

Module : **Spring & JavaEE**

Enseignant(s) : Spring Team

Classe(s) : **4INFOB**

Nombre de pages : **03**

Documents autorisés: **OUI**

Calculatrice autorisée: **NON**

Internet autorisée: **NON**

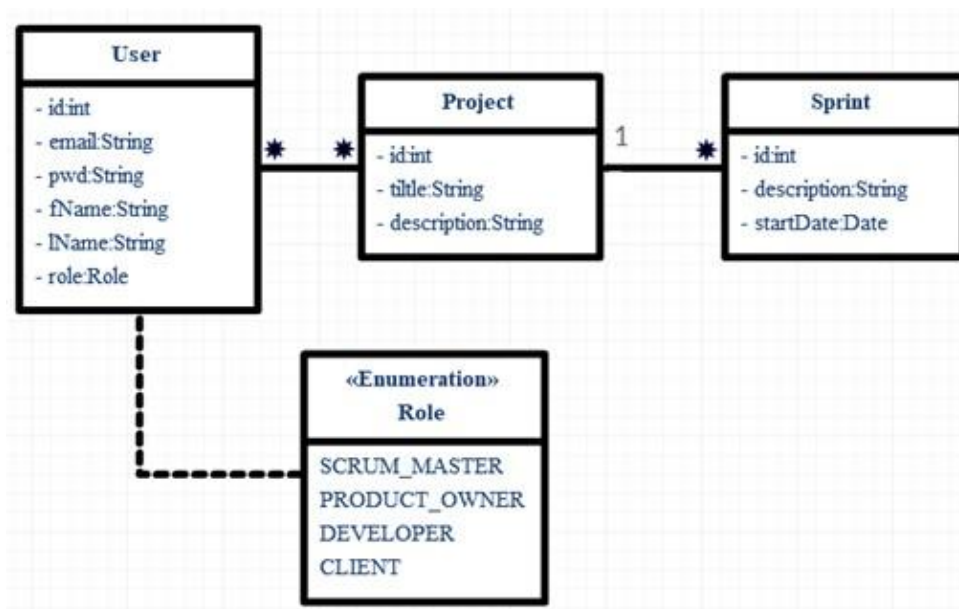
Date : 28/04/2020

**La validation de l'épreuve est appliquée sur la base d'un code source exécutable.
Aucun code source non fonctionnel n'est comptabilisé lors de la validation.**

On vous propose d'implémenter une application simplifiée de gestion des équipes qui utilisent la méthode Agile Scrum à Esprit.

-) Un Utilisateur (**User**) de notre application peut avoir l'un des rôles (**Role**) suivants au niveau des projets (**Project**) sur lesquels il intervient : **Scrum Master, Product Owner, Developer ou Client**.
-) Un utilisateur peut travailler sur plusieurs projets. Un projet peut avoir plusieurs intervenants.
-) Un projet est composé de plusieurs Itérations (**Sprint**). Un Sprint est lié à un unique projet.
-) Les User Stories ne seront pas traitées dans notre étude de cas.

Ci-dessous le diagramme de classes :



Partie I Spring Boot-Spring Core–Spring DataJPA–Spring REST MVC–Log4J (15 points) :

I.1. (4 points) :

Implémenter les entités qui permettent de générer le schéma de la base de données comme illustré dans le diagramme de classes sachant que :

-) Les identifiants des trois entités sont auto-générés avec la stratégie «IDENTITY».
-) L'association User-Project représente une relation bidirectionnelle. Le Project est le Child (Slave).
-) L'association Sprint-Project représente une relation bidirectionnelle. Le Project est le Child (Slave).
-) Une énumération Role est à créer pour gérer les rôles.

I.2. (11 points) :

Développer le code nécessaire dans une classe annotée par **@RestController** qui fait appel aux différents services.(Exposition des services avec Spring REST MVC).

- a) Ajouter les deux utilisateurs ci-dessous, en respectant la signature suivante **(2pts)** :
public void addUser (User user)

role	fName	lName	email	pwd
DEVELOPER	Mohamed	BOUHLEL	ab@esprit.tn	dev
CLIENT	Kais	ALLANI	ka@neoxam.tn	cli

- b) Ajouter les deux projets ci-dessous, en respectant la signature suivante **(2pts)** :
public void addProject (Project project)

title	description
MAP	Gestion de Mandats, Assignations et Projets
OTDAV	Gestion des Œuvres

- c) Affecter le projet MAP aux deux utilisateurs Mohamed BOUHLEL et Kais ALLANI, en respectant la signature suivante **(2pts)** :

public void assignProjectToUser(int projectId, int userId)

- d) Lister TOUS les projets, en respectant la signature suivante **(1pts)** :
public List<Project> getAllProjects()

- e) Lister les projets affectés à l'utilisateur Mohamed BOUHLEL, en respectant la signature suivante **(2pts)** :

public List<Project> getProjectsByUser(int userId)

- f) Ajouter un Sprint et l'affecter au projet MAP, comme indiqué ci-dessous, et en respectant la signature suivante **(2pts)** :

public void addSprintAndAssignToProject(String description, Date startDate, int idProject)

Sprint (description)	Sprint (startDate)
Sprint Mandat	la date d'aujourd'hui

Partie II JSF (5 points) :

Les captures d'écrans listées ci-dessous sont représentées à titre indicatif, il est de votre responsabilité d'implémenter les pages web en s'inspirant des dites captures.

II.1. Authentification (2 point) : Le formulaire accepte un login et password, notifie une éventuelle erreur d'authentification via un FacesMessage et redirige l'utilisateur vers son espace selon son Rôle :

-) Un espace pour les utilisateurs qui ont le rôle CLIENT ou PRODUCT OWNER.
-) Un espace pour les utilisateurs qui ont le rôle DEVELOPER ou SCRUM MASTER.

Outil de Gestion de Projet pour les équipes Agiles

Connexion

Login (Email)

Mot de passe

Méthode Agile utilisée : SCRUM

II.2. Espace Fonctionnel (Product Owner + Client) (1 points) : L'espace fonctionnel permet au Product Owner et au Client de voir la liste de **tous** les Projets.

Outil de Gestion de Projet pour les équipes Agiles

Bonjour Kais ALLANI | [Déconnexion](#)

Vous êtes dans l'espace pour CLIENT et PRODUCT_OWNER

Libellé	Description
MAP	Gestion de Mandats, Assignations et Projets
OTDAV	Gestion des Oeuvres

Méthode Agile utilisée : SCRUM

II.3. Espace Technique (Developer + Scrum Master) (2 points) : L'espace Technique permet aux développeurs et Scrum Master de voir les projets **qui leur sont affectés uniquement**.

Outil de Gestion de Projet pour les équipes Agiles

Bonjour Mohamed BOUHLEL | [Déconnexion](#)

Vous êtes dans l'espace pour DEVELOPER et SCRUM_MASTER

Libellé	Description
MAP	Gestion de Mandats, Assignations et Projets

Méthode Agile utilisée : SCRUM

Bon Travail !