

Rapport projODT

Réaliser un outil de recherche de mots-clefs dans des fichiers de type
OpenDocument

Présenté le 5 Janvier 2015

devant le jury composé de LEMAIRE Marc et de LIU Tianxiao.

Table des matières

| | |
|--|-----------|
| Introduction | 3 |
| De la conception à la réalisation | 4 |
| Manuel de l'utilisateur | 5 |
| <i>Mode graphique</i> | <i>5</i> |
| Utilisation simple | 5 |
| Changer la racine | 7 |
| Opérateurs logiques | 8 |
| <i>Mode console</i> | <i>9</i> |
| Actions | 10 |
| Options | 11 |
| Conclusion | 13 |

Introduction

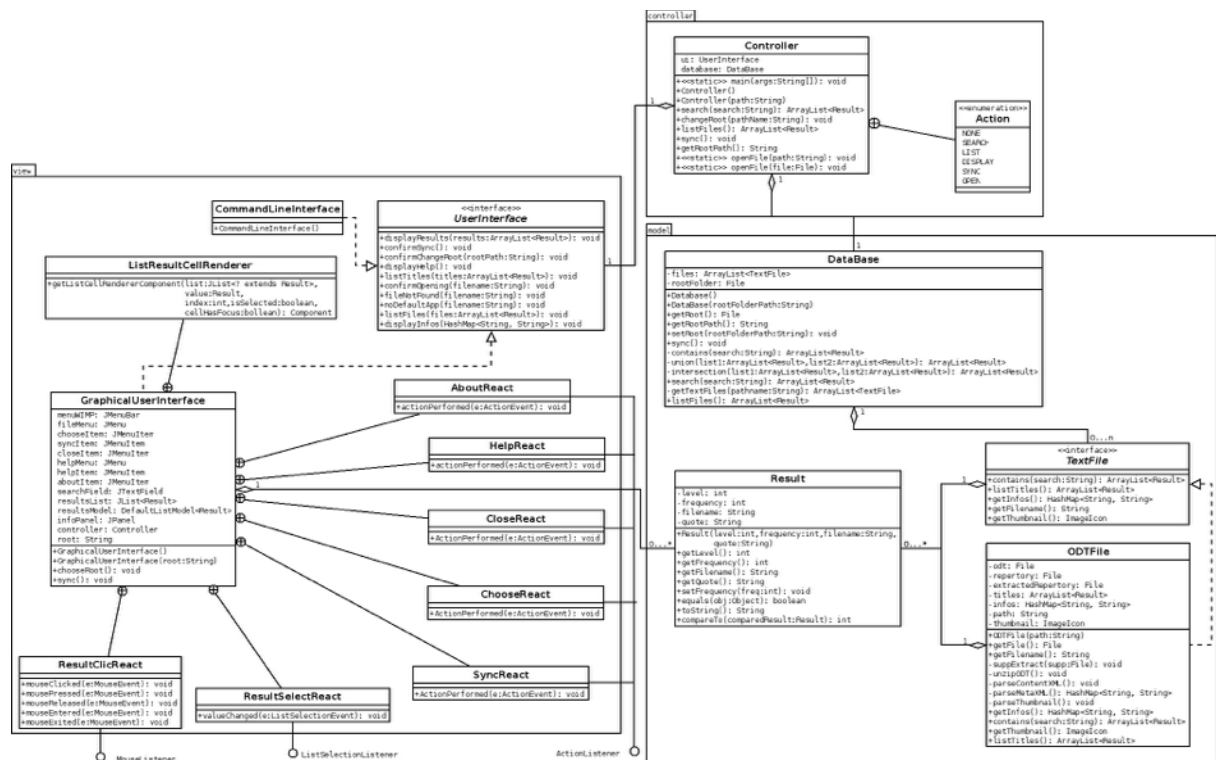
Dans le contexte du module de P.O.O. - Java du cursus Licence 2 MI, nous avons réalisé ce projet sur un ensemble de 7 semaines.

Ce projet consiste à réaliser un outil de recherche performant de mots-clefs présents dans les titres de différents fichiers de type OpenDocument à partir d'un dossier racine.

Le fait de travailler sur ces types de fichiers a amené plutôt naturellement le nom du projet comme étant ProjODT, contraction de « projet » et de « odt ».

L'environnement utilisé afin de mener à bien ce projet est constitué, d'une part du logiciel et IDE Eclipse, nous permettant de programmer et compiler en Java facilement, ainsi que des versions 7 (respectivement 8) de Java pour Vincent (respectivement Louis). Le site GitHub nous a également permis de pouvoir facilement réaliser ce projet de groupe afin de mieux travailler sur ce même projet sans crainte, notamment éviter les éventuels problèmes intervenant lorsque différentes personnes travaillent sur le même fichier.

Pour réaliser ce projet, nous nous sommes séparés les tâches. D'une part, Louis réalisant principalement le packaging « view », et d'autre part, Vincent réalisant le packaging « model ». Cependant, malgré cette séparation des tâches, il en est resté un travail de groupe sur chacune des parties du programme, à savoir que chacun donnait son avis et/ou améliorations sur le travail réalisé par l'autre personne.

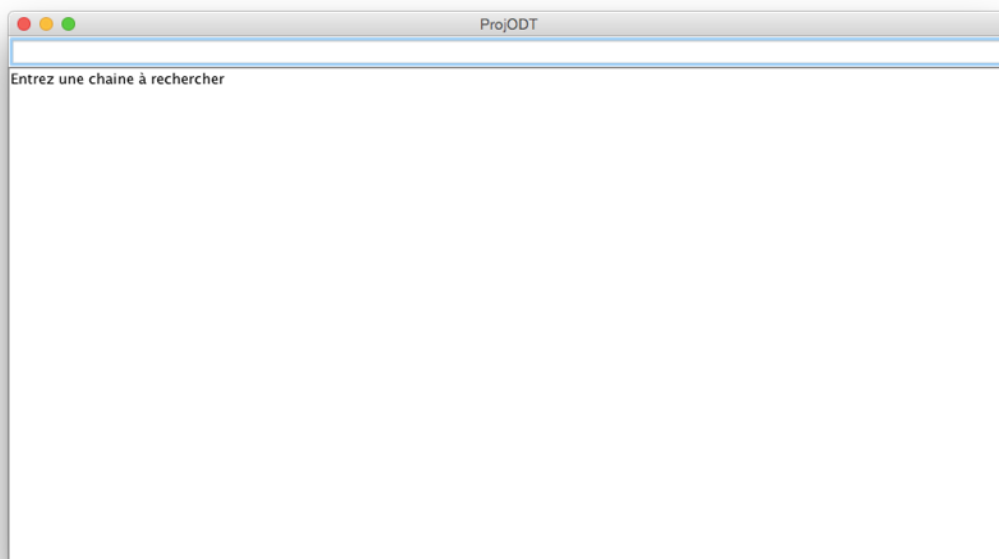


Manuel de l'utilisateur

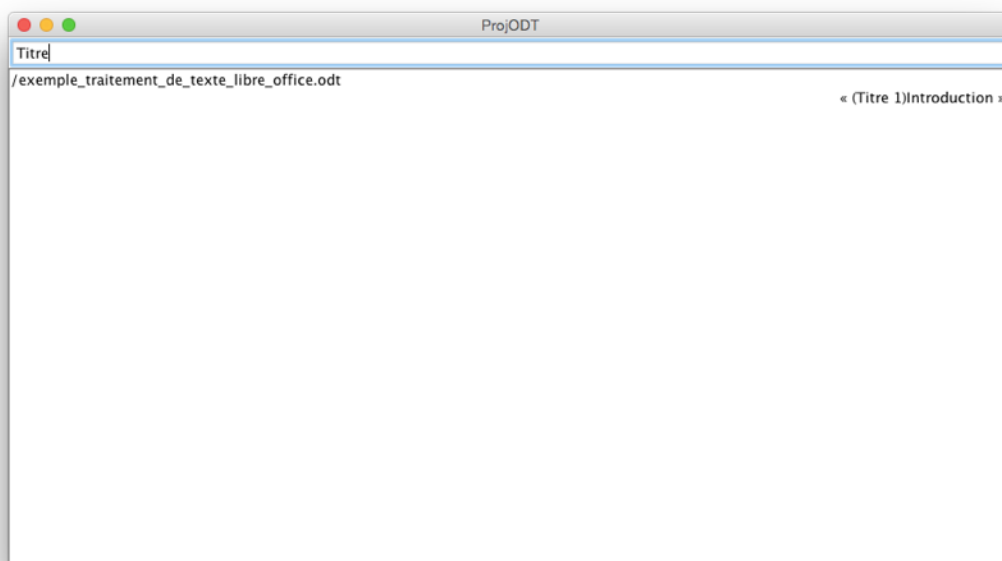
Mode graphique

Utilisation simple

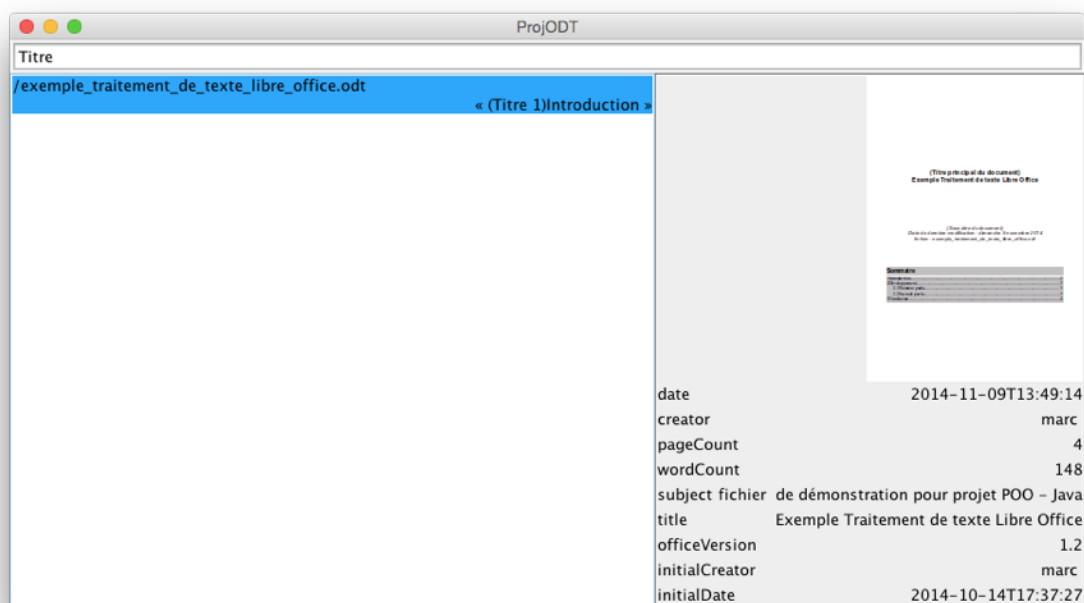
Pour démarrer, utiliser votre méthode habituelle.



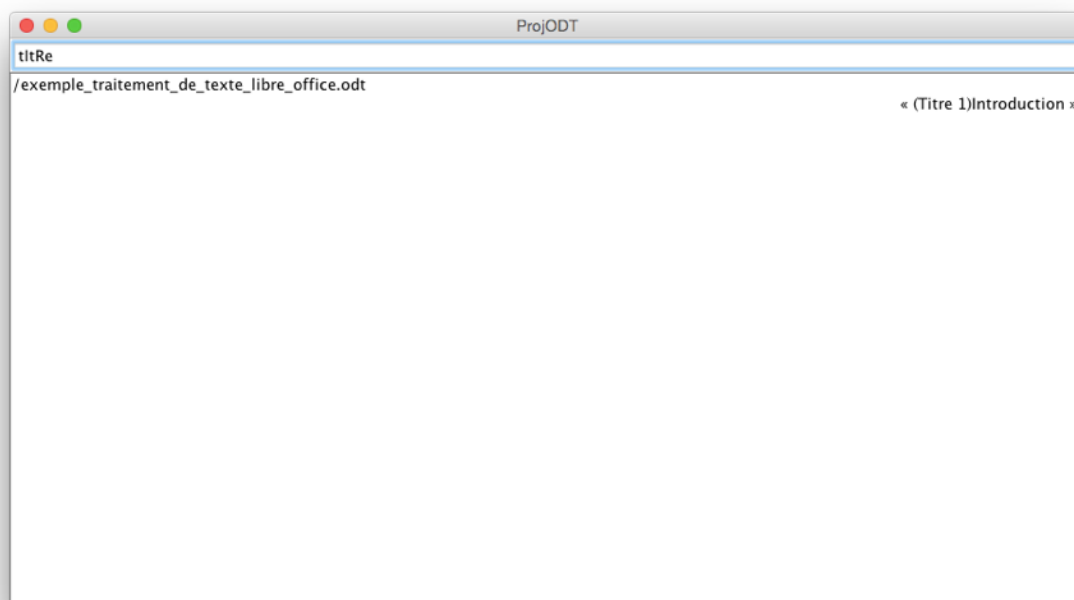
Entrez alors votre recherche dans la barre de recherche en haut de la fenêtre. La recherche s'effectue automatiquement, pas besoin de valider.



À droite se trouve le chemin d'accès du fichier, à gauche le titre dans lequel le résultat a été trouvé. Cliquez sur un résultat pour afficher les détails du fichier ou double-cliquez sur le chemin d'accès pour ouvrir le fichier.



La recherche n'est pas sensible à la casse.

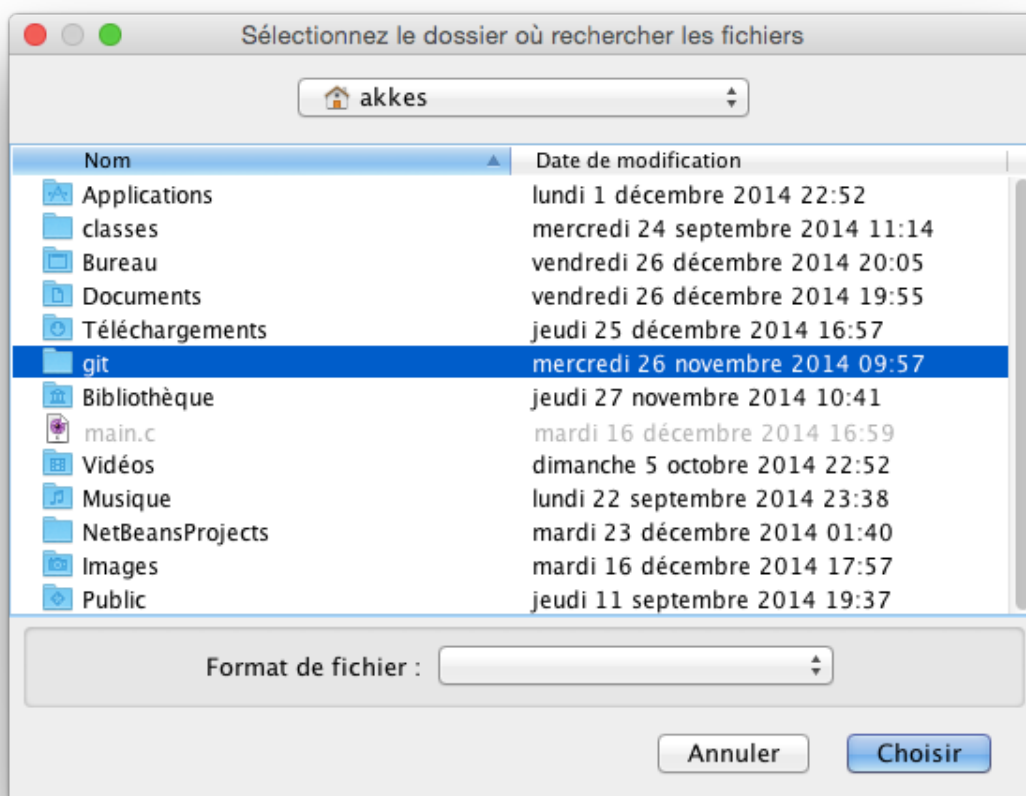


Changer la racine

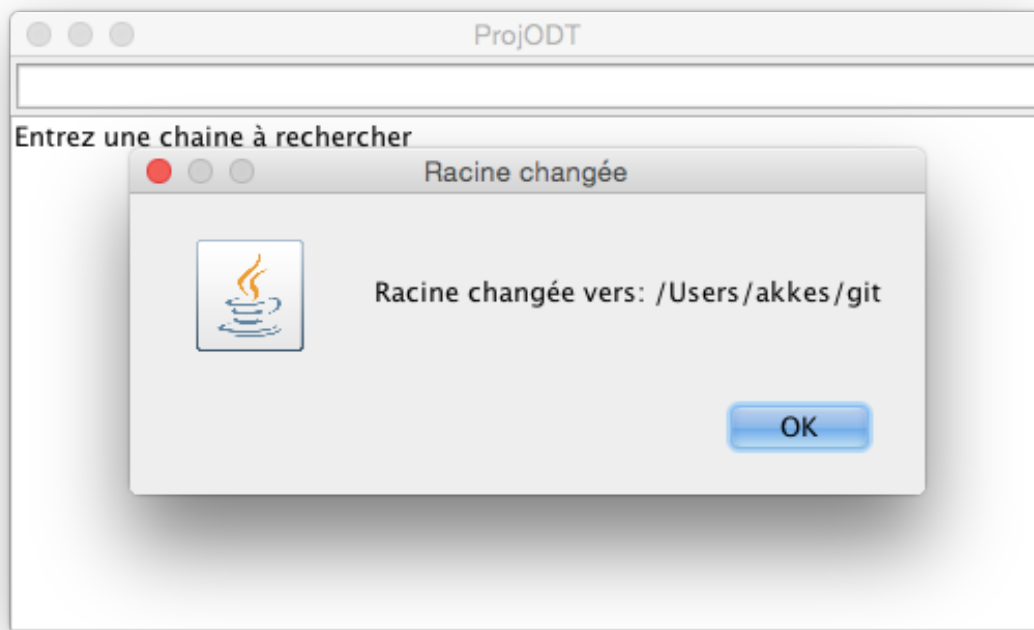
Il est possible de changer le dossier de recherche des fichiers ODT. Pour cela allez dans fichier>Changer la racine.

Sélectionnez le dossier voulu.

| | |
|-------------------|----|
| Changer la racine | ⌘R |
| Synchroniser | ⌘S |
| Fermer | ⌘W |



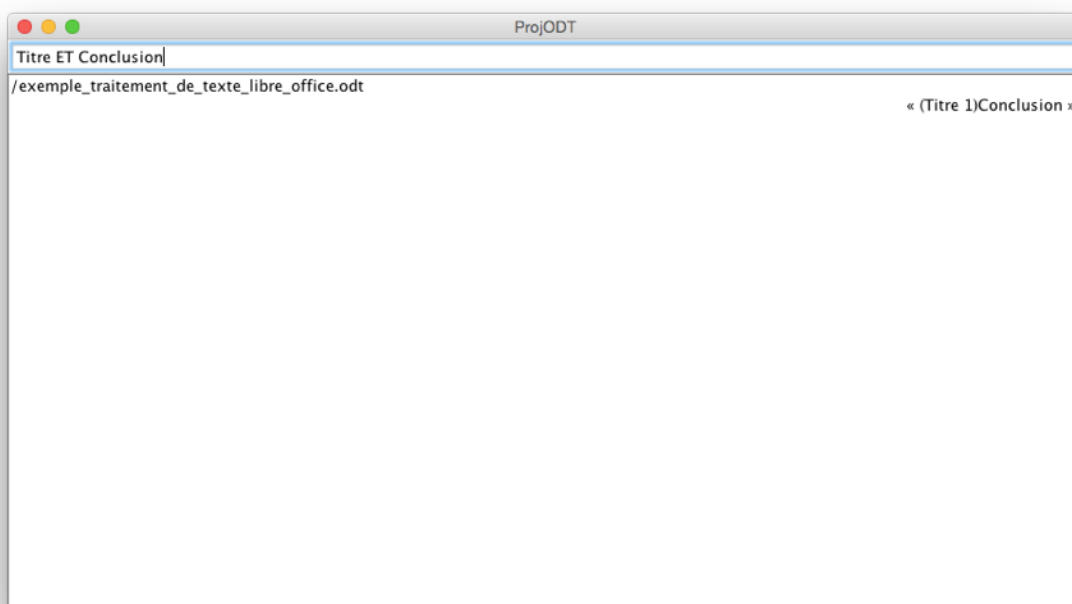
Et appuyez sur Choisir.

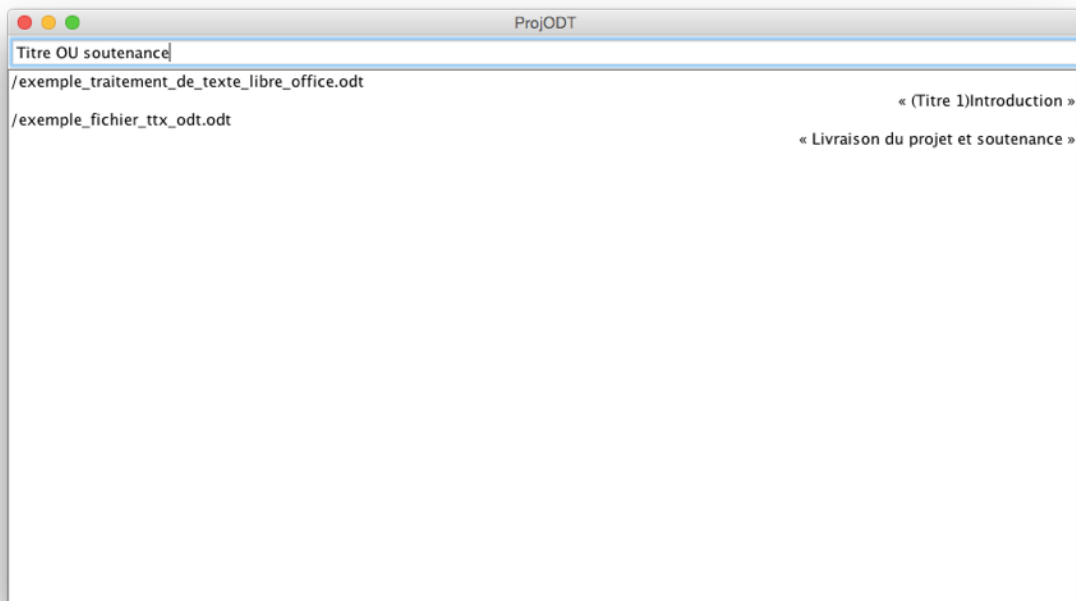


La racine est maintenant changée

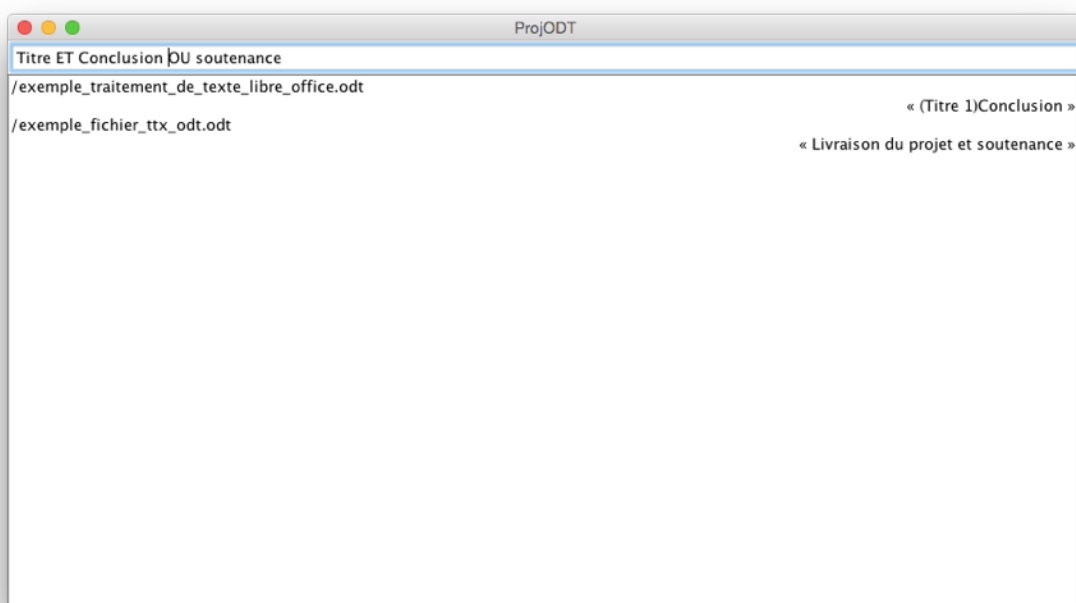
Opérateurs logiques

Vous pouvez également utiliser les opérateurs logiques "ET" et "OU" dans vos recherches.





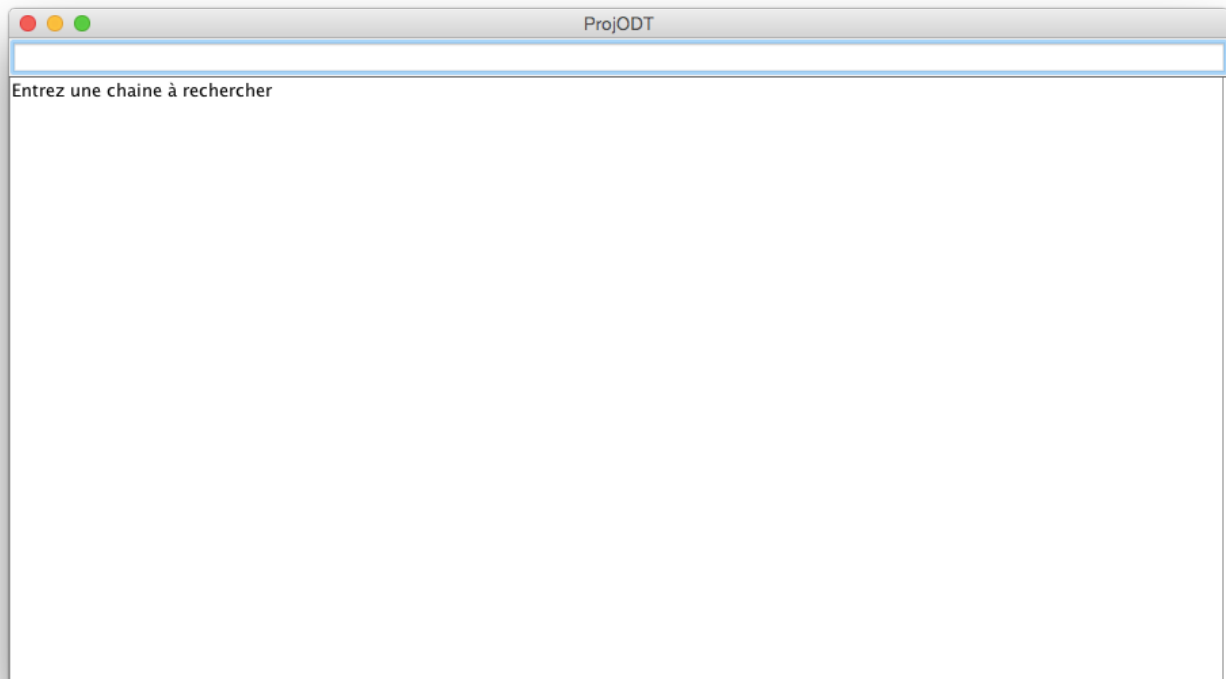
Ou même combiner ces opérateurs. Les ET sont prioritaires sur les OU.



Mode console

Le programme est également utilisable en mode console. Sans action, le programme se lance en mode graphique.

```
tamyris:~$ Java -jar projODT.jar
```



Actions

Différentes actions sont disponibles:

- `search` permet de recherche dans la base dans les titres des différents fichiers.

```
tamyris:~$ Java -jar projODT.jar search Introduction
1 résultat:
(Titre 1)Introduction dans /Users/akkes/git/ProjODT/
exemple_traitement_de_texte_libre_office.odt
tamyris:~$
```

Les opérateurs conditionnels (ET/OU) sont aussi disponibles

- `sync` permet de prendre en compte les changements et les nouveaux fichiers dans la base de donnée.

```
tamyris:~$ Java -jar projODT.jar sync
La base de données à bien été synchronisé.
tamyris:~$
```

- `list` permet de lister les fichiers présents dans la base de données.

```
tamyris:~$ Java -jar projODT.jar list
2 fichiers dans la base de données:
```

```
* /Users/akkes/git/ProjODT/exemple_fichier_ttx_odt.odt
* /Users/akkes/git/ProjODT/
exemple_traitement_de_texte_libre_office.odt
tamyris:~$
```

- open permet d'ouvrir le meilleur résultat dans le traitement de texte par défaut.

```
tamyris:~$ Java -jar projODT.jar open Introduction
Ouverture de /Users/akkes/git/ProjODT/
exemple_traitement_de_texte_libre_office.odt
tamyris:~$
```

- display permet de lister tout les titres d'un fichier ODT.

```
tamyris:~$ Java -jar projODT.jar display /Users/akkes/git/
ProjODT/exemple_traitement_de_texte_libre_office.odt
6 titres dans /Users/akkes/git/ProjODT/
exemple_traitement_de_texte_libre_office.odt:
>Exemple Traitement de texte Libre Office
->(Titre 1)Introduction
->(Titre 1)Developpement
->(Titre 2)Premiere partie
->(Titre 2)Seconde partie
->(Titre 1)Conclusion
tamyris:~$
```

Options

À ces actions s'ajoutent deux options:

- -h, --help permet d'obtenir de l'aide sur le programme

```
tamyris:~$ Java -jar projODT.jar --help
ProjODT - v1.0 - (c) Vincent Monot & Louis Desportes
Utilitaire de recherche dans une base d'ODT
```

Commandes:

display: affiche tous les titres présents dans un fichier

search: recherche le/les termes donnés dans la base de données. Peut être utilisé avec ET/OU pour affiner les recherches (priorité au ET)

open: ouvre le fichier adéquat

sync: synchronise la base de données

`list:` liste les fichiers présents dans la base de données

Options

`-d, --database` : définit la racine de la base de données à utiliser

`-h, --help` : affiche l'aide

tamyris:~\$

- `-d, --database` permet de spécifier un répertoire de travail différent de l'actuel

tamyris:~\$ `Java -jar projODT.jar search Introduction -d git`

1 résultat:

(Titre 1) `Introduction` dans `/Users/akkes/git/ProjODT/exemple_traitement_de_texte_libre_office.odt`

tamyris:~\$

Conclusion

Les principales difficultés rencontrées ont été de s'adapter à la manière de programmer de l'autre personne. En effet, on ne travaille pas tous de la même manière et cela rend les choses plus difficiles.

Une deuxième grosse difficulté a été la coordination des fichiers du programme. En effet, malgré la grande aide procurée par git, il est toujours difficile de pouvoir parfaitement se coordonner quant aux changements réalisés sur les différents fichiers par différentes personnes.

