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)

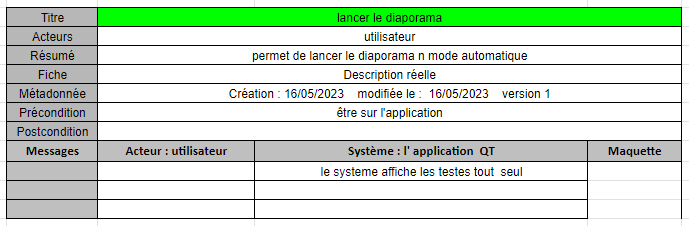
# Compléments de spécifications externes.

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.

## 

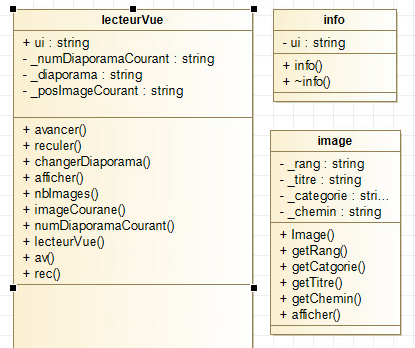
# Scénarios

## scénario lancer diaporama



# Diagramme de classe (UML)

## Le diagramme de classes UML



## 

## Dictionnaire des éléments

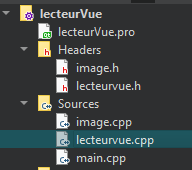
|  |  |  |  |
| --- | --- | --- | --- |
| **Classe image** | | | |
| **Nom attribut** | **Signification** | **Type** | **Exemple** |
| rang | *rang* *de* *l'image* *au* *sein* *du* *diaporama*  *auquel* *l'image* *est* *associée* | unsigned int | 1 |
| **\_titre** | *intitulé* *de* *l'image* | string | “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** |
| ui | le pointeur vers la présentation | Ui |  |
| \_numDiaporamaCourant | le numero du diaporama actuel | int | 1 |
| \_diaporama | la liste des diapo | Image |  |
| \_posImageCourant | entier représentant la position de l’image courante dans \_diaporama | int | 1 |

different dossiers

## liste des dossiers

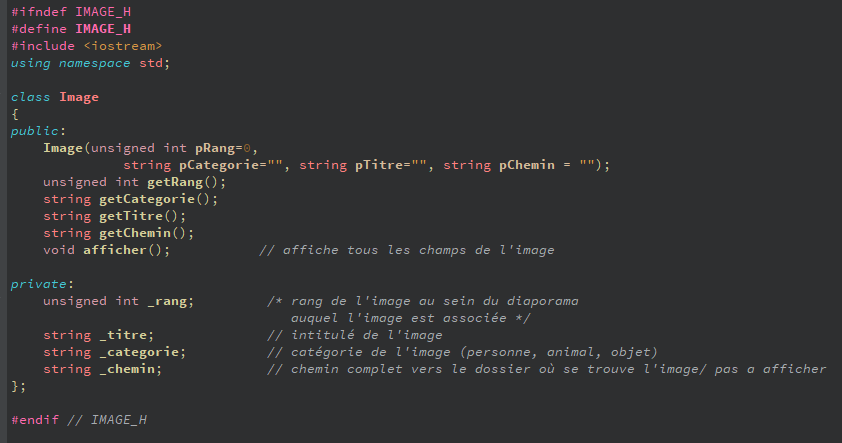


## 

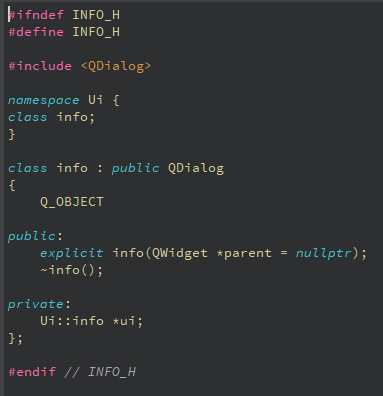
## Liste et rôle des fichiers de cette version

|  |  |
| --- | --- |
| Nom de la classe | Description de la classe |
| 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 |

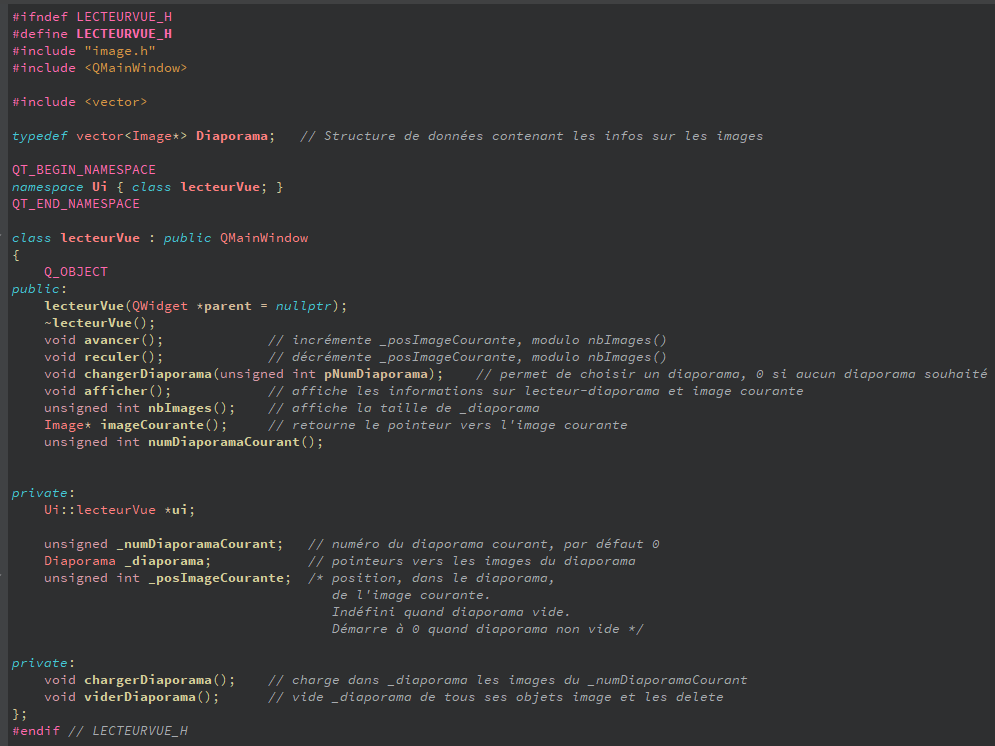
### Fichier Image.h



### Fichier Info.h



### Fichier lecteurVue.h



# 

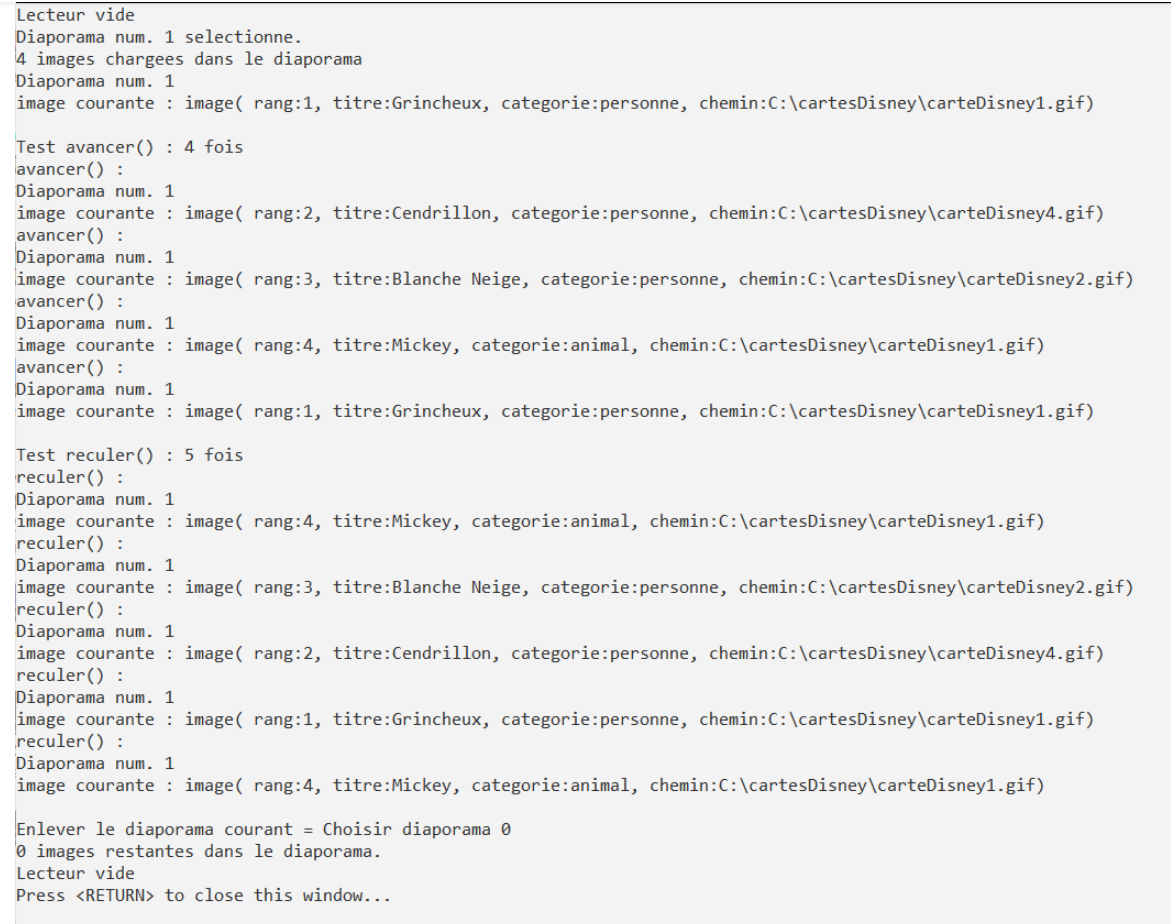
# Partie test de bon fonctionnement

## main.cpp

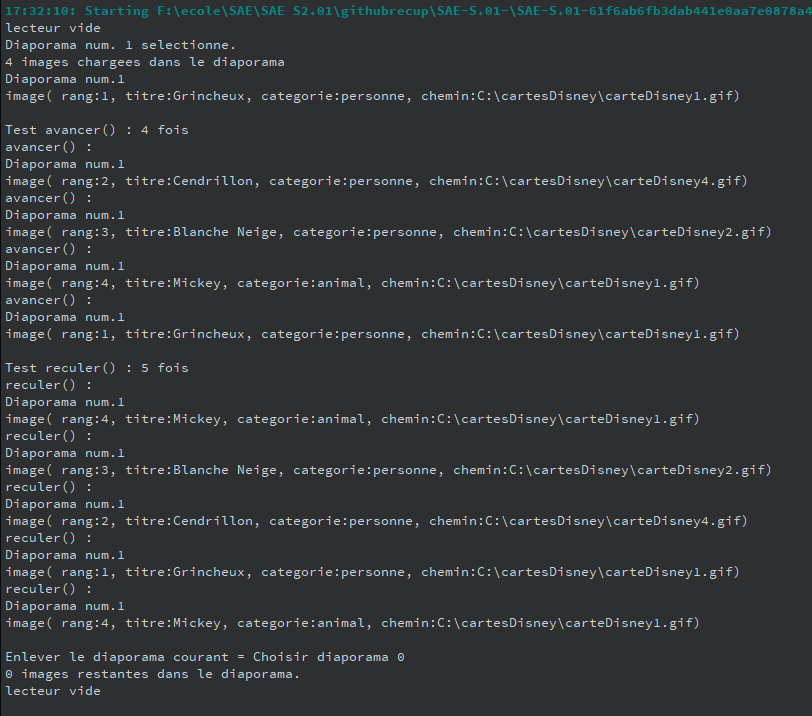


## 

## Résultats attendu après l’exécution du programme complété

**

## Résultat obtenus après avoir exécuter le code complété

**

# 

## 

## 

## 

## 

# Diagramme états-transitions-actions

## Le diagramme

*Non réalisé*

## Dictionnaire des états, événements et Actions

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

Non réalisé

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

Non réalisé

### 

# Bilan

## Dépôt Git où trouver le projet complet

## 

## Temps global de travail

Environs 3 heures pour création de cette version en partant des dossiers fournis

## Apprentissages majeurs

Codes-en c++

## Difficultés majeures

Coder le permettant de mettre les images dans l’ordre

## Points positifs / négatifs de l’activité

Points négatifs, version réalisé tout seul.

Point positif, meilleur compréhension de c++