

Dossier de conception

S2.01 - Développement d'une application

Lecteur de diaporama



LUCAS Liam | TDIII, TP5

HAMID Enzo | TDIII, TP5

Groupe III-TP5-5

Documentation.....	3
Énumérations.....	3
ModeLecteur.....	3
Utilité de l'énumération.....	3
Valeurs.....	3
EtatLecteur.....	3
Utilité de l'énumération.....	3
Valeurs.....	3
EtatChargementLecteur.....	3
Utilité de l'énumération.....	3
Valeurs.....	3
Classes.....	4
Diaporama.....	4
Attributs.....	4
Méthodes.....	4
Lecteur.....	8
Définition générale de la classe.....	8
Attributs.....	8
Méthodes.....	8
Image.....	12
Définition générale de la classe.....	12
Attributs.....	12
Méthodes.....	12
Diagramme de classe.....	14
Documentation des éléments d'interface.....	15
Label.....	15
Bouton.....	15
LineEdit.....	15
Action.....	15
StatusBar.....	16
Diagramme D'État-Transition.....	16
Classique.....	16
Matriciel.....	17

Documentation

Énumérations

ModeLecteur

Utilité de l'énumération

ModeLecteur indique si un lecteur est en mode de défilement automatique ou manuel.

Valeurs

ModeManuel = 0 : Le lecteur est en mode de défilement manuel.

ModeAuto = 1 : Le lecteur est en mode de défilement automatique.

EtatLecteur

Utilité de l'énumération

EtatLecteur indique si un lecteur est actif (en lecture) ou inactif (hors lecture).

Valeurs

Inactif = 0 : Le lecteur est inactif (aucune lecture n'est en cours).

Actif = 1 : Le lecteur est actif (une lecture est en cours).

EtatChargementLecteur

Utilité de l'énumération

EtatChargementLecteur indique si un lecteur est vide (pas de diaporama chargé) ou rempli (un diaporama chargé).

Valeurs

Vide = 0 : Le lecteur est vide (aucun diaporama n'est chargé).

Rempli = 1 : Le lecteur est rempli (un diaporama est chargé).

Classes

Diaporama

Attributs

Spécificateur d'accès	Type	Nom	Signification
private	unsigned short int	mIdentifiantDiaporama	identifiant du diaporama
private	string	mTitreDiaporama	titre du diaporama
private	unsigned short int	mVitesseDefilement	vitesse de defilement
private	vector<LocalisationsImagesDiaporama>	mLocalisationsImagesDiaporama	vecteur de la position des images dans le diaporama et dans le vecteur images chargées du lecteur

Méthodes

Spécificateur d'accès	Type	Nom
private	unsigned short int	getIdentifiantDiaporama
private	void	setIdentifiantDiaporama
private	string	getTitreDiaporama
private	void	setTitreDiaporama
private	unsigned short int	getVitesseDefilement
private	vector<LocalisationsImagesDiaporama>	getLocalisationsImagesDiaporama
private	void	setLocalisationsImagesDiaporama
private	LocalisationsImagesDiaporama	getPosImageDansDiaporama
private	void	setPosImageDansDiaporama
private	unsigned int	nbImages
private	void	pushBackLocalisationsImagesDansDiaporama

private	void	changerValeursMembres
private	void	avancer
private	void	reculer

unsigned short int Diaporama::getIdentifiantDiaporama();

Signification : renvoie l'identifiant du diaporama

void Diaporama::setIdentifiantDiaporama(pNewIdentifiantDiaporama);

Signification : modifie l'identifiant du diaporama

@pNewIdentifiantDiaporama :

- Type : unsigned short int
- Signification : nouvel identifiant du diaporama
- Type de passage : par valeur

unsigned short int Diaporama::getTitreDiaporama();

Signification : renvoie le titre du diaporama

void Diaporama::setTitreDiaporama(pNewTitreDiaporama);

Signification : modifie le titre du diaporama

@pNewTitreDiaporama :

- Type : string
- Signification : nouveau titre du diaporama
- Type de passage : par valeur

unsigned short int Diaporama::getVitesseDefilement();

Signification : renvoie la vitesse de défilement du diaporama

void Diaporama::setVitesseDefilement(pNewVitesseDefilement);

Signification : modifie la vitesse de défilement du diaporama

@pNewVitesseDefilement :

- Type : unsigned short int
- Signification : nouvelle vitesse de défilement du diaporama
- Type de passage : par valeur

vector<LocalisationsImagesDiaporama>

Diaporama::getLocalisationsImagesDiaporama();

Signification : renvoie le membre mLocalisationsImagesDiaporama du diaporama

void

Diaporama::setLocalisationsImagesDiaporama(pNewLocalisationsImagesDiaporama);

Signification : modifie le membre mLocalisationsImagesDiaporama du diaporama

@pNewLocalisationsImagesDiaporama :

- Type : vector<LocalisationsImagesDiaporama>
- Signification : nouvelle valeur du membre mLocalisationsImagesDiaporama du diaporama
- Type de passage : par valeur

LocalisationsImagesDiaporama Diaporama::getPosImageDansDiapo(pIndex);

Signification : renvoie le membre mLocalisationsImagesDiaporama à l'index donné du diaporama

@pIndex :

- Type : unsigned short int
- Signification : index dans le membre mLocalisationsImagesDiaporama
- Type de passage : par valeur

void Diaporama::setPosImageDansDiapo(pIndex, pLocalisation);

Signification : modifie le membre mLocalisationsImagesDiaporama à l'index donné du diaporama

@pIndex :

- Type : unsigned short int
- Signification : index dans le membre mLocalisationsImagesDiaporama
- Type de passage : par valeur

@pLocalisation :

- Type : vector<LocalisationsImagesDiaporama>
- Signification : nouvelle valeur du membre mLocalisationsImagesDiaporama du diaporama
- Type de passage : par valeur

unsigned int Diaporama::nbImages();

Signification : renvoie le nombre d'images à partir de mLocalisationsImagesDiaporama

void

Diaporama::pushBackLocalisationsImagesDansDiaporama(pLocalisationsImagesDiaporama);

Signification : push back un objet LocalisationsImagesDiaporama au membre mLocalisationsImagesDiaporama

@pLocalisationsImagesDiaporama :

- Type : LocalisationsImagesDiaporama
- Signification : valeur à rajouter dans le membre mLocalisationsImagesDiaporama
- Type de passage : par valeur

void Diaporama::changerValeursMembres(pNewIdentifiantDiaporama, pNewTitreDiaporama, pNewVitesseDefilement, pNewLocalisationsImagesDiaporama);

Signification : push back un objet LocalisationsImagesDiaporama au membre mLocalisationsImagesDiaporama

@pNewIdentifiantDiaporama :

- Type : unsigned short int
- Signification : nouvelle valeur du membre mIdentifiantDiaporama
- Type de passage : par valeur

@pNewTitreDiaporama :

- Type : string
- Signification : nouvelle valeur du membre mTitreDiaporama
- Type de passage : par valeur

@pNewVitesseDefilement :

- Type : unsigned short int
- Signification : nouvelle valeur du membre mVitesseDefilement
- Type de passage : par valeur

@pNewLocalisationsImagesDiaporama :

- Type : vector<LocalisationsImagesDansDiaporama>
- Signification : nouvelle valeur du membre mLocalisationsImagesDiaporama
- Type de passage : par valeur

void Diaporama::avancer(pPosImageCourante);

Signification : permet d'avancer jusqu'à la prochaine image

@pPosImageCourante :

- Type : unsigned int
- Signification : position de l'image courante
- Type de passage : par référence

void Diaporama::reculer(pPosImageCourante);

Signification : permet de reculer jusqu'à la précédente image

@pPosImageCourante :

- Type : unsigned int
- Signification : position de l'image courante
- Type de passage : par référence

Lecteur

Définition générale de la classe

Un lecteur sert à afficher les images du diaporama en cours. Il dispose de 2 modes de fonctionnement : défilement automatique des images toutes les 2 secondes, et défilement manuel des images après interaction de l'utilisateur.

Attributs

Spécificateur d'accès	Type	Nom	Signification
private	EtatLecteur	mEtatLecteur	état du lecteur
private	ModeLecteur	mModeLecteur	mode du lecteur
private	EtatChargementLecteur	mEtatChargementLecteur	état de chargement du lecteur
private	unsigned short int	mPosDiaporamaCourant	position index du diaporama courant du le lecteur
private	unsigned short int	mPosImageCourante	position image courante du lecteur
private	vector<Diaporama>	mListeDiaporamasCharges	vecteur des diaporamas
private	vector<Image>	mListeImagesChargees	vecteur des images du lecteur

Méthodes

Spécificateur d'accès	Type	Nom
private	EtatLecteur	getEtatLecteur
private	void	setEtatLecteur
private	ModeLecteur	getModeLecteur
private	void	setModeLecteur
private	EtatChargementLecteur	getEtatChargementLecteur
private	void	setEtatChargementLecteur
private	unsigned short int	getPosDiaporamaCourant
private	LocalisationsImagesDiapor	setPosDiaporamaCourant

	ama	
private	unsigned short int	getPosImageCourante
private	void	setPosImageCourante
private	vector<Diaporama>	getListeDiaporamasCharges
private	void	setListeDiaporamasCharges
private	vector<Image>	getListImagesChargees
private	void	setListImagesChargees
private	void	chargerImages
private	void	chargerDiaporamas
private	void	triCroissantRang
private	void	afficherImageCouranteDansDiaporamaCourant

EtatLecteur Lecteur::getEtatLecteur();

Signification : renvoie l'état du lecteur

void Lecteur::setEtatLecteur(pNewEtatLecteur);

Signification : modifie l'état du lecteur

@pNewEtatLecteur:

- Type : EtatLecteur
- Signification : nouvel état du lecteur
- Type de passage : par valeur

ModeLecteur Lecteur::getModelLecteur();

Signification : renvoie le mode du lecteur

void Lecteur::setModeLecteur(pNewModeLecteur);

Signification : modifie le mode du lecteur

@pNewModeLecteur :

- Type : ModeLecteur
- Signification : nouveau mode du lecteur
- Type de passage : par valeur

EtatChargementLecteur Lecteur::getEtatChargementLecteur();

Signification : renvoie l'état de chargement du lecteur

void Lecteur::setEtatChargementLecteur(pNewEtatChargementLecteur);

Signification : modifie l'état de chargement du lecteur

@pNewEtatChargementLecteur :

- Type : EtatChargementLecteur
- Signification : nouvel état de chargement du lecteur

- Type de passage : par valeur

unsigned short int Lecteur::getPosDiaporamaCourant();

Signification : renvoie le diaporama courant

void Lecteur::setPosDiaporamaCourant(pNewPosDiaporamaCourant);

Signification : modifie le diaporama courant

@pNewPosDiaporamaCourant :

- Type : unsigned short int
- Signification : nouvelle position du diaporama courant
- Type de passage : par valeur

unsigned short int Lecteur::getPosImageCourante();

Signification : renvoie l'image courante

void Lecteur::setPosImageCourante(pNewPosImageCourante);

Signification : modifie l'image courante

@pNewPosImageCourante :

- Type : unsigned short int
- Signification : nouvelle position de l'image courante
- Type de passage : par valeur

vector<Diaporama> Lecteur::getListeDiaporamasCharges();

Signification : renvoie la liste des diaporamas chargés

void Lecteur::setListeDiaporamasCharges(pNewListeDiaporamasCharges);

Signification : modifie la liste des diaporamas chargés

@pNewListeDiaporamasCharges :

- Type : vector<Diaporama>
- Signification : nouvelle liste de diaporamas
- Type de passage : par valeur

vector<Image> Lecteur::getListImagesChargees();

Signification : renvoie la liste des images chargées

void Lecteur::setListImagesChargees(pNewListImagesChargees);

Signification : modifie la liste des images chargées

@pNewListImagesChargees :

- Type : vector<Image>
- Signification : nouvelle liste d'images
- Type de passage : par valeur

void Lecteur::chargerImages(plImagesACharger);

Signification : charge les images (ajoute des objets Image au vecteur en paramètre et initialise les objets

@plImagesACharger :

- Type : vector<Image>
- Signification : liste d'images dans laquelle charger les images

- Type de passage : par référence

void Lecteur::chargerDiaporamas(pDiaporamasACharger);

Signification : charge les diaporamas (ajoute des objets Diaporama au vecteur en paramètre et initialise les objets

@pDiaporamasACharger :

- Type : vector<Diaporama>
- Signification : liste de diaporamas dans laquelle charger les diaporamas
- Type de passage : par référence

void Lecteur::triCroissantRang(pDiaporama);

Signification : tri le diaporama par ordre croissant des rang avec un tri à bulle

@pDiaporama :

- Type : Diaporama
- Signification : diaporama à trier
- Type de passage : par référence

void Lecteur::afficherImageCouranteDansDiaporamaCourant(pDiaporamaCourant, pldImageCourante, plImage);

Signification : affiche l'image courante

@pDiaporamaCourant :

- Type : Diaporama
- Signification : diaporama courant
- Type de passage : par référence

@pldImageCourante :

- Type : unsigned short int
- Signification : identifiant de l'image courante
- Type de passage : par valeur

@plImage :

- Type : Image
- Signification : image à afficher
- Type de passage : par référence

Image

Définition générale de la classe

Une image représente un fichier d'image spécifique identifié grâce à son chemin.

Attributs

Spécificateur d'accès	Type	Nom	Signification
private	unsigned short int	mIdentifiantImage	identifiant de l'image
private	string	mTitreImage	titre de l'image
private	string	mCategoriImage	catégorie de l'image
private	string	mCheminImage	chemin absolu de l'image

Méthodes

Spécificateur d'accès	Type	Nom
private	void	nomFonction1
private	int	nomFonction2

unsigned short int Image::getIdentifiantImage();

Signification : renvoie l'identifiant de l'image

void Image::setIdentifiantImage(pNewIdentifiantImage);

Signification : modifie l'identifiant de l'image

@pNewIdentifiantImage :

- Type : unsigned short int
- Signification : nouvel identifiant de l'image
- Type de passage : par valeur

unsigned short int Image::getTitreImage();

Signification : renvoie le titre de l'image

void Image::setTitreImage(pNewTitreImage);

Signification : modifie le titre de l'image

@pNewTitreImage :

- Type : string
- Signification : nouvel identifiant de l'image
- Type de passage : par valeur

string Image::getCategoriImage();

Signification :renvoie la catégorie de l'image

void Image::setCategoriImage(pNewCategoriImage);

Signification : modifie la catégorie de l'image

@pNewCategoriImage :

- Type : string
- Signification : nouvelle catégorie de l'image
- Type de passage : par valeur

string Image::getCheminImage();

Signification :renvoie le chemin absolu de l'image

void Image::setCheminImage(pNewCheminImage);

Signification : modifie le chemin absolu de l'image

@pNewCheminImage :

- Type : string
- Signification : nouveau chemin de l'image
- Type de passage : par valeur

void Image::changerValeursMembres(pNewIdentifiantImage, pNewTitreImage, pNewCategoriImage, pNewCheminImage);

Signification : modifie mIdentifiantImage, mTitreImage, mCategoriImage et mCheminImage avec les valeurs des paramètres

@pNewIdentifiantImage :

- Type : unsigned short int
- Signification : nouvel identifiant de l'image
- Type de passage : par valeur

@pNewTitreImage :

- Type : string
- Signification : nouveau titre de l'image
- Type de passage : par valeur

@pNewCategoriImage :

- Type : string
- Signification : nouvelle catégorie de l'image
- Type de passage : par valeur

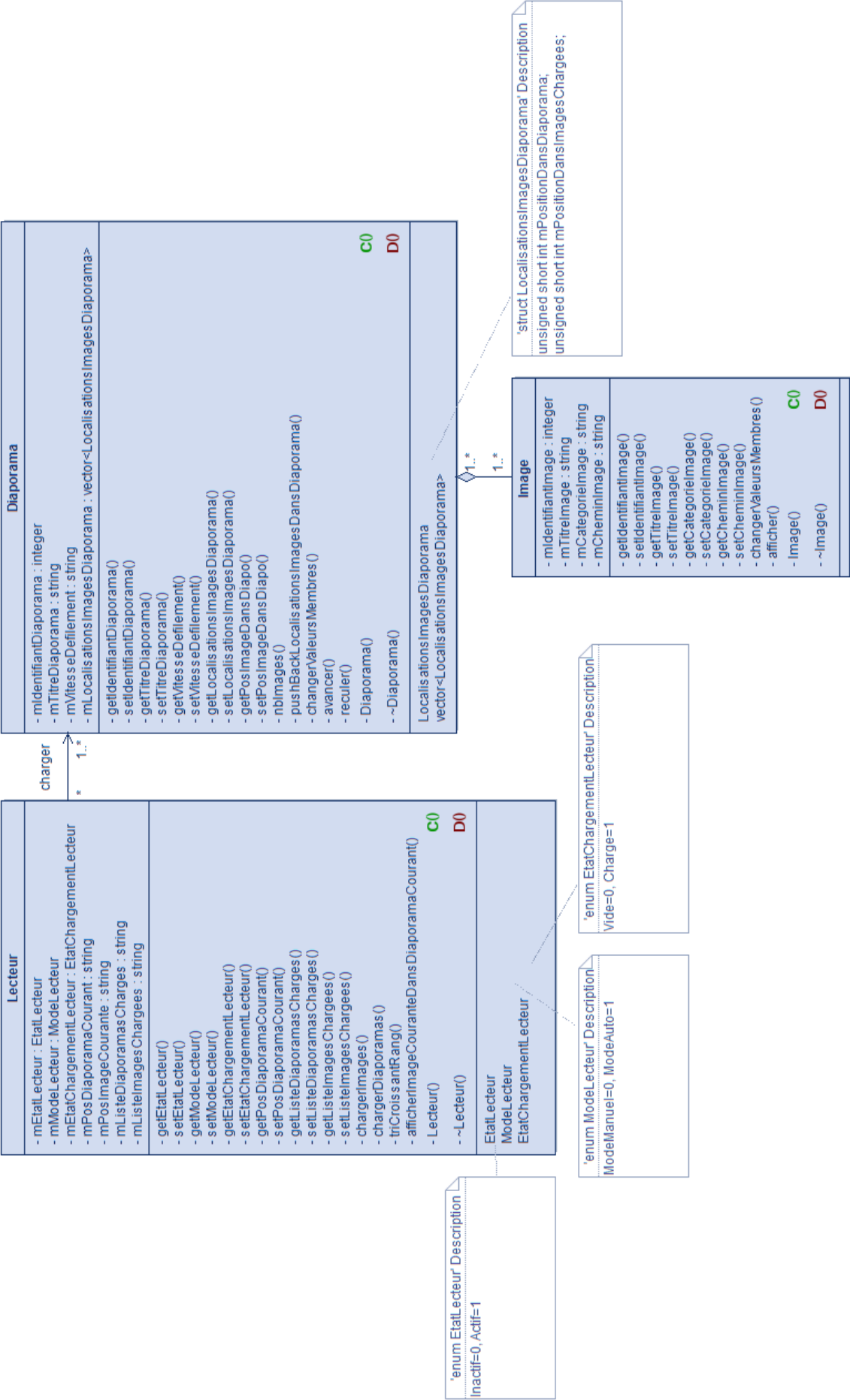
@pNewCheminImage :

- Type : string
- Signification : nouveau chemin de l'image
- Type de passage : par valeur

string Image::afficher();

Signification :permet d'afficher une image

Diagramme de classe



Documentation des éléments d'interface

Label

Label_Image :

Zone de la fenêtre où les diapositives apparaîtront

Bouton

Button_Precedent :

Bouton permettant de reculer d'une image ou alors de passer en mode de lecture manuel si le mode automatique est activé (mise à jour du mode dans la barre de status)

Button_Suivant :

Bouton permettant d'avancer d'une image ou alors de passer en mode de lecture manuel si le mode automatique est activé (mise à jour du mode dans la barre de status)

Button_Lancer :

Active le mode de lecture automatique en respectant la vitesse de défilement indiquée (mise à jour du mode dans la barre de status)

Button_Arreter :

Désactive le mode automatique du lecteur et revient en manuel (mise à jour du mode dans la barre de status)

LineEdit

LE_Filtre_Diapo :

Champ de saisie permettant de saisir une catégorie de diapositive à chercher & charger

Action

Action_A_Propos_De :

« Bouton » du menu Menu_Aide permettant d'ouvrir une page de dialogue contenant les informations sur le logiciel

Action_Quitter :

« Bouton » du menu Menu_Fichier permettant de fermer l'application

Action_Charger_Diaporama :

« Bouton » du menu Menu_Param permettant de charger le diaporama de la catégorie préalablement saisie dans le champ de saisie LE_Filtre_Diapo si celle-ci existe

Action_Enlever_Diaporama :

« Bouton » du menu Menu_Param permettant de décharger le diaporama en cours de lecture et ainsi de vider le lecteur

Action_Vitesse_Defilement :

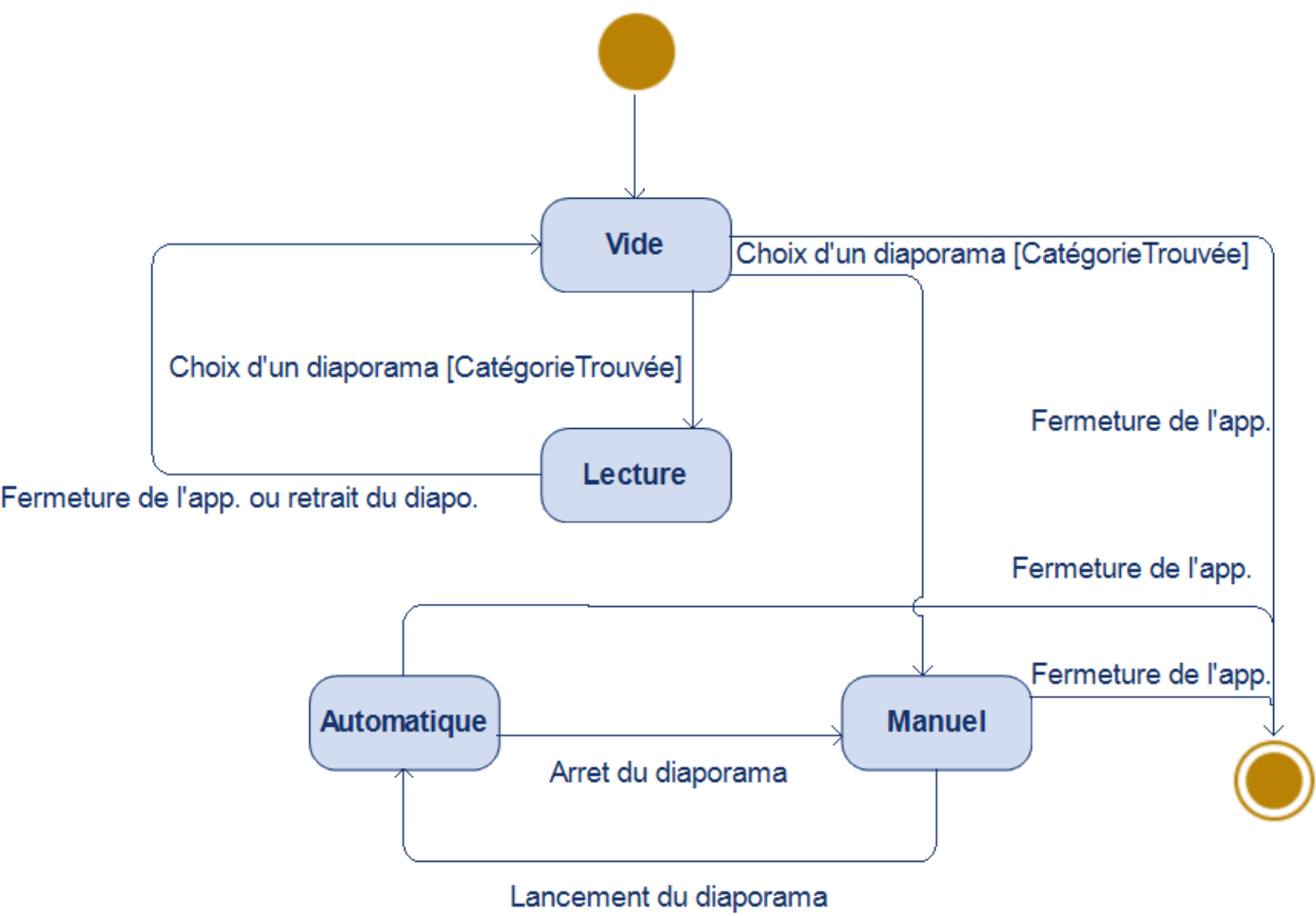
« Bouton » du menu Menu_Param permettant d'ouvrir une fenêtre de dialogue afin de modifier la vitesse de défilement du diaporama actuellement chargé

StatusBar

Statusbar : contient un texte indiquant le mode de lecture actuellement choisi

Diagramme D'État-Transition

Classique



Matriciel

États \ Événements	Choix d'un diaporama	Lancement du diaporama	Arrêt du diaporama ou Suivant/Précédent	Retrait du diapo.	Fermeture de l'app.
Vide [état initial]	[CatégorieTrouvée] Lecture				
EnLecture				Vide	Vide
Automatique	Manuel		Manuel		
Manuel		Automatique			