

<b>Rapport</b>
----------------

simo-  
19/06/2020

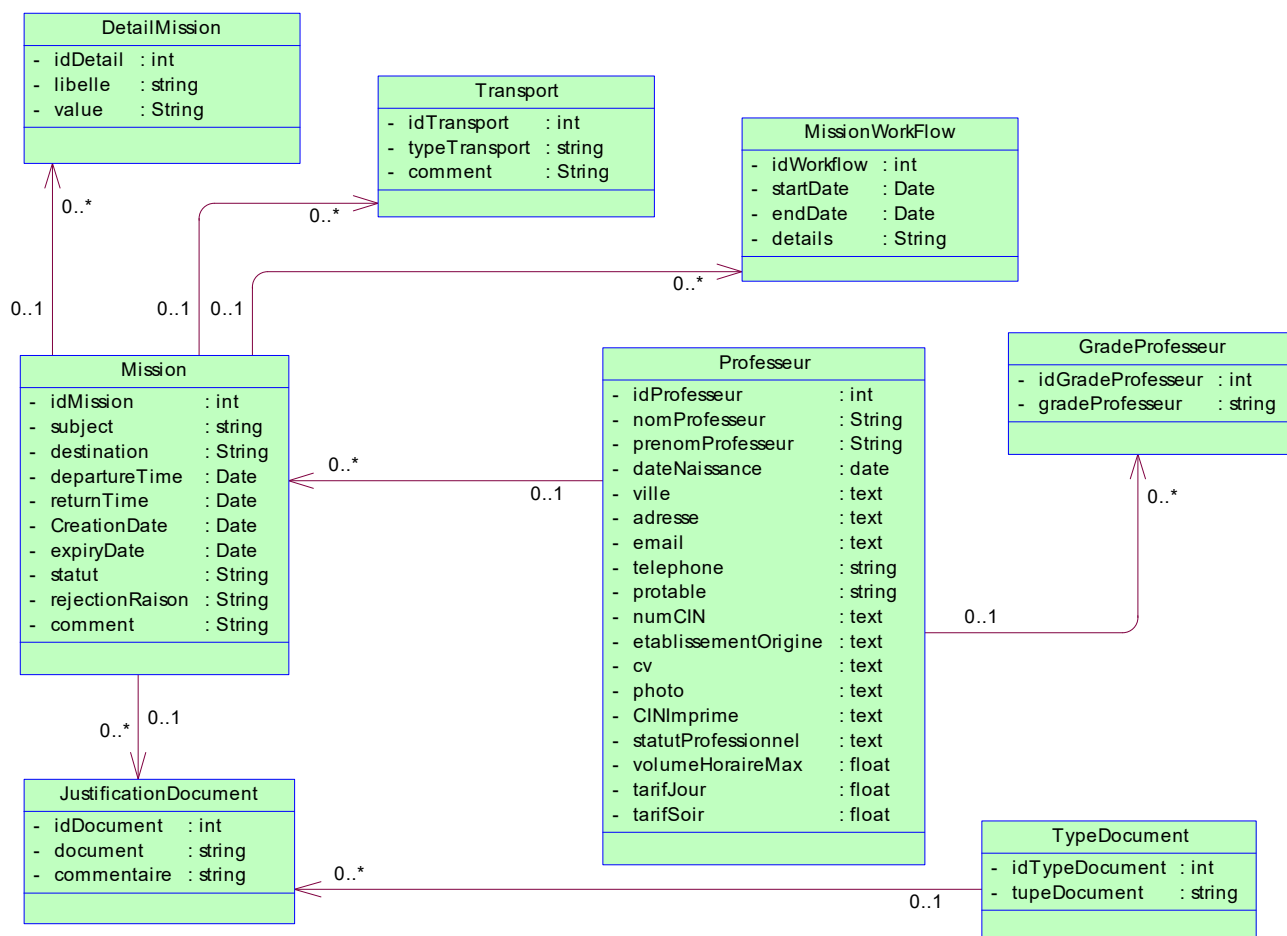
**Sommaire**

1 Diagrammes MOO.....	3
1.1 Diagrammes au niveau du modèle.....	3
1.1.1 Diagramme DiagrammeClasses_1.....	3
2 Listes des objets au niveau du modèle.....	4
2.1 Objets communs.....	4
2.1.1 Liste des associations.....	4
2.1.2 Liste des attributs étendus du modèle Diagramme.....	4
2.1.3 Liste des diagrammes.....	4
2.2 Objets des diagrammes de classe.....	4
2.2.1 Liste des classes.....	4
2.2.2 Liste des attributs de classe.....	5
3 Objets principaux des diagrammes de classe.....	8
3.1 Objets des diagrammes au niveau du modèle.....	8
3.1.1 Classe DetailMission.....	8
3.1.2 Classe GradeProfesseur.....	8
3.1.3 Classe JustificationDocument.....	9
3.1.4 Classe Mission.....	9
3.1.5 Classe MissionWorkFlow.....	14
3.1.6 Classe Professeur.....	14
3.1.7 Classe Transport.....	18
3.1.8 Classe TypeDocument.....	18

## 1 Diagrammes MOO

### 1.1 Diagrammes au niveau du modèle

#### 1.1.1 Diagramme DiagrammeClasses 1



## 2 Listes des objets au niveau du modèle

### 2.1 Objets communs

#### 2.1.1 Liste des associations

<i>Nom</i>	<i>Code</i>	<i>Classe B</i>	<i>Classe A</i>	<i>Rôle A</i>	<i>Rôle B</i>	<i>Multipli cité A</i>	<i>Multipli cité B</i>
Association_1	association 1	GradeProf esseur	Professeur			0..1	0..*
Association_2	association 2	Transport	Mission			0..1	0..*
Association_3	association 3	Justificatio nDocumen t	Mission			0..1	0..*
Association_4	association 4	Justificatio nDocumen t	TypeDocu ment			0..1	0..*
Association_5	association 5	Mission	Professeur			0..1	0..*
Association_6	association 6	DetailMiss ion	Mission			0..1	0..*
Association_7	association 7	MissionW orkFlow	Mission			0..1	0..*

#### 2.1.2 Liste des attributs étendus du modèle Diagramme

<i>Nom</i>	<i>Type de données</i>	<i>Valeur</i>	<i>Nom de la cible</i>
JarFileName	(Chaîne)	%Code%.jar	Java
MessageDestinations	(Texte)		Java
SchemaNamespace	(Chaîne)	http://www.w3.org/2001/XMLSchema	WSDL for Java
SoapBindingTransport	(Chaîne)	http://schemas.xmlsoap.org/soap/http	WSDL for Java
SoapBodyNamespace	(Chaîne)		WSDL for Java

#### 2.1.3 Liste des diagrammes

<i>Nom</i>	<i>Code</i>
DiagrammeClasses_1	DiagrammeClasses_1

### 2.2 Objets des diagrammes de classe

#### 2.2.1 Liste des classes

<i>Nom</i>	<i>Code</i>	<i>Parent</i>	<i>Génère r</i>	<i>Visibilit é</i>	<i>Abstrait</i>	<i>Type de classific ateur</i>
DetailMission	DetailMission	Modèle orienté objet 'Diagramme'	X	public		Classe
GradeProfesseur	GradeProfesseur	Modèle orienté objet 'Diagramme'	X	public		Classe
JustificationDocum ent	JustificationDocum ent	Modèle orienté objet 'Diagramme'	X	public		Classe
Mission	Mission	Modèle orienté objet 'Diagramme'	X	public		Classe

<i>Nom</i>	<i>Code</i>	<i>Parent</i>	<i>Générér</i>	<i>Visibilit é</i>	<i>Abstrait</i>	<i>Type de classific ateur</i>
MissionWorkFlow	MissionWorkFlow	Modèle orienté objet 'Diagramme'	X	public		Classe
Professeur	Professeur	Modèle orienté objet 'Diagramme'	X	public		Classe
Transport	Transport	Modèle orienté objet 'Diagramme'	X	public		Classe
TypeDocument	TypeDocument	Modèle orienté objet 'Diagramme'	X	public		Classe

### 2.2.2 Liste des attributs de classe

<i>Nom</i>	<i>Code</i>	<i>Affiché</i>	<i>Domaine</i>	<i>Type de donnée s</i>	<i>Visibilit é</i>	<i>Valeur initiale</i>	<i>Classific ateur</i>	<i>Lecture -seule</i>
idProfesseur	idProfesseur	X		int	private		Professeur	
nomProfesseur	nomProfesseur	X		String	private		Professeur	
prenomProfesseur	prenomProfesseur	X		String	private		Professeur	
dateNaissance	dateNaissance	X		date	private		Professeur	
ville	ville	X		text	private		Professeur	
adresse	adresse	X		text	private		Professeur	
email	email	X		text	private		Professeur	
telephone	telephone	X		string	private		Professeur	
protable	protable	X		string	private		Professeur	
numCIN	numCIN	X		text	private		Professeur	
etablissementOrigine	etablissementOrigine	X		text	private		Professeur	
cv	cv	X		text	private		Professeur	
photo	photo	X		text	private		Professeur	
CINImprime	cINImprime	X		text	private		Professeur	
statutProfessionnel	statutProfessionnel	X		text	private		Professeur	
volumeHoraireMax	volumeHoraireMax	X		float	private		Professeur	
tarifJour	tarifJour	X		float	private		Professeur	
tarifSoir	tarifSoir	X		float	private		Professeur	

<i>Nom</i>	<i>Code</i>	<i>Affiché</i>	<i>Domaine</i>	<i>Type de donnée</i> <i>s</i>	<i>Visibilit</i> <i>é</i>	<i>Valeur</i> <i>initiale</i>	<i>Classific</i> <i>ateur</i>	<i>Lecture</i> <i>-seule</i>
idGradeP rofesseur	idGradeP rofesseur	X		int	private		GradePro fesseur	
gradePro fesseur	gradePro fesseur	X		string	private		GradePro fesseur	
idDocum ent	idDocum ent	X		int	private		Justificati onDocu ment	
documen t	documen t	X		string	private		Justificati onDocu ment	
comment aire	comment aire	X		string	private		Justificati onDocu ment	
idTypeD ocument	idTypeD ocument	X		int	private		TypeDoc ument	
tupeDocu ment	tupeDocu ment	X		string	private		TypeDoc ument	
idMissio n	idMissio n	X		int	private		Mission	
subject	subject	X		string	private		Mission	
destinatio n	destinatio n	X		String	private		Mission	
departure Time	departure Time	X		Date	private		Mission	
returnTi me	returnTi me	X		Date	private		Mission	
Creation Date	creation Date	X		Date	private		Mission	
expiryDa te	expiryDa te	X		Date	private		Mission	
statut	statut	X		String	private		Mission	
rejection Raison	rejection Raison	X		String	private		Mission	
comment	comment	X		String	private		Mission	
idTransp ort	idTransp ort	X		int	private		Transpor t	
typeTran sport	typeTran sport	X		string	private		Transpor t	
comment	comment	X		String	private		Transpor t	
idDetail	idDetail	X		int	private		DetailMi ssion	
libelle	libelle	X		string	private		DetailMi ssion	
value	value	X		String	private		DetailMi ssion	
idWorkfl ow	idWorkfl ow	X		int	private		Mission WorkFlo w	

<i>Nom</i>	<i>Code</i>	<i>Affiché</i>	<i>Domaine</i>	<i>Type de donnée s</i>	<i>Visibilit é</i>	<i>Valeur initiale</i>	<i>Classific ateur</i>	<i>Lecture -seule</i>
startDate	startDate	X		Date	private		Mission WorkFlo w	
endDate	endDate	X		Date	private		Mission WorkFlo w	
details	details	X		String	private		Mission WorkFlo w	

### **3 Objets principaux des diagrammes de classe**

#### **3.1 Objets des diagrammes au niveau du modèle**

##### **3.1.1 Classe DetailMission**

###### **3.1.1.3 Aperçu du code de la classe DetailMission**

```

/*****
* Module: DetailMission.java
* Author: simo-
* Purpose: Defines the Class DetailMission
*****/

```

```

import java.util.*;

/** @pdOid fabde4a0-3664-4bc7-a74a-1547e4ca8e6c */
public class DetailMission {
    /** @pdOid d41b393f-b57d-41ff-9e28-7fb4c2961312 */
    private int idDetail;
    /** @pdOid 2aa48921-8a63-4e48-9059-63b5090f37cb */
    private string libelle;
    /** @pdOid 14016bd0-dec9-43e1-8dcf-b38c68681594 */
    private String value;
}

```

###### **3.1.1.4 Début de la classe DetailMission**

```

/*****
* Module: DetailMission.java
* Author: simo-
* Purpose: Defines the Class DetailMission
*****/

```

###### **3.1.1.5 Importations de la classe DetailMission**

```

import java.util.*;

```

###### **3.1.1.10 Attribut idDetail de la classe DetailMission**

###### **3.1.1.11 Attribut libelle de la classe DetailMission**

###### **3.1.1.12 Attribut value de la classe DetailMission**

##### **3.1.2 Classe GradeProfesseur**

###### **3.1.2.3 Aperçu du code de la classe GradeProfesseur**

```

/*****
* Module: GradeProfesseur.java
* Author: simo-
* Purpose: Defines the Class GradeProfesseur
*****/

```

```

import java.util.*;

/** @pdOid feb41eb5-58d5-4a8c-8365-7d7ba4eded2f */
public class GradeProfesseur {
    /** @pdOid 3db347cb-83c6-4fef-9393-545e1e87ee16 */
    private int idGradeProfesseur;
    /** @pdOid b33ccb74-b257-48bc-a7b6-4a863116cdc5 */
    private string gradeProfesseur;
}

```



**3.1.2.4 Début de la classe GradeProfesseur**

```

/*****
* Module: GradeProfesseur.java
* Author: simo-
* Purpose: Defines the Class GradeProfesseur
*****/

```

**3.1.2.5 Importations de la classe GradeProfesseur**

```
import java.util.*;
```

**3.1.2.10 Attribut idGradeProfesseur de la classe GradeProfesseur****3.1.2.11 Attribut gradeProfesseur de la classe GradeProfesseur****3.1.3 Classe JustificationDocument****3.1.3.3 Aperçu du code de la classe JustificationDocument**

```

/*****
* Module: JustificationDocument.java
* Author: simo-
* Purpose: Defines the Class JustificationDocument
*****/

```

```
import java.util.*;
```

```

/** @pdOid d588960d-9bed-469e-bead-7bd4fce2903e */
public class JustificationDocument {
    /** @pdOid 8d9fd793-8b69-4beb-972b-5390270dd64f */
    private int idDocument;
    /** @pdOid 88c874f9-29a2-4b52-8bfc-6131c69fe713 */
    private string document;
    /** @pdOid 4fdd0bc4-399d-47a6-852c-8d8aa3701df1 */
    private string commentaire;
}

```

**3.1.3.4 Début de la classe JustificationDocument**

```

/*****
* Module: JustificationDocument.java
* Author: simo-
* Purpose: Defines the Class JustificationDocument
*****/

```

**3.1.3.5 Importations de la classe JustificationDocument**

```
import java.util.*;
```

**3.1.3.10 Attribut idDocument de la classe JustificationDocument****3.1.3.11 Attribut document de la classe JustificationDocument****3.1.3.12 Attribut commentaire de la classe JustificationDocument****3.1.4 Classe Mission****3.1.4.3 Aperçu du code de la classe Mission**

```

/*****
* Module: Mission.java
* Author: simo-
* Purpose: Defines the Class Mission
*****/

```

```

import java.util.*;

/** @pdOid 14dd9b60-d905-4e22-96d5-b21fc9e6aa3c */
public class Mission {
    /** @pdOid 205a121c-064d-42cc-8f32-670d4fbfb61a */
    private int idMission;
    /** @pdOid d7b497a1-6abb-4210-808d-aa9cd5953ab8 */
    private String subject;
    /** @pdOid 438a3fc9-18a6-40cd-ac5e-7c799cf9702a */
    private String destination;
    /** @pdOid d5ffe298-f0d0-47c6-9f82-e0dd78e34b62 */
    private Date departureTime;
    /** @pdOid b6e6633c-512f-4113-a6ea-06ab0693e3fb */
    private Date returnTime;
    /** @pdOid 87ed6ce9-55ab-41d7-8e14-dd123452e637 */
    private Date creationDate;
    /** @pdOid 9ec1b284-f8d9-40a1-bb2b-4bd60d4e5d97 */
    private Date expiryDate;
    /** @pdOid db3b89f3-f08d-4a11-aacf-68f8f0003ce1 */
    private String statut;
    /** @pdOid 8db7d75c-5239-4dde-abc2-8a8dfce572b2 */
    private String rejectionRaison;
    /** @pdOid b01fdd6d-56d8-4729-820f-bf540e6a631f */
    private String comment;

    /** @pdRoleInfo migr=no name=Transport assc=association2 coll=java.util.Collection impl=java.util.HashSet mult=0..* */
    public java.util.Collection<Transport> transport;
    /** @pdRoleInfo migr=no name=JustificationDocument assc=association3 coll=java.util.Collection impl=java.util.HashSet mult=0..* */
    public java.util.Collection<JustificationDocument> justificationDocument;
    /** @pdRoleInfo migr=no name=DetailMission assc=association6 coll=java.util.Collection impl=java.util.HashSet mult=0..* */
    public java.util.Collection<DetailMission> detailMission;
    /** @pdRoleInfo migr=no name=MissionWorkFlow assc=association7 coll=java.util.Collection impl=java.util.HashSet mult=0..* */
    public java.util.Collection<MissionWorkFlow> missionWorkFlow;

    /** @pdGenerated default getter */
    public java.util.Collection<Transport> getTransport() {
        if (transport == null)
            transport = new java.util.HashSet<Transport>();
        return transport;
    }

    /** @pdGenerated default iterator getter */
    public java.util.Iterator getIteratorTransport() {
        if (transport == null)
            transport = new java.util.HashSet<Transport>();
        return transport.iterator();
    }

    /** @pdGenerated default setter
     * @param newTransport */
    public void setTransport(java.util.Collection<Transport> newTransport) {
        removeAllTransport();
        for (java.util.Iterator iter = newTransport.iterator(); iter.hasNext();)
            addTransport((Transport)iter.next());
    }

    /** @pdGenerated default add
     * @param newTransport */
    public void addTransport(Transport newTransport) {
        if (newTransport == null)

```

```

    return;
    if (this.transport == null)
        this.transport = new java.util.HashSet<Transport>();
    if (!this.transport.contains(newTransport))
        this.transport.add(newTransport);
}

/** @pdGenerated default remove
 * @param oldTransport */
public void removeTransport(Transport oldTransport) {
    if (oldTransport == null)
        return;
    if (this.transport != null)
        if (this.transport.contains(oldTransport))
            this.transport.remove(oldTransport);
}

/** @pdGenerated default removeAll */
public void removeAllTransport() {
    if (transport != null)
        transport.clear();
}

/** @pdGenerated default getter */
public java.util.Collection<JustificationDocument> getJustificationDocument() {
    if (justificationDocument == null)
        justificationDocument = new java.util.HashSet<JustificationDocument>();
    return justificationDocument;
}

/** @pdGenerated default iterator getter */
public java.util.Iterator getIteratorJustificationDocument() {
    if (justificationDocument == null)
        justificationDocument = new java.util.HashSet<JustificationDocument>();
    return justificationDocument.iterator();
}

/** @pdGenerated default setter
 * @param newJustificationDocument */
public void setJustificationDocument(java.util.Collection<JustificationDocument> newJustificationDocument) {
    removeAllJustificationDocument();
    for (java.util.Iterator iter = newJustificationDocument.iterator(); iter.hasNext();)
        addJustificationDocument((JustificationDocument)iter.next());
}

/** @pdGenerated default add
 * @param newJustificationDocument */
public void addJustificationDocument(JustificationDocument newJustificationDocument) {
    if (newJustificationDocument == null)
        return;
    if (this.justificationDocument == null)
        this.justificationDocument = new java.util.HashSet<JustificationDocument>();
    if (!this.justificationDocument.contains(newJustificationDocument))
        this.justificationDocument.add(newJustificationDocument);
}

/** @pdGenerated default remove
 * @param oldJustificationDocument */
public void removeJustificationDocument(JustificationDocument oldJustificationDocument) {
    if (oldJustificationDocument == null)
        return;
    if (this.justificationDocument != null)
        if (this.justificationDocument.contains(oldJustificationDocument))
            this.justificationDocument.remove(oldJustificationDocument);
}

```

```

}

/** @pdGenerated default removeAll */
public void removeAllJustificationDocument() {
    if (justificationDocument != null)
        justificationDocument.clear();
}

/** @pdGenerated default getter */
public java.util.Collection<DetailMission> getDetailMission() {
    if (detailMission == null)
        detailMission = new java.util.HashSet<DetailMission>();
    return detailMission;
}

/** @pdGenerated default iterator getter */
public java.util.Iterator getIteratorDetailMission() {
    if (detailMission == null)
        detailMission = new java.util.HashSet<DetailMission>();
    return detailMission.iterator();
}

/** @pdGenerated default setter
 * @param newDetailMission */
public void setDetailMission(java.util.Collection<DetailMission> newDetailMission) {
    removeAllDetailMission();
    for (java.util.Iterator iter = newDetailMission.iterator(); iter.hasNext(); )
        addDetailMission((DetailMission)iter.next());
}

/** @pdGenerated default add
 * @param newDetailMission */
public void addDetailMission(DetailMission newDetailMission) {
    if (newDetailMission == null)
        return;
    if (this.detailMission == null)
        this.detailMission = new java.util.HashSet<DetailMission>();
    if (!this.detailMission.contains(newDetailMission))
        this.detailMission.add(newDetailMission);
}

/** @pdGenerated default remove
 * @param oldDetailMission */
public void removeDetailMission(DetailMission oldDetailMission) {
    if (oldDetailMission == null)
        return;
    if (this.detailMission != null)
        if (this.detailMission.contains(oldDetailMission))
            this.detailMission.remove(oldDetailMission);
}

/** @pdGenerated default removeAll */
public void removeAllDetailMission() {
    if (detailMission != null)
        detailMission.clear();
}

/** @pdGenerated default getter */
public java.util.Collection<MissionWorkFlow> getMissionWorkFlow() {
    if (missionWorkFlow == null)
        missionWorkFlow = new java.util.HashSet<MissionWorkFlow>();
    return missionWorkFlow;
}

/** @pdGenerated default iterator getter */

```

```

public java.util.Iterator getIteratorMissionWorkFlow() {
    if (missionWorkFlow == null)
        missionWorkFlow = new java.util.HashSet<MissionWorkFlow>();
    return missionWorkFlow.iterator();
}

/** @pdGenerated default setter
 * @param newMissionWorkFlow */
public void setMissionWorkFlow(java.util.Collection<MissionWorkFlow> newMissionWorkFlow) {
    removeAllMissionWorkFlow();
    for (java.util.Iterator iter = newMissionWorkFlow.iterator(); iter.hasNext();
        addMissionWorkFlow((MissionWorkFlow)iter.next());
    }

/** @pdGenerated default add
 * @param newMissionWorkFlow */
public void addMissionWorkFlow(MissionWorkFlow newMissionWorkFlow) {
    if (newMissionWorkFlow == null)
        return;
    if (this.missionWorkFlow == null)
        this.missionWorkFlow = new java.util.HashSet<MissionWorkFlow>();
    if (!this.missionWorkFlow.contains(newMissionWorkFlow))
        this.missionWorkFlow.add(newMissionWorkFlow);
}

/** @pdGenerated default remove
 * @param oldMissionWorkFlow */
public void removeMissionWorkFlow(MissionWorkFlow oldMissionWorkFlow) {
    if (oldMissionWorkFlow == null)
        return;
    if (this.missionWorkFlow != null)
        if (this.missionWorkFlow.contains(oldMissionWorkFlow))
            this.missionWorkFlow.remove(oldMissionWorkFlow);
}

/** @pdGenerated default removeAll */
public void removeAllMissionWorkFlow() {
    if (missionWorkFlow != null)
        missionWorkFlow.clear();
}
}

```

#### **3.1.4.4 Début de la classe Mission**

```

/*****
 * Module: Mission.java
 * Author: simo-
 * Purpose: Defines the Class Mission
 *****/

```

#### **3.1.4.5 Importations de la classe Mission**

```
import java.util.*;
```

#### **3.1.4.10 Attribut idMission de la classe Mission**

#### **3.1.4.11 Attribut subject de la classe Mission**

#### **3.1.4.12 Attribut destination de la classe Mission**

#### **3.1.4.13 Attribut departureTime de la classe Mission**

#### **3.1.4.14 Attribut returnTime de la classe Mission**

**3.1.4.15 Attribut CreationDate de la classe Mission****3.1.4.16 Attribut expiryDate de la classe Mission****3.1.4.17 Attribut statut de la classe Mission****3.1.4.18 Attribut rejectionRaison de la classe Mission****3.1.4.19 Attribut comment de la classe Mission****3.1.5 Classe MissionWorkflow****3.1.5.3 Aperçu du code de la classe MissionWorkflow**

```

/*****
 * Module: MissionWorkflow.java
 * Author: simo-
 * Purpose: Defines the Class MissionWorkflow
 *****/

import java.util.*;

/** @pdOid e1fc50f0-2d14-4f7b-b3a8-488cd1d940dc */
public class MissionWorkflow {
    /** @pdOid 901b625a-14be-4659-ba0b-bf67e683e3f2 */
    private int idWorkflow;
    /** @pdOid d739850c-7827-4b96-8c78-053a26c49215 */
    private Date startDate;
    /** @pdOid 338a78d1-3dcb-4256-839b-b6e66e9c6e28 */
    private Date endDate;
    /** @pdOid b24ce404-df46-485f-8d97-ac832eb37b63 */
    private String details;
}

```

**3.1.5.4 Début de la classe MissionWorkflow**

```

/*****
 * Module: MissionWorkflow.java
 * Author: simo-
 * Purpose: Defines the Class MissionWorkflow
 *****/

```

**3.1.5.5 Importations de la classe MissionWorkflow**

```
import java.util.*;
```

**3.1.5.10 Attribut idWorkflow de la classe MissionWorkflow****3.1.5.11 Attribut startDate de la classe MissionWorkflow****3.1.5.12 Attribut endDate de la classe MissionWorkflow****3.1.5.13 Attribut details de la classe MissionWorkflow****3.1.6 Classe Professeur****3.1.6.3 Aperçu du code de la classe Professeur**

```

/*****
 * Module: Professeur.java
 * Author: simo-
 * Purpose: Defines the Class Professeur
 *****/

```

\*\*\*\*\*/

```
import java.util.*;

/** @pdOid 493c62f7-8bfd-4854-a07d-9929af9447d0 */
public class Professeur {
    /** @pdOid fc9f850e-980d-45c4-bb62-3815a8b41cb8 */
    private int idProfesseur;
    /** @pdOid f31d3857-f160-42f8-bc06-a57438143cf2 */
    private String nomProfesseur;
    /** @pdOid 89f3215a-24f4-4d39-aa73-0b732aed45ed */
    private String prenomProfesseur;
    /** @pdOid 5afea36a-f8dc-43a4-9494-4fb3e3f34ffd */
    private date dateNaissance;
    /** @pdOid 85f5ceac-cdc9-4cd9-b657-250bacd3ce96 */
    private text ville;
    /** @pdOid 9d5d32b6-ff9e-44ad-bcc5-3b2c1e466972 */
    private text adresse;
    /** @pdOid 7f2cdb14-e82b-45e8-9e79-3f1bdc85848 */
    private text email;
    /** @pdOid 1bd5b573-996f-403c-a27d-34f725c0bc6f */
    private string telephone;
    /** @pdOid 2c5a3f8f-75fa-4643-a40e-e215c8254348 */
    private string protable;
    /** @pdOid 41be3d1e-e8d6-4dd8-b8b1-93abed292e07 */
    private text numCIN;
    /** @pdOid 6709e41c-958d-436b-b9a7-6801f811449d */
    private text etablissementOrigine;
    /** @pdOid 945c85fc-8771-4388-9d06-d5af5b5efa8f */
    private text cv;
    /** @pdOid 722aec27-add6-4d92-8e62-983f3789e0fe */
    private text photo;
    /** @pdOid 40aecb42-0a6d-4fbc-ae79-9294d7868248 */
    private text cINImprime;
    /** @pdOid f54ab524-8529-4eaa-9afd-ba89e041c52c */
    private text statutProfessionnel;
    /** @pdOid c3cd68ec-b18b-4d55-8661-6772459e244f */
    private float volumeHoraireMax;
    /** @pdOid efa3c52a-d18d-403a-b9b6-2d8ba6bc22d5 */
    private float tarifJour;
    /** @pdOid 0903636e-3e77-4efd-99c2-03517f5d66c9 */
    private float tarifSoir;

    /** @pdRoleInfo migr=no name=GradeProfesseur assc=association1 coll=java.util.Collection impl=java.util.HashSet mult=0..* */
    public java.util.Collection<GradeProfesseur> gradeProfesseur;
    /** @pdRoleInfo migr=no name=Mission assc=association5 coll=java.util.Collection impl=java.util.HashSet mult=0..* */
    public java.util.Collection<Mission> mission;

    /** @pdGenerated default getter */
    public java.util.Collection<GradeProfesseur> getGradeProfesseur() {
        if (gradeProfesseur == null)
            gradeProfesseur = new java.util.HashSet<GradeProfesseur>();
        return gradeProfesseur;
    }

    /** @pdGenerated default iterator getter */
    public java.util.Iterator getIteratorGradeProfesseur() {
        if (gradeProfesseur == null)
            gradeProfesseur = new java.util.HashSet<GradeProfesseur>();
        return gradeProfesseur.iterator();
    }
}
```

```
/** @pdGenerated default setter
 * @param newGradeProfesseur */
public void setGradeProfesseur(java.util.Collection<GradeProfesseur> newGradeProfesseur) {
    removeAllGradeProfesseur();
    for (java.util.Iterator iter = newGradeProfesseur.iterator(); iter.hasNext();
        addGradeProfesseur((GradeProfesseur)iter.next());
    }

/** @pdGenerated default add
 * @param newGradeProfesseur */
public void addGradeProfesseur(GradeProfesseur newGradeProfesseur) {
    if (newGradeProfesseur == null)
        return;
    if (this.gradeProfesseur == null)
        this.gradeProfesseur = new java.util.HashSet<GradeProfesseur>();
    if (!this.gradeProfesseur.contains(newGradeProfesseur))
        this.gradeProfesseur.add(newGradeProfesseur);
}

/** @pdGenerated default remove
 * @param oldGradeProfesseur */
public void removeGradeProfesseur(GradeProfesseur oldGradeProfesseur) {
    if (oldGradeProfesseur == null)
        return;
    if (this.gradeProfesseur != null)
        if (this.gradeProfesseur.contains(oldGradeProfesseur))
            this.gradeProfesseur.remove(oldGradeProfesseur);
}

/** @pdGenerated default removeAll */
public void removeAllGradeProfesseur() {
    if (gradeProfesseur != null)
        gradeProfesseur.clear();
}

/** @pdGenerated default getter */
public java.util.Collection<Mission> getMission() {
    if (mission == null)
        mission = new java.util.HashSet<Mission>();
    return mission;
}

/** @pdGenerated default iterator getter */
public java.util.Iterator getIteratorMission() {
    if (mission == null)
        mission = new java.util.HashSet<Mission>();
    return mission.iterator();
}

/** @pdGenerated default setter
 * @param newMission */
public void setMission(java.util.Collection<Mission> newMission) {
    removeAllMission();
    for (java.util.Iterator iter = newMission.iterator(); iter.hasNext();
        addMission((Mission)iter.next());
    }

/** @pdGenerated default add
 * @param newMission */
public void addMission(Mission newMission) {
    if (newMission == null)
        return;
    if (this.mission == null)
```



```

        this.mission = new java.util.HashSet<Mission>();
        if (!this.mission.contains(newMission))
            this.mission.add(newMission);
    }

    /** @pdGenerated default remove
     * @param oldMission */
    public void removeMission(Mission oldMission) {
        if (oldMission == null)
            return;
        if (this.mission != null)
            if (this.mission.contains(oldMission))
                this.mission.remove(oldMission);
    }

    /** @pdGenerated default removeAll */
    public void removeAllMission() {
        if (mission != null)
            mission.clear();
    }
}

```

### **3.1.6.4 Début de la classe Professeur**

```

/*****
 * Module: Professeur.java
 * Author: simo-
 * Purpose: Defines the Class Professeur
 *****/

```

### **3.1.6.5 Importations de la classe Professeur**

```
import java.util.*;
```

### **3.1.6.10 Attribut idProfesseur de la classe Professeur**

### **3.1.6.11 Attribut nomProfesseur de la classe Professeur**

### **3.1.6.12 Attribut prenomProfesseur de la classe Professeur**

### **3.1.6.13 Attribut dateNaissance de la classe Professeur**

### **3.1.6.14 Attribut ville de la classe Professeur**

### **3.1.6.15 Attribut adresse de la classe Professeur**

### **3.1.6.16 Attribut email de la classe Professeur**

### **3.1.6.17 Attribut telephone de la classe Professeur**

### **3.1.6.18 Attribut protable de la classe Professeur**

### **3.1.6.19 Attribut numCIN de la classe Professeur**

### **3.1.6.20 Attribut etablissementOrigine de la classe Professeur**

### **3.1.6.21 Attribut cv de la classe Professeur**

### **3.1.6.22 Attribut photo de la classe Professeur**

### **3.1.6.23 Attribut CINImprime de la classe Professeur**

**3.1.6.24 Attribut statutProfessionnel de la classe Professeur****3.1.6.25 Attribut volumeHoraireMax de la classe Professeur****3.1.6.26 Attribut tarifJour de la classe Professeur****3.1.6.27 Attribut tarifSoir de la classe Professeur****3.1.7 Classe Transport****3.1.7.3 Aperçu du code de la classe Transport**

```

/*****
 * Module: Transport.java
 * Author: simo-
 * Purpose: Defines the Class Transport
 *****/

```

```

import java.util.*;

/** @pdOid aabf0570-3980-4e28-8ea7-7f5dc17abcea */
public class Transport {
    /** @pdOid eebe27ca-6a65-4f40-a3e5-93e5a6765edb */
    private int idTransport;
    /** @pdOid ee63d25e-b544-4232-aa1a-cb1378e82777 */
    private String typeTransport;
    /** @pdOid 3f186a79-4df7-4e63-be9f-2ee6dfc86279 */
    private String comment;
}

```

**3.1.7.4 Début de la classe Transport**

```

/*****
 * Module: Transport.java
 * Author: simo-
 * Purpose: Defines the Class Transport
 *****/

```

**3.1.7.5 Importations de la classe Transport**

```

import java.util.*;

```

**3.1.7.10 Attribut idTransport de la classe Transport****3.1.7.11 Attribut typeTransport de la classe Transport****3.1.7.12 Attribut comment de la classe Transport****3.1.8 Classe TypeDocument****3.1.8.3 Aperçu du code de la classe TypeDocument**

```

/*****
 * Module: TypeDocument.java
 * Author: simo-
 * Purpose: Defines the Class TypeDocument
 *****/

```

```

import java.util.*;

/** @pdOid 4fa3e13a-7397-4c03-ad2d-b226f3590c87 */
public class TypeDocument {

```

```

/** @pdOid 94464e8c-9266-4dfe-bd54-37d1694ec460 */
private int idTypeDocument;
/** @pdOid f696a25c-78a1-45ad-8bd5-3cd1482a9d51 */
private String tupeDocument;

/** @pdRoleInfo migr=no name=JustificationDocument asse=association4 coll=java.util.Collection impl=java.util.HashSet
mult=0..* */
public java.util.Collection<JustificationDocument> justificationDocument;

/** @pdGenerated default getter */
public java.util.Collection<JustificationDocument> getJustificationDocument() {
    if (justificationDocument == null)
        justificationDocument = new java.util.HashSet<JustificationDocument>();
    return justificationDocument;
}

/** @pdGenerated default iterator getter */
public java.util.Iterator getIteratorJustificationDocument() {
    if (justificationDocument == null)
        justificationDocument = new java.util.HashSet<JustificationDocument>();
    return justificationDocument.iterator();
}

/** @pdGenerated default setter
 * @param newJustificationDocument */
public void setJustificationDocument(java.util.Collection<JustificationDocument> newJustificationDocument) {
    removeAllJustificationDocument();
    for (java.util.Iterator iter = newJustificationDocument.iterator(); iter.hasNext(); )
        addJustificationDocument((JustificationDocument)iter.next());
}

/** @pdGenerated default add
 * @param newJustificationDocument */
public void addJustificationDocument(JustificationDocument newJustificationDocument) {
    if (newJustificationDocument == null)
        return;
    if (this.justificationDocument == null)
        this.justificationDocument = new java.util.HashSet<JustificationDocument>();
    if (!this.justificationDocument.contains(newJustificationDocument))
        this.justificationDocument.add(newJustificationDocument);
}

/** @pdGenerated default remove
 * @param oldJustificationDocument */
public void removeJustificationDocument(JustificationDocument oldJustificationDocument) {
    if (oldJustificationDocument == null)
        return;
    if (this.justificationDocument != null)
        if (this.justificationDocument.contains(oldJustificationDocument))
            this.justificationDocument.remove(oldJustificationDocument);
}

/** @pdGenerated default removeAll */
public void removeAllJustificationDocument() {
    if (justificationDocument != null)
        justificationDocument.clear();
}
}

```

### 3.1.8.4 Début de la classe TypeDocument

```

/*****

```

---

\* Module: TypeDocument.java

\* Author: simo-

\* Purpose: Defines the Class TypeDocument

\*\*\*\*\*/

### **3.1.8.5 Importations de la classe TypeDocument**

import java.util.\*;

### **3.1.8.10 Attribut idTypeDocument de la classe TypeDocument**

### **3.1.8.11 Attribut tupeDocument de la classe TypeDocument**

