

TECHNIQUES INTERNET AVANCEES
PROJET 2014-2015

TEXTFRIENDS

Mickaël TURNEL

Thanouson THAO NANTHAKOUMAN

Kévin GRILLET

Line POUVARET

Introduction

But du projet :

Il consiste en le développement d'une plate-forme permettant aux utilisateurs d'interagir autour d'un document partagé de type document texte non enrichi. Cette plate-forme est donc composée d'un client lourd, d'un client léger, d'un serveur et d'une base de données. Le client léger et le client lourd sont indépendants l'un de l'autre et communiquent tous deux avec le serveur qui fait le lien avec la base de données.

Plan :

I. Rapport de conception

1. *Diagrammes de modélisation UML*
2. *Plan du site*
3. *Templates*
4. *Autres schémas entités-association*

II. Description des fonctionnalités

III. Technologies de TIA mises en œuvre

1. *Client lourd*
2. *Client léger*
3. *Serveur*
4. *Base de données*

IV. Origine du code

1. *Librairies utilisées*
2. *Codes trouvés sur Internet*

V. Organisation

1. *Estimation du temps de travail*
2. *Répartition des tâches*
3. *Outil de gestionnaire de version*

VI. Ressources utilisées

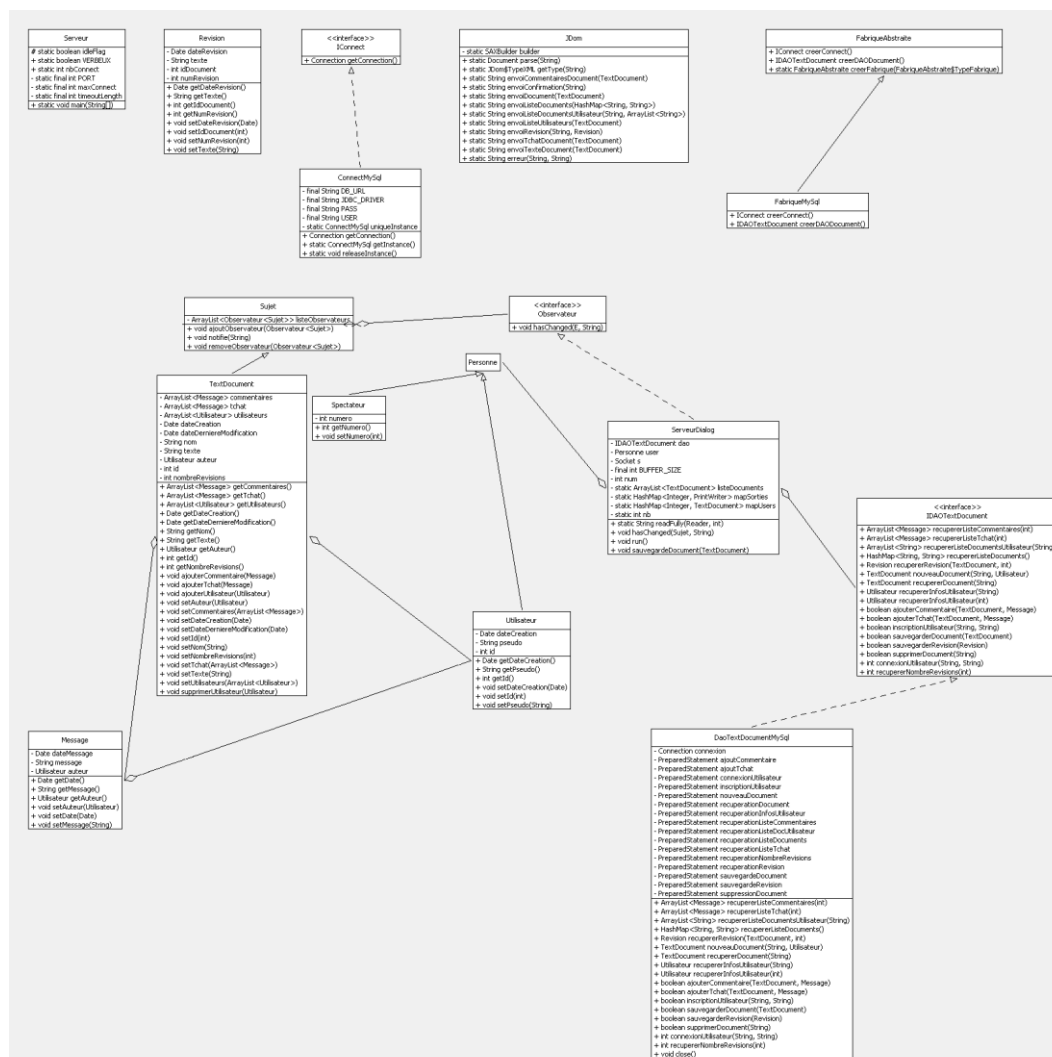
VII. Conclusion

1. *Bilan critique*
2. *Impressions*

I. Rapport de conception

1. Diagrammes de modélisation UML

- Diagramme de Classe – Client Léger :



[illegible]

2. *Plan du site*

- Client Léger :

- Barre de navigation (accessible partout sur le site, dans le header commun) :
 - Logo du site (lien sur la page d'index)
 - Accueil (lien sur la page d'index)
 - Inscription (lien vers la page d'inscription)
 - Gestion (lien vers la page de gestion, si l'utilisateur est connecté)
 - Connexion (formulaire intégré à la barre de navigation en haut)
 - Déconnexion (bouton de déconnexion, si l'utilisateur est connecté)
- Index (page d'accueil) :
 - Barre de navigation (en haut)
 - Liste de tous les documents de la base (avec lien vers la page du document sur l'image de chaque document)
- Gestion :
 - Barre de navigation (en haut)
 - Liste des documents dont l'utilisateur est l'auteur (avec icône de suppression pour chaque document)
 - Bouton de création de document
- Connexion (Redirection vers cette page en cas d'erreur de connexion via la barre) :
 - Barre de navigation (en haut)
 - Formulaire simple (Pseudo, Mot de passe) avec gestion d'affichage d'erreur
- Inscription :
 - Barre de navigation (en haut)
 - Formulaire simple (Pseudo, Mot de passe, Confirmer mot de passe) avec gestion d'affichage d'erreur
- Page d'un document :
 - Barre de navigation (en haut)
 - Nom du document
 - Boutons de rafraichissement et de sauvegarde du document
 - Contenu du document (textarea avec numéros de ligne + ligne actuelle indiquée)
 - Liste des utilisateurs connectés au document à droite
 - Chat à droite
 - Commentaires en dessous du contenu du document

- Client Lourd : Pas de plan

3. *Templates*

Client Lourd : Aucun.

Client Léger :

Connexion :

[Text Friends](#) [Accueil](#) [Inscription](#)

Connexion

Pseudo

Mot de passe

Cet utilisateur n'existe pas !

© Line Pouvaret 2015

Inscription :

[Text Friends](#) [Accueil](#) [Inscription](#)

Inscription

Pseudo

Mot de passe

Confirmer le mot de passe

© Line Pouvaret 2015

Index en mode connecté :

Text Friends


Accueil

Inscription


Gestion

Mickhopes


Se déconnecter




Nouveau
Auteur : user




Youpi !
Auteur : user




AHAHAHAHAHAHAHAH
Auteur : SuperAdmin



dfg
Auteur : SuperAdmin



TIA-2015
Auteur : Mickhopes



NOM_DU_DOCUMENT_CREE
Auteur : user

© Line Pouvaret 2015


Index en mode non connecté :

Text Friends


Accueil

Inscription


Se connecter




Nouveau
Auteur : user




Youpi !
Auteur : user




AHAHAHAHAHAHAHAH
Auteur : SuperAdmin



dfg
Auteur : SuperAdmin



TIA-2015
Auteur : Mickhopes



NOM_DU_DOCUMENT_CREE
Auteur : user

© Line Pouvaret 2015

Gestion de ses documents :

Nouveau Document



TIA-2015
✖

© Line Pouvaret 2015

Document en mode non connecté (modifications impossibles) :

[Text Friends](#) [Accueil](#) [Inscription](#)

TIA-2015

Rafraîchir

Envoyer

1 coucou

2 tu vas bien ?

3

4 oui oui et toi ?

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

33

34

35

36

37

38

39

40

41

42

43

44

45

46

47

48

Utilisateurs :

Tchat :

Miokhopes : Test
test : youpi
test : Franchement c'est trop fort
test : Test
test : coucou
test : je parie
test : encore

Commentaires

Ce document est vraiment bien !

Miokhopes - 03/06/2016

Vraiment, Vraiment bien !

Miokhopes - 03/06/2016

Oui, tu n'as pas tort

test - 03/06/2016

J'essais de mettre un commentaire avec plein de lignes

Miokhopes - 03/06/2016

J'essais de mettre un commentaire avec plein de lignes

Miokhopes - 03/06/2016

retour à la ligne

Miokhopes - 03/06/2016

© Line Pouaret 2015

Document en mode connecté :

Text Friends Accueil Inscription Gestion

Mikhopos Se déconnecter

TIA-2015

Rafatche

Envoyer

1	coucou
2	tu vas bien ?
3	
4	oui oui et toi ?
5	
6	
7	
8	
9	
10	
11	
12	
13	
14	
15	
16	
17	
18	
19	
20	
21	
22	
23	
24	
25	
26	
27	
28	
29	
30	
31	
32	
33	
34	
35	
36	
37	
38	
39	
40	
41	
42	
43	
44	
45	
46	
47	
48	

Utilisateurs :

* Mikhopos

Tchat :

Mikhopos : Test
test : youpi
test : Franchement c'est trop fort
test : Test
test : coucou
test : je parle
test : encore

Envoyer

Commentaires

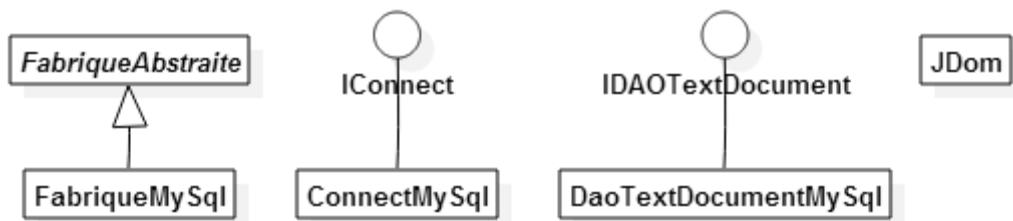
Ce document est vraiment bien !	Mikhopos - 03/06/2016
Vraiment, Vraiment bien !	Mikhopos - 03/06/2016
Oui, tu n'as pas tort	test - 03/06/2016
J'essais de mettre un commentaire avec plein de lignes	Mikhopos - 03/06/2016
J'essais de mettre un commentaire avec plein de lignes	Mikhopos - 03/06/2016
retour à la ligne	Mikhopos - 03/06/2016

Envoyer

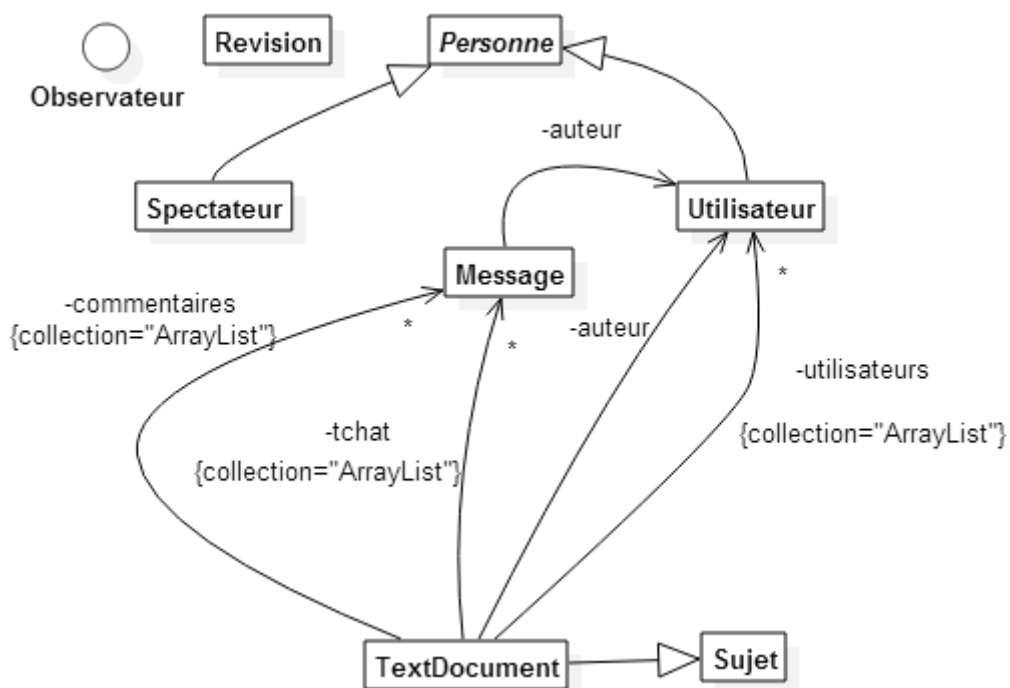
© Line Pouillet 2015

4. *Autres schémas entités-association*

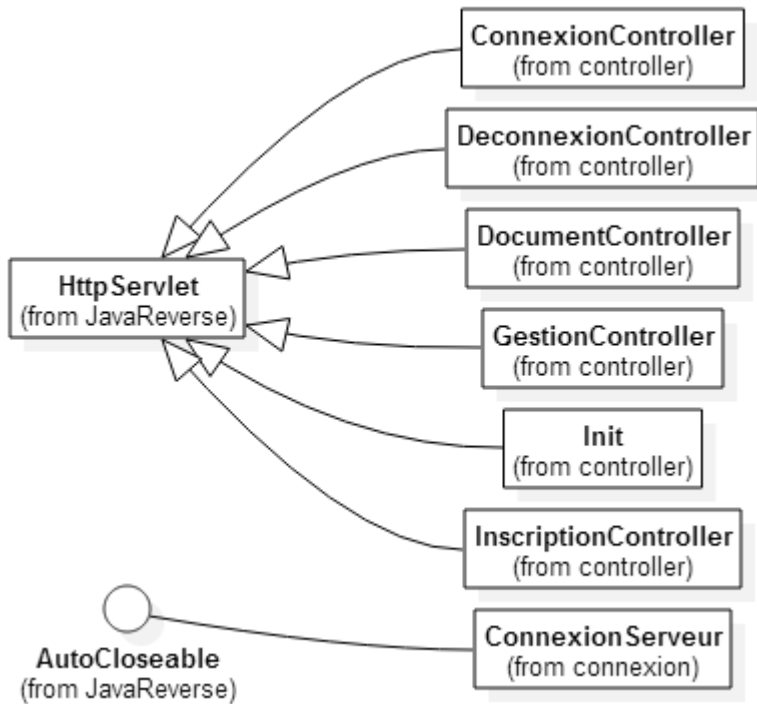
- Diagramme UML – Interface Serveur – Base de données :



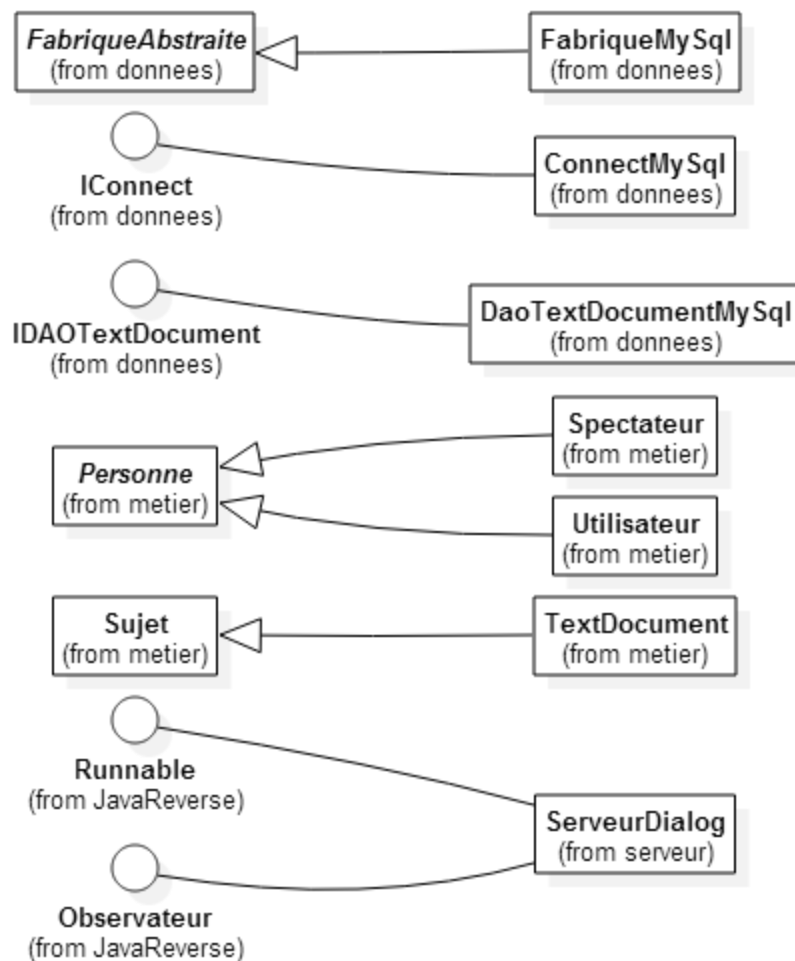
- Diagramme UML - Hiérarchie Métier :



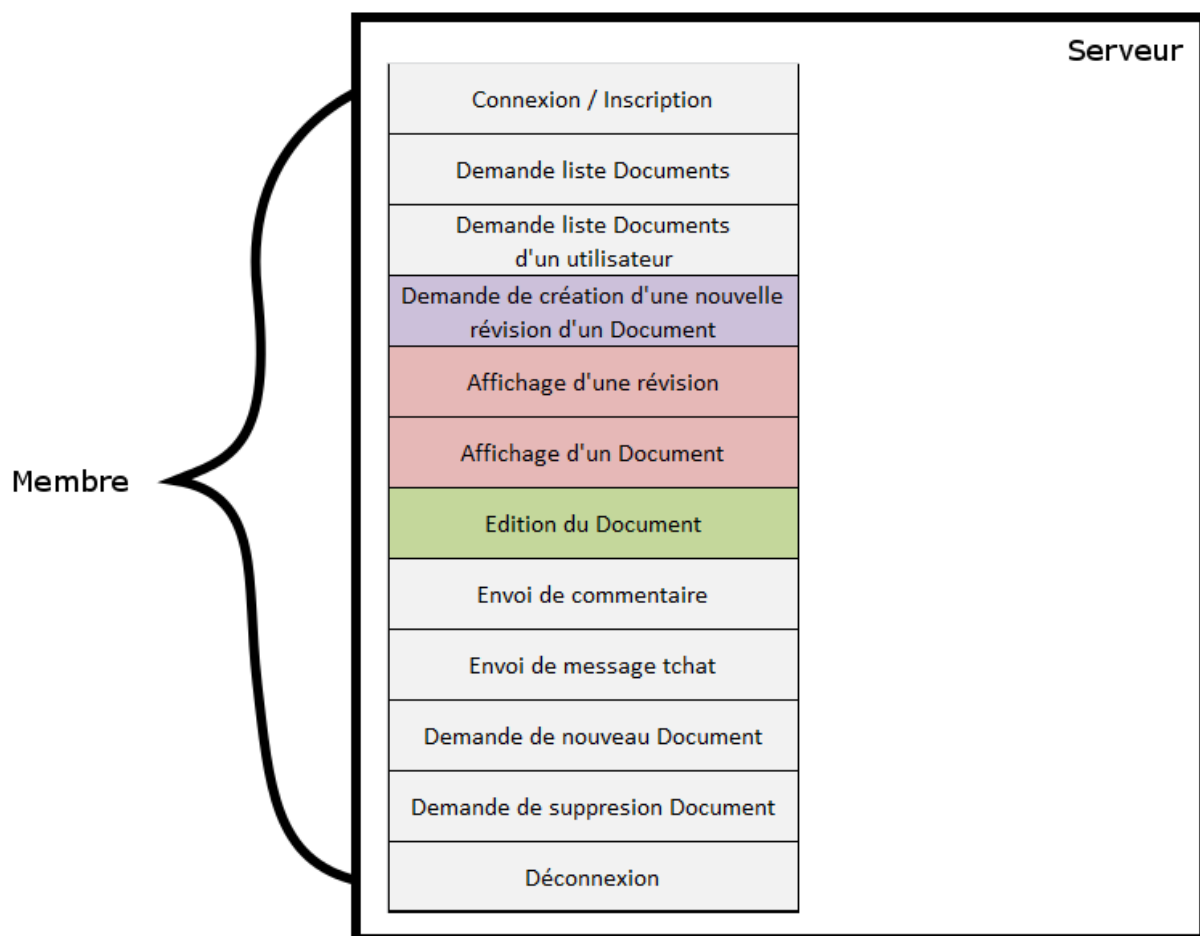
- Diagramme UML – Hiérarchie Client Léger :

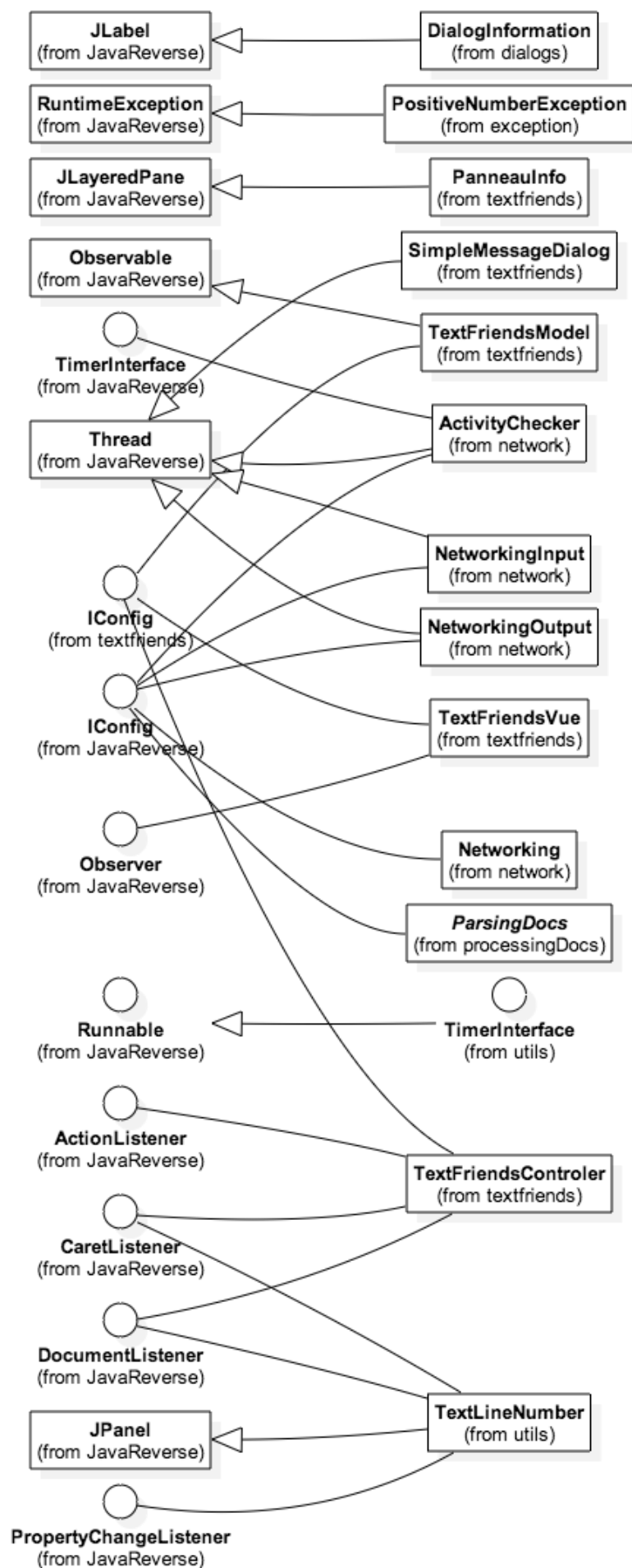


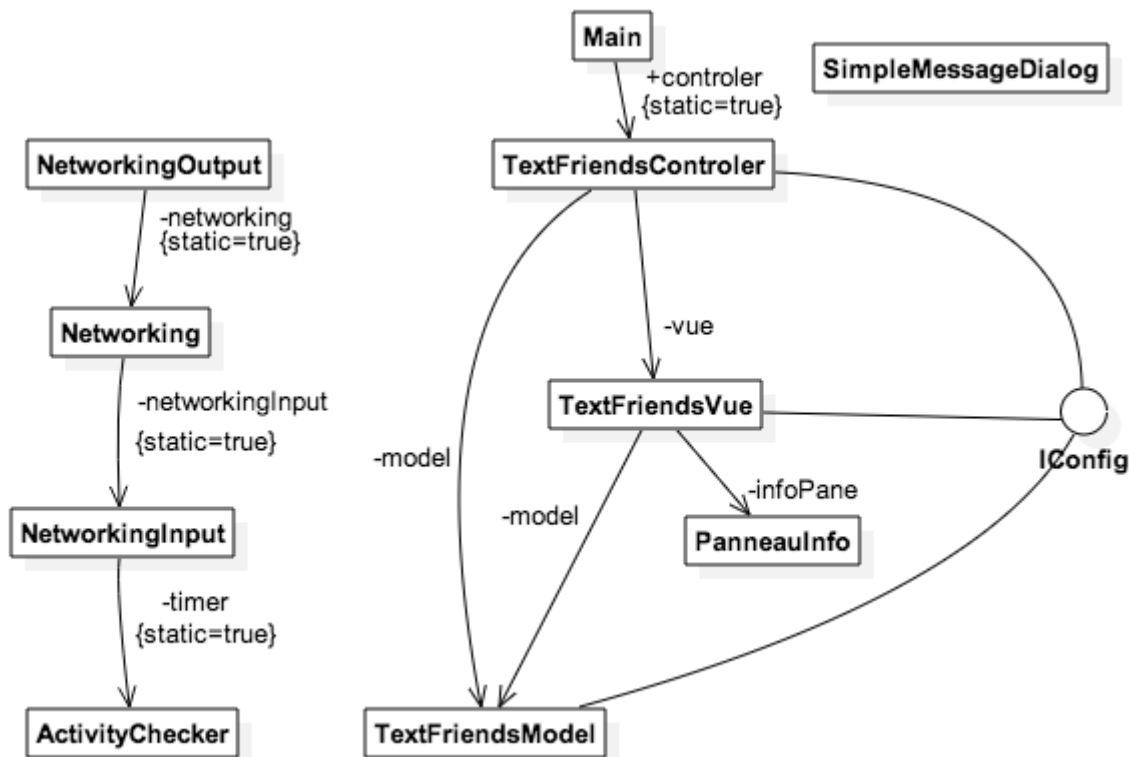
- Diagramme UML – Hiérarchie Serveur :



- Diagramme Requêtes Client (Voir fichier Excel) :







II. Description des fonctionnalités

- Client Léger :

Fonctionnalités implantées:

- connexion d'un utilisateur (à condition qu'il soit enregistré dans la base de données)
- déconnexion d'un utilisateur (qui est connecté)
- inscription d'un nouvel utilisateur (à condition qu'il ne soit pas déjà inscrit et que les mots de passe correspondent)
- affichage de la liste des documents dans index
- affichage de la liste des documents d'un utilisateur dans gestion
- ajout/suppression d'un document dans gestion
- affichage d'un document en lecture seule en mode déconnecté (tchat et commentaires en lecture seule également)
- possibilité d'ajouter un commentaire sur un document en mode connecté
- possibilité d'ajouter un message dans le tchat sur un document en mode connecté
- affichage de la liste des utilisateurs connectés à un document
- édition, sauvegarde et rafraichissement d'un document en mode connecté (en mode déconnecté : rafraichissement seulement)
- Ctrl-Z : Undo déjà intégré dans le textarea qui correspond au document
- affichage des numéros de ligne du document ainsi que ligne actuelle en rouge

Fonctionnalités non implantées :

- recherche d'un document dans les listes (gestion et index)
- édition d'un document en temps réel (rafraichissement et sauvegarde à chaque modification)
- messages du tchat enregistrés tant que l'utilisateur est en session (effacement à la déconnexion)
- documents privés protégés par mot de passe
- révisions d'un document
- messagerie privée (peer-to-peer)

- Améliorations possibles du client léger :

- cookie de session pour rester connecté ("Remember me")
- recherche dynamique (avec utilisation d'expressions rationnelles) : la liste des documents se met à jour à chaque fois qu'un utilisateur entre un caractère dans le champ de recherche
- miniatures réelles des documents sur les listes
- gestion du profil utilisateur (modification mot de passe, etc.)
- une connexion admin (permettant de gérer les commentaires, chat, documents et utilisateurs)
- texte enrichi dans un document
- coloration syntaxique du texte de document (pour du code source)
- une option pour télécharger le document sous un format ".txt"
- ajout d'un logo personnalisé en rapport avec le nom de l'application ("TextFriends")
- hébergement web
- gestion d'amis

- Client Lourd :

Fonctionnalités implémentées :

- Inscription d'un utilisateur.
- Connexion / Déconnexion d'un utilisateur.
- Affichage / Consultation de Documents ou de ses Documents.
- Suppression de l'un de ses Documents.
- Edition d'un Document.
- Envoi de message Chat.
- Affichage des renseignements liés au Document.
- Toutes les fonctionnalités d'un Bloc-Note classique (Copier, coller, sauvegarde, charger ...).

Fonctionnalités non implémentées :

- **Messagerie privée (Client – Client).**
- **Ecriture d'un commentaire.**
- **Certaines requêtes dysfonctionnelles à consolider.**
- **Edition en direct.**

III. Technologies de TIA mises en œuvre

1. *Client lourd*

- Client-Serveur d'application ;
- Client-Serveur données ;
- XML pour les transactions avec le serveur ;
- Swing pour tout ce qui est affichage ;
- Il fut un temps Singleton et MVC afin d'augmenter la lisibilité du code. Le manque de temps nous a pris de court et cela a impliqué des mesures destructives des « patterns ».

2. *Client léger*

Architecture MVC

Nous avons utilisé des JSP pour les vues de notre site.

Nous avons associé un header et un footer (JSP modulables) communs à toutes les JSP (sauf document.jsp)

Les servlets nous servent de contrôleurs (action lors d'envoi de formulaire et d'accès à une page).

Nous nous servons de deux JavaBeans "Message" et "TextDocument".

Développement côté client

Nous avons utilisé des script Javascript dans un fichier custom.js inclus dans le footer. (et d'autres scripts inclus directement dans la page document.jsp)

Nous nous sommes servis de JQuery car plus simple d'utilisation et pour la possibilité d'utiliser Ajax ainsi que d'un plugin nommé JQueryLinedTextArea qui permet d'afficher les numéros de ligne sur un textarea. Ajax nous a permis de récupérer une page directement dans le Javascript nous permettant ainsi de rafraichir certaines informations dans une page sans rafraichir cette dernière.

Pour ce qui est de l'apparence du site, nous avons inclus le framework Bootstrap ainsi qu'un fichier custom.css afin de personnaliser le plus possible notre interface.

La barre de navigation de base de bootstrap est d'ailleurs particulièrement pratique lors de la navigation sur le site.

Connexion au serveur

La connexion au serveur se fait via socket TCP et ne comporte qu'une seule instance de connexion (Singleton).

Nous avons une classe qui parse le message envoyé par le serveur (chaîne de caractères au format XML) et permet de détecter le type de message.

Et également une autre classe qui va générer les requêtes sous la forme d'une chaîne de caractères (au format XML) à envoyer au serveur.

3. *Serveur*

Design Pattern

La connexion en base de données se fait via 3 types de design pattern. Fabrique Abstraite pour créer notre fabrique correspondant au type de base de données à utiliser. Singleton Pour la connexion en elle-même à la base de données et L'utilisation de Data Access Object pour pouvoir communiquer avec la base de données. Le pattern Observateur pour pouvoir notifier un changement dans un document.

XML et DTD

Le serveur et les clients discutent en utilisant un format XML, le client a donc les DTDs correspondantes aux messages que lui envoie le serveur et inversement pour le serveur. Les fichiers XML sont gérés par la librairie JDOM.

Base de données

Le serveur fait l'intermédiaire entre le client et la base de données. On utilise la librairie JDBC pour pouvoir donc se connecter à notre base de données. Cependant la base de données est stockée sur le site alwaysdata.com car c'était plus pratique pour pouvoir tester notre application.

TCP

Le serveur et le client communique via TCP.

Thread

Le serveur crée un thread pour chaque utilisateur qui se connecte dessus.

IV.Origine du code

1. *Librairies utilisées*

- Client Léger –

JDOM2, JQuery, JQuery Lined TextArea, Framework Bootstrap

- Client Lourd –

JDOM2 : La librairie facilitant la lecture et la génération de fichiers XML.

- Serveur –

JDOM2, JDBC

2. *Codes trouvés sur Internet*

Chacun des codes trouvés sur internet est signalé dans notre code avec un lien indiquant où.

Client Léger –

JQuery Lined Textarea : code trouvé sur internet pour pouvoir changer la ligne sélectionnée avec un Lined Textarea

Bootstrap : code trouvé sur internet pour la génération du formulaire d'inscription

Serveur –

TCP : code trouvé sur internet pour pourvoir un certain nombre de caractère du StreamReader

Client Lourd –

TextLineNumber : Classe qui gère la numérotation des lignes (La banderole sur la gauche).

LineOffset : Classe qui complète TextLineNumber ; Elle permet de connaître la position du chariot sur la ligne courante. Partiellement personnel consolidé, partiellement pris sur Internet.

V. Organisation

1. Estimation du temps de travail

Mickaël TURNEL : Environ 80 heures réparties sur les 2 dernières semaines + environ 10 heures avant ça.

Thanouson THAO NANTHAKOUMAN : Environ $6 \times 24 = 144$ heures étalées sur environs 14 jours.

Kévin GRILLET : Environ 120 heures étalés dès le lancement (24 Mars) jusqu'au 24 Avril.

Line POUVARET : Environ 60 heures réparties sur les 2 dernières semaines

2. Répartition des tâches

Mickaël TURNEL : Serveur + Aide sur Client Léger

Thanouson THAO NANTHAKOUMAN : Interface Client Lourd – Serveur + Un peu de Swing

Kévin GRILLET : Swing + Debug Client Lourd

Line POUVARET : Client Léger

3. Outil de gestionnaire de version

GIT – GitHub, Inc.

Nous avons utilisé GitHub comme gestionnaire de version et grâce à **Rémi Emonet**, nous avons pu créer un projet privé GitHub.

VI. Ressources utilisées

Cours, Internet (Java, Oracle), StackOverflow, JSFIDDLE.

VII. Conclusion

1. Bilan critique

Avec du recul, en écrivant ce texte, il est clair que l'ensemble est encore à un stade (avancé) de développement. On aurait tous voulu avoir le temps d'implanter les fonctionnalités ainsi que les idées en surplus d'améliorations.

De nombreux problèmes et dysfonctionnements ont survécu ces dix derniers jours, avec une certaine pression sur la date échéante qui s'approchait petit à petit. Puis les détails ont été rapidement fixés voire calés pour faire tourner les deux types de Client.

Cela n'est pas dû aux difficultés à mettre en place les technologies mais plutôt de les relier pour assurer une connexion (en TCP) ET une connectivité permanente (mais avec un système de « Timeout ») qui nous a pris une majorité du temps.

Nous nous n'attendions pas à des résultats sublimes, nous avons essayé de proposer un éditeur de texte non pas exhaustif sur ses fonctionnalités mais étudié pour une mise en réseau et à long terme pour s'approcher du concept de « Titanpad ».

Le Serveur ayant été livré, avec les requêtes joignables, assez tardivement (Mi-Avril), nous nous sommes consacrés aux demandes possibles par le Client sans nous plonger dans l'esthétisme, ni l'ergonomie, ni la sécurité des informations et les idées d'amélioration non plus.

La communication du Client Léger et du Client Lourd avec le Serveur (implicitement avec la Base de Données) est, nous pensons, l'essentiel de notre développement.
Sans cette interface Client – Serveur et vice-versa, rien n'aurait été possible.

2. Impressions

Nous avons eu une bonne impression de ce projet parce que cette année, il est bien plus complet et plus précis, autant pour ses spécificités (Technologies) que son domaine d'application (Logiciel). Et ce, malgré son sujet bien trop vaste, pas d'idées précises de ce qui est attendu de l'application.

Nous devons avouer que suivre un schéma Model View Controller (Framework) est particulièrement pratique mais complexe, cela a été instructif.
De même en ce qui concerne XML, JSP et Servlets qui nous ont permis de nous ouvrir vers un nouvel horizon : Le Web.

Dans l'ensemble, pour convenir des solutions à des obstacles durant le développement, nous nous sommes orientés vers des sites Internet. A titre personnel, cela signifie qu'il fallait énormément consulter pour fabriquer selon nos désirs ce que l'on souhaitait ou simplement répondre à une problématique d'utilisation d'objets ou de fonctions, c'est de la documentation. Il est donc important de citer que de nombreux outils semblaient être à notre disposition mais que nous manquions d'expérience pour les appréhender sans doutes et avec convenance.

On laisse le dernier mot à Kévin G., Kévin, c'est à vous :

« Ctrl+P »