

# Examen\_Pratique

Semestre : 1 ☒ 2 ☐

Session : Principale ☒ Rattrapage ☐

Module : Framework Java Spring

Enseignant(s) : Ben Khelifa Seddiki Rabeb, Bessrou Mohamed Amine, Anniba Wael

Niveau(s) : 5INFOGL, 5INFOB

Documents autorisés : OUI ☒ NON ☐

Nombre de pages : 4

Date : 28/11/2015

Heure: 9H

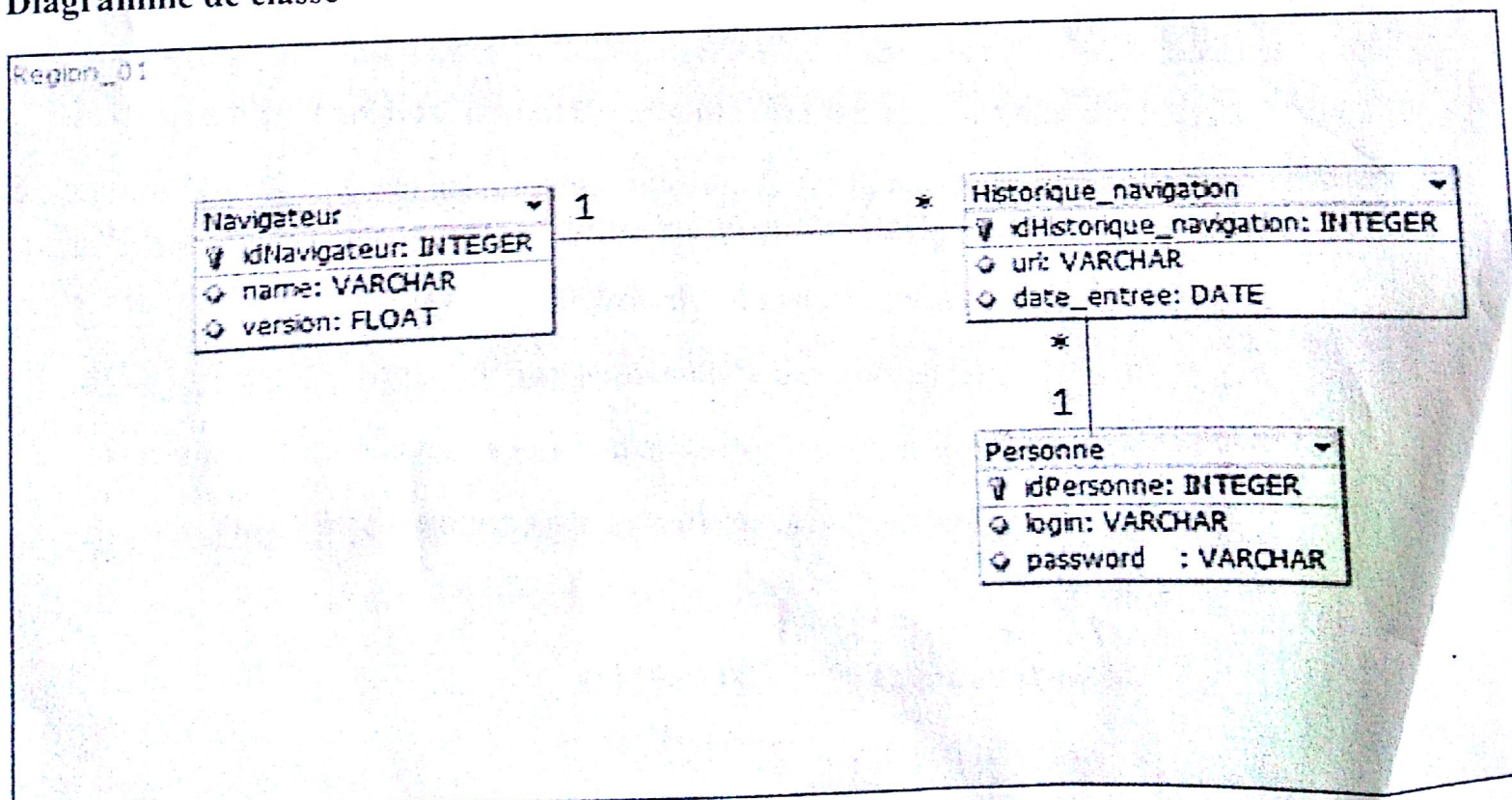
Durée 1H30

Nous vous présentons un module simplifié d'une application de gestion des historiques de navigation d'une personne à partir d'un navigateur. Nous vous demandons d'implémenter ce module avec l'environnement de travail suivant :

Framework Spring, JPA2.0, Mysql5, AOP, Spring-MVC, Serveur Web(Tomcat7)

NB : la nomenclature des méthodes, des classes, ou des packages est au choix.

Diagramme de classe





## Partie A: 10 pts

### Projet\_Metier :Spring IOC,JPA

- Créer un projet Spring avec Maven intitulé « **Examen-Navigation** » .
- Implémenter le diagramme de classes structures en packages : Le Projet suit une architecture 3-tiers et applique le design pattern DAO (de manière à ce que chaque entité dispose d'un DAO).
- Injecter les beans relatives aux diagrammes dans le fichier de configuration de Spring (que vous créez de votre choix).
- Gérer la connexion à la BD que vous créez (de votre choix) via Spring -JPA
- Mapper les entités via des métas donnés.
- Implémenter les diverses relations présentées dans le diagramme.
- Implémenter les services suivants au moyen des fonctions correspondantes :
  - Créer un navigateur
  - Chercher un navigateur
  - Créer une Personne
  - Chercher une personne
  - Déterminer l'historique de navigation d'une personne sur des urls, via un navigateur.
  - Afficher la liste des historiques de navigation.

### TAF :

1. Générer le schéma de la BD et vérifier la création des tables avec les relations. **5pts**
2. Créer une classe de test (Test Unitaire) pour chacune des fonctionnalités (services) que vous avez déjà implémentées précédemment comme suit :

2.1 : Créer le navigateur suivant :« *'firefox, '2.5'* » :**1pt**

2.2 : Créer la personne suivante « *'admin,'admin'* » :**1pt**

2.3 : La personne déjà créée dispose des historiques des connexions suivants

Suite à sa navigation sur le navigateur **firefox** précédemment crée : **3pts**

« *'www.springide.org, '20/11/2015'* »

« *'www.facebook.com, '20/11/2015'* »



## PARTIE B :5pts

### Spring AOP

1. Créer un Aspect nommé **Performance** qui permet de calculer la durée de la navigation de la personne sur le navigateur en tenant compte de son historique de navigation. La durée est calculée en seconde de manière à avoir l'affichage suivant au niveau de la console : (2.5pts)

**ElapsedTime :\*\*\*s**

2. Lors de la navigation de la personne dans une url une exception se déclenche. Déclencher cette exception de manière à afficher au niveau de la console le message suivant : (2.5pts)

**Connexion TimeOut.**

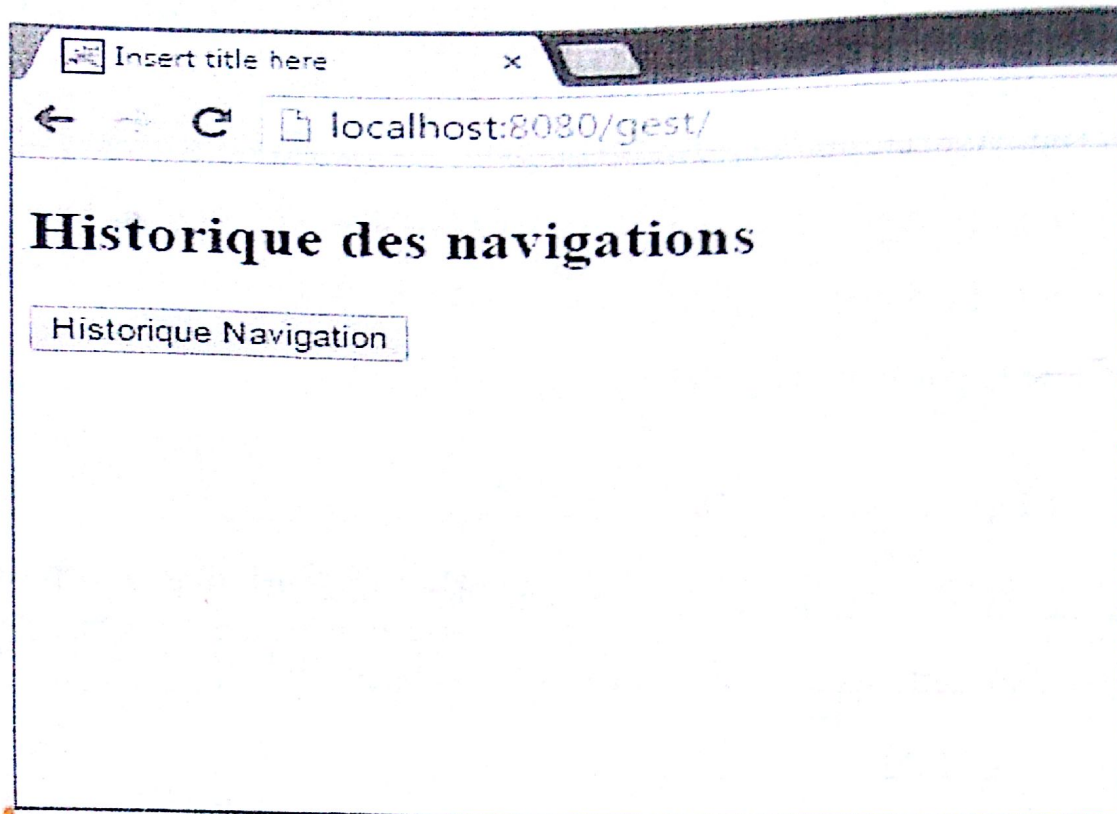
## PARTIE C :5pts

### Spring MVC

- Créer un projet Web avec Maven « **Examen-Navigation-Web** » qui consomme votre projet spring (créé dans la partie A)
- Charger au niveau de ce projet le contexte du projet métier (créé dans la partie A).
- Utiliser le framework spring MVC ,et Dispatch Servlet comme servlet de spring.
- Injecter les beans adéquats à votre projet Web dans le fichier de configuration de spring que vous créez de votre choix
- Créer **une page** web qui permet d'afficher l'historique de navigation de la personne.

### TAF :

L'exécution du projet Web sous le serveur Tomcat7 doit permettre l'affichage de la page demandée comme dépeint dans les figures suivantes : **5pts**



Un clic sur le bouton Historique Navigation permet l'affichage de la liste comme suit :

