



Lecteur de diaporamas – Dossier d’analyse et de conception

BAROS Arthur, TD2 – TP4
BERGOS Ugo, TD2- TP4
JEANNIN Guillem, TD2 – TP4

Table des matières

Version 1 – Orienté objet	3
1. Diagramme des classes UML	3
2. Attributs et méthodes des classes	4
2.1 Classe Image.....	4
2.2 Classe Lecteur.....	5
2.3 Classe ImageDansDiaporama.....	7
2.4 Classe Diaporama	8
Version 2 – Projet QT Graphique	10
1. Diagramme état-transition	10
1.1 Format diagramme	10
1.2 Format matriciel	10
1.3 Commentaire	11
Version 2_MVP – Modèle MVP	13
1. Diagramme des classes UML.....	13
2. Attributs et méthodes des classes	14
2.1 Classe Modele	14
2.2 Classe LecteurVue	15
2.3 Classe Presentation	16
2.4 Classe Diaporama	17
2.5 Classe ImageDansDiaporama.....	19
2.6 Classe Image.....	20

Version 1 – Orienté objet

1. Diagramme des classes UML



2. Attributs et méthodes des classes

2.1 Classe Image

Classe Image			
Nom attribut	Signification	Type	Exemple
titre	L'intitulé de l'image.	Chaine de caractères	Cendrillon
categorie	La catégorie de l'image.	Chaine de caractères	Personnage
chemin	Le chemin de l'image.	Chaine de caractères	C:\\princesses\\Disney.jpg

Déclaration méthode	Signification / but
Image (string pTitre, string pCategorie, string pChemin)	Constructeur par défaut de la classe Image.
Image (const Image& pImage)	Constructeur de copie.
~Image()	Destructeur de la classe Image.
string getCategorie() const	Getter pour récupérer la catégorie d'une image.
void setCategorie(string)	Setter pour retourner la catégorie d'une image.
string getTitre() const	Getter pour récupérer le titre d'une image.
void setTitre(string)	Setter pour retourner le titre d'une image.
string getChemin() const	Getter pour récupérer le chemin d'accès d'une image.
void setChemin(string)	Setter pour retourner le chemin d'accès d'une image.
void afficher() const	Procédure pour afficher une image, c'est-à-dire toutes ses caractéristiques : titre, catégorie et chemin d'accès.

2.2 Classe Lecteur

Classe Lecteur			
Nom attribut	Signification	Type	Exemple
localisationDiaporamas	Tableau dynamique de toutes les diapositives.	Diaporamas	/
posDiapoCourant	Le numéro du diaporama courant.	Entier non signé	5
modeAuto	Signal si le diaporama est en lecture automatique ou non..	booléen	5

Déclaration méthode	Signification / but
Lecteur(const Diaporamas& = Diaporamas(), unsigned int=0, bool=false)	Constructeur par défaut de la classe Lecteur.
Lecteur(const Lecteur&)	Constructeur par copie de la classe Lecteur.
~Lecteur()	Destructeur de la classe Lecteur.
void setDiapos(const Diaporamas&)	Setter permettant de configurer le diaporama courant.
Diaporamas getDiapos() const	Getter permettant d'obtenir le diaporama courant.
void setPosDiapoCourant (unsigned int)	Setter permettant de configurer le numéro du diaporama courant.
unsigned int nbDiapos() const	Getter permettant d'obtenir le nombre de diaporamas.
Unsigned int getPosDiapoCourant() const	Getter permettant d'obtenir le numéro du diaporama courant.

void setMode(bool)	Setter pour définir le mode de défilement du diaporama.
bool getMode() const	Getter permettant d'obtenir le mode de défilement courant du diaporama.
void declencherAction(char)	Procédure pour déclencher une action spécifique basée sur le caractère fourni en paramètre.
Void saisieVerifChoixActionSurlImageC ourante (char&)	Saisit et vérifie le choix de l'action à effectuer sur l'image courante du diaporama.
unsigned int saisieVerifChoixDiaporama()	Saisit et vérifie le choix du diaporama à afficher.
void charger(Images&)	Procédure pour charger les images.
void charger(Diaporamas&)	Procédure pour charger les diapositives pour le diaporama, utilisant les images chargées.
Diaporama getDiapoCourant() const	Permet d'obtenir le diaporama courant.
void afficherImageCouranteDansDiap oramaCourant() const	Procédure pour afficher les diapositives lors du diaporama.

2.3 Classe ImageDansDiaporama

Classe ImageDansDiaporama			
Nom attribut	Signification	Type	Exemple
pos	Rang de l'image dans le tableau d'image.	Entier non signé	8
rang	Rang de l'image dans le diaporama.	Entier non signé	6

Déclaration méthode	Signification / but
ImageDansDiaporama(const Image& = Image(), unsigned int=0,unsigned int=0)	Constructeur par défaut.
ImageDansDiaporama(const ImageDansDiaporama&)	Constructeur par copie.
~ImageDansDiaporama()	Destructeur de la classe ImageDansDiaporama.
unsigned int getPos() const	Getter pour obtenir la position d'une image dans le diaporama.
void setPos(unsigned int)	Setter pour définir la position d'une image dans le diaporama.
unsigned int getRang() const	Getter pour obtenir le rang d'une image dans le diaporama.
void setRang(unsigned int)	Setter pour définir le rang d'une image dans le diaporama.

2.4 Classe Diaporama

Classe Diaporama			
Nom attribut	Signification	Type	Exemple
titre	Le titre du diaporama	Chaine de caractères	Diaporama tutor
vitesseDefilement	La vitesse de défilement des images (si mode automatique).	Entier non signé	5
localisationImages	Le tableau dynamique contenant les ImagesDansDiaporama.	ImagesDansDiaporama (=Vector <ImageDansDiaporama>)	/
posImageCourante	La position de l'image courante.	Entier non signé	0

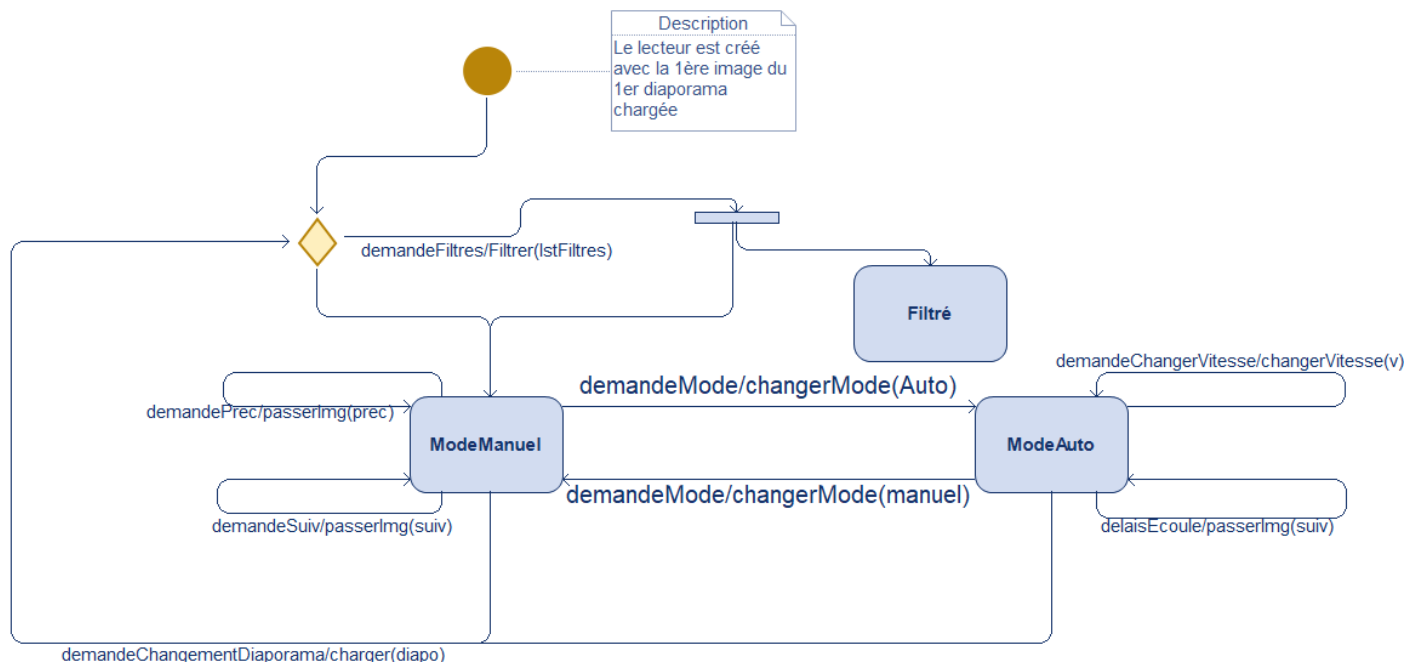
Déclaration méthode	Signification / but
Diaporama (const ImagesDansDiaporama& = ImagesDansDiaporama(), string="Diaporama par défaut", unsigned int=2, unsigned int=0)	Constructeur par défaut.
Diaporama(const Diaporama&)	Constructeur par copie.
~Diaporama()	Destructeur du Diaporama.
string getTitre() const	Getter permettant d'avoir le titre du diaporama.
void setTitre(string)	Setter pour définir le titre du diaporama à partir de la chaine de caractère en paramètre.
unsigned int getVitesseDefilement() const	Getter permettant d'avoir la vitesse de défilement d'une image.
void setVitesseDefilement(unsigned int)	Setter pour définir la vitesse de défilement d'un diaporama.
unsigned int getPosImageCourante() const	Getter permettant d'avoir le numéro de la position de l'image courante.
void setPosImageCourante(unsigned int)	Setter pour définir le numéro de la position de l'image courante.
ImagesDansDiaporama getLocalisationImages() const	Getter permettant d'avoir la localisation des images.

void setLocalisationImages(ImageDans Diaporama&)	Setter pour définir un vecteur d'images.
ImageDansDiaporama getImageCourante() const	Permet d'obtenir l'Image courante du diaporama
void avancer()	Permet d'avancer dans le diaporama.
void reculer()	Permet de reculer dans le diaporama.
unsigned int nbImages() const	Permet d'avoir le nombre d'images dans le diaporama.
void triCroissantRang()	Permet de trier les images de manière croissante dans le vecteur d'images.
Void pushBackImage()	Place une image en dernière position du diaporama.
Void clear()	Permet de supprimer le Diaporama courant.

Version 2 – Projet QT Graphique

1. Diagramme état-transition

1.1 Format diagramme



1.2 Format matriciel

Éléments application	btnAuto	btnPrec	btnSuiv	btnVit	btnFiltres	btnCharger	timer
Événements -> Etats ↓	demandeMode	demandePrec	demandeSuiv	demandeChangerVitesse	demandeFiltres	demandeChangementDiaporama	delaisEcoule
Filtré	---	---	---	---	---	---	---
Mode Manuel	changerMode (Auto) : ModeAuto	PasserImg(prec) : ModeManuel	PasserImg(suiv) : ModeManuel	---	---	Charger(diapo) : ModeManuel	---
Mode Auto	changerMode (Manuel) :	PasserImg(prec) : ModeAuto	PasserImg(suiv) : ModeAuto	changerVitesse(v) : ModeAuto	---	Charger(diapo) : ModeManuel	PasserImg(suiv) : ModeAuto

1.3 Commentaire

Le diaporama a toujours une image et un diaporama chargés.

```
passerImg(cible)
```

```
    chargerImg(cible)
```

```
    afficherImg()
```

```
    afficherInfosImg()
```

```
    if (mode == Auto) {lancerDelais()}
```

```
demanderPrec()
```

```
    charger(ImgPrec)
```

```
    if (mode == Auto) {mode = Manuel}
```

```
demanderPrec()
```

```
    charger(ImgSuiv)
```

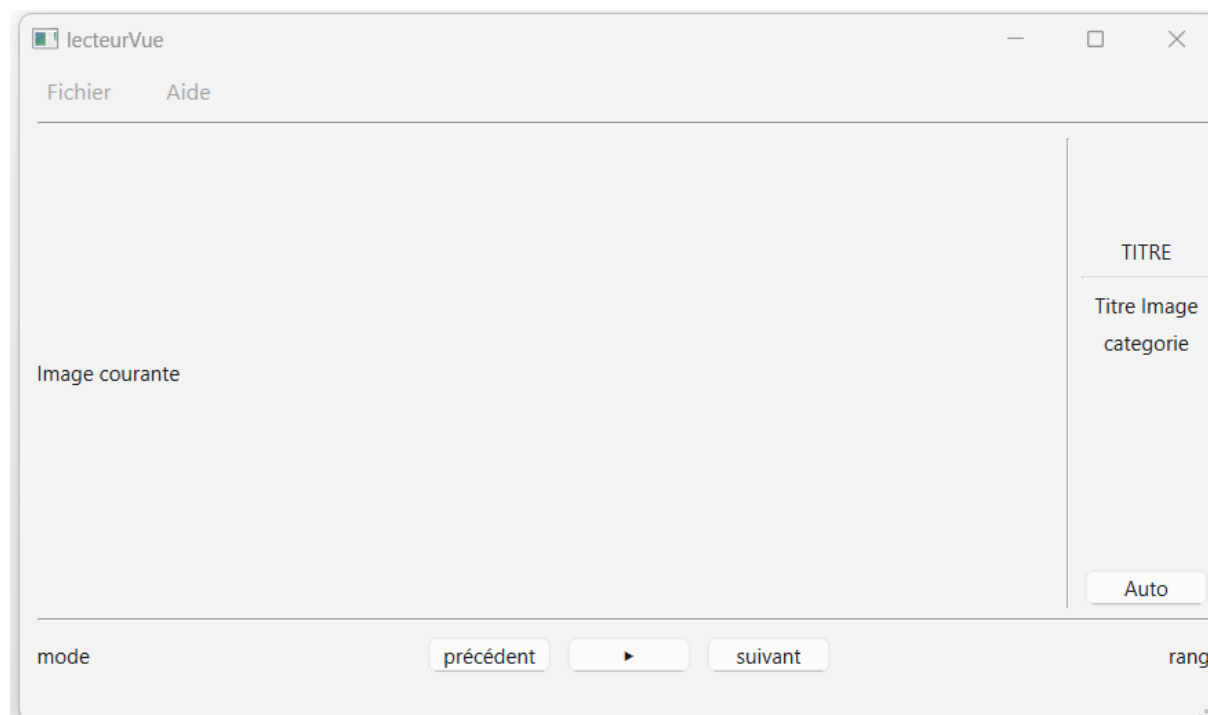
```
    if (mode == Auto) {mode = Manuel}
```

```
changerMode(mode)
```

```
    setMode(mode)
```

```
    afficherMode(mode)
```

2. Liens entre éléments d'interface et fonctionnalités



Le lecteur, qui se lance en affichant la première image du premier diaporama dès que possible, propose deux modes de défilement, qu'il est possible de sélectionner de diverses façons : le lecteur est automatiquement en mode manuel, initialement, et peut être passé en mode automatique grâce au bouton « Auto » en bas à droite, puis peut-être repassé en mode manuel en re cliquant sur ce même bouton ou en cliquant sur l'un des 3 du milieu (« précédent », pause et « suivant »). Le mode actuel est ensuite affiché en bas à gauche tandis que le rang de l'image, lui se trouve à la même hauteur mais à l'autre extrémité.

Lors du mode manuel, les boutons « précédent » et « suivant » servent à naviguer dans les images du diaporama courant.

Lors du mode automatique, c'est la vitesse de défilement (modifiable dans Fichier -> Paramètres -> Vitesse de défilement) qui règle la vitesse à laquelle le lecteur remplace l'image affichée par la suivante du diaporama

Le titre du diaporama courant ainsi que les informations de l'image courante sont affichés dans la colonne de droite, au-dessus du bouton « Auto ».

Si l'utilisateur ne souhaite voir que certaines images du diaporama, il peut appliquer de filtres via le menu déroulant « Fichier » (Fichier -> Paramètres -> Filtres)

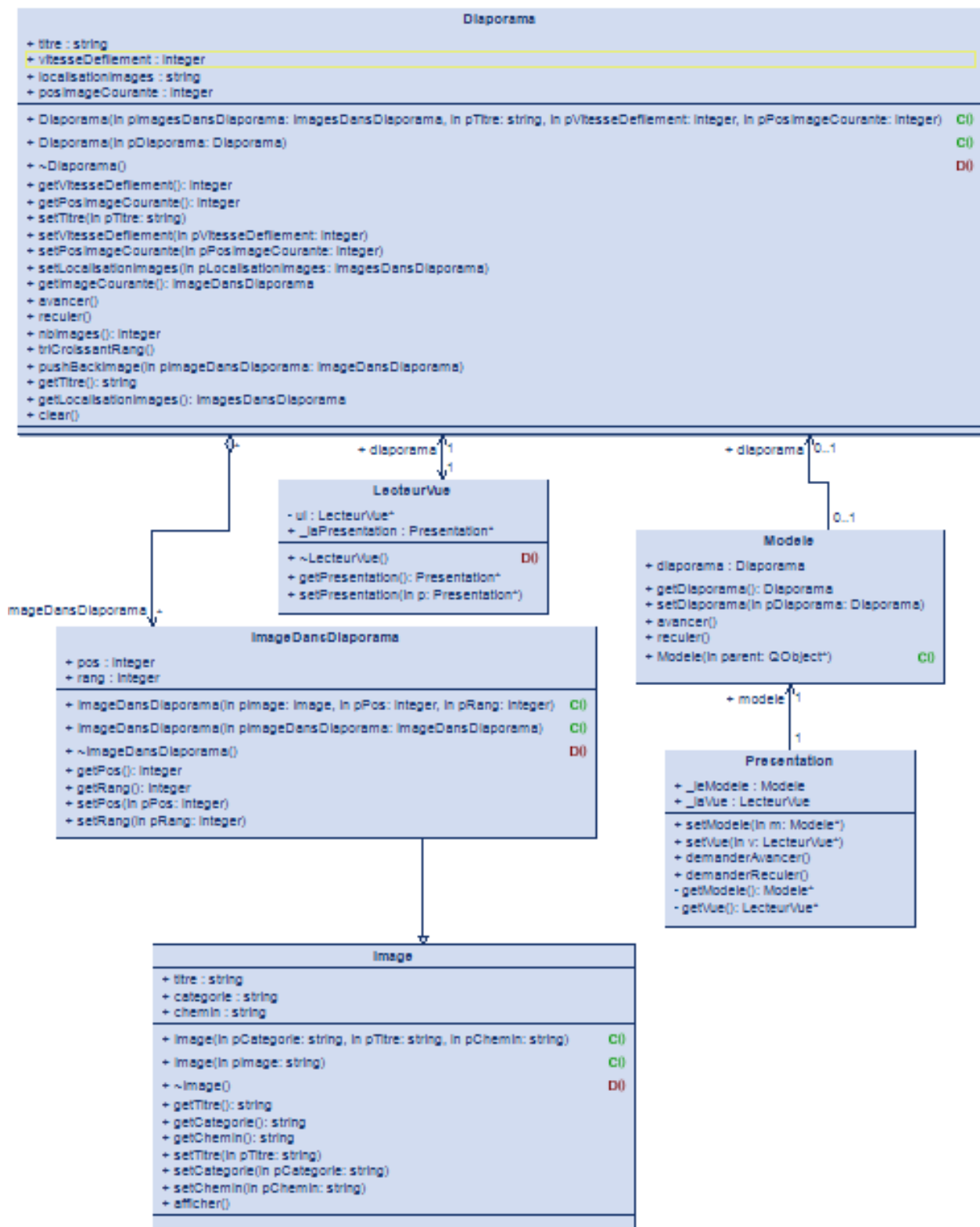
S'il souhaite simplement changer de diaporama, il peut cliquer sur « Charger diaporama », dans le menu déroulant « Fichier »

Le menu d'aide, en haut à gauche, sert quant à lui à accéder à la section « A propos de », qui ouvre une boîte de Message indiquant les auteurs et la version de l'application.

Enfin, le bouton « quitter », situé dans le menu déroulant « Fichier » (dans le coin supérieur gauche), permet de quitter l'application.

Version 2_MVP – Modèle MVP

1. Diagramme des classes UML



2. Attributs et méthodes des classes

2.1 Classe Modele

Classe Modele			
Nom attribut	Signification	Type	Exemple
diaporama	Pointeur vers le Diaporama	Diaporama*	/

Déclaration méthode	Signification / but
explicit Modele(QWidget *parent = nullptr)	Constructeur de la classe
Diaporama* getDiaporama()	Récupère le pointeur vers le Diaporama
void setDiaporama(Diaporama*)	Définit le pointeur vers le Diaporama
avancer()	Fait avancer le Diaporama
reculer()	Fait reculer le Diaporama

2.2 Classe LecteurVue

Classe LecteurVue			
Nom attribut	Signification	Type	Exemple
Presentattion	Pointeur vers la Presentation	Presentation *	/

Déclaration méthode	Signification / but
explicit LecteurVue(QWidget *parent = nullptr)	Constructeur de la classe Lecteur
~lecteurvue()	Destructeur de la classe lecteurvue
Presentation* getPresentation()	Récupère le pointeur vers la Presentation
void setPresentation(Presentation*)	Définit le pointeur vers la Presentation
void demanderFiltres()	Slot lié à l'appui du bouton « Filtres »
void demanderVitesseDefilement()	Slot lié à l'appui du bouton « Vitesse de défilement »
void demanderChargerDiaporama()	Slot lié à l'appui du bouton « Charger diaporama »
void demanderQuitter()	Slot lié à l'appui du bouton « Quitter »
void demanderAProposDe()	Slot lié à l'appui du bouton « A propos de »
void demanderPrecedent()	Slot lié à l'appui du bouton « Précédent »
void demanderSuivant()	Slot lié à l'appui du bouton « Suivant »
void demanderPause()	Slot lié à l'appui du bouton Pause
void demanderChangerModeDefilement()	Slot lié à l'appui du bouton du mode de défilement

2.3 Classe Presentation

Classe Presentation			
Nom attribut	Signification	Type	Exemple
_leModele	Pointeur vers le Modele	Modele*	/
_laVue	Pointeur vers LecteurVue	LecteurVue*	/

Déclaration méthode	Signification / but
explicit Presentation(QWidget *parent = nullptr)	Constructeur de la classe Presentation
Modele* getModele()	Récupère le pointeur vers le Modele
LecteurVue* getVue()	Récupère le pointeur vers LecteurVue
void setModele(Modele *m)	Définit le pointeur vers le Modele
void setVue(LecteurVue *v)	Définit le pointeur vers LecteurVue
void demandeAvancer()	Avancer dans le diaporama
void demandeReculer()	Reculer dans le diaporama

2.4 Classe Diaporama

Classe Diaporama			
Nom attribut	Signification	Type	Exemple
titre	Le titre du diaporama	Chaine de caractères	Diaporama tutor
vitesseDefilement	La vitesse de défilement des images (si mode automatique).	Entier non signé	5
localisationImages	Le tableau dynamique contenant les ImagesDansDiaporama.	ImagesDansDiaporama (=Vector <ImageDansDiaporama>)	/
posImageCourante	La position de l'image courante.	Entier non signé	0

Déclaration méthode	Signification / but
Diaporama (const ImagesDansDiaporama& = ImagesDansDiaporama(), string="Diaporama par défaut", unsigned int=2, unsigned int=0)	Constructeur par défaut.
Diaporama(const Diaporama&)	Constructeur par copie.
~Diaporama()	Destructeur du Diaporama.
string getTitre() const	Getter permettant d'avoir le titre du diaporama.
void setTitre(string)	Setter pour définir le titre du diaporama à partir de la chaine de caractère en paramètre.
unsigned int getVitesseDefilement() const	Getter permettant d'avoir la vitesse de défilement d'une image.
void setVitesseDefilement(unsigned int)	Setter pour définir la vitesse de défilement d'un diaporama.
unsigned int getPosImageCourante() const	Getter permettant d'avoir le numéro de la position de l'image courante.
void setPosImageCourante(unsigned int)	Setter pour définir le numéro de la position de l'image courante.

ImagesDansDiaporama getLocalisationImages() const	Getter permettant d’avoir la localisation des images.
void setLocalisationImages(ImagesDans Diaporama&)	Setter pour définir un vecteur d’images.
ImageDansDiaporama getImageCourante() const	Permet d’obtenir l’Image courante du diaporama
void avancer()	Permet d’avancer dans le diaporama.

2.5 Classe ImageDansDiaporama

Classe ImageDansDiaporama			
Nom attribut	Signification	Type	Exemple
pos	Rang de l'image dans le tableau d'image.	Entier non signé	8
rang	Rang de l'image dans le diaporama.	Entier non signé	6

Déclaration méthode	Signification / but
ImageDansDiaporama(const Image& = Image(), unsigned int=0,unsigned int=0)	Constructeur par défaut.
ImageDansDiaporama(const ImageDansDiaporama&)	Constructeur par copie.
~ImageDansDiaporama()	Destructeur de la classe ImageDansDiaporama.
unsigned int getPos() const	Getter pour obtenir la position d'une image dans le diaporama.
void setPos(unsigned int)	Setter pour définir la position d'une image dans le diaporama.
unsigned int getRang() const	Getter pour obtenir le rang d'une image dans le diaporama.
void setRang(unsigned int)	Setter pour définir le rang d'une image dans le diaporama.

2.6 Classe Image

Classe Image			
Nom attribut	Signification	Type	Exemple
titre	L'intitulé de l'image.	Chaine de caractères	Cendrillon
categorie	La catégorie de l'image.	Chaine de caractères	Personnage
chemin	Le chemin de l'image.	Chaine de caractères	C:\\princesses\\Disney.jpg

Déclaration méthode	Signification / but
Image (string pTitre, string pCategorie, string pChemin)	Constructeur par défaut de la classe Image.
Image (const Image& pImage)	Constructeur de copie.
~Image()	Destructeur de la classe Image.
string getCategorie() const	Getter pour récupérer la catégorie d'une image.
void setCategorie(string)	Setter pour retourner la catégorie d'une image.
string getTitre() const	Getter pour récupérer le titre d'une image.
void setTitre(string)	Setter pour retourner le titre d'une image.
string getChemin() const	Getter pour récupérer le chemin d'accès d'une image.
void setChemin(string)	Setter pour retourner le chemin d'accès d'une image.
void afficher() const	Procédure pour afficher une image, c'est-à-dire toutes ses caractéristiques : titre, catégorie et chemin d'accès.