TD1
TP1
Loustau-Cazaux David
Gomes Yoann
1ère année en BUT Informatique à l'IUT d'Anglet

URL du dépôt Github : https://github.com/DLCSharp/lecteurDiaporama

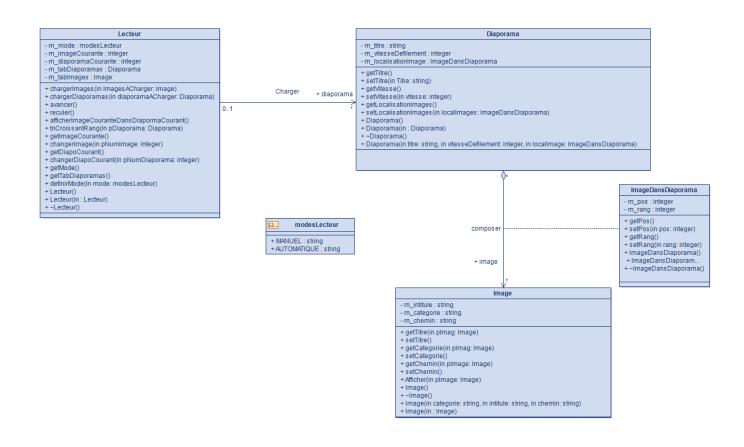
S2.01 - Développement d'une application

Dossier d'analyse et conception des versions

Sommaire

Sommaire	2
1. Diagramme de classes UML	3
2. Documentation des classes	4
3. Diagramme d'état transition	9
4. Dictionnaire des états	10
5. Dictionnaire des événements	10
6. Format matriciel	11
7. Lien entre éléments d'interface et fonctionnalités	12
8. Diagramme de classes UML (v2 MVP)	15
9. Dictionnaire des classes (v2 MVP)	16
10. Bilan sur cette SAE	27

1. Diagramme de classes UML



2. Documentation des classes

Classe Image					
Attributs:					
<u>Intitulé</u>	<u>Signification</u>	But			
m_intitule	L'intitulé de l'image	Donner un nom (intitulé) à l'image			
m_categorie	La catégorie de l'image	Attribuer une catégorie à l'image			
m_chemin	Le chemin absolu d'où est stockée l'image sur le système	Savoir où se situe l'image sur le système			
<u>Méthodes :</u>					
<u>Intitulé</u>	<u>Signification</u>	<u>But</u>			
afficher()	afficher l'image	Permet d'afficher tous les champs de l'image			
setIntitule()	setteur de m_intitule	Définir la nouvelle valeur de l'attribut m_intitule			
getIntitule()	getteur de m_intitule	Récupérer la valeur de m_intitule			
setCategorie()	setteur de m_categorie	Définir la nouvelle valeur de l'attribut m_categorie			
getCategorie()	getteur de m_categorie	Récupérer la valeur de m_categorie			
setChemin()	setteur de m_chemin	Définir la nouvelle valeur de l'attribut m_chemin			
getChemin()	getteur de m_chemin	Récupérer la valeur de m_chemin			
~Image()	Destructeur	Supprime une instance d'Image			
Image()	Constructeur par défaut	Créer une nouvelle instance d'Image			
Image(intitule : string, categorie : string, chemin : string)	Constructeur avec définition de la valeur de tous les attributs	Créer une nouvelle instance d'Image avec la valeur donnée pour chaque attribut			
Image(Images)	Constructeur de copie	Créer une nouvelle instance avec les information d'un autre instance source			

Classe ImageDansDiapo (Représentation sous forme de classe de l'association entre les classes Image et Diaporama)

۸.	++,	-ih	LIFO	
А	ιu	ıυ	uts	

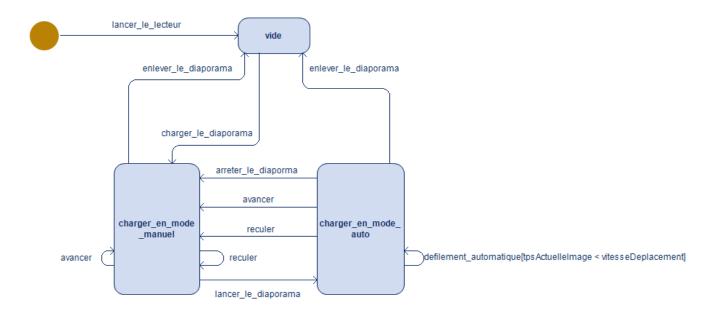
Attributs:		
<u>Intitulé</u>	<u>Signification</u>	<u>But</u>
m_pos	position de l'image dans le tableau des images	Pouvoir rangé les images dans un tableau avec un identifiant pour pouvoir les utiliser
m_rang	rang de l'image dans le diaporama	Connaître l'ordre d'affichage dans le diaporama
<u>Méthodes :</u>		
<u>Intitulé</u>	<u>Signification</u>	<u>But</u>
setPos()	setteur de m_pos	Définir la nouvelle valeur de l'attribut m_pos
getPos()	getteur de m_pos	Récupérer la valeur de m_pos
setRang()	setteur de m_Rang	Définir la nouvelle valeur de l'attribut m_rang
getRang()	getteur de m_Rang	Récupérer la valeur de m_rang
~ImageDansDiapo()	Destructeur de l'instance	Supprime une instance d'ImageDansDiapo
ImageDansDiapo()	Constructeur par défaut	Créer une nouvelle instance d'ImageDansDiapo
ImageDansDiapo(pos : int, rang : int)	Constructeur avec définition de la valeur de tous les attributs	Créer une nouvelle instance d'ImageDansDiapo avec la valeur donnée pour chaque attribut
ImageDansDiapo(ImageDansDiapo)	Constructeur de copie	Créer une nouvelle instance avec les information d'un autre instance source

Classe Diaporama					
Attributs:					
<u>Intitulé</u>	<u>Signification</u>	<u>But</u>			
m_titre	TItre du diaporama	Donner un nom (titre) au diaporama			
m_vitesseDefilement	Vitesse d'affichage du diaporama	Donner une vitesse d'affichage au diaporama			
m_LocalisationImage	Tableau contenant les image du diaporama	Stocker toute les images qui vont composer le diaporama			
Méthodes:					
<u>Intitulé</u>	<u>Signification</u>	But			
setTitre()	setteur de m_titre	Définir la nouvelle valeur de l'attribut m_titre			
getTitre()	getteur de m_titre	Récupérer la valeur de m_titre			
setVitesse()	setteur de m_vitesseDefilement	Définir la nouvelle valeur de l'attribut m_vitesseDefilement			
getVitesse()	getteur de m_vitesseDefilement	Récupérer la valeur de m_vitesseDefilement			
getLocalisationImage	getteur de m_LocalisationImage	Définir la nouvelle valeur de l'attribut m_LocalisationImage			
setLocalisationImage	setteur de m_LocalisationImage	Récupérer la valeur de m_LocalisationImage			
~Diaporama()	Destructeur de l'instance	Supprime une instance de Diaporama			
Diaporama()	Constructeur par défaut	Créer une nouvelle instance de Diaporama			
Diaporama(Diaporama)	Constructeur de copie Créer une nouvelle instavec les information d'un instance source				
Diaporama(titre : string, vitesseDefilement : int, localImage : ImageDansDiaporama)	Constructeur avec définition de la valeur de tous les attributs	Créer une nouvelle instance de Diaporama et pouvoir préciser chaque attribut			

Classe Lecteur				
Attributs:				
<u>Intitulé</u>	<u>Signification</u>	<u>But</u>		
m_mode	Représente le mode de lecture	Savoir comment doit fonctionner le lecteur (manuel ou automatique)		
m_ImageCourante	Position de l'image - représentée par une ImageDansDiapo dans chaque instance de Diaporama - dans m_tabImages	Savoir dans quel position du tableau des images on se trouve		
m_DiaporamaCourant	Position	Savoir dans quel position du tableau des diaporamas on se trouve		
m_tabDiaporamas	Tableau contenant tous les diaporama	Pouvoir utiliser le tableau pour ensuite afficher le diaporama		
m_tablmages	Tableau contenant tous les Images	Pour utiliser le tableau pour ensuite afficher l'image		
Méthodes :				
<u>Intitulé</u>	Signification	<u>But</u>		
definirMode()	setteur de m_mode	Définir la nouvelle valeur de l'attribut m_mode		
getMode()	getteur de m_mode	Récupérer la valeur de m_mode		
avancer()	Passe à l'image suivante	Permet de passer à l'image suivant		
reculer ()	Passe à l'image précedente	Permet de passer à l'image précédente		
~Lecteur()	Destructeur de l'instance	Supprime une instance de Lecteur		
Lecteur()	Constructeur par défaut Créer une nouvelle inst Lecteur			
Lecteur(Lecteur)	Constructeur de copie	Créer une nouvelle instance avec les information d'un autre instance source		
changerDiaporama()	Changement de diaporama	Permet de changer quel diaporama est utiliser		
changerImage()	Changement d'Image	Permet de changer quel Image est utiliser		

trilmageCroissantRang()	Tri des Images par ordre croissant de rang	Permet de trier les images par ordre croissant par rapport à leurs rang dans la diaporama
chargerImages	Charger les Images	Permet de rajouter une Image dans le tableau des Images
chargerDiaporamas()	Charger les diaporama	Permet de rajouter un diaporama dans le tableau des diaporamas
getImageCourante()	getteur de m_ImageCourante	Récupérer la valeur de m_ImageCourante
getTabDiaporamas()	getteur de m_TabDiaporamas	Récupérer une valeur dans le tableau m_TabDiaporamas
getDiapoCourant()	getteur de m_DiapoCourante	Récupérer la valeur de m_DiapoCourante
afficherImageCouranteDansDia poramaCourant()	Affiche l'image sur le lecteur	Permet d'afficher l'image en position courante du tableau sur le diaporama en position courante dans le tableau

3. Diagramme d'état transition



4. Dictionnaire des états

nom de l'état	Signification
vide	Le Lecteur est vide donc aucun diaporama a été chargé
charger_en_manuel	Le Lecteur à un diaporama chargé et il est en mode Manuel
charger_en_auto	Le Lecteur à un diaporama chargé et il est en mode Auto

5. Dictionnaire des événements

nom de l'événement	Signification
lancer_le_lecteur	L'utilisateur a lancé le lecteur dans son état vide
charger_le_diaporama	Permet de charger un diaporama dans le lecteur
enlever_le_diaporama	Retire le diaporama du lecteur il repasse en état vide
avancer	Change l'image du diaporama afficher avec l'image suivante et si le Lecteur est en mode Auto alors il passe en mode Manuel
reculer	Change l'image du diaporama afficher avec l'image précédente et si le Lecteur est en mode Auto alors il passe en mode Manuel
lancer_le_diaporama	Lance le diaporama en mode Auto
arreter_le_diaporama	Arrête le mode Auto du Lecteur et passe en mode Manuel
défilement_automatique	Si le Lecteur est en mode Auto il va faire défiler les images automatiquement avec une vitesse de défilement prédéfinis qui va correspondre à vitesseDéplacement

6. Format matriciel

Éléments d'interface déclencheurs des événements →		Charger un diaporama	Enlever un diaporama	BoutonAvanc er	BoutonRecule r	bouton lancer le diaporama	BoutonArrêter LeDiaporama	
Événements → États ↓	lancer_le_lecte ur	charger_le_dia porama	enlever_le_dia porama	avancer	reculer	lancer_le_diap orama	arreter_le_diap orama	defilement_aut omatique
vide		charger_en_mo de_Manuel	:		::			
charger_en_mode _Manuel			vide	charger_en_m ode_Manuel	charger_en_m ode_Manuel	charger_en_m ode_auto		
charger_en_mode _auto			vide	charger_en_m ode_Manuel	charger_en_m ode_Manuel		charger_en_m ode_Manuel	charger_en_m ode_auto

7. Lien entre éléments d'interface et fonctionnalités

Informations générales sur l'interface :

- L'interface de l'application est pour le moment présentée sous sa forme où un diaporama est chargé afin de permettre la visibilité d'un coup d'oeil de tous ses éléments.
- Tout texte sous ce format "<texte>" représente l'emplacement d'une information dont le texte est défini lors de l'exécution, par exemple : l'intitulé du diaporama courant.
- Le nom interne de chaque widget est précisé entre parenthèses en italique.
- f

Actions de la barre de menu de l'interface :

Sous-menu "Fichier"

<u>Charger un diaporama (actionFichierChargerDiapo)</u>: Permet de charger un diaporama dans le lecteur, cela correspond à l'événement Charger_le_diaporama.

<u>Enlever un diaporama (actionFichierEnleverDiapo)</u>: Permet de retirer le diaporama courant du lecteur, cela correspond à l'événement enlever le diaporama.

Quitter (actionFichierQuitter): Permet de quitter l'application.

Sous-menu "Paramètres" > Sous-menu "Vitesse de défilement"

Les actions ci-dessous permettent de définir une vitesse de défilement pour le diaporama courant.

<u>1 seconde</u>: Choix prédéfini pour l'utilisateur, définit la vitesse de défilement sur 1 seconde.

<u>2 secondes</u>: Choix prédéfini pour l'utilisateur, définit la vitesse de défilement sur 2 secondes.

<u>3 secondes</u>: Choix prédéfini pour l'utilisateur, définit la vitesse de défilement sur 3 secondes.

<u>Vitesse personnalisée</u>: Permet à l'utilisateur de définir une vitesse de défilement personnalisée en entrant la valeur dans une boîte de dialogue pas encore implémentée.

Sous-menu "Aide"

<u>A propos de (actionAideAPropos):</u> Permet d'ouvrir une fenêtre qui affichera la version de l'application, la date de création de l'application, ainsi que les auteurs.

Elements de l'interface liées au diaporama courant :

<u>Diaporama Courant (labelDiapoCourant)</u>: Étiquette dont le texte affiche l'intitulé de diaporama courant.

Elements de l'interface liées à l'image courante :

<u>Titre de la boîte de regroupement "groupBoxImages" (Pas de nom spécifié en interne)</u>: Affiche le numéro de l'image courante ainsi que le nombre total d'images contenues dans le diaporama courant.

<u>Intitule et catégorie (labellmage)</u>: Dans cette version, étiquette dont le texte affiche l'intitulé et la catégorie de l'image courante. Dans les versions futures, elle permettra d'afficher l'image elle-même.

Boutons d'actions de l'interface :

<u>Avancer (btnAvancer)</u>: Dans les versions futures, permet de passer à l'image suivante dans le lecteur. Si le lecteur est en mode automatique, définis également le mode du lecteur sur manuel, correspond à l'événement avancer. Désactive également le bouton "arrêter le diaporama" en raison du passage en mode manuel.

<u>Reculer (btnReculer):</u> Dans les versions futures, permet de passer à l'image précédente dans le lecteur. Si le lecteur est en mode automatique, définis également le mode du lecteur sur manuel, correspond à l'événement avancer. Désactive également le bouton "arrêter le diaporama" en raison du passage en mode manuel.

<u>Lancer le diaporama (btnLancer)</u>: Dans les versions futures, permet de lancer le défilement automatique du diaporama, passe ainsi le lecteur en mode automatique. Si le lecteur est déjà en mode automatique, passe directement à la première image du diaporama courant.

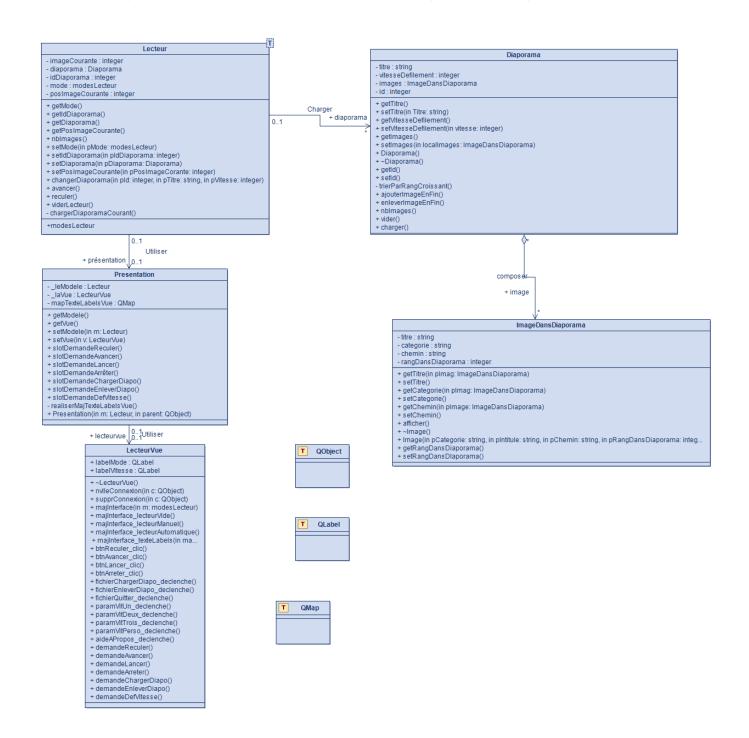
<u>Arrêter le diaporama (btnArreter):</u> Dans les versions futures, arrete le défilement automatique du diaporama, et passe le lecteur en mode manuel. Désactive également ce bouton en raison du passage en mode manuel.

Elements de la barre d'état de l'interface :

Mode (Pas de nom spécifié en interne): Étiquette dont le texte affiche le mode actuel du lecteur.

<u>Vitesse (Pas de nom spécifié en interne)</u> : Étiquette dont le texte affiche la vitesse de défilement du diaporama courant.

8. Diagramme de classes UML (v2 MVP)



9. Dictionnaire des classes (v2 MVP)

Classe ImageDansDiaporama				
Attributs:				
<u>Intitulé</u>	<u>Signification</u>	<u>But</u>		
titre	L'intitulé de l'image	Donner un nom (intitulé) à l'image		
categorie	La catégorie de l'image	Attribuer une catégorie à l'image		
chemin	Le chemin absolu d'où est stockée l'image sur le système	Savoir où se situe l'image sur le système		
rangDansDiaporama	Indique le rang dans le diaporama de l'image	Savoir dans quel ordre les images vont défiler		
<u>Méthodes :</u>				
<u>Intitulé</u>	<u>Signification</u>	<u>But</u>		
afficher()	afficher l'image	Permet d'afficher tous les champs de l'image		
setTitre (string pTitre)	setteur de titre	Définir la nouvelle valeur de l'attribut titre		
getTitre()	getteur de titre	Récupérer la valeur de titre		
setCategorie (string pCategorie)	setteur de categorie	Définir la nouvelle valeur de l'attribut categorie		
getCategorie()	getteur de categorie	Récupérer la valeur de categorie		
setChemin (string pChemin)	setteur de chemin	Définir la nouvelle valeur de l'attribut chemin		
getChemin()	getteur de chemin	Récupérer la valeur de chemin		
~ImageDansDiaporama()	Destructeur	Supprime une instance d'Image		
ImageDansDiaporama(unsigned int pRangDansDiaporama=0, string pCategorie="", string pTitre="", string pChemin = "")	Constructeur avec définition de la valeur de tous les attributs	Créer une nouvelle instance d'Image avec la valeur donnée pour chaque attribut		
getRangDansDiaporama	getteur de rangDansDiaporama	Définir la nouvelle valeur de l'attribut rangDansDiaporama		
setRangDansDiaporama (unsigned int pRangDansDiaporama)	setteur de rangDansDiaporama	Récupérer la valeur de rangDansDiaporama		

	Classe Diaporama				
Attributs:					
<u>Intitulé</u>	<u>Signification</u>	<u>But</u>			
titre	Tltre du diaporama	Donner un nom (titre) au diaporama			
vitesseDefilement	Vitesse d'affichage du diaporama	Donner une vitesse d'affichage au diaporama			
Images	Vecteur contenant les image du diaporama	Stocker toute les images qui vont composer le diaporama			
id	identifiant du diaporama	Permet d'identifier chaque diaporama pour les différencier			
<u>Méthodes</u> :					
<u>Intitulé</u>	<u>Signification</u>	<u>But</u>			
setTitre(string pTitre)	setteur de m_titre	Définir la nouvelle valeur de l'attribut m_titre			
getTitre()	getteur de m_titre	Récupérer la valeur de m_titre			
setVitesseDefilement(unsigned int pVitesseDefilement)	setteur de m_vitesseDefilement	Définir la nouvelle valeur de l'attribut m_vitesseDefilement			
getVitesse()	getteur de m_vitesseDefilement	Récupérer la valeur de m_vitesseDefilement			
getImages()	getteur de m_LocalisationImage	Définir la nouvelle valeur de l'attribut m_LocalisationImage			
setImages(const ImagesDiaporama& pImages)	setteur de m_LocalisationImage	Récupérer la valeur de m_LocalisationImage			
~Diaporama()	Destructeur de l'instance	Supprime une instance de Diaporama			
Diaporama()	Constructeur par défaut	Créer une nouvelle instance de Diaporama			
setId(unsigned int pld)	setteur de id Définir la nouvelle vale l'attribut id				
getId()	getteur de id	Récupérer la valeur de id			
trierParRangCroissant()	Trier les images du diaporama par ordre de rang croissant	Tri les images du diaporama par ordre de rang croissant en utilisant la méthode du tri bulle			
ajouterImageEnFin(ImageDans Diaporama* pImage)	Sous-programme pour ajouter une image à la fin du diaporama	Ajoute une image à la fin du vecteur "Images" contenant les images du diaporama			

enleverImageEnFin()	Sous-programme pour enlever la dernière image du diaporama	Enlève l'image se trouvant en dernière position du vecteur "Images" du diaporama
nblmages()	nombre d'images	Permet de savoir combien d'image se trouve dans le diaporama
vider()	vide le diaporama	Permet de vider donc enlever toute les image composant le diaporama
charger()	Charge les image du diaporama	Charge toute les images associées au diaporama dans l'attribut images

Classe Lecteur		
Attributs:		
<u>Intitulé</u>	<u>Signification</u>	<u>But</u>
mode	Représente le mode du lecteur	Savoir comment doit fonctionner le lecteur (manuel ou automatique)
posImageCourante	Position de l'image - représentée par une ImageDansDiapo dans chaque instance de Diaporama - dans m_tabImages	Savoir dans quel position du tableau des images on se trouve
diaporama	Pointeur vers le diaporama courant chargé dans le lecteur	Permet de récupérer le diaporama voulu et de le manipuler
idDiaporama	Numéro identifiant du diaporama courant	Permet d'attribuer un identifiant unique à chaque diaporama pour tous les différencier
ImageCourante	Pointeur vers l'image courante du diaporama courant chargé par le lecteur	Permet de savoir sur quel Image nou somme positionner
modes_lecteur	Enuméré listant les différents modes du lecteur	Lister les différents modes dans lesquels le lecteur peut être. Peut prendre les valeurs : VIDE, MANUEL, AUTOMATIQUE
Méthodes :		
<u>Intitulé</u>	<u>Signification</u>	<u>But</u>
setMode()	setteur de mode	Définir la nouvelle valeur de l'attribut mode
getMode()	getteur de m_mode	Récupérer la valeur de m_mode
avancer()	Passe à l'image suivante du diaporama courant	Permet de passer à l'image suivant
reculer ()	Passe à l'image précédente du diaporama courant	Permet de passer à l'image précédente
changerDiaporama()	Change de diaporama	Permet de changer le diaporama courant à un autre diaporama
getIdDiaporama()	getteur de idDiaporama	Récupérer la valeur de idDiaporama
getDiaporama()	getteur de Diaporama	Récupérer la valeur de Diaporama

getPosImageCourante()	getteur de poslmageCourante	Récupérer la valeur de poslmageCourante
nbImages()	nombre d'image dans le lecteur	Permet de connaître le nombre total d'image charger dans le lecteur
setIdDiaporama()	setteur de idDiaporama	Définir la nouvelle valeur de l'attribut idDiaporama
setDiaporama()	setteur de Diaporama	Définir la nouvelle valeur de l'attribut Diaporama
setPosImageCourante()	setteur de poslmageCourante	Définir la nouvelle valeur de l'attribut poslmageCourante
viderLecteur()	Vide le lecteur	Permet de retirer le diaporama courant du lecteur et de repasser le lecteur en mode VIDE
chargerDiaporamaCourant()	Charge le diaporama courant	Charger le diaporama courant et passe le lecteur en mode MANUEL

	Classe Presentation	
Attributs:		
<u>Intitulé</u>	Signification	<u>But</u>
_leModele	Pointeur vers Lecteur	Créer un lien entre le modèle (Lecteur) et Presentation
_laVue	Pointeur vers Lecteurvue	Créer un lien entre la vue (LecteurVue) et Presentation
mapTexteLabelsVue	Dictionnaire	Dictionnaire
Méthodes :		
<u>Intitulé</u>	Signification	<u>But</u>
getModele()	getteur de _leModele	Récupérer (retourne) la valeur de _leModele
getVue()	getteur de _leVue	Récupérer (retourne) la valeur de _laVue
setModele(Lecteur *m)	setteur de _laVue	Définir la valeur de l'attribut _leModele selon la valeur passée en argument
setVue(LecteurVue *v)	setteur de _laVue	Définir la valeur de l'attribut _laVue selon la valeur passée en argument
slotDemandeReculer()	Slot pour la réception du signal de demande correspondant émis par l'instance de LecteurVue connecté	Effectuer les actions nécessaires avec le modèle pour l'action dédiée au slot. Recoit le signal de demande "signalDemandeReculer" correspondant émit par l'instance de LecteurVue connectée.
slotDemandeAvancer()	Slot prévu pour la réception du signal correspondant émis par l'instance de LecteurVue connecté	Effectuer les actions nécessaires avec le modèle pour l'action dédiée au slot. Recoit le signal de demande "signalDemandeAvancer" correspondant émit par l'instance de LecteurVue connectée.

slotDemandeLancer()	Slot prévu pour la réception du signal correspondant émis par l'instance de LecteurVue connecté	Effectuer les actions nécessaires avec le modèle pour l'action dédiée au slot. Recoit le signal de demande "signalDemandeLancer" correspondant émit par l'instance de LecteurVue connectée.
slotDemandeArreter()	Slot prévu pour la réception du signal correspondant émis par l'instance de LecteurVue connecté	Effectuer les actions nécessaires avec le modèle pour l'action dédiée au slot. Recoit le signal de demande "signalDemande" correspondant émit par l'instance de LecteurVue connectée.
slotDemandeChargerDiapo()	Slot prévu pour la réception du signal correspondant émis par l'instance de LecteurVue connecté	Effectuer les actions nécessaires avec le modèle pour l'action dédiée au slot. Recoit le signal de demande "signalDemandeChargerDiapo" correspondant émit par l'instance de LecteurVue connectée.
slotDemandeEnleverDiapo()	Slot prévu pour la réception du signal correspondant émis par l'instance de LecteurVue connecté	Effectuer les actions nécessaires avec le modèle pour l'action dédiée au slot. Recoit le signal de demande "signalDemandeEnleverDiapo" correspondant émit par l'instance de LecteurVue connectée.
slotDemandeDefVitesse(unsigned int vitesse)	Slot prévu pour la réception du signal correspondant émis par l'instance de LecteurVue connecté	Effectuer les actions nécessaires avec le modèle pour l'action dédiée au slot. Recoit le signal de demande "signalDemandeDefVitesse" correspondant émit par l'instance de LecteurVue connectée.
realiserMajTexteLabelsVue()	Sous-programme permettant de demander la mise à jour du texte des étiquettes de la vue	Permet de déterminer le nouveau texte à afficher pour les étiquettes de la vue puis à en demander l'affichage.

Classe LecteurVue		
Attributs:		
<u>Intitulé</u>	<u>Signification</u>	<u>But</u>
labelMode	Pointeur vers un widget étiquette (classe qLabel)	Affiche le mode actuel du lecteur, défini selon l'enumération modes_lecteur. Affiche "lecteur vide" si le lecteur est vide.
labelVitesse	Pointeur vers un widget étiquette (classe qLabel)	Affiche la vitesse de défilement du diaporama courant chargé par le lecteur
mapTexteLabels	qMap contenant le texte mis à jour chaque étiquette de la vue	Associe chaque étiquette de la vue en tant que clé à son texte courant à appliquer à chaque étiquette
Méthodes:		
<u>Intitulé</u>	<u>Signification</u>	<u>But</u>
LecteurVue(QWidget *parent = nullptr)	Constructeur par défaut créé par Qt	Créer une nouvelle instance de LecteurVue en y associant optionnelement un parent
~LecteurVue()	Destructeur	Supprime une instance de LecteurVue
nvlleConnexion(QObject *c)	Sous-programme de connexion des signaux de demande d'une instance de LecteurVue aux slots d'une instance de Presentation passée en paramètres	Connecter les signaux de demande (signalDemande()) d'une instance de LecteurVue aux slots de réception attirés à cette demande (slotDemande()) d'une instance de Presentation
supprConnexion(QObject *c)	Sous-programme de déconnexion des signaux de demande d'une instance de LecteurVue aux slots d'une instance de Presentation passée en paramètres	Déconnecter les signaux de demande (signalDemande()) d'une instance de LecteurVue aux slots de réception attirés à cette demande (slotDemande()) d'une instance de Presentation

majInterface (Lecteur::modesLecteur m)	Sous-programme qui détermine la mise à jour d'interface à effectuer en fonction du mode du lecteur passé en paramètres	Déterminer quel sous-programme appelé pour effectuer la mise à jour de l'interface de la vue en fonction du mode du lecteur
majInterface_lecteurVide()	Sous-programme de mise à jour d'interface dans le cas où le lecteur est vide	Mettre à jour les attributs des éléments de l'interface de la vue pour le cas où le lecteur est vide
majInterface_lecteurManuel()	Sous-programme de mise à jour d'interface dans le cas où le lecteur est en mode manuel	Mettre à jour les attributs des éléments de l'interface de la vue pour le cas où le lecteur est en mode manuel
majInterface_lecteurAutomatique()	Sous-programme de mise à jour d'interface dans le cas où le lecteur est en mode automatique	Mettre à jour les attributs des éléments de l'interface de la vue pour le cas où le lecteur est en mode automatique
majInterface_texteLabels(const QMap <qstring, qstring=""> &mapTexteLabels)</qstring,>	Sous-programme de mise à jour du texte des étiquettes de l'interface de la vue en fonction de la qMap passée en paramètre	Mettre à jour les attributs des étiquettes de l'interface de la vue avec le texte associé contenu dans la qMap passée en paramètre
btnReculer_clic()	Slot du signal de clic du bouton "Reculer"	Permet lors du clic du bouton "Reculer" d'émettre un signal
btnAvancer_clic()	Slot du signal de clic du bouton "Avancer"	Permet lors du clic du bouton "Avancer" d'émettre un signal
btnLancer_clic()	Slot du signal de clic du bouton "Lancer le diaporama"	Permet lors du clic du bouton "Lancer le Diaporama" d'émettre le signal de demande correspondant à la présentation
btnArreter_clic()	Slot du signal de clic du bouton "Arrêter le diaporama"	Permet lors du clic du bouton "Arrêter le Diaporama" d'émettre le signal de demande correspondant à la présentation
fichierChargerDiapo_declenche()	slot du signal de clic de l'action "Charger un diaporama"	Permet lors du clic de l'action "Charger un diaporama" d'émettre le signal de demande correspondant à la présentation
fichierEnleverDiapo_declenche()	slot du signal de clic de l'action "Enlever le diaporama"	Permet lors du clic de l'action "Enlever le diaporama"

		d'émettre le signal de demande correspondant à la présentation
fichierQuitter_declenche()	slot du signal de clic de l'action "Quitter"	Permet lors du clic de l'action "Quitter" d'émettre le signal de demande correspondant à la présentation
paramVitUn_declenche()	slot du signal de clic de l'action "1 seconde" dans "vitesse de défilement"	Permet lors du clic de l'action "1 seconde" d'émettre le signal de demande correspondant à la présentation
paramVitDeux_declenche()	slot du signal de clic de l'action "2 secondes" dans "vitesse de défilement"	Permet lors du clic de l'action "2 secondes" d'émettre le signal de demande correspondant à la présentation
paramVitTrois_declenche()	slot du signal de clic de l'action "3 secondes" dans "vitesse de défilement"	Permet lors du clic de l'action "3 secondes" d'émettre le signal de demande correspondant à la présentation
paramVitPerso_declenche()	slot du signal de clic de l'action "Vitesse Personnalisée" dans "vitesse de défilement"	Permet lors du clic de l'action "Vitesse Personnalisée" d'émettre le signal de demande correspondant à la présentation
aideAPropos_declenche()	slot du signal de clic de l'action "A propos de"	Permet lors du clic de l'action "A propos de" d'émettre le signal de demande correspondant à la présentation
signalDemandeReculer()	signal emit par btnReculer_clic()	Signal émit par btnReculer_clic() destiné à être reçu dans le slot correspondant attitré dans l'instance de Presentation connectée.
signalDemandeAvancer()	signal emit par btnAvancer_clic()	Signal émit par btnAvancer_clic() destiné à être reçu dans le slot correspondant attitré dans l'instance de Presentation connectée.
signalDemandeLancer()	signal emit par btnLancer_clic()	Signal émit par btnLancer_clic() destiné à être reçu dans le slot correspondant attitré dans l'instance de Presentation connectée.

signalDemandeArreter()	signal emit par btnArreter_clic()	Signal émit par btnArreter_clic() destiné à être reçu dans le slot correspondant attitré dans l'instance de Presentation connectée.
signalDemandeChargerDiapo()	signal emit par fichierChargerDiapo_declenche ()	Signal émit par fichierChargerDiapo_declench e() destiné à être reçu dans le slot correspondant attitré dans l'instance de Presentation connectée.
signalDemandeEnleverDiapo()	signal emit par fichierEnleverDiapo_declenche ()	Signal émit par fichierEnleverDiapo_declench e() destiné à être reçu dans le slot correspondant attitré dans l'instance de Presentation connectée.
signalDemandeDefVitesse(unsign ed int vitesse)	signal emit par fichierQuitter_declenche()	Signal émit par fichierQuitter_declenche() destiné à être reçu dans le slot correspondant attitré dans l'instance de Presentation connectée.

10. Bilan sur cette SAE

Voici ci-dessous notre retour commun sur cette SAE:

- Ce que vous avez appris / développé :

Version 1:

Cette version nous a permis d'améliorer nos connaissances en termes de conception de classes. Elle nous a également permis d'apprendre à "transformer" une application réalisée en programmation procédurale en une application réalisée en programmation objet.

Version 2:

Nous avons développé comment disposer les éléments graphiques sur une interface ainsi que d'effectuer des connexions de signal à slots.

Version 2 MVP:

Nous avons appris à implémenter la structure du patron de conception MVP (Modèle - Vue - Présentation).

Ce qui a été difficile :

Version 1:

Nous avons eu du mal à interpréter le sujet, ce qui a eu pour conséquence une sur-complication de notre travail, nous avons donc dû retoucher à l'utilisation de tableaux, à la base nous utilisions des tableaux dynamiques pour stocker les pointeurs vers les classes car on ne savait pas si on avait le droit d'utiliser les vecteurs "vector" c++ de la v0 car on ne les a pas vu en cours, il y avait donc un manque de transparence à ce niveau-là selon nous.

Version 2:

Correctement utiliser les layouts a été plutôt compliqué au début. Mise à part ça, nous n'avons pas eu de grandes difficultés.

Version 2 MVP:

Nous avons eu quelques difficultés pour la communication Modèle - Présentation.

Le temps passé :

Version 1:

Comme mentionné précédemment, nous avons perdu plusieurs heures à cause du problème d'utilisation de tableaux au lieu de vecteur. Nous avons passé une vingtaine d'heures sur cette version.

Version 2:

Nous avons passé une dizaine d'heures sur cette version.

Version 2 MVP:

Nous avons passé une dizaine d'heures sur cette version. La documentation nous a pris une assez grande partie du temps.

- Ce que vous auriez pu faire mieux :

Version 1:

Nous aurions pu mieux nous organiser, c'est-à-dire commencer à travailler plus tôt afin d'avoir le temps de se corriger si on part dans une mauvaise direction en se renseignant.

Version 2:

Nous n'avons pas simplement idée de comment nous aurions pu faire mieux pour le moment.

Version 2 MVP:

Nous ne sommes pas entièrement certains de notre démarche sur la façon de mettre à jour la vue depuis la présentation, elle manque peut-être d'optimisation et/ou est incomplète.de Permet lors du clique du bouton Lancer un Diaporama d'émettre un signal