

IPST-CNAM
Architecture Logicielles
NSY 205
Mercredi 12 Février 2020

Durée : **2 h 30**
Enseignant : LAFORGUE Jacques

1ère Session NSY 205

CORRECTION

1ère PARTIE – SANS DOCUMENT (durée : 1h15)

1. QCM (35 points)

Mode d'emploi :

Ce sujet est un QCM dont les questions sont de 3 natures :

- **les questions à 2 propositions**: dans ce cas une seule des 2 propositions est bonne.
 - +1 pour la réponse bonne
 - -1 pour la réponse fausse
- **les questions à 3 propositions** dont 1 seule proposition est bonne
 - + 1 pour la réponse bonne
 - -1/2 pour chaque réponse fausse
- **les questions à 3 propositions** dont 1 seule proposition est fausse
 - + 1/2 pour chaque réponse bonne
 - -1 pour la réponse fausse

Il s'agit de faire une croix dans les cases de droite en face des propositions.

On peut remarquer que cocher toutes les propositions d'une question revient à ne rien cocher du tout (égal à 0).

Si vous devez raturer une croix, faites-le correctement afin qu'il n'y ait aucune ambiguïté.

N'oubliez pas d'inscrire en en-tête du QCM, votre nom et prénom.

Vous avez droit à **4 points** négatifs sans pénalité.

NOM:	PRENOM:
------	---------

Une architecture logicielle permet d'aider le chef de projet à répartir le travail de développement en fonction des compétences de chacun.		1.
1	OUI	X
2	NON	

Une architecture logicielle contient le dossier de test de validation du SI.		2.
1	OUI	
2	NON	X

Une Configuration Architecturale est composée de 3 parties : l' Architecture Applicative , l' Architecture Technique , et l' Architecture Physique .		3.
1	OUI	
2	NON	X

L' Architecture Fonctionnelle contient la description de l'allocation (traçabilité) entre les exigences du besoin et les <u>composants</u> de la Configuration Architecturale .		4.
1	OUI	X
2	NON	

L' Architecture Dynamique identifie et caractérise tous les composants d'une Configuration Architecturale qui sont liés par un connecteur créé dynamiquement grâce au principe d'injection de dépendance.		5.
1	OUI	
2	NON	X

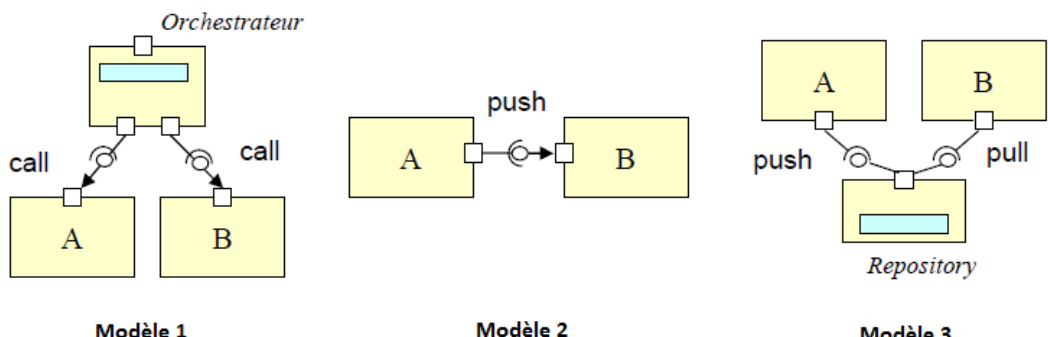
Dans la démarche d'architecture d'un Système d'Information, l' Architecture Dynamique :		6.
1	définit sur quoi, et où, s'exécutent les composants de son architecture.	
2	définit le comportement dynamique interne de chacun des composants de son architecture.	X
3	définit la coopération des composants entre eux au sein de son architecture.	X

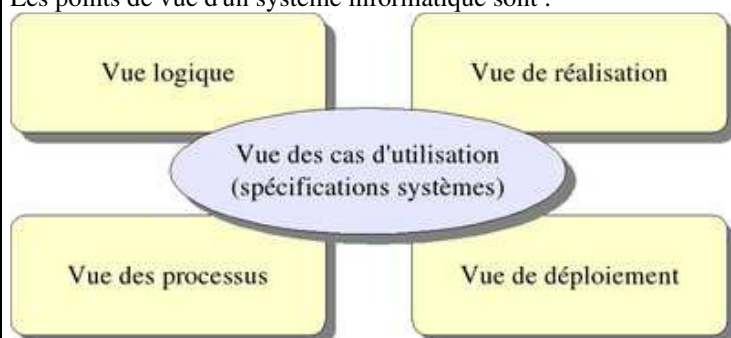
Dans la démarche d'architecture d'un Système d'Information, l' Architecture Technique identifie où doivent être exécutés les composants du SI.		7.
1	OUI	
2	NON	X

Dans la démarche d'architecture d'un SI, il existe deux Configurations Architecturales à aborder : la Configuration Architecturale Statique et la Configuration Architecturale Dynamique		8.
1	OUI	
2	NON	X

Le modèle de Kruchten définit 5 points de vue d'un SI : la vue logique , la vue des processus , la vue de réalisation , la vue de déploiement et la vue des cas d'utilisation . A quelle vue correspond la Configuration Technique ?		9.
1	la vue logique	
2	la vue de réalisation	
3	la vue de déploiement	X

Dans la démarche d'architecture, un connecteur entre deux composants permet de définir à la fois, un lien local de communication entre deux composants et un lien <u>distant</u> de communication entre deux composants situés dans des machines différentes.		10.
1	OUI	X
2	NON	

Soit les 3 modèles de coopération de composants suivants :		11.
 <p> Modèle 1 : Un orchestrateur au sommet appelle deux composants A et B en dessous. Les liens sont étiquetés 'call'. Modèle 2 : Un composant A à gauche pousse des données vers un composant B à droite. Le lien est étiqueté 'push'. Modèle 3 : Deux composants A et B sont au-dessus d'un composant 'Repository' en dessous. A pousse vers le Repository (étiqueté 'push') et le Repository tire de B (étiqueté 'pull'). </p>		
Quel est le mode de contrôle de chacun ces modèles :		
1	Modèle 1 = contrôle orienté procédure Modèle 2 = contrôle orienté données Modèle 3 = contrôle orienté évènements	
2	Modèle 1 = contrôle orienté procédure Modèle 2 = contrôle orienté évènements Modèle 3 = contrôle orienté données	X
3	Modèle 1 = contrôle orienté évènements Modèle 2 = contrôle orienté données Modèle 3 = contrôle orienté procédure	

Les points de vue d'un système informatique sont :		12.
		
1	La vue de réalisation permet de visualiser l'organisation des constituants dans l'environnement de test d'intégration et de validation.	X
2	La vue des processus décrit les interactions entre les différents processus situés dans les différents composants de son architecture.	
3	La vue logique permet de visualiser l'organisation des composants de son architecture.	X

Le principe de persistance en base de données est une propriété intrinsèque aux architectures à base de composants dans lesquelles tous les composants enregistrent en temps réel ses données dans un support de persistance qui est la base de données.		13.
1	OUI	
2	NON	X

Dans une architecture J2EE, un EJB Session Stateless permet de garder en mémoire un certain état de l'EJB.		14.
1	OUI	
2	NON	X

Une couche de persistance des données d'un Système d'Information est :		15.
1	une couche logicielle développée soi-même permettant de garder en base de données les attributs des objets (données).	X
2	une couche logicielle qui permet de garder en mémoire d'un composant dynamique les attributs des objets (données).	

Dans un ORM (Object Relationnel Mapping), on veut réaliser le mapping d'une relation 1..* entre deux classes, par exemple Département et Employé (un département contient plusieurs employés). Le principe de ce mapping est réalisé :		16.
1	sans table de jointure, en mettant la clef primaire de la table EMPLOYE dans la table DEPARTEMENT.	
2	sans table de jointure, en mettant la clef primaire de la table DEPARTEMENT dans la table EMPLOYE.	X

En JPA, l'annotation @OneToMany permet de traduire le mapping d'une relation *.*		17.
1	OUI	
2	NON	X

<p style="text-align: center;"><i>Tier métier</i> <i>Tier accès aux données</i></p>		18.
Dans ce schéma, le composant DAO (Data Access Object) :		
1	réalise le mapping entre les données des classes métiers et le SGBD.	X
2	transmet aux classe métiers les schémas de description des tables de la SGBD afin que les classes métiers réalisent les traitements de mise à jour de la base de données.	
3	contient toutes les instruction de mises à jour de la base de données.	X

Dans une architecture à base de composant, l'Injection de Dépendance permet de lier dynamiquement entre eux les composants locaux et/ou distants.		19.
1	OUI	X
2	NON	

Un EJB (Entreprise Java Beans) session est un composant d'une architecture J2EE. Une instance d'un tel composant peut être utilisé par plusieurs utilisateurs Web.		20.
1	OUI	
2	NON	X

<p>Ce schéma montre qu'il existe 2 façons différentes qu'un client utilise un Composant, soit directement (le lien "utilise") soit en passant par la couche d'Intergiciel qui sert d'intermédiaire entre le Client et les Composants.</p>		21.
1	OUI	
2	NON	X

<p>Un client envoie 2 requêtes consécutives (REQ1, REQ2) à un EJB session Stateless. Le EJB Stateless assure que si la requête REQ1 change l'état de l'EJB alors la requête REQ2 s'exécute avec l'état qui a été changé par la requête précédente REQ1.</p>		22.
1	OUI	
2	NON	X

<p>Dans une architecture J2EE, les EJB session Stateless et Stateful sont des classes</p>		23.
1	que l'on doit instancier dans un programme orchestrateur.	
2	qui sont instanciées par le container d'EJB.	X

<p>Un EJB session Statefull est une classe dont les attributs sont tous des Entity. Ainsi les attributs de la classe persistent en base de données, ce qui permet que l'état du EJB Stateful soit maintenu en mémoire.</p>		24.
1	OUI	
2	NON	X

<p>Un Message Driven Beans est un composant de l'architecture JEE qui s'abonne à un canal d'évènement.</p>		25.
1	Ce composant peut être mis en commun entre différents utilisateurs Web	
2	Différents utilisateurs peuvent envoyer des évènements dans ce même canal d'évènement.	X

<p>Le type de communication entre les composants d'une architecture MOM est une communication :</p>		26.
1	uniquement asynchrone	
2	uniquement synchrone	
3	asynchrone ou synchrone	X

<p>Dans une architecture J2EE les EJB Session sont tous des composants sans état interne. Leur état est géré en Session.</p>		27.
1	OUI	
2	NON	X

<p>Dans une architecture à base de composants, ces derniers sont toujours centralisés sur un serveur unique.</p>		28.
1	OUI	
2	NON	X

Un Web Service est un service informatique qui repose sur les standards de RMI.		29.
1	OUI	
2	NON	X
Dans une architecture Web Services qui repose sur le protocole SOAP, l'interface de définition d'un service est décrite par un fichier écrit au standard UDDI.		30.
1	OUI	
2	NON	X
Le protocole SOAP des architectures WEB Services est un protocole basé sur le déploiement sur les postes clients d' "amorces" permettant au client d'utiliser le service distant d'un serveur de type WEB Services.		31.
1	OUI	X
2	NON	
Le protocole SOAP (Simple Object Access Protocol) est un protocole qui permet une utilisation des services distants en mode RPC (Remote Process Call).		32.
1	OUI	X
2	NON	
Le principe d'un MOM est d'utiliser un composant logiciel qui sert d'intermédiaire entre les producteurs et les consommateurs		33.
1	OUI	X
2	NON	
SOA (Services Oriented Architecture) est :		34.
1	un framework Java de développement de Système d'Information	
2	un modèle d'architecture pour la réalisation de Système d'Information	X
3	une démarche de structuration du S.I. qui suit le principe de processus orchestrés	X
Dans une démarche d'architecture basée sur les principes SOA, les processus métier sont accessibles par des interfaces WSDL		35.
1	OUI	X
2	NON	

Fin du QCM

Suite (Tournez la page)

2. Questions libres (15 points)

Chaque question est notée sur 5 points.

Vous répondez à ces questions sur une copie vierge double en mettant bien le numéro de la question, sans oublier votre nom et prénom.

Vous mettez le QCM dans cette copie vierge double.

QUESTION NUMERO 1

Comparer les deux types d'architecture JEE et REST en citant 5 points importants qui les opposent et en les commentant.

1/ REST utilise les standards du Web (http) alors que JEE utilise les standards Java (JSP, Servlet, EJB). Ce qui fait de REST une solution plus adaptée à la réalisation des sites internet.

2/ REST utilise des RESSOURCES au lieu de DONNEES structurées comme c'est le cas de JEE. Cela en fait un système plus simple à mettre en œuvre

3/ REST est par principe un SI sans état (Stateless) sur le serveur alors que JEE non. En REST cela oblige à reporter cette gestion d'état sur le poste client et simplifie la conception du serveur.

4/ En REST les requêtes sont des URI d'accès à une RESSOURCE alors qu'en JEE ce sont des URI d'accès à des Servlets ou des JSP

5/ En REST le CRUD est décrit à travers les types de requête http (GET, POST, DELETE, PATCH, PUT) alors qu'en JEE le CRUD est décrit à travers le code de traitement des Servlets. Ce qui entraîne une disparité hétérogène et non standardisée en JEE et ne permet pas une interopérabilité entre les différents SI.

QUESTION NUMERO 2

Expliquez ce qu'est la "Configuration Architecturale" dans la démarche d'architecture d'un Système d'Information.

La Configuration Architecturale est la description de tous les COMPOSANTS du futur SI en mettant en évidence les liens de dépendance entre ces COMPOSANTS, Le sens de cette dépendance, la nature de ce lien sous la forme d'un CONNECTEUR. Cette description est faite sous la forme d'un schéma de diagramme des composants, accompagnée du rôle précis de chacun des composants.

Elle permet de savoir quels composants sont utilisés dans chaque cas d'utilisation et de vérifier ainsi que tous les cas d'utilisation sont réalisables.

La Configuration Architecturale permet de caractériser les composants en fonction de leur type fonctionnel (interface utilisatrice, traitements, données, ...) et donc leur répartition dans les différentes "tiers" du SI.

QUESTION NUMERO 3

Dans une architecture à base de composant de type JEE, expliquez le rôle des EJB Stateful et Stateless.

Le rôle des EJB Stateful et Stateless est d'implémenter les traitements métiers de l'application sans s'occuper de leur déploiement. Ces deux EJB sont des composants gérés en SESSION de chaque utilisateur du SI. Ainsi chaque utilisateur a ses propres EJB.

Le EJB session Stateful permet de gérer un état mémoire pendant tout le temps de l'utilisation de la session par l'utilisateur (ex gestion d'un panier de commande, gestion du profil de connexion).

Le EJB session Stateless qui est sans état mémoire, permet de réaliser les actions réalisées par l'utilisateur (exemple: la mise à jour de la base de données).

Fin de la 1^{ère} partie sans document

2ème PARTIE – AVEC DOCUMENT (durée : 1h15)**3. PROBLEME (50 points)**

Nous voulons faire l'architecture d'un Système d'Information dont l'objectif est de permettre aux utilisateurs de ce SI de réserver des salles de réunion sur des créneaux horaires (et qui pourrait aussi servir de planning connecté que nous ne traiterons pas dans le cadre de ce sujet).

Tout utilisateur doit s'inscrire et s'authentifier pour utiliser le SI.

Tout utilisateur doit pouvoir consulter toutes les salles et réservations connues du SI (mais ne s'affichent que les créneaux horaires des réservations).

Un utilisateur qui crée une réservation d'une salle :

- saisie le créneau horaire,
- précise, s'il veut une salle équipée d'un appareil de projection,
- précise le nombre de participant, ou précise les participants (*) à la réunion.

En fonction de toutes ces informations, le système lui propose la liste des salles disponibles correspondant à ses critères. Ensuite, il choisit parmi ces salles, celle qui veut.

(*) Le SI déduit de la liste des participants tous leurs créneaux libres pour faire son calcul.

Il faut que le système soit robuste à ce que deux utilisateurs ne peuvent pas réserver en même temps deux créneaux en intersection pour une même salle de réunion.

Tout utilisateur peut visualiser les réservations qu'il a créées. Ainsi, à tout moment, il peut, ajouter ou supprimer des participants.

Tout utilisateur peut consulter toutes les réunions de la semaine auxquelles il participe en tant qu'organisateur ou simple invité (cela lui fait une partie de son agenda).

Automatiquement le système supprime les réservations qui sont plus vieilles de 30 jours.

Il existe deux types de notification :

- une notification qui alerte un utilisateur de sa participation à une réunion ;
- une notification qui alerte un participant que la réunion commence dans 10mn.

Ces deux notifications sont réalisées par l'envoi d'un mail, et aussi sur une application Android d'un téléphone mobile si le participant a installé l'application et donné son numéro lors de l'inscription.

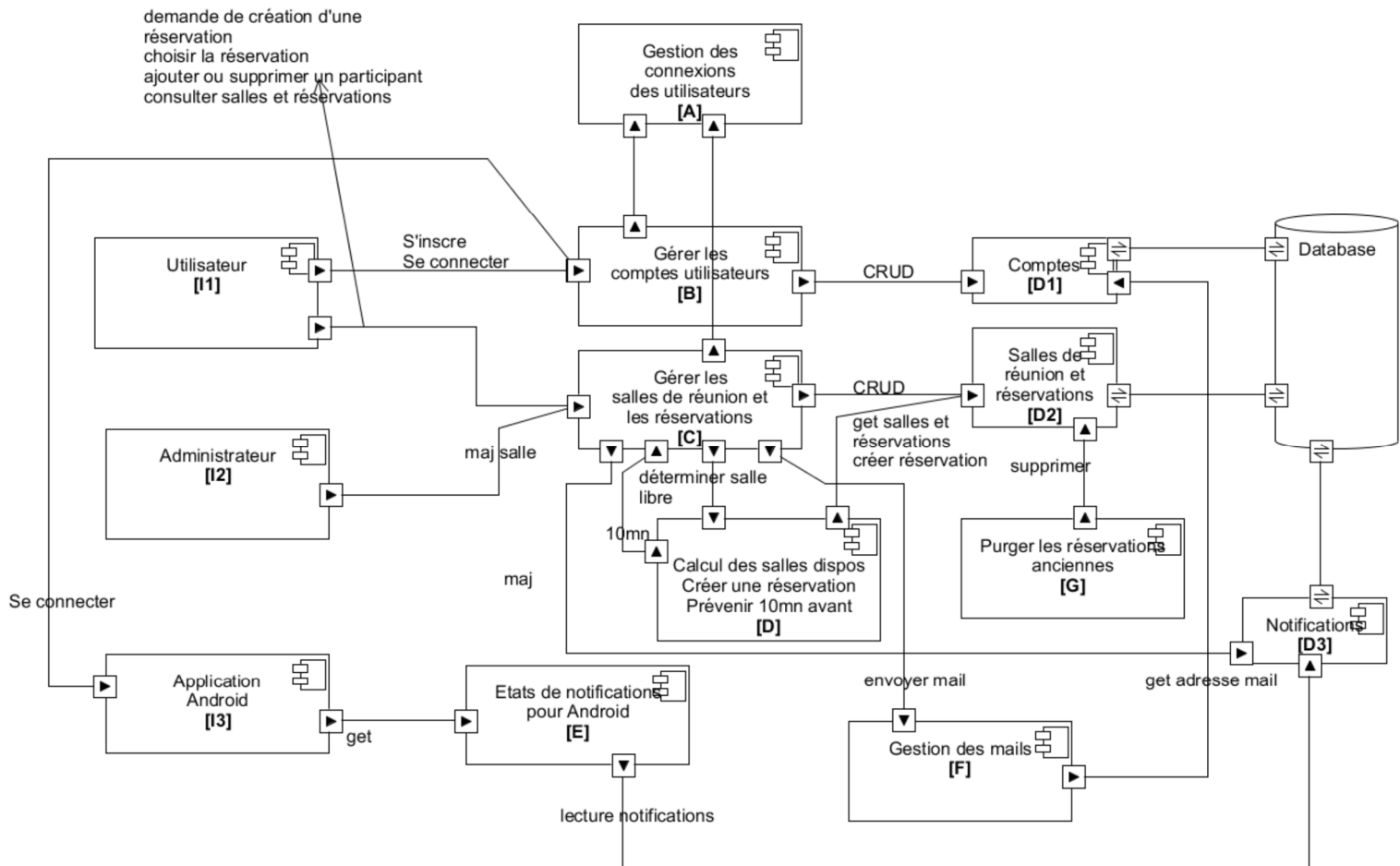
L'administrateur du site peut mettre à jour la liste des salles de réunions disponibles (nouvelle salle, modification de la capacité, mise à jour des moyens de projection, suppression ou blocage d'une salle).

Pour toute modification d'une salle de réunion utilisée, un mail est envoyé automatiquement aux créateurs de ces réservations afin de les prévenir.

Question 1 :

Faire la Configuration Architecturale de ce SI sous la forme d'un schéma d'architecture de composants et faire un texte de commentaires dans lequel vous précisez le rôle de chacun des composants, et leurs comportements (aspects dynamiques).

Pour rendre plus compréhensible votre commentaire, vous pouvez, dans le schéma, associer une lettre de l'alphabet à chacun des composants.



Les composants I1, I2 et I3 sont les interfaces utilisateurs, administrateur et Androids. Ils affichent les informations et réalisent les actions. Ces actions sont traitées par les composants B, C et E, comme indiqué sur le schéma.

Le composant D est un composant qui permet de rendre exclusif la création d'une réservation. Il calcule les salles disponibles en fonction de la demande de réservation.

Les composants D1, D2 et D3 sont les données du SI : les Comptes qui contiennent l'identifiant et l'adresse mail de chaque utilisateurs, les Salles de réunions et réservations et les Notifications pour les application Android.

Pour gérer la notification sur l'application Android (I3), le composant E permet de lire les notifications créées. Ainsi I3 demande régulièrement ces dernières au composant E qui les lit via D3. Ces notifications étant stockées en base de données, l'application Android s'initialise avec les dernières notifications lors de son lancement.

Le composant G est un composant toujours en tâche de fond permettant de supprimer les réservations trop vieilles.

Le composant F est le composant qui permet d'envoyer des mails. Il récupère les adresses mails enregistrées dans les comptes des utilisateurs.

Le composant D contient une tâche de fond permettant de prévenir les participants qu'une réunion commence dans 10mn.

A travers le rôle de chacun des composants et les liens de communication nommés du schéma, nous pouvons projeter facilement les cas d'utilisation issus du sujet. Exemple, la création d'une réservation :

- I1 permet de saisir les critères de demande de réservation
- I1 demande à C de traiter cette demande qui demande à D de calculer les salles disponibles dont la liste est retournée à I1
- I1 choisit la salle parmi cette liste. Le choix est envoyé à C qui demande à D de créer de manière exclusive la création de la réservation. Si la réservation est toujours réalisable, la réservation est créée via D2 et C met à jour les notifications et demande à F d'envoyer le mail de notification.

Question 2 :

L'application Android est en Java. Elle est capable de faire des requêtes AJAX.

Tous les utilisateurs et l'administrateur utilisent un navigateur Web.

En utilisant la technologie JEE faire le **schéma d'architecture technique** ET/OU faire une **description** précise de l'architecture technique en s'appuyant sur le schéma précédent, de l'ensemble de ce SI. Précisez bien la nature des composants JEE que vous utilisez. Situez ces composants dans les différents layers de l'architecture JEE (Client Layer, Web Layer, Business Logic Layer, Persistence Layer, Data base layer).

On reprend le même schéma que précédemment en faisant les adaptations et les qualifications qui suivent.

Le composant A est une EJB Stateful qui contient pour chaque utilisateur les éléments de connexion.

Le composant B se décompose en les composants B1 et B2 : B1 contient les pages JSP et Servlets, et B2 est un EJB Stateless pour réaliser l'inscription et la connexion au site.

Le composant C se décompose en les composants C1 et C2 : C1 contient les pages JSP et Servlets, et C2 est un EJB Stateless pour réaliser la demande de création d'une réservation, le choix d'une salle, ajouter ou supprimer un participant, consulter les salles de réunion, prévenir qu'une réunion commence dans 10mn.

Le composant D est un EJB Singleton qui calcule les salles disponibles, créer une réservation d'une manière synchrone. Ce composant contient un thread permettant de prévenir les participants qu'une réunion commence dans 10mn.

Le composant G est un EJB Singleton qui contient un thread qui supprime les réservations vieilles de plus de 30 jours.

Le composant E se décompose en les composants E1 et E2 : E1 contient un servlet et E2 est un EJB Stateless qui met à jour les notifications et qui retourne les notifications.

Le composant F est un EJB Stateless qui utilise une API SMTP permettant d'envoyer des mails. Ce composant utilise l'Entity Compte pour connaître l'adresse mail de quelqu'un.

Les composants D1, D2, et D3 sont des Entity qui permettent de consulter et mettre à jour les Comptes, Salles de réunion et réservations et les Notifications dans la base de données.

Les composants I1 et I2 sont des navigateurs Web et I3 est une application Android. I1 et I2 utilisent les servlets et les pages JSP de B1, C1 et E1.

I1, I2 et I3 appartiennent au Client Layer.

C1, B1 et E1 appartiennent au Web Layer

A, B2, C2, D, E2, F et G appartiennent au Business Logic Layer

D1, D2 et D3 appartiennent au Persistence Layer

Database appartient au Data base layer.

Fin du sujet