

BUT Informatique Semestre 2 2022-2023

IUT de Bayonne et du Pays Basque, Département Informatique

S2.01 « *Développement d’une application* »

**Lecteur de diaporamas – Dossier d’analyse et de conception**

ESPINASSE Virgile, TD 1 - TP 2

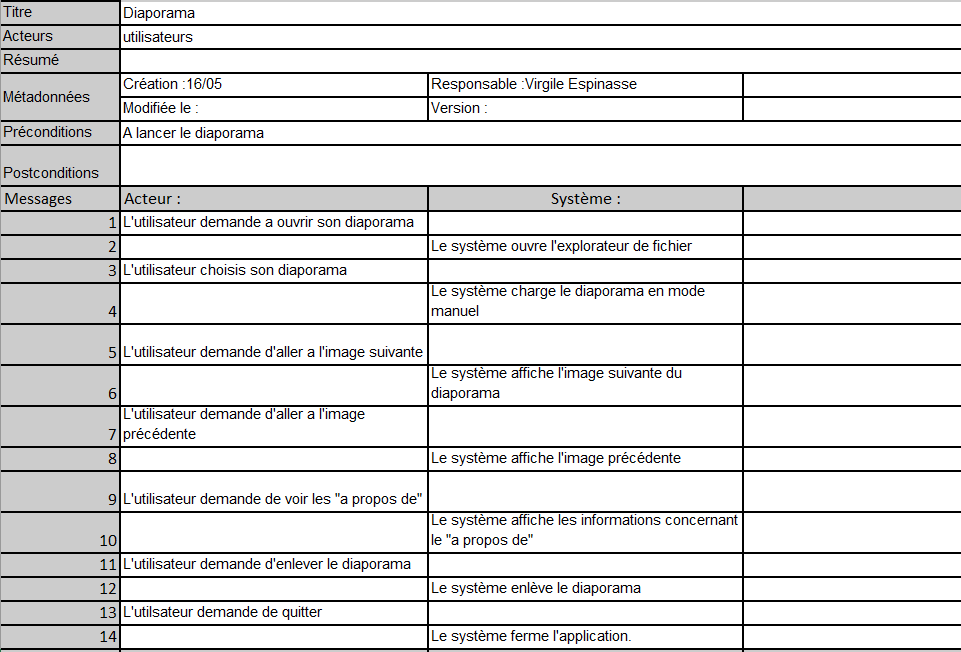
FOISSAC Mathieu, TD1 - TP 2

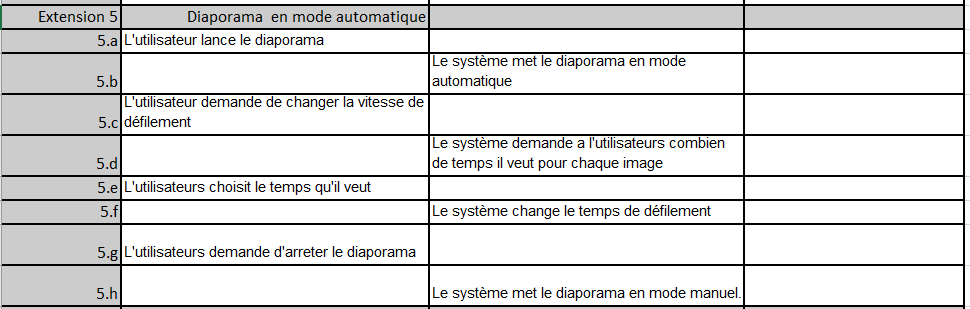
HIRIBARREN Adrien, TD 1 - TP 2

## Compléments de spécifications externes.

Il n’y a pas de compléments en plus des spécifications déjà données dans le sujet.

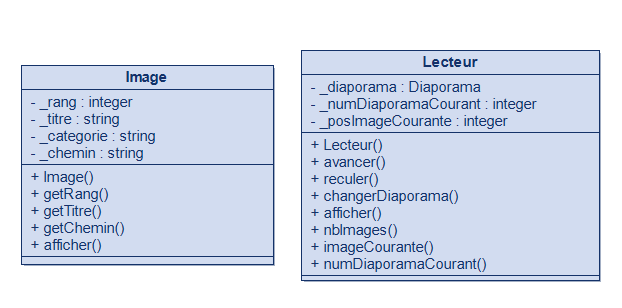
## Scénarios





# Version v0 – Version console seule

## Diagramme de classe (UML)



1. Dictionnaire des éléments pour chaque classe

|  |  |  |  |
| --- | --- | --- | --- |
| **Classe Image** | | | |
| **Nom attribut** | **Signification** | **Type** | **Exemple** |
| \_rang | Correspond au rang de l’image dans le diaporama. | unsigned int | 1 |
| \_titre | Correspond au titre de l’image. | string | “Chateau” |
| \_categorie | Correspond à la catégorie de l’image. | String | “Structure” |
| \_chemin | Correspond au chemin de l’image. | string | “/images/x.png |

|  |  |  |  |
| --- | --- | --- | --- |
| **Classe Lecteur** | | | |
| **Nom attribut** | **Signification** | **Type** | **Exemple** |
| \_diaporama | Correspond à la pile d’image du diaporama. | Vecteur<image> |  |
| \_numDiaporamaCourant | Correspond au numéro du diaporama courant. | unsigned int | 1 |
| \_posImageCourante | Correspond à la position de l’image courante dans le diaporama. | Unsigned int | 3 |

1. Dictionnaire des méthodes

Classe lecteur :

public:

Lecteur();

void avancer(); // incrémente \_posImageCourante, modulo nbImages()

void reculer(); // décrémente \_posImageCourante, modulo nbImages()

void changerDiaporama(unsigned int pNumDiaporama); // permet de choisir un diaporama, 0 si aucun diaporama souhaité

void afficher(); // affiche les informations sur lecteur-diaporama et image courante

unsigned int nbImages(); // affiche la taille de \_diaporama

Image\* imageCourante(); // retourne le pointeur vers l'image courante

unsigned int numDiaporamaCourant();

private:

void chargerDiaporama(); // charge dans \_diaporama les images du \_numDiaporamaCourant

void viderDiaporama(); // vide \_diaporama de tous ses objets image et les delete

};

Classe image :

public:

Image(unsigned int pRang=0,

string pCategorie="", string pTitre="", string pChemin = "");

unsigned int getRang(); // renvoie le rang de l’image

string getCategorie(); // renvoie la catégorie de l’image

string getTitre(); // renvoie le titre de l’image

string getChemin(); // renvoie le chemin d’accès de l’image

void afficher(); // affiche tous les champs de l'image

## Implémentation et tests

### 4.1 Implémentation

Liste et rôle des fichiers de cette version :

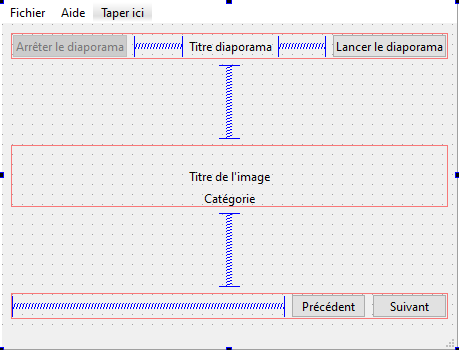
|  |  |
| --- | --- |
| lecteur.h | Spécification de la classe Lecteur |
| lecteur.cpp | Corps de la classe Lecteur |
| image.h | Spécification de la classe Image |
| image.cpp | Corps de la classe Image |
| main.cpp | Teste les méthodes de la classe Lecteur |

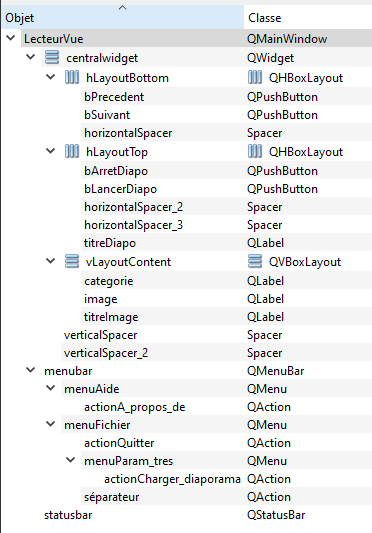
### 4.2 Test

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Classe** | **Description** | **Valeur(s)en entrée** | **Resultat(a)attendu** | **resultat obtenu** |
| valide 1 | charger le diaporama | \_numDiaporamaCourant=1 | Lecteur vide ! Diaporama num. 1 selectionne.  4 images chargees dans le diaporama Diaporama num. 1 selectionne. Image courante: image( rang:1, titre:Grincheux, categorie:personne, chemin:C:\cartesDisney\carteDisney1.gif) | Lecteur vide ! Diaporama num. 1 selectionne.  4 images chargees dans le diaporama Diaporama num. 1 selectionne. Image courante: image( rang:1, titre:Grincheux, categorie:personne, chemin:C:\cartesDisney\carteDisney1.gif) |
| valide 2 | avancer() au rang 1 |  | avancer() :  Diaporama num. 1 selectionne. Image courante: image( rang:1, titre:Grincheux, categorie:personne, chemin:C:\cartesDisney\carteDisney1.gif) | avancer() :  Diaporama num. 1 selectionne. Image courante: image( rang:1, titre:Grincheux, categorie:personne, chemin:C:\cartesDisney\carteDisney1.gif) |
| valide 3 | avancer() au rang 2 |  | avancer() :  Diaporama num. 1 selectionne. Image courante: image( rang:2, titre:Cendrillon, categorie:personne, chemin:C:\cartesDisney\carteDisney4.gif) | avancer() :  Diaporama num. 1 selectionne. Image courante: image( rang:2, titre:Cendrillon, categorie:personne, chemin:C:\cartesDisney\carteDisney4.gif) |
| valide 4 | avancer() au rang 3 |  | avancer() :  Diaporama num. 1 selectionne. Image courante: image( rang:3, titre:Blanche Neige, categorie:personne, chemin:C:\cartesDisney\carteDisney2.gif) | avancer() :  Diaporama num. 1 selectionne. Image courante: image( rang:3, titre:Blanche Neige, categorie:personne, chemin:C:\cartesDisney\carteDisney2.gif) |
| valide 5 | avancer() au rang 4 |  | avancer() :  Diaporama num. 1 selectionne. Image courante: image( rang:4, titre:Mickey, categorie:animal, chemin:C:\cartesDisney\carteDisney1.gif) | avancer() :  Diaporama num. 1 selectionne. Image courante: image( rang:4, titre:Mickey, categorie:animal, chemin:C:\cartesDisney\carteDisney1.gif) |
| valide 6 | reculer() au rang 4 |  | reculer() :  Diaporama num. 1 selectionne. Image courante: image( rang:4, titre:Mickey, categorie:animal, chemin:C:\cartesDisney\carteDisney1.gif) | reculer() :  Diaporama num. 1 selectionne. Image courante: image( rang:4, titre:Mickey, categorie:animal, chemin:C:\cartesDisney\carteDisney1.gif) |
| valide 7 | reculer() au rang 3 |  | reculer() :  Diaporama num. 1 selectionne. Image courante: image( rang:3, titre:Blanche Neige, categorie:personne, chemin:C:\cartesDisney\carteDisney2.gif) | reculer() :  Diaporama num. 1 selectionne. Image courante: image( rang:3, titre:Blanche Neige, categorie:personne, chemin:C:\cartesDisney\carteDisney2.gif) |
| valide 8 | reculer() au rang 2 |  | reculer() :  Diaporama num. 1 selectionne. Image courante: image( rang:2, titre:Cendrillon, categorie:personne, chemin:C:\cartesDisney\carteDisney4.gif) | reculer() :  Diaporama num. 1 selectionne. Image courante: image( rang:2, titre:Cendrillon, categorie:personne, chemin:C:\cartesDisney\carteDisney4.gif) |
| valide 9 | reculer() au rang 1 |  | reculer() :  Diaporama num. 1 selectionne. Image courante: image( rang:1, titre:Grincheux, categorie:personne, chemin:C:\cartesDisney\carteDisney1.gif) | reculer() :  Diaporama num. 1 selectionne. Image courante: image( rang:1, titre:Grincheux, categorie:personne, chemin:C:\cartesDisney\carteDisney1.gif) |
| valide 10 | reculer() au rang 4 |  | reculer() :  Diaporama num. 1 selectionne. Image courante: image( rang:4, titre:Mickey, categorie:animal, chemin:C:\cartesDisney\carteDisney1.gif) | reculer() :  Diaporama num. 1 selectionne. Image courante: image( rang:4, titre:Mickey, categorie:animal, chemin:C:\cartesDisney\carteDisney1.gif) |
| valide 11 | enlever le diaporama | \_numDiaporamaCourant=0 | Enlever le diaporama courant = Choisir diaporama 0 0 images restantes dans le diaporama. Lecteur vide ! | Enlever le diaporama courant = Choisir diaporama 0 0 images restantes dans le diaporama. Lecteur vide ! |

# Version v1 – projet Graphique seul

## Éléments d’interface





Pour la partie graphique de la v1, nous nous servons d’un widget principal vertical nommé centralwidget. Il est composé des trois layout hLayoutBottom qui ont les bouttons bPrédedent pour aller à l'image précédente et bSuivant pour aller à l'image suivante. Le layout hLayoutTop contient les éléments du haut, avec le boutton pour arrêter le diaporama bArretDiapo, pour lancer bLancerDiapo et le titre au milieu le QLabel titreDiapo. Le dernier layout vLayoutContent est celui du milieu. Il contient les éléments concernant les images avec le titre de l’image, le QLabel categorie, l’image image, et le titre titreImage. La barre de menu a les QMenu menuAide pour ouvrir l’aide, menuFichier pour ouvrir le choix des fichiers.

## Implémentation et tests

### 6.1 Implémentation

Liste et rôle des fichiers de cette version :

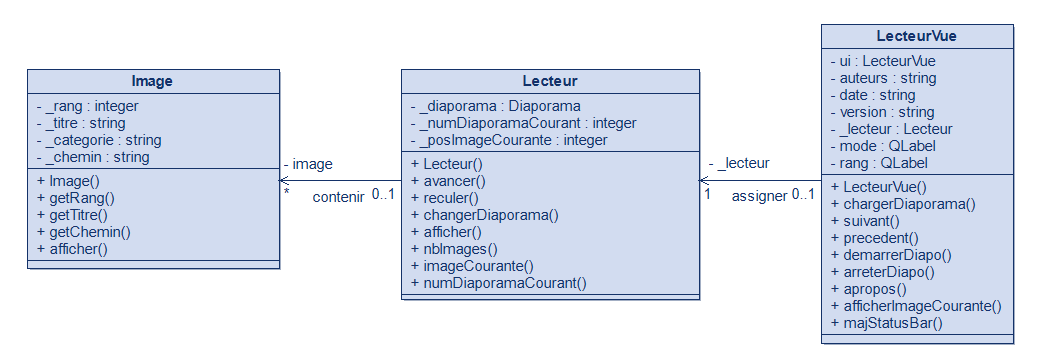
|  |  |
| --- | --- |
| lecteurVue.h | Spécification de la classe graphique Qt contenant l’interface du lecteur de diaporamas |
| lecteurVue.cpp | Corps de la classe LecteurVue |
| lecteurvue.ui | Fichier du dessin de l’interface réalisé par QtDesigner |
| main.cpp | Teste les méthodes de la classe Lecteur |

### 6.2 Test

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Classe** | **Description** | **Valeur(s)en entrée** | **Résultat(a)attendu** | **Résultat obtenu** | **Remarque** |
| valide 1 | Charge un diaporama |  | statusbar "Mode : Manuel" | statusbar "Mode : Manuel" | Message dans la console "Je charge le diaporama.". Label d'état "Mode : Manuel" |
| valide 2 | Lancement du diaporama |  | bArretDiapo->isEnabled() == true bLancerDiapo->isEnabled() == false statusbar "Mode automatique" | bArretDiapo->isEnabled() == true bLancerDiapo->isEnabled() == false statusbar "Mode automatique" | Message dans la console "Démarrage du diaporama" Bouton Lancer le diaporama désactivé, Bouton Arrêter le diaporama désactivé, label en bas à gauche "Mode : automatique" |
| valide 3 | Image suivante |  | Message dans la console "Image suivante" | Message dans la console "Image suivante" | Seulement le message en console permet de voir que la fonction marche |
| valide 4 | Image précédente |  | Message dans la console "Image précédente" | Message dans la console "Image précédente" | Seulement le message en console permet de voir que la fonction marche |
| valide 5 | Arrêter le diaporama |  | bLancerDiapo->isEnabled() == true bArretDiapo->isEnabled() == false statusbar "Mode: Manuel" | bLancerDiapo->isEnabled() == true bArretDiapo->isEnabled() == false statusbar "Mode: Manuel" | Message dans la console "Arrêt du diaporama". Bouton lancer activé, bouton arrêt désactivé, label d'état "Mode : Manuel". |

# Version v2 –

## Diagramme de classes (UML)



1. Dictionnaire des éléments pour la nouvelle classe

|  |  |  |  |
| --- | --- | --- | --- |
| **Classe LecteurVue** | | | |
| **Nom attribute** | **Signification** | **Type** | **Exemple** |
| ui | Contient tous les éléments de l’interface graphique réalisé avec QT Designer. | UI::LecteurVue |  |
| auteurs | Chaine contenant le nom des auteurs du projet. | Const string | “Mathieu, Virgile et Adrien” |
| date | Chaine contenant la date de création du dépôt. | Const string | “02/05/23” |
| version | Chaine contenant la version actuelle de l’application. | Const string | “V2” |
| \_lecteur | Correspond au lecteur d’images de l’application. | Lecteur |  |
| mode | Correspond au label affiché dans la status bar indiquant le mode de l’application | QLabel | “Mode: non chargé” |
| rang | Correspond au label affiché dans la status bar indiquant le rang de l’image courante. | QLabel | “Rang: 2/3” |

1. Dictionnaire des méthodes de la classe LecteurVue

public:

LecteurVue(QWidget \*parent = nullptr);

~LecteurVue();

public slots:

void chargerDiaporama(); // Charge le diaporama

void suivant(); // Affiche l'image suivante dans le diaporama

void precedent(); // Affiche l'image précedente dans le diaporama

void demarrerDiapo(); // Passe le diaporama en mode automatique

void arreterDiapo(); // Passe le diaporama en mode manuel

void apropos(); //Ouvre boite de dialogue contenant les informations sur le diapo

void afficherImageCourante(); // Affiche l'image courant du lecteur sur l'interface graphique

void majStatusBar(bool); // Met à jour la status bar de l'application.

## Implémentation et tests

### Implémentation (v2)

Liste et rôle des fichiers de cette version :

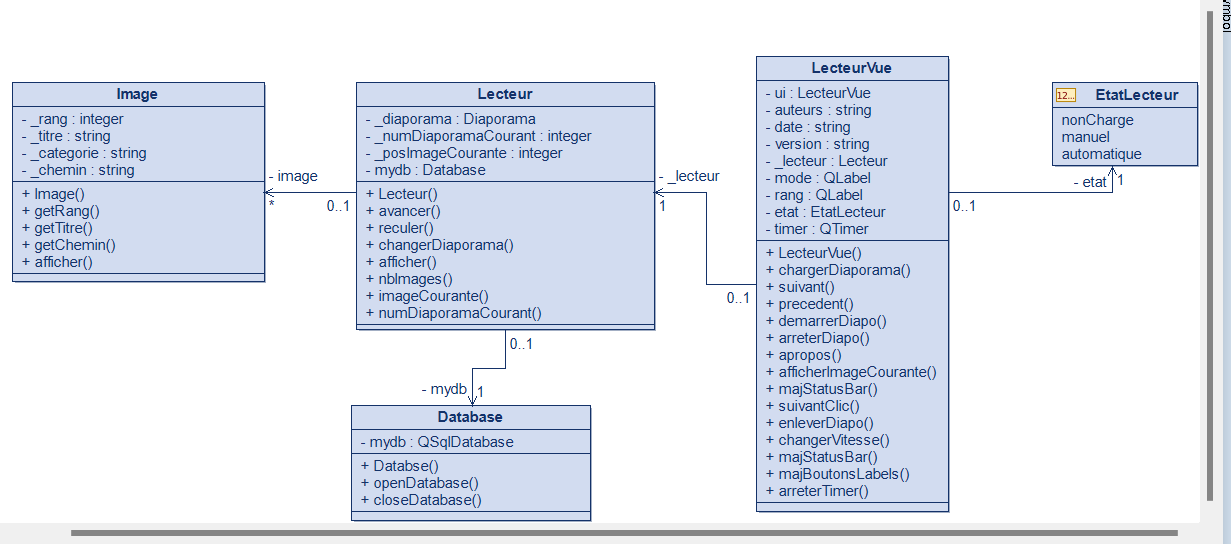
|  |  |
| --- | --- |
| lecteurVue.h | Spécification de la classe graphique Qt contenant l’interface du lecteur de diaporamasPréciser le rôle |
| lecteurVue.cpp | Corps de la classe LecteurVue. |
| lecteurvue.ui | Fichier du dessin de l’interface réalisé par QtDesigner |
| lecteur.h | Spécification de la classe Lecteur.Préciser le rôle |
| lecteur.cpp | Corps de la classe Lecteur |
| image.h | Spécification de la classe ImagePréciser le rôle |
| image.cpp | Corps de la classe Image |
| main.cpp | ?? |

### Tests (v2)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Classe** | **Description** | **Valeur(s)en entrée** | **Résultat(a)attendu** | **résultat obtenu** | **Messages dans la console attendu et obtenu** |
| valide 1 | chargement du diaporama | \_numDiaporamaCourant=1 | première image affichée, label de rang de l'image affiche le rang de l'image, label du mode du diaporama affiché. | première image affichée, label de rang de l'image affiche le rang de l'image, label du mode du diaporama affiché. | Diaporama num. 1 selectionne.  3 images chargees dans le diaporama |
| valide 2 | Lancement du diaporama | \_numDiaporamaCourant=1 bArretDiapo->isEnabled()==false bLancerDiapo->isEnabled()==true | bArretDiapo->isEnabled() == true bLancerDiapo->isEnabled() == false statusbar "Mode: Manuel" | bArretDiapo->isEnabled() == true bLancerDiapo->isEnabled() == false statusbar "Mode: Manuel" | "Démarrage du diaporama" |
| valide 3 | avancer() au rang 1 | \_posImageCourante==1 | \_posImageCourante=2, affichage de l'image rang 2 | \_posImageCourante=2, affichage de l'image rang 2 |  |
| valide 4 | avancer() au rang 2 | \_posImageCourante==2 | \_posImageCourante= 3, affichage de l'image rang 3 | \_posImageCourante =3, affichage de l'image rang 3 |  |
| valide 5 | avancer() au rang 3 | \_posImageCourante==3 | \_posImageCourante =1, affichage de l'image rang 1 | \_posImageCourante =1, affichage de l'image rang 1 |  |
| valide 6 | reculer() au rang 2 | \_posImageCourante==2 | \_posImageCourante= 1, affichage de l'image rang 1 | \_posImageCourante= 1, affichage de l'image rang 1 |  |
| valide 7 | reculer() au rang 1 | \_posImageCourante==1 | \_posImageCourante= 1, affichage de l'image rang 1 | \_posImageCourante= 1, affichage de l'image rang 1 |  |
| valide 8 | reculer() au rang 3 | \_posImageCourante==3 | \_posImageCourante= 2, affichage de l'image rang 2 | \_posImageCourante= 2, affichage de l'image rang 2 |  |
| valide 9 | a propos de |  | affiche une fenêtre avec les informations du diaporama | affiche une fenêtre avec les informations du diaporama |  |
| valide 10 | quitter |  | quitte le diaporama | quitte le diaporama |  |
| valide 11 | Arrêter le diaporama |  | bLancerDiapo->isEnabled() == true bArretDiapo->isEnabled() == false statusbar "Mode: Manuel" | bLancerDiapo->isEnabled() == true bArretDiapo->isEnabled() == false statusbar "Mode: Manuel" | "Arrêt du diaporama" |
| valide 12 | n'importe quelle action si diaporama n'est pas chargé | \_numDiaporamaCourant=0 | crash du diaporama | crash du diaporama |  |

# Version v5 –

## Diagramme de classes (UML)



1. Dictionnaire des éléments pour les nouveaux éléments

|  |  |  |  |
| --- | --- | --- | --- |
| **Classe LecteurVue** | | | |
| **Nom attribute** | **Signification** | **Type** | **Exemple** |
| etat | Correspond à l’état de l’application. | EtatLecteur | nonCharge, manuel, automatique |
| timer | Correspond au timer de l’application lorsque ce dernier est en mode automatique. | QTimer |  |

1. Dictionnaire des méthodes

Classe LecteurVue :

public:

LecteurVue(QWidget \*parent = nullptr);

~LecteurVue();

public slots:

public slots:

void chargerDiaporama(); // Charge le diaporama

void enleverDiaporama(); // Enlève le diaporama

void changerVitesse(); // Ouvre une boite de dialogue pour changer la vitesse du diaporama

void suivant(); // Affiche l'image suivante dans le diaporama

void suivantClic();// Affiche l'image suivante dans le diaporama, considérant un clic de l'utilisateur sur "suivant"

void precedentClic(); // Affiche l'image précedente dans le diaporama, considérant un clic de l'utilisateur sur "précédent"

void demarrerDiapo(); // Passe le diaporama en mode automatique

void arreterDiapo(); // Passe le diaporama en mode manuel

void apropos(); //Ouvre boite de dialogue contenant les informations sur le diapo

void afficherImageCourante(); // Affiche l'image courant du lecteur sur l'interface graphique

void majStatusBar(); // Met à jour la status bar de l'application

void majBoutonsLabels(bool); // Met à jour les boutons et les labels de l'application

void arreterTimer();

Classe Database :

public:

Database();

bool openDatabase(); // Permet de se connecter à la base de données, en renvoyant true si la connexion est réussie, false sinon

void closeDatabase(); // Permet de fermer la connexion à la base de données

## Implémentation et tests

### Implémentation (v5)

Liste et rôle des fichiers de cette version :

|  |  |
| --- | --- |
| lecteurVue.h | Spécification de la classe graphique Qt contenant l’interface du lecteur de diaporamasPréciser le rôle |
| lecteurVue.cpp | Corps de la classe LecteurVue |
| lecteurvue.ui | Fichier du dessin de l’interface réalisé par QtDesigner |
| lecteur.h | Spécification de la classe LecteurPréciser le rôle |
| lecteur.cpp | Corps de la classe Lecteur |
| image.h | Spécification de la classe ImagePréciser le rôle |
| image.cpp | Corps de la classe Image |
| main.cpp | ?? |

## Bilan

Lien du dépôt GitHub : <https://github.com/ToxykAuBleu/LecteurDiaporama>

Temps global de travail : séances de SaE + 10 heures en dehors de ces séances.

Difficulté majeure : collaborer avec Git sur le même fichier (éviter les merge qui casse tout).

Points négatifs :

* le sujet ne nous a pas trouvé si attirant que ça.
* il y a eu de grosses différences entre ce qui a été dit durant la 1ère séance encadré et ce qui est marqué finalement sur le sujet, et il y a eu des oublis (ex : le fichier .gitignore qui nous n’a été pas délivré).

Avec toutes les séances de SaE, nous pensons qu’on n’en a pas eu assez. En effet, nous n’avons qu’à peine terminé la version 6 que nous devons déjà rendre les livrables pour la SaE.