Université de Pau et des Pays de l'Adour - IUT DE BAYONNE - PAYS-BASQUE année 2022-2023

Département Informatique saé 20.1

Saé 2.01 – Développement d’une application

Lecteur de diaporamas – Dossier d’Analyse et conception

[Compléments de spécifications externes. 2](#_70sejrur0mkq)

[Scénarios 2](#_jjoz6pfh3yyr)

[scénario lancer diaporama 3](#_aeu47cs7wfnq)

[Diagramme de classe (UML) 4](#_wao3xkyzk6fi)

[Le diagramme de classes UML 4](#_msx5t6w6ahcg)

[Dictionnaire des éléments 5](#_1mu1f96ypno4)

[liste des dossiers 7](#_n9lpzh7wxhiu)

[Liste et rôle des fichiers de cette version 8](#_kjzjzrrr7j4d)

[signal est SLOT (dans lecteurVue.h) 9](#_d1h7xxaplylr)

[**Les fichiers 9**](#_n2xaqqe6xahs)

[entêtes 9](#_du4bnfpnglbb)

[database 10](#_ncpspadfmk8u)

[demandeDiaporama 10](#_g4ugjl44zqv1)

[image 11](#_ks4agx6svg7z)

[info 11](#_276caj8r94rv)

[lecteurVue 12](#_uyc1zmlqqsow)

[modif\_diaporama 13](#_65ysnmdw70hq)

[vit 14](#_ngg74sauro0p)

[**interface graphique 15**](#_hqgeczconxwd)

[model QT 15](#_8qcbrrlhamzd)

[Prévisualisation 16](#_nhoohczfdnyw)

[teste 16](#_wbvccqgnfuzg)

[main 16](#_er9em5557h2m)

[résultats attendu 17](#_wx8ugxng5khp)

[résultat au finale 19](#_i4bmh34w8mum)

[Diagramme états-transitions-actions 19](#_rk55zpwxa1yn)

[le diagramme 19](#_yhkrb27orhm2)

[Dictionnaire des états, événements et Actions 20](#_kfjlgbv9b0tw)

[Dictionnaire des états du diaporama 20](#_2vhi5ihneu4l)

[Dictionnaire des événements faisant changer le diaporama d’état 21](#_o69ago7s5293)

[Bilan 22](#_ccyj8ykuh0f9)

[Dépôt Git où trouver le projet complet 22](#_21tk7djf4uix)

[Temps global de travail 22](#_ihkl6pa4gjx)

[Apprentissages majeurs 23](#_r34upwa9o1m7)

[Difficultés majeures 23](#_1chrgsgav1x)

[Points positifs / négatifs de l’activité 23](#_vjrivoo1se5i)

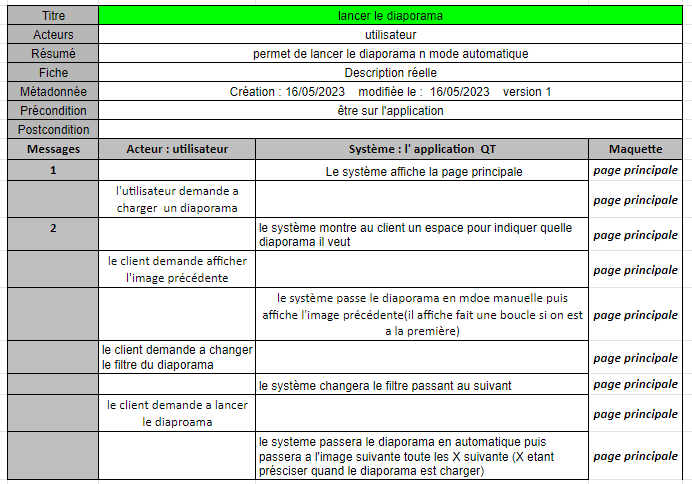
# Compléments de spécifications externes.

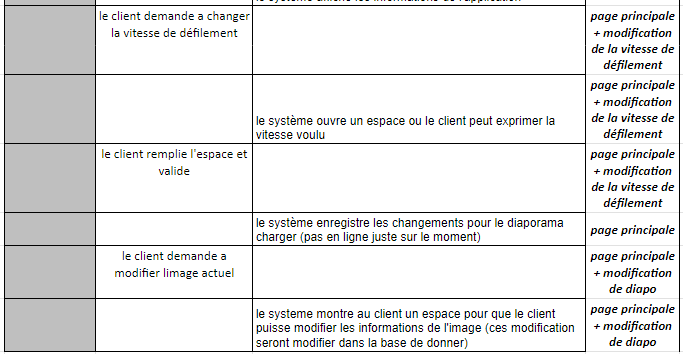
l’application devra pouvoir faire un filtre;filtrant les images du diaporama par catégorie : “personnage”,”animal”,”objet”,”tous”,ce mode a été rajouté pour pouvoir avoir le diaporama sans filtre.  
Ceci va être appliqué grâce à un bouton qui permettra de switcher entre les modes. Tu appuie sur le bouton sa change de mode.

## 

# Scénarios

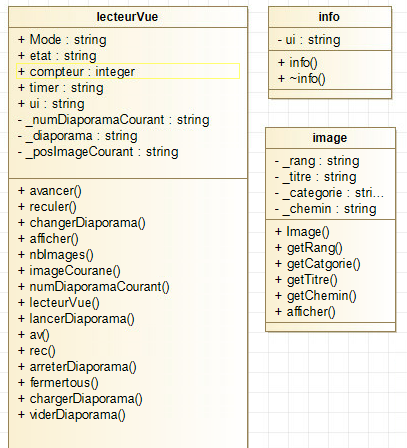
## Scénario lancer diaporama





# Diagramme de classe (UML)

## Le diagramme de classes UML



## Dictionnaire des éléments

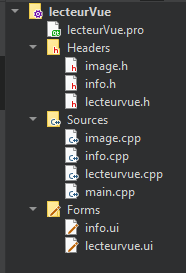
|  |  |  |  |
| --- | --- | --- | --- |
| **Classe image** | | | |
| **Nom attribut** | **Signification** | **Type** | **Exemple** |
| rang | *rang* *de* *l'image* *au* *sein* *du* *diaporama*  *auquel* *l'image* *est* *associée* | unsigned int | 1 |
| **\_titre** | *intitulé* *de* *l'image* | string | “disney.git” |
| **\_categorie** | *catégorie de l'image (personne, animal, objet)* | string | “objet” |
| **\_chemin** | *chemin complet vers le dossier où se trouve l'image/ pas a afficher* | string | “/carte/disney.git” |

|  |  |  |  |
| --- | --- | --- | --- |
| **Classe info** | | | |
| **Nom attribut** | **Signification** | **Type** | **Exemple** |
| ui | le pointeur vers la partie graphique | Ui |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **Classe lecteurVue** | | | |
| **Nom attribut** | **Signification** | **Type** | **Exemple** |
| Mode | l’enum listant les modes de l’application | enum |  |
| lst\_categorie[4] | la liste contenant les different catégorie d’image possible | string |  |
| **cat\_actuelle** | c’est un compteur ou indicateur pour la catégorie actuelle | int | 0 |
| CategorieImageCourant | la variable contenant la categorie d’image actuelle pour le diaporama | string | “tous” |
| etat | la variable indiquant l'état de l’application | Mode | manuel |
| compteur | un compteur permettant de compter les seconde pour faire défiler les images | int | 0 |
| timer | l’objet QTimer permettant de faire le timer | QTimer |  |
| **vitesse\_defilement** | l’entier qui indique la vitesse de défilement du diaporama | int | 2 |
| **chemin** | la chaine de caractère du chemin pour les images | string | F:\\ecole\\SAE\\SAE S2.01 |

Tous les diffèrent dossiers/fichiers utilisés

## Liste des dossiers

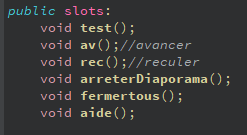


## Liste et rôle des fichiers de cette version

|  |  |
| --- | --- |
| Nom du fichier |  |
| image.h | Spécification de la classe Image |
| info.h | Spécification de la classe info |
| lecteurVue.h | Spécification de la classe lecteurVue |
| Image.cpp | Corps de la classe database |
| info.cpp | Corps de la classe database |
| lecteurVue.cpp | Corps de la classe lecteurVue |
| main.cpp | Teste les méthodes de la classe Lecteur |
| Info.ui | Fenêtre graphique contenant les informations qui doivent apparaitre lors du clique sur le bouton associé |
| Lecteurvue.ui | Fenêtre principale ou vas défiler les diaporamas |

### 

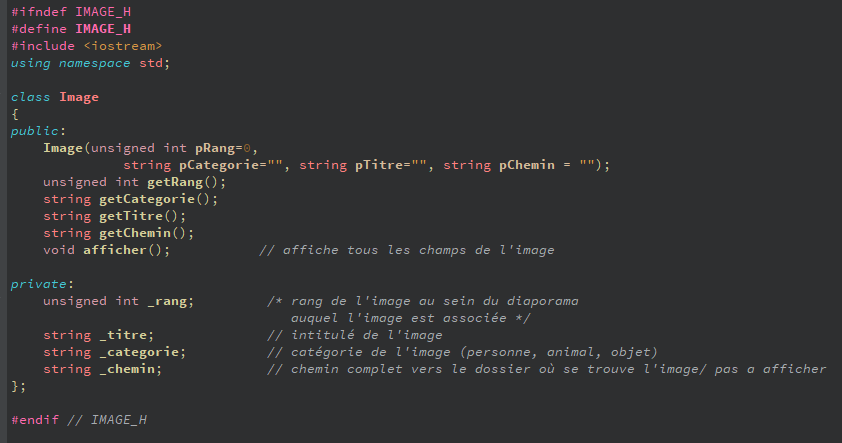
## signal est SLOT (dans lecteurVue.h)

**

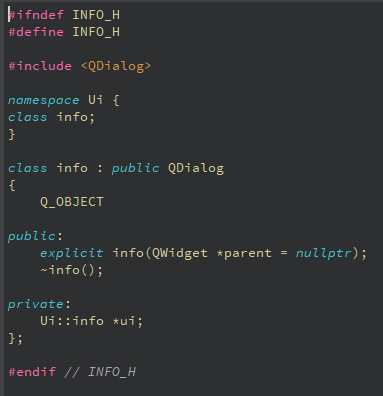
# Les fichiers

## Les entêtes des fichiers

### Fichier image.h

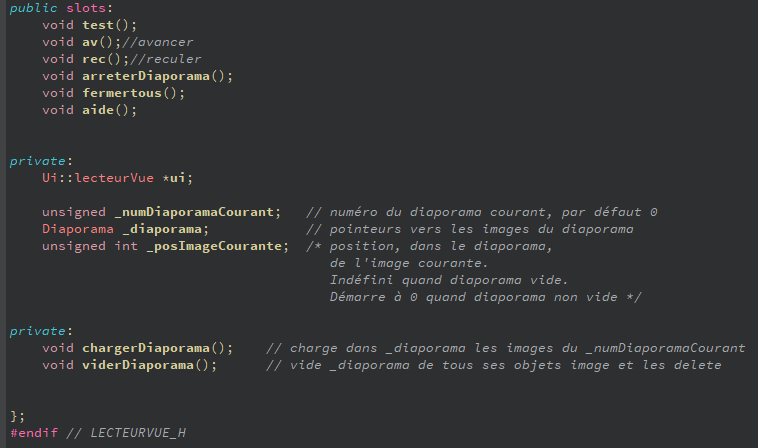


### Fichier info.h



### Fichier lecteurVue.h

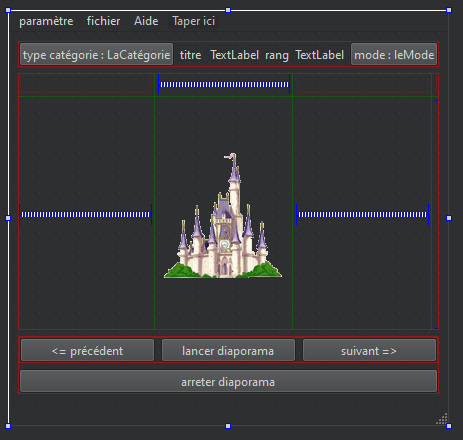




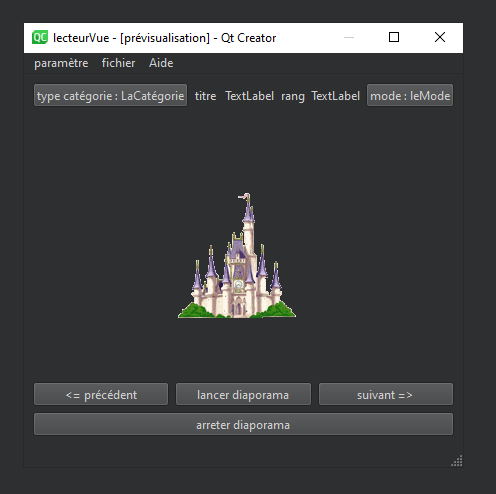
# Interface graphique

## Model QT dans le fichier.ui

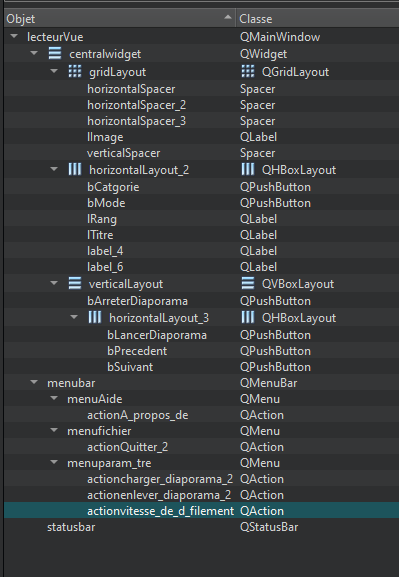
### 

**

## Prévisualisation

**

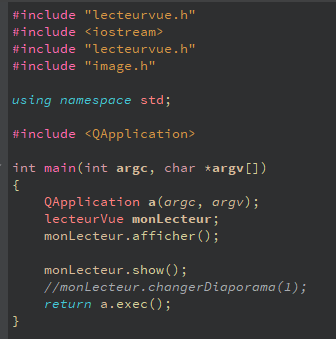
## élément de l’interface graphique



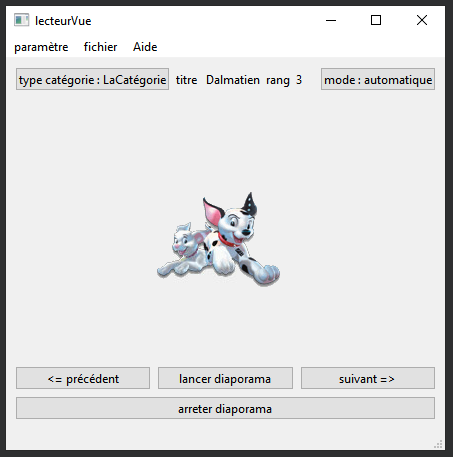
### 

# teste

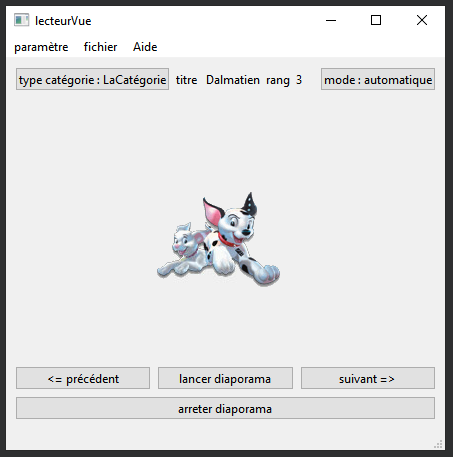
## main



## résultats attendu



## résultat au finale

**

# 

## 

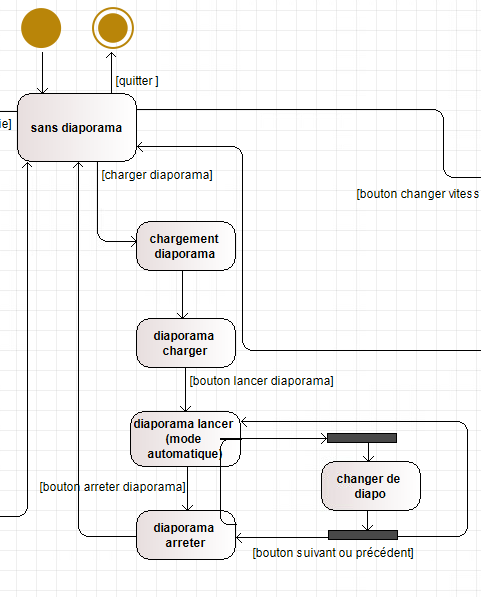
## 

## 

## 

# Diagramme états-transitions-actions

## le diagramme

**

## 

## Dictionnaire des états, événements et Actions

### **Dictionnaire** **des états du diaporama**

|  |  |
| --- | --- |
| *nomEtat* | *Signification* |
| sans diaporama | l’etat qu’est le lecteur quand il est lancer |
| diaporama charger | l'etat ou le lecteur à un diaporama charger |
| diaporama lancer | l’etat ou le diaporama est lancer en mode automatique |
| diaporama arrêter | l’etat ou le diaporama est en mode manuel |
| changer de diapo | l”etat ou le diaporama va changer de diapositive en mode manuel ou automatique |

### Dictionnaire des événements faisant changer le diaporama d’état

|  |  |
| --- | --- |
| *nomEvénement* | *Signification* |
| quitter | si le bouton quitter est pressé |
| charger diaporama | si le bouton charger diaporama est pressé |
| bouton lancer diaporama | si le bouton lancer diaporama est pressé |
| bouton arrêté diaporama | si le bouton arrêter diaporama est pressé |
| bouton suivant ou précédent | si l’un des 2 bouton est pressé |

### 

# Bilan

## Dépôt Git où trouver le projet complet

## Temps global de travail

Environs 5 h 30 pour création de cette version en partant de la précédente

## Apprentissages majeurs

Création et gestion d’un timer

## Difficultés majeures

Difficulté de compression pour coder le timer et bien le mettre en place

## Points positifs / négatifs de l’activité

Pour cette version rien de négatif, pour ce qui est des points positifs, amélioration de la compréhension pour coder un timer