**Architecture Logicielle : Gestion Pizzeria**

Pour ce projet, n’ayant pas pu faire fonctionner struts sur notre environnement de travail lors du cours, nous avons décidé d’utiliser **spring-boot** afin de s’affranchir de la configuration d’un serveur JEE.On utilise les page **jsp** couplé avec **jstl** pour afficher les pages du site.

Nous utilisons une base de données my-sql pour persister les informations.

Nous avons utilisé git pour le travail collaboratif.

Nous avons utiliser les dépendance suivante dans notre projet:

<artifactId>spring-boot-starter-data-jpa</artifactId>

<artifactId>spring-boot-starter-web</artifactId>

<artifactId>spring-boot-starter-test</artifactId>

<artifactId>spring-boot-starter-tomcat</artifactId>

<artifactId>jstl</artifactId>

<artifactId>tomcat-embed-jasper</artifactId>

<artifactId>mysql-connector-java</artifactId>

Suite au démarrage technique très difficile du projet (plusieurs jours ,problème de machine virtuelle, méconnaissance de spring jsp jstl, prise en main de git). nous avons eu beaucoup de mal à travailler dès le début en équipe sur une plateforme commune à partir de git. il a fallu revoir le projet à la baisse.

Nous avons décidé que l’objectif de notre projet serai de présenté un site semi-fonctionnel sur des données minimalistes afin de pouvoir présenter une première version visuelle et navigable dans le délai imposé.

les fonctionnalités implémentées sont les suivantes :

Gérant :

Gestion des ingrédients contenus dans les pizzas

Gestion carte contenant des pizzas

Liste des commandes passées par le client

Modifier le status de la commande, voir le détails

Liste des Clients

User => utilisateur avec un compte client

VISITEUR => Client sont compte mais qui ont passé une commande

Client :

Gestion d’un panier en session (liste de pizzas)

Confirmation du panier,

persistance de la commande en l’affectant au USER en session

User :

Inscription d’un user (ROLE USER)

Connexion d’un user (USER ou ADMINISTRATOR(gérant)) enregistrement en session.

Gestion du menu client, menu gérant est du droit d’accès des pages suivant l’utilisateur connecté.

Nous avons organisé en package l’application suivant

3 parties:

io.ldnr.teamc.pizzeria.Client

io.ldnr.teamc.pizzeria.Gerant

io.ldnr.teamc.pizzeria.user

Dans chacun de ces packages on trouve un ou différents controllers représentant les fonctionnalités de chaque partie

Nous avons créé un package pour chaque table de donnée contenant le mapping de la table et son repository afin de dissocier les datas des controllers.

Les différentes jsp se trouvent dans 3 dossiers représentant les 3 parties du projet.

Représentation des packages

Une image contenant texte

Description générée automatiquement

Représentation des jsp

Une image contenant texte

Description générée automatiquement

**Gestion des équipes :**

Fonctionnalités principales de chacun :

Sébastien BATUT

Prise en charge de trouver une solution technique pour pouvoir démarrer le projet, création du projet, dépendances, package. Test affichage simples de données jsp à partir de la table employee, company, adresse de l’exercice vue en cours pour avoir un exemple.

Partie gérant

Mappage table ingrédient,pizza

Partie gerant

Création de la gestion des ingrédients, controller + vue

Création gestion carte controller , vue pizza upadte

+ détails commandes

fettha AMAROUCH

création de l’uml + création base de donnée

gestion du panier, persistance du panier en commande

brian THUR

aide au démarrage technique git,

partie gérant

création des vues gestion carte,

gestion liste user, listecommande , modification status

Maxo

Gestion user : inscription , connexion (session) , déconnexion

Creation menu(css), gestion des liens connexion,deconnexion

Brian , Sebastien :

Intégration + gestion de la navigabilité + sécurité page

TOUS :

Entraide mutuelle de chacun afin de faire évoluer le projet vers un objectif final.

Remarques :

pour ce petit projet nous avons dû consacrer un temp important pour prendre en main les outils comme git,

Au niveau de la base de données n’ayant pas de serveur partagé cela nous a ralenti.

Durant ce projet nous avons essayé de nous aider afin que tout le monde puisse gérer une partie du projet, et surtout d’intégrer les modifications de chacun afin d’obtenir un projet final.