

Cahier des charges Royal

Steve Benedick
Jean Meyblun
Kévin Hagner

23 septembre 2011



Introduction

Notre projet a pour but la gestion informatique des emprunts et achats de bandes dessinées (bd) réalisé par un particulier. Pour cela nous allons partir d'une application existante, *Royal*¹, qui permet la saisie et l'enregistrement de bandes dessinées, et l'améliorer. En effet, Royal ne permet qu'une saisie manuelle de chaque ouvrage, et ne gère ni les bibliothèques d'emprunt ni les dates d'emprunt et de retour. Ainsi nous allons permettre à un utilisateur de simplifier sa gestions de bd, d'une part en améliorant l'application Royal et d'une autre part en développant une application mobile pour Android.

L'amélioration consistera en :

- L'automatisation de l'ajout d'une bd grâce à son code barre qui permettra la recherche et le remplissage automatique de l'ensemble des informations de l'ouvrage.
- L'ajout d'un système de bibliothèque, de durée d'emprunt, de date d'emprunt, tout cela pour pouvoir notifier à l'utilisateur s'il doit penser à rendre certains de ses ouvrages, d'où proviennent ses ouvrages et combien de temps il lui reste pour les lire.
- La synchronisation avec l'application Android.

L'application Android doit permettre :

- La capture des codes barres d'un ensemble de bd.
- L'envoi des codes barres à l'application améliorée de Royal qui grâce à la synchronisation sera capable d'ajouter automatiquement un lot de bd.

Le projet est à but personnel, il n'a comme unique objectif que de servir d'aide mémoire à un utilisateur. Il ne s'agit par exemple pas d'un système de gestion de bibliothèque ou de stocks des bd restantes dans une librairie. Aucune relation ne sera possible entre différents utilisateurs, ni de mises en commun des œuvres lues (statistiques des livres les plus lus par la « communauté », notation, commentaires...).

1. Site officiel du projet Royal : www.royal-project.org

Table des matières

I	Définition des fonctionnalités	5
1	Améliorations de priorité 1	5
1.1	Relatives au client PC	5
1.2	Relative au client Androïd	5
2	Améliorations de priorité 2	6
2.1	Relative au client Androïd	6
3	Améliorations de priorité 3	6
3.1	Relatives au client PC	6
3.2	Relatives au client Androïd	6
4	Améliorations de priorité 4	6
4.1	Relatives au client PC	6
4.2	Relatives au client Androïd	7
5	Utilisation type	8
5.1	Récupération des ISBN	8
5.1.1	Récupération par lot	9
5.1.2	Scann d'un livre unique	9
5.2	Utilisation de l'application sur ordinateur	9
5.3	Autres fonctionnalités	11
5.3.1	Gestion des dates de retour	11
5.3.2	Modification des informations d'un livre	11
II	Analyse de l'existant	12
6	Fonctionnement de l'application Royal	12
6.1	Introduction	12
6.2	Aspect fonctionnel de Royal	12
6.3	Aspect technique de Royal	12
6.4	Analyse de la base de données déjà existante	12
III	Planning prévisionnel	13
7	Étapes du projets	13
8	Éstimation du temps nécessaire	13
9	Planification des délais	14
IV	Définition des contraintes	15
10	Contraintes de temps	15
11	Contraintes de qualités	15
11.1	Charte ergonomique	15
11.2	Norme d'écriture	15
12	Contraintes matérielles et technologiques	15
12.1	Adaptation aux outils utilisés dans Royal	15
12.2	Compatibilité du client lourd	15
12.3	Compatibilité du client mobile	15

13 Contraintes de realisation **15**
13.1 Designation des acteurs 15
13.2 Cycle de vie 16

Première partie

Définition des fonctionnalités

Cette partie liste toutes les fonctionnalités attendues dans le programme. Celle - ci sont classées par priorités.

Priorité 1 : Priorité la plus élevée. Il s'agit des premières fonctionnalités à implémenter.

Priorité 2 : Élément nécessaire. Il s'agit de l'implémentation réelle des fonctionnalités principales du programme mais nécessitant la mise en place de structures préalables (Les éléments de priorité 1).

Priorité 3 : Éléments secondaires, ceux - ci seront effectués dans un second temps. Il s'agit d'améliorations au programme de base, mais pas de fonctionnalités nécessaires.

Priorité 4 : Éléments de « luxe ». Ces fonctions ne sont pas demandées mais rajoutent néanmoins une plus value au programme. Elles seront implémentées uniquement si la vitesse d'avancée dans le projet nous le permet.

1 Améliorations de priorité 1

1.1 Relatives au client PC

Vérification du code barre Après l'entrée du code barre par l'utilisateur le programme devra exécuter une vérification de ce celui - ci pour voir si il correspond bien à quelque chose se référant à un livre ou non. Cette vérification sera effectuée sur la syntaxe de l'expression : si le code barre commence par 978 ou 979 et s'il est bien composé de 13 chiffres, il sera considéré comme valide.

Recherche sur l'API Google book Le programme doit être capable de récupérer des informations sur les différents livres. Afin de récupérer ces informations nous utiliserons l'API Google Book car elle offre un grand nombre d'informations via l'ISBN du livre.

Complétion automatique des champs Après la récupération de ces informations le programme auto - complètera tout les champs de caractéristiques de l'ouvrage dont la réponse a été trouvée. Si l'utilisateur souhaite néanmoins utiliser une autre information que celle fournie par Google Books, il pourra toujours modifier la valeur écrite dans les champs automatiquement.

Ajouts des champs manquants Royal ayant initialement été prévu pour la gestion des collections personnels il nous faudra ajouter des champs tels que le lieu de l'emprunt, la date de l'emprunt ainsi que la date de retours du livre afin de permettre la gestion de livres empruntés.

Connexion à une adresse Gmail Le client qui gèrera la capture du code barre étant sous Android, la communication entre les deux programmes s'effectuant par mail, la plus simple des méthodes sera de passer par une boîte Gmail.

Récupération des mails Cette fonction de lecture des mails par le client lourd nécessitera une identification des mails ayant été envoyés par le client Android. Cette identification se fera à l'aide d'une syntaxe spécifique du sujet du mail. Une fois le mail identifié le logiciel récupérera les ISBN présents dans le message puis le supprimera.

1.2 Relative au client Android

Mise en place de l'interface graphique Une interface graphique proposant un menu donnant accès aux différentes fonctions de l'application du client Android devra être créée. Cette interface sera modifiée tout au long de la création du client Android afin de permettre l'accès aux nouvelles fonctions. Elle se devra d'être élégante et ergonomique.

Récupération et traitement du code barre La librairie open-source : ZXing² permet le traitement de codes barre en Java. C'est à dire qu'à l'aide d'une simple photo, elle est capable de repérer le code barre présent sur celle ci et de retourner sa valeur.

Envois du mail contenant l'ISBN via Gmail Afin de faire parvenir le code ISBN au client lourd l'application Android devra, après le décodage du code barre, envoyer l'ISBN correspondant par mail. On utilisera tout d'abord le compte Gmail, car tous les utilisateurs Android en possèdent un. Le mail aura une syntaxe spéciale dans son sujet afin d'être distingué par le client lourd des autres mails.

2. Site officiel de ZXing : <http://code.google.com/p/zxing/>

2 Améliorations de priorité 2

2.1 Relative au client Androïd

Vérification de la validité du code barre récupéré L'application vérifiera que le code barre récupéré est bien associé à un livre. Si le code barre ne correspond pas à un livre (Si il ne commence ni par 978, ni par 979), ou si le code barre n'a pas été reconnu sur la photo, l'utilisateur en sera averti via un message d'erreur.

Connexion à l'API Google Books depuis le client Androïd Cette connexion a pour but la permission de confirmer, depuis le client Androïd que celui-ci est répertorié dans la base de données Google. Si jamais il n'y est pas, l'utilisateur pourra directement écrire le nom du livre depuis son téléphone, et ainsi, n'aura pas besoin de reconnaître l'ouvrage ultérieurement par son ISBN.

3 Améliorations de priorité 3

3.1 Relatives au client PC

Connexion à une autre adresse mail Cette fonction aura pour but de permettre à l'utilisateur de choisir entre l'adresse par défaut (l'adresse Gmail liée au compte Androïd) et une autre adresse que l'utilisateur choisirait lui-même. Pour permettre de choisir d'autres adresses mails, le logiciel devra gérer plusieurs types de messagerie. Il faudra donc être capable de régler « manuellement » les paramètres de connexion tels que les serveurs IMAP / POP et SMTP à interroger.

Stocker les lieux d'emprunts dans une base de données Afin d'éviter à l'utilisateur de devoir écrire pour chacun des livres empruntés le nom de la bibliothèque dans laquelle l'emprunt a été effectué, une base de données stockant des lieux d'emprunts sera créée. L'utilisateur n'aura plus qu'à sélectionner son lieu d'emprunts dans une liste. Si le lieu d'emprunt n'existe pas dans la base de données l'utilisateur pourra le rajouter.

Gestion des ISBN indisponible sur Google Books Pour les ouvrages dont l'ISBN n'est associé à aucune information sur Google Books le logiciel affichera un message informant l'utilisateur que l'ajout automatique est impossible pour ce livre afin que l'utilisateur puisse les entrer manuellement.

Ajout d'un système de rappel de date de retour Le logiciel devra émettre des alertes à l'utilisateur quand la date de fin d'emprunt d'un livre approche. Cette fonction devra pouvoir être désactivée par l'utilisateur s'il ne l'estime pas nécessaire.

Conversion d'un ISBN 13 ou code barre Un champ sera ajouté afin que l'utilisateur puisse entrer le code barre de son livre ou l'ISBN à 13 chiffres dans les informations sur une bd. Il sera converti en ISBN à 10 chiffres. C'est cette information qui sera sauvegardée dans la base de données du logiciel et utilisé.

3.2 Relatives au client Androïd

Vérification des éléments disponible via l'ISBN L'application vérifiera sur internet que le code ISBN récupéré est bien associé à un livre. Si aucun livre n'est trouvé, l'utilisateur sera averti que les informations sur le livre ne sont pas disponibles. Un mail sera quand même envoyé au client lourd afin de rappeler à l'utilisateur qu'il doit rentrer les informations manuellement.

Envoi du mail contenant l'ISBN via une autre adresse mail Cette fonction aura pour but de permettre à l'utilisateur de choisir entre l'adresse par défaut (l'adresse Gmail liée au compte Androïd) et une autre, qu'il aura choisie de ses propres soins pour l'envoi des ISBN au client lourd.

4 Améliorations de priorité 4

4.1 Relatives au client PC

Tri des bandes dessinées L'affichage des bandes dessinées pourra être géré en fonction de la série, de l'auteur, de la date de parution ou de la date de lecture. L'affichage en fonction de l'auteur ou de la collection sont déjà des fonctionnalités présentes sur l'application Royal mais elles sont limitées et proposent aussi un affichage en fonction du type alors que le type de l'ouvrage ne fait pas partie des informations que l'on peut saisir. Cette révision de l'affichage aura pour principal but de faciliter la recherche de l'utilisateur parmi ses bandes dessinées.

Amélioration de la recherche d'images La fonction de recherche d'images implantée dans Royal ne propose pas toujours une image adaptée à notre album c'est pourquoi nous allons attribuer automatiquement l'image de la couverture de la bande dessinée en fonction de celle présente sur Google Book. Cette image pourra toujours être changée manuellement par une image personnelle comme on pouvait déjà le faire sur Royal.

Synchronisation des ISBN présentes dans la base de données Afin d'améliorer la fonction du client Androïd vérifiant les ISBN déjà existant, une fonction de synchronisation sera implémentée afin de lui faire parvenir les ISBN des bandes dessinées déjà présentes dans le client lourd susceptibles de manquer (Si le client lourd est utilisé depuis avant la création du client Androïd, ou si l'utilisateur change de téléphone par exemple). Cette synchronisation sera faite par email ou usb (peut être par bluetooth ou wifi).

Reception de l'ISBN par un autre moyen Cette fonction permet à l'utilisateur de recevoir les codes ISBN par un autre moyen que l'envoi d'un mail. (Solutions à étudier : usb, bluetooth, serveur, etc.)

4.2 Relatives au client Androïd

Gestion des ISBN des bandes dessinées déjà lues Tous les ISBN des bandes dessinées scannées seront conservés dans un fichier sous l'application Androïd afin de garder un historique des bandes dessinées déjà lues. Si l'ISBN a déjà été scanné précédemment, un message d'information sera envoyé au client. Toutefois il lui sera demandé s'il veut quand même renvoyer le code ISBN au cas où le livre aurait été supprimé sur le client lourd.

Synchronisation de l'historique des ISBN des livres déjà lue Une synchronisation des ISBN présents sur le client lourd, avec le client Androïd sera possible. L'historique présent dans l'application Androïd sera effacé et remplacé par ceux importés depuis le client lourd.

Envois de l'ISBN par un autre moyen Cette fonction permet d'envoyer les ISBN des livres scannés au client lourd par un autre moyen que l'envoi d'un mail. (Solution à étudier : usb, bluetooth, serveur, etc.)

5 Utilisation type

Dans cette partie sera évoqué un exemple d'utilisation type du programme. Les actions de l'utilisateur seront décrites, et les processus techniques (en italique) que l'action engendre également.

Notez que cette utilisation type se base sur des fonctionnalités de priorité 3 et 4. Il est donc probable que l'application ne permette finalement pas toutes ces fonctionnalités.

5.1 Récupération des ISBN

À la première utilisation du programme, l'adresse email à utiliser pour la synchronisation est demandée à l'utilisateur. *Au lancement du programme, celui-ci vérifiera l'existence d'un fichier de configuration contenant les caractéristiques de connexion à la boîte email. Si ce fichier n'existe pas, il sera directement proposé au client de renseigner ces informations.*

L'utilisateur pourra aussi choisir d'utiliser l'adresse email liée au compte Android par défaut.



FIGURE 1 – Premier démarrage de l'application

Ensuite, c'est une fenêtre à quatre boutons qui se présente à l'utilisateur. Sur le premier est indiqué « Scanner un livre », sur le second, « Scanner une liste de livre », « Configuration de l'adresse E-Mail » et « À propos de Royal_ ».

Le bouton de configuration de l'adresse E-Mail ouvre la même page que précédemment. L'à propos ne consiste qu'en l'affichage d'une simple fenêtre contenant des informations sur les auteurs.



FIGURE 2 – Menu de l'application Royal_Scanner



FIGURE 3 – Capture de l'ISBN

5.1.1 Récupération par lot

Si l'utilisateur se tourne vers la capture d'une liste, un message apparaît, demandant de viser le code barre du livre avec la caméra de l'appareil. Il faut appuyer sur « ok », puis, l'appareil photo s'allume. *C'est en fait à ce moment que Royal fait appel à la librairie ZXing afin de récupérer le code barre du livre, et d'en déduire l'ISBN.* L'utilisateur prend donc en photo le code barre du livre. Un message s'affiche à présent à l'écran, avec marqué dessus « Vérification du livre » avec une barre de progression en dessous. *Le programme tente en fait d'interroger Google Book pour voir si des informations sur l'ouvrage sont disponibles.*

Une fois la barre remplie, une autre phrase s'affiche : « Ouvrage identifié : < Nom de l'ouvrage > » la remplacera. Ensuite un autre apparaît demandant si c'est la fin du lot, ou si, au contraire, d'autres livres restent à scanner. *Les ISBN, avec le nom de la bd sont écrits les un à la suite de l'autre dans un fichier.*

L'utilisateur opte pour le second cas, afin d'enregistrer mes trois autres bd dans le logiciel. Malheureusement, arrivé au dernier, le portable lui indique que l'ouvrage est introuvable et lui demande donc de rentrer à la main le titre du livre. *Le nom du livre est contenu dans le client Androïd, en plus du numéro d'ISBN pour que, une fois sur le client lourd, l'utilisateur sache encore de quel livre il s'agit sans pour autant avoir besoin de comparer l'ISBN.*

Une fois toutes les bd de scannés, l'utilisateur choisi de terminer l'enregistrement. *À ce moment là, la date du scann est ajoutée au début du fichier contenant les ISBN.*

Peu de temps après, un nouveau message apparaît : « Un e-mail de synchronisation a bien été envoyé sur votre boîte mail, veuillez effectuer un import depuis votre ordinateur ». *Le fichier précédemment créé avec la date du scann, le numéro d'ISBN et le titre du livre est envoyé par e-mail à l'adresse définie pour l'exportation. Le titre du mail correspond à une chaîne identifiable par le client lourd : **Royal_***

5.1.2 Scann d'un livre unique

Nous choisirons cette fois le mode de scann simple, adapté pour un seul livre. L'appareil photo se rallume, on prend la photo du code barre, puis le nom de l'ouvrage réapparaît sur l'écran. On confirme, et l'exportation s'effectue apparemment avec succès. *D'un point de vue interne au programme, le comportement est similaire entre le mode de scann par lot et celui par unité, la récursivité en moins.*

De retour devant l'ordinateur, nous ré effectuons une importation. Encore une fois, une fenêtre s'ouvre, avec les informations auto - complétées de l'ouvrage, il ne reste plus qu'à choisir la bibliothèque dans la quelle c'est effectué l'emprunt, à confirmer, et le livre s'ajoute dans la liste principale!

5.2 Utilisation de l'application sur ordinateur

Au premier démarrage du client PC, l'utilisateur se retrouve de nouveau face à une fenêtre me demandant de choisir quelle adresse e-mail il allait utiliser pour la synchronisation avec le « client Androïd ». C'est la même adresse que toute à l'heure qui doit être renseignée ici. *À l'exécution du programme, le fichier de sauvegarde de configuration **resources/royal.properties** est lu. Si les champs **email_address**, **email_port** et **email_protocol** sont vides, on propose à l'utilisateur de renseigner ces informations, soit de façon manuelle, soit en choisissant simplement la configuration de base liée au compte Androïd du téléphone (l'adresse GMail de l'utilisateur, donc).*

Ensuite nous arrivons sur la page centrale du programme. Celle - ci ressemble à une liste sur la quelle tous livres de l'utilisateur sont répertoriés. Elle est vide pour le moment, et les boutons de gestion sont bien visible

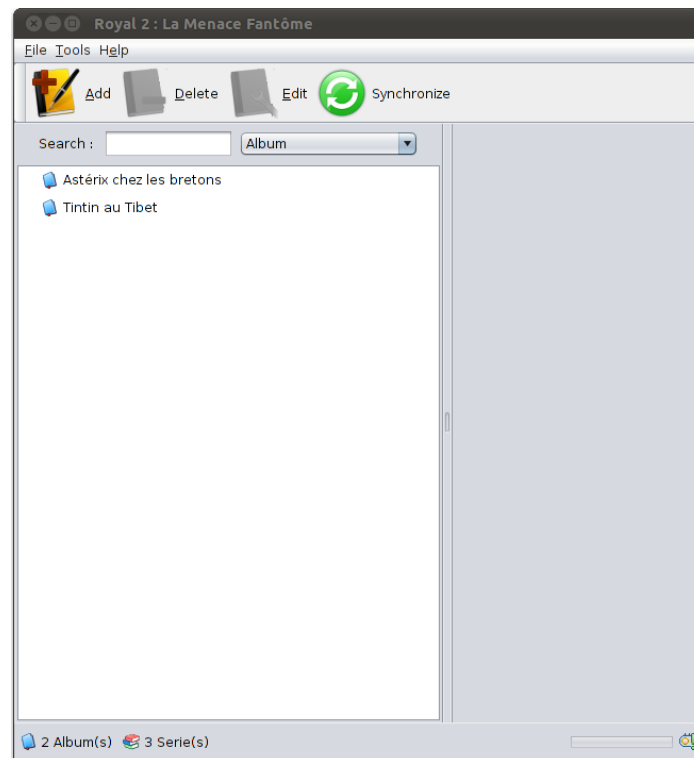


FIGURE 4 – Page principale de l'application

sur la partie haute de la fenêtre. On y retrouve un bouton pour ajouter un ouvrage, pour en supprimer un, pour l'édition, et enfin pour l'importation.

Après un clique sur le bouton d'importation, une autre fenêtre apparaît, indiquant d'attendre pendant l'opération. On peut voir en dessous une barre de progression, qui se remplit, au fur et à mesure de l'avancée des étapes : « Connexion au serveur mail », puis « Identification mail », après « Recherche des fichiers », et enfin « Téléchargement des ISBN ». *L'application va en effet se connecter sur la boîte email de l'utilisateur et faire une recherche sur le sujet des messages contenus dans la boîte de réception. Si un sujet est exactement composé de l'appellation « **Royal** », alors le contenu du message est téléchargé, et la recherche se termine. Il est à noter que chaque importation n'importera qu'un mail à la fois. Le contenu du mail est ensuite traité : dans un premier temps la date de scann du lieux est conservé, puis chaque couple d'information (ISBN / Titre) pour chaque livre.*

Après ce temps d'attente, la fenêtre disparaît, et une nouvelle affiche le récapitulatif des livres importés. Le numéro ISBN est marqué dans une colonne, à côté, est écrit « Téléchargement des informations », au fur et à mesure que le temps passe, les informations s'affichent pour chaque livre, sauf celui pour le quel un message d'erreur a été émis lors de la prise de photo de l'ISBN. L'utilisateur pourra toujours rentrer lui même les informations dans les champs laissés vide par défaut. *C'est à ce moment que la recherche des informations sur les livres seront effectuées. L'affichage sera évolutif, et l'API Google Dock sera interrogée pour chaque livre séparément. Un message d'attente, ou de recherche en cours sera écrit pour les livres en attente de traitement. Ceux dont une réponse de l'API sera déjà arrivé posséderont des champs éditables contenant par défaut les informations retournées, ou rien, si l'ISBN était introuvable.*

Une fois satisfait de la présentation de livre, il faut cliquer sur « Confirmer les informations des livres ». La prochaine fenêtre me demande si j'ai acheté ces ouvrages ou si je les ai empruntés dans une bibliothèque. On indiquera l'emprunt.

Le programme demandera ensuite dans quelle bibliothèque l'emprunt aura été effectué. Vu que l'installation est toute fraîche, aucune bibliothèque n'est existante pour le moment, la liste est vide. Un bouton d'ajout sera à ce moment disponible. Un clique dessus produira l'ouverture d'une nouvelle fenêtre, demandant de renseigner le nom de la bibliothèque, son numéro de téléphone, la durée maximale de l'emprunt, et ses horaires d'ouverture. *Les bibliothèques sont présentes dans une table à part de la base de données. À ce niveau, nous pouvons choisir de référer le lot à une bibliothèque déjà existante, ou à en créer une nouvelle (dans ce cas, une entrée sera évidemment créer dans la table des bibliothèques). Notons que la notion de lot est quelque chose d'abstrait, d'un point de vue technique, aucune relation n'existe entre les livres. Il ne s'agit uniquement que d'un procédé évitant à l'utilisateur de rentrer des informations redondante pour chacun des livres. Ainsi, un identifiant vers*

la bibliothèque, et la date d'emprunt est conservé pour chaque livre.

De retour à la page principale de Royal, la liste des ouvrages que j'ai importé est à présent marquée dans la liste. Nous pouvons lire à droite quelques informations sur l'ouvrage sélectionné, dont la date de retour et la bibliothèque concernée.

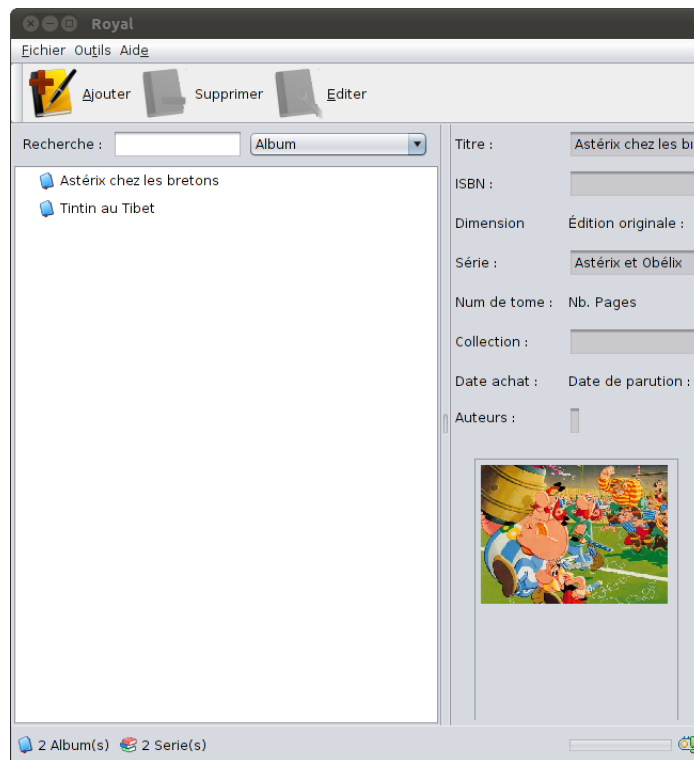


FIGURE 5 – Page principale du programme non vide

5.3 Autres fonctionnalités

5.3.1 Gestion des dates de retour

Les dates de retour des ouvrages sont visibles dans les caractéristiques des livres. Un compte à rebours des jours restant avant la date critique sera aussi disponible dans la liste des livres. Une alerte sera paramétrable dans les options permettant de rappeler à l'utilisateur les livres dont la date critique est proche, au démarrage de l'application. Dans les informations sur l'œuvre, le client pourra, évidemment indiquer avoir rendu le livre. Dans ce cas, plus aucune alerte ne sera émis sur le livre, ni d'affichage des jours restant.

5.3.2 Modification des informations d'un livre

Il suffira de cliquer sur le nom du livre qu'on comptera modifier pour voir un aperçu des informations contenues à son égard. À ce moment, un bouton « Modifier » apparait en haut de la fenêtre. Un clique sur celui - ci produira l'ouverture d'une nouvelle fenêtre donnant à l'utilisateur la possibilité de modifier toutes les informations.

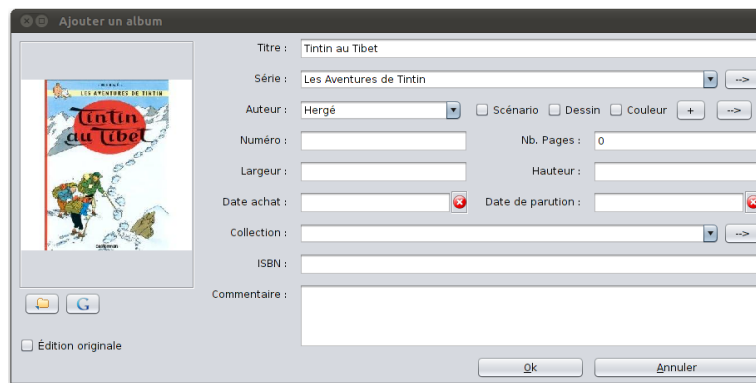


FIGURE 6 – Fenêtre d’édition des informations d’un livre

Deuxième partie

Analyse de l’existant

6 Fonctionnement de l’application Royal

Comme nous pouvons l’observer le MCD existant n’est pas parfaitement optimisé et certaines des informations ne sont jamais utilisé dans l’aplication Royal.

6.1 Introduction

Nous n’avions pas d’existant imposé à proprement parlé, cependant nous avons été guidé vers Royal, un projet existant réalisé par des étudiants du département informatique l’an passé. Il apparait être une des solutions actuelles la plus en accord avec la demande client à savoir de manière générale, « la facilitation de gestion d’une bédéthèque ».

De ce fait, cet article traitera du fonctionnement de Royal. Royal est une application libre³ issus d’une amélioration de Birdy, une autre application libre de gestion de livre.

6.2 Aspect fonctionnel de Royal

Royal permet la saisie d’informations sur une bande dessinée (BD), tel que son titre, son auteur, son image de couverture (recherchée sur internet), sa date de parution, etc.. Il permet d’obtenir une liste de l’ensemble des BDs, que l’on peut trier selon divers critères et réorganiser selon des collections de BDs. Royal permet l’enregistrement de BDs, mais aussi d’informations concernant les auteurs ou encore les collections. L’on peut aussi importer un ensemble de BD à partir d’une base de donnée existante. De plus Royal intègre un système multilangue concernant son interface utilisateur, ainsi que d’une bibliothèque d’aide d’utilisation.

6.3 Aspect technique de Royal

Royal est développé en Java et utilise la bibliothèque *Swing* pour son environnement graphique. Le projet à été développé sous Linux, mais reste néanmoins exécutable sous d’autres systèmes d’exploitations comme Windows 7 par exemple, avec plus ou moins de succès. L’ensemble des données stockées est fait via *Hibernate*, c’est un framework gérant la persistance des objets en base de données. C’est un outil lourd et complexe mais très puissant que nous allons réutiliser étant donné sa mise en place existante sous Royal qui permet un transfert simplifié entre objets java et base de données.

6.4 Analyse de la base de données déjà existante

3. Selon les termes de la GNU Lesser General Public License

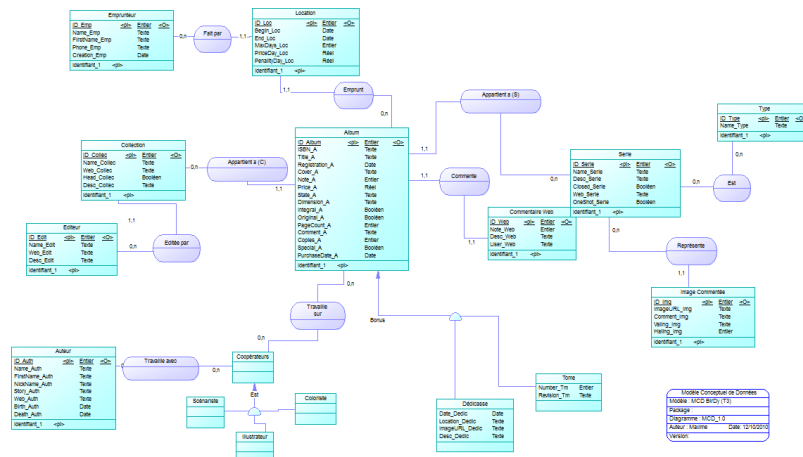


FIGURE 7 – MCD Repris de la documentation déjà existante de Royal

Troisième partie

Planning prévisionnel

7 Étapes du projets

Notre projets comporte 4 étapes principales. Chacune de ces étapes aura une date de remise définie des le debut du projet.

La rédaction du cahier des charges Dans cette étape nous avons eu principalement besoin de rédiger la demande du client. Nous avons aussi eu à étudier le fonctionnement du logiciel Royal que l'on a choisi d'améliorer afin d'arriver au but visé. Les contraintes liées à l'amélioration de ce logiciels nous permettent de définir la faisabilité des différentes fonctions.

La rédaction du dossier d'analyse Dans cette étape nous aurons à approfondir l'analyse ainsi que la conception des différentes fonctions ainsi que le fonctionnement du logiciel Royal. Nous définirons aussi les différents outils utilisé afin de développer le projet tels que Eclipse⁴ pour le developpement JAVA ou encore PowerAMC⁵ pour la réalisation des MCD.

La réalisation d'un premier prototype Afin de rendre un prototype, le 5 Décembre 2011, répondant aux principales demandes de l'utilisateur nous développerons les fonctions de priorités 1 et 2. Pour permettre l'implémentation de certaines de ces fonctions, nous aurons à modifier la base de données en suivant le models modifié dans l'étape précédente.

La finalisation du projet Cette étape sera la dernière de notre projet et devra être effectuée avant le 19 Janvier 2012 (date de livraison de notre projet). Pour cette étape nous développerons les fonctions de priorités inférieur afin de répondre à des besoin moins important.

8 Estimation du temps nécessaire

Pour la réalisation des estimations de temps nécessaire à la réussite de ce projet, nous nous sommes basé sur deux méthodes.

La première méthode choisie est l'*estimation par analogie*. Elle correspond à se fier à la conception du cahier des charges pour l'estimation du temps que prendra l'intégralité du projet.

4. Site de l'IDE Eclipse : <http://www.eclipse.org/>

5. Site officiel de PowerAMC : <http://www.sybase.fr/products/modelingdevelopment/poweramc>

		Durée proportionnelle	Durée concrète
Opportunité	Étude préalable	10%	60h
Élaboration	Conseption de la solution détaillée	30%	180h
Construciton	Développement	50%	300h
Transition	Mise en œuvre	10%	60h

Avec cette méthode, nous avons une estimation de travail de $60 + 180 + 300 + 60 = 600$ heures.

9 Planification des délais

Pour la planification des délais, nous avons utilisé la méthode du *diagramme de GANTT*.

Répartition des taches Projet Royal_											
	10/10/11	17/10/11	24/10/11	31/10/11	07/11/11	14/11/11	21/11/11	28/11/11	05/12/11	12/12/11	Total
Kevin			3 hr	13 hr	6,5 hr	14 hr	12 hr	14 hr	6 hr		68,5 hr
Ajouts des champs manquants			3 hr	9 hr							12 hr
Stocker les lieux d'emprunts dans une base de données						14 hr	4 hr				18 hr
Gestion des ISBN indisponible sur Google Books				4 hr							4 hr
Ajout d'un système de rappel de date de retour							8 hr				8 hr
Tri des bandes dessinée								10 hr			10 hr
Synchronisation des ISBN présentes dans la base de données					6,5 hr						6,5 hr
Reception de l'ISBN par un autre moyen								4 hr	6 hr		10 hr
Steve	6 hr	14 hr	14 hr	10 hr	10 hr	14 hr	12 hr				80 hr
Mise en place de l'interface graphique	6 hr	8 hr									14 hr
Récupération et traitement du code barre		4 hr									4 hr
Envois du mail contenant l'ISBN via Gmail			4 hr	6 hr							10 hr
Vérification de la validité du code barre récupéré			4 hr								4 hr
Connexion à l'API Google Books depuis le client Android				4 hr	4 hr						8 hr
Envois du mail contenant l'ISBN via une autre adresse mail					6 hr						6 hr
Gestion des ISBN des bandes dessinées déjà lues						14 hr					14 hr
Synchronisation de l'historique des ISBN des livres déjà lue							12 hr				12 hr
Scan de plusieurs isbn		2 hr									2 hr
Configuration adresse d'envoi de l'isbn			6 hr								6 hr
Jean	6 hr	14 hr	14 hr	12 hr	7 hr	10 hr	6,5 hr				69,5 hr
Vérification du code barre	4 hr										4 hr
Recherche sur l'API Google book	2 hr	10 hr									12 hr
Ajout des informations dans la base de données		4 hr	11 hr								15 hr
Connexion à une adresse Gmai			3 hr	3 hr							6 hr
Récupération des mails				4 hr							4 hr
Connexion à l'API Google Books depuis le client Android					7 hr	1 hr					8 hr
Connexion à une autre adresse mail				4 hr							4 hr
Conversion d'un ISBN 13 ou code barre				1 hr							1 hr
Amélioration de la recherche d'images							6 hr				6 hr
Synchronisation des ISBN présentes dans la base de données						3 hr	0,5 hr				3,5 hr
Envois de l'ISBN par un autre moyen						6 hr					6 hr
Total	12 hr	28 hr	31 hr	35 hr	23,5 hr	38 hr	30,5 hr	14 hr	6 hr		218 hr

FIGURE 8 – Estimation de notre temps de travail

Quatrième partie

Définition des contraintes

10 Contraintes de temps

Pour notre projet nous avons un temps imparti avec au moins trois étapes donc au moins trois livrables. Ces remises ne nous permettent aucuns retard donc chaque étape du projet devra être terminée dans les temps impartis. Les autres contraintes de temps que nous pourrions rencontrer lors de ce projet sont les heures que l'on devra consacrer à nos études de façon extérieure au projet (par exemple pendant les partiels).

11 Contraintes de qualités

11.1 Charte ergonomique

Notre projet étant l'amélioration d'un programme déjà existant nous aurons à respecter la charte graphique déjà existante afin de concorder avec le reste du programme. De même, pour l'application Androïd, nous aurons à nous adapter à une utilisation tactile de l'application.

11.2 Norme d'écriture

Royal est un projet open - source. Nous nous devons de redistribuer le code source qu'on créera. Pour ce fait, des normes d'écritures doivent être mise en place afin de faciliter la compréhension de notre code par d'autres personnes susceptible de reprendre le projet. Nous devons également écrire les commentaires dans notre programme en anglais, comme c'est le cas dans Royal à présent.

12 Contraintes matérielles et technologiques

12.1 Adaptation aux outils utilisés dans Royal

Pour notre projet nous devons nous adapter aux outils déjà utilisés. Une bonne partie de notre travail consistera à comprendre le fonctionnement de ces outils.

12.2 Compatibilité du client lourd

Le client lourd devra pouvoir s'installer sous Linux et si possible sur Windows et MacOS. Pour permettre cette compatibilité nous devons veiller à fournir les différentes bibliothèques adaptées aux différents systèmes d'exploitation avec notre programme.

12.3 Compatibilité du client mobile

L'application mobile devra être compatible avec les téléphones Androïd à partir de la version 2.2. Nous devons aussi faire attention à ce que les différentes bibliothèques utilisées soient déjà installées sur le téléphone, ou dans le cas contraire les installer automatiquement.

13 Contraintes de réalisation

13.1 Designation des acteurs

Pour ce projet notre maître d'ouvrage est Maître Ogier, pour ce qui est des maître d'oeuvre notre équipe est composée de Hagner Kevin le chef de projet, Meyblum Jean ainsi que Benedick Steve. Afin pour faciliter la communication entre notre tuteur et nous, nous avons choisi d'organiser des réunions une à deux fois par semaine. De même Maître Ogier pourra suivre le développement de notre projet sur notre dépôt Git.⁶ Pour ce qui est de la communication entre les membres de notre trinôme nous avons créé un canal irc⁷ afin de pouvoir faciliter la discussion à plusieurs. De même nous pourrions travailler ensemble soit à l'IUT soit chez l'un des membres si besoin.

6. Page Github de notre dépôt Git : https://github.com/spydemon/Royal_

7. canal #royal_ sur le serveur irc.rezosup.org

13.2 Cycle de vie

Afin de développer parallèlement différents modules une fois la phase notre cahier des charge validé nous avons choisi un cycle de vie en W. En effet comme dans ce modèle l'intégration est déjà étudiée lors de la conception globale des fonctionnalités nous pouvons concevoir, développer et tester chaque fonctionnalité séparément. De même ce cycle de vie nous permet de développer notre projet en fonction des priorités de nos tâches ce qui nous permet de garantir au minimum le rendu d'un projet répondant aux demandes de base.