

Projet MF Banking

Kévin POGORZELSKI, François GUILLAIN, Tahar BAKIR
Bastien LEMALE, Guillaume VAROQUAUX et Pierre DAL-PRA

15 juin 2012

Plan

- Présentation des fonctionnalités implémentées (user stories)
- Organisation du projet
 - Management du travail
 - Environnement de développement
- Architecture du projet
 - Objets du modèle
 - Organisation des modules
- Présentation technique
 - Back-office
 - Front-office
 - Tests (unitaires et d'intégration)
 - Web Services

Présentation des fonctionnalités implémentées (user stories)

Sprint 0+1

- Mise en place de l'environnement de développement
- Consulter sur ma page d'accueil le solde de mes comptes espèce
- Accéder à ma page de login
- Me logger
- BONUS : Page admin basique

Sprint 2

- Consulter le détail d'un compte espèce avec le détail des opérations d'un mois donné + détail des opérations carte à part
- Consulter le détail des opérations carte pour un mois donné

Sprint 3

- Réaliser des virements internes
- Réaliser des virements externes
- Exporter au format Excel le relevé des opérations pour un mois donné
- Consulter l'historique de mes virements
- Consulter sur ma page d'accueil l'encours carte sur chacun de mes comptes espèce
- Consulter sur ma page d'accueil le solde prévisionnel de chacun de mes comptes espèce

Sprint 4

- Ajout de webservices REST et SOAP
- BONUS : Amélioration de la page admin
- BONUS : Mise à jour automatique du solde, du solde prévisionnel et de l'encours carte
- BONUS : Développement de deux webapps et d'une application Android pour tester les webservices

Organisation du projet

Management du travail

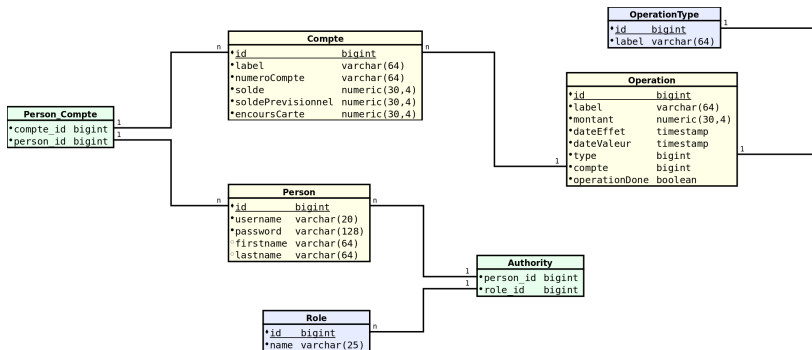
- Gestion du projet selon la méthode Scrum
- Part importante du projet réalisée en Pair Programming
- Polyvalence des développeurs
- Mise en place de conventions
- Définition de “Fini Fini” :
 - Développé
 - Testé (en local et sur le serveur d'intégration)
 - Refactoré
 - Revu par un pair
 - Documenté (Javadoc)

Environnement de développement

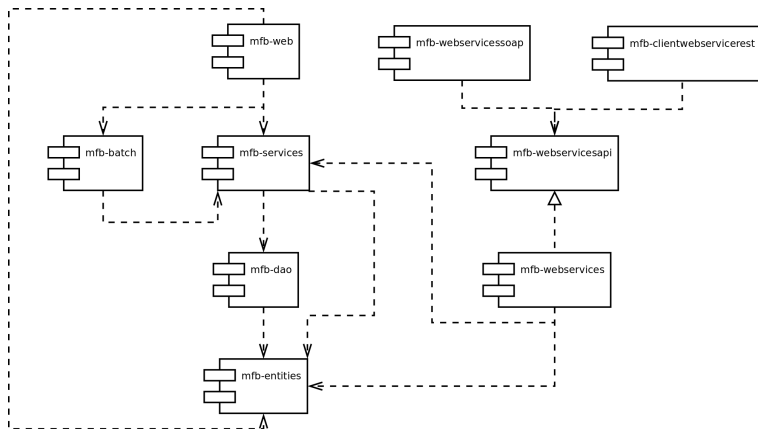
- Build : Maven
- Intégration continue : Jenkins → Build incassable
- SCM : Git
- Serveur : Tomcat
- Bases de données : PostgreSQL, HSQLDB

Architecture du projet

Diagramme E/A



Organisation des modules Maven du projet



Présentation technique

Back-office

- Spring (IOC, DB, TX, MVC, Validation. . .) & Spring Security
- Persistence
 - ORM : Hibernate + JPA
 - Versionnement des schémas : Liquibase
 - Pool de connexion : Tomcat JDBC
 - Cache de 2nd niveau : Ehcache
- Spring Batch
- Logging : SLF4J & Logback

Front-office

- Internationalisation (Spring)
- Apache Tiles
- Twitter Bootstrap
- Apache POI

Tests

- Tests unitaires :
 - Junit
 - DBUnit (via spring-db-unit)
 - Mockito
- Tests d'intégration → Selenium :
 - Intégration à Maven
 - Déploiement sur un serveur Jetty
 - Utilise une base PostgreSQL de test (remplie/vidée à chaque test), suite à des problèmes avec HSQLDB

Web Services

- Apache CXF
 - SOAP
 - REST (JSON, via Jackson)
- Conversion entités → DTO avec jDTO

Merci de votre attention !

Des questions ?