

Projet/TP de module

Apprentissage des langues

Thibaut Roudel

IMR- 1^e année
Année 2016-2017

Table des matières

1 Introduction.....	1
2 Organisation du code.....	2
3 Guide d'utilisation.....	6
4 Justification des choix.....	10
5 Conclusion.....	11

1 Introduction

Dans le cadre de mon apprentissage du langage de programmation orienté objet qu'est le Java, j'ai développé une première version d'une application d'apprentissage des langues, disponible avec une interface graphique en Java Swing.

Ce projet s'est découpé en plusieurs étapes :

- Étude du cahier des charges
- Partie 1 sans interface graphique :
 - Élaboration du diagramme de classes
 - Développement de l'application
- Partie 2 avec interface graphique :
 - Reprise du code précédent avec adaptations
 - Mise en place du modèle MVC (Model View Controller)
 - Révision du diagramme de classes
 - Réalisation de maquettes
 - Développement du code

Le présent rapport a pour but de présenter la seconde partie du projet puisqu'un précédent rapport a déjà été fait sur la première partie du projet. Pour ce faire, je commencerai par présenter l'organisation du code avec le nouveau diagramme de classes, puis je ferai un guide d'utilisation ayant pour but d'aider un nouvel utilisateur. Puis je justifierai mes choix qui m'ont semblé important pour le bon fonctionnement du service ou pour pousser plus loin mon apprentissage du langage de programmation. Enfin, je conclurai sur ce projet avec les difficultés rencontrées, les apports

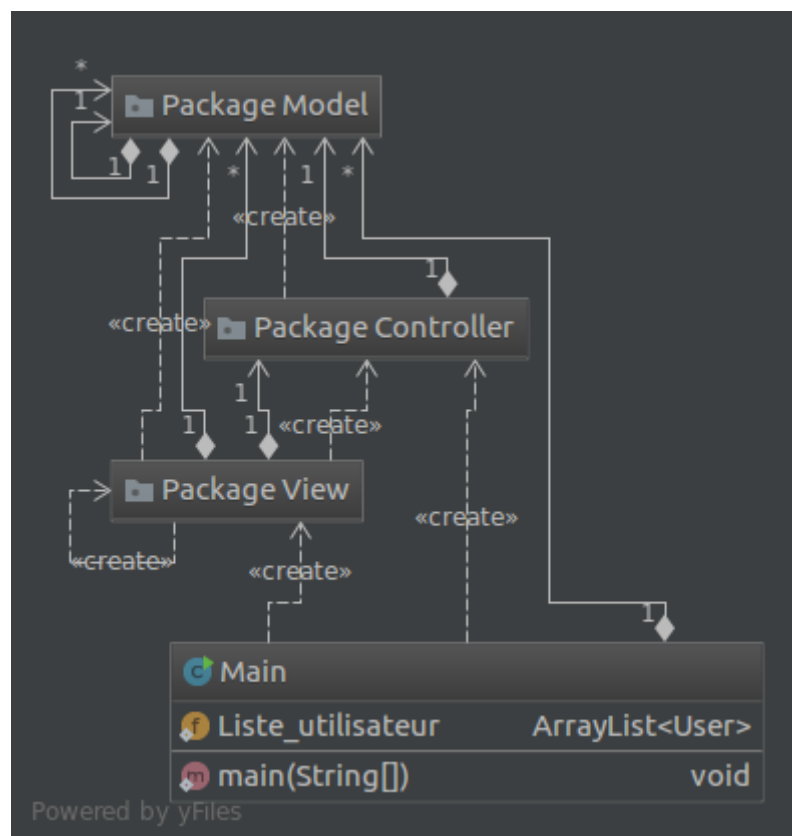
1 Introduction

du projets et les évolutions auxquelles j'ai pu penser.

2 Organisation du code

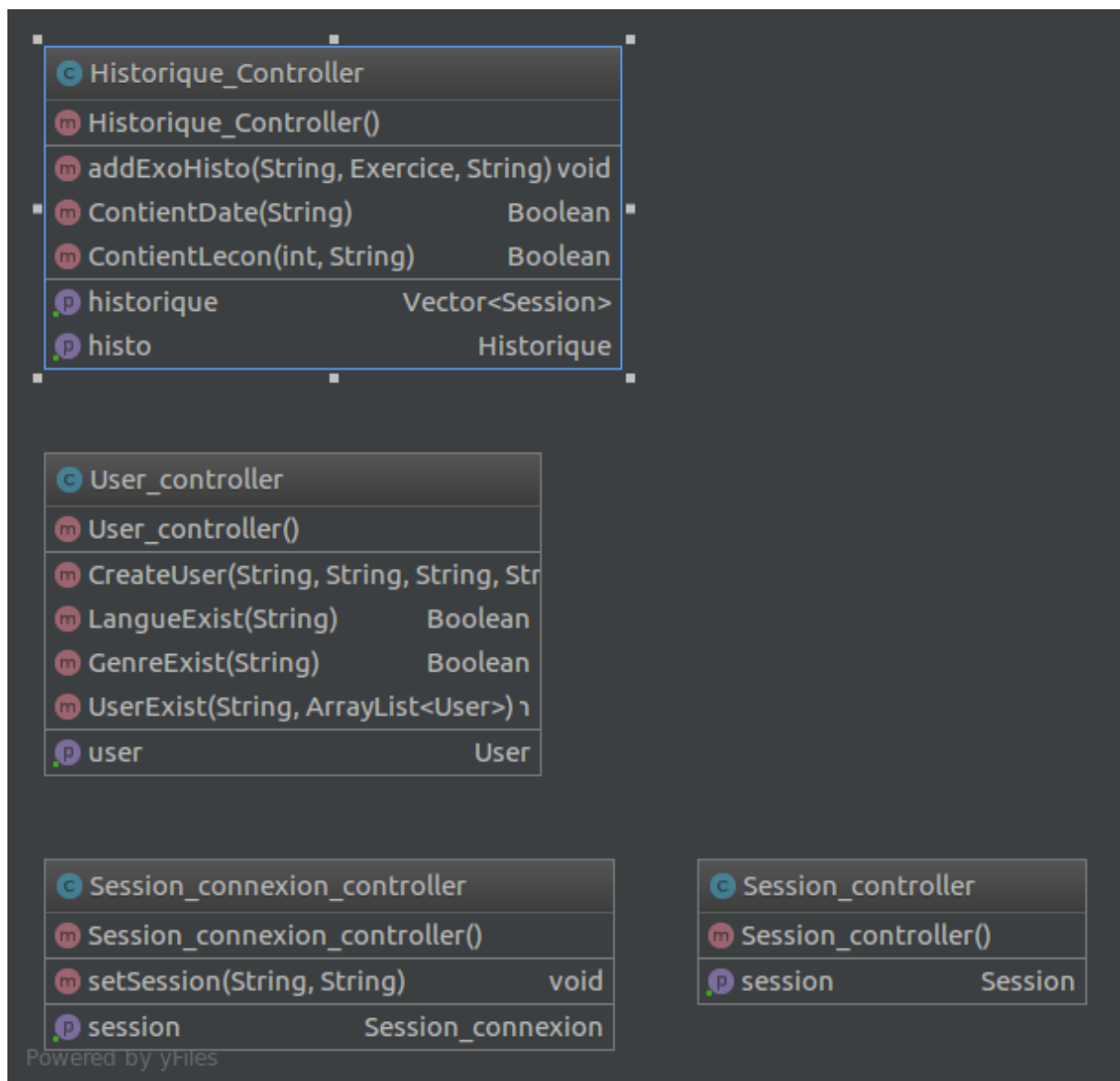
Le code est présenté par module qui tiennent compte du modèle MVC.

Diagramme vue de la racine :



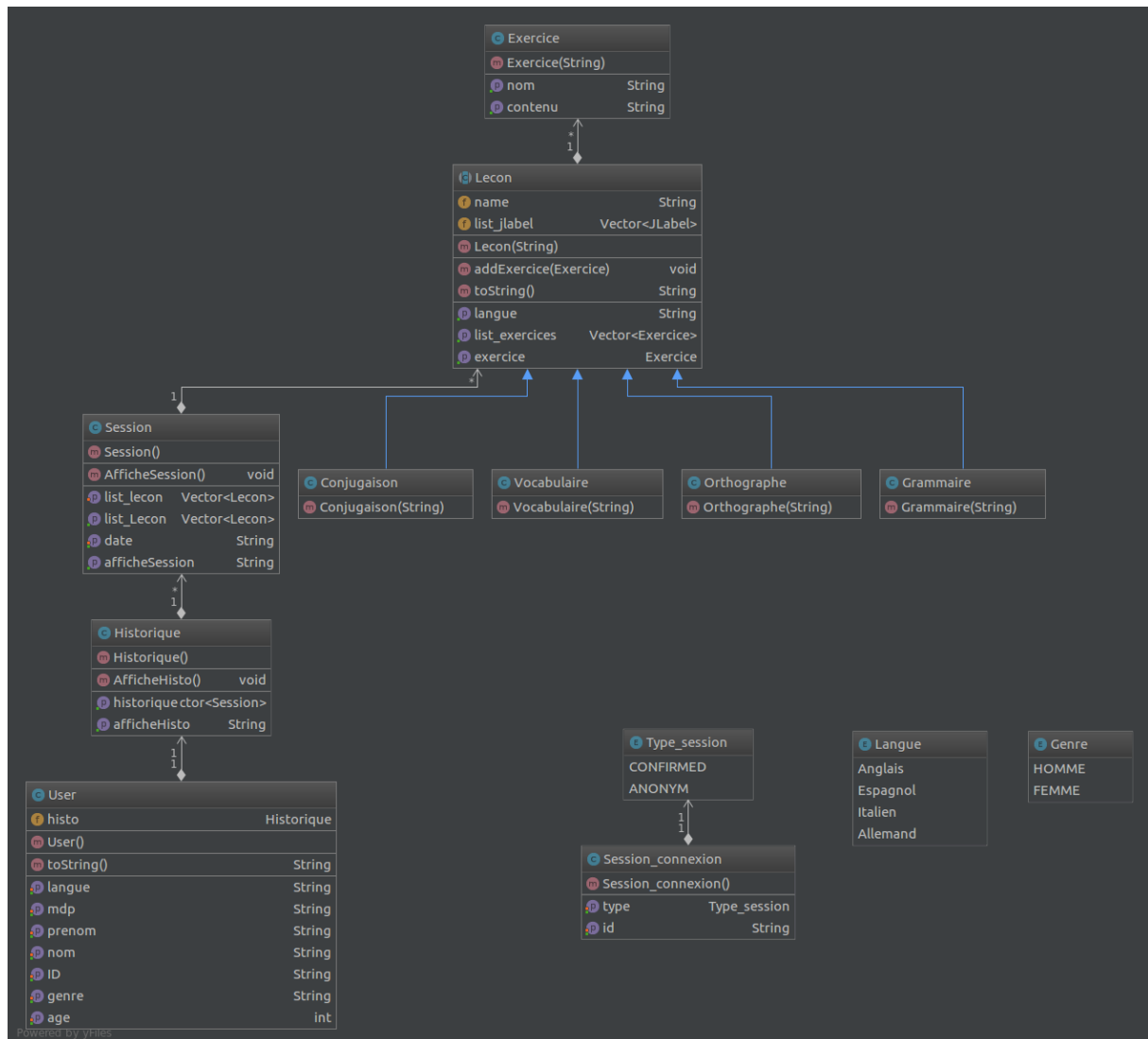
2 Organisation du code

Diagramme du package controller :



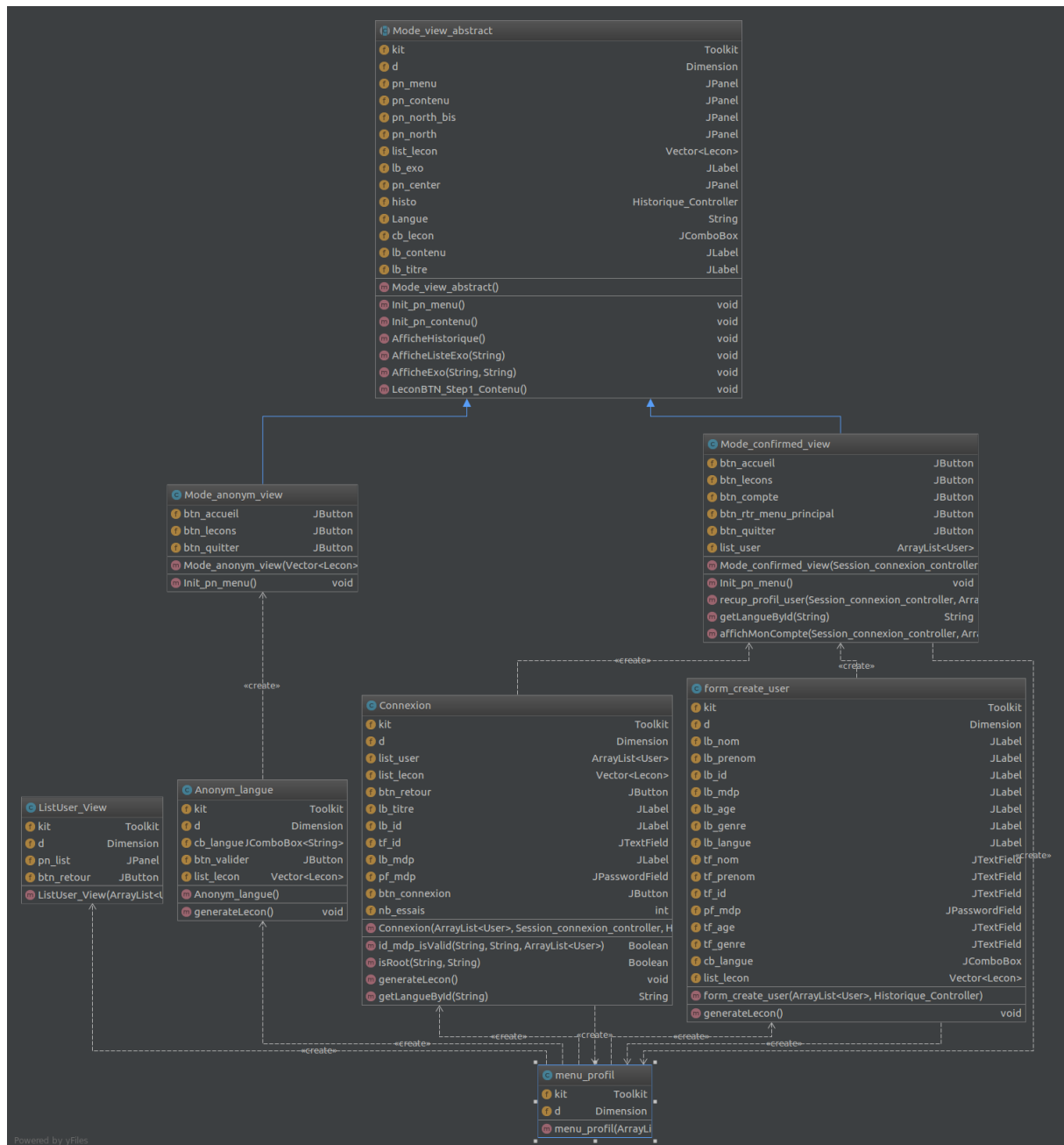
2 Organisation du code

Diagramme du package model :



2 Organisation du code

Diagramme du package view :



Pour plus d'informations sur le code en lui-même ainsi que les méthodes de chaque classe, une JavaDoc est disponible avec ce rapport.

3 Guide d'utilisation

Au lancement, l'utilisateur se trouve face à un menu lui laissant 4 choix possibles :

- Se connecter (en utilisant ses identifiants)
- Créer un compte
- Anonyme (aucun identifiant nécessaire)
- Liste des utilisateurs
- Quitter (fermeture de l'application)

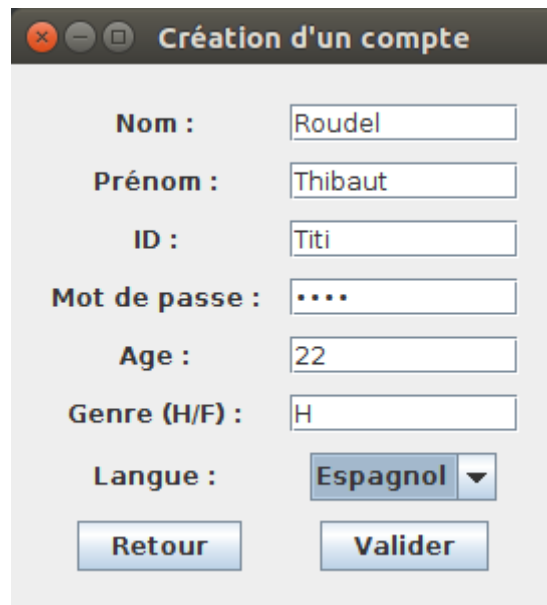


Illustration 1: Vue du menu principal (vue1)

Afin de simuler un utilisateur, nous allons créer un compte pour commencer.

3 Guide d'utilisation

Un formulaire apparaît, nécessitant le remplissage de champs tels que le prénom, le nom, un ID, un mot de passe, l'âge, un genre (à préciser avec soit la lettre '**H**' soit '**F**' et une langue dans une liste.



Création d'un compte

Nom :

Prénom :

ID :

Mot de passe :

Age :

Genre (H/F) :

Langue :

Illustration 2: Vue du formulaire d'inscription (vue2)

Après avoir validé, si les champs sont valides on arrive sur un accueil « d'utilisateur enregistré ».

Mais revenons en arrière, puisqu'on a un compte on peut cliquer sur « Se connecter » au menu principal et renseigner nos identifiants comme suit.

3 Guide d'utilisation

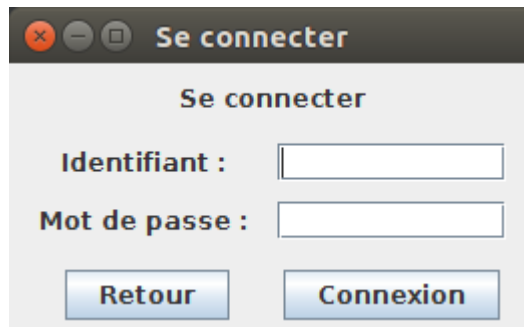


Illustration 3: Vue de l'interface de connexion (vue3)

Si la connexion est validée on rejoint l'accueil d'utilisateur enregistré. Avec un message de confirmation de connexion.

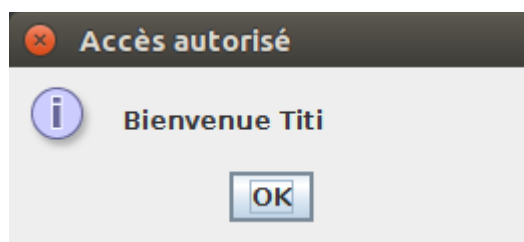


Illustration 4: Confirmation de connexion

Dans le cas contraire, où les identifiants saisis sont incorrects un message d'erreur apparaît informant l'utilisateur du nombre d'essais qu'il lui reste. Dans le cas où les 3 essais de connexion de l'utilisateur sont épuisés, ce dernier est reconduit vers le menu principal.

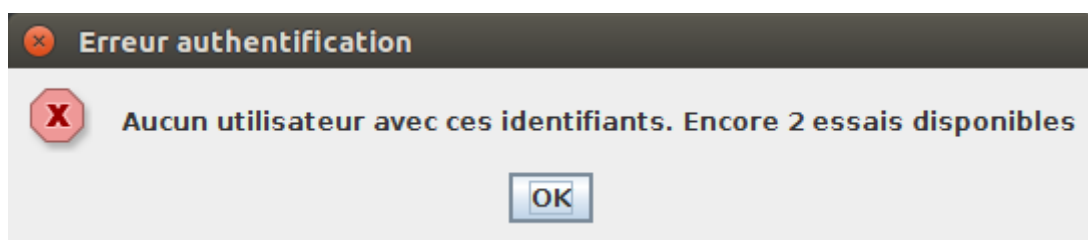


Illustration 5: Erreur lors de la connexion

3 Guide d'utilisation

Supposons maintenant que les étapes précédentes se sont bien passées, l'utilisateur est conduit, après connexion, vers son accueil qui lui affiche son historique s'il en existe un.



Illustration 6: Accueil mode utilisateur enregistré (View6)

Puisque nous sommes dans le cas d'un nouvel utilisateur, aucun historique n'existe donc nous allons effectuer quelques leçons. Pour ce faire, il suffit de cliquer sur le bouton « Leçon », actualisant alors le contenu de la fenêtre, en proposant une liste déroulante de types de leçons adaptées à la langue de l'utilisateur. Lors de la sélection d'un des choix possibles, instantanément une nouvelle liste déroulante fait son apparition proposant les exercices disponibles pour le type de leçon choisi.

3 Guide d'utilisation

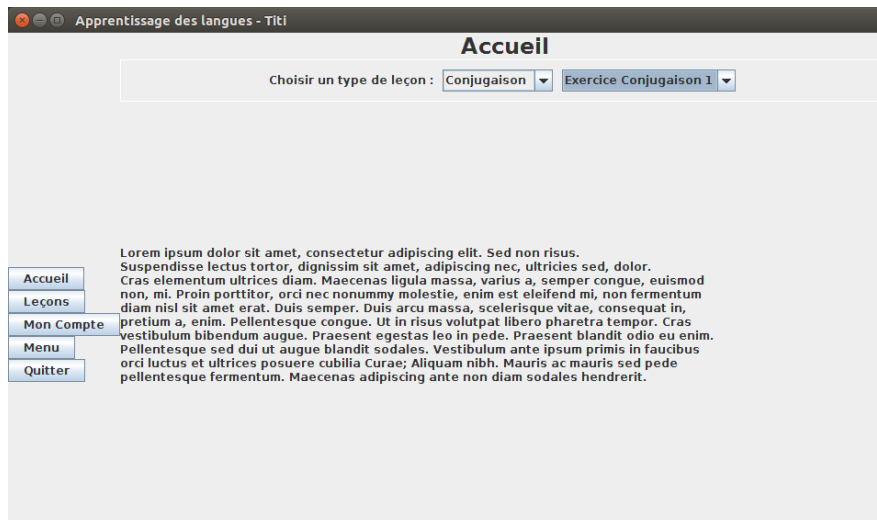


Illustration 7: Leçon mode utilisateur enregistré (vue7)

En choisissant un exercice, son contenu apparaît et il est automatiquement ajouté à l'historique de l'utilisateur comme exercice effectué. Maintenant que plusieurs exercices ont été fait, on peut les retrouver dans l'historique en revenant à l'accueil.



Illustration 8: Accueil avec historique utilisateur enregistré (vue8)

De plus, l'utilisateur peut voir les informations de son compte grâce au bouton « Mon Compte ». Enfin il est possible de revenir au menu

3 Guide d'utilisation

principal de l'application par le bouton « Menu ».

L'ensemble des fonctionnalités présentées précédemment son bien-sûr disponibles en mode anonyme(cf. Annexes 1 et 2), excepté l'accès au compte puisqu'en tant qu'utilisateur anonyme il n'existe pas de compte.

Si l'on revient au menu principal, il reste une fonctionnalité, la liste des utilisateurs. En cliquant sur le bouton, une fenêtre avec les informations des comptes des utilisateurs apparaît. Cette liste dispose déjà de données ajoutées dans le code.

4 Justification des choix

Au cours de ce projet j'ai pu monter en compétences et notamment dans la prise de décision.

Mon premier choix, est un choix non pas pour le fonctionnement du service mais pour mon apprentissage. En effet, j'ai essayé d'utilisé un maximum de Layout et notamment le GridBagLayout. Celui-ci s'est avéré relativement complexe à comprendre mais très pratique puisqu'il m'a permis d'ajouter diverses contraintes graphiques à mes objets visuels.

Ensuite, j'ai choisi de laisser l'accès à un menu permanent pour l'utilisateur, pour des raisons pratiques, car il me semble plus rapide d'en disposer quand bon me semble plutôt que de devoir effectuer une succession de clics sur des boutons « retour ».

Afin de réaliser mon précédent choix, j'ai été amené a implémenté une architecture graphique relativement complexes à l'aide de Jpanel et de Layout imbriqués les uns dans les autres. Cependant ceci m'a permis un découpage de la fenêtre permanent.

5 Conclusion

Au cours de l'application et l'approfondissement de mes compétences en Java et JavaSwing, j'ai rencontré diverses problèmes.

Tout d'abord, je n'ai jamais vu de modèle MVC, donc j'ai essayé d'implémenté ce que j'en avais compris, tout en gardant à l'esprit tout au long du projet que j'avais peut-être fait une erreur. Malgré mon manque d'expérience j'ai réussi à mener à bout ce projet. Ensuite, comme dit dans la précédente partie, l'architecture graphique m'a beaucoup pris de temps. En effet, un problème que je n'ai pu résoudre est apparu. Le menu gauche de l'utilisateur enregistré ou anonyme, se déplace selon le contenu principal de la fenêtre. De plus, je me suis aperçu que les contenus de type texte n'effectuaient pas de retour à la ligne automatique, donc j'ai choisi d'utiliser le code HTML pour remplacer cette fonctionnalité.

Enfin, j'ai pensé a beaucoup d'évolutions que j'apporterai prochainement à l'application. Tout d'abord, après avoir discuté avec mes camarades j'ai réalisé que mon système pouvait être plus ergonomique en me contentant de recharger le contenu de la fenêtre au lieu d'ouvrir une nouvelle fenêtre. Aussi, comme dit précédemment, je pense modifier mon système, en modifiant le modèle MVC actuellement en place, dans le but de le corriger. De plus, j'ai préparé une évolution qui me semblerait utile, un mode administrateur. Pour finir, je pense implémenter une base de données grâce aux cours de base de données que nous voyons en cours.

Bien entendu, cette application m'a beaucoup apporté, notamment en organisation et en anticipation, puisqu'il faut réfléchir avant de développer au diagramme de classes du système. Permettant ainsi une programmation plus facile et surtout une réduction d'erreur causée par la perte du développeur dans son propre code.

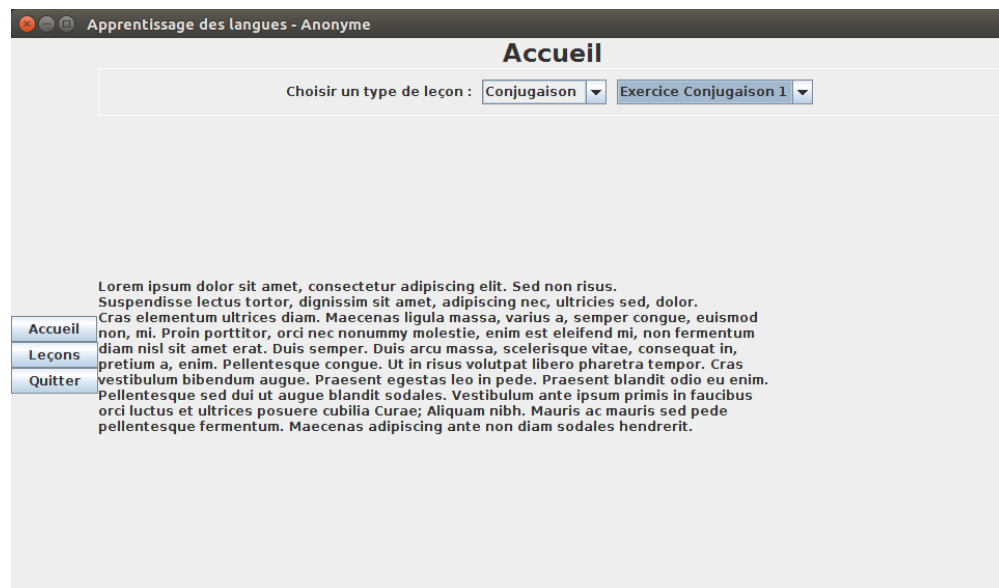
6 Annexes

6 Annexes

Annexe 1 : Vue Accueil en mode anonyme



Annexe 2 : Vue Lecon en mode anonyme



6 Annexes

Annexe 3 : Schéma d'architecture des objets de l'interface en mode anonyme et utilisateur enregistré

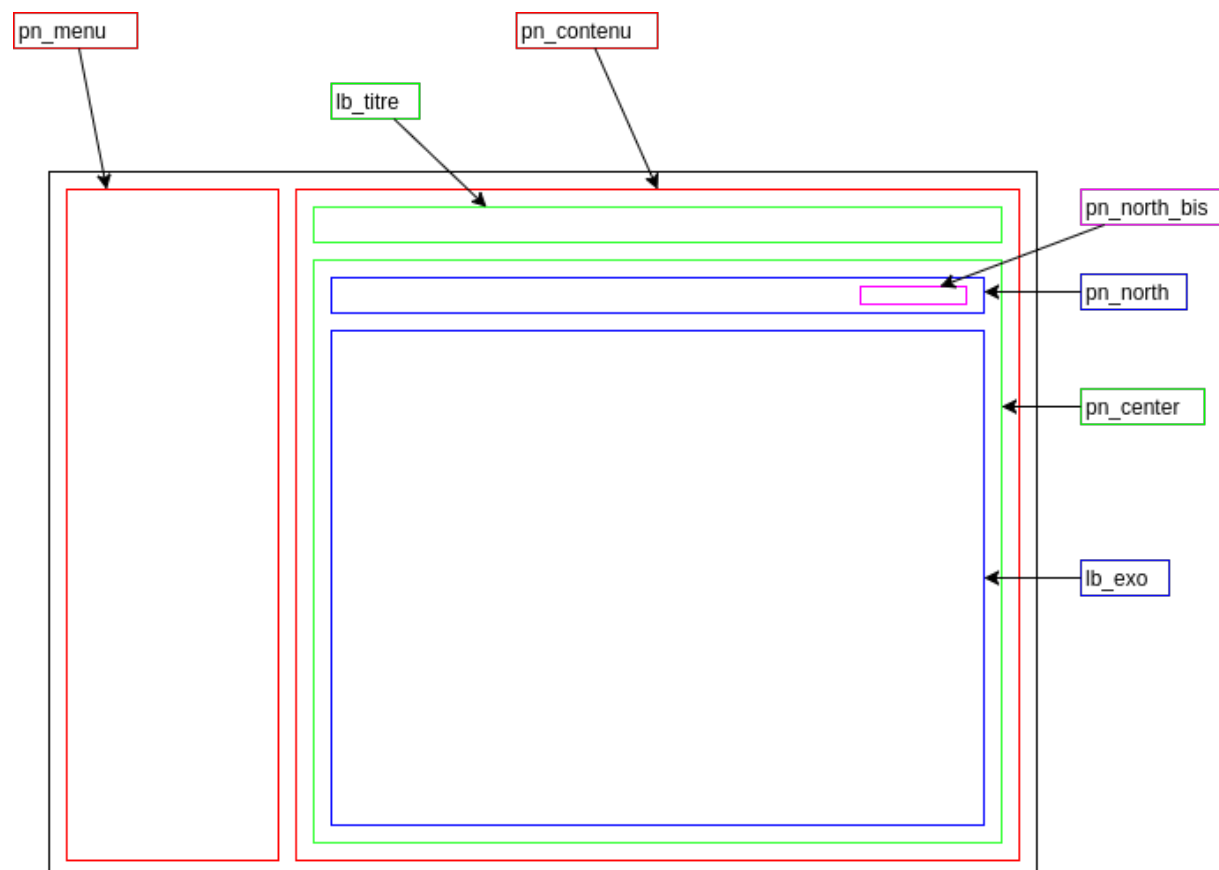


Illustration 9: Schéma d'architecture des objets de l'interface en mode anonyme et enregistré