ShiftPlanning
Paquetage
Version 1.0

Diagramme de classes package 1

1. Introduction

Ce document permet d'avoir une vue sur l'architecture du système en se basant sur différents aspects du système. Cette architecture influencera bien évidemment la conception. Donc toute modification sera répercutée sur les cas d'utilisation de l'application.

2. Objectif du logiciel

2.1 Contexte :

Durant la réalisation de ce projet notre équipe est appelée à : Mettre en œuvre un développement orienté objet en appliquant l'approche UP qui est guidé par les UCs. Exploiter les Capacités UML Réutiliser les patrons de conception

- Planification et suivi de l'avancement des activités d'un projet donné.
- L'affectation des ressources humaines et matérielles.
- Communication des données.
- Mise en œuvre d'une assistance au planning des séances.
- Consultation du planning.
- Impression du planning.
- La validation des activités propres au projet et dont les modifications doivent se faire par le superviseur du projet.
- Proposition et identification des fonctions utiles au contexte international
- Choix de référentiel (commun, personnel)
- Adaptation d'une sauvegarde hebdomadaire et automatique.
- Fonctionnement sous différents navigateurs et systèmes d'exploitation
- Réalisation d'une Interface simple et ergonomique.
- Mise à jour des activités lors d'une connexion.
- Fonctionnement en mode non connecté.
- Connexion via une messagerie. i18nProject

2.2 Besoins non fonctionnels :

L'architecture de l'application devrait permettre le fonctionnement en mode non connecté. La connexion à une messagerie doit être assurée par une interface simple et ergonomique

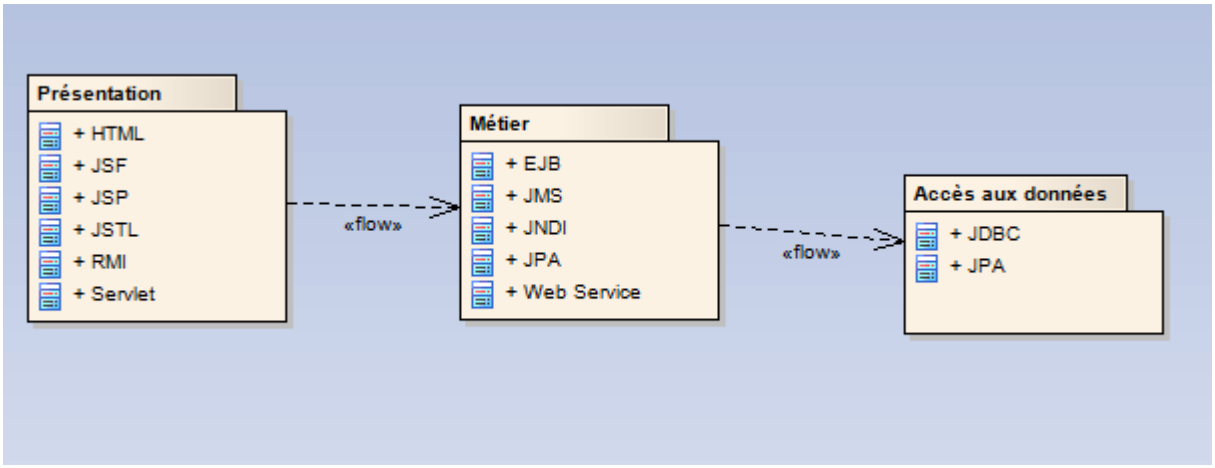
3. structure

3.1 Vue des couches :

L'architecture en trois tiers est une architecture de programmation qui propose de séparer une application en trois parties :

- La présentation qui s'occupe de l'affichage sur le poste de travail des données du système et interaction avec l'utilisateur
- Les traitements métier ensemble de règles métiers de l'application
- La couche accès de données qui manipule et conserve les données.

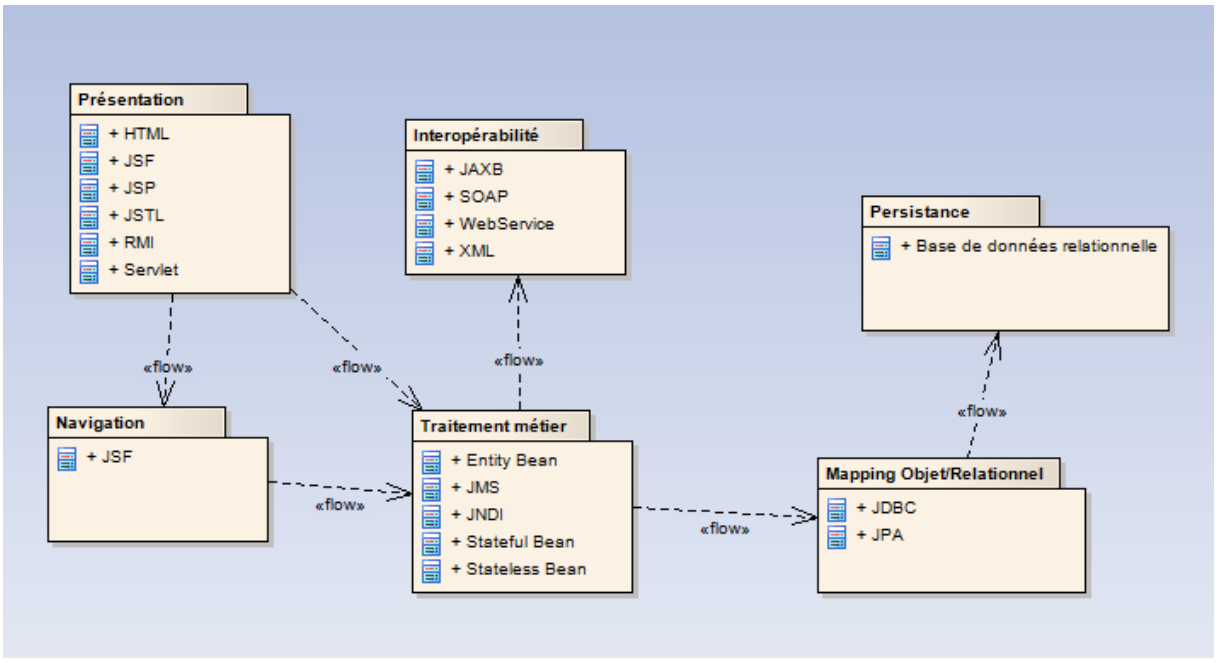
Ci-après un diagramme de paquetage UML représentant ces trois couches. Chaque sous-système contient les technologies JAVA EE qui seront utilisées dans l'application



3.1 Architecture applicative :

Le précèdent modèle en trois couches peut être affiné pour être plus fidèle a l'architecture finale de l'application.

Ci-après un diagramme de paquetage décrivant les couches de l'application



Couche traitement métier

En pratique on trouve au niveau de la couche métier

Des entités dont la persistance est assurée par la couche de mapping

Des stateless Beans qui proposent des méthodes pour manipuler les entités (crud)

Des message-driven beans qui assurent les traitements asynchrones

Les API JNDI pour accéder au service de nommage et javaMail pour envoyer des emails pour le consultant externe

Couche de mapping objet/relationnel

Elle transforme la représentation physique des données en une représentation objet et inversement Ce

mécanisme sera assuré par JPA qui utilise le protocole JDBC pour exécuter les appels SQL

Couche d'interopérabilité

Pour dialoguer avec les partenaires

4. Mode non connecté

Pour répondre aux besoins de l'usabilité de l'application en mode non connecté plusieurs solutions existent

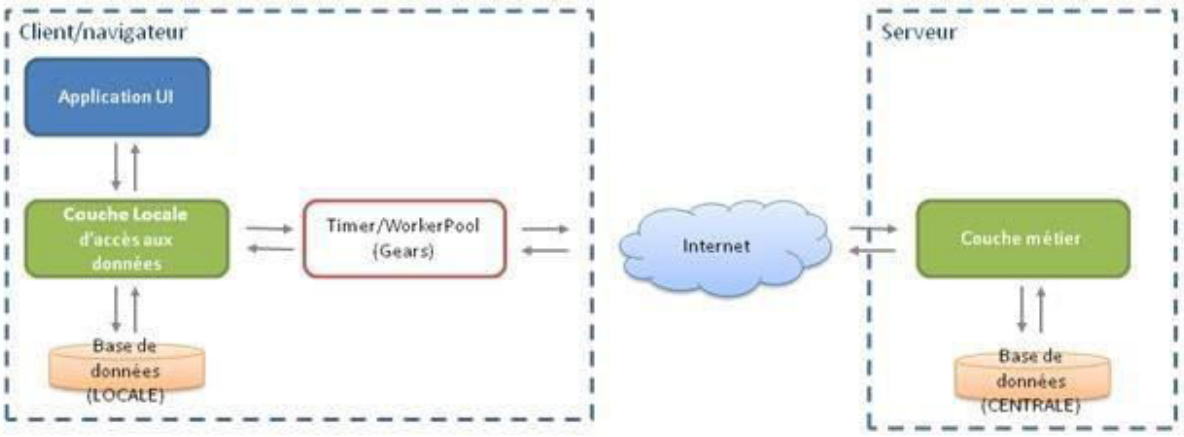
A ce jour on trouve 3 initiatives qui facilitent la mise en place du mode déconnecté sur des applications web :

DOJO le Framework DOJO fournit des extensions dédiées, AIR, et GEARS qu'on va utiliser et qui correspond à la réponse de Google a la problématique du déconnecté

GEARS s'installe sous forme de plugin disponible pour les navigateur courants ce plugin va mettre a notre disposition deux éléments principaux :

-Un mini-serveur http qui portera techniquement le nom de local server ce composant s'occupe de mettre en cache l'ensemble des ressources et de les servir une fois déconnecté

-Une base de données SQL Lite qui va stocker les données



Grace à ce mode de fonctionnement l'application sera capable de travailler systématiquement en mode déconnecté. Des mécanismes (timer et workerPool Gear) sont mis en place pour synchroniser (lorsque la connexion internet le permet) les données en local et les données en central