Rapport projet mdoc

Création d’une application WEB d’évaluation de produits culturels

SOMMAIRE

I. Le projet 1

II. Organisation du projet 1

A. Organisation du travail 1

B. Présentation de l’équipe 1

C. Choix des outils de développement 1

III. Analyse des besoins et spécifications du projet 2

A. Besoins 2

B. Fonctionnalités 2

C. Spécifications techniques : technologies retenues 2

D. Architecture de l’application 2

IV. Réalisation du projet web 3

A. Environnement matériel 3

B. Environnement logiciel 3

C. Modèle de données de l’application 3

D. Initialisation du projet 3

E. Nos extensions 3

# Le projet

La projet consiste à réaliser une application web d’évaluations de produits culturels (albums de musique, films ou DVDs, jeux vidéo, livres) qui propose à ses utilisateurs de donner leur avis (par des notes et des commentaires) sur de tels produits, récents ou non.

D’après l’énoncé on peut déjà distinguer des « entités » : Utilisateur, Produits culturel, Catégorie de produit, Commentaire. Nous allons voir les différents outils et technologies utilisés pour la réalisation de l’application. Le détail de leur utilisation sera décrit dans le cœur du projet.

# Organisation du projet

## Organisation du travail

## Présentation de l’équipe

## Anas AOUAD

Système d’exploitation : Linux

## Mamadou BAH Sarifou

Système d’exploitation : Mac OSX Mavericks 10.9

## Shazad KHAN

Système d’exploitation : Mac OSX Mavericks 10.9

## Choix des outils de développement

# Analyse des besoins et spécifications du projet

## Besoins

## Fonctionnalités

## Spécifications techniques : technologies retenues

= JEE/AppEngine/HTML 5/ JSP / CSS3 / JS / JQUERY / RestFull

* Eclipse JEE :

Environnement de développement optimisé pour les applications web developpés en JEE.

* Google AppEngine Server 1.9.5 :

Plateforme d’hébergement d’applications web basé sur les serveurs de Google, supportant le langage Java.

* SVN + schéma svn
* MAVEN
* API Restfull

## Architecture de l’application

L’architecture JEE utilise le pattern reconnu MVC qui sépare l’application en 3 couches distinctes :

* *Le Modèle* qui contient la logique de l’application, il peut accéder à une base de données pour la persistance.
* *La Vue* qui est la partie chargée de la présentation (IHM)
* *Le Contrôleur* qui gère la synchronisation entre la vue et le modèle, réagit à l’action de l’utilisateur met à jour le modèle et informe la vue des modifications.

=schéma de l’architecture de l’appli

🡺 schéma appli en ligne (GAPPEngine) + Cron tâche planifié en fond+ Architecture Servlet JSP + récupération restfull api tomato

# Réalisation du projet web

## Environnement matériel

## Environnement logiciel

## Modèle de données de l’application

## Initialisation du projet

Pour démarrer le projet, on crée un projet de type « Maven » avec l’archétype *« A REMPLIR »* à partir de Eclipse.

## Nos extensions