

# Comic Book Reader

Projet IN204

Agathe BEUCHER et Magali CASAMAYOU

ENSTA Paris

21/03/2024



INSTITUT  
POLYTECHNIQUE  
DE PARIS

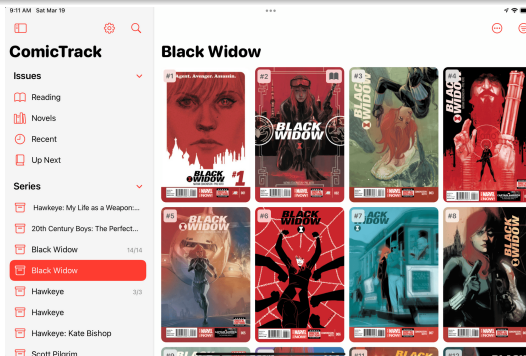
- 1 Analyse d'usage
- 2 Analyse fonctionnelle
- 3 Architecture gros grain
- 4 Démonstrateur
- 5 Pistes d'amélioration
- 6 Modularité et extensibilité
- 7 Conclusion

- 1 Analyse d'usage
- 2 Analyse fonctionnelle
- 3 Architecture gros grain
- 4 Démonstrateur
- 5 Pistes d'amélioration
- 6 Modularité et extensibilité
- 7 Conclusion

# Contexte d'utilisation

## Objectif

Fournir une expérience de lecture de bandes dessinées numérique fluide et agréable



# Cahier des charges

## Navigation intuitive

Parcourir facilement le comic, avec des boutons de navigation et raccourcis clavier

## Affichage Personnalisé

Zoom in/out et redimensionnement de la fenêtre pour l'adapter à l'écran.

## Marquage

Ajout de marque-page pour reprendre la lecture à un endroit spécifique.

# Cahier des charges

## Création de comicbook

Possibilité d'extraire une ou plusieurs pages dans un fichier .CBZ.

## Support de Formats Variés

Prise en charge des formats de fichiers courants tels que CBZ, CBR.

## Table des matières et Informations

Navigation à travers une table des matières et affichage d'information sur le comic

# Définition interactions utilisateur/logiciel

## ① Interface Principale :

- Les pages de la bande dessinée affichées dans la zone centrale
- Barre de menu pour accéder aux différentes fonctionnalités (Ouvrir, extraire, zoom, redimensionnement...)
- Affichage d'informations et de table des matières sur le livre

## ② Navigation et Lecture :

- Utilisation de boutons de navigation (avant, arrière), raccourcis clavier, ou clics pour naviguer entre les pages.

## ③ Affichage Personnalisé :

- Options de zoom et de mise en page dans la barre d'outils.
- Menu contextuel pour ajuster les paramètres d'affichage.

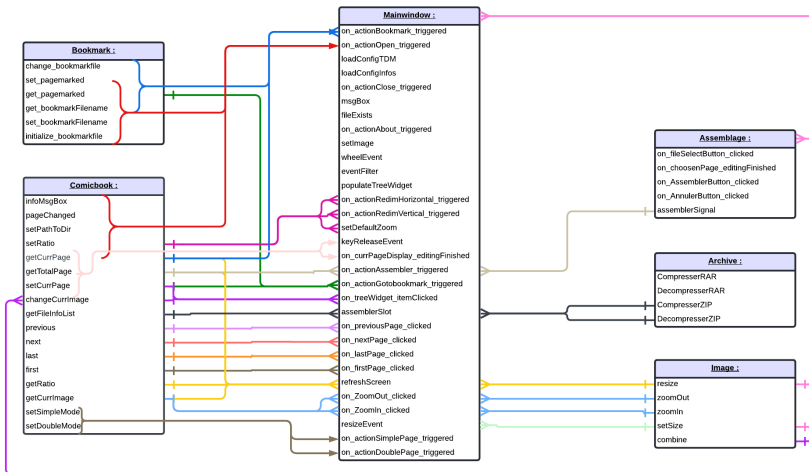
## ④ Marquage et Signets :

- Bouton dédié pour marquer les pages.
- Bouton dédié pour aller à la page marquée.

- 1 Analyse d'usage
- 2 Analyse fonctionnelle**
- 3 Architecture gros grain
- 4 Démonstrateur
- 5 Pistes d'amélioration
- 6 Modularité et extensibilité
- 7 Conclusion



# Blocs fonctionnel du programme



# Définition interactions avec l'environnement au niveau fonctionnel

## Interface mainwindow

### QPushButton

- ZoomIn
- ZoomOut
- firstPage
- lastPage
- nextPage
- ...

### QMenu

- menuAffichage
- menuBookmark
- menuFile

### QAction

- actionDoublePage
- actionSimplePage
- actionRedimHor
- actionRedimVer
- action Bookmark
- ...

### QLabel

- lastBookmark
- totalPageDisplay
- screen

### QtreeWidget

- treeWidget

### QListWidget

- Infos

### QLineEdit

- currPageDisplay

# Définition interactions avec l'environnement au niveau fonctionnel

## Interface assemblage window

### QPushButton

- AnnulerButton
- AssemblerButton
- fileSelectButton

### QLabel

- lieudarchive
- pageaassembler

### QlineEdit

- chosenPage

- 1 Analyse d'usage
- 2 Analyse fonctionnelle
- 3 Architecture gros grain**
- 4 Démonstrateur
- 5 Pistes d'amélioration
- 6 Modularité et extensibilité
- 7 Conclusion

# Classes principales

- ① Classe **comicbook** :
  - Gère la manipulation du livre et la gestion des pages
- ② Classe **archive** :
  - Gère la compression et la décompression d'archive
- ③ Classe **assemblage** :
  - Gère l'extraction et la compression de page du livre
- ④ Classe **bookmark** :
  - Gère le marquage de page, la mémorisation et la navigation du marque-page
- ⑤ Classe **image** :
  - Gère le traitement et le redimensionnement des images

# Classe **comicbook**

## Fields

- pathToDir
- tabPathToImage
- ratio
- currPage
- totalPage
- simpleMode
- doubleMode
- coverPageMode
- curImage

## Méthodes

- setPathToDir
- setRatio
- setSimpleMode
- setDoubleMode
- setCurrPage
- changeCurImage
- next
- previous
- last
- first
- ...

# Classe **bookmark**

## Fields

- page\_marked
- bookmarkFilename

## Méthodes

- set\_bookmarkFilename
- set\_pagemarked
- get\_pagemarked
- get\_bookmarkFilename
- initialize\_bookmarkfile
- change\_bookmarkfile

# Classe **assemblage**

## Fields

- zipPath
- pageList
- maxPage

## Méthodes

- on\_fileSelectButton\_clicked
- on\_choosenPage\_editingFinished
- on\_AssemblerButton\_clicked
- on\_AnnulerButton\_clicked



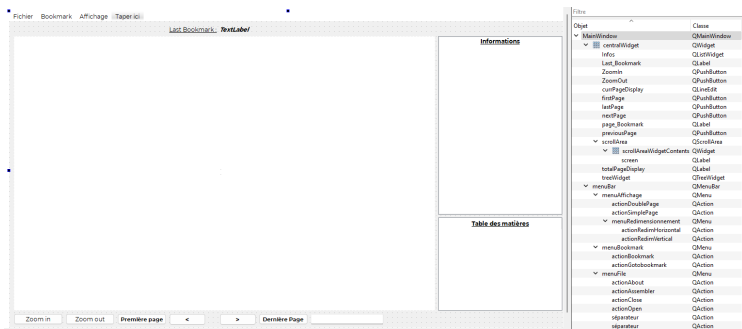
# Classe archive

## Méthodes

- Compresser
- Décompresser

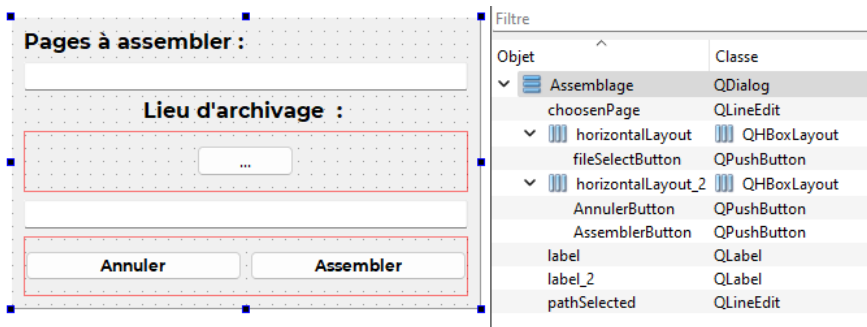
- 1 Analyse d'usage
- 2 Analyse fonctionnelle
- 3 Architecture gros grain
- 4 Démonstrateur**
- 5 Pistes d'amélioration
- 6 Modularité et extensibilité
- 7 Conclusion

# Interface graphique



mainwindow.ui

# Interface graphique



assemblagewindow.ui

# Prototype



- 1 Analyse d'usage
- 2 Analyse fonctionnelle
- 3 Architecture gros grain
- 4 Démonstrateur
- 5 Pistes d'amélioration**
- 6 Modularité et extensibilité
- 7 Conclusion

## Fonctionnalités à ajouter

### Gestion de la mémoire

Optimisation de l'occupation mémoire et la préparation des "pages" à afficher à l'écran

### Gestion de la collection

Création d'une bibliothèque d'ouvrage accessible et navigable directement sur la plateforme

### Gestion d'une application

Extrapolation du logiciel sur application (gestion tactile...)

- 1 Analyse d'usage
- 2 Analyse fonctionnelle
- 3 Architecture gros grain
- 4 Démonstrateur
- 5 Pistes d'amélioration
- 6 Modularité et extensibilité**
- 7 Conclusion



## Modularité et extensibilité

Possibilité de pouvoir étendre le logiciel avec de nouveaux formats d'image, de nouveaux formats de livre et aussi de nouveaux algorithmes de filtrage d'image sans avoir besoin de recoder mais uniquement en **dérivant des objets**.

- 1 Analyse d'usage
- 2 Analyse fonctionnelle
- 3 Architecture gros grain
- 4 Démonstrateur
- 5 Pistes d'amélioration
- 6 Modularité et extensibilité
- 7 Conclusion**

# Conclusion

Merci pour votre attention !