



Rapport Atelier de Gestion de Projet

Janvier 2018

Hervé-Madelein Attolou

Loïc Bachelot

David de Castilla

Professeurs référents : Tianxiao LIU & Dan VODISLAV

Introduction	2
Description du sujet	2
Organisation du projet	2
Résultats	2
Conclusion	2
Remerciements	2
Annexe	2

Introduction

Description du sujet

Le projet s'appuie sur le code du TP 4¹.

Celui-ci est une simulation d'une agence de banque. Dans cette banque, les clients font la queue et effectuent des opérations.

Le but de cette simulation est d'analyser le temps d'attente et le temps de réponse dans cette agence.

Notre version ajoute quelques fonctionnalités:

- Les clients sont maintenant associés à un compte sur lequel ils doivent effectuer son opération.
- Les données des comptes et des statistiques sont sauvegardées dans une base de données avec Hibernate.
- Les statistiques à afficher sont maintenant plus nombreuses :
 - Clients servis vs clients non servis
 - Occupation des caissiers en fonction du temps
 - Répartition des types d'opérations
 - Soldes des comptes à la fin de la simulation
 - Nombre d'opération par compte
 - Taille de la file d'attente en fonction du temps.

Organisation du projet

Organisation Technique

Le code que nous avons produit est stocké sur un serveur de versionnage GIT sur le site GitLab². Cet outil nous permet également de mettre en place un système d'intégration continue (CI) associé à Gradle³.

Afin de communiquer de manière instantanée, nous avons décidé d'utiliser l'outil Slack⁴ qui, couplé à des plugins, nous permet d'être informé des activités sur le serveur GIT (notamment les erreurs de test et de compilation et les push de chacun des membres).

¹ <https://depinfo.u-cergy.fr/~tliu/ens/coo/coo-td4-spring.pdf>

² <https://gitlab.com/daviddecastilla/AGP>

³ <https://gradle.org/>

⁴ <https://slack.com/>

La répartition des tâches est également présente sur GitLab, dans les issues. Ceci nous permet d'explicitier le travail de chacun et de séparer les nombreuses fonctionnalités en tâches unitaires hiérarchisées.

Notre environnement de travail se complète par l'utilisation de l'IDE IntelliJ de JetBrains⁵ et de Java 8 entouré de nombreux frameworks.

La qualité du code est vérifiée par SonarLint⁶ et contribue à produire un projet qui satisfait les standards de la programmation orientée objet .

Conception

Ajout du framework Hibernate

Nous avons commencé par ajouter le framework Hibernate au projet pour remplacer JDBC. Pour cela, nous avons dû annoter les différentes classes concernées par la persistance des données. Après avoir annoté les différentes classes, il a fallu configurer hibernate, et notamment lui communiquer classes concernées par les annotations. Cette configuration a permis de créer le schéma d'exportation pour permettre à hibernate de sauvegarder les différentes informations dans la base de données.

À la fin de la simulation, une méthode est appelée pour récupérer les différentes informations à stocker, et les enregistrer dans la base de données.

Le programme peut ensuite accéder aux informations, notamment pour afficher les différents graphique à la fin de la simulation.

Gestion des comptes en banques

Nous avons ensuite créé la classe BankAccount et implémenter les modifications qu'elle a engendrée dans le programme :

- Association des nouveaux clients avec un compte :
 - Création du compte si le numéro de compte ne correspond pas à un compte existant.
 - Association si le numéro de compte correspond à un compte existant.
- Mise en place d'un solde de compte aléatoire.
- processOperation dans AbstractOperation et implémentation dans les classes filles, méthode qui permet la gestion du solde.
- Sauvegarde du compte dans la base de données

⁵ <https://www.jetbrains.com/idea/>

⁶ <https://www.sonarlint.org/>

Logs AOP avec Spring

En partant du système de log initial de journalisation déjà présent dans le TP 4, nous avons ajouté une fonction de rappel lorsqu'un client est servi.

L'opération traitée se retrouve alors dans le fichier de logs sous l'exemple suivant:

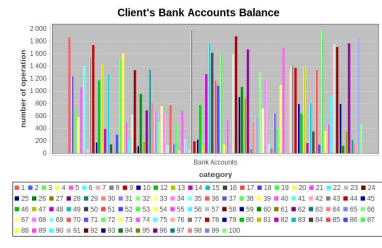
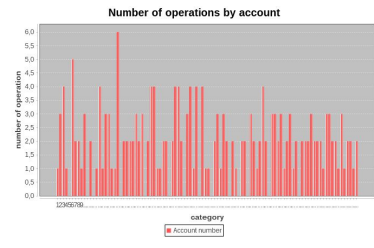
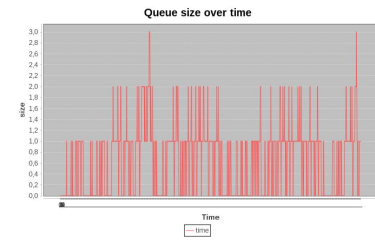
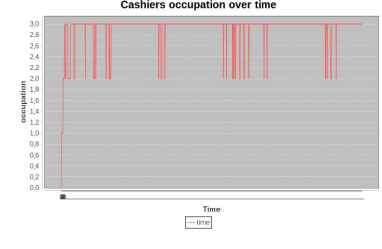
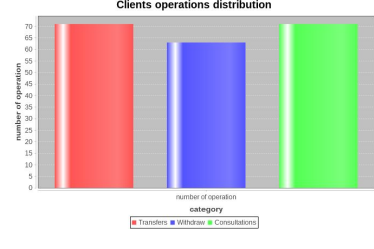
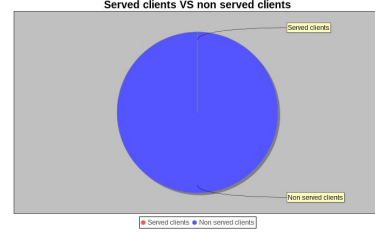
```
-----  
Dear boss, today, there is another client who made an operation  
Operation : Withdraw  
-----
```

Statistiques et graphiques

Nous avons fini par ajouter les différents graphiques de statistiques. Pour cela nous avons:

- créer les requêtes nécessaires sur la base de données pour la création des dataset.
- Les données sont ensuite formatées dans un dataset
- Le dataset est ensuite affiché sous forme de graphique.

Résultats

		
Solde des comptes à la fin de la simulation	Nombre d'opération par compte	Taille de la file d'attente en fonction du temps.
		
Occupation des caissiers en fonction du temps	Répartition des types d'opérations	Clients servis vs clients non servis

Conclusion

Dans notre projet, les points fort qui se sont illustrés sont d'abord notre organisation assez précise. Après plusieurs années maintenant à travailler ensemble et ayant mené à bien de nombreux projets, professionnels et scolaires, nous connaissons l'organisation à mettre en place pour optimiser nos efforts.

De plus, dans ce projet les compétences nécessaires était déjà maîtrisées en cours et ne demandaient pas forcément un travail complémentaire de recherche.

Enfin, les conditions exceptionnelles de collocation et de vie dans un même espace, nous ont permis de travailler dans un environnement connu de tous les membres de l'équipe pour produire un projet de qualité.

Remerciements

Nous tenons à remercier Monsieur Tianxiao LIU qui nous a permis de réaliser ce projet malgré des conditions exceptionnelles.