

PLAN DE TEST

Projet Java EE

Nicolas Léotier



(Aix*Marseille
université

TABLE DES MATIERES

- ⊙ **Introduction**
 - Présentation de l'application
 - Contexte de réalisation
 - Choix technologiques
- ⊙ **Projet à tester**
 - Liste des modules
 - Date de livraison
- ⊙ **Documents joints**
 - Documents généraux
 - Documents techniques
 - Documentation
- ⊙ **Environnement de test**
 - Sites de test
 - Configurations matérielles
 - Outils de test
 - Bases de test / Données de test
- ⊙ **Tests à réaliser**
 - Liste des modules à tester
 - Exigences

- ❖ Présentation de l'application
- ❖ Contexte de réalisation
- ❖ Choix technologiques
- ❖ Coordonnées du responsable des tests

INTRODUCTION

Présentation de l'application

L'objectif est de créer un gestionnaire d'annuaire pouvant enregistrer une personne et lui attribuer un groupe.

- Une personne est représentée par un ensemble d'informations : identifiant, nom, prénom, adresse électronique, site WEB, date de naissance et mot de passe.
- Chaque personne est placée dans un groupe. Un groupe est composé de quelques dizaines de personnes (par exemple les étudiants du M2 FSIL 2015/2016). Un groupe a donc un nom et un identifiant.
- L'application doit présenter une liste de groupes, une liste de personnes de chaque groupe et une vue détaillée de chaque personne (sauf adresse électronique et date de naissance). Une fonction de recherche doit être offerte.
- L'application doit être fonctionnelle si nous avons plusieurs milliers de personnes et plusieurs centaines de groupes.
- L'application doit permettre à chaque personne de modifier sa description.
- Les personnes présentes dans l'annuaire peuvent avoir accès à toutes les informations.

Contexte de réalisation

Le travail est effectué durant l'année scolaire 2016-2017 en 2^e année de Master à la faculté des sciences de Luminy. Il participe à la validation de la matière « Java EE » enseigné par Monsieur Jean-Luc Massat. L'IDE Eclipse est utilisé pour faciliter le développement.

Choix technologiques

- La spécification **JEE** pour la technologie **Java 8** d'Oracle.
- **JDBC** comme interface de programmation afin de gérer la persistance des JavaBeans.
- Le framework libre **Spring 1.4.x** pour construire et définir l'infrastructure.
- Le SGBDR **MySQL 5.x**.
- **Maven 3.x** pour gérer les dépendances du projet.

- ❖ Liste des modules
- ❖ Date de livraison

PROJET A TESTER

Liste des modules

- Le module **dao** regroupe les classes permettant de gérer les accès à la BDD :
 - **GroupPersonDao** est une interface qui définit les méthodes élémentaires afin de créer, modifier, détruire et accéder aux données.
 - **JdbcGroupPersonDao** implante l'interface GroupPersonDao et utilise le module JdbcTemplate de Spring.
- Le module **models** regroupe les POJOs de l'application :
 - **Group**
 - **Person**

Date de livraison

Le projet est composé de deux livrables :

- Le lundi **21 novembre 2016** le stockage (DAO/JDBC) doit être fonctionnel :
 - Conception et création de la base de données.
 - Conception et création des JavaBeans permettant de représenter les données de l'annuaire.
 - Utilisation de la technologie JDBC pour gérer la persistance des JavaBeans.
- Le lundi **12 décembre 2016** la couche métier (Spring) et la couche présentation (JSP/Spring MVC) devront être implémentés :
 - Prendre en charge l'accès et la modification des données métier.
 - Mettre en place une application WEB basée sur la technologie Spring MVC/JSP/Servlet.

- ❖ Documents généraux
- ❖ Documents techniques
- ❖ Documentation

DOCUMENTS JOINTS

Documents généraux

- Un rapport (fichier **rapport.pdf**).

Documents techniques

- Un cahier des charges (fichier **CdC.pdf**).
- Les sources de l'application WEB (répertoire **web**).
- Les sources Java (répertoire **java**).
- Les documents du plan de test (répertoire **tests**).
 - Politique de test : ressources, outils de tests, phases de tests.
 - Plan de tests unitaires.
 - Plan de tests d'intégration.

Documentation

- La javadoc des sources Java (répertoire **javadocs**).
- Un wiki de l'utilisation de l'application pourra être fournis.

- ❖ Sites de test
- ❖ Configurations matérielles
- ❖ Outils de test
- ❖ Bases de test / Données de test

ENVIRONNEMENT DE TEST

Sites de test

Le site sera hébergé en local, lors du développement la plateforme de développement **WAMP 3.0** a été utilisé afin de créer un serveur hébergeant **MySQL 5.7**. **Apache Tomcat 8.0** sera utilisé par la suite pour afficher les pages web.

Configurations matérielles

Afin de pouvoir utiliser l'application, il faut que le système soit compatible Java 8 :

Windows

- ⊙ Windows 10 (8u51 et versions ultérieures)
- ⊙ Windows 8.x (version de bureau)
- ⊙ Windows 7 SP1
- ⊙ Windows Server 2008 R2 SP1 (64 bits)
- ⊙ Windows Server 2012 et 2012 R2 (64 bits)
- ⊙ RAM : 128 Mo
- ⊙ Espace disque : 124 Mo pour JRE ; 2 Mo pour Java Update
- ⊙ Processeur : processeur Pentium 2 266 MHz minimum
- ⊙ Navigateurs Web : Internet Explorer 9 et versions ultérieures, Firefox

Mac OS X

- ⊙ Ordinateur Mac Intel exécutant la version Mac OS X 10.8.3+, 10.9+
- ⊙ Privilèges administrateur pour l'installation
- ⊙ Navigateur 64 bits
- ⊙ Un navigateur 64 bits (Safari ou Firefox par exemple) est requis pour exécuter Oracle Java sur Mac OS X.

Linux

- ⊙ Ubuntu Linux 12.04 LTS, 13.x
- ⊙ Ubuntu Linux 14.x (8u25 et versions ultérieures)
- ⊙ Ubuntu Linux 15.04 (8u45 et versions ultérieures)
- ⊙ Ubuntu Linux 15.10 (8u65 et versions ultérieures)
- ⊙ Navigateurs Web : Firefox

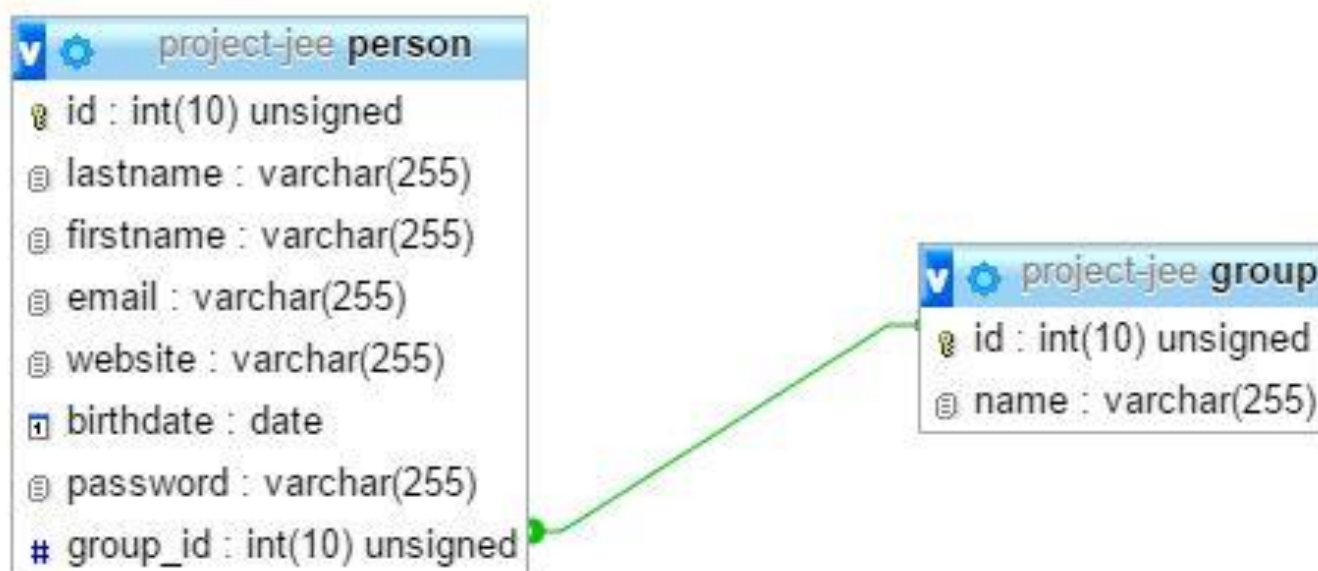
Outils de test

- Le framework **Junit 4.0** pour les tests unitaires.
- **Emma 0.6** pour la couverture de code. (**EclEmma 2.3** comme plugin Eclipse)
- **FindBugs 3.0** pour l'analyse statique du code.

Bases de test / Données de test

Il est possible d'importer le squelette de la BDD en utilisant le fichier **setup.sql** disponible à la racine du projet.

Diagramme UML de la BDD :



Contenu de la table **group** pour les tests :

Id	Name
1	M2 Informatique Luminy
2	M1 Biologie Saint-Jérôme

Contenu de la table **person** pour les tests :

Id	Firstname	Lastname	Email	Website	Birthdate	Password	Group_id
1	Léotier	Nicolas	nicolas@leotier.fr	Leotier.fr	1992-02-02	Pass	1
2	Gades	Paul	Paul.gades@gmail.com	NULL	NULL	Pass	2

- ❖ Liste des modules à tester
- ❖ Exigences

TESTS À RÉALISER

Liste des modules à tester

Le module **dao** est seul actuellement testable, les POJOs du module **models** ne seront pas testés de par leur nature simpliste, de plus ils sont utilisés dans le dao, ce qui nous permet d'assurer leur intégrité.

Exigences

- Prévoir une classe de test unitaire **Junit** pour valider les méthodes offertes par l'implantation.
- Les composants (DAO, DataSource, tests JUnit) doivent absolument être connectés et paramétrés par une couche **Spring**.
- Prévoir une phase de préparation de la base de données pour insérer des données cohérentes.