

Télécom SudParis
Année scolaire 2016/2017

Projet Introduction au génie logiciel pour applications orientées objet
CSC 4102

ÉTUDE DE CAS « PROCÈS VERBAL D'ASSEMBLÉE GÉNÉRALE DE COPROPRIÉTÉ »

WANG Yuwei
ZHENG Qi

Enseignant responsable du groupe de TP : **BAC Christian**

16 Novembre 2016

Sommaire

1 Remarques préliminaires.....	3
2 Spécification et préparation des tests de validation.....	4
2.1 Spécification.....	4
2.1.1 Diagrammes de cas d'utilisation.....	4
2.1.2 Priorités des cas d'utilisation.....	5
2.1.3 Scénarios, préconditions, et postconditions.....	5
2.2 Préparation des tests de validation.....	6
3 Conception préliminaire.....	9
3.1 Aspects statiques, liste des classes.....	9
3.2 Aspects statiques, diagramme de classes.....	9
3.3 Aspects dynamique, diagrammes d'interaction (de séquence ou de communications)....	9
4 Conception détaillée et préparation des tests unitaires.....	11
4.1 Diagramme de classes de la conception détaillée.....	11
4.2 Fiche des classes.....	11
4.2.1 Classe AssembleeGenerale.....	11
4.3 Préparation des tests unitaires.....	14
4.3.1 Diagramme de machine à états et invariant de la classe AssembleeGenerale.....	14
4.3.2 Préparation des tests unitaires de la classe AssembleeGenerale.....	14

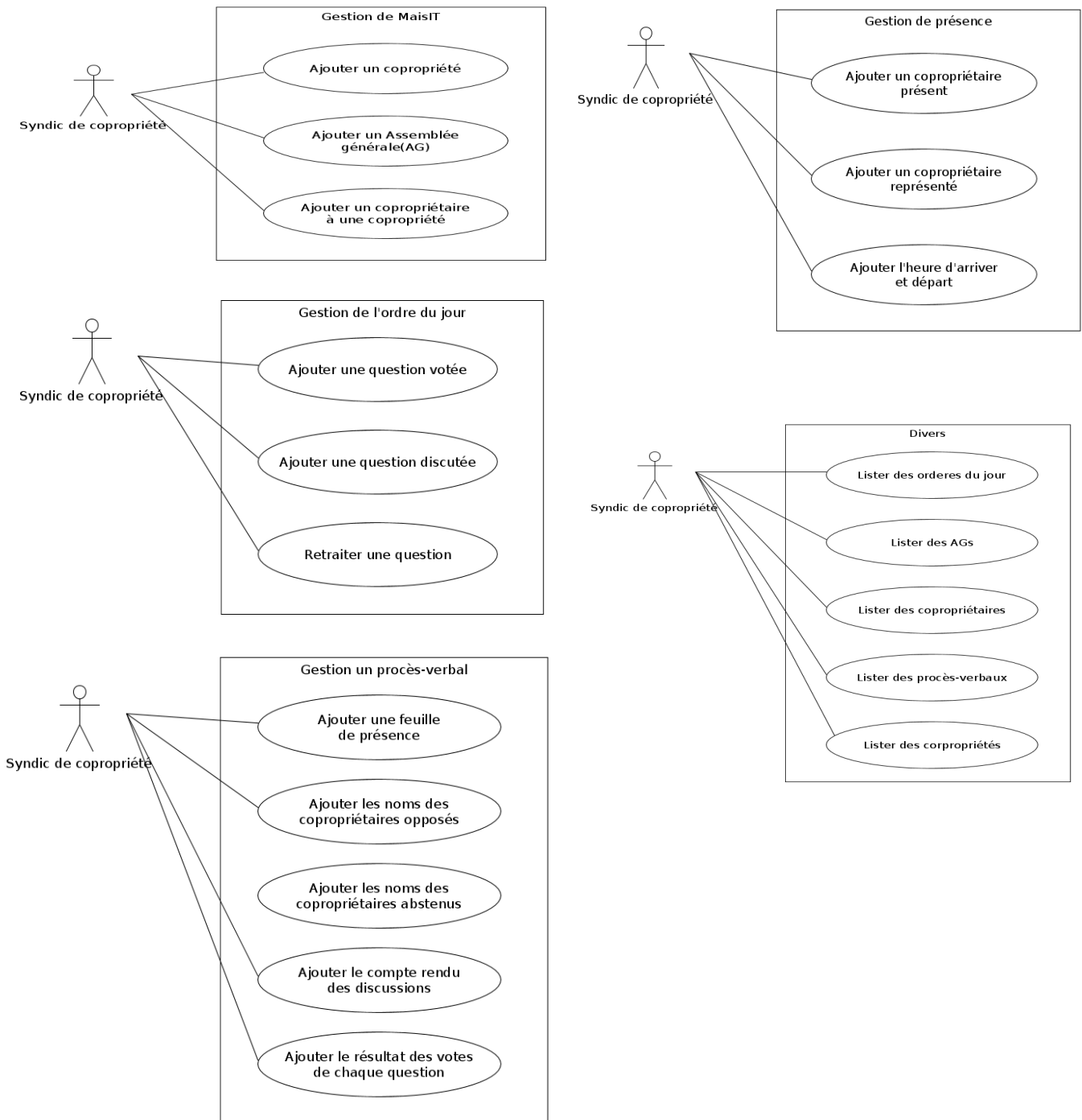
1 Remarques préliminaires

| Vous mettez dans cette section toutes les particularités que vous portez à la connaissance des lecteurs / évaluateurs.

2 Spécification et préparation des tests de validation

2.1 Spécification

2.1.1 Diagrammes de cas d'utilisation



2.1.2 Priorités des cas d'utilisation

Les cas d'utilisation de priorité **Haute** sont :

Ajouter un copropriété, Ajouter un AG, Ajouter un copropriétaire à un AG, Ajouter une question votée, Ajouter une question discutée, Ajouter un copropriétaire présent, Ajouter un copropriétaire représenté, Ajouter l'heure d'arrivée et départ, Ajouter une feuille de présence, Ajouter les noms des copropriétaires opposés, Ajouter les noms des copropriétaires abstenus, Ajouter le compte rendu des discussions, Ajouter le résultat des votes de chaque question.

Les cas d'utilisation de priorité **Moyenne** sont :

lister des ordres du jour, lister des AGs, lister des copropriétaires, lister des procès-verbaux, lister des copropriétés.

Le cas d'utilisation de priorité **Basse** est : Retraiter une question.

2.1.3 Scénarios, préconditions, et postconditions

Les cas d'utilisation choisis pour les préconditions et les postconditions sont :

- f1 : Ajouter une question discutée
- f2 : Ajouter un copropriétaire présent
- f3 : Ajouter un copropriétaire représenté
- f4 : Ajouter les noms des copropriétaires opposés
- f5 : Ajouter le compte rendu des discussions
- f6 : Ajouter le résultat des votes de chaque question

Précondition de f1 : $\text{copropriété existe} \wedge \text{l'assemblée générale existe} \wedge \text{l'ordre du jour existe} \wedge \text{la question n'existe pas}$

Postcondition de f1 : $\text{copropriété existe} \wedge \text{l'assemblée générale existe} \wedge \text{l'ordre du jour existe} \wedge \text{la question existe} \wedge \text{cette question ne peut pas être votée}$

Précondition de f2 : $\text{copropriété existe} \wedge \text{l'assemblée générale existe} \wedge \text{la feuille de présence existe} \wedge \text{ce copropriétaire existe} \wedge \text{cette personne ne s'inscrit pas}$

Postcondition de f2 : $\text{copropriété existe} \wedge \text{l'assemblée générale existe} \wedge \text{la feuille de présence existe} \wedge \text{ce copropriétaire existe} \wedge \text{cette personne s'inscrit dans la feuille de présence}$

Précondition de f3 : $\text{copropriété existe} \wedge \text{l'assemblée générale existe} \wedge \text{la feuille de présence existe} \wedge \text{ce copropriétaire existe} \wedge \text{cette personne ne s'inscrit pas} \wedge \text{son mandataire est unique et ponctuel}$

Postcondition de f3 : $\text{copropriété existe} \wedge \text{l'assemblée générale existe} \wedge \text{la feuille de présence existe} \wedge \text{ce copropriétaire existe} \wedge \text{son mandataire est unique et ponctuel} \wedge \text{cette personne est représentée}$

Précondition de f4 : $\text{corpropriété existe} \wedge \text{l'assemblée générale existe} \wedge \text{ce copropriétaire existe} \wedge \text{cette personne est présente ou représentée} \wedge \text{cette personne s'est opposé lors d'un vote}$

Postcondition de f4 : $\text{corpropriété existe} \wedge \text{l'assemblée générale existe} \wedge \text{ce copropriétaire existe} \wedge \text{cette personne est présente ou représentée} \wedge \text{le nombre de vote opposé incrémenté}$

Précondition de f5 : $\text{corpropriété existe} \wedge \text{l'assemblée générale existe} \wedge \text{l'ordre du jour existe} \wedge \text{le procès-verbal existe} \wedge \text{des questions est discutées} \wedge \text{le compte rendu des discussions n'existe pas}$

Postcondition de f5 : $\text{corpropriété existe} \wedge \text{l'assemblée générale existe} \wedge \text{l'ordre du jour existe} \wedge \text{le procès-verbal existe} \wedge \text{des questions est discutées} \wedge \text{le compte rendu des discussions existe}$

Précondition de f6 : $\text{corpropriété existe} \wedge \text{l'assemblée générale existe} \wedge \text{l'ordre du jour existe} \wedge \text{le procès-verbal existe} \wedge \text{cette question pour voter existe} \wedge \text{cette question est votées}$

Postcondition de f6 : $\text{corpropriété existe} \wedge \text{l'assemblée générale existe} \wedge \text{l'ordre du jour existe} \wedge \text{le procès-verbal existe} \wedge \text{cette question pour voter existe} \wedge \text{cette question est votées} \wedge \text{le résultat des votes est noté}$

2.2 Préparation des tests de validation

Les cas d'utilisation choisis pour construire les tables de décisions des tests de validation sont :

- F1 : Ajouter un copropriétaire présent
- F2 : Ajouter les noms des copropriétaires opposés
- F3 : Ajouter le compte rendu des discussions

F1			1	2	3	4	5	6
Précondition	corpropriété	existe	F	T	T	T	T	T
	l'assemblée générale	existe		F	T	T	T	T
	la feuille de présence	existe			F	T	T	T
	ce copropriétaire	existe				F	T	T
	cette personne	s'inscrit					F	T

Postcondition	Ajouter un copropriétaire présent accepté	F	F	F	F	F	T
	Nombre de jeux de tests	1	1	1	1	1	1

F2			1	2	3	4	5	6
Précondition	copropriété	existe	F	T	T	T	T	T
	l'assemblée générale	existe		F	T	T	T	T
	ce copropriétaire	existe			F	T	T	T
	cette personne	présente ou représentée				F	T	T
	cette personne	s'est opposé lors d'un vote					F	T
Postcondition	Ajouter les noms des copropriétaires opposés accepté		F	F	F	F	F	T
	Nombre de jeux de tests		1	1	1	2	2	2

F3			1	2	3	4	5	6	7
Précondition	copropriété	existe	F	T	T	T	T	T	T
	l'assemblée générale	existe		F	T	T	T	T	T
	l'ordre du jour	existe			F	T	T	T	T

	le procès-verbal	existe				F	T	T	T
	des questions	est discutées					F	T	T
	le compte rendu des discussions	existe						F	T
Postcondition	Ajouter les noms des copropriétaires opposés accepté		F	F	F	F	F	F	T
	Nombre de jeux de tests		1	1	1	1	1	1	1

3 Conception préliminaire

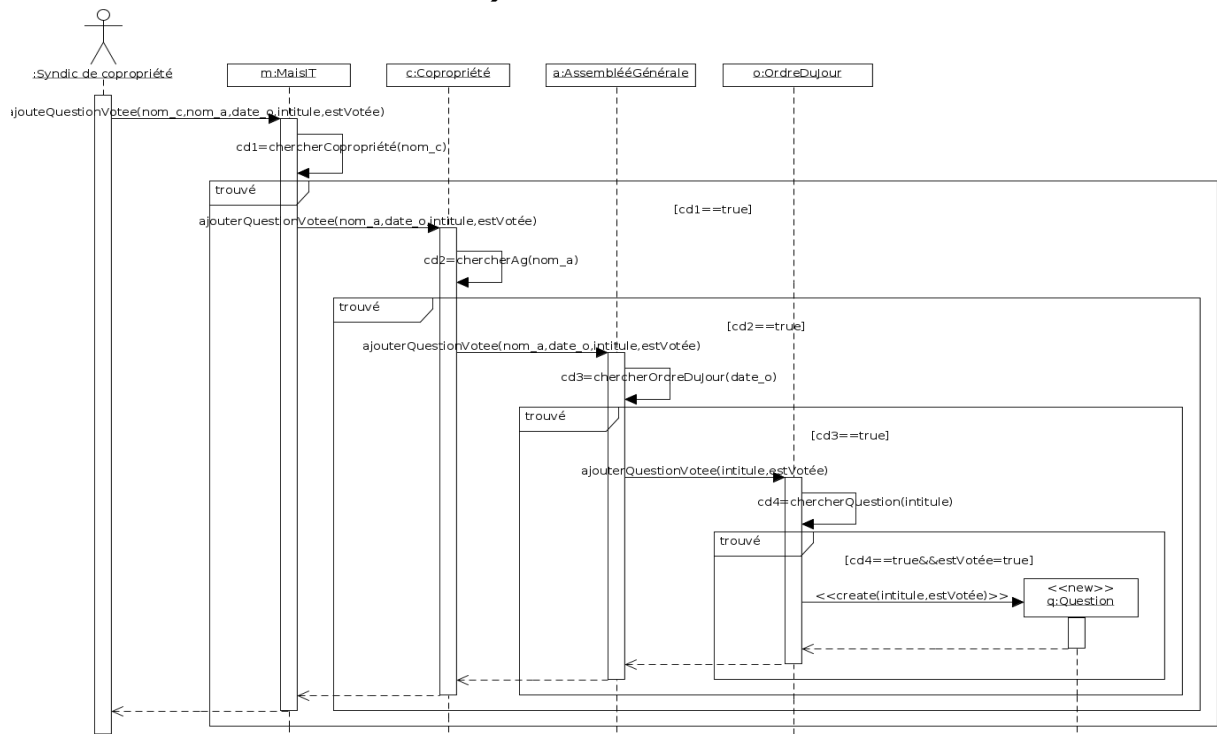
3.1 Aspects statiques, liste des classes

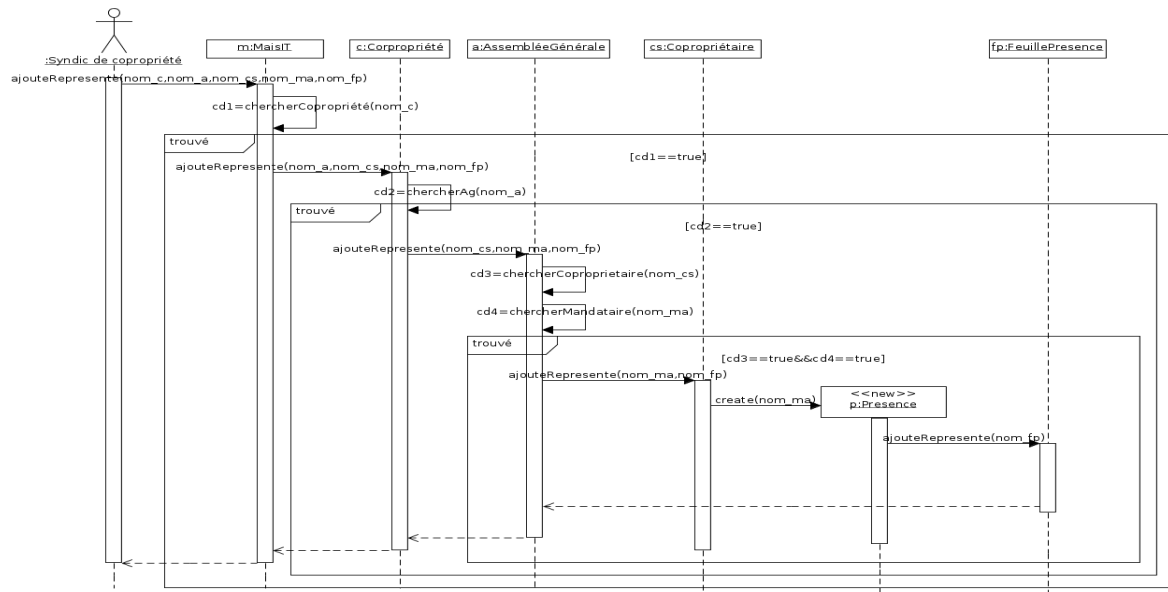
MaisIT, Copropriété, AssembléeGénérale, Copropriétaire, Presence, FeuillePresence, OrdreDuJour, Question, ResultatVote, CompteRendu, ProcesVerbal, TypeVote(enums)

3.2 Aspects statiques, diagramme de classes

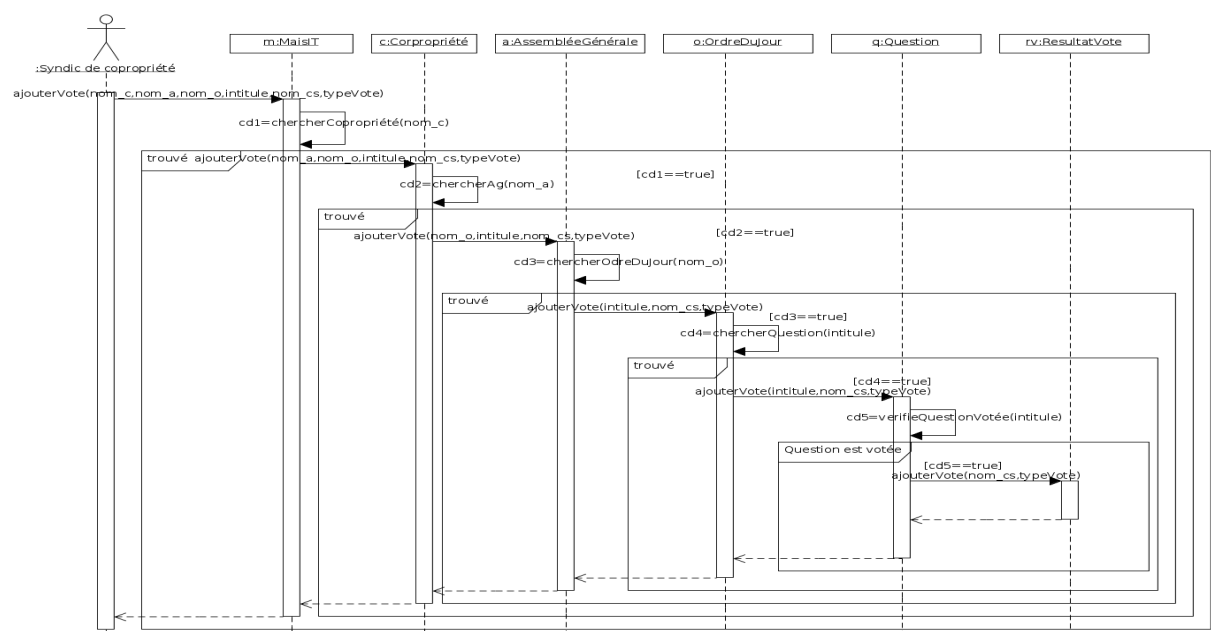
Le diagramme de classes est raffiné dans la section " Diagramme de classes de la conception détaillée " de ce document.

3.3 Aspects dynamique, diagrammes d'interaction (de séquence ou de communications)





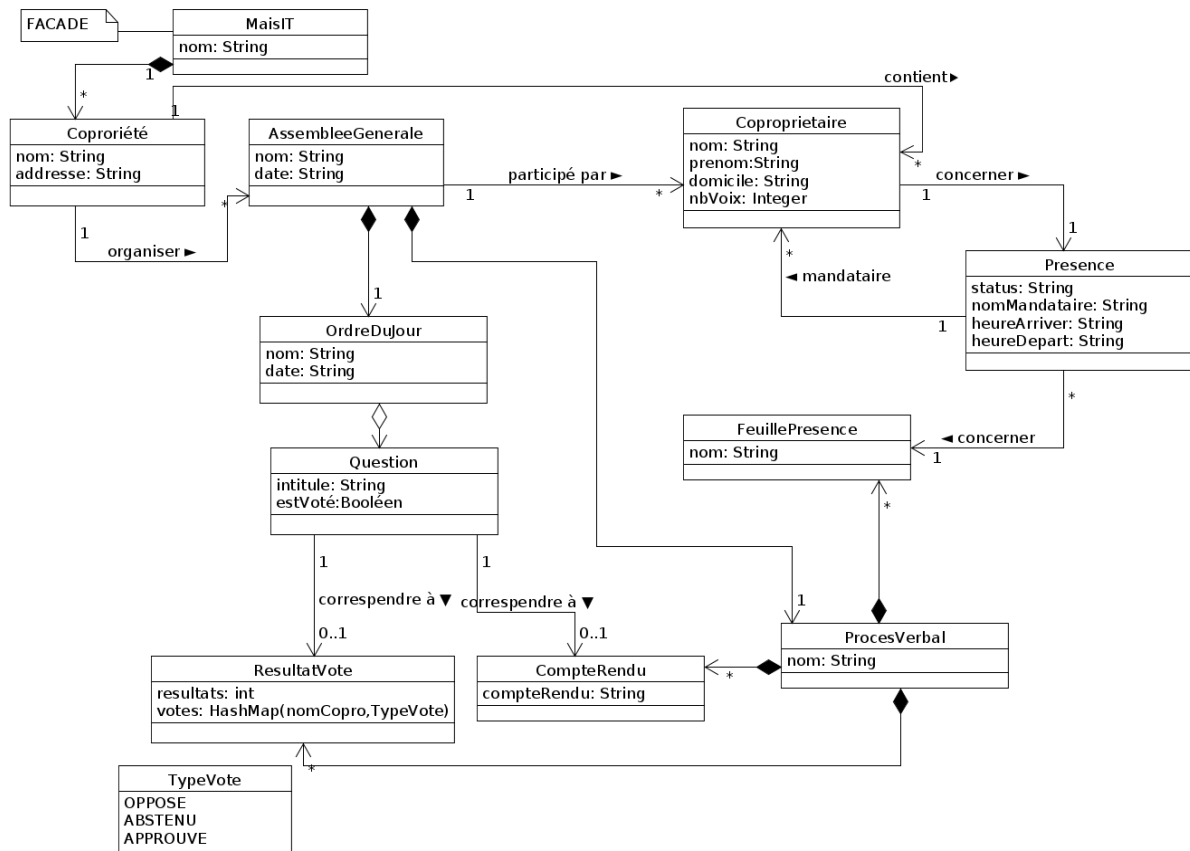
« ajouter un copropriétaire représenté à la feuille de présence d'une assemblée générale d'une copropriété »



« ajouter le vote d'un copropriétaire à une question de l'ordre du jour d'une assemblée générale d'une copropriété »

4 Conception détaillée et préparation des tests unitaires

4.1 Diagramme de classes de la conception détaillée



4.2 Fiche des classes

4.2.1 Classe AssembléeGenerale

AssembléeGenerale
<p>< - attributs - ></p> <ul style="list-style-type: none"> - nom : String - date : String <p>< - attributs « association » - ></p> <ul style="list-style-type: none"> - lesCoproprietaire : Collection de Coproprietaire - ordreDuJour : OrdreDuJour

- procesVerbal : ProcesVerbal

< - **constructeur** - >

+ constructeur(String nom, String date)

+ destructeur()

< - **opérations** - >

+ getNom() : String

+ getDate() : String

< - **opérations « cas d'utilisation »** - >

+ ajouterOrdreDuJour(String nomOrdreDuJour) : Booléen

+ ajouterProcesVerbal(String nomProcesVerbal) : Booléen

+ ajouterQuestionVotee(String intitule, Boolean estVotée) : Booléen

+ ajouteRepresente(String nomMandataire, String nomFeuillePresence): Booléen

+ ajouterVote(String intitule, String nomCoproprietaire, Boolean typeVote): Booléen

+ supprimerOrdreDuJour(String nomOrdreDuJour) : Booléen

+ supprimerProcesVerbal(String nomProcesVerbal) : Booléen

< - **opérations pour les nouveaux cas d'utilisation** - >

+ listerCoproprietaires()

+ listerOrdreDuJour()

+ listerProcesVerbal()

< - **opérations de recherche** - >

+ chercherCoproprietaire(String nom) : Booléen

+ chercherOrdreDuJour(String nom) : Booléen

+ chercherProcesVerbal(String nom) : Booléen

```
+ chercherMandataire(String nomMandataire) :Booléen
```

```
constructeur(String nom, String date)
```

```
    nom = this.nom
```

```
    date = this.date
```

```
    lesCoproprietaire = construction d'une collection vide de Coproprietaire
```

```
    ordreDuJour = construction d'un vide OrdreDuJour
```

```
    procesVerbal = construction d'un vide ProcesVerbal
```

```
void destructeur()
```

```
    pour tout coproprietaires dans lesCorproprietaires faire coproprietaire.destructeur()
```

```
    pour ordredujour faire ordreDuJour.destructeur()
```

```
    pour procesverbal faire procesVerbal.destructeur()
```

```
String getNom()
```

```
    retourner this.nom
```

```
String getDate()
```

```
    retourner this.date
```

```
Boolean chercherCoproprietaire(String nom)
```

```
for c in lesCoproprietaires do
```

```
    si nom de c est nom alors retourner c
```

```
retourner null
```

Boolean ajouterQuestionVotee(String nomOrdreDuJour, String intitule, Boolean estVotee)

Question q = chercherQuestion(intitule)

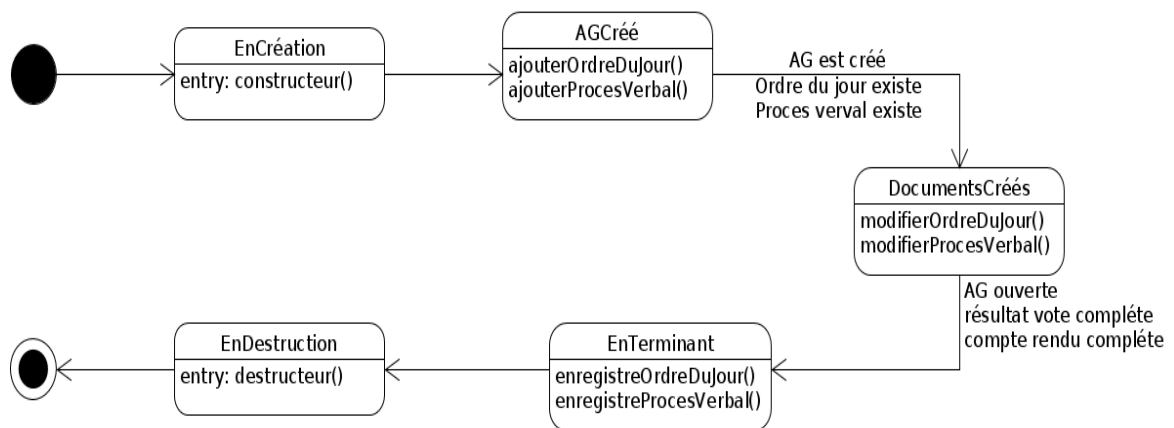
si(q == null)

q = constructeurQuestion(intitule, estVotee)

ajouter q à la collection lesQuestion

4.3 Préparation des tests unitaires

4.3.1 Diagramme de machine à états et invariant de la classe AssembléeGenerale



Invariant de l'assemblée générale:

$$(\neg \text{AGCréé} \wedge \neg \text{DocumentCréés} \wedge \neg \text{EnTerminant}) \vee (\text{AGCréé} \wedge \neg \text{DocumentCréés} \wedge \neg \text{EnTerminant}) \vee (\text{AGCréé} \wedge \text{DocumentCréés} \wedge \neg \text{EnTerminant}) \vee (\text{AGCréé} \wedge \text{DocumentCréés} \wedge \text{EnTerminant})$$

$$= (\neg \text{DocumentCréés} \wedge \neg \text{EnTerminant}) \vee (\text{AGCréé} \wedge \text{DocumentCréés})$$

$$= \neg (\text{DocumentCréés} \vee \text{EnTerminant}) \vee (\text{AGCréé} \wedge \text{DocumentCréés})$$

$$= (\text{DocumentCréés} \vee \text{EnTerminant}) \Rightarrow (\text{AGCréé} \wedge \text{DocumentCréés})$$

4.3.2 Préparation des tests unitaires de la classe `AssembleeGenerale`

Table de décision des tests unitaires de « `ajouterOrdreDuJour(String nomOrdreDuJour)` : Booléen »

		1	2	3
Précondition	AGCrée	F	T	T
	DocumentCrée		F	T
Postcondition	\negordreDuJour = ordreDuJour	F	T	F
Exception	Levée d'une exception	Oui	Non	Oui
Effet	Ordre du jour créé	F	T	F
	Nombre de jeux de tests	1	1	1

Table de décision des tests unitaires de « `ajouterProcesVerbal(String nomProcesVerbal)` : Booléen »

		1	2	3
Précondition	AGCrée	F	T	T
	DocumentCrée		F	T
Postcondition	\negprocesVerbal = procesVerbal	F	T	F
Exception	Levée d'une exception	Oui	Non	Oui

Effet	ProcesVerbal créé	F	T	F
	Nombre de jeux de tests	1	1	1