



UNIVERSITÉ  
CAEN  
NORMANDIE

## 2ÈME LICENCE INFORMATIQUE

GROUPE 2B

RAPPORT

---

# Éditeur de livres dont vous êtes le héros

---

AUTEURS :

WAKARY DOUCOURE

FAICAL SID AHMED RAHMANI

MATEJ TRPKOVSKI

MICHEL KAZADI KALALA

20 Avril 2021

# Table des matières

<b>Introduction</b>	<b>1</b>
<b>1 Organisation du projet</b>	<b>2</b>
1.1 Répartition des tâches . . . . .	2
1.2 Architecture du programme . . . . .	3
1.3 Diagramme des packages . . . . .	4
<b>2 Elements techniques</b>	<b>8</b>
2.1 Système de sauvegarde . . . . .	8
2.2 Système de Création d'un Livre . . . . .	9
<b>3 Conclusion</b>	<b>10</b>
3.1 Objectifs remplis? . . . . .	10
3.2 Améliorations possibles . . . . .	10
3.3 Leçons retenues . . . . .	10

# Introduction

La programmation informatique notamment celle Orientée Objet a une place très importante au sein du cursus informatique. C'est pourquoi pour mettre en pratique nos connaissances et nous développer en Java, l'unité d'enseignement Conception Logicielle nous propose divers projets sur lesquels travailler.

Par groupes d'au plus 4 personnes, il nous a été demandé de réaliser une application parmi plusieurs proposées, notre choix s'est porté sur *"l'éditeur de Livre dont vous êtes le héros"* dont l'énoncé est le suivant :

« : Les livres dont vous êtes le héros (ou LDVEH) sont des jeux de rôles en solitaire dont la narration est décomposées en paragraphes, dispersés dans le livre. Des liens, en fonction des choix du lecteur, permettent d'aller d'un paragraphe à l'autre. Dans un premier temps, il s'agira de développer un éditeur de texte qui maintienne le graphe du jeu et qui puisse réorganiser aléatoirement le livre. L'affichage de la ou les solutions (paragraphe devant être traversés pour gagner) devront être calculées ainsi que la difficulté du jeu (proportions de solutions au regard des chemins possibles). Dans un second temps, il s'agira d'ajouter un système de rencontres et de combat, ainsi que la gestion d'objets ou d'indices qui peuvent être nécessaire pour gagner. Enfin, un mode lecture permettant de jouer au LDVEH pourra être implanté.»

La réalisation de ce projet se base principalement sur l'éditeur de Texte qui permettra de :

- Construire un Livre (Création des paragraphes [noeuds] <sup>1</sup> qui constituent un Livre).
- Jouer à un Livre ( Jouer une histoire <sup>2</sup> déjà créée ).

---

1. Tout au long de ce rapport, on utilisera le mot noeud pour désigner un paragraphe et vis versa.

2. On utilisera le mot histoire pour désigner un livre et vis versa.

# 1 Organisation du projet

## 1.1 Répartition des tâches

Afin de produire le jeu dans la limite du temps donné, nous avons dû élaborer un plan de travail, ainsi que répartir les tâches. Nous avons donc décidé dans un premier temps , de nous séparer en binôme . Michel et Faical ont d'abord travaillé sur l'implémentation du graphe<sup>3</sup> et au même moment wakary et Matej essayaient de trouver une façon optimale de créer l'interface graphique de base (L'éditeur de Texte).

Malgré le plan de travail établi , la conception de ce dernier (l'interface graphique) a demandé bien plus de temps que prévu, notamment dû au fait que wakary et Matej devaient utiliser Java Swing<sup>4</sup>, qu'ils ne maîtrisaient pas encore en ce moment-là .Ainsi l'implémentation du système de graphe étant terminée, Michel et Faical ont pu commencer en parallèle des recherches sur la création de l'interface graphique mais aussi sur le système des sauvegardes des livres (histoires) qui interviendront après leurs créations. Dans un second temps, Michel et Faical se sont aussi occupés de l'implémentation du système de construction d'un Livre et du système pour Jouer un Livre .

---

3. un graphe permet de mettre en relation plusieurs noeuds .

4. bibliothèque graphique en Java

## 1.2 Architecture du programme

Concernant l'architecture du programme, nous avons opté pour 5 packages :

**graphs** : qui contient les classes relatives au système de graphe.

Le package graphs est composé de classes :

- **Noeud** : classe de base pour créer un Noeud (paragraphe dans un Livre). Elle contient des attributs : description (énoncé du texte contenu dans un paragraphe) , question et identifiant (id unique).
- **Réponse** : une réponse est constituée d'un texte ainsi qu'une référence vers le paragraphe (Noeud) de destination.

**createStory** : qui comme son nom l'indique permet de créer une histoire (Livre) sur une interface graphique

Ce package est composé de :

- **AddReponse** : cette classe permet d'ajouter une réponse à un Noeud .
- **EditReponse** : Elle permet de modifier une réponse , en changeant son noeud de destination ou son texte.
- **CreateBookPage** : La classe CreateBookPage contient toutes les fonctionnalités(les zones de texte, les boutons etc) qui permettent de créer une histoire.
- **ManageBook** : Cette classe s'occupe de la gestion de toutes les autres classes de ce package.

**playStory** : Il permet de jouer une histoire créée au préalable.

Le package playStory est composé de :

- **GamePage** : cette classe contient toutes les fonctionnalités qui permettent de jouer à une histoire.
- **playBookPage** : classe qui permet de choisir une histoire pour la jouer.

**welcome** : package principal qui permet l'exécution du programme.

Il est composé de :

- **StartPage** : classe de démarrage et page d'accueil du programme.
- **Main** : classe qui permet d'exécuter le programme .

**tools** : qui est constitué de toutes les classes considérées comme étant des outils importants pour le bon déroulement du programme.

Ce package est constitué de :

- **ImagePanel** : cette classe sert à l’affichage des images utilisées sur l’interface graphique.
- **SettingsPage** : classe qui permet le paramétrage de l’interface graphique (font Size ,font Color, fontType ).
- **XmlBackup** : classe qui s’occupe de la sauvegarde d’une histoire.

### 1.3 Diagramme des packages

Diagramme du Package graphs

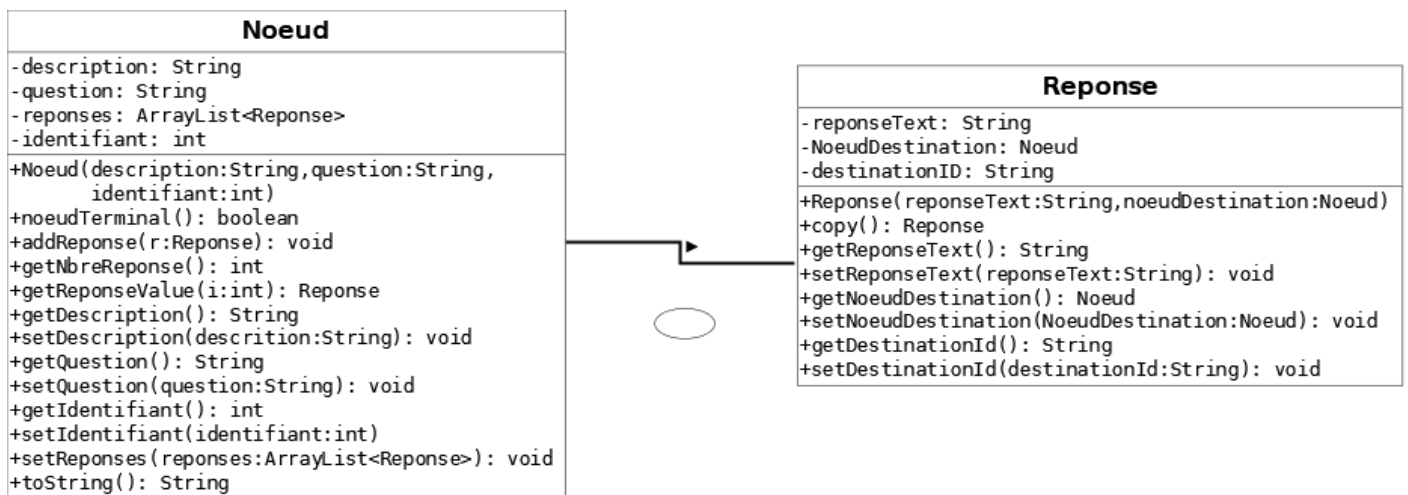


Diagramme du Package playStory

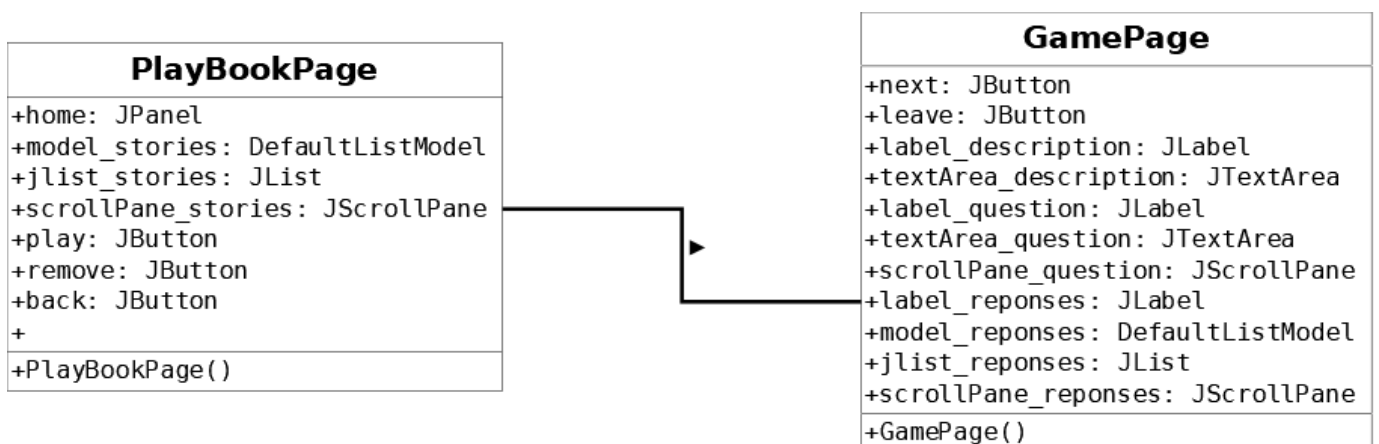
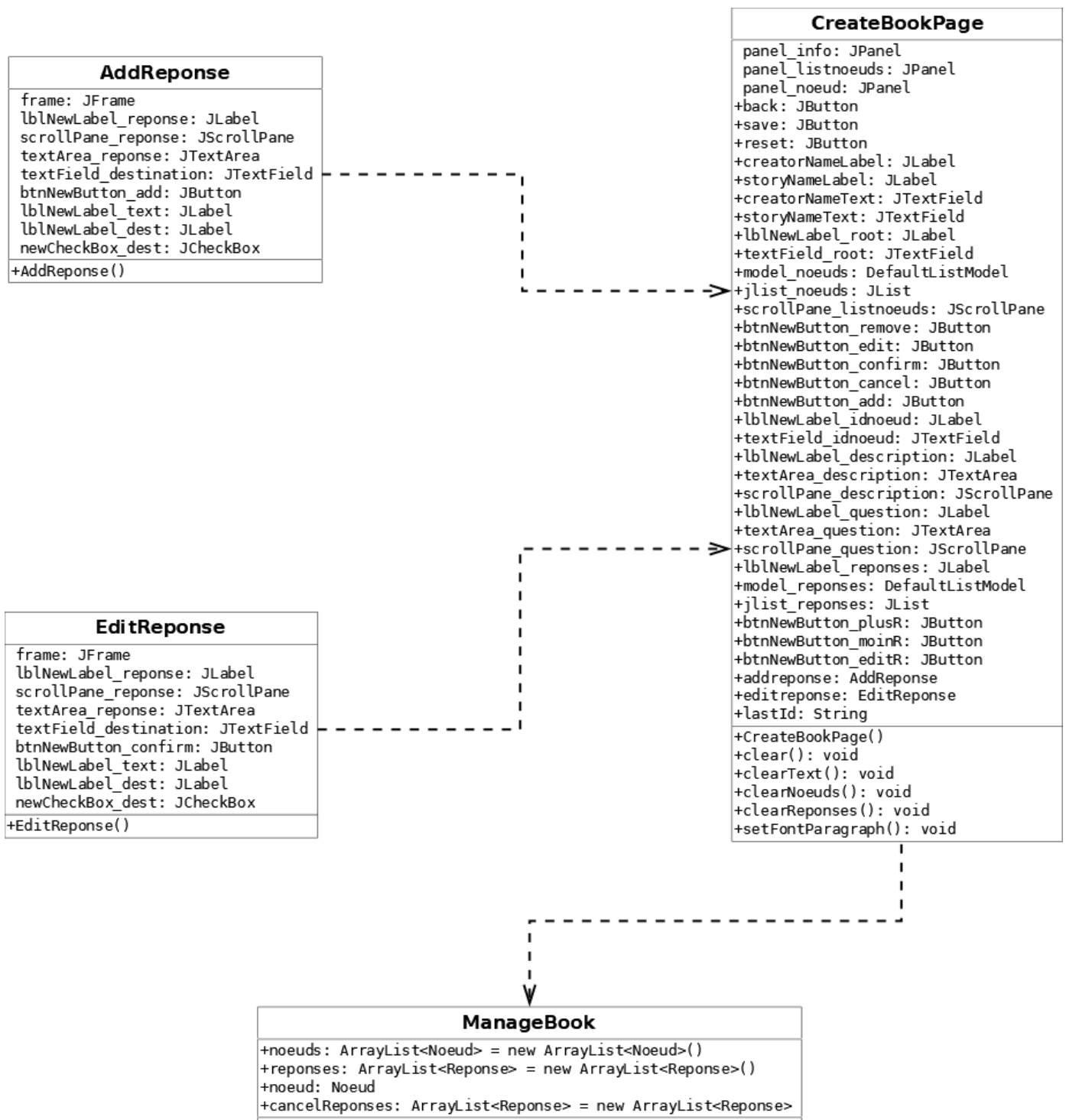
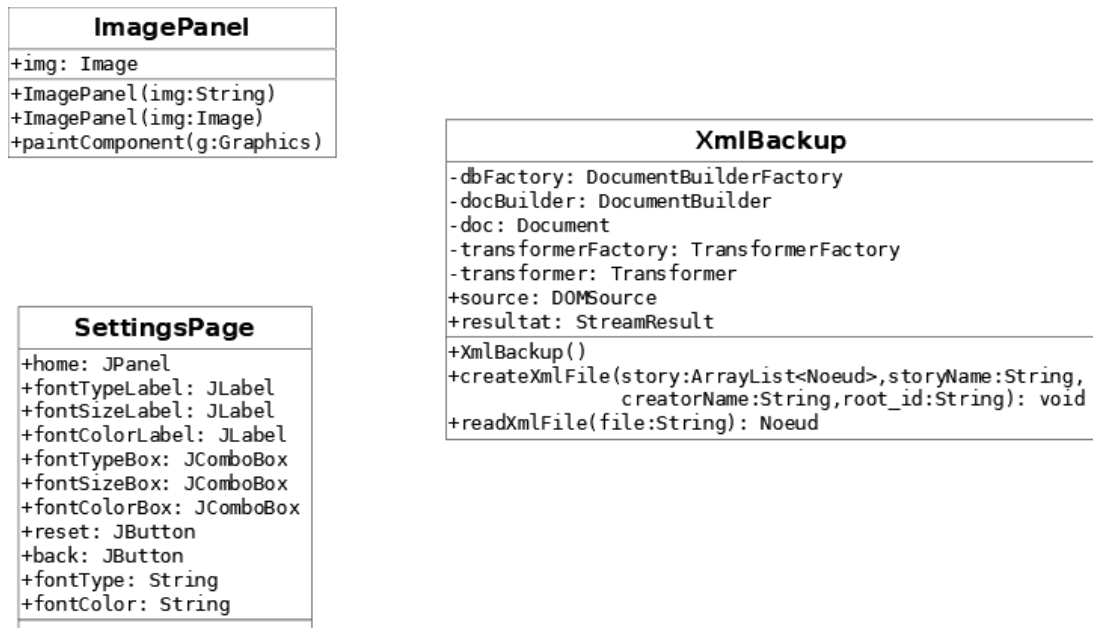


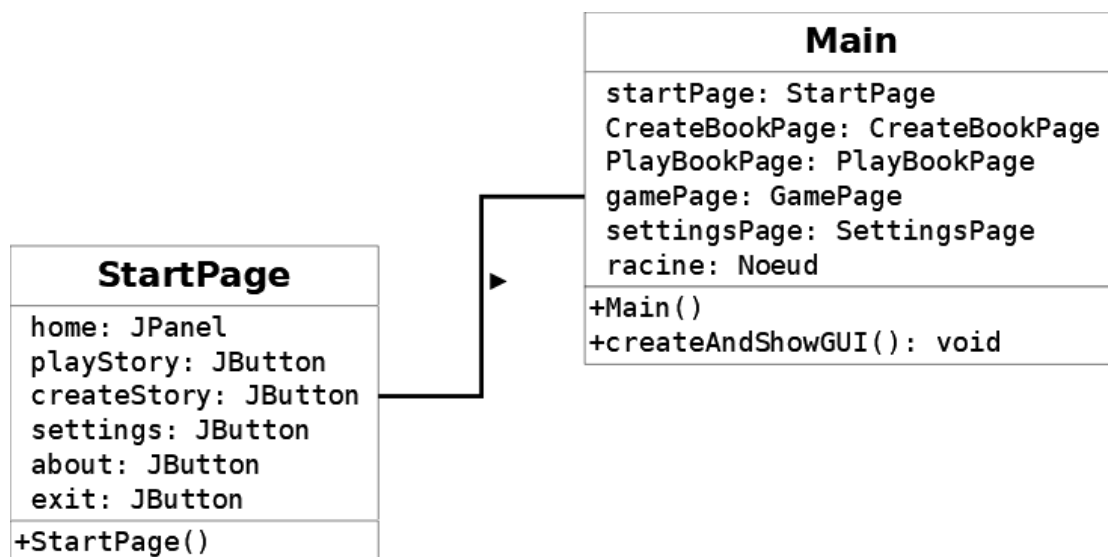
Diagramme du Package createStory



## Diagramme du Package tools

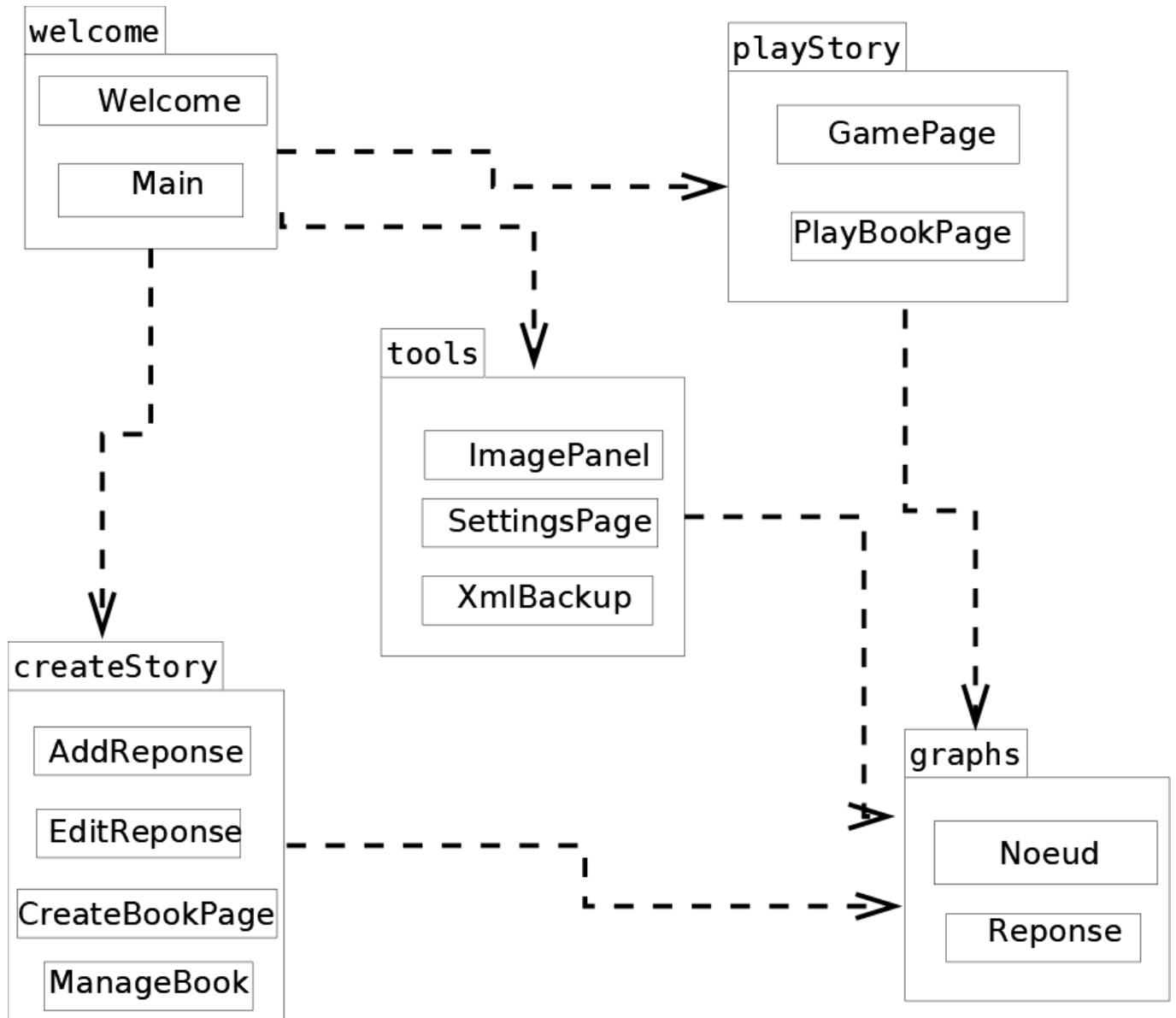


## Diagramme du Package welcome





## Diagramme des Packages



## 2 Elements techniques

### 2.1 Système de sauvegarde

L'implémentation de notre projet (Éditeur de livres dont vous êtes le héros) nécessitait un système de sauvegarde pour conserver les livres déjà créés, un peu comme une bibliothèque où l'on trouve un ensemble de livres. Nous avons réfléchi à mille et une façon d'implémenter ce système et nous avons finalement décidé de faire la sauvegarde en utilisant XML (L'extensible Markup Language)<sup>5</sup>. Ce système comprend deux parties :

- *la transformation d'une liste de noeuds en Chaîne de Caractères (String) et sa sauvegarde dans un fichier.*
- *la transformation d'un fichier (ensemble de Chaîne de Caractères) en liste des noeuds.*

#### 1. From Node to String

Pour pouvoir facilement transformer une liste de noeuds en chaîne de Caractères, nous avons créé une fonction qui prend en paramètre : la liste des noeuds, le titre du livre, l'auteur du livre et le noeud racine.<sup>6</sup>. En utilisant XML, nous avons stocké tous ces éléments dans des balises. Vu qu'il s'agit d'une liste de noeuds, nous avons utilisé une boucle For pour parcourir la liste entière et pour chaque noeud, on parcourt toutes ses réponses. À la fin de chaque transformation, un fichier contenant cette transformation est créé puis conservé.

#### 2. From String to Node

La transformation d'un fichier contenant des chaînes de Caractères en liste de noeuds, est une opération inverse à la première. Pour y arriver, nous avons créé une méthode qui prend en paramètre le fichier à transformer, dans le corps de cette méthode, nous avons initialisé une liste de noeud ainsi qu'un noeud racine, grâce à XML, nous avons pu récupérer la valeur du noeud racine dans le fichier et à partir du noeud racine, nous avons fait une opération récursive pour pouvoir récupérer la liste des noeuds ainsi que leurs réponses respectives.

---

5. Langage de balisage extensible

6. noeud considéré comme étant le début du livre.

## 2.2 Système de Création d'un Livre

Nous avons créé une interface graphique permettant de créer un Livre . Cette interface contient 3 zones principales :

- *Zone d'informations.*
- *Zone de création d'un paragraphe.*
- *Zone liste des noeuds.*

### 1. Zone d'informations

Cette zone permet de remplir les informations importantes lors de la création d'un livre, notamment le nom de l'auteur ainsi que le titre du livre.

### 2. Zone de création d'un paragraphe

Cette zone permet de créer un paragraphe.

Elle est constituée de :

- un champs de texte permettant de remplir l'identifiant du paragrahe.
- un champs de texte permettant de remplir la description du paragrahe.
- un champs de texte permettant de remplir la question du paragrahe.
- un champs de texte permettant de remplir les reponses du paragrahe

### 3. Zone liste de noeuds

Cette zone comprend la liste de tous les noeuds déjà créés, Elle comprend aussi les boutons ;

- remove : pour supprimer un noeud.
- edit : pour modifier un noeud

## **3 Conclusion**

### **3.1 Objectifs remplis ?**

Réaliser un travail de groupe , nécessite une organisation, une répartition des tâches, une gestion du temps. Nous avons eu un peu de mal à mettre en place toutes ces recommandations pour mener à bien notre projet à cause de la situation sanitaire actuelle et au fait que l'intégralité du projet a été réalisée à distance.

Néanmoins, nous avons presque tout implémenté en ce qui concerne les fonctions de base de notre programme. En commençant par implémenter le système de graphes, qui permet de mettre en relation l'ensemble des noeuds, ensuite par la création d'un éditeur de texte, qui permet de créer une histoire et enfin par la réalisation d'une interface graphique qui permet de jouer à une histoire.

### **3.2 Améliorations possibles**

Concernant les améliorations possibles, on aurait pu ajouter : l'affichage des solutions possibles (paragraphe devant être traversé pour gagner), un système de rencontres et de combat ainsi que la gestion d'objets ou d'indices qui peuvent être nécessaire pour gagner.

### **3.3 Leçons retenues**

Tout au long de notre projet, nous avons retenu plusieurs leçons : la nécessité d'une bonne communication dans un groupe de travail, la gestion du temps, l'importance d'une bonne répartition des tâches etc.

Qu'à cela ne tienne, la réalisation de ce projet a été une très belle expérience pour nous, ça nous a permis de mettre en pratique les notions de programmation orientée objet (Java) apprises en cours.