Document d'Architecture Technique - Projet ITI 4

Application Android
Œnologie HEI

Etudiants : Thomas CHEVALIER
Etienne LE BOURLOUT

Nom du superviseur : David Dubois

Résumé du projet :

Le projet sera une application Android d'accompagnement pour les séances de l'association œnologie d'HEI. Le but de cette application est de rendre les séances plus interactives. Elle pourra présenter les vins des différentes séances ainsi que de proposer des quizz à la fin de la séance. Une partie gestion de stock sera disponible pour les administrateurs, à savoir les membres de l'association.

Références du document :

Référence	DAT Application Android Œnologie HEI
Nom du projet	Application Android Œnologie HEI

Validation :

Nom Validateur	Date	Validation (O/N)	Commentaires

Versions :

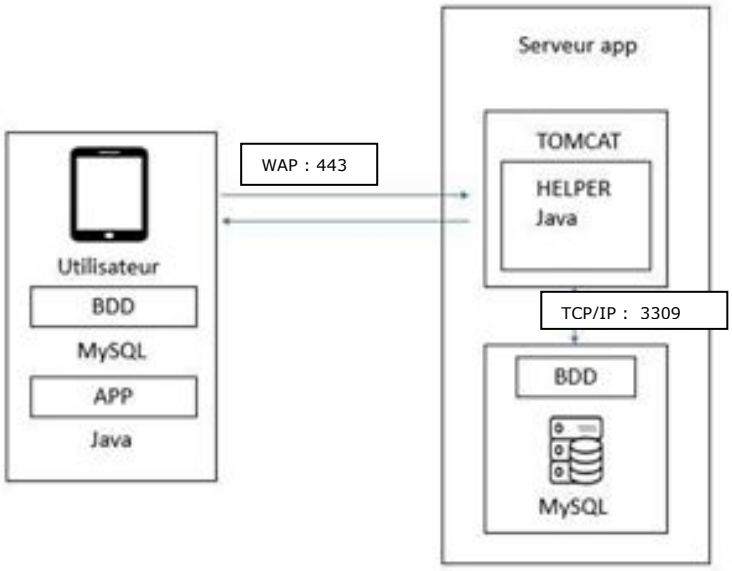
Numéro de Version	Date	Etat	Auteur(s)	Remarque(s) / modification(s) majeure(s)
Version 1.0	06/12/17	Fini	Chevalier /Le Bourlout	

Sommaire

CHEVALIER Thomas
LE BOURLOUT Etienne

1. ARCHITECTURE TECHNIQUE GENERALE

1.1 SCHEMA GLOBAL D'ARCHITECTURE



1.2 PLATEFORME TECHNIQUE

Type	*OS/Plateforme*	*Logiciel*	*Version*
Serveur de base de données	*Linux*	*MySQL*	*MySQL 57.19*
Serveur web	*Linux*	*Apache*	*Apache 2.4*
Serveur d'application	*Linux*	*Tomcat*	*8.0.15*
Application	*Windows*	*Android Studio*	*3.0.1*
Langage	*Java*		*8*

CHEVALIER Thomas
LE BOURLOUT Etienne

1.3 FLUX

De	Vers	Visibilité / Protocole	Port
Utilisateur	Apache	Internet / HTTPS	8080
Apache	Tomcat	Intranet / AJP 13	443
Tomcat	BDD	Intranet	3309

2. BATCHS / INTERFACES

2.1 TRAITEMENT 1

2.1.1 DESCRIPTION

Récupérer les questions du quizz sur l'application depuis la base de données au début de la séance.

2.1.2 FREQUENCE ET MODE D'EXECUTION

Exécution automatique à chaque début de séance.

2.1.3 DESCRIPTION DES ENTREES ET DES SORTIES

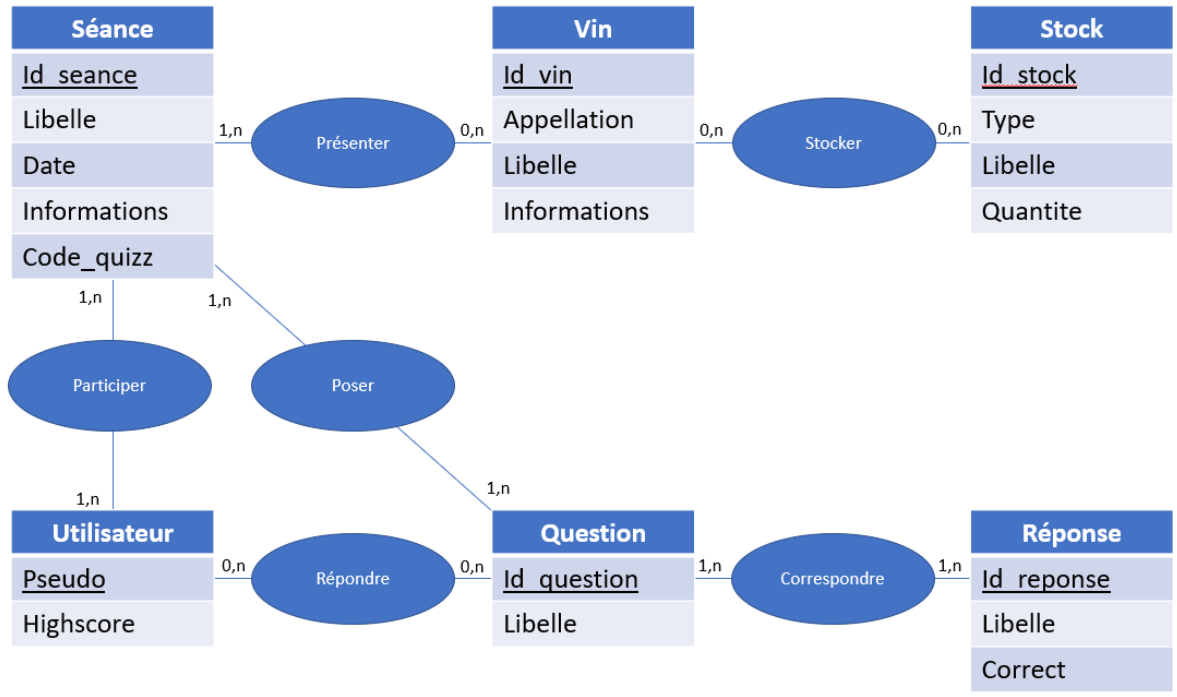
Suppression des anciennes questions du quizz et récupération des questions actuelles sur le mobile du l'utilisateur.

2.1.4 DESCRIPTION DU PROCESSUS DE GESTION D'ERREUR

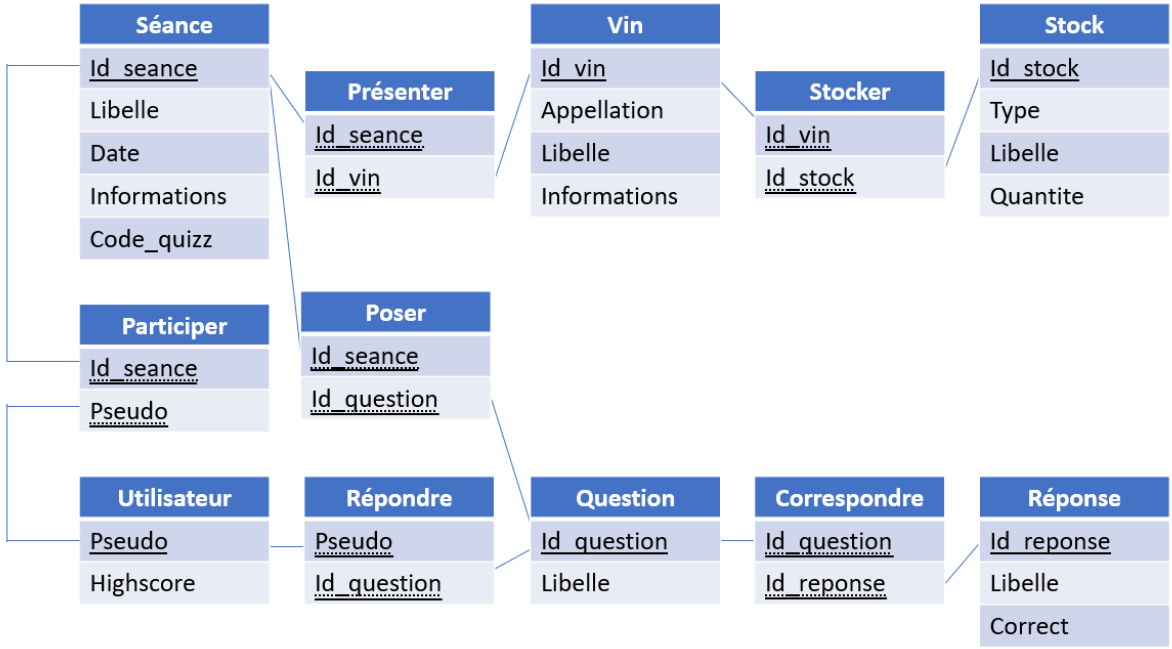
Renvoi d'une exception.

3. DESCRIPTION DES DONNEES

3.1 MODELE CONCEPTUEL (1.0)

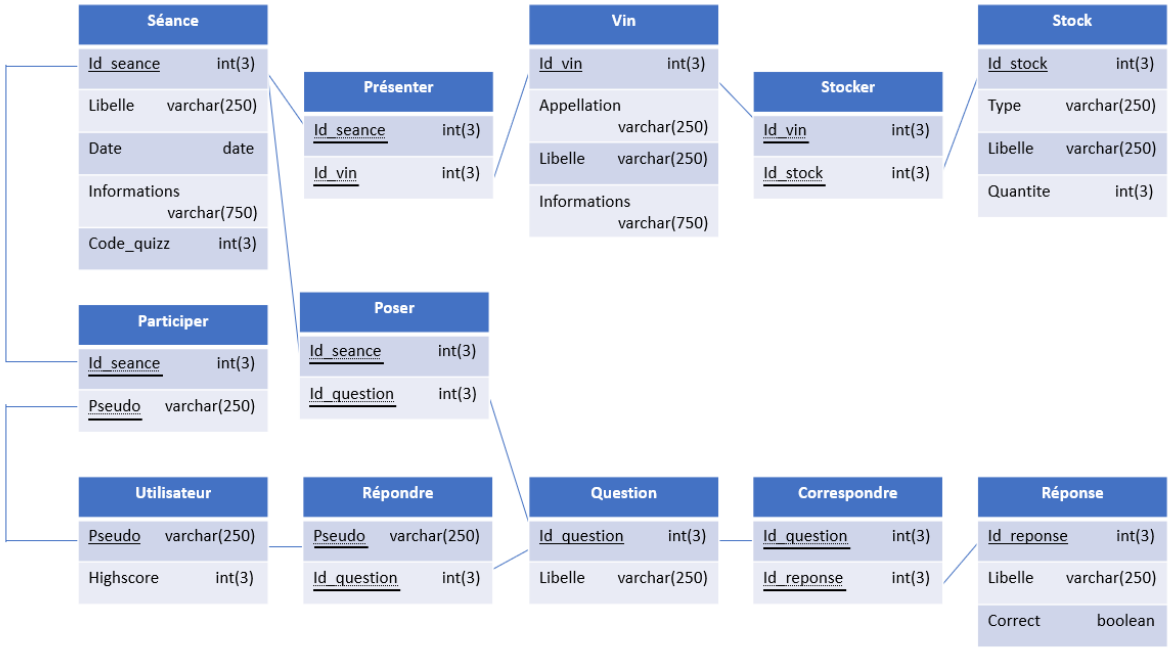


3.2 MODELE LOGIQUE (1.0)



CHEVALIER Thomas
LE BOURLOUT Etienne

3.3 MODELE PHYSIQUE (1.0)



4. DESCRIPTION DU CODE

4.1 HISTORIQUE DE LA SOLUTION

- BDD MySQL : Nous avons choisi ce logiciel car il est gratuit, simple d'utilisation, et compatible avec IntellIJ. De plus nous sommes habitués à travailler dessus. My SQL est utilisé par plusieurs grandes applications comme DropBox, AirBnB, Pinterest il est donc fiable.

- Android Studio : Pour coder l'application mobile, nous utiliserons ce logiciel car il est gratuit, et il s'agit de l'IDE pour le codage Android officiel de Google. C'est d'ailleurs le plus utilisé par les entreprises pour développer des applications Android.

- Nous utiliserons le langage Java. Nous avons choisi ce langage car c'est le langage le plus utilisé dans le monde en 2016, on peut donc dire qu'il est fiable. Si nous rencontrons certains problèmes avec ce langage, nous trouverons facilement la solution car la communauté l'utilisant est grande. On peut rajouter qu'il est open source, ce qui signifie que nous pouvons utiliser un nombre important de fonctionnalités mise à jour par la communauté.

- Pour le développement du serveur, nous utiliserons IntelliJ IDEA, en tant qu'étudiants il nous est gratuit. De plus, nous avons déjà effectué des développement web sur cet outil en cours, nous sommes donc familiers à son

utilisation. Il est fiable et facile d'utilisation. Il a été élu meilleur IDE pour Java selon le site Stackshare.io.

- Le serveur sera sur Apache Tomcat. Cet outil est compris dans Intellij ce qui facilite son utilisation. De plus, il est utilisé par un grand nombre de site web comme eBay, Evernote ou TripAdvisor. Il est populaire et simple d'utilisation.

4.2 ARCHITECTURE DU CODE (1.0)

Diagramme de séquence : Quizz

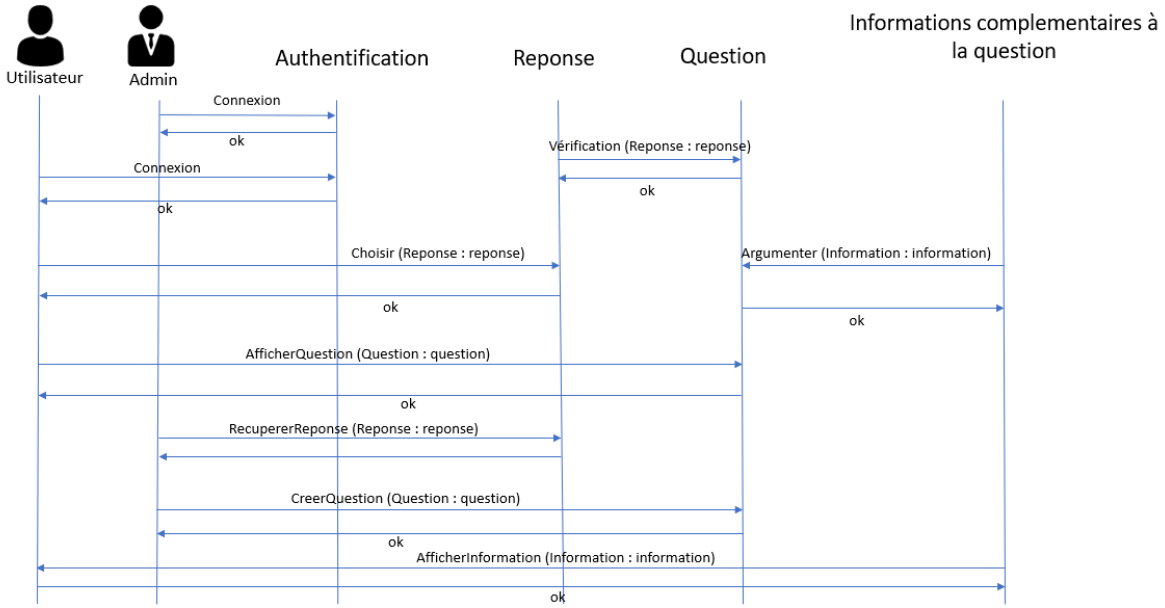
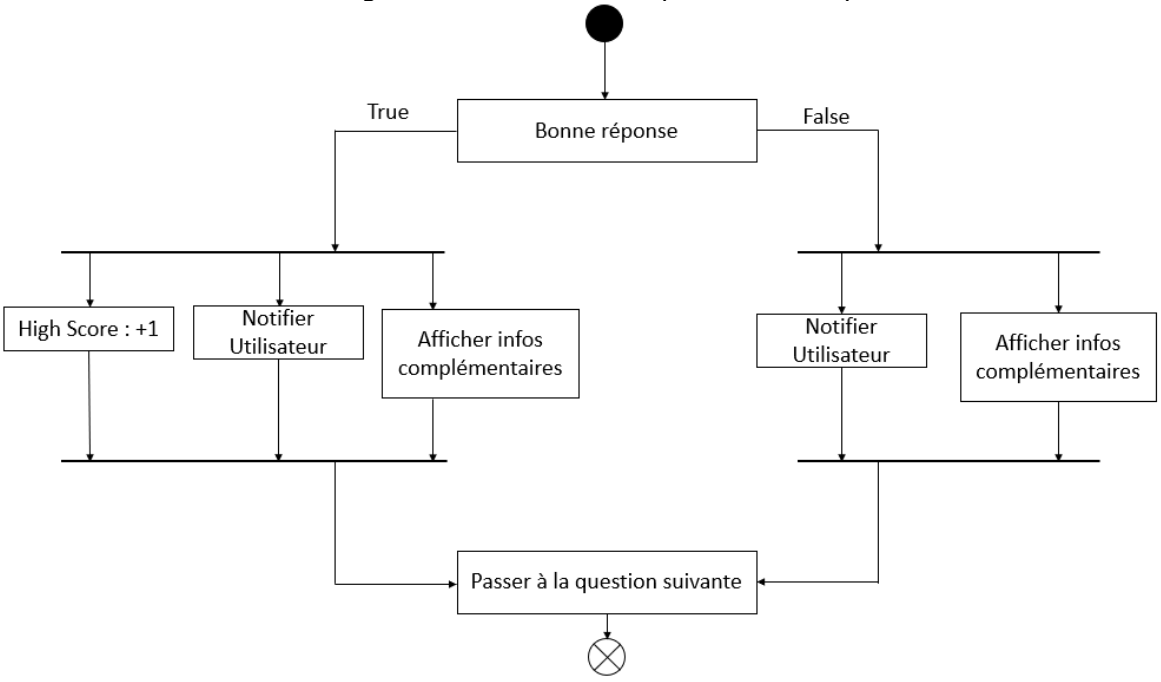
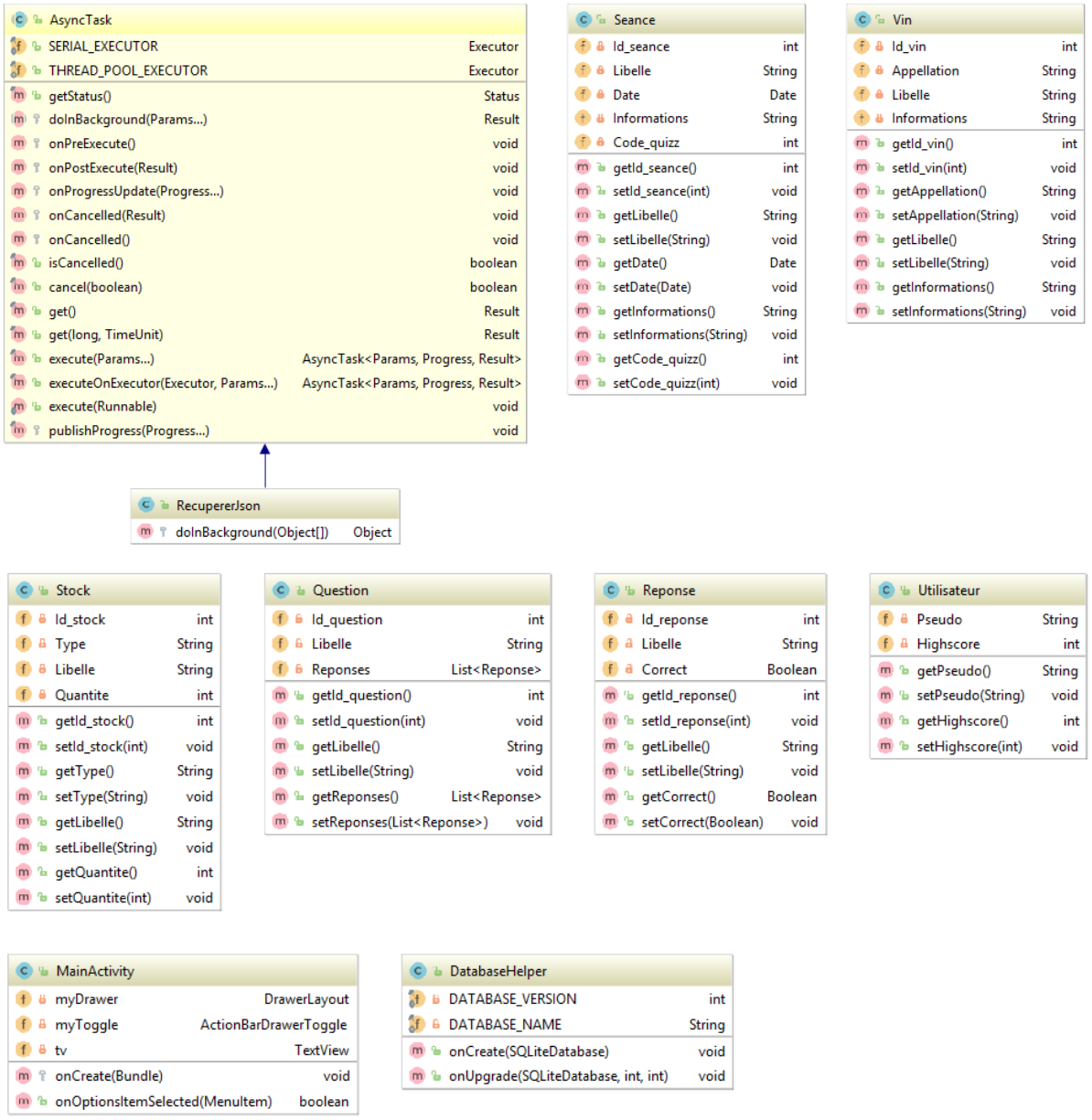


Diagramme d'activité : réponse à une question



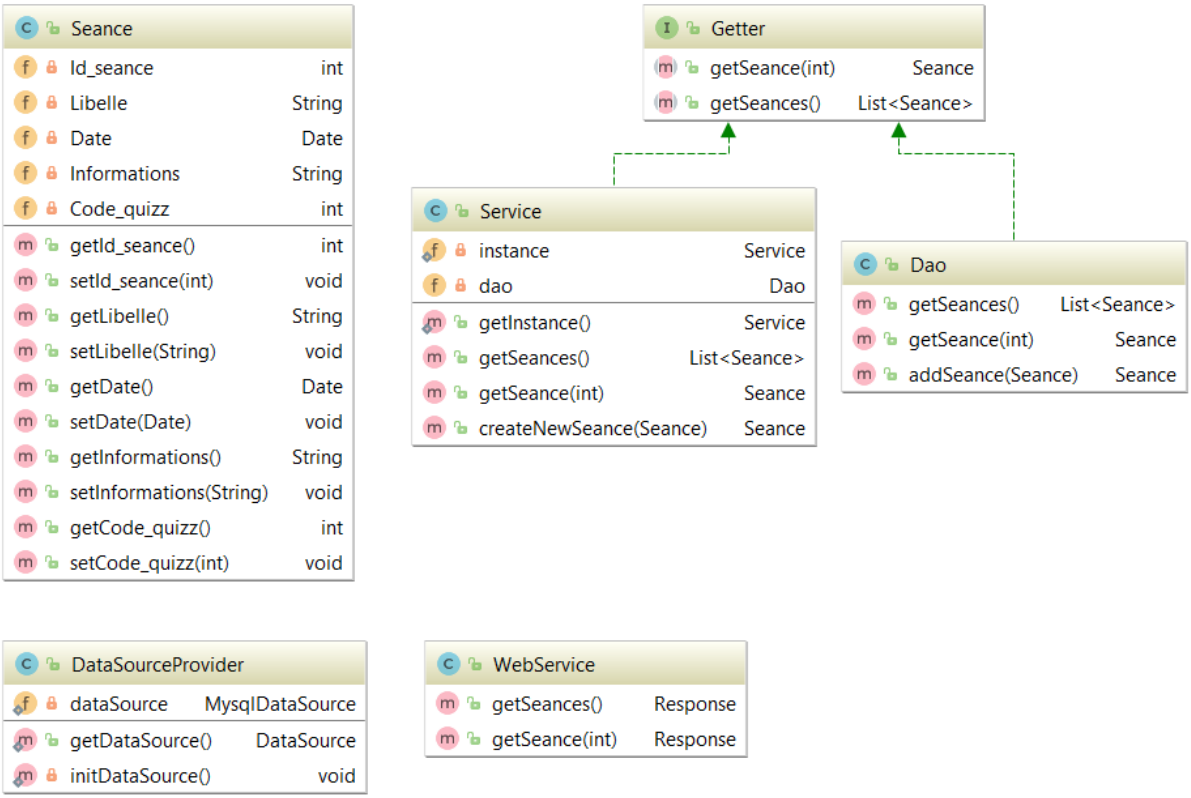
CHEVALIER Thomas
LE BOURLOUT Etienne

Diagramme de classe : Application Mobile



CHEVALIER Thomas
LE BOURLOUT Etienne

Diagramme de classe : serveur Tomcat



Sur ce diagramme de classe, nous avons seulement représenter la classe séance pour une meilleure lisibilité, mais il est applicable pour toutes les autres classes (vin, stock, question, réponse, utilisateur) présentes sur le diagramme précedent.

4.3 MECANISMES D'IDENTIFICATION ET D'AUTHENTIFICATION

L'accès à l'authentification sera disponible à tout moment pour les administrateurs, et seulement pendant la séance à laquelle les participants extérieurs à l'association assistent.

Pour les administrateurs, l'authentification se fera par une combinaison « code_quizz / pseudo » spécifique (c.f. la base de données) pour qu'ils puissent accèder au stock et à la partie gestion de l'application.

Pour les participants à une séance, l'authentification sur notre application se fera simplement avec un pseudonyme. Pendant la séance, le président de l'association donnera un code d'accès (code_quizz) aux participants pour qu'ils puissent rentrer leur pseudonyme. Ils pourront alors démarrer le quizz puis laisser un avis à la fin de celui-ci sur la séance à laquelle ils viennent de participer.

Nous avons choisi ce système car nous estimons qu'il n'est pas nécessaire qu'un utilisateur s'enregistre, les seules données qu'il nous communique étant des réponses à des questions n'interessant que l'association pour pouvoir s'auto-évaluer et ainsi pouvoir s'améliorer au fil de l'année.

4.4 GESTION DES DIFFERENTES LANGUES

Nous avons choisi en accord avec le président de l'association que l'application serait en français. En effet, les participants à une séance étant français (hors étudiant internationaux), les seules personnes ne parlant pas notre langue à qui pourrait être présentée l'application seraient des personnes rencontrées lors des évenements auxquels participe l'association. Cependant, sauf exception, ces évènements se déroulent dans la région lilloise.

4.5 DESCRIPTION DU PROCESSUS DE GESTION D'ERREUR

Les erreurs seront gérées par des exceptions. Nous créerons donc une classe à cet effet.

4.6 GESTION DES ACCES CONCURRENTS

Le seul problème d'accès concurrent que nous pourrions rencontrer est que deux administrateurs se connectent en même temps pour modifier le stock.
Dans ce cas, nous envisageons de bloquer l'accès au compte administrateur si une session est déjà ouverte.

4.7 SECURITE

C.f. Partie 4.3

5. PLATEFORMES MATERIELLES

5.1 ENVIRONNEMENT PRECONISE

L'application pourra être installée sur n'importe quel téléphone Android ayant une connexion internet et une version minumum d'android 5.1 (Lollipop).

5.2 SPECIFICITES RELATIVES AUX PERFORMANCES

Le serveur d'application devra supporter une charge d'envoie et de récupération de données d'environ 50 personnes simultanément.

6. ANNEXE : NORMES ET STANDARDS DE REALISATION

L'application doit répondre aux normes et standards en vigueur au Conseil de l'Europe.
Si certaines directives n'ont pu être mises en œuvre, en préciser les raisons.