base « login, pass, adresse ». Normalement, il n'y a pas besoin de le modifier vous créer la base sur votre PC à parti du fichier « localhost.sql »
- Une javadoc présente dans le dossier « docs »

Ce projet nous a permis de mettre en pratique les connaissances acquises en TD/TP. Ces connaissances sont désormais des acquis qui nous serviront dans notre vie professionnelle.