Titre du projet (à modifier)

**Groupe No :** à remplir par les soins du groupe.

Rapport d’élaboration : variante avec composants

à remplir par les soins du groupe.

Nom prénom

Nom prénom

Nom prénom

**Filière :** Informatique

**Technologies :** à remplir par les soins du groupe.

**Date du rendu :** à remplir par les soins du groupe

**Superviseurs :** Prof. Houda Chabbi Drissi

Prof. Pierre Kuonen

Prof. Omar Abou Khaled

**Client:** à remplir par les soins du groupe.

**Table des matières**

1 Changements par rapport à la phase de création 3

2 Présentation des composants utilisés 3

3 Diagrammes de séquence et de communication 3

3.1 Cas 1 : Nom du cas d'utilisation 3

3.1.1 Diagramme(s) de séquence 3

3.1.2 Diagramme(s) de communication 4

3.1.3 Transaction(s) 4

3.2 Cas 2: Nom du cas d'utilisation 6

3.2.1 Diagramme(s) de séquence 6

3.2.2 Diagramme(s) de communication 6

3.2.3 Transaction(s) 6

4 Liste des objets et composants 6

5 Diagramme de classes 6

6 Concurrence 7

6.1 Récapitulatif sur les transactions 7

6.2 Présentation d’un scénario concurrentiel 8

7 Diagramme de composants 8

8 Signatures 9

# Changements par rapport à la phase de création

Changement dans le cas d’utilisation qui modifie le compte (Rajout d’un bouton éditable).

# Présentation des composants utilisés

Pour chacun des composants que vous pensez utiliser, vous devez en faire :

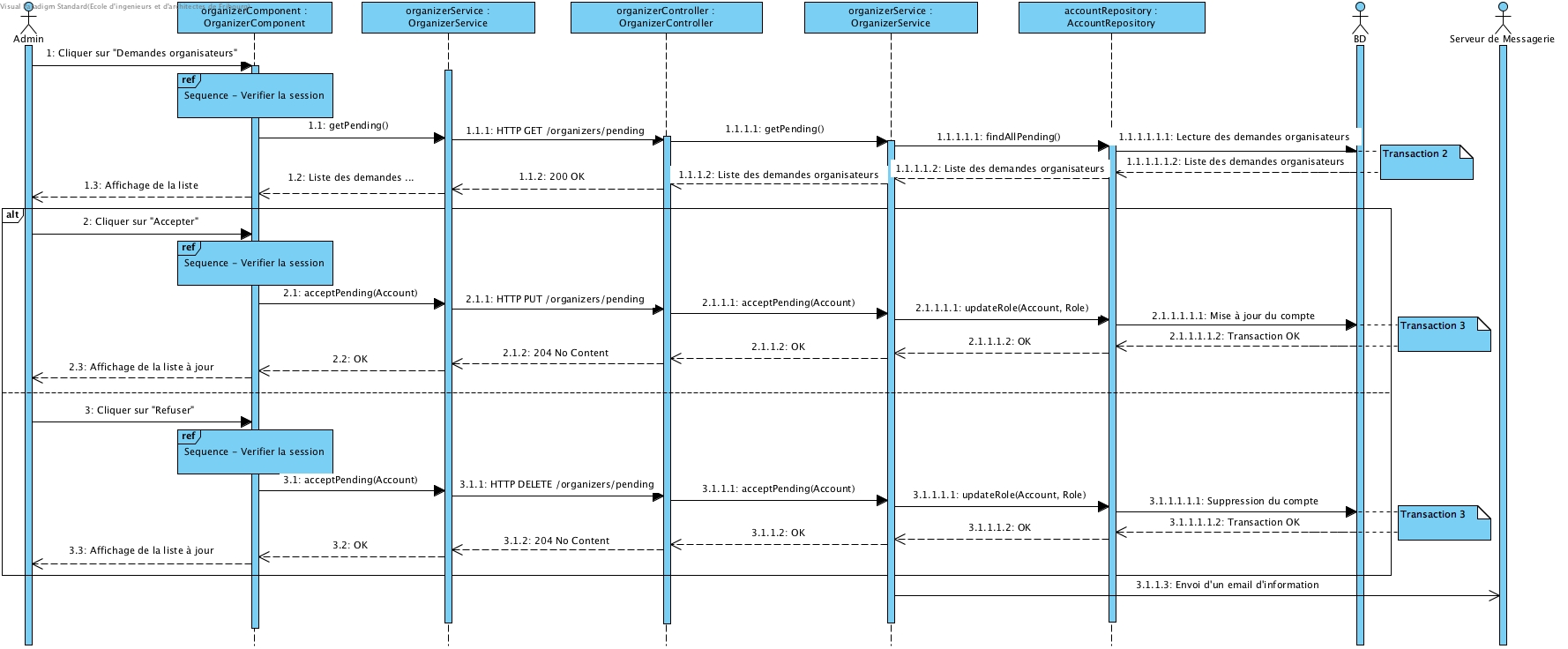
* Une présentation globale (enlever la virgule)
* Son utilité par rapport à votre projet
* Une présentation détaillée des contraintes que cela vous impose :
  + Au niveau de la modélisation en générale de votre application (à rajouter)
  + Au niveau de la modélisation de votre base de donnéesec
  + Au niveau des accès à la base
  + Au niveau du transactionnel

# Diagrammes de séquence et de communication

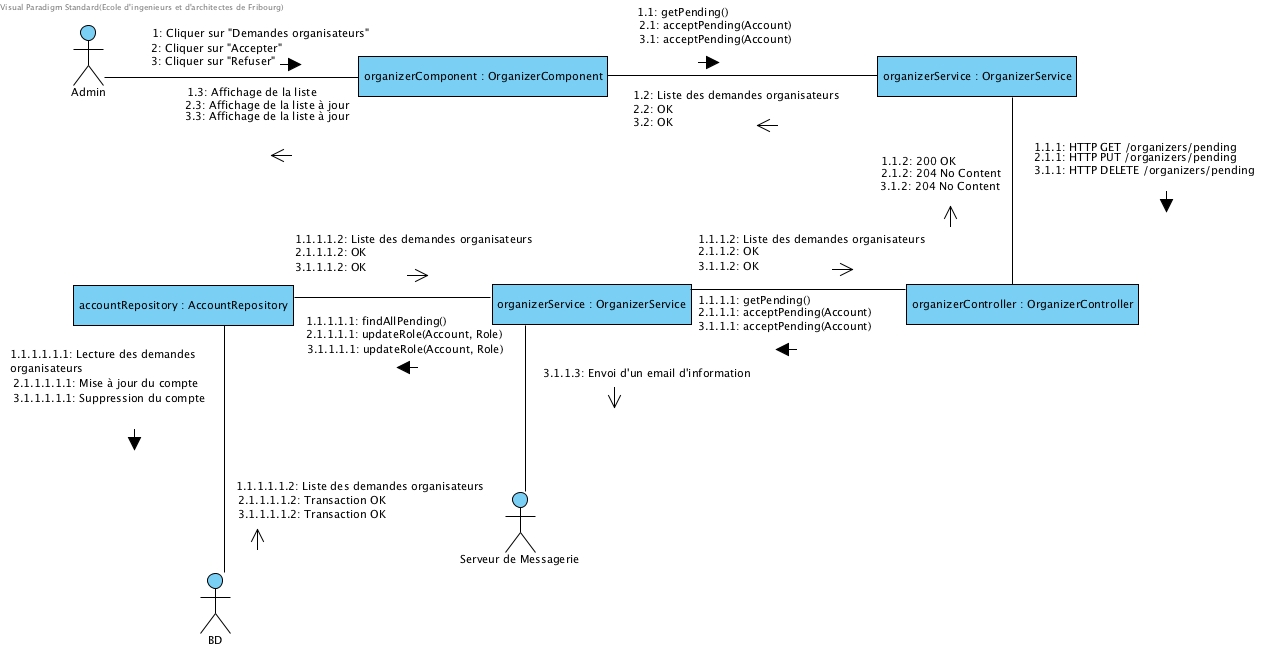
## Cas 1 : Définir les privilèges

Un administrateur peut accepter la demande d’une personne qui souhaite devenir organisateur.

### Diagramme(s) de séquence



### Diagramme(s) de communication



### Transactions

1. **Transaction 2**

La transaction va chercher la liste des personnes inscrites souhaitant devenir des organisateurs.

**Type de la transaction :** Lecture

**Isolation :** Serializable

Dans le cas ou deux admins (A1, A2) sont connectées, les deux ont chargées la liste des personnes qui souhaitent devenir un organisateur, si A1 a validé (ajouté) ou invalidé (supprimé) un organisateur après que A2 ait récupéré la liste, dans la vue de A2, il y aura une personne n’existant plus dans la base de données.

**Table(s) touchée(s) :** Account

**Fréquence :** Rare

**Période(s) :** La transaction est constamment active.

1. **Transaction 3**

La transaction va changer le rôle du compte souhaitant devenir organisateur.

**Type de la transaction :** Ecriture

**Isolation :** Serializable

Dans le cas ou deux admins (A1, A2) sont connectées, les deux ont chargées la liste des personnes qui souhaitent devenir un organisateur, si A1 a validé (ajouté) ou invalidé (supprimé) un organisateur après que A2 ait récupéré la liste, dans la vue de A2, il y aura une personne n’existant plus dans la base de données.

**Table(s) touchée(s) :** Account

**Fréquence :** Rare

**Période(s) :** La transaction est constamment active.

## Cas 2: Nom du cas d'utilisation

Résumé.....

### Diagramme(s) de séquence

### Diagramme(s) de communication

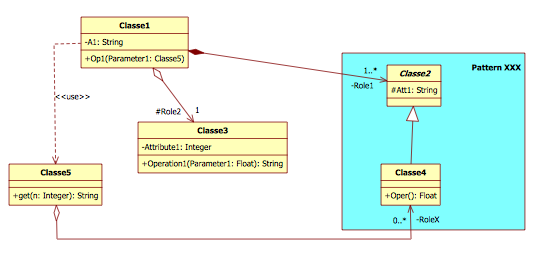
### Transaction(s)

# Liste des objets et composants

Dans ce chapitre indiquez la liste de tous les objets et composants (au format de la table ci-dessous) que vous utilisez dans vos diagrammes de séquence et de communication. Pour chaque objet indiquez dans quel(s) cas d'utilisation(s) il est utilisé et de quelle classe il est une instance.

|  |  |  |  |
| --- | --- | --- | --- |
| No | Nom de l’objet ou du composant | Nome de la classe | No. Des Use Case |
| 1 | ObjetA | ClasseB | 1,3,6 |
| 2 | ComposantX |  | 2 |
| … | … | … | ... |

# Diagramme de classes



Commentez le diagramme et en particulier l’utilisation des patterns (si vous en avez utilisé). C'est un diagramme de classe de conception, vous devez y indiquer la multiplicité (par défaut =1) et les noms des rôles. Il ne devrait y avoir plus que des agrégations (simple ou composition), des généralisations (ou implémentation), éventuellement des classe associations des<<use>> ou des <<create>>. Si vous pensez que vous avez besoin d'autres relations, discutez-en avec votre professeur. Faites usage de la notions de «stéréotype» si vous pensez que certaines classes ne seront pas vraiment implémentées comme des classes (au sens habituel du terme). Si vous utilisez des composants proposez des solutions à votre professeur de GL pour les faire apparaître dans votre diagramme de classe.

**ATTENTION :** Les diagrammes de classe doivent être lisibles. Si nécessaire utilisé un format A3 ou éclatez votre diagramme en plusieurs parties.

# Concurrence

## Récapitulatif sur les transactions

Avoir une vue synthétique de toutes les transactions votre application. Les transactions pouvant attaquer la base de données pendant des périodes différentes de temps période, ceci apparaitra dans le tableau. Pour chaque transaction, définir quelles tables sont utilisées ainsi que le mode d’accès, lecture(R) ou écriture (W).

Si vous avez des périodes temporelles dans votre application, qui font qu’elle a des comportements différenciés suivant ces périodes, alors faire un tableau par période.

* Période1 - **avant inscription**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Nom  Transactions | Table1 | … | … | … | Table\_n | Niveau Isolation |
| … | R | R | R |  |  | 1 |
| … | R | W |  |  |  | 0 |
| … | R | R | R/W | R | R | 3 |
| … |  | R |  |  |  |  |

* Période2 - **pendant inscription**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Nom  Transactions | Table1 | … | … | … | Table\_n | Niveau Isolation |
| … | R | R | R |  |  | 1 |
| … | R | W |  |  |  | 0 |
| … | R | R | R/W | R | R | 3 |
| … |  | R |  |  |  |  |

## Présentation d’un scénario concurrentiel

Indiquez un scénario clair avec un problème de concurrence dans votre application. Indiquez clairement :

1. Les noms des transactions en jeu
2. Les problèmes qui risquent d’arriver
3. Expliquez alors clairement comment s’applique la philosophie de la gestion de concurrence que vous avez choisie sur ce cas et avec laquelle vous évitez les problèmes.

# Diagramme de composants

Il s’agit de représenter les composants et les dépendances explicites entre interfaces. ATTENTION : Les interfaces doivent être détaillées et, pour les composants que vous avez développés, il faut indiquer la liste des classes que contient le composant.

# Signatures

Fribourg, le ……………..............

Etudiant1 Etudiant2 Etudiant3