

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>

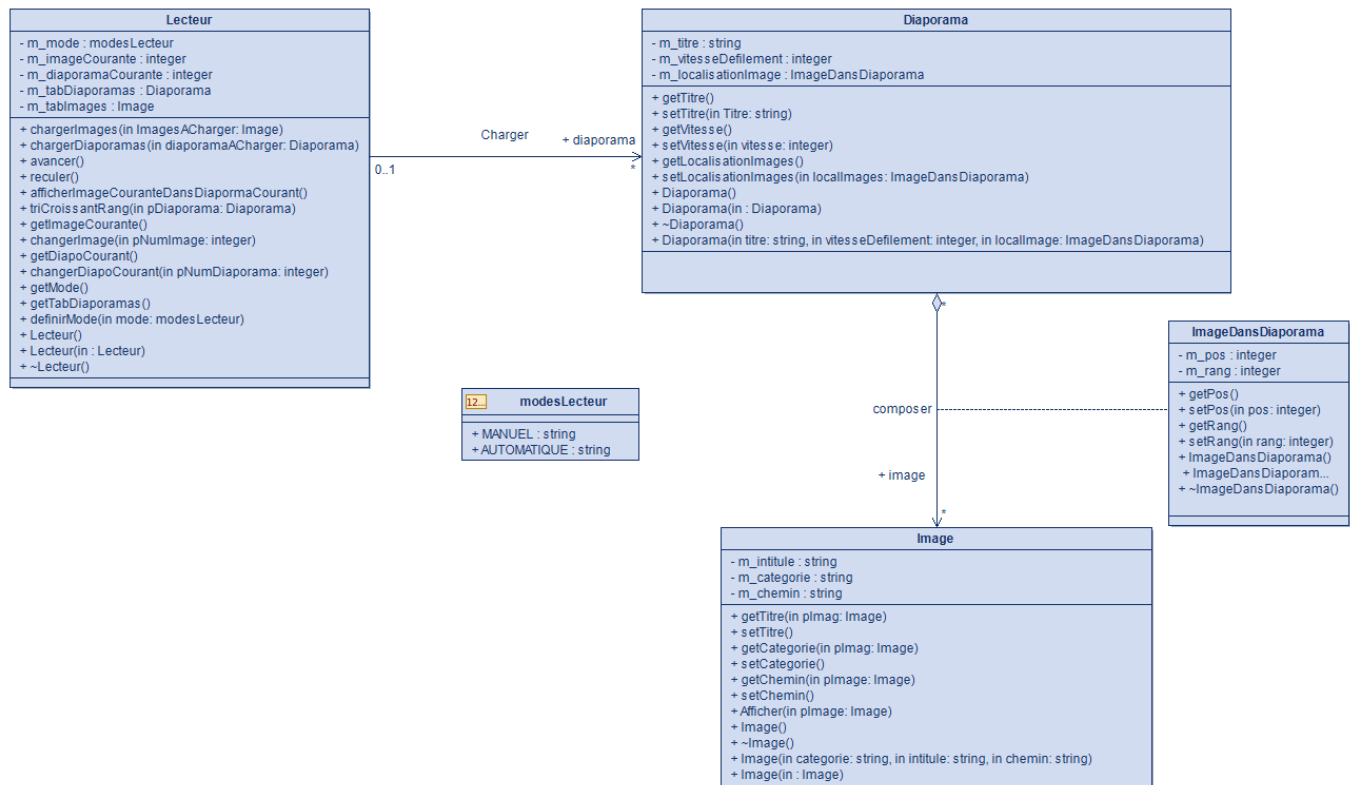
## S2.01 - Développement d'une application

Dossier d'analyse et conception de la v1 & v2

# Sommaire

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

# 1. Diagramme de classes UML



## 2.Documentation des classes

Classe Image		
<u>Attributs :</u>		
<u>Intitulé</u>	<u>Signification</u>	<u>But</u>
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(Image)	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)**

Attributs:

<u>Intitulé</u>	<u>Signification</u>	<u>But</u>
m_pos	position de l'image dans le tableau des images	Pouvoir ranger 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

Méthodes :

<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 informations d'une autre instance source

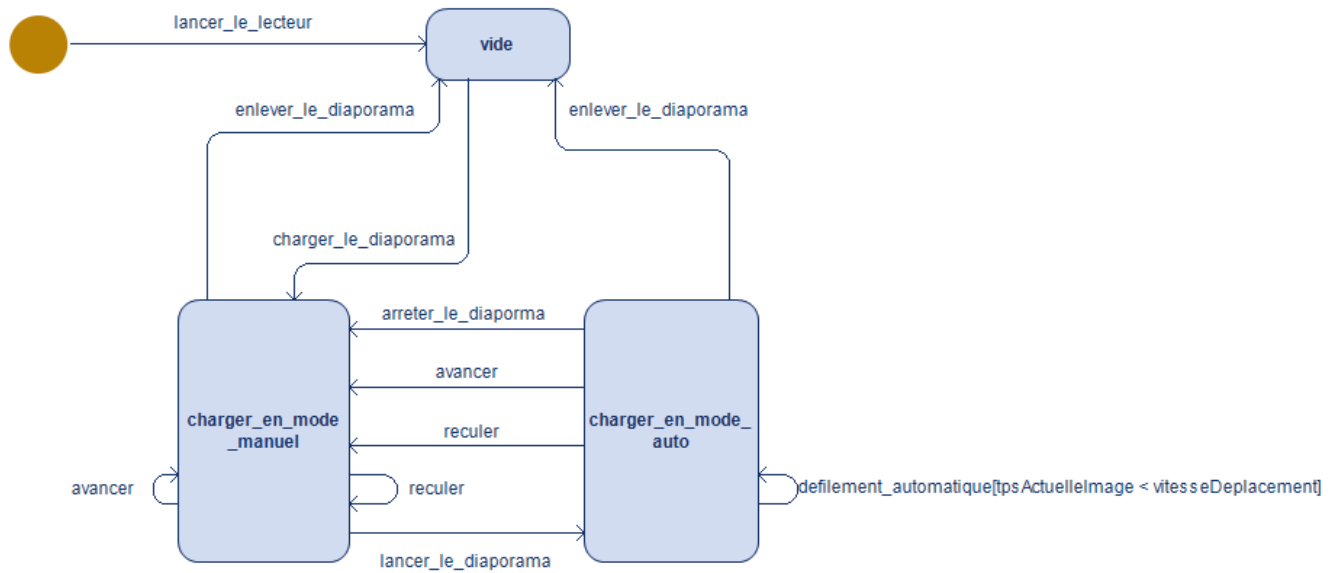
Classe Diaporama		
<u>Attributs:</u>		
<u>Intitulé</u>	<u>Signification</u>	<u>But</u>
m_titre	Titre du diaporama	Donner un nom (titre) au diaporama
m_vitesseDefilement	Vitesse d'affichage du diaporama	Donner une vitesse d'affichage au diaporama
m_LocalisationImage	Tableau contenant les image du diaporama	Stocker toute les images qui vont composer le diaporama
<u>Méthodes :</u>		
<u>Intitulé</u>	<u>Signification</u>	<u>But</u>
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 instance avec les information d'un autre instance source
Diaporama(titre : string, vitesseDefilement : int, locallmage : ImageDansDiaporama )	Constructeur avec définition de la valeur de tous les attributs	Créer une nouvelle instance de Diaporama et pouvoir préciser chaque attribut

Classe Lecteur		
<u>Attributs:</u>		
<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_tabImages	Tableau contenant tous les Images	Pour utiliser le tableau pour ensuite afficher l'image
<u>Méthodes :</u>		
<u>Intitulé</u>	<u>Signification</u>	<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 précédente	Permet de passer à l'image précédente
~Lecteur()	Destructeur de l'instance	Supprime une instance de Lecteur
Lecteur()	Constructeur par défaut	Créer une nouvelle instance de 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

triImageCroissantRang()	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
afficherImageCouranteDansDiaporamaCourant()	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	BoutonRecul er	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"**

Charger un diaporama (*actionFichierChargerDiapo*) : Permet de charger un diaporama dans le lecteur, cela correspond à l'événement Charger\_le\_diaporama.

Enlever un diaporama (*actionFichierEnleverDiapo*) : 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.

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

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

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

Vitesse personnalisée : 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"

A propos de (*actionAideAPropos*): 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 :

Diaporama Courant (*labelDiapoCourant*) : Étiquette dont le texte affiche l'intitulé de diaporama courant.

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

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

Intitule et catégorie (*labelImage*) : 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 :

Avancer (*btnAvancer*): Dans les versions futures, permet de passer à l'image suivante dans le lecteur. Si le lecteur est en mode automatique, définit 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.

Reculer (*btnReculer*): Dans les versions futures, permet de passer à l'image précédente dans le lecteur. Si le lecteur est en mode automatique, définit 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.

Lancer le diaporama (*btnLancer*): 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.

Arrêter le diaporama (*btnArreter*): Dans les versions futures, arrête 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.

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

## 8. 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 :**

Pas de grandes difficultés.

- 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.

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