|  |
| --- |
|  |
| Gestion de Bibliothèque |
| Document d'architecture et de conception |
|  |
| **PORCHET Gaétan**  **MELLAH Ayoub**  **RESH Boris**  **SOULET Corentin**  **KERVEVAN Arthur** |
| **(13/03/2018)** |

|  |
| --- |
|  |

# Suivi des changements

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Numéro de version** | **Date**  (jj-mm-aaaa) | **Numéro de figure, table ou section** | **Type de changement** | **Description du changement** | **Originateur de la demande de changement** |
| **1** | 13/03/2018 |  | Version originale |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

Table des matières

[Suivi des changements 2](#_Toc507515622)

[1. Introduction 5](#_Toc507515623)

[1.1 Description du projet 5](#_Toc507515624)

[1.2 Buts de ce plan 5](#_Toc507515625)

[1.3 Définitions, abréviations et acronymes utilisés dans ce document 5](#_Toc507515626)

[1.4 Hypothèses associées à ce plan 5](#_Toc507515627)

[1.5 Références 5](#_Toc507515628)

[2. Exigences 6](#_Toc507515629)

[2.1 Exigences fonctionnelles 6](#_Toc507515630)

[2.2 Exigences non-fonctionnelles 6](#_Toc507515631)

[3. Contraintes de conception 7](#_Toc507515632)

[4. Modèle conceptuel de communication (MCC) 8](#_Toc507515633)

[Définition de l'organisation 8](#_Toc507515634)

[Diagramme de contexte 8](#_Toc507515635)

[Diagramme conceptuel de flux 8](#_Toc507515636)

[5. Modèle conceptuel des traitements 10](#_Toc507515637)

[Le modèle conceptuel des traitements 10](#_Toc507515638)

[Le concept d'événement 10](#_Toc507515639)

[Définition d'un processus 10](#_Toc507515640)

[Opération 10](#_Toc507515641)

[La synchronisation 10](#_Toc507515642)

[Construction du MCT 10](#_Toc507515643)

[6. Modèle conceptuel des données 12](#_Toc507515644)

[7. Interfaces 13](#_Toc507515645)

[7.1 Interfaces utilisateur 13](#_Toc507515646)

[7.2 Interfaces matérielles 13](#_Toc507515647)

[7.3 Interfaces logicielles 13](#_Toc507515648)

[7.4 Interfaces de communications 13](#_Toc507515649)

[8. Matrice de traçabilité 14](#_Toc507515650)

**1. Introduction**

## 1.1 Description du projet

L'objectif de ce projet est de présenter un modèle MCD (statique) fonctionnel pouvant être transformé en une application (dynamique). La demande émane d'une Bibliothèque qui souhaite pouvoir gérer ses stocks d'ouvrages, ses adhérents ainsi que les emprunts et les retours.

## 1.2 Buts de ce plan

Au travers de ce plan, vous pourrez retrouver l'ensemble des informations nécessaires à la compréhension et à l'utilisation de l'application. En premier lieu, nous expliciterons l'ensemble des contraintes et exigences inhérentes au projet, toutes les hypothèses formulées, ainsi que les choix effectués. Vous trouverez également plusieurs modèles (MCC, MCT, MCD) permettant de détailler notre conception, et mettre en avant les limites de ce projet. Enfin, nous présenterons les éléments qui composeront l'application.

## 1.3 Définitions, abréviations et acronymes utilisés dans ce document

|  |  |
| --- | --- |
| **Sigle** | **Description** |
| **MCT**  **MCC**  **MCD**  **HPT** | Modèle conceptuel des traitements  Modèle conceptuel de communication  Modèle conceptuel des données  Hypothèse |

## 1.4 Hypothèses associées à ce plan

HPT1 : Toutes les personnes enregistrées sont des adhérents qui ont accès à la bibliothèque

HPT2 : Les non adhérents n’ont pas accès à la bibliothèque.

HPT3 : Un livre est édité par un unique éditeur.

HPT4 : S’il y a différents exemplaires d’un même ouvrage, différents ISBN leur seront attribués.

HPT5 : La contrainte « Doit être à jour dans ses cotisations pour emprunter » sera vérifier de manière dynamique et n’est pas traduite dans le modèle statique (MCD).

HPT6 : Rien n’interdit un adhérent de continuer à faire des emprunts après d’avoir perdu ou détruit un livre.

HTP7 : Chaque livre est associé à une unique durée de prêt qui servira d’information lors d’un emprunt.

HTP8 : La durée du prêt est notée en jours.

HTP9 : La disponibilité d’un livre sera vérifiée dans le modèle dynamique et non pas le statique.

HTP10 : Le remboursement n’apparaitra que sur le modèle dynamique puisqu’il sera calculé en fonction de la date de retour effective.

## 1.5 Références

[Cette section doit énumérer tous les documents qui sont cités en référence dans votre plan. Spécifiez pour chaque document son titre, son ou ses auteur(s), la date ainsi que toute autre information pertinente.   
  
Par exemple, il peut s’agir de normes qui ont été utilisées lors de la rédaction de ce document, de livres que vous avez consultés afin de mieux connaître certaines technologies relatives à votre projet ou de sites web contenant de l’information sur UML.  
  
Identifiez et numérotez les références en utilisant le préfixe REF.

Exemples :

Ref-1 : SEL (version 1.0), projet XYZ, Acme Inc, Département du développement logiciel, décembre 2011

Ref-2 : Document d’architecture (version 1.5), projet XYZ, Acme Inc, Département du développement logiciel, janvier 2012]

# 2. Exigences

## 2.1 Exigences fonctionnelles

[Cette section doit énumérer toutes les exigences fonctionnelles liées à votre logiciel. Soyez précis. Généralement, ces exigences constituent les fonctions du logiciel (ce qu’il doit faire).

*Une exigence fonctionnelle est une exigence définissant une fonction du système à développer. Ce que le système doit faire.*

Dans le cas d’un logiciel effectuant la gestion des dossiers étudiants d’un, une exigence fonctionnelle pourrait être l’affichage d’un dossier étudiant ou bien le calcul de sa côte Z.

*Le système doit enregistrer la commande du client.*

*Chaque commande doit avoir un identifiant unique (ORDER\_ID)*  
  
Identifiez et numérotez les exigences fonctionnelles en utilisant le préfixe EF.]

## 2.2 Exigences non-fonctionnelles

[Cette section doit énumérer toutes les exigences non-fonctionnelles liées à votre logiciel. Soyez précis. Généralement, ces exigences constituent des propriétés vérifiables que votre logiciel doit posséder. Elles sont habituellement liées à l’usage, à l’efficacité, à la fiabilité, à la maintenance ou à la réutilisation.   
  
Dans le cas d’une application web effectuant la gestion de la paye des employés d’une firme, des exemples d’exigences non-fonctionnelles pourraient être que l’application puisse supporter 10 connexions simultanées ou que l’application soit compatible avec les navigateurs Internet Explorer et Firefox et Safari.

*Le système doit permettre à l’utilisateur d’accéder au solde de son compte en moins de 5 secondes.*

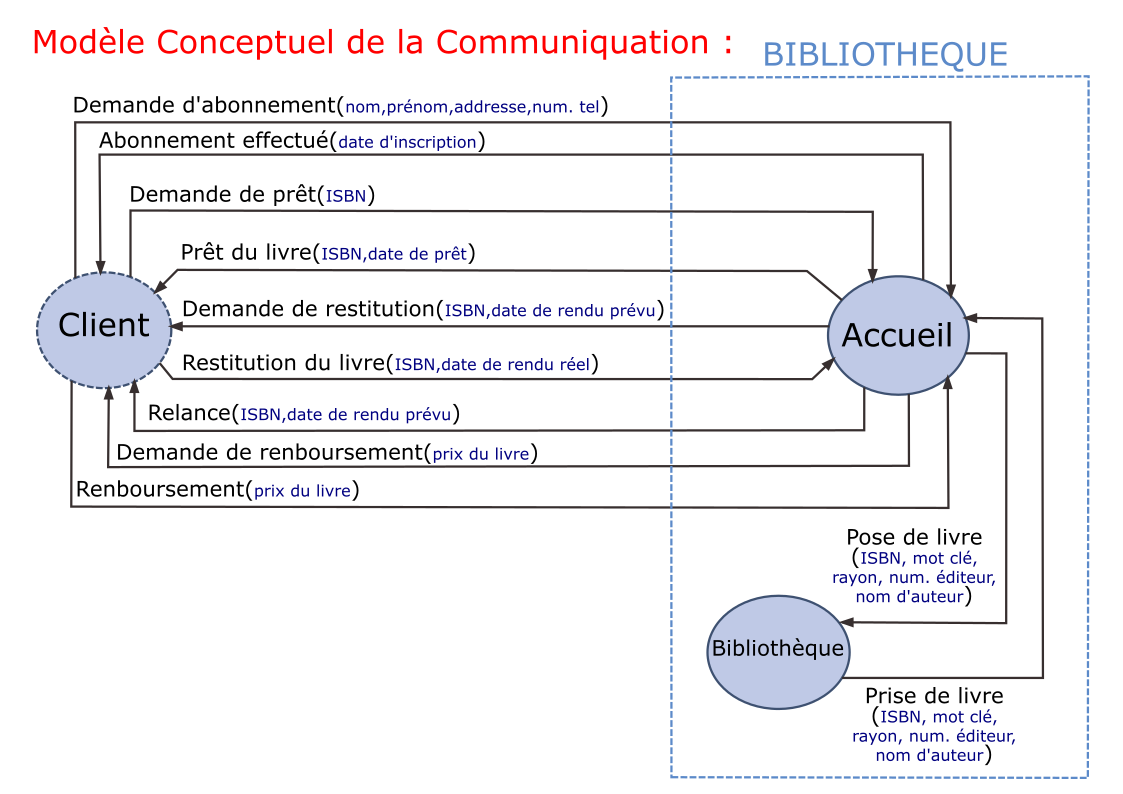
Identifiez et numérotez les exigences non-fonctionnelles en utilisant le préfixe ENF.]

# 3. Contraintes de conception

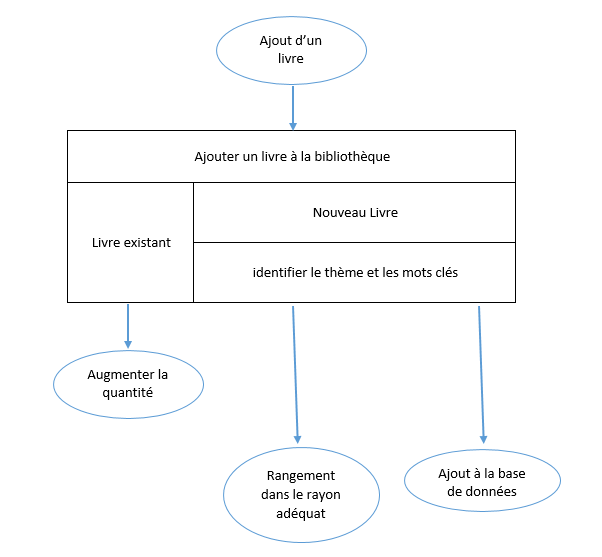
[Décrivez les contraintes dont vous devez tenir compte dans la conception et la réalisation de votre logiciel. Les contraintes sont des restrictions affectant une partie de votre logiciel ou son entièreté.

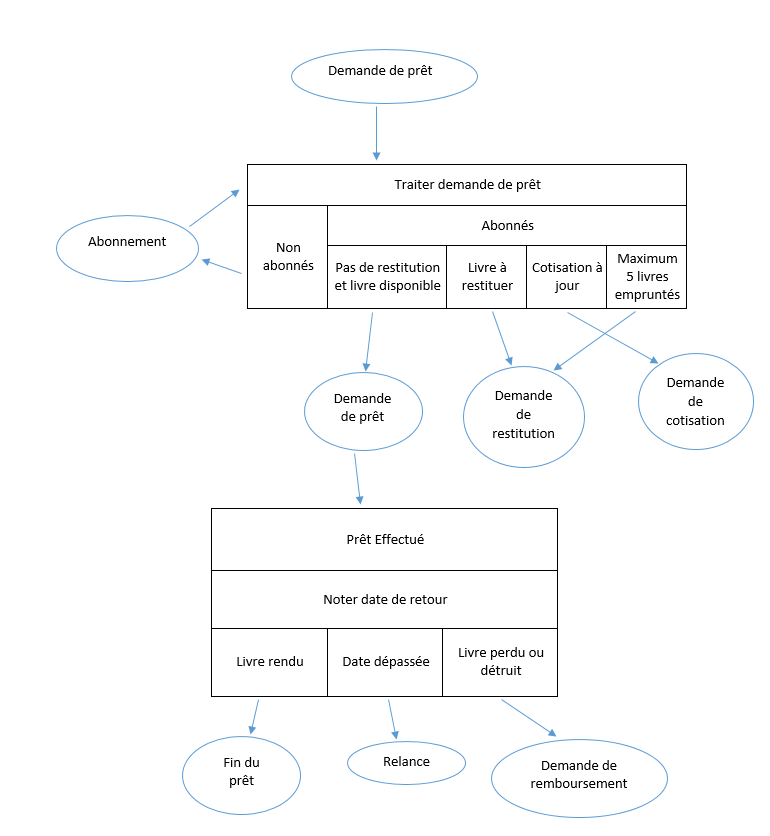
Par exemple, une contrainte de conception pourrait être la nécessité d’utiliser le cadriciel (« framework ») Hibernate dans le cadre du développement d’une application Java ou de faire l’utilisation du serveur d’authentification corporatif du client.  
  
Identifiez et numérotez les contraintes en utilisant le préfixe CON.]

# 4. Modèle conceptuel de communication (MCC)

**

**5. Modèle conceptuel des traitements**





# 6. Modèle conceptuel des données

# 7. Interfaces

## 7.1 Interfaces utilisateur

[Spécifiez l’apparence de l’interface utilisateur de votre application. Il peut s’agir d’un descriptif textuel ou d’une image prototype. Si votre application contient plusieurs écrans importants, fournir plusieurs spécifications ou images.  
  
Identifiez et numérotez les interfaces utilisateur en utilisant le préfixe IU.]

## 7.2 Interfaces matérielles

[Spécifiez les interfaces que votre application peut avoir avec des composantes physiques. Ceci est surtout nécessaire dans le cadre de projet où l’application sera embarquée sur une composante physique, par exemple une machine à café ou un parcomètre.   
  
Identifiez et numérotez les interfaces utilisateur en utilisant le préfixe IM.]

## 7.3 Interfaces logicielles

[Spécifiez les interfaces de votre application peut avoir avec des composantes logicielles. Par exemple, il peut s’agir d’une base de données, d’un service web.   
  
Identifiez et numérotez les interfaces logicielles en utilisant le préfixe IL.]

## 7.4 Interfaces de communications

[Spécifiez les interfaces que votre application peut avoir avec des composantes de communication. Par exemple, il peut s’agir de l’Internet, d’un réseau privé (LAN) ou bien des ondes cellulaires.  
  
Identifiez et numérotez les interfaces logicielles en utilisant le préfixe IC.]

# 8. Matrice de traçabilité

[La matrice de traçabilité effectue les liens de référence entre les caractéristiques issues des besoins listés dans le document des spécifications logicielles et les cas d’utilisation de ce document.   
  
Le tableau suivant donne un exemple d'une matrice de traçabilité.]

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Caractéristique / Besoin du client** | **Cas d’utilisation** | **Exigence fonctionnelle** | **Exigence non fonctionnelle** | **Contrainte** |
| Authentifier les utilisateurs | CU-1 | EF-1 | ENF-1 | CON-1 |
| Voir les dossiers étudiants | CU-2 | EF-2 | ENF-3 ENF-4 | CON-5 |
| Calculer cote Z | CU-3 | EN-5 | ENF-5 ENF-4  ENF-3 | CON-2 |