BUT Informatique Semestre 2 2023-2024

IUT de Bayonne et du Pays Basque, Département Informatique

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

CLEMENCEAU Edouard, TD 2 - TP 4

MASSON Rafael, TD 2 - TP 4

VINET LATRILLE Jules, TD 2 - TP 4

1ere Année de formation

Lien GitHub : <https://github.com/jvlatrille/LecteurDiaporama.git>

**Sommaire.**

Table des matières

[Version 1 : Projet non graphique Orienté Objets 5](#__RefHeading___Toc3076_3189070178)

[Diagramme des classes UML 5](#__RefHeading___Toc3078_3189070178)

[Attributs et méthodes de chaque classe 6](#__RefHeading___Toc3080_3189070178)

[Image 6](#__RefHeading___Toc3082_3189070178)

[ImageDansDiapo 7](#__RefHeading___Toc3084_3189070178)

[Diaporama 8](#__RefHeading___Toc3086_3189070178)

[Lecteur 9](#__RefHeading___Toc3088_3189070178)

[Version 2 : Projet non graphique Orienté Objets 10](#__RefHeading___Toc3090_3189070178)

[Diagramme état-transition de l’application sous ses 2 formes 10](#__RefHeading___Toc3092_3189070178)

[Diagramme état-transition 10](#__RefHeading___Toc3094_3189070178)

[Version matricielle 10](#__RefHeading___Toc3096_3189070178)

[Documentation des liens entre éléments d’interface et fonctionnalités 10](#__RefHeading___Toc3098_3189070178)

[Boutons (QPushBouton) : 11](#__RefHeading___Toc3100_3189070178)

[Actions de la barre de menu (QMenuBar, QMenu et QAction) : 11](#__RefHeading___Toc3102_3189070178)

[Version 2 MVP : 12](#__RefHeading___Toc3104_3189070178)

[Diagramme des classes UML 12](#__RefHeading___Toc3106_3189070178)

[Attributs et méthodes de chaque classe 13](#__RefHeading___Toc3108_3189070178)

[Diaporama 13](#__RefHeading___Toc3110_3189070178)

[ImageDansDiaporama 14](#__RefHeading___Toc3112_3189070178)

[Lecteur 14](#__RefHeading___Toc3114_3189070178)

[lecteurVue 15](#__RefHeading___Toc3116_3189070178)

[Modele 17](#__RefHeading___Toc3118_3189070178)

[Presentation 18](#__RefHeading___Toc3120_3189070178)

[Version 3 MVP : 20](#__RefHeading___Toc3122_3189070178)

[Diagramme des classes UML 20](#__RefHeading___Toc3124_3189070178)

[Attributs et méthodes de chaque classe 21](#__RefHeading___Toc3126_3189070178)

[apropos 21](#__RefHeading___Toc3128_3189070178)

[Diaporama 21](#__RefHeading___Toc3130_3189070178)

[ImageDansDiaporama 22](#__RefHeading___Toc3132_3189070178)

[Lecteur 23](#__RefHeading___Toc3134_3189070178)

[lecteurVue 24](#__RefHeading___Toc3136_3189070178)

[Modele 27](#__RefHeading___Toc3138_3189070178)

[Presentation 29](#__RefHeading___Toc3140_3189070178)

[Version 4 MVP : 31](#__RefHeading___Toc3122_3189070178_Copie_)

[Diagramme des classes UML 31](#__RefHeading___Toc3124_3189070178_Copie_)

[Attributs et méthodes de chaque classe 32](#__RefHeading___Toc3126_3189070178_Copie_)

[apropos 32](#__RefHeading___Toc3128_3189070178_Copie_)

[Diaporama 32](#__RefHeading___Toc3130_3189070178_Copie_)

[ImageDansDiaporama 33](#__RefHeading___Toc3132_3189070178_Copie_)

[Lecteur 34](#__RefHeading___Toc3134_3189070178_Copie_)

[lecteurVue 36](#__RefHeading___Toc3136_3189070178_Copie_)

[Modele 38](#__RefHeading___Toc3138_3189070178_Copie_)

[Presentation 40](#__RefHeading___Toc3140_3189070178_Copie_)

[Version 5 MVP : 42](#__RefHeading___Toc3122_3189070178_Copie1)

[Diagramme des classes UML 42](#__RefHeading___Toc3124_3189070178_Copie1)

[Attributs et méthodes de chaque classe 43](#__RefHeading___Toc3126_3189070178_Copie1)

[apropos 43](#__RefHeading___Toc3128_3189070178_Copie1)

[charger 43](#__RefHeading___Toc3128_3189070178_Copie2)

[Diaporama 44](#__RefHeading___Toc3130_3189070178_Copie1)

[ImageDansDiaporama 45](#__RefHeading___Toc3132_3189070178_Copie1)

[Lecteur 46](#__RefHeading___Toc3134_3189070178_Copie1)

[lecteurVue 47](#__RefHeading___Toc3136_3189070178_Copie1)

[Modele 50](#__RefHeading___Toc3138_3189070178_Copie1)

[Presentation 52](#__RefHeading___Toc3140_3189070178_Copie1)

[vit 54](#__RefHeading___Toc3128_3189070178_Copie3)

[Version 7 MVP : 55](#__RefHeading___Toc3122_3189070178_Copie2)

[Diagramme des classes UML 55](#__RefHeading___Toc3124_3189070178_Copie2)

[Attributs et méthodes de chaque classe 56](#__RefHeading___Toc3126_3189070178_Copie2)

[apropos 56](#__RefHeading___Toc3128_3189070178_Copie4)

[charger 56](#__RefHeading___Toc3128_3189070178_Copie5)

[database 57](#__RefHeading___Toc3128_3189070178_Copie7)

[Diaporama 57](#__RefHeading___Toc3130_3189070178_Copie2)

[ImageDansDiaporama 59](#__RefHeading___Toc3132_3189070178_Copie2)

[Lecteur 60](#__RefHeading___Toc3134_3189070178_Copie2)

[lecteurVue 61](#__RefHeading___Toc3136_3189070178_Copie2)

[Modele 63](#__RefHeading___Toc3138_3189070178_Copie2)

[Presentation 65](#__RefHeading___Toc3140_3189070178_Copie2)

[vit 67](#__RefHeading___Toc3128_3189070178_Copie6)

[Bilan 68](#__RefHeading___Toc3142_3189070178)

[Ce que l’on a appris 68](#__RefHeading___Toc3144_3189070178)

[Ce qu’on a aimé / pas aimé 68](#__RefHeading___Toc3146_3189070178)

[Ce qui a été difficile 68](#__RefHeading___Toc3148_3189070178)

[Le temps passé (sur conception / code) 68](#__RefHeading___Toc3150_3189070178)

[Ce que vous auriez pu faire de mieux (avec le recul) 69](#__RefHeading___Toc3152_3189070178)

[Ce qui pourrait être amélioré dans la SAE 69](#__RefHeading___Toc3154_3189070178)

# **Version 1 : Projet non graphique Orienté Objets**

## Diagramme des classes UML



## Attributs et méthodes de chaque classe

### Image

|  |  |  |
| --- | --- | --- |
| Attributs | | |
| Nom | Type | Signification |
| m\_titre | Chaîne de caractères | Titre de l’image |
| m\_categorie | Chaîne de caractères | Catégorie de l’image |
| m\_chemin | Chaîne de caractères | Chemin de l’image |

|  |  |
| --- | --- |
| Méthodes | |
| Nom | Signification et but |
| Image() | Constructeur par défaut |
| getTitre() | Getter qui permet de récupérer le titre de l'image |
| getChemin() | Getter qui permet de récupérer le chemin de l'image |
| getCategorie() | Getter qui permet de récupérer la catégorie de l'image |
| setTitre() | Setter qui permet de définir le titre de l'image |
| setChemin() | Setter qui permet de définir le chemin de l'image |
| setCategorie() | Setter qui permet de définir la catégorie de l'image |
| afficher() | Afficher tous les détails de l'image |

### ImageDansDiapo

|  |  |  |
| --- | --- | --- |
| Attributs | | |
| Nom | Type | Signification |
| m\_rang | Entier non signé | Rang de l'image dans le diaporama |
| m\_pos | Entier non signé | Position de l'image dans le diaporama |

|  |  |  |
| --- | --- | --- |
| Méthodes | | |
| Nom | Signification et but | |
| ImageDansDiaporama() | Constructeur par défaut | Créer une instance d'image dans un diaporama avec des valeurs par défaut |
| getRang() | Getter qui permet de récupérer le rang de l'image | |
| getPos() | Getter qui permet de récupérer la position de l'image | |
| setRang() | Setter qui permet de définir le rang de l'image | |
| setPos() | Setter qui permet de définir la position de l'image | |
| afficherImageCourante() | Afficher les détails (position et rang) de l'image dans le diaporama | |

### Diaporama

|  |  |  |
| --- | --- | --- |
| Attributs | | |
| Nom | Type | Signification |
| m\_titre | Chaîne de caractères | Titre du diaporama |
| m\_vitesseDefilement | Entier non signé | Vitesse de défilement du diaporama |
| m\_posImageCourante | Entier non signé | Position de l'image courante dans le diaporama |

|  |  |  |
| --- | --- | --- |
| Méthodes | | |
| Nom | Signification et but | |
| Diaporama() | Constructeur par défaut | Créer une instance de diaporama avec des valeurs par défau |
| getTitre() | Getter qui permet de récupérer le titre du diaporama | |
| getVitesseDefilement() | Getter qui permet de récupérer la vitesse de défilement du diaporama | |
| getNombreImages() | Getter qui permet de récupérer le nombre d'images dans le diaporama | |
| getPosImageCourante() | Getter qui permet de récupérer la position de l'image courante dans le diaporama | |
| setTitre() | Setter qui permet de définir le titre du diaporama | |
| setVitesseDefilement() | Setter qui permet de définir la vitesse de défilement du diaporama | |
| setNombreImages() | Setter qui permet de définir le nombre d'images dans le diaporama | |
| setPosImageCourante() | Setter qui permet de définir la position de l'image courante dans le diaporama | |
| ajouterImage() | Ajouter une image dans le diaporama | |
| avancer() | Avancer vers l'image suivante dans le diaporama | |
| reculer() | Reculer vers l'image précédente dans le diaporama | |
| afficherImageCouranteDansDiaporamaCourant() | Afficher les détails de l'image courante dans le diaporama | |
| triCroissantRang() | Trier les images du diaporama par leur rang | |

### Lecteur

|  |  |  |
| --- | --- | --- |
| Attributs | | |
| Nom | Type | Signification |
| m\_numDiapoCourant | Entier non signé | Numéro du diaporama courant |

|  |  |  |
| --- | --- | --- |
| Méthodes | | |
| Nom | Signification et but | |
| Lecteur() | Constructeur par défaut | Créer une instance de lecteur de diaporamas avec des valeurs par défaut |
| getNumDiapoCourant() | Getter qui permet de récupérer le numéro du diaporama courant | |
| getNombreDiapos() | Getter qui permet de récupérer le nombre total de diaporamas | |
| setAllDiapos() | Setter qui permet de définir toutes les diapositives | |
| setNumDiapoCourant() | Setter qui permet de définir le numéro du diaporama courant | |
| setNombreDiapos() | Setter qui permet de définir le nombre total de diaporamas | |
| declencherAction() | Déclencher une action dans le lecteur | |
| saisieChoixActionImageCourante() | Saisir le choix d'action pour l'image courante | |
| saisieChoixDiaporama() | Saisir le choix de diaporama | |
| chargerImage() | Charger les images | |
| chargerDiapos() | Charger les diaporamas | |
| afficherDiapoCourant() | Afficher le diaporama courant | |

# **Version 2 : Projet non graphique Orienté Objets**

## Diagramme état-transition de l’application sous ses 2 formes

### Diagramme état-transition

Une image contenant texte, capture d’écran, diagramme, Parallèle

Description générée automatiquement

### Version matricielle

<https://docs.google.com/spreadsheets/d/1wFnkd9DNGp6v6HGY9ECIkjlzgzVSTF-KcPUflqhTCMQ/edit?usp=sharing>

## Documentation des liens entre éléments d’interface et fonctionnalités

### Boutons (QPushBouton) :

Précédent : Permet de retourner à l’image précédente lorsque l’on est en mode de défilement manuel.

Pause : Permet de mettre le défilement en pause lorsqu’on est en mode de défilement automatique.

Suivant : Permet de passer à l’image suivante lorsque l’on est en mode de défilement manuel.

### Actions de la barre de menu (QMenuBar, QMenu et QAction) :

Fichier > Quitter : Permet de fermer l’application.

Fichier > Charger diaporama : Permet de charger un nouveau diaporama (qu’il y en ait un déjà chargé ou pas).

Paramètres > Enlever diaporama : Permet d’enlever le diaporama qui est chargé.

Paramètres > Vitesse de défilement : Permet d’ouvrir une boite de dialogue qui donne la possibilité de changer la vitesse de défilement des images en défilement automatique.

Mode > Automatique : Permet de changer de mode de défilement pour passer en mode automatique . Par défaut le mode est en manuel.

Mode > Manuel : Permet de changer de mode de défilement pour passer en mode manuel. Par défaut le mode est en manuel.

Aide > A propos : Permet d’afficher la version actuelle du lecteur de diaporama et ses auteurs.

# **Version 2 MVP :**

## Diagramme des classes UML

Une image contenant texte, capture d’écran, diagramme, nombre

Description générée automatiquement

## Attributs et méthodes de chaque classe

### Diaporama

|  |  |  |
| --- | --- | --- |
| Attributs | | |
| Nom | Type | Signification |
| id | Entier non signé | Identifiant du diaporama |
| titre | Chaîne de caractères | Titre du diaporama |
| vitesseDefilement | Entier non signé | Vitesse de défilement des images du diaporama |
| image | ImagesDiaporama | Vecteur de pointeurs sur les objets ImageDansDiaporama du diaporama |

|  |  |
| --- | --- |
| Méthodes | |
| Nom | Signification et but |
| Diaporama() | Constructeur du diaporama |
| ~Diaporama() | Destructeur de diaporama |
| getId() | Getter pour l'identifiant du diaporama |
| getTitre() | Getter pour le titre du diaporama |
| getVitesseDefilement() | Getter pour la vitesse de défilement des images |
| getImages() | Getter pour les images du diaporama |
| nbImages() | Retourne le nombre d'images contenus dans le diaporama |
| setId() | Setter pour l'identifiant du diaporama |
| setTitre() | Setter pour le titre du diaporama |
| setVitesseDefilement() | Setter pour la vitesse de défilement des images |
| setImages() | Setter pour les images du diaporama |
| ajouterImageEnFin() | Ajoute une image à la fin du diaporama |
| enleverImageEnFin() | Enlève la dernière image du diaporama |
| vider() | Enlève toutes les images du diaporama |
| charger() | Charge les images associées au diaporama courant |
| trierParRangCroissant() | Trie les images du diaporama par ordre de rang croissant |

### ImageDansDiaporama

|  |  |  |
| --- | --- | --- |
| Attributs | | |
| Nom | Type | Signification |
| rangDansDiaporama | Entier non signé | Rang de l'image dans le diaporama |
| titre | Chaîne de caractères | Intitulé de l'image |
| categorie | Chaîne de caractères | Catégorie de l'image (personne, animal, objet) |
| chemin | Chaîne de caractères | Chemin complet vers le dossier où se trouve l'image |

|  |  |
| --- | --- |
| Méthodes | |
| Nom | Signification et but |
| ImageDansDiaporama() | Constructeur de l’image dans le diaporama |
| ~ImageDansDiaporama() | Destructeur de l’image dans le diaporama |
| getRangDansDiaporama() | Getter pour le rang de l'image dans le diaporama |
| getCategorie() | Getter pour la catégorie de l'image |
| getTitre() | Getter pour le titre de l'image |
| getChemin() | Getter pour le chemin complet vers le dossier de l'image |
| afficher() | Affiche tous les champs de l'image |
| setRangDansDiaporama() | Setter pour le rang de l'image dans le diaporama |
| setTitre(sting) | Setter pour le titre de l'image |
| setCategorie() | Setter pour la catégorie de l'image |
| setChemin() | Setter pour le chemin complet vers le dossier de l'image |

### Lecteur

|  |  |  |
| --- | --- | --- |
| Attributs | | |
| Nom | Type | Signification |
| idDiaporama | Entier non signé | Identifiant du diaporama courant |
| posImageCourante | Entier non signé | Position de l'image courante |
| diaporama | diaporama | Pointeur vers le diaporama associé au lecteur |

|  |  |
| --- | --- |
| Méthodes | |
| Nom | Signification et but |
| Lecteur() | Constructeur du lecteur |
| ~Lecteur() | Destructeur du lecteur |
| getIdDiaporama() | Getter qui permet de récupérer l'identifiant du diaporama courant |
| getDiaporama() | Getter qui permet de récupérer le pointeur vers le diaporama associé |
| getPosImageCourante() | Getter qui permet de récupérer la position de l'image courante |
| lecteurVide() | Vérifie si aucun diaporama n'est associé au lecteur |
| getImageCourante() | Retourne le pointeur vers l'image courante |
| nbImages() | Retourne la taille de la collection pointée par diaporama |
| setIdDiaporama() | Setter qui permet de définir l'identifiant du diaporama courant |
| setDiaporama() | Setter qui permet de définir le pointeur vers le diaporama associé |
| setPosImageCourante() | Setter qui permet de définir la position de l'image courante |
| changerDiaporama() | Choisir un diaporama avec possibilité de spécifier l'identifiant, le titre et la vitesse |
| avancer() | Incrémente la position de l'image courante |
| reculer() | Décrémente la position de l'image courante |
| viderLecteur() | Enlève le diaporama courant du lecteur s'il existe |
| chargerDiaporamaCourant() | Charge dans le lecteur l'image dans le diaporama courant |

### lecteurVue

|  |  |  |
| --- | --- | --- |
| Attributs | | |
| Nom | Type | Signification |
| ui | ui | Pointeur vers l'interface utilisateur de la fenêtre lecteurVue |
| m\_MaPresentation | Presentation | Pointeur vers l'objet de présentation associé à la vue lecteurVue |

|  |  |
| --- | --- |
| Méthodes | |
| Nom | Signification et but |
| ~lecteurVue() | Constructeur du lecteur vue |
| ~lecteurVue() | Destructeur du lecteur vue |
| sl\_suivant() | Slot pour gérer l'action suivante |
| sl\_precedent() | Slot pour gérer l'action précédente |
| sl\_pause() | Slot pour gérer la pause |
| sl\_chargerDiapo() | Slot pour gérer le chargement du diaporama |
| sl\_quitter() | Slot pour gérer la fermeture de la fenêtre lecteurVue |
| sl\_enleverDiapo() | Slot pour gérer la suppression du diaporama |
| sl\_vitesseDefilement() | Slot pour gérer le changement de vitesse de défilement |
| sl\_modeAuto() | Slot pour gérer le passage en mode automatique |
| sl\_modeManuel() | Slot pour gérer le passage en mode manuel |
| sl\_aPropos() | Slot pour afficher les informations "À propos" |
| getPresentation() | Getter pour récupérer le pointeur vers la présentation associé |
| setPresentation() | Setter pour définir la présentation associé |

### Modele

|  |  |  |
| --- | --- | --- |
| Attributs | | |
| Nom | Type | Signification |
| m\_titre | Chaîne de caractères | Titre du modèle |
| m\_vitesseDefilement | Entier non signé | Vitesse de défilement des images |
| lecteurVide | bool | Si le lecteur est vide ou non |
| m\_posImageCourante | Entier non signé | Position de l'image courante |
| m\_localisationImages | vector | Vecteur d'images dans le diaporama |
| m\_MonDiapo | Diaporama | Pointeur vers le diaporama associé |
| images | ImagesDiaporama | Vecteur de pointeurs vers les objets ImageDansDiaporama |

|  |  |
| --- | --- |
| Méthodes | |
| Nom | Signification et but |
| Modele() | Constructeur du modele |
| getVitesseDefilement() | Getter pour la vitesse de défilement des images |
| getPosImageCourante() | Getter pour la position de l'image courante |
| nbImages() | Retourne le nombre d'images dans le diaporama |
| getDiaporama() | Getter pour récupérer le pointeur vers le diaporama associé |
| setVitesseDefilement() | Setter pour définir la vitesse de défilement des images |
| setDiaporama() | Setter pour définir le pointeur vers le diaporama associé |
| setPosImageCourante() | Setter pour définir la position de l'image courante |
| avancer() | Passe à l’image suivante |
| reculer() | Reviens à l’image précédente |
| triCroissantRang() | Trie les images par ordre croissant de leur rang dans le diaporama |

### Presentation

\*pas d’attributs\*

|  |  |
| --- | --- |
| Méthodes | |
| Nom | Signification et but |
| Presentation() | Constructeur de la présentation |
| getModele() | Getter pour récupérer le pointeur vers l'objet modèle associé |
| getVue() | Getter pour récupérer le pointeur vers l'objet vue associé |
| setModel() | Setter pour définir l'objet modèle associé |
| setVue() | Setter pour définir l'objet vue associé |
| demanderAvancer() | Slot qui demande d'avancer dans le diaporama (émission du signal s\_avancer) |
| demanderReculer() | Slot qui demande de reculer dans le diaporama (émission du signal s\_reculer) |
| demanderDepartArretAuto() | Slot qui demande de démarrer ou arrêter le défilement automatique (émission du signal s\_departArret) |
| demanderChangerVitesse() | Slot qui demande de changer la vitesse de défilement (émission du signal s\_changerVitesse) |
| demanderChangerModeAutomatique() | Slot qui demande de changer le mode de défilement (émission du signal s\_changerModeAuto) |
| demanderChangerModeManuel() | Slot qui demande de changer le mode de défilement (émission du signal s\_changerModeManuel) |
| demanderChargerDiapo() | Slot qui demande de charger le diaporama (émission du signal s\_chargerDiapo) |
| demanderQuitter() | Slot qui demande de quitter l’application (émission du signal s\_quitter) |
| demanderEnleverDiapo() | Slot qui demande d’enlever le diaporama (émission du signal s\_enleverDiapo) |
| demanderAPropos() | Slot qui demande d’afficher le informations sur l’application (émission du signal s\_aPropos) |
| s\_avancer() | Signale la demande d'avancer dans le diaporama |
| s\_reculer() | Signale la demande de reculer dans le diaporama |
| s\_departArret() | Signale la demande de démarrer ou arrêter le défilement automatique |
| s\_changerVitesse() | Signale la demande de changer la vitesse de défilement |
| s\_changerModeAuto() | Signale la demande de passer en mode de défilement automatique |
| s\_changerModeManuel() | Signale la demande de passer en mode de défilement manuel |
| s\_chargerDiapo() | Signale la demande de charger un diaporama |
| s\_quitter() | Signale la demande de quitter l'application |
| s\_enleverDiapo() | Signale la demande de supprimer un diaporama |
| s\_aPropos() | Signale la demande d'afficher les informations "À propos" |

# **Version 3 MVP :**

## Diagramme des classes UML

Une image contenant texte, capture d’écran, diagramme, nombre

Description générée automatiquement

## Attributs et méthodes de chaque classe

### apropos

|  |  |  |
| --- | --- | --- |
| Attributs | | |
| Nom | Type | Signification |
| ui | Ui | Pointeur vers l'interface utilisateur de la fenêtre apropos |

|  |  |
| --- | --- |
| Méthodes | |
| Nom | Signification et but |
| ~apropos() | Destructeur de apropos |

### Diaporama

|  |  |  |
| --- | --- | --- |
| Attributs | | |
| Nom | Type | Signification |
| id | Entier non signé | Identifiant du diaporama |
| titre | Chaîne de caractères | Titre du diaporama |
| vitesseDefilement | Entier non signé | Vitesse de défilement des images du diaporama |
| images | ImagesDiaporama | Vecteur de pointeurs sur les objets ImageDansDiaporama du diaporama |
| PosImageCourante | Entier non signé | Position de l’image courante dans le diaporama |

|  |  |
| --- | --- |
| Méthodes | |
| Nom | Signification et but |
| Diaporama() | Constructeur du diaporama |
| ~Diaporama() | Destructeur de diaporama |
| unsigned int getId() const | Getter pour l'identifiant du diaporama |
| string getTitre() const | Getter pour le titre du diaporama |
| int getVitesseDefilement() const | Getter pour la vitesse de défilement des images |
| ImagesDiaporama getImages() const | Getter pour les images du diaporama |
| ImageDansDiaporama\* getImageCourante() const | Getter de l’image courante le diaporama |
| unsigned int getPosImageCourante() const | Getter de la position de l’image courante dans le diaporama |
| unsigned int nbImages() const | Retourne le nombre d'images contenus dans le diaporama |
| void setId(unsigned int pId) | Setter pour l'identifiant du diaporama |
| void setTitre(string pTitre) | Setter pour le titre du diaporama |
| void setVitesseDefilement(unsigned int pVitesseDefilement) | Setter pour la vitesse de défilement des images |
| void setImages(const ImagesDiaporama& pImages) | Setter pour les images du diaporama |
| void ajouterImageEnFin(ImageDansDiaporama\* pImage) | Ajoute une image à la fin du diaporama |
| void enleverImageEnFin() | Enlève la dernière image du diaporama |
| void vider() | Enlève toutes les images du diaporama |
| void charger() | Charge les images associées au diaporama courant |
| void trierParRangCroissant() | Trie les images du diaporama par ordre de rang croissant |

### ImageDansDiaporama

|  |  |  |
| --- | --- | --- |
| Attributs | | |
| Nom | Type | Signification |
| rangDansDiaporama | Entier non signé | Rang de l'image dans le diaporama |
| titre | Chaîne de caractères | Intitulé de l'image |
| categorie | Chaîne de caractères | Catégorie de l'image (personne, animal, objet) |
| chemin | Chaîne de caractères | Chemin complet vers le dossier où se trouve l'image |

|  |  |
| --- | --- |
| Méthodes | |
| Nom | Signification et but |
| ImageDansDiaporama(unsigned int pRangDansDiaporama=0,  string pCategorie="defaut", string pTitre="defaut", string pChemin = ":/images/Disney\_tapis.gif") | Constructeur de l’image dans le diaporama |
| ~ImageDansDiaporama() | Destructeur de l’image dans le diaporama |
| unsigned int getRangDansDiaporama() const | Getter pour le rang de l'image dans le diaporama |
| string getCategorie() const | Getter pour la catégorie de l'image |
| string getTitre() const | Getter pour le titre de l'image |
| string getChemin() const | Getter pour le chemin complet vers le dossier de l'image |
| void afficher() | Affiche tous les champs de l'image |
| void setRangDansDiaporama (unsigned int pRangDansDiaporama) | Setter pour le rang de l'image dans le diaporama |
| void setTitre (string pTitre) | Setter pour le titre de l'image |
| void setCategorie (string pCategorie) | Setter pour la catégorie de l'image |
| void setChemin (string pChemin) | Setter pour le chemin complet vers le dossier de l'image |

### Lecteur

|  |  |  |
| --- | --- | --- |
| Attributs | | |
| Nom | Type | Signification |
| idDiaporama | Entier non signé | Identifiant du diaporama courant en BDD |
| posImageCourante | Entier non signé | Position de l'image courante dans le diaporama |
| diaporama | Diaporama | Pointeur vers le diaporama associé au lecteur |

|  |  |
| --- | --- |
| Méthodes | |
| Nom | Signification et but |
| Lecteur() | Constructeur du lecteur |
| ~Lecteur() | Destructeur du lecteur |
| unsigned int getIdDiaporama() const | Getter qui permet de récupérer l'identifiant du diaporama courant |
| Diaporama\* getDiaporama() const | Getter qui permet de récupérer le pointeur vers le diaporama associé |
| unsigned int getPosImageCourante() const | Getter qui permet de récupérer la position de l'image courante |
| bool lecteurVide() const | Vérifie si aucun diaporama n'est associé au lecteur |
| ImageDansDiaporama\* getImageCourante() const | Retourne le pointeur vers l'image courante |
| unsigned int nbImages() const | Retourne la taille de la collection pointée par diaporama |
| void afficher() | Affiche les informations sur lecteur + éventuellement diaporama et image courante |
| void setIdDiaporama(unsigned int pIdDiaporama) | Setter qui permet de définir l'identifiant du diaporama courant |
| void setDiaporama (Diaporama\* pDiaporama) | Setter qui permet de définir le pointeur vers le diaporama associé |
| void setPosImageCourante(unsigned int pPosImageCourante) | Setter qui permet de définir la position de l'image courante |
| void changerDiaporama(unsigned int pId, string pTitre="", unsigned int pVitesse=0) | Choisir un diaporama avec possibilité de spécifier l'identifiant, le titre et la vitesse |
| void avancer() | Incrémente la position de l'image courante |
| void reculer() | Décrémente la position de l'image courante |
| void viderLecteur() | Enlève le diaporama courant du lecteur s'il existe |
| void chargerDiaporamaCourant() | Charge dans le lecteur l'image dans le diaporama courant |

### lecteurVue

|  |  |  |
| --- | --- | --- |
| Attributs | | |
| Nom | Type | Signification |
| ui | Ui | Pointeur vers l'interface utilisateur de la fenêtre lecteurVue |
| m\_MaPresentation | Presentation | Pointeur vers l'objet de présentation associé à la vue lecteurVue |

|  |  |
| --- | --- |
| Méthodes | |
| Nom | Signification et but |
| lecteurVue() | Constructeur du lecteur vue |
| ~lecteurVue() | Destructeur du lecteur vue |
| Presentation \* getPresentation() const | Getter pour récupérer le pointeur vers la présentation associé |
| void setPresentation(Presentation \*) | Setter pour définir la présentation associé |
| void initializeConnections() | Connexion entre la présentation et le modèle |
| void demanderAvancer() | Slot qui demande d'avancer dans le diaporama (exécute demanderAvancer() de Presentation) |
| void demanderReculer() | Slot qui demande de reculer dans le diaporama (exécute demanderReculer() de Presentation) |
| void demanderDepartArretAuto() | Slot qui demande de démarrer ou arrêter le défilement automatique (exécute demanderDepartArretAuto() de Presentation) |
| void demanderChangerVitesse() | Slot qui demande de changer la vitesse de défilement (exécute demanderChangerVitesse() de Presentation) |
| void demanderChangerModeAutomatique() | Slot qui demande de changer le mode de défilement (exécute demanderChangerModeAutomatique() de Presentation) |
| void demanderChangerModeManuel() | Slot qui demande de changer le mode de défilement (exécute demanderChangerModeManuel() de Presentation) |
| void demanderChargerDiapo() | Slot qui demande de charger le diaporama (exécute demanderChargerDiapo() de Presentation) |
| void quitterApplication() | Slot qui quitte l’application |
| void demanderEnleverDiapo() | Slot qui demande d’enlever le diaporama (exécute demanderEnleverDiapo() de Presentation) |
| void demanderAPropos() | Slot qui demande d’afficher le informations sur l’application (exécute demanderAPropos() de Presentation) |
| void majPresentation(const QString &titreDiapo, const QString &titreImage, const QString &categorie, const QString &rang, const QString &chemin) | Slot qui met à jour les informations de la fenêtre. |

### Modele

|  |  |  |
| --- | --- | --- |
| Attributs | | |
| Nom | Type | Signification |
| m\_titre | Chaîne de caractères | Titre du modèle |
| m\_vitesseDefilement | Entier non signé | Vitesse de défilement des images |
| m\_posImageCourante | Entier non signé | Position de l'image courante |
| m\_localisationImages | Vector <ImageDansDiaporama> | Vecteur d'images dans le diaporama |
| m\_MonDiapo | Diaporama\* | Pointeur vers le diaporama associé |
| images | ImagesDiaporama | Vecteur de pointeurs vers les objets ImageDansDiaporama |
| lecteur | Lecteur\* | Pointeur vers le lecteur |
| UnEtat | enum | Type énuméré contenant l’état du modèle (manuel ou automatique) |
| \_etat | UnEtat | Etat du modèle |

|  |  |
| --- | --- |
| Méthodes | |
| Nom | Signification et but |
| Modele() | Constructeur du modele |
| void triCroissantRang() | Trie les images par ordre croissant de leur rang dans le diaporama |
| bool lecteurVide() const | Si le lecteur est vide ou non |
| unsigned int getVitesseDefilement() const | Getter pour la vitesse de défilement des images |
| unsigned int getPosImageCourante() const | Getter pour la position de l'image courante |
| unsigned int nbImages() const | Retourne le nombre d'images dans le diaporama |
| UnEtat getEtat() const | Getter pour récupérer l’état du modèle |
| Diaporama\* getDiaporama() const | Getter pour récupérer le pointeur vers le diaporama associé |
| void setVitesseDefilement(unsigned int) | Setter pour définir la vitesse de défilement des images |
| void setDiaporama(Diaporama \*) | Setter pour définir le pointeur vers le diaporama associé |
| void setPosImageCourante(unsigned int pPosImageCourante) | Setter pour définir la position de l'image courante |
| void avancer() | Passe à l’image suivante |
| void reculer() | Reviens à l’image précédente |
| void departArretAuto() | Active ou arrête le mode automatique |
| void changerVitesse() | Change la vitesse de défilement |
| void etatAutomatique() | Met l’état du modèle à automatique |
| void etatManuel() | Met l’état du modèle à manuel |
| void chargerDiapo() | Charge un diaporama |
| void enleverDiapo() | Enlève un diaporama |
| void aPropos() | Affiche la fenêtre de dialogue apropos |
| Lecteur \* getLecteur() const | Getter qui retourne le lecteur |
| void setLecteur(Lecteur \* newLecteur) | Setter du lecteur |
| void imageChange(const QString &titreDiapo, const QString &titreImage, const QString &categorie, const QString &rang, const QString &chemin) | Signal qui est connecté à majPresentation, permettant la mise à jour des informations affichées sur la fenêtre principale |

### Presentation

|  |  |  |
| --- | --- | --- |
| Attributs | | |
| Nom | Type | Signification |
| \_leModele | Modele\* | Le modèle |
| \_laVue | lecteurVue\* | La vue |

|  |  |
| --- | --- |
| Méthodes | |
| Nom | Signification et but |
| Presentation() | Constructeur de la présentation |
| void setModel(Modele\* m) | Setter pour définir l'objet modèle associé |
| void setVue(lecteurVue\* v) | Setter pour définir l'objet vue associé |
| Modele \*getModele() | Getter pour récupérer le pointeur vers l'objet modèle associé |
| lecteurVue \*getVue() | Getter pour récupérer le pointeur vers l'objet vue associé |
| ImageDansDiaporama\* getImageActuelle() | Getter pour récupérer l’image actuelle |
| Diaporama \* getDiapoActuel() | Getter pour récupérer le diaporama actuel |
| void demanderAvancer() | Demande d'avancer dans le diaporama (avancer() dans \_leModele) |
| void demanderReculer() | Demande de reculer dans le diaporama (reculer() dans \_leModele) |
| void demanderDepartArretAuto() | Demande de démarrer ou arrêter le défilement automatique (departArretAuto() dans \_leModele) |
| void demanderChangerVitesse() | Demande de changer la vitesse de défilement (changerVitesse() dans \_leModele) |
| void demanderChangerModeAutomatique() | Demande de changer le mode de défilement (etatAutomatique() dans \_leModele) |
| void demanderChangerModeManuel() | Demande de changer le mode de défilement (etatManuel() dans \_leModele) |
| void demanderChargerDiapo() | Demande de charger le diaporama (chargerDiapo() dans \_leModele) |
| void demanderEnleverDiapo() | Demande d’enlever le diaporama (enleverDiapo() dans \_leModele) |
| void demanderAPropos() | Demande d’afficher le informations sur l’application (aPropos() dans \_leModele) |

# **Version 4 MVP :**

## Diagramme des classes UML

Une image contenant texte, capture d’écran, diagramme, nombre

Description générée automatiquement

## Attributs et méthodes de chaque classe

### apropos

|  |  |  |
| --- | --- | --- |
| Attributs | | |
| Nom | Type | Signification |
| ui | Ui | Pointeur vers l'interface utilisateur de la fenêtre apropos |

|  |  |
| --- | --- |
| Méthodes | |
| Nom | Signification et but |
| ~apropos() | Destructeur de apropos |

### Diaporama

|  |  |  |
| --- | --- | --- |
| Attributs | | |
| Nom | Type | Signification |
| id | Entier non signé | Identifiant du diaporama |
| titre | Chaîne de caractères | Titre du diaporama |
| vitesseDefilement | Entier non signé | Vitesse de défilement des images du diaporama |
| images | ImagesDiaporama | Vecteur de pointeurs sur les objets ImageDansDiaporama du diaporama |
| PosImageCourante | Entier non signé | Position de l’image courante dans le diaporama |

|  |  |
| --- | --- |
| Méthodes | |
| Nom | Signification et but |
| Diaporama() | Constructeur du diaporama |
| ~Diaporama() | Destructeur de diaporama |
| unsigned int getId() const | Getter pour l'identifiant du diaporama |
| string getTitre() const | Getter pour le titre du diaporama |
| int getVitesseDefilement() const | Getter pour la vitesse de défilement des images |
| ImagesDiaporama getImages() const | Getter pour les images du diaporama |
| ImageDansDiaporama\* getImageCourante() const | Getter de l’image courante le diaporama |
| unsigned int getPosImageCourante() const | Getter de la position de l’image courante dans le diaporama |
| unsigned int nbImages() const | Retourne le nombre d'images contenus dans le diaporama |
| void setId(unsigned int pId) | Setter pour l'identifiant du diaporama |
| void setTitre(string pTitre) | Setter pour le titre du diaporama |
| void setVitesseDefilement(unsigned int pVitesseDefilement) | Setter pour la vitesse de défilement des images |
| void setImages(const ImagesDiaporama& pImages) | Setter pour les images du diaporama |
| void ajouterImageEnFin(ImageDansDiaporama\* pImage) | Ajoute une image à la fin du diaporama |
| void enleverImageEnFin() | Enlève la dernière image du diaporama |
| void vider() | Enlève toutes les images du diaporama |
| void charger() | Charge les images associées au diaporama courant |
| void trierParRangCroissant() | Trie les images du diaporama par ordre de rang croissant |

### ImageDansDiaporama

|  |  |  |
| --- | --- | --- |
| Attributs | | |
| Nom | Type | Signification |
| rangDansDiaporama | Entier non signé | Rang de l'image dans le diaporama |
| titre | Chaîne de caractères | Intitulé de l'image |
| categorie | Chaîne de caractères | Catégorie de l'image (personne, animal, objet) |
| chemin | Chaîne de caractères | Chemin complet vers le dossier où se trouve l'image |

|  |  |
| --- | --- |
| Méthodes | |
| Nom | Signification et but |
| ImageDansDiaporama(unsigned int pRangDansDiaporama=0,  string pCategorie="defaut", string pTitre="defaut", string pChemin = ":/images/Disney\_tapis.gif") | Constructeur de l’image dans le diaporama |
| ~ImageDansDiaporama() | Destructeur de l’image dans le diaporama |
| unsigned int getRangDansDiaporama() const | Getter pour le rang de l'image dans le diaporama |
| string getCategorie() const | Getter pour la catégorie de l'image |
| string getTitre() const | Getter pour le titre de l'image |
| string getChemin() const | Getter pour le chemin complet vers le dossier de l'image |
| void afficher() | Affiche tous les champs de l'image |
| void setRangDansDiaporama (unsigned int pRangDansDiaporama) | Setter pour le rang de l'image dans le diaporama |
| void setTitre (string pTitre) | Setter pour le titre de l'image |
| void setCategorie (string pCategorie) | Setter pour la catégorie de l'image |
| void setChemin (string pChemin) | Setter pour le chemin complet vers le dossier de l'image |

### Lecteur

|  |  |  |
| --- | --- | --- |
| Attributs | | |
| Nom | Type | Signification |
| idDiaporama | Entier non signé | Identifiant du diaporama courant en BDD |
| posImageCourante | Entier non signé | Position de l'image courante dans le diaporama |
| diaporama | Diaporama | Pointeur vers le diaporama associé au lecteur |

|  |  |
| --- | --- |
| Méthodes | |
| Nom | Signification et but |
| Lecteur() | Constructeur du lecteur |
| ~Lecteur() | Destructeur du lecteur |
| unsigned int getIdDiaporama() const | Getter qui permet de récupérer l'identifiant du diaporama courant |
| Diaporama\* getDiaporama() const | Getter qui permet de récupérer le pointeur vers le diaporama associé |
| unsigned int getPosImageCourante() const | Getter qui permet de récupérer la position de l'image courante |
| bool lecteurVide() const | Vérifie si aucun diaporama n'est associé au lecteur |
| ImageDansDiaporama\* getImageCourante() const | Retourne le pointeur vers l'image courante |
| unsigned int nbImages() const | Retourne la taille de la collection pointée par diaporama |
| void afficher() | Affiche les informations sur lecteur + éventuellement diaporama et image courante |
| void setIdDiaporama(unsigned int pIdDiaporama) | Setter qui permet de définir l'identifiant du diaporama courant |
| void setDiaporama (Diaporama\* pDiaporama) | Setter qui permet de définir le pointeur vers le diaporama associé |
| void setPosImageCourante(unsigned int pPosImageCourante) | Setter qui permet de définir la position de l'image courante |
| void changerDiaporama(unsigned int pId, string pTitre="", unsigned int pVitesse=0) | Choisir un diaporama avec possibilité de spécifier l'identifiant, le titre et la vitesse |
| void avancer() | Incrémente la position de l'image courante |
| void reculer() | Décrémente la position de l'image courante |
| void viderLecteur() | Enlève le diaporama courant du lecteur s'il existe |
| void chargerDiaporamaCourant() | Charge dans le lecteur l'image dans le diaporama courant |

### lecteurVue

|  |  |  |
| --- | --- | --- |
| Attributs | | |
| Nom | Type | Signification |
| ui | Ui | Pointeur vers l'interface utilisateur de la fenêtre lecteurVue |
| m\_MaPresentation | Presentation | Pointeur vers l'objet de présentation associé à la vue lecteurVue |

|  |  |
| --- | --- |
| Méthodes | |
| Nom | Signification et but |
| lecteurVue() | Constructeur du lecteur vue |
| ~lecteurVue() | Destructeur du lecteur vue |
| Presentation \* getPresentation() const | Getter pour récupérer le pointeur vers la présentation associé |
| void setPresentation(Presentation \*) | Setter pour définir la présentation associé |
| void initializeConnections() | Connexion entre la présentation et le modèle |
| void demanderAvancer() | Slot qui demande d'avancer dans le diaporama (exécute demanderAvancer() de Presentation) |
| void demanderReculer() | Slot qui demande de reculer dans le diaporama (exécute demanderReculer() de Presentation) |
| void demanderChangerVitesse() | Slot qui demande de changer la vitesse de défilement (exécute demanderChangerVitesse() de Presentation) |
| void demanderChangerModeAutomatique() | Slot qui demande de changer le mode de défilement (exécute demanderChangerModeAutomatique() de Presentation) |
| void demanderChangerModeManuel() | Slot qui demande de changer le mode de défilement (exécute demanderChangerModeManuel() de Presentation) |
| void demanderChargerDiapo() | Slot qui demande de charger le diaporama (exécute demanderChargerDiapo() de Presentation) |
| void quitterApplication() | Slot qui quitte l’application |
| void demanderEnleverDiapo() | Slot qui demande d’enlever le diaporama (exécute demanderEnleverDiapo() de Presentation) |
| void demanderAPropos() | Slot qui demande d’afficher le informations sur l’application (exécute demanderAPropos() de Presentation) |
| void majPresentation(const QString &titreDiapo, const QString &titreImage, const QString &categorie, const QString &rang, const QString &chemin) | Slot qui met à jour les informations de la fenêtre. |

### Modele

|  |  |  |
| --- | --- | --- |
| Attributs | | |
| Nom | Type | Signification |
| m\_titre | Chaîne de caractères | Titre du modèle |
| m\_vitesseDefilement | Entier non signé | Vitesse de défilement des images |
| m\_posImageCourante | Entier non signé | Position de l'image courante |
| m\_localisationImages | Vector <ImageDansDiaporama> | Vecteur d'images dans le diaporama |
| m\_MonDiapo | Diaporama\* | Pointeur vers le diaporama associé |
| images | ImagesDiaporama | Vecteur de pointeurs vers les objets ImageDansDiaporama |
| lecteur | Lecteur\* | Pointeur vers le lecteur |
| UnEtat | enum | Type énuméré contenant l’état du modèle (manuel ou automatique) |
| \_etat | UnEtat | Etat du modèle |
| timer | QTimer\* | Intervalle de temps entre deux images d’un diaporama qui défile automatiquement |

|  |  |
| --- | --- |
| Méthodes | |
| Nom | Signification et but |
| Modele() | Constructeur du modele |
| void triCroissantRang() | Trie les images par ordre croissant de leur rang dans le diaporama |
| bool lecteurVide() const | Si le lecteur est vide ou non |
| unsigned int getVitesseDefilement() const | Getter pour la vitesse de défilement des images |
| unsigned int getPosImageCourante() const | Getter pour la position de l'image courante |
| unsigned int nbImages() const | Retourne le nombre d'images dans le diaporama |
| UnEtat getEtat() const | Getter pour récupérer l’état du modèle |
| Diaporama\* getDiaporama() const | Getter pour récupérer le pointeur vers le diaporama associé |
| void setVitesseDefilement(unsigned int) | Setter pour définir la vitesse de défilement des images |
| void setDiaporama(Diaporama \*) | Setter pour définir le pointeur vers le diaporama associé |
| void setPosImageCourante(unsigned int pPosImageCourante) | Setter pour définir la position de l'image courante |
| void avancer() | Passe à l’image suivante |
| void reculer() | Reviens à l’image précédente |
| void avanceAuto() | Passe à l’image suivante (si le mode est en automatique) |
| void changerVitesse() | Change la vitesse de défilement |
| void etatAutomatique() | Met l’état du modèle à automatique |
| void etatManuel() | Met l’état du modèle à manuel |
| void chargerDiapo() | Charge un diaporama |
| void enleverDiapo() | Enlève un diaporama |
| void aPropos() | Affiche la fenêtre de dialogue apropos |
| Lecteur \* getLecteur() const | Getter qui retourne le lecteur |
| void setLecteur(Lecteur \* newLecteur) | Setter du lecteur |
| void imageChange(const QString &titreDiapo, const QString &titreImage, const QString &categorie, const QString &rang, const QString &chemin) | Signal qui est connecté à majPresentation, permettant la mise à jour des informations affichées sur la fenêtre principale |

### Presentation

|  |  |  |
| --- | --- | --- |
| Attributs | | |
| Nom | Type | Signification |
| \_leModele | Modele\* | Le modèle |
| \_laVue | lecteurVue\* | La vue |
| timer | QTimer\* | Intervalle de temps entre deux images d’un diaporama qui défile automatiquement |

|  |  |
| --- | --- |
| Méthodes | |
| Nom | Signification et but |
| Presentation() | Constructeur de la présentation |
| ~Presentation() | Destructeur de la présentation |
| void setModel(Modele\* m) | Setter pour définir l'objet modèle associé |
| void setVue(lecteurVue\* v) | Setter pour définir l'objet vue associé |
| Modele \*getModele() | Getter pour récupérer le pointeur vers l'objet modèle associé |
| lecteurVue \*getVue() | Getter pour récupérer le pointeur vers l'objet vue associé |
| ImageDansDiaporama\* getImageActuelle() | Getter pour récupérer l’image actuelle |
| Diaporama \* getDiapoActuel() | Getter pour récupérer le diaporama actuel |
| void demanderAvancer() | Demande d'avancer dans le diaporama (avancer() dans \_leModele) |
| void demanderReculer() | Demande de reculer dans le diaporama (reculer() dans \_leModele) |
| void autoAdvance() | Demande de faire avancer le modèle automatiquement (avanceAuto() dans \_leModele) |
| void demanderChangerVitesse() | Demande de changer la vitesse de défilement (changerVitesse() dans \_leModele) |
| void demanderChangerModeAutomatique() | Demande de changer le mode de défilement (etatAutomatique() dans \_leModele) |
| void demanderChangerModeManuel() | Demande de changer le mode de défilement (etatManuel() dans \_leModele) |
| void demanderChargerDiapo() | Demande de charger le diaporama (chargerDiapo() dans \_leModele) |
| void demanderEnleverDiapo() | Demande d’enlever le diaporama (enleverDiapo() dans \_leModele) |
| void demanderAPropos() | Demande d’afficher le informations sur l’application (aPropos() dans \_leModele) |
| void avancerBoucle() | Avance dans le modèle |

# **Version 5 MVP :**

## Diagramme des classes UML

Une image contenant texte, capture d’écran, diagramme, nombre

Description générée automatiquement

## Attributs et méthodes de chaque classe

### apropos

|  |  |  |
| --- | --- | --- |
| Attributs | | |
| Nom | Type | Signification |
| ui | Ui | Pointeur vers l'interface utilisateur de la fenêtre apropos |

|  |  |
| --- | --- |
| Méthodes | |
| Nom | Signification et but |
| ~apropos() | Destructeur de apropos |

### charger

|  |  |  |
| --- | --- | --- |
| Attributs | | |
| Nom | Type | Signification |
| ui | Ui | Pointeur vers l'interface utilisateur de la fenêtre apropos |
| diaporamasDisponibles | Diaporama | Stocke les diaporamas disponibles |

|  |  |
| --- | --- |
| Méthodes | |
| Nom | Signification et but |
| ~charger() | Destructeur de charger |
| void updateDiaporamas(const Diaporamas &diaporamas) | Permet de remplir menu déroulant avec les diaporamas |
| unsigned int getSelectedDiaporamaId() const | Retourne l’id du diaporama courant |
| void diaporamaSelectionne(unsigned int diaporamaId) | Signal qui permet un envoi d’information à la vue |
| void transmettreInformations() | Slot qui emet le signal diaporamaSelectionne |

### Diaporama

|  |  |  |
| --- | --- | --- |
| Attributs | | |
| Nom | Type | Signification |
| id | Entier non signé | Identifiant du diaporama |
| titre | Chaîne de caractères | Titre du diaporama |
| vitesseDefilement | Entier non signé | Vitesse de défilement des images du diaporama |
| images | ImagesDiaporama | Vecteur de pointeurs sur les objets ImageDansDiaporama du diaporama |
| posImageCourante | Entier non signé | Position de l’image courante dans le diaporama |

|  |  |
| --- | --- |
| Méthodes | |
| Nom | Signification et but |
| Diaporama() | Constructeur du diaporama |
| ~Diaporama() | Destructeur de diaporama |
| unsigned int getId() const | Getter pour l'identifiant du diaporama |
| string getTitre() const | Getter pour le titre du diaporama |
| int getVitesseDefilement() const | Getter pour la vitesse de défilement des images |
| ImagesDiaporama getImages() const | Getter pour les images du diaporama |
| ImageDansDiaporama\* getImageCourante() const | Getter de l’image courante le diaporama |
| unsigned int getPosImageCourante() const | Getter de la position de l’image courante dans le diaporama |
| unsigned int nbImages() const | Retourne le nombre d'images contenus dans le diaporama |
| void setId(unsigned int pId) | Setter pour l'identifiant du diaporama |
| void setTitre(string pTitre) | Setter pour le titre du diaporama |
| void setVitesseDefilement(unsigned int pVitesseDefilement) | Setter pour la vitesse de défilement des images |
| void setImages(const ImagesDiaporama& pImages) | Setter pour les images du diaporama |
| void ajouterImageEnFin(ImageDansDiaporama\* pImage) | Ajoute une image à la fin du diaporama |
| void enleverImageEnFin() | Enlève la dernière image du diaporama |
| void vider() | Enlève toutes les images du diaporama |
| void charger() | Charge les images associées au diaporama courant |
| void trierParRangCroissant() | Trie les images du diaporama par ordre de rang croissant |

### ImageDansDiaporama

|  |  |  |
| --- | --- | --- |
| Attributs | | |
| Nom | Type | Signification |
| rangDansDiaporama | Entier non signé | Rang de l'image dans le diaporama |
| titre | Chaîne de caractères | Intitulé de l'image |
| categorie | Chaîne de caractères | Catégorie de l'image (personne, animal, objet) |
| chemin | Chaîne de caractères | Chemin complet vers le dossier où se trouve l'image |

|  |  |
| --- | --- |
| Méthodes | |
| Nom | Signification et but |
| ImageDansDiaporama(unsigned int pRangDansDiaporama=0,  string pCategorie="defaut", string pTitre="defaut", string pChemin = ":/images/Disney\_tapis.gif") | Constructeur de l’image dans le diaporama |
| ~ImageDansDiaporama() | Destructeur de l’image dans le diaporama |
| unsigned int getRangDansDiaporama() const | Getter pour le rang de l'image dans le diaporama |
| string getCategorie() const | Getter pour la catégorie de l'image |
| string getTitre() const | Getter pour le titre de l'image |
| string getChemin() const | Getter pour le chemin complet vers le dossier de l'image |
| void afficher() | Affiche tous les champs de l'image |
| void setRangDansDiaporama (unsigned int pRangDansDiaporama) | Setter pour le rang de l'image dans le diaporama |
| void setTitre (string pTitre) | Setter pour le titre de l'image |
| void setCategorie (string pCategorie) | Setter pour la catégorie de l'image |
| void setChemin (string pChemin) | Setter pour le chemin complet vers le dossier de l'image |

### Lecteur

|  |  |  |
| --- | --- | --- |
| Attributs | | |
| Nom | Type | Signification |
| idDiaporama | Entier non signé | Identifiant du diaporama courant en BDD |
| posImageCourante | Entier non signé | Position de l'image courante dans le diaporama |
| diaporama | Diaporama | Pointeur vers le diaporama associé au lecteur |

|  |  |
| --- | --- |
| Méthodes | |
| Nom | Signification et but |
| Lecteur() | Constructeur du lecteur |
| ~Lecteur() | Destructeur du lecteur |
| unsigned int getIdDiaporama() const | Getter qui permet de récupérer l'identifiant du diaporama courant |
| Diaporama\* getDiaporama() const | Getter qui permet de récupérer le pointeur vers le diaporama associé |
| unsigned int getPosImageCourante() const | Getter qui permet de récupérer la position de l'image courante |
| bool lecteurVide() const | Vérifie si aucun diaporama n'est associé au lecteur |
| ImageDansDiaporama\* getImageCourante() const | Retourne le pointeur vers l'image courante |
| unsigned int nbImages() const | Retourne la taille de la collection pointée par diaporama |
| void afficher() | Affiche les informations sur lecteur + éventuellement diaporama et image courante |
| void setIdDiaporama(unsigned int pIdDiaporama) | Setter qui permet de définir l'identifiant du diaporama courant |
| void setDiaporama (Diaporama\* pDiaporama) | Setter qui permet de définir le pointeur vers le diaporama associé |
| void setPosImageCourante(unsigned int pPosImageCourante) | Setter qui permet de définir la position de l'image courante |
| void changerDiaporama(unsigned int pId, string pTitre="", unsigned int pVitesse=0) | Choisir un diaporama avec possibilité de spécifier l'identifiant, le titre et la vitesse |
| void avancer() | Incrémente la position de l'image courante |
| void reculer() | Décrémente la position de l'image courante |
| void viderLecteur() | Enlève le diaporama courant du lecteur s'il existe |
| void chargerDiaporamaCourant() | Charge dans le lecteur l'image dans le diaporama courant |

### lecteurVue

|  |  |  |
| --- | --- | --- |
| Attributs | | |
| Nom | Type | Signification |
| ui | Ui | Pointeur vers l'interface utilisateur de la fenêtre lecteurVue |
| m\_MaPresentation | Presentation | Pointeur vers l'objet de présentation associé à la vue lecteurVue |
| monLecteur | Lecteur | Pointeur vers l’objet du lecteur associé à la vue |
| modele | Modele | Pointeur vers l'objet de modèle associé à la vue lecteurVue |
| InfosDiaporama | enregistrement | Contient l’id, le titre et la vitesseDefilement du diaporama |
| diaporamas | Diaporamas | Vecteur contenant tous les InfosDiaporamas |

|  |  |
| --- | --- |
| Méthodes | |
| Nom | Signification et but |
| lecteurVue() | Constructeur du lecteur vue |
| ~lecteurVue() | Destructeur du lecteur vue |
| Presentation \* getPresentation() const | Getter pour récupérer le pointeur vers la présentation associé |
| void setPresentation(Presentation \*) | Setter pour définir la présentation associé |
| void initializeConnections() | Connexion entre la présentation et le modèle |
| void majInterface(Modele::UnEtat) | Met à jour les bouttons cliquables en fonction de l’état du modèle |
| void listeDiaporamas(Diaporamas &diaporamas) | Permet de charger ‘en dur’ les informations des diaporamas |
| void demanderAvancer() | Slot qui demande d'avancer dans le diaporama (exécute demanderAvancer() de Presentation) |
| void demanderReculer() | Slot qui demande de reculer dans le diaporama (exécute demanderReculer() de Presentation) |
| void demanderChangerVitesse() | Slot qui demande de changer la vitesse de défilement (exécute demanderChangerVitesse() de Presentation) |
| void demanderChangerModeAutomatique() | Slot qui demande de changer le mode de défilement (exécute demanderChangerModeAutomatique() de Presentation) |
| void demanderChangerModeManuel() | Slot qui demande de changer le mode de défilement (exécute demanderChangerModeManuel() de Presentation) |
| void demanderChargerDiapo() | Slot qui demande de charger le diaporama (exécute demanderChargerDiapo() de Presentation) |
| void recevoirDiaporamaSelectionne(unsigned int diaporamaId) | Met à jour les informations sur le diaporama avec celles reçues de la fenetre charger |
| void quitterApplication() | Slot qui quitte l’application |
| void demanderEnleverDiapo() | Slot qui demande d’enlever le diaporama (exécute demanderEnleverDiapo() de Presentation) |
| void demanderAPropos() | Slot qui demande d’afficher le informations sur l’application (exécute demanderAPropos() de Presentation) |
| void majPresentation(const QString &titreDiapo, const QString &titreImage, const QString &categorie, const QString &rang, const QString &chemin) | Slot qui met à jour les informations de la fenêtre. |

### Modele

|  |  |  |
| --- | --- | --- |
| Attributs | | |
| Nom | Type | Signification |
| m\_titre | Chaîne de caractères | Titre du modèle |
| m\_vitesseDefilement | Entier non signé | Vitesse de défilement des images |
| m\_posImageCourante | Entier non signé | Position de l'image courante |
| m\_localisationImages | Vector <ImageDansDiaporama> | Vecteur d'images dans le diaporama |
| m\_MonDiapo | Diaporama\* | Pointeur vers le diaporama associé |
| images | ImagesDiaporama | Vecteur de pointeurs vers les objets ImageDansDiaporama |
| lecteur | Lecteur\* | Pointeur vers le lecteur |
| UnEtat | enum | Type énuméré contenant l’état du modèle (manuel ou automatique) |
| \_etat | UnEtat | Etat du modèle |
| timer | QTimer\* | Intervalle de temps entre deux images d’un diaporama qui défile automatiquement |

|  |  |
| --- | --- |
| Méthodes | |
| Nom | Signification et but |
| Modele() | Constructeur du modele |
| void triCroissantRang() | Trie les images par ordre croissant de leur rang dans le diaporama |
| bool lecteurVide() const | Si le lecteur est vide ou non |
| unsigned int getVitesseDefilement() const | Getter pour la vitesse de défilement des images |
| unsigned int getPosImageCourante() const | Getter pour la position de l'image courante |
| unsigned int nbImages() const | Retourne le nombre d'images dans le diaporama |
| UnEtat getEtat() const | Getter pour récupérer l’état du modèle |
| Diaporama\* getDiaporama() const | Getter pour récupérer le pointeur vers le diaporama associé |
| void setVitesseDefilement(unsigned int) | Setter pour définir la vitesse de défilement des images |
| void setDiaporama(Diaporama \*) | Setter pour définir le pointeur vers le diaporama associé |
| void setPosImageCourante(unsigned int pPosImageCourante) | Setter pour définir la position de l'image courante |
| void chargerDiapo(unsigned int id, const QString &titre, int vitesseDefilement) | Méthode pour charger un diaporama |
| void avancer() | Passe à l’image suivante |
| void reculer() | Reviens à l’image précédente |
| void avanceAuto() | Passe à l’image suivante (si le mode est en automatique) |
| void changerVitesse() | Change la vitesse de défilement |
| void etatAutomatique() | Met l’état du modèle à automatique |
| void etatManuel() | Met l’état du modèle à manuel |
| void enleverDiapo() | Enlève un diaporama |
| void aPropos() | Affiche la fenêtre de dialogue apropos |
| Lecteur \* getLecteur() const | Getter qui retourne le lecteur |
| void setLecteur(Lecteur \* newLecteur) | Setter du lecteur |
| void imageChange(const QString &titreDiapo, const QString &titreImage, const QString &categorie, const QString &rang, const QString &chemin) | Signal qui est connecté à majPresentation, permettant la mise à jour des informations affichées sur la fenêtre principale |

### Presentation

|  |  |  |
| --- | --- | --- |
| Attributs | | |
| Nom | Type | Signification |
| \_leModele | Modele\* | Le modèle |
| \_laVue | lecteurVue\* | La vue |
| timer | QTimer\* | Intervalle de temps entre deux images d’un diaporama qui défile automatiquement |
| intervalleTimer | entier | Intervalle de temps du timer (ici défini à 1000 ms) |

|  |  |
| --- | --- |
| Méthodes | |
| Nom | Signification et but |
| Presentation() | Constructeur de la présentation |
| ~Presentation() | Destructeur de la présentation |
| void setModel(Modele\* m) | Setter pour définir l'objet modèle associé |
| void setVue(lecteurVue\* v) | Setter pour définir l'objet vue associé |
| Modele \*getModele() | Getter pour récupérer le pointeur vers l'objet modèle associé |
| lecteurVue \*getVue() | Getter pour récupérer le pointeur vers l'objet vue associé |
| ImageDansDiaporama\* getImageActuelle() | Getter pour récupérer l’image actuelle |
| Diaporama \* getDiapoActuel() | Getter pour récupérer le diaporama actuel |
| int getVitesse() | Getter pour récupérer la vitesse de défilement |
| void demanderAvancer() | Demande d'avancer dans le diaporama (avancer() dans \_leModele) |
| void demanderReculer() | Demande de reculer dans le diaporama (reculer() dans \_leModele) |
| void autoAdvance() | Demande de faire avancer le modèle automatiquement (avanceAuto() dans \_leModele) |
| void demanderChangerVitesse(vit\* v) | Demande de changer la vitesse de défilement (changerVitesse() dans \_leModele) |
| void demanderChangerModeAutomatique() | Demande de changer le mode de défilement (etatAutomatique() dans \_leModele) |
| void demanderChangerModeManuel() | Demande de changer le mode de défilement (etatManuel() dans \_leModele) |
| void demanderChargerDiapo() | Demande de charger le diaporama (chargerDiapo() dans \_leModele) |
| void demanderEnleverDiapo() | Demande d’enlever le diaporama (enleverDiapo() dans \_leModele) |
| void demanderAPropos() | Demande d’afficher le informations sur l’application (aPropos() dans \_leModele) |
| void avancerBoucle() | Avance dans le modèle |
| void ajusterVitesseDiaporama(int vitesse) | Permet de démarrer le timer avec la vitesse choisie |

### vit

|  |  |  |
| --- | --- | --- |
| Attributs | | |
| Nom | Type | Signification |
| ui | Ui | Pointeur vers l'interface utilisateur de la fenêtre vit |

|  |  |
| --- | --- |
| Méthodes | |
| Nom | Signification et but |
| ~vit() | Destructeur de vit |

# **Version 7 MVP :**

## Diagramme des classes UML

Une image contenant texte, capture d’écran, diagramme, nombre

Description générée automatiquement

## Attributs et méthodes de chaque classe

### apropos

|  |  |  |
| --- | --- | --- |
| Attributs | | |
| Nom | Type | Signification |
| ui | Ui | Pointeur vers l'interface utilisateur de la fenêtre apropos |

|  |  |
| --- | --- |
| Méthodes | |
| Nom | Signification et but |
| ~apropos() | Destructeur de apropos |

### charger

|  |  |  |
| --- | --- | --- |
| Attributs | | |
| Nom | Type | Signification |
| ui | Ui | Pointeur vers l'interface utilisateur de la fenêtre apropos |
| diaporamasDisponibles | Diaporama | Stocke les diaporamas disponibles |

|  |  |
| --- | --- |
| Méthodes | |
| Nom | Signification et but |
| ~charger() | Destructeur de charger |
| void updateDiaporamas(const Diaporamas &diaporamas) | Permet de remplir menu déroulant avec les diaporamas |
| unsigned int getSelectedDiaporamaId() const | Retourne l’id du diaporama courant |
| void diaporamaSelectionne(unsigned int diaporamaId) | Signal qui permet un envoi d’information à la vue |
| void transmettreInformations() | Slot qui emet le signal diaporamaSelectionne |

### database

|  |  |  |
| --- | --- | --- |
| Attributs | | |
| Nom | Type | Signification |
| maBD | QSqlDatabase | Objet base de donnée |

|  |  |
| --- | --- |
| Méthodes | |
| Nom | Signification et but |
| database() | Constructeur de database |
| ~database() | Destructeur de database |
| bool ouvrirBD() | Ouvre la connexion à la base de données |
| bool fermerBD() | Ferme la connexion à la base de données |
| QSqlDatabase getDatabase() const | Récupère l'objet base de données |

### Diaporama

|  |  |  |
| --- | --- | --- |
| Attributs | | |
| Nom | Type | Signification |
| id | Entier non signé | Identifiant du diaporama |
| titre | Chaîne de caractères | Titre du diaporama |
| vitesseDefilement | Entier non signé | Vitesse de défilement des images du diaporama |
| images | ImagesDiaporama | Vecteur de pointeurs sur les objets ImageDansDiaporama du diaporama |
| posImageCourante | Entier non signé | Position de l’image courante dans le diaporama |
| maBD | database | Objet de la base de donnée |

|  |  |
| --- | --- |
| Méthodes | |
| Nom | Signification et but |
| Diaporama() | Constructeur du diaporama |
| ~Diaporama() | Destructeur de diaporama |
| unsigned int getId() const | Getter pour l'identifiant du diaporama |
| string getTitre() const | Getter pour le titre du diaporama |
| int getVitesseDefilement() const | Getter pour la vitesse de défilement des images |
| ImagesDiaporama getImages() const | Getter pour les images du diaporama |
| ImageDansDiaporama\* getImageCourante() const | Getter de l’image courante le diaporama |
| unsigned int getPosImageCourante() const | Getter de la position de l’image courante dans le diaporama |
| unsigned int nbImages() const | Retourne le nombre d'images contenus dans le diaporama |
| void setId(unsigned int pId) | Setter pour l'identifiant du diaporama |
| void setTitre(string pTitre) | Setter pour le titre du diaporama |
| void setVitesseDefilement(unsigned int pVitesseDefilement) | Setter pour la vitesse de défilement des images |
| void setImages(const ImagesDiaporama& pImages) | Setter pour les images du diaporama |
| void ajouterImageEnFin(ImageDansDiaporama\* pImage) | Ajoute une image à la fin du diaporama |
| void enleverImageEnFin() | Enlève la dernière image du diaporama |
| void vider() | Enlève toutes les images du diaporama |
| void charger() | Charge les images associées au diaporama courant |
| void trierParRangCroissant() | Trie les images du diaporama par ordre de rang croissant |

### ImageDansDiaporama

|  |  |  |
| --- | --- | --- |
| Attributs | | |
| Nom | Type | Signification |
| rangDansDiaporama | Entier non signé | Rang de l'image dans le diaporama |
| titre | Chaîne de caractères | Intitulé de l'image |
| categorie | Chaîne de caractères | Catégorie de l'image (personne, animal, objet) |
| chemin | Chaîne de caractères | Chemin complet vers le dossier où se trouve l'image |

|  |  |
| --- | --- |
| Méthodes | |
| Nom | Signification et but |
| ImageDansDiaporama(unsigned int pRangDansDiaporama=0,  string pCategorie="defaut", string pTitre="defaut", string pChemin = ":/images/Disney\_tapis.gif") | Constructeur de l’image dans le diaporama |
| ~ImageDansDiaporama() | Destructeur de l’image dans le diaporama |
| unsigned int getRangDansDiaporama() const | Getter pour le rang de l'image dans le diaporama |
| string getCategorie() const | Getter pour la catégorie de l'image |
| string getTitre() const | Getter pour le titre de l'image |
| string getChemin() const | Getter pour le chemin complet vers le dossier de l'image |
| void afficher() | Affiche tous les champs de l'image |
| void setRangDansDiaporama (unsigned int pRangDansDiaporama) | Setter pour le rang de l'image dans le diaporama |
| void setTitre (string pTitre) | Setter pour le titre de l'image |
| void setCategorie (string pCategorie) | Setter pour la catégorie de l'image |
| void setChemin (string pChemin) | Setter pour le chemin complet vers le dossier de l'image |

### Lecteur

|  |  |  |
| --- | --- | --- |
| Attributs | | |
| Nom | Type | Signification |
| idDiaporama | Entier non signé | Identifiant du diaporama courant en BDD |
| posImageCourante | Entier non signé | Position de l'image courante dans le diaporama |
| diaporama | Diaporama | Pointeur vers le diaporama associé au lecteur |

|  |  |
| --- | --- |
| Méthodes | |
| Nom | Signification et but |
| Lecteur() | Constructeur du lecteur |
| ~Lecteur() | Destructeur du lecteur |
| unsigned int getIdDiaporama() const | Getter qui permet de récupérer l'identifiant du diaporama courant |
| Diaporama\* getDiaporama() const | Getter qui permet de récupérer le pointeur vers le diaporama associé |
| unsigned int getPosImageCourante() const | Getter qui permet de récupérer la position de l'image courante |
| bool lecteurVide() const | Vérifie si aucun diaporama n'est associé au lecteur |
| ImageDansDiaporama\* getImageCourante() const | Retourne le pointeur vers l'image courante |
| unsigned int nbImages() const | Retourne la taille de la collection pointée par diaporama |
| void afficher() | Affiche les informations sur lecteur + éventuellement diaporama et image courante |
| void setIdDiaporama(unsigned int pIdDiaporama) | Setter qui permet de définir l'identifiant du diaporama courant |
| void setDiaporama (Diaporama\* pDiaporama) | Setter qui permet de définir le pointeur vers le diaporama associé |
| void setPosImageCourante(unsigned int pPosImageCourante) | Setter qui permet de définir la position de l'image courante |
| void changerDiaporama(unsigned int pId, string pTitre="", unsigned int pVitesse=0) | Choisir un diaporama avec possibilité de spécifier l'identifiant, le titre et la vitesse |
| void avancer() | Incrémente la position de l'image courante |
| void reculer() | Décrémente la position de l'image courante |
| void viderLecteur() | Enlève le diaporama courant du lecteur s'il existe |
| void chargerDiaporamaCourant() | Charge dans le lecteur l'image dans le diaporama courant |

### lecteurVue

|  |  |  |
| --- | --- | --- |
| Attributs | | |
| Nom | Type | Signification |
| ui | Ui | Pointeur vers l'interface utilisateur de la fenêtre lecteurVue |
| m\_MaPresentation | Presentation | Pointeur vers l'objet de présentation associé à la vue lecteurVue |
| monLecteur | Lecteur | Pointeur vers l’objet du lecteur associé à la vue |
| modele | Modele | Pointeur vers l'objet de modèle associé à la vue lecteurVue |
| InfosDiaporama | enregistrement | Contient l’id, le titre et la vitesseDefilement du diaporama |
| diaporamas | Diaporamas | Vecteur contenant tous les InfosDiaporamas |

|  |  |
| --- | --- |
| Méthodes | |
| Nom | Signification et but |
| lecteurVue() | Constructeur du lecteur vue |
| ~lecteurVue() | Destructeur du lecteur vue |
| Presentation \* getPresentation() const | Getter pour récupérer le pointeur vers la présentation associé |
| void setPresentation(Presentation \*) | Setter pour définir la présentation associé |
| void initializeConnections() | Connexion entre la présentation et le modèle |
| void majInterface(Modele::UnEtat) | Met à jour les bouttons cliquables en fonction de l’état du modèle |
| void listeDiaporamas(Diaporamas &diaporamas) | Permet de charger ‘en dur’ les informations des diaporamas |
| void demanderAvancer() | Slot qui demande d'avancer dans le diaporama (exécute demanderAvancer() de Presentation) |
| void demanderReculer() | Slot qui demande de reculer dans le diaporama (exécute demanderReculer() de Presentation) |
| void demanderChangerVitesse() | Slot qui demande de changer la vitesse de défilement (exécute demanderChangerVitesse() de Presentation) |
| void demanderChangerModeAutomatique() | Slot qui demande de changer le mode de défilement (exécute demanderChangerModeAutomatique() de Presentation) |
| void demanderChangerModeManuel() | Slot qui demande de changer le mode de défilement (exécute demanderChangerModeManuel() de Presentation) |
| void demanderChargerDiapo() | Slot qui demande de charger le diaporama (exécute demanderChargerDiapo() de Presentation) |
| void recevoirDiaporamaSelectionne(unsigned int diaporamaId) | Met à jour les informations sur le diaporama avec celles reçues de la fenetre charger |
| void quitterApplication() | Slot qui quitte l’application |
| void demanderEnleverDiapo() | Slot qui demande d’enlever le diaporama (exécute demanderEnleverDiapo() de Presentation) |
| void demanderAPropos() | Slot qui demande d’afficher le informations sur l’application (exécute demanderAPropos() de Presentation) |
| void majPresentation(const QString &titreDiapo, const QString &titreImage, const QString &categorie, const QString &rang, const QString &chemin) | Slot qui met à jour les informations de la fenêtre. |

### Modele

|  |  |  |
| --- | --- | --- |
| Attributs | | |
| Nom | Type | Signification |
| m\_titre | Chaîne de caractères | Titre du modèle |
| m\_vitesseDefilement | Entier non signé | Vitesse de défilement des images |
| m\_posImageCourante | Entier non signé | Position de l'image courante |
| m\_localisationImages | Vector <ImageDansDiaporama> | Vecteur d'images dans le diaporama |
| m\_MonDiapo | Diaporama\* | Pointeur vers le diaporama associé |
| images | ImagesDiaporama | Vecteur de pointeurs vers les objets ImageDansDiaporama |
| lecteur | Lecteur\* | Pointeur vers le lecteur |
| UnEtat | enum | Type énuméré contenant l’état du modèle (manuel ou automatique) |
| \_etat | UnEtat | Etat du modèle |
| timer | QTimer\* | Intervalle de temps entre deux images d’un diaporama qui défile automatiquement |

|  |  |
| --- | --- |
| Méthodes | |
| Nom | Signification et but |
| Modele() | Constructeur du modele |
| void triCroissantRang() | Trie les images par ordre croissant de leur rang dans le diaporama |
| bool lecteurVide() const | Si le lecteur est vide ou non |
| unsigned int getVitesseDefilement() const | Getter pour la vitesse de défilement des images |
| unsigned int getPosImageCourante() const | Getter pour la position de l'image courante |
| unsigned int nbImages() const | Retourne le nombre d'images dans le diaporama |
| UnEtat getEtat() const | Getter pour récupérer l’état du modèle |
| Diaporama\* getDiaporama() const | Getter pour récupérer le pointeur vers le diaporama associé |
| void setVitesseDefilement(unsigned int) | Setter pour définir la vitesse de défilement des images |
| void setDiaporama(Diaporama \*) | Setter pour définir le pointeur vers le diaporama associé |
| void setPosImageCourante(unsigned int pPosImageCourante) | Setter pour définir la position de l'image courante |
| void setEtat(UnEtat e) | Setter pour définir l’état du modèle |
| void chargerDiapo(unsigned int id, const QString &titre, int vitesseDefilement) | Méthode pour charger un diaporama |
| void avancer() | Passe à l’image suivante |
| void reculer() | Reviens à l’image précédente |
| void avanceAuto() | Passe à l’image suivante (si le mode est en automatique) |
| void changerVitesse(vit\*) | Change la vitesse de défilement |
| void etatAutomatique() | Met l’état du modèle à automatique |
| void etatManuel() | Met l’état du modèle à manuel |
| void enleverDiapo() | Enlève un diaporama |
| void aPropos() | Affiche la fenêtre de dialogue apropos |
| Lecteur \* getLecteur() const | Getter qui retourne le lecteur |
| void setLecteur(Lecteur \* newLecteur) | Setter du lecteur |
| void imageChange(const QString &titreDiapo, const QString &titreImage, const QString &categorie, const QString &rang, const QString &chemin) | Signal qui est connecté à majPresentation, permettant la mise à jour des informations affichées sur la fenêtre principale |
| void vitesseChangee(int vitesse) | Signal émis quand la vitesse change |

### Presentation

|  |  |  |
| --- | --- | --- |
| Attributs | | |
| Nom | Type | Signification |
| \_leModele | Modele\* | Le modèle |
| \_laVue | lecteurVue\* | La vue |
| timer | QTimer\* | Intervalle de temps entre deux images d’un diaporama qui défile automatiquement |
| intervalleTimer | entier | Intervalle de temps du timer (ici défini à 1000 ms) |

|  |  |
| --- | --- |
| Méthodes | |
| Nom | Signification et but |
| Presentation() | Constructeur de la présentation |
| ~Presentation() | Destructeur de la présentation |
| void setModel(Modele\* m) | Setter pour définir l'objet modèle associé |
| void setVue(lecteurVue\* v) | Setter pour définir l'objet vue associé |
| Modele \*getModele() | Getter pour récupérer le pointeur vers l'objet modèle associé |
| lecteurVue \*getVue() | Getter pour récupérer le pointeur vers l'objet vue associé |
| ImageDansDiaporama\* getImageActuelle() | Getter pour récupérer l’image actuelle |
| Diaporama \* getDiapoActuel() | Getter pour récupérer le diaporama actuel |
| int getVitesse() | Getter pour récupérer la vitesse de défilement |
| void demanderAvancer() | Demande d'avancer dans le diaporama (avancer() dans \_leModele) |
| void demanderReculer() | Demande de reculer dans le diaporama (reculer() dans \_leModele) |
| void autoAdvance() | Demande de faire avancer le modèle automatiquement (avanceAuto() dans \_leModele) |
| void demanderChangerVitesse(vit\* v) | Demande de changer la vitesse de défilement (changerVitesse() dans \_leModele) |
| void demanderChangerModeAutomatique() | Demande de changer le mode de défilement (etatAutomatique() dans \_leModele) |
| void demanderChangerModeManuel() | Demande de changer le mode de défilement (etatManuel() dans \_leModele) |
| void demanderChargerDiapo() | Demande de charger le diaporama (chargerDiapo() dans \_leModele) |
| void demanderEnleverDiapo() | Demande d’enlever le diaporama (enleverDiapo() dans \_leModele) |
| void demanderAPropos() | Demande d’afficher le informations sur l’application (aPropos() dans \_leModele) |
| void avancerBoucle() | Avance dans le modèle |
| void ajusterVitesseDiaporama(int vitesse) | Permet de démarrer le timer avec la vitesse choisie |

### vit

|  |  |  |
| --- | --- | --- |
| Attributs | | |
| Nom | Type | Signification |
| ui | Ui | Pointeur vers l'interface utilisateur de la fenêtre vit |

|  |  |
| --- | --- |
| Méthodes | |
| Nom | Signification et but |
| ~vit() | Destructeur de vit |

# **Bilan**

## Ce que l’on a appris

Nous avons appris à mieux comprendre un code qui nous était inconnu et à réaliser des classes à partir de ce code. Nous avons aussi perfectionné notre usage de Git/GitHub pour la sauvegarde de fichiers.

## Ce qu’on a aimé / pas aimé

Dans la version 1, nous avons apprécié la mise en place de GitHub (bien que périlleuse dans un premier temps). Aussi, la réflexion pour créer le diagramme de classe était agréable (nous avons une bonne cohésion de groupe).

Nous n'avons pas apprécié le fait de repasser à plusieurs le code pour trouver les erreurs dans les nombreux fichiers, et faire en sorte de l’optimiser et qu’il fonctionne au mieux (même si nous sommes conscient que c’est normal / nécessaire si nous souhaitons rendre un travail de qualité).

Dans la version 3, nous avons vraiment eu du mal à changer d’image et à mettre à jour la fenêtre de par notre mauvaise compréhension du modèle MVP ; en conséquence, nous avons pris énormément de retard sur cette version.

## Ce qui a été difficile

L’assimilation du code a été périlleuse, mais le plus compliqué était trouver toutes les méthodes. Nous avons dû repasser plusieurs fois par-dessus pour être sûr de ne pas en oublier, ce qui nous a pris du temps.

## Le temps passé (sur conception / code)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | CLEMENCEAU Edouard | | MASSON Rafael | | VINET LATRILLE Jules | |
| Versions | Conception | Code | Conception | Code | Conception | Code |
| Version 1 | 3h | 4h | 3h | 4h30 | 2h | 6h30 |
| Version 2 | 2h | 1h | 2h | 1h | 3h30 | 6h |
| Version 2 MVP | 2h30 | 2h30 | 2h | 3h | 2h30 | 7h |
| Version 3 MVP | 4h | 10h | 8h | 5h | 4h | 18h |
| Version 4 MVP | 1h | 4h | 3h | 2h | 1h | 4h |
| Version 5 MVP | 1h | 5h | 4h | 3h | 1h | 7h |
| Version 7 MVP | 2h | 6h | 5h | 4h | 2h | 8h |

## Ce que vous auriez pu faire de mieux (avec le recul)

Nous aurions certainement pu améliorer nos connexions entre fichiers, car le patron MVP n’est pas tout à fait respecté. En effet, nous avons eu beaucoup de mal à faire le lien entre nos différents objets.

## Ce qui pourrait être amélioré dans la SAE

Ce qui pourrait être amélioré dans la SAE serait d'avoir plus de cours encadrés pour pouvoir poser davantage de questions aux professeurs, ou bien d'allouer plus d'heures globales (encadrées / en autonomie) afin de permettre aux étudiants de perfectionner leur travail.

Aussi, habituer les étudiants à GIT dès le début de la première année serait une bonne idée. Ce n’est pas compliqué à utiliser, et ça leur permettrait de mieux organiser leur travail.