

UNIVERSITÉ DU QUÉBEC À MONTRÉAL

Spécifications des exigences d'un logiciel

Wiki de films

TRAVAIL PRÉSENTÉ À

MAKARENKOV Vladimir

DANS LE CADRE DU COURS

PROJET D'ANALYSE ET DE MODÉLISATION

INM5151 GROUPE 20

PAR

ALLARD Edmond - ALLE20109301

FRECHETTE DEMERS Jean-Sébastien - FREJ31088803

LEVAC Julien - LEVJ12058508

POIRIER Alexandre - POIA10108904

RHANEM Mourad - RHAM29099008

REMIS LE

9 JUIN 2015

Historique des versions

Version	Description	Auteur(s)	Date
1.0	Finalisation du document de SEL	<ul style="list-style-type: none">• ALLARD Edmond,• FRECHETTE DEMERS Jean-Sébastien,• LEVAC Julien,• POIRIER Alexandre,• RHANEM Mourad,	9 juin 2015

Table des matières

1 Introduction.....	11
1.1 Objectifs.....	11
1.2 Portée.....	11
1.3 Définitions, acronymes et abréviations.....	12
1.4 Documents de références.....	13
1.5 Aperçu du document.....	14
2 Description générale du logiciel.....	15
2.1 Perspective du produit.....	15
2.2 Vue d'ensemble des fonctions du produit.....	17
2.3 Description des utilisateurs.....	22
2.4 Contraintes d'ordre général.....	23
2.5 Hypothèses et dépendances.....	23
2.6 Répartition des exigences.....	24
3 Description détaillée.....	25
3.1 Interfaces externes.....	26
3.2 Spécifications fonctionnelles.....	72
3.2.1 Diagramme de séquences.....	72
3.2.2 Dictionnaire de données.....	90
3.2.3 Contrats.....	95
3.2.4 Modèle objets et modèle relationnel.....	127
3.3 Exigences d'opérations, de communications et de performance.....	129
3.3.1 Nombre de terminaux à supporter.....	129
3.3.2 Nombre d'utilisateurs simultanés.....	129
3.3.3 Quantité et types d'information à traiter.....	129
3.3.4 Nombre de transactions à traiter par unité de temps sous conditions normales et en période de pointe.....	129
3.4 Exigences logiques de la base de données.....	130
3.4.1 Types d'informations utilisées par les différentes fonctions.....	130
3.4.2 Fréquence d'utilisation.....	130

3.4.3 Capacité d'accès.....	130
3.4.4 Exigences de rétention des données.....	130
3.5 Contraintes de conception.....	131
3.6 Exigences non fonctionnelles.....	131
3.7 Organisation des exigences spécifiques.....	133
4 Informations complémentaires.....	134
4.1 Annexes.....	134

Liste des tableaux

Tableau 1: Cas d'utilisation « S'inscrire».....	27
Tableau 2: Cas d'utilisation «Consulter l'aide ou la FAQ».....	28
Tableau 3: Cas d'utilisation «Consulter les règlements».....	29
Tableau 4: Cas d'utilisation «Consulter la liste de films déjà visionnés».....	30
Tableau 5: Cas d'utilisation «Ajouter un film à la liste de films déjà visionnés».....	31
Tableau 6: Cas d'utilisation «Consulter les recommandations personnalisées».....	32
Tableau 7: Cas d'utilisation «Créer la fiche d'un film».....	33
Tableau 8: Cas d'utilisation «Rechercher un film».....	34
Tableau 9: Cas d'utilisation «Consulter la fiche d'un film».....	35
Tableau 10: Cas d'utilisation «Éditer la fiche d'un film».....	36
Tableau 11: Cas d'utilisation «Critiquer un film».....	37
Tableau 12: Cas d'utilisation «Éditer une note donnée à un film».....	38
Tableau 13: Cas d'utilisation «Éditer une critique donnée à un film».....	39
Tableau 14: Cas d'utilisation «Rapporter un abus sur un film».....	40
Tableau 15: Cas d'utilisation «Rapporter un abus sur une critique».....	41
Tableau 16: Cas d'utilisation « Consulter un profil publique».....	43
Tableau 17: Cas d'utilisation «Rapporter un abus d'un membre».....	44
Tableau 18: Cas d'utilisation «Consulter son compte».....	45
Tableau 19: Cas d'utilisation «Éditer son compte».....	47
Tableau 20: Cas d'utilisation «Envoyer un message privé à un membre».....	49
Tableau 21: Cas d'utilisation «Consulter la liste des messages privés reçus».....	50
Tableau 22: Cas d'utilisation «Consulter la liste des messages privés envoyés».....	51
Tableau 23: Cas d'utilisation «Lire un messages privés».....	52
Tableau 24: Cas d'utilisation «Supprimer un messages privés».....	53
Tableau 25: Cas d'utilisation «Consulter la listed'abus».....	54
Tableau 26: Cas d'utilisation «Retirer un abus de la liste d'abus».....	55
Tableau 27: Cas d'utilisation «Suspendre un membre».....	57
Tableau 28: Cas d'utilisation «Avertir un membre».....	59

Tableau 29: Cas d'utilisation «Censurer une critique».....	61
Tableau 30: Cas d'utilisation «Supprimer une critique».....	63
Tableau 31: Cas d'utilisation «Restaurer une version précédente de la fiche d'un film».....	64
Tableau 32: Cas d'utilisation «Protéger un film contre les modifications».....	65
Tableau 33: Cas d'utilisation «Supprimer un film».....	66
Tableau 34: Cas d'utilisation «Bannir un membre».....	67
Tableau 35: Cas d'utilisation «Promouvoir un membre».....	69
Tableau 36: Cas d'utilisation «Effectuer un erestauration de la base de données».....	71
Tableau 37: Dictionnaire de données : Utilisateur.....	90
Tableau 38: Dictionnaire de données : Invite.....	90
Tableau 39: Dictionnaire de données : Membre.....	90
Tableau 40: Dictionnaire de données : Modérateur.....	90
Tableau 41: Dictionnaire de données : Administrateur.....	91
Tableau 42: Dictionnaire de données : Profil.....	91
Tableau 43: Dictionnaire de données : Compte.....	91
Tableau 44: Dictionnaire de données : BoiteCourriel.....	91
Tableau 45: Dictionnaire de données : Courriel.....	92
Tableau 46: Dictionnaire de données : Film.....	92
Tableau 47: Dictionnaire de données : Role.....	92
Tableau 48: Dictionnaire de données : Acteur.....	93
Tableau 49: Dictionnaire de données : NoteDonnee.....	93
Tableau 50: Dictionnaire de données : Recommandation.....	93
Tableau 51: Dictionnaire de données : MotCle.....	93
Tableau 52: Dictionnaire de données : Critique.....	94
Tableau 53: Dictionnaire de données : AbusModérateur.....	94
Tableau 54: Dictionnaire de données : Abus.....	94
Tableau 55: Contrat getInscription.....	95
Tableau 56: Contrat postInscription.....	96
Tableau 57: Contrat getFaq.....	96
Tableau 58: Contrat getReglements.....	97
Tableau 59: Contrat getFilmVisionne.....	97

Tableau 60: Contrat ajouterFilmAListeFilmVisionne.....	98
Tableau 61: Contrat getModifierNote.....	99
Tableau 62: Contrat setNote.....	99
Tableau 63: Contrat getAjoutCritique.....	100
Tableau 64: Contrat setCritique.....	100
Tableau 65: Contrat getModificationCritique.....	101
Tableau 66: Contrat getRecommandationsPersonnalisees.....	101
Tableau 67: Contrat getAjouterFilm.....	102
Tableau 68: Contrat ajouterFilm.....	102
Tableau 69: Contrat getRechercheFilm.....	103
Tableau 70: Contrat getFilm.....	103
Tableau 71: Contrat getFilm.....	104
Tableau 72: Contrat getEditerFilm.....	104
Tableau 73: Contrat getEditerFilm.....	105
Tableau 74: Contrat signalerFilm.....	105
Tableau 75: Contrat signalerCritique.....	106
Tableau 76: Contrat getRechercherProfilPublique.....	106
Tableau 77: Contrat rechercherProfil.....	107
Tableau 78: Contrat getProfilPublique.....	107
Tableau 79: Contrat getSignalerMembre.....	108
Tableau 80: Contrat signalerMembre.....	108
Tableau 81: Contrat getCompte.....	109
Tableau 82: Contrat getModificationCompte.....	109
Tableau 83: Contrat setCompte.....	110
Tableau 84: Contrat getCreerMessage.....	110
Tableau 85: Contrat rechercherDestinataires.....	111
Tableau 86: Contrat postMessage.....	111
Tableau 87: Contrat getMessagesRecus.....	112
Tableau 88: Contrat getMessagesEnvoyes.....	112
Tableau 89: Contrat getMessage.....	113
Tableau 90: Contrat supprimerMessage.....	113

Tableau 91: Contrat getListeAbus.....	114
Tableau 92: Contrat getResoudreAbus.....	114
Tableau 93: Contrat postResoudreAbus.....	115
Tableau 94: Contrat getSuspendreMembre.....	115
Tableau 95: Contrat postAvertissementMembre.....	116
Tableau 96: Contrat sauvegarderFormulaireAvertissement.....	116
Tableau 97: Contrat postAvertissementMembre.....	117
Tableau 98: Contrat sauvegarderFormulaireSuspension.....	117
Tableau 99: Contrat getCensurerCritique.....	118
Tableau 100: Contrat setCensureCritique.....	118
Tableau 101: Contrat getSupprimerCritique.....	119
Tableau 102: Contrat setSuppressionCritique.....	119
Tableau 103: Contrat sauvegarderFormulaireSuppressionCritique.....	120
Tableau 104: Contrat getRestaurationFicheFilm.....	120
Tableau 105: Contrat sauvegarderFicheFilm.....	121
Tableau 106: Contrat restaurerFicheFilm.....	121
Tableau 107: Contrat getProtectionFicheFilm.....	122
Tableau 108: Contrat protegerFicheFilm.....	122
Tableau 109: Contrat getSuppressionFicheFilm.....	123
Tableau 110: Contrat supprimerFicheFilm.....	123
Tableau 111: Contrat bannirMembre.....	124
Tableau 112: Contrat getPromouvoirMembre.....	124
Tableau 113: Contrat setPromotion.....	125
Tableau 114: Contrat getVersionsBaseDeDonnees.....	125
Tableau 115: Contrat restaurationBaseDeDonnees.....	126

Liste des figures

Figure 1: Trois principales fonctionnalités du wiki de films.....	16
Figure 2: Architecture du Wiki de films.....	17
Figure 3: Diagramme de cas d'utilisation.....	25
Figure 4: Diagramme de séquences : «Ajouter un film à la liste de films déjà visionnés».....	72
Figure 5: Diagramme de séquences : «Avertir un membre».....	72
Figure 6: Diagramme de séquences : «Bannir un membre».....	73
Figure 7: Diagramme de séquences : «Censurer une critique».....	73
Figure 8: Diagramme de séquences : «Consulter aide et FAQ».....	74
Figure 9: Diagramme de séquences : «Consulter la fiche d'un film».....	74
Figure 10: Diagramme de séquences : «Consulter la liste d'abus».....	74
Figure 11: Diagramme de séquences : «Consulter la liste de films déjà visionnés».....	75
Figure 12: Diagramme de séquences : «Consulter la liste de message privés reçus».....	75
Figure 13: Diagramme de séquences : «Consulter la liste de messages privés envoyés».....	76
Figure 14: Diagramme de séquences : «Consulter recommandations personnalisées».....	76
Figure 15: Diagramme de séquences : «Consulter règlements».....	77
Figure 16: Diagramme de séquences : «Consulter son compte».....	77
Figure 17: Diagramme de séquences : «Consulter un profil public».....	78
Figure 18: Diagramme de séquences : «Critiquer un film».....	78
Figure 19: Diagramme de séquences : «Créer la fiche d'un film».....	79
Figure 20: Diagramme de séquences : «Éditer une critique donnée à un film».....	79
Figure 21: Diagramme de séquences : «Effectuer une restauration de la base de données».....	80
Figure 22: Diagramme de séquences : «Envoyer un message privé à un membre».....	81
Figure 23: Diagramme de séquences : «Lire un message privé».....	81
Figure 24: Diagramme de séquences : «Promouvoir un membre».....	82
Figure 25: Diagramme de séquences : «Protéger un film contre les modifications».....	82
Figure 26: Diagramme de séquences : «Rapporter un abus d'un membre».....	83
Figure 27: Diagramme de séquences : «Rapporter un abus sur un film».....	83
Figure 28: Diagramme de séquences : «Rapporter un abus sur une critique».....	84

Figure 29: Diagramme de séquences : «Rechercher un film».....	84
Figure 30: Diagramme de séquences : «Restaurer une version précédente de la fiche d'un film».....	85
Figure 31: Diagramme de séquences : «Retirer un abus de la liste d'abus».....	85
Figure 32: Diagramme de séquences : «S'inscrire».....	86
Figure 33: Diagramme de séquences : «Supprimer un film de la base de données».....	86
Figure 34: Diagramme de séquences : «Supprimer un message privé».....	87
Figure 35: Diagramme de séquences : «Supprimer une critique».....	87
Figure 36: Diagramme de séquences : «Suspendre un membre».....	87
Figure 37: Diagramme de séquences : «Éditer la fiche d'un film».....	88
Figure 38: Diagramme de séquences : «Éditer son compte».....	89
Figure 39: Diagramme de séquences : «Éditer une note donnée à un film».....	89
Figure 40: Modèle objet.....	127
Figure 41: Schéma relationnel.....	128
Figure 42: Page d'accueil du Wiki de films.....	134
Figure 43: Liste de films déjà visionnés.....	135
Figure 44: Fiche d'un film.....	136
Figure 45: Création de la fiche d'un film.....	137
Figure 46: Autre fiche d'un film.....	138
Figure 47: Version mise à jour de la liste de films déjà visionnés.....	139
Figure 48: Inscription.....	140

1 Introduction

1.1 Objectifs

Ce document s'adresse à la direction, aux développeurs qui programmeront le logiciel et aussi aux clients potentiels qui sont intéressés par le produit et les services offerts.

Les objectifs sont de détailler le système, définir ses exigences et contraintes, définir les fonctions et le déroulement de celles-ci, l'environnement dans lequel le système va pouvoir être implémenté ainsi que les intervenants et les niveaux d'utilisation du système.

1.2 Portée

Notre site a pour principal objectif de permettre aux visiteurs et aux utilisateurs de trouver rapidement de nouveaux films à écouter en utilisant l'avis des autres usagers.

Une des principales fonctionnalités sera de permettre à l'utilisateur de pouvoir ajouter un film dans sa liste des films visionnés. Cette liste servira de base pour notre système afin de recommander des films.

Le site aura un système de gestion décentralisé. Les utilisateurs auront la possibilité d'ajouter les films dans la banque de données.

Notre système prendra en compte différents rôles : les utilisateurs non-inscrits/non authentifiés, les utilisateurs inscrits/authentifiés, les modérateurs, les administrateurs, et l'administrateur de base de données.

Le site fonctionnera sous un serveur Tomcat. Le style architectural utilisé sera de type

Modèle-Vue-Contrôleur. Le contrôleur sera écrit en Java avec la structure logicielle Spring. La base de données sera de type MySQL. La structure logicielle Hibernate servira à relier notre base de données avec du code Java afin d'offrir de la persistance de données. La vue sera générée avec HTML5 et mise en forme à l'aide de feuilles de style CSS3.

1.3 Définitions, acronymes et abréviations

Git : logiciel de gestion de versions décentralisé. (Wikipedia)

Grails : *framework open source* de développement agile d'applications web basé sur le langage Groovy et sur le patron de conception Modèle-Vue-Contrôleur. (Wikipedia)

Groovy : langage de programmation orienté objet destiné à la plate-forme Java. Il constitue une alternative au langage Java pour cette plate-forme et est inspiré de Python, Ruby et Smalltalk. (Wikipedia)

Tomcat (Apache) : serveur web pour des applications Java.

MySQL : Système de gestion de bases de données relationnelles. (Wikipedia)

Java : Langage de programmation orienté objet.

Spring : Framework (structure logicielle) permettant de définir et de construire une application Java. (Wikipedia)

MySQL : Système de gestion de bases de données relationnelles. (Wikipedia)

DBA : Administrateur de base de données (DataBase Administrator).

SGBD : Système de gestion de base de données

Hibernate : Framework (structure logicielle) open source permettant de gérer la persistance des objets dans une base de données relationnelle (Wikipedia).

Structure logicielle (*framework* en anglais) : ensemble cohérent de composants logiciels qui permet de mettre en place la structure principale d'une partie ou de l'ensemble du logiciel.

HTML : L'*Hypertext Markup Language*, est le format de données conçu pour représenter les pages web. (Wikipedia)

CSS : Les feuilles de style en cascade, généralement appelées CSS de l'anglais *Cascading Style Sheets*, forment un langage informatique qui décrit la présentation des documents HTML et XML. (Wikipedia)

Javascript : JavaScript est un langage de programmation de scripts, exécuté côté client, principalement employé dans les pages web interactives mais aussi pour les serveurs. (Wikipedia)

1.4 Documents de références

- INTERNET MOVIE DATABASE (Page consultée le 7 juin 2015). *Site de l'Internet Movie Database*, [en ligne], <http://www.imdb.com>
- ROTTEN TOMATOES (Page consultée le 7 juin 2015). *Site de Rotten Tomatoes*, [en ligne], <http://www.rottentomatoes.com>
- THE MOVIE DB (Page consultée le 7 juin 2015). *Site de TMDb*, [en ligne], <http://www.themoviedb.org>
- MOVIELENS (Page consultée le 7 juin 2015). *Site de MovieLens*, [en ligne], <http://www.movielens.org>

1.5 Aperçu du document

La première section du présent document consiste en l'introduction à la conception du projet. Elle contient aussi une liste de définitions de termes qui se retrouveront dans ce document.

La deuxième section contient une description du produit: son but, son utilité, à qui il s'adresse mais aussi ses spécifications et exigences techniques.

La troisième section présente le travail de conception du projet. On y retrouve les cas d'utilisation détaillés, le modèle objet, le diagramme de la base de données, une description des fonctions principales du système et plusieurs contraintes du système y sont expliquées.

La quatrième section contient l'annexe qui comprend des captures de notre interface.

2 Description générale du logiciel

2.1 Perspective du produit

Cette section a pour but de de donner une première présentation des principaux objectifs devant être atteints par le système Wiki de films. La manière d'atteindre ces buts y sera aussi décrite comment à l'aide d'un diagramme expliquant l'architecture du système.

Tel que présenté dans la figure x, les trois principales fonctionnalités qui caractérisent le système sont les suivantes :

Consulter et créer des fiches de films :

- Noter des films
- Recevoir des recommandations de films

Plusieurs autres fonctionnalités permettant, entres autres, la gestion de ce système sont aussi essentielles et seront décrites dans la suite de ce document. Cependant, les trois fonctionnalités précédentes permettent à Wiki de films de se distinguer des principaux sites web qui forment la compétition puisque nous ne connaissons pas de sites web permettant à la fois la recommandation de films et l'ajout de contenu par de simples membres du site.

La figure 2 permet quant à elle de montrer le style architectural utilisé pour ce système : il s'agit du style «Modèle-vue-contrôleur».

L'utilisateur reçoit d'abord la vue générée par le modèle. La vue sera une page web en HTML avec du code CSS et du code Javascript. L'utilisateur décide d'une action sur la vue et cette action (par exemple, cliquer sur un bouton) est envoyée au contrôleur. Le contrôleur est un ensemble de classes qui permettent d'exécuter, du côté serveur, la requête voulue. Le contrôleur va donc modifier au besoin le modèle et mettre à jour la vue de l'utilisateur avec les bonnes informations provenant du modèle mis à jour. Finalement, c'est au niveau du modèle que se trouve toute l'information liée à la base de données. Ce modèle est représenté

sous forme de diagramme de classes et de schéma relationnel à la section 3.2.4 de ce document.

En résumé, le système Wiki de films est donc un site web utilisant une architecture assez classique pour ce type de système mais il se distingue de la compétition par ses fonctionnalités qui font en sorte qu'il est permet aussi la contribution directe des membres du site (en plus de la recommandation de films).

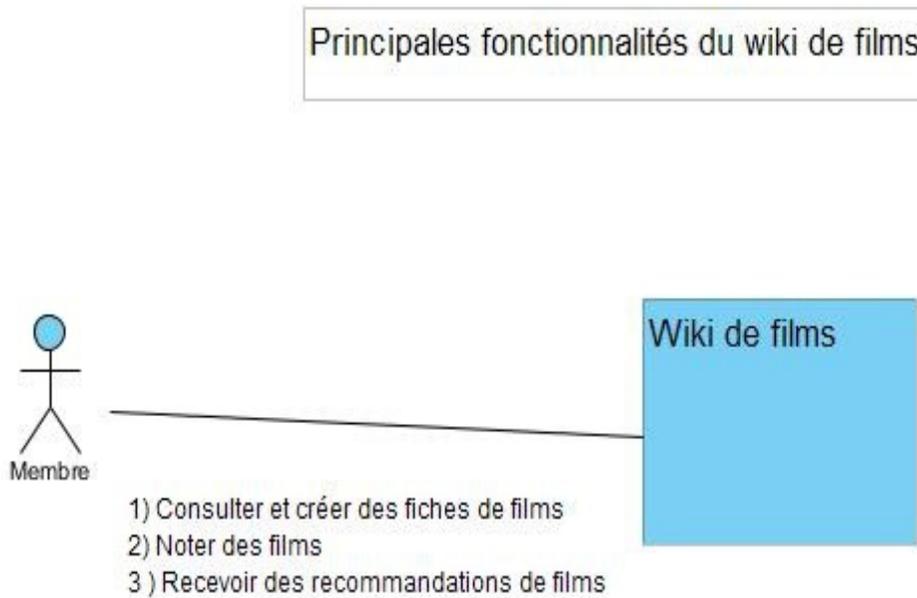


Figure 1: Trois principales fonctionnalités du wiki de films

Architecture du Wiki de films : architecture de type modèle-vue-contrôleur

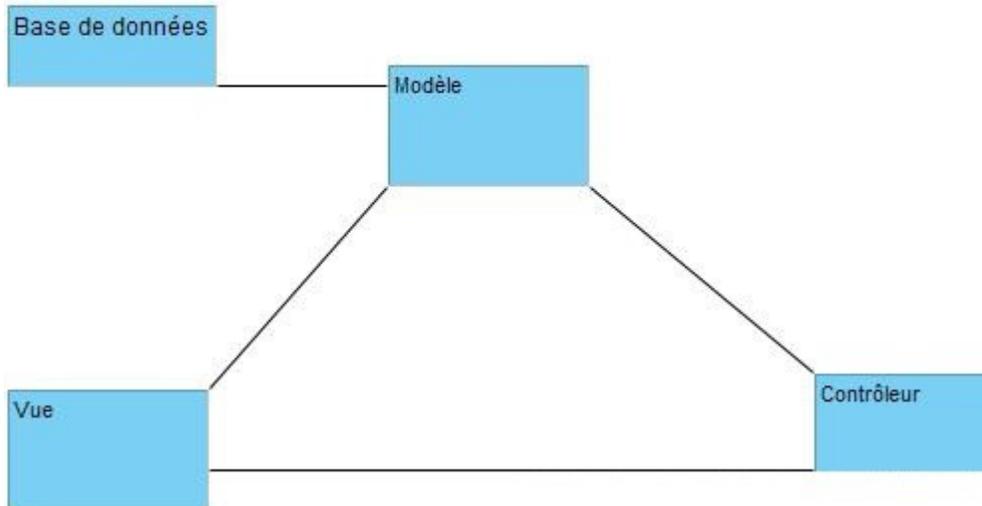


Figure 2: Architecture du Wiki de films

2.2 Vue d'ensemble des fonctions du produit

Cette section est reprise du document de présentation du concept de wiki de films et présente une vue d'ensemble des fonctions du produit. C'est la section des scénarios d'opération du document de présentation du concept.

- Utilisateur visiteur
 - S'inscrire
 - Il accède à la page d'inscription. Il y saisit son adresse courriel ainsi que son nom d'utilisateur et un mot de passe. Le système lui envoie une confirmation par courriel pour valider son inscription. Il clique sur le lien inclus dans le courriel pour activer son compte.
 - Voir les informations d'un film
 - Il accède à la page d'un film où sont disposées les informations concernant le film en question.

- Utilisateur régulier (inscrit)
 - Gérer profil (modifier ses paramètres)
 - L'utilisateur inscrit accède à sa page de profil où il dispose de champs d'informations personnelles qu'il peut modifier. Il peut valider les modifications qu'il souhaite effectuer ou les annuler en cliquant sur le bouton correspondant.
 - Ajouter un film
 - L'utilisateur accède à un formulaire pour ajouter un film. L'utilisateur remplit les différents champs requis. Le système vérifie automatiquement si le nom du film a déjà été ajouté. Si c'est le cas, un message d'erreur informe l'utilisateur que le film existe déjà. Sinon le film est ajouté dans la base de données.
 - Modifier les informations d'un film
 - L'utilisateur accède à la page d'un film et clique sur modifier. Il peut ainsi modifier les informations d'un film tels que la description, les acteurs et les photos.
 - Ajouter un film à sa liste
 - L'utilisateur qui parcourt la page d'un film peut cliquer sur un menu déroulant afin de sélectionner l'une de ces trois options : Vu, Abandonné ou Prévu afin d'indiquer l'état du film. Un deuxième menu déroulant apparaît ensuite en dessous afin d'inviter l'utilisateur à ajouter une note. L'utilisateur n'est pas obligé de noter le film immédiatement. L'utilisateur doit confirmer son choix en cliquant sur le bouton «Ajouter à la liste».
 - Gérer sa liste de films notés
 - Un utilisateur peut accéder à sa liste de film directement dans le menu principal. Chaque film dans sa liste comprend un menu déroulant permettant de changer l'état du film (Vu, Abandonné, Prévu) ou de changer le note.
 - Supprimer un film de la liste
 - À partir de la liste de film, un utilisateur peut supprimer un film en cliquant sur le

bouton supprimer à côté du film. Un message de confirmation apparaîtra afin de confirmer le choix de l'utilisateur.

- Noter un film
 - Un utilisateur peut noter un film au moment de l'ajouter ou encore lorsqu'il souhaite supprimer un film de sa liste.
- Critiquer un film
 - Il accède à la page d'un film et clique sur le bouton «Ajouter une critique». Il écrit un titre, rédige une critique et publie cette dernière.
- Modifier sa critique d'un film
 - L'utilisateur accède à la page d'un film. Il clique sur le bouton «Éditer» sur une de ses critiques afin de la modifier.
- Rechercher un film (avec titre, genre, acteur, etc...)
 - Dans le champ de recherche, l'utilisateur sélectionne le ou les critères de recherche du film (titre, genre, acteur, etc.). Le système effectue ensuite une requête sur la base de données des films en fonction des critères saisis par l'utilisateur, puis il affiche le résultat de la recherche.
- Signaler un film pour modération
 - Un utilisateur qui voit un film qui n'existe pas ou dont la description n'est pas conforme aux règles du site peut cliquer sur le bouton «Signaler un film». Le film est ensuite entré dans la liste des films à modérer. Cette liste est accessible à tous les modérateurs.
- Signaler une critique pour modération
 - Un utilisateur qui voit un critique qui n'est pas conforme aux règles du site peut signaler la critique. La critique est ensuite envoyée dans la liste des critiques à modérer.

- Modérateur
 - Retourner à une ancienne version de la fiche d'un film
 - Le modérateur accède à la liste des versions de la fiche d'un film. Il peut sélectionner une de ces versions pour la restaurer.
 - Modifier une critique
 - En consultant une critique, un modérateur peut cliquer sur le bouton «modifier». L'interface d'édition des critiques apparaît et contient l'ancienne critique dont le modérateur peut modifier le texte. Ce même modérateur peut ensuite publier la critique ainsi modifiée.
 - Supprimer une critique
 - En consultant une critique, un modérateur peut cliquer sur le bouton «Supprimer». Un message de confirmation s'affiche avant la suppression définitive.
 - Modérer un film
 - La liste comprend tous les films qui ont été ajoutés par les utilisateurs. Un modérateur peut alors cliquer sur un film afin de s'assurer si une modération est requise.
 - Si le film existe déjà dans la liste d'une façon ou d'une autre, un modérateur peut supprimer le film. Le système invite le modérateur à mettre un avertissement au dossier si nécessaire.
 - Si les informations du film enfreignent les règles du site, le modérateur peut supprimer ou modifier le film pour être conforme aux règles. Le système invite le modérateur à mettre un avertissement au dossier si nécessaire.
 - Si le film est conforme, le modérateur peut rapporter la demande de modération comme abusive. Auquel cas, l'utilisateur concerné reçoit un avertissement.

- Modérer une critique
 - La modération d'une critique se fait de la même façon que celle d'un film.
- Protéger un film contre les modifications
 - Le modérateur dispose d'un bouton au niveau de la page du film pour activer la protection du film contre les modifications.
- Suspendre un utilisateur
 - Le modérateur accède à la page de profil du membre où il dispose d'un bouton pour suspendre le membre en question. De là, il peut définir la durée de suspension de ce membre. Par la suite, ce membre est averti de sa suspension.
- Supprimer un film
 - Sur la page de chaque film, le modérateur peut cliquer sur le bouton supprimer. Un message de confirmation apparaît pour demander la confirmation.
- Modifier un film
 - Un modérateur peut utiliser l'option «Modifier un film» qui permet d'éditer le titre, la description et l'image d'un film.
- Administrateur
 - Bannir un utilisateur
 - L'administrateur accède à la page de profil du membre où il dispose d'un bouton pour bannir le membre en question. Par la suite, le membre est averti de son bannissement.
 - Promouvoir un utilisateur
 - L'administrateur accède à la page profil du membre où il dispose d'une option pour promouvoir l'utilisateur. Il peut ensuite choisir de promouvoir le membre en question au rôle de modérateur ou d'administrateur. Par la suite, ce membre est averti de sa promotion.

- Administrateur de base de données
 - Retourner à une sauvegarde de la base de données des films
 - L'administrateur de base de données dispose d'une liste de sauvegardes de la base de données des films. Il peut sélectionner une des sauvegardes pour la restaurer.
- Système
 - Recommander film (Page film)
 - Lorsqu'une fiche est créée pour un nouveau film dans la base de données, le système utilise l'algorithme de recommandation pour créer la liste de recommandations initiales. L'algorithme l'utilisera des critères comme le genre, les thèmes, acteurs, réalisateur, scénariste, mais aussi d'autres critères comme le pays d'origine et la durée. Si une fiche est modifiée, le système recalcule de façon similaire les recommandations.
 - Recommander film (Page Accueil)
 - Après que l'utilisateur ait noté un nombre suffisant de films, le système affichera une liste des films qui seront retournés par un algorithme de recommandation différent de celui des films.

2.3 Description des utilisateurs

Notre site étant de langue française, nous visons un public francophone.

Le visiteur est typiquement une personne ayant trouvé le site web via un moteur de recherche ou par un lien d'un autre site. C'est donc à nous de l'attirer afin d'en faire un membre régulier. Un membre est un utilisateur inscrit. Typiquement, il visite le film de façon sporadique selon ces besoins et ces envies d'écouter de nouveaux films.

Un modérateur est un utilisateur qui fréquente le site plus régulièrement. Il est probablement un cinéphile qui cherche à s'impliquer directement dans le développement du wiki.

Un administrateur est principalement un membre de l'équipe de développement ou de la compagnie qui gère le site. Les administrateurs originaux ont tous une éducation universitaire et étudient en Informatique.

2.4 Contraintes d'ordre général

Dans le document de présentation du concept, il était question de différentes technologies employées pour la création du système Wiki de films. Les mêmes technologies seront employées mais une couche technologique additionnelle sera également utilisée. Il s'agit de la structure logicielle Grails qui suit la pratique «Convention plutôt que configuration». Cela permet de réduire de beaucoup le code nécessaire au bon déroulement du site. Grails s'intègre dans l'univers de développement Java.

Les autres technologies employées seront donc répartis au niveau du client, du serveur et de la base de données.

Au niveau du serveur, l'hébergement se fera sous un serveur Tomcat. Le code sera en Groovy (supporté par la plateforme Java) dans un style architectural de type Modèle-vue-contrôleur à l'aide de la structure logicielle Grails décrite au début de cette section. Hibernate permettra la gestion des requêtes à la base de données.

La base de données sera de type mySQL et les requêtes seront écrites en HQL(Hibernate Query Langage).

2.5 Hypothèses et dépendances

Notre site utilisera des technologies récentes telle que le HTML5, des feuilles de style (CSS), du Javascript. Ces technologies ne sont pas compatibles avec tous les navigateurs. Nous considérons que notre site web ne sera compatible qu'avec les navigateurs Firefox et Google Chrome à jour ou encore Internet Explorer 10. Nous n'excluons

pas la compatibilité avec d'autres navigateurs, mais les bugs et les problèmes d'affichage ne seront pas supportés.

2.6 Répartition des exigences

La base même du site est la collection de fiche de film. Il faut donc que dans la première version, un utilisateur puisse insérer des nouveaux films dans la base de données.

Le site web devrait permettre aux utilisateurs de noter les films. Ces films seront sauvegardés dans une liste que l'utilisateur pourra consulter. Ces notes permettront de générer des suggestions personnalisées en fonction des utilisateurs. Ces exigences sont désirables et devrait être dans le premier livrable.

Les modérateurs devront être capables de contrôler ceux souhaitant nuire au site web. Bien que cette fonctionnalité soit désirable, elle n'est pas essentielle lors de la première sortie du site puisque l'achalandage sera plutôt faible.

Nous espérons aussi ajouter une série de fonctionnalités optionnelles.

À partir des notes des utilisateurs, nous espérons générer une liste des films les mieux notés. Cette liste requiert toutefois un niveau d'achalandage acceptable.

L'ajout d'un forum intégré avec les comptes utilisateurs du site serait appréciable pour stimuler les échanges entre membres.

Il serait souhaitable d'inclure des fonctionnalités pour permettre aux membres de partager leurs listes de film sur d'autres réseaux sociaux. Dans le cas contraire, nous pourrions permettre aux utilisateurs de rendre leur liste privée.

3.1 Interfaces externes

Convention pour ID : CU[numéro d'acteur][numéro de use case]

Numéro d'acteurs :

0. Utilisateur
1. Utilisateur non-inscrit
2. Membre
3. Modérateur
4. Administrateur
5. Administrateur de la base de données

ex: CU101 est le cas d'utilisation 1 de l'acteur «Utilisateur non-inscrit».

ID :	CU101	
Nom :	S'inscrire	
Acteurs :	Utilisateur non-inscrit	
Objectif :	Créer un compte dans le système Wiki de films	
Résumé :	Après avoir entré l'adresse du site web de wiki de films, l'utilisateur non-inscrit est invité à se connecter au site ou à créer un nouveau compte. L'utilisateur non-inscrit choisit l'option pour s'inscrire, remplit le formulaire de création de compte et soumet le formulaire. Un nouveau compte pour l'utilisateur est ajouté à la base de données.	
Type :	Essentiel	
Références :	getInscription (CO101), postInscription (102)	
Pré-condition :	Ne pas être authentifié Être sur la page d'accueil	
Post-condition :	Être authentifié dans le nouveau compte qui vient d'être créé	
Interactions :	Aucune	
	Acteurs	Systeme
1.	Choisit l'option pour s'inscrire	
2.		Envoie le formulaire d'inscription
3.	Remplit le formulaire d'inscription et le soumet	
4.		Ajoute un nouveau compte dans la base de données avec les données correspondantes à celles du formulaire
Scénarios alternatifs (Exceptions)		
E101		
a. Champs obligatoires manquant dans le formulaire d'inscription		

Tableau 1: Cas d'utilisation « S'inscrire »

ID :	CU001	
Nom :	Consulter l'aide ou la FAQ	
Acteurs :	Utilisateur, utilisateur non-inscrit, membre, modérateur, administrateur	
Objectif :	Consulter la page web du site qui donne les informations sur les questions fréquemment demandées et l'aide générale.	
Résumé :	L'utilisateur clique sur le lien d'aide sur la page d'accueil du site et obtient une autre page web en retour lui présentant de l'information sur les questions fréquemment posées (FAQ) et de l'aide générale.	
Type :	Optionnel	
Références :	getFaq (CO001)	
Pré-condition :	Être sur la page d'accueil du site wiki de films	
Post-condition :	La page web demandé a été envoyée	
Interactions :	Aucune	
	Acteurs	Système
1.	Choisit l'option de l'aide et FAQ	
2.		Envoie la page web d'aide et de FAQ
Scénarios alternatifs (Exceptions)		
E001		
Aucun		

Tableau 2: Cas d'utilisation «Consulter l'aide ou la FAQ»

ID :	CU002	
Nom :	Consulter les règlements	
Acteurs :	Utilisateur, utilisateur non-inscrit, membre, modérateur, administrateur	
Objectif :	Consulter la page web du site qui donne les informations sur les règlements du site.	
Résumé :	L'utilisateur clique sur le lien de règlement sur la page d'accueil du site et obtient une autre page web en retour lui présentant de l'information sur les règlements du site web.	
Type :	Essentiel	
Références :	getReglements (CO102)	
Pré-condition :	Être sur la page d'accueil du site wiki de films	
Post-condition :	La page web demandé a été envoyée	
Interactions :	Aucune	
	Acteurs	Système
1.	Choisit l'option de règlements	
2.		Envoie la page web des règlements
Scénarios alternatifs (Exceptions)		
E001		
Aucun		

Tableau 3: Cas d'utilisation «Consulter les règlements»

ID :	CU201	
Nom :	Consulter la liste de films déjà visionnés	
Acteurs :	Membre, modérateur, administrateur	
Objectif :	Afficher la liste de films déjà visionnés pour que le membre puisse la consulter.	
Résumé :	Le membre accède à l'ensemble des films visionnés en cliquant sur le lien correspondant	
Type :	Essentiel	
Références :	getFilmVisionne (CO201)	
Pré-condition :	Être authentifié	
Post-condition :	La liste de films déjà visionnés a été envoyée par le serveur.	
Interactions :	Aucune	
	Acteurs	Système
1.	Clique sur le lien de ses films déjà visionnés.	
2.		Génère et envoie une page web affichant tous les films déjà visionnés.
Scénarios alternatifs (Exceptions)		
E201		
Aucun		

Tableau 4: Cas d'utilisation «Consulter la liste de films déjà visionnés»

ID :	CU202	
Nom :	Ajouter un film à la liste de films déjà visionnés	
Acteurs :	Membre, modérateur, administrateur	
Objectif :	Ajouter un film à la liste de films déjà visionnés	
Résumé :	Le membre clique sur l'option pour ajouter le film à la liste de films déjà visionnés.	
Type :	Essentiel	
Références :	ajouterFilmAListeFilmVisionne (CO202)	
Pré-condition :	Être authentifié Être sur la fiche d'un film	
Post-condition :		
Interactions :	Étend «Consulter fiche de film» (CU206)	
	Acteurs	Système
1.	Clique sur l'option pour ajouter le film à la liste de films déjà visionnés et donne une note	
2.		Ajoute le film à la liste de films déjà visionnés dans la base de données pour le membre concerné avec la note donnée
Scénarios alternatifs (Exceptions)		
E202		
<ul style="list-style-type: none"> a. Tous les films de la base de données ont déjà été ajoutés à la liste de films déjà visionnés. b. Le film a déjà été ajouté à la liste de films déjà visionnés. 		

Tableau 5: Cas d'utilisation «Ajouter un film à la liste de films déjà visionnés»

ID :	CU203	
Nom :	Consulter les recommandations personnalisées	
Acteurs :	Membre, modérateur, administrateur	
Objectif :	Obtenir une liste de films recommandés par le système.	
Résumé :	En consultant la page d'accueil, le site génère une liste de films qui ne sont pas dans la liste et qui correspondent à certains critères.	
Type :	Désirable	
Références :	getRecommandationsPersonnalisees (208)	
Pré-condition :	Être authentifié, Avoir une liste de films vus non vide	
Post-condition :	Aucune	
Interactions :	Aucune	
	Acteurs	Système
1.	Le membre ouvre sa page d'accueil.	
2.		Le système génère des recommandations et les envoie.
Scénarios alternatifs (Exceptions)		
E203		
a. Aucun film n'est recommandé.		

Tableau 6: Cas d'utilisation «Consulter les recommandations personnalisées»

ID :	CU204	
Nom :	Créer la fiche d'un film	
Acteurs :	Membre, modérateur, administrateur	
Objectif :	Permet à un membre d'ajouter un nouveau film dans la liste dans la base de données.	
Résumé :	Un formulaire qu'il doit remplir en totalité. Quand il a terminé, il clique sur le bouton «Ajouter». Le film est ensuite ajouté à la base de données.	
Type :	Essentiel	
Références :	getAjouterFilm (CO209), ajouterFilm (210)	
Pré-condition :	Être authentifié	
Post-condition :	Tous les champs du formulaire sont remplis Le film a été ajouté dans la base de données.	
Interactions :	Aucune	
	Acteurs	Système
1.	Clique sur le bouton «ajouter film»	
2.		Affiche un formulaire à l'utilisateur
3.	Remplis le formulaire et clique le bouton Ajouter	
4.		Les données sont transformées en objets puis stockées dans la base de données
Scénarios alternatifs (Exceptions)		
E204		
a. Si l'utilisateur n'a pas rempli tous les champs, l'utilisateur reçoit un message d'erreur l'invitant à finir de remplir le formulaire.		

Tableau 7: Cas d'utilisation «Créer la fiche d'un film»

ID :	CU205	
Nom :	Rechercher un film	
Acteurs :	Membre, modérateur, administrateur	
Objectif :	Permettre de trouver un film particulier à partir de critère.	
Résumé :	Le membre peut remplir différentes informations qu'ils souhaitent afin de trouver un film. Les informations laissées vides sont ignorées. Le système lui affiche en retour les films correspondant au critère.	
Type :	Désirable	
Références :	getRechercheFilm (CO211), getFilm (CO212)	
Pré-condition :	Être authentifié	
Post-condition :	Au moins 1 film correspondant aux critères de recherche	
Interactions :	Aucune	
	Acteurs	Système
1.	Clique sur le bouton «Rechercher un film»	
2.		Envoie un formulaire avec la liste des critères
3.	Remplis le formulaire et envoie le formulaire	
4.		Envoie une liste de film correspondant
Scénarios alternatifs (Exceptions)		
E205		
<ul style="list-style-type: none"> a. Le système invite l'utilisateur a réduire son nombre de critère et informe l'utilisateur que le film ne se trouve peut-être pas dans la base de données. 		

Tableau 8: Cas d'utilisation «Rechercher un film»

ID :	CU206	
Nom :	Consulter la fiche d'un film	
Acteurs :	Membre, modérateur, administrateur	
Objectif :	Permet à un utilisateur de consulter la fiche de film afin de prendre connaissance des informations du film.	
Résumé :	L'utilisateur clique sur le titre d'un film. Le système affiche la fiche du film en retour.	
Type :	Essentiel	
Références :	getFilm (CO213)	
Pré-condition :	Être authentifié	
Post-condition :	Aucune	
Interactions :	Est étendu par : «Critiquer un film» (CU209) «Éditer une note donnée à un film» (CU210) «Rapporter un abus sur un film» (CU212) «Rapporter un abus sur une critique» (CU213) «Éditer la fiche d'un film» (CU207) «Ajouter un film à la liste de films déjà visionnés» (CU202) «Restaurer une version précédente de la fiche d'un film» (CU307) «Rapporter un abus d'un membre» (CU215)	
	Acteurs	Système
1.	Consulte la page d'un film	
2.		Envoie la vue du film avec les recommandations mises à jour.
Scénarios alternatifs (Exceptions)		

Tableau 9: Cas d'utilisation «Consulter la fiche d'un film»

ID :	CU207	
Nom :	Éditer la fiche d'un film	
Acteurs :	Membre, modérateur, administrateur	
Objectif :	En consultant la fiche d'un film, un membre peut éditer sa fiche.	
Résumé :	L'écran d'édition est un formulaire pré-rempli avec les informations existantes. Le membre peut modifier les informations puis cliquer sur «confirmer».	
Type :	Essentiel	
Références :	getEditerFilm (CO214), getEditerFilm (CO215)	
Pré-condition :	Être authentifié Le film n'est pas protégé	
Post-condition :	Aucune	
Interactions :	Étend : «Consulter la fiche d'un film» (CU206)	
	Acteurs	Système
1.	Consulte la page d'un film et clique sur le bouton éditer	
2.		Récupère les informations sur le film et envoie un formulaire pré-rempli à l'utilisateur
3.	Modifie les informations et confirme les modifications	
4.		Fait une sauvegarde des anciennes informations dans le serveur et modifie les informations
Scénarios alternatifs (Exceptions)		

Tableau 10: Cas d'utilisation «Éditer la fiche d'un film»

ID :	CU209	
Nom :	Critiquer un film	
Acteurs :	Membre, modérateur, administrateur	
Objectif :	Ajouter une critique par un membre pour un film	
Résumé :	Le membre clique sur l'option permettant de donner une critique au film qu'il est en train de consulter, entre ensuite la critique voulue et soumet le tout.	
Type :	Optionnel	
Références :	getAjoutCritique (CU205), setCritique (CO206)	
Pré-condition :	Être authentifié Le film critiqué doit être dans la liste de films déjà visionnés	
Post-condition :	La version de la base de données est correctement restaurée	
Interactions :	Étend «Consulter la fiche d'un film» (CU206)	
	Acteurs	Système
1.	Clique sur l'option permettant de donner une critique au film	
2.		Ouvre une boîte permettant d'entrer une critique
3.	Entre une critique et soumet	
4.		Ajoute la critique par le membre pour le film voulue dans la base de données
Scénarios alternatifs (Exceptions)		
E209		
a. Le film a déjà été critiqué par ce membre.		

Tableau 11: Cas d'utilisation «Critiquer un film»

ID :	CU210	
Nom :	Éditer une note donnée à un film	
Acteurs :	Membre, modérateur, administrateur	
Objectif :	Changer la note donnée à un film	
Résumé :	Le membre clique sur l'option permettant de changer la note donnée au film et donne la nouvelle note voulue.	
Type :	Optionnel	
Références :	getModifierNote (CO203), setNote (CO204)	
Pré-condition :	Être authentifié Le film a déjà une note donnée par le membre qui veut modifier la note	
Post-condition :	La note du film est la nouvelle note donnée par le membre	
Interactions :	Étend «Consulter la fiche d'un film» (CU206)	
	Acteurs	Systeme
1.	Clique sur l'option pour changer la note du film	
2.		Met la note en mode édition
3.	Donne une nouvelle note et soumet	
4.		Met à jour la note dans la base de données
Scénarios alternatifs (Exceptions)		
E210		
a. Le film n'a pas encore de note.		

Tableau 12: Cas d'utilisation «Éditer une note donnée à un film»

ID :	CU211	
Nom :	Éditer une critique donnée à un film	
Acteurs :	Membre, modérateur, administrateur	
Objectif :	Changer la critique donnée par le membre	
Résumé :	Le membre clique sur l'option permettant de changer la critique donnée au film et donne la nouvelle critique voulue.	
Type :	Optionnel	
Références :	setCritique (CO206), getModificationCritique (CO207)	
Pré-condition :	Être authentifié Être sur la fiche d'un film	
Post-condition :	La nouvelle critique du film correspond à celle voulue par le membre qui l'a modifié.	
Interactions :	Étend «Consulter la fiche d'un film» (CU206)	
	Acteurs	Système
1.	Clique sur l'option pour changer la critique du film	
2.		Met la critique en mode édition
3.	Donne une nouvelle critique et soumet	
4.		Met à jour la critique dans la base de données
Scénarios alternatifs (Exceptions)		
E211		
a. Le film n'a pas encore de critique donnée par ce membre.		

Tableau 13: Cas d'utilisation «Éditer une critique donnée à un film»

ID :	CU212	
Nom :	Rapporter un abus sur un film	
Acteurs :	Membre, modérateur, administrateur	
Objectif :	Permet à un membre de signaler au modérateur la présence d'une infraction au règlement dans la fiche d'un film.	
Résumé :	Lors de la consultation de la fiche d'un film, un membre peut cliquer sur le bouton «signaler». Le film sera enregistré dans la liste des films à modérer.	
Type :	Essentiel	
Références :	signalerFilm (CO216)	
Pré-condition :	Être authentifié Être sur la fiche d'un film	
Post-condition :	Aucune	
Interactions :	Étend «Consulter la liste d'abus» (CU301)	
	Acteurs	Système
1.	Clique sur le bouton «signaler»	
2.		Un lien vers le film est ajouté à la liste des films à modérer.
Scénarios alternatifs (Exceptions)		

Tableau 14: Cas d'utilisation «Rapporter un abus sur un film»

ID :	CU213	
Nom :	Rapporter un abus sur une critique	
Acteurs :	Membre, modérateur, administrateur	
Objectif :	Permet à un membre de signaler au modérateur la présence d'une infraction au règlement dans une critique.	
Résumé :	Lors de la consultation d'une critique, un membre peut cliquer sur le bouton «signaler». La critique sera enregistrée dans la liste des critiques à modérer.	
Type :	Optionnel	
Références :	signalerCritique (CO217)	
Pré-condition :	Être authentifié Être sur la fiche d'un film	
Post-condition :	Aucune	
Interactions :	Étend «Consulter la fiche d'un film» (CU206)	
	Acteurs	Système
1.	Clique sur le bouton «signaler»	
2.		Un lien vers la critique est ajouté à la liste des films à modérer.
Scénarios alternatifs (Exceptions)		

Tableau 15: Cas d'utilisation «Rapporter un abus sur une critique»

ID :	CU214	
Nom :	Consulter un profil public	
Acteurs :	Membre, modérateur, administrateur	
Objectif :	Consulter le profil d'un membre	
Résumé :	Rechercher le profil d'un membre et afficher ses informations publiques.	
Type :	Essentiel pour la modération et optionnel pour la consultation par les membres	
Références :	getRechercherProfilPublique (CO218), rechercherProfil (CO219), getProfilPublique (CO220)	
Pré-condition :	Être authentifié	
Post-condition :	Aucune	
Interactions :	Est étendu par : «Rapporter un abus d'un membre» (CU215) «Suspendre un membre» (CU303) «Avertir un membre» (CU304) «Bannir un membre» (CU401) «Promouvoir un membre» (CU402)	
	Acteurs	Systeme
1.	Clique sur rechercher profil public.	
2.		Renvoie la page de recherche de profil public.
3.	Rechercher un membre avec son nom utilisateur.	
4.		Afficher le résultat de la recherche
5.	Cliquer sur l'utilisateur	

	recherché	
6.		Renvoie la page de profil de l'utilisateur
Scénarios alternatifs (Exceptions)		

Tableau 16: Cas d'utilisation « Consulter un profil publique »

ID :	CU215	
Nom :	Rapporter un abus d'un membre	
Acteurs :	Membre, modérateur, administrateur	
Objectif :	Permet à l'acteur de signaler à la modération la présence d'une infraction au règlement faite par un membre	
Résumé :	Signaler à la modération la présence d'une infraction au règlement faite par un membre	
Type :	Optionnel	
Références :	signalerFilm (CO216)	
Pré-condition :	Être authentifié	
Post-condition :	Aucune	
Interactions :	Étend «Consulter un profil publique» (CU214)	
	Acteurs	Systeme
1.	Cliquer sur le bouton signaler a côté du nom d'utilisateur	
2.		Affiche un formulaire de rapport d'abus
3.	Écrit pourquoi il rapporte ce membre pour modération	
4.	Envoie le rapport	
5.		Un lien vers le membre est ajouté à la liste des membres à modérer.
Scénarios alternatifs (Exceptions)		

Tableau 17: Cas d'utilisation «Rapporter un abus d'un membre»

ID :	CU216	
Nom :	Consulter son compte	
Acteurs :	Membre, modérateur, administrateur	
Objectif :	Afficher les informations relatives à son compte.	
Résumé :	Il accède à sa page de compte où il dispose de champs d'informations personnelles qu'il peut modifier.	
Type :	Essentiel	
Références :	getCompte (CO223)	
Pré-condition :	Être authentifié	
Post-condition :	Aucune	
Interactions :	Est étendu par : «Éditer son compte» (CU217)	
	Acteurs	Systeme
1.	Accède à son compte	
2.		Affiche les informations relatives à son compte
Scénarios alternatifs (Exceptions)		

Tableau 18: Cas d'utilisation «Consulter son compte»

ID :	CU217	
Nom :	Éditer son compte	
Acteurs :	Membre, modérateur, administrateur	
Objectif :	Être capable de modifier les informations relatives à son compte.	
Résumé :	Il accède à sa page de compte où il dispose de champs d'informations personnelles qu'il peut modifier. Il peut valider les modifications qu'il souhaite effectuer ou les annuler en cliquant sur le bouton correspondant.	
Type :	Essentiel	
Références :	getModificationCompte (CO224), setCompte (CO225)	
Pré-condition :	Être authentifié Consulter son compte	
Post-condition :	Aucune	
Interactions :	Étend «Consulter son compte» (CU216)	
	Acteurs	Systeme
1.	Clique sur "modifier compte"	
2.		Envoie le contenu des champs afin de modifier les informations du compte
3.	Entre les nouvelles informations et sauvegarde	
4.		Sauvegarde les informations et affiche le profil du compte avec les nouvelles informations
Scénarios alternatifs (Exceptions)		
E217		
a. Un champ n'est pas saisi correctement		

- b. Cliquer sur annuler**
- c. Ne pas sauvegarder les informations**

Tableau 19: Cas d'utilisation «Éditer son compte»

ID :	CU218	
Nom :	Envoyer un message privé à un membre	
Acteurs :	Membre, modérateur, administrateur	
Objectif :	Envoyer un message privé à un ou plusieurs membres qui recevront ce message dans leurs listes respectives de messages privés reçus	
Résumé :	Le membre crée un nouveau message privé où il indique le ou les destinataires, le sujet et le corps du message, puis l'expédie	
Type :	Optionnel	
Références :	getCreerMessage (CO226), rechercherDestinataires (CO227), postMessage (CO228)	
Pré-condition :	Être authentifié	
Post-condition :	Le message à envoyer se trouve dans la liste des messages privés envoyés de l'expéditeur et dans la liste des messages privés reçus de chacun des destinataires du message	
Interactions :	Aucune	
	Acteur	Système
1.	Clique sur "créer un nouveau message"	
2.		Affiche la page de création du nouveau message
3.	Entre le corps du message, le sujet, puis clique sur "ajouter des expéditeurs"	
4.		Affiche la liste des membres
5.	Sélectionne les membres auxquels il souhaite envoyer le	

	message, puis clique sur “Ajouter” et cliquer sur envoyer	
6.		Ajoute le message à envoyer dans la liste des messages privés envoyés de l’expéditeur et dans la liste des messages privés reçus de chacun des destinataires du message
Scénarios alternatifs (Exceptions)		
E218 <ol style="list-style-type: none"> a. Le membre clique sur “envoyer un message privé” sur le profil public d’un membre, auquel cas le message créé contiendra automatiquement comme destinataire le membre auquel appartient le profil public à partir duquel le message a été créé b. Le membre clique sur “envoyer” sans avoir au préalable indiqué de destinataires 		

Tableau 20: Cas d'utilisation «Envoyer un message privé à un membre»

ID :	CU219	
Nom :	Consulter la liste des messages privés reçus	
Acteurs :	Membre, modérateur, administrateur	
Objectif :	Afficher la liste des messages reçus par le membre en question	
Résumé :	Le système affiche la liste des messages reçus par le membre après qu'il ait cliqué sur "Messages reçus"	
Type :	Optionnel	
Références :	getMessagesRecus (CO229),	
Pré-condition :	Être authentifié	
Post-condition :	Aucune	
Interactions :	Est étendu par : «Supprimer un message privé» (CU222) «Lire un message privé» (CU221)	
	Acteurs	Système
1.	Clique sur "Messages reçus"	
2.		Génère puis envoie la liste des messages reçus par le membre en question
Scénarios alternatifs (Exceptions)		

Tableau 21: Cas d'utilisation «Consulter la liste des messages privés reçus»

ID :	CU220	
Nom :	Consulter la liste des messages privés envoyés	
Acteurs :	Membre, modérateur, administrateur	
Objectif :	Afficher la liste des messages envoyés par le membre en question	
Résumé :	Le système affiche la liste des messages envoyés par le membre après qu'il ait cliqué sur "Messages envoyés"	
Type :	Optionnel	
Références :	getMessagesEnvoyes (CO230)	
Pré-condition :	Être authentifié	
Post-condition :	Aucune	
Interactions :	Est étendu par : «Supprimer un message privé» (CU222) «Lire un message privé» (CU221)	
	Acteurs	Système
1.	Clique sur "Messages envoyés"	
2.		Génère puis envoie la liste des messages envoyés par le membre en question
Scénarios alternatifs (Exceptions)		

Tableau 22: Cas d'utilisation «Consulter la liste des messages privés envoyés»

ID :	CU221	
Nom :	Lire un message privé	
Acteurs :	Membre, modérateur, administrateur	
Objectif :	Accéder au contenu d'un message privé	
Résumé :	Suite à l'accès à la liste des messages reçus ou envoyés, le membre sélectionne un de ces messages pour accéder à son contenu	
Type :	Optionnel	
Références :	getMessage (CO231)	
Pré-condition :	Être authentifié	
Post-condition :	Le contenu du message doit être affiché dans son intégralité	
Interactions :	Étend «Consulter la liste des messages privés reçus» (CU219) «Consulter la liste des messages privés envoyés» (CU220)	
	Acteurs	Système
1.	Clique sur le message contenu dans la liste des messages affichés	
2.		Envoie la page comportant le contenu du message sélectionné
Scénarios alternatifs (Exceptions)		

Tableau 23: Cas d'utilisation «Lire un messages privés»

ID :	CU222	
Nom :	Supprimer un message privé	
Acteurs :	Membre, modérateur, administrateur	
Objectif :	Le message supprimé ne doit plus se retrouver dans la liste des messages du membre qui l'a supprimé	
Résumé :	Le membre sélectionne le message à supprimer puis clique sur "Supprimer", ce qui conduit au retrait de ce message de la liste des messages du membre qui l'a supprimé	
Type :	Optionnel	
Références :	supprimerMessage (CO232)	
Pré-condition :	Être authentifié Consulter la liste des messages privés reçus ou envoyés	
Post-condition :	Le message supprimé doit être absent de la liste des messages reçus ou envoyés selon celle à partir de laquelle ce message a été supprimé	
Interactions :	Étend : «Consulter la liste des messages privés reçus» (CU219) «Consulter la liste des messages privés envoyés» (CU220)	
	Acteurs	Système
1.	Sélectionne le message à supprimer puis clique sur "Supprimer"	
2.		Retire le message supprimé de la liste des messages du membre qui l'a supprimé et envoie la page précédente (liste des messages reçus ou envoyés)
Scénarios alternatifs (Exceptions)		

Tableau 24: Cas d'utilisation «Supprimer un messages privés»

ID :	CU301	
Nom :	Consulter la liste d'abus	
Acteurs :	Modérateur, administrateur	
Objectif :	Accéder à la liste d'abus pour s'informer s'il y a de la modération à faire sur le site	
Résumé :	Le modérateur accède à la page de la liste d'abus	
Type :	Essentiel	
Références :	getListeAbus (CO301)	
Pré-condition :	Être authentifié	
Post-condition :	Aucune	
Interactions :	Est étendu par : «Retirer un abus de la liste d'abus» (CU302)	
	Acteurs	Systeme
1.	Clique sur le lien de la page de liste d'abus disponible sur toute les pages du site	
2.		Renvoie la page avec tous les abus actifs sur le site
Scénarios alternatifs (Exceptions)		

Tableau 25: Cas d'utilisation «Consulter la listed'abus»

ID :	CU302	
Nom :	Retirer un abus de la liste d'abus	
Acteurs :	Modérateur, administrateur	
Objectif :	Retirer de la liste un abus qui a été résolu pour en informer les autres modérateurs	
Résumé :	Un modérateur note un abus comme résolu	
Type :	Essentiel	
Références :	getResoudreAbus (CO302)	
Pré-condition :	Être authentifié	
Post-condition :	Aucune	
Interactions :	Étend «Consulter la liste d'abus» (CU301)	
	Acteurs	Système
1.	Le modérateur clique sur le bouton "Résolu"	
2.		Le système revoit le formulaire de résolution
3.	Le modérateur remplit le formulaire (problème, mesure prise).	
4.		Le système archive le(s) formulaire(s) d'abus avec le formulaire du modérateur
Scénarios alternatifs (Exceptions)		
E302		
<ul style="list-style-type: none"> a. Le nom donné doit être valide. b. Tous les champs doivent être remplis. 		

Tableau 26: Cas d'utilisation «Retirer un abus de la liste d'abus»

ID :	CU303	
Nom :	Suspendre un membre	
Acteurs :	Modérateur	
Objectif :	Suspendre un membre pour une durée limité	
Résumé :	Le modérateur accède au profil d'un membre ce qui lui permet de le suspendre	
Type :	Essentiel	
Références :	getSuspendreMembre (CO304), postAvertissementMembre (CO307), sauvegarderFormulaireSuspension (CO308)	
Pré-condition :	Être authentifié	
Post-condition :	Aucune	
Interactions :	Étend «Consulter un profil public» (CU214)	
	Acteurs	Systeme
1.	Clique sur le bouton suspendre qui n'est visible qu'aux modérateurs	
2.		Renvoie le formulaire de suspension et les précédents formulaires de suspension
3.	Remplit le formulaire de suspension (Durée, raison)	
4.		Archive le formulaire et empêche le membre de se connecter pour la durée de la suspension
Scénarios alternatifs (Exceptions)		
E303		
a. Si le membre est un modérateur ou un administrateur, le bouton suspendre n'est pas affiché.		

- b. Si le membre est déjà suspendu, le bouton est non cliquable et affiche “Membre déjà suspendu”.**
- c. Si le membre est déjà banni, le bouton est non cliquable et affiche “Membre banni”.**

Tableau 27: Cas d'utilisation «Suspendre un membre»

ID :	CU304	
Nom :	Avertir un membre	
Acteurs :	Modérateur, administrateur	
Objectif :	Avertir un membre dont le comportement n'est pas acceptable	
Résumé :	Le modérateur accède au profil d'un membre ce qui lui permet l'envoi d'un message d'avertissement	
Type :	Optionnel	
Références :	postMessage (CO228), getSuspendreMembre (CO304), postAvertissementMembre (CO305), sauvegarderFormulaireAvertissement (CO306)	
Pré-condition :	Être authentifié	
Post-condition :	Aucune	
Interactions :	Étend «Consulter un profil public» (CU214)	
	Acteurs	Système
1.	Clique sur le bouton avertir qui n'est visible qu'aux modérateurs	
2.		Renvoie le formulaire d'avertissement et les précédant formulaire d'avertissement
3.	Remplit le formulaire d'avertissement (raison)	
4.		Archive le formulaire
Scénarios alternatifs (Exceptions)		
E304		
a. Si le membre est un modérateur ou un administrateur, le bouton		

avertir n'est pas affiché.

- b. Si le membre est déjà suspendu, le bouton est non cliquable et affiche "Membre déjà suspendue".**
- c. Si le membre est déjà banni, le bouton est non cliquable et affiche "Membre banni".**

Tableau 28: Cas d'utilisation «Avertir un membre»

ID :	CU305	
Nom :	Censurer une critique	
Acteurs :	Modérateur, administrateur	
Objectif :	Censurer une critique pour qu'elle soit conforme aux règlements du site web	
Résumé :	Similaire à l'édition d'une critique mais faite sur la critique d'un autre utilisateur	
Type :	Optionnel	
Références :	getCensurerCritique (CO309), setCensureCritique (CO310)	
Pré-condition :	Être authentifié	
Post-condition :	Aucune	
Interactions :	Aucune	
	Acteurs	Systeme
1.	Cliquer sur le bouton "Censurer" là ou devrait être le bouton éditer lorsqu'on est auteur	
2.		Renvoie le formulaire d'édition des modérateurs
3.	Sélectionne le texte à censurer	
4.		Modifie le texte affiché de façon à ce que les changements apparaissent comme ceci [Le texte a été modifié de façon à être conforme aux règles du site]
Scénarios alternatifs (Exceptions)		
E305		

a. Si sélection du texte en entier le système revoit le formulaire de suppression de critique.

Tableau 29: Cas d'utilisation «Censurer une critique»

ID :	CU306	
Nom :	Supprimer une critique	
Acteurs :	Modérateur, administrateur	
Objectif :	Supprimer la critique d'un membre de la page d'un film	
Résumé :	Similaire à la suppression d'une critique par un membre mais la critique est gardée pour conserver la note	
Type :	Optionnel	
Références :	getSupprimerCritique (CO311), setSuppressionCritique (CO312)	
Pré-condition :	Être authentifié	
Post-condition :	Aucune	
Interactions :	Aucune	
	Acteurs	Système
1.	Cliquer sur le bouton "Supprimer" qui n'est visible qu'à ceux qui ont les droit de modération	
2.		Renvoi le formulaire de suppression de critique
3.	Remplit le formulaire de suppression de critique (Raison, auteur)	
4.		Archive le formulaire
5.		Affiche à la place du texte "La critique a dû être supprimée puisqu'elle n'était pas conforme aux règles du site web"

Scénarios alternatifs (Exceptions)

Tableau 30: Cas d'utilisation «Supprimer une critique»

ID :	CU307	
Nom :	Restaurer une version précédente de la fiche d'un film	
Acteurs :	Modérateur, administrateur	
Objectif :	Effectuer un retour en arrière mais seulement pour une seule fiche de film	
Résumé :	Le modérateur effectue le retour en choisissant l'option	
Type :	Désirable	
Références :	getRestaurationFicheFilm (CO314), sauvegarderFicheFilm (CO315), restaurerFicheFilm (CO316)	
Pré-condition :	Être authentifié	
Post-condition :	La nouvelle fiche du film correspond à la version précédente voulue.	
Interactions :	Étend «Consulter la fiche d'un film» (CU206)	
	Acteurs	Système
1.	Sélectionner l'option au haut de la page pour restaurer la fiche du film	
2.		Revoit la liste des derniers 10 points de sauvegarde
3.	Sélectionne le point de sauvegarde désiré	
4.		Met une copie du point de sauvegarde comme étant le dernier point de sauvegarde
Scénarios alternatifs (Exceptions)		

Tableau 31: Cas d'utilisation «Restaurer une version précédente de la fiche d'un film»

ID :	CU308	
Nom :	Protéger un film contre les modifications	
Acteurs :	Modérateur, administrateur	
Objectif :	Protéger un film de façon à empêcher les modifications pour des raisons d'abus par exemple	
Résumé :	Le modérateur protéger la page en choisissant l'option	
Type :	Désirable	
Références :	getProtectionFicheFilm (CO317), protegerFicheFilm (CO318)	
Pré-condition :	Être authentifié	
Post-condition :	La fiche du film ne peut plus être modifiée pour une durée donnée.	
Interactions :	Étend «Consulter la fiche d'un film» (CU206)	
	Acteurs	Système
1.	Sélectionner l'option au haut de la page pour protéger contre les modifications	
2.		Renvoie le formulaire de demande de temps de protection
3.	Indique la durée de protection de la page	
4.		Empêche toute modification à la page pour la durée de temps donnée, sauf si l'éditeur a des droits de modération
Scénarios alternatifs (Exceptions)		
E308		
<ul style="list-style-type: none"> a. Si le modérateur soumet une période de temps inférieure à 1 heure, le système protège la page pour une heure. b. Remplacer le message de raison, facultatif. 		

Tableau 32: Cas d'utilisation «Protéger un film contre les modifications»

ID :	CU309	
Nom :	Supprimer un film	
Acteurs :	Modérateur, administrateur	
Objectif :	Le modérateur supprime le film en choisissant l'option correspondante	
Résumé :	Supprimer un film qui n'existe pas ou ne devrait pas se trouver sur la base de données	
Type :	Désirable	
Références :	getSuppressionFicheFilm (CO319), supprimerFicheFilm (CO320)	
Pré-condition :	Être authentifié	
Post-condition :	Le film a été supprimé de la base de données	
Interactions :	Étend «Consulter la fiche d'un film» (CO206)	
	Acteurs	Système
1.	Sélectionner l'option au haut de la page pour supprimer le film	
2.		Demande une confirmation au modérateur
3.	Confirme	
4.		Bloque tout accès à la page et enlève toute référence sur le site
Scénarios alternatifs (Exceptions)		

Tableau 33: Cas d'utilisation «Supprimer un film»

ID :	CU401	
Nom :	Bannir un membre	
Acteur :	Administrateur	
Objectif :	Restreindre définitivement l'usage du site au membre banni	
Résumé :	L'administrateur bannit le membre définitivement à partir de la page de profil de ce dernier.	
Type :	Essentiel	
Références :	bannirMembre (CO401)	
Pré-condition :	Être authentifié	
Post-condition :	Le membre ciblé est banni (il n'a plus accès à son compte).	
Interactions :	Étend «Consulter un profil public» (CU214)	
	Acteurs	Systeme
1.	Clique sur "Bannir ce membre" sur la page de profil du membre à bannir	
2.		Bannit le membre
Scénarios alternatifs (Exceptions)		

Tableau 34: Cas d'utilisation «Bannir un membre»

ID :	CU402	
Nom :	Promouvoir un membre	
Acteurs :	Administrateur	
Objectif :	Promouvoir un membre au rôle de modérateur ou d'administrateur	
Résumé :	L'administrateur promeut le membre au rang d'administrateur ou de modérateur à partir de la page de profil de ce dernier.	
Type :	Essentiel	
Références :	getPromouvoirMembre (CO402), setPromotion (CO403)	
Pré-condition :	Être authentifié	
Post-condition :	Le membre est promu au rang ciblé.	
Interactions :	Étend «Consulter un profil public» (CU214)	
	Acteurs	Système
1.	Clique sur “Promouvoir ce membre” sur la page de profil du membre à promouvoir	
2.		Envoie la page de promotion avec “Administrateur” ou “Modérateur” comme options de promotion selon le statut du membre à promouvoir
3.	Choisit l'option “Administrateur” ou “Modérateur” si disponible puis clique sur “Promouvoir”	
4.		Promeut le membre en question selon l'option sélectionnée

Scénarios alternatifs (Exceptions)
E402 a. Le bouton “Promouvoir ce membre” n’est pas disponible si le membre en question est un administrateur

Tableau 35: Cas d'utilisation «Promouvoir un membre»

ID :	CU501	
Nom :	Effectuer une restauration de la base de données	
Acteurs :	Administrateur de la base de données	
Objectif :	Restaurer la base de données à une version antérieure.	
Résumé :	Restaurer la base de données à une version antérieure à partir d'une liste de versions antérieures de la base de données	
Type :	Essentiel	
Références :	getVersionsBaseDeDonnees (CO501), restaurationBaseDeDonnees (CO502)	
Pré-condition :	Être authentifié	
Post-condition :	La base de donnée est rétablie à la version antérieure voulue.	
Interactions :	Aucune	
	Acteurs	Système
1.	Cliquer pour afficher la liste des versions antérieures de la base de données	
2.		Afficher la liste des versions antérieures de la base de données
3.	Cliquer sur la version antérieure désirée	
4.		La version de la base de données est correctement restaurée
Scénarios alternatifs (Exceptions)		
E501		
<ul style="list-style-type: none"> a. La liste est vide b. Une erreur au niveau du système de gestion de la base de données est 		

survenue

Tableau 36: Cas d'utilisation «Effectuer un erestauration de la base de données»

3.2 Spécifications fonctionnelles

3.2.1 Diagramme de séquences

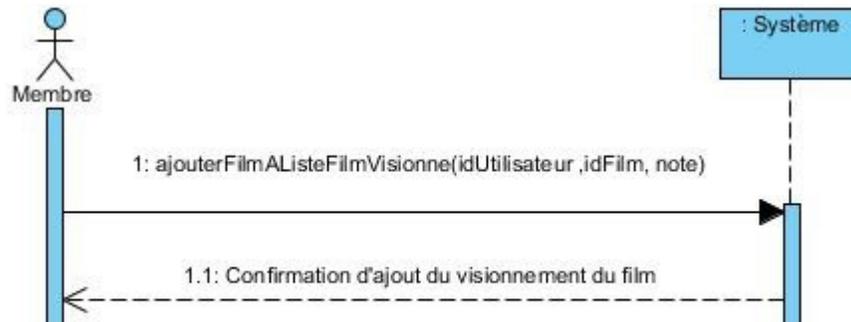


Figure 4: Diagramme de séquences : «Ajouter un film à la liste de films déjà visionnés»

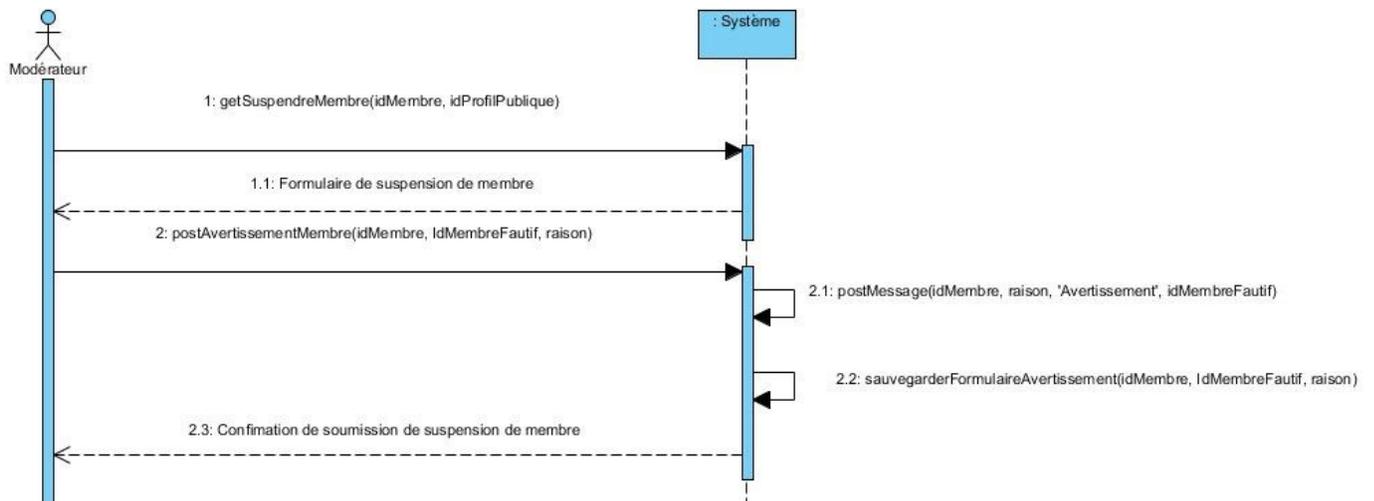


Figure 5: Diagramme de séquences : «Avertir un membre»

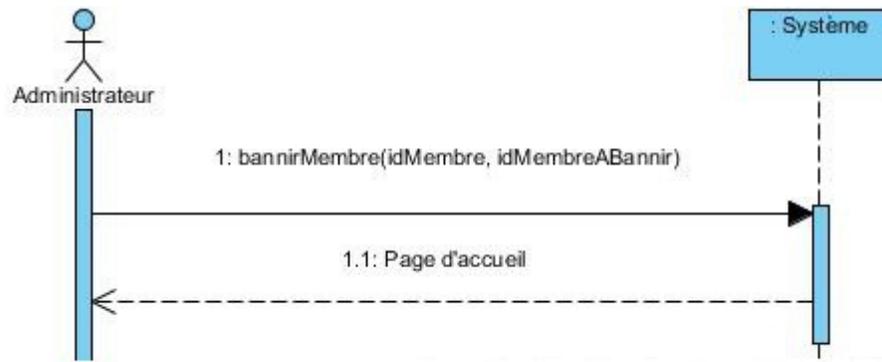


Figure 6: Diagramme de séquences : «Bannir un membre»

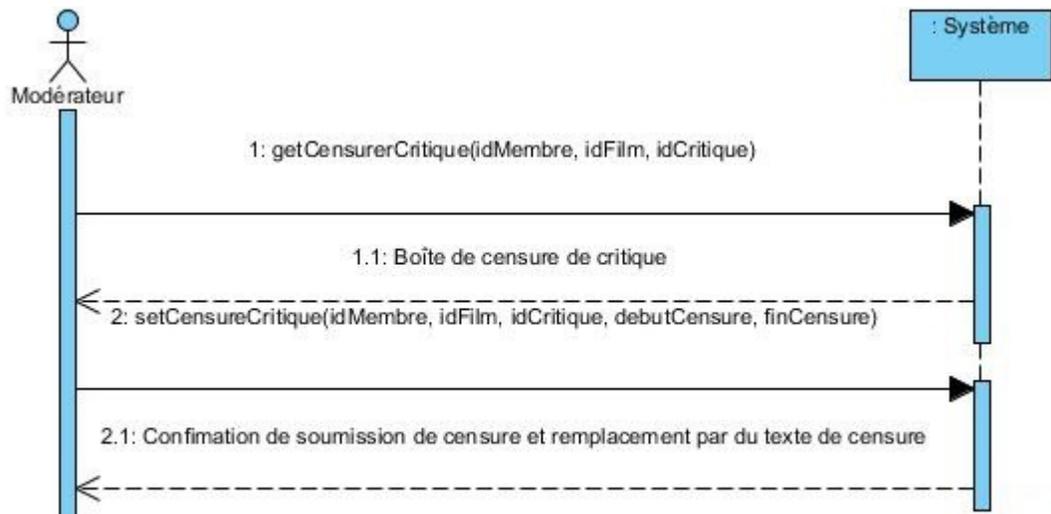


Figure 7: Diagramme de séquences : «Censurer une critique»

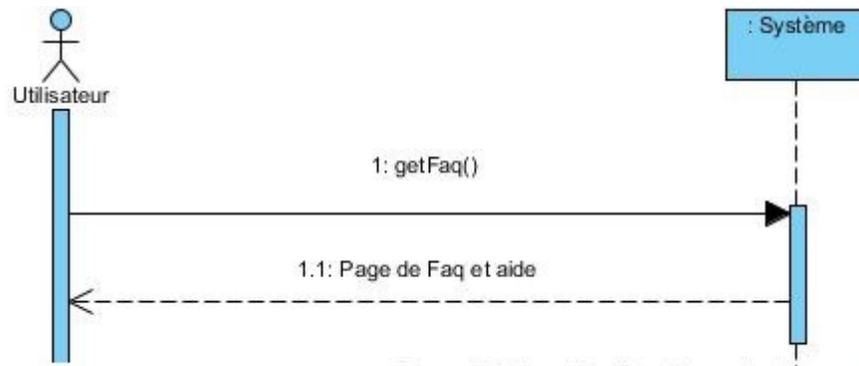


Figure 8: Diagramme de séquences : «Consulter aide et FAQ»

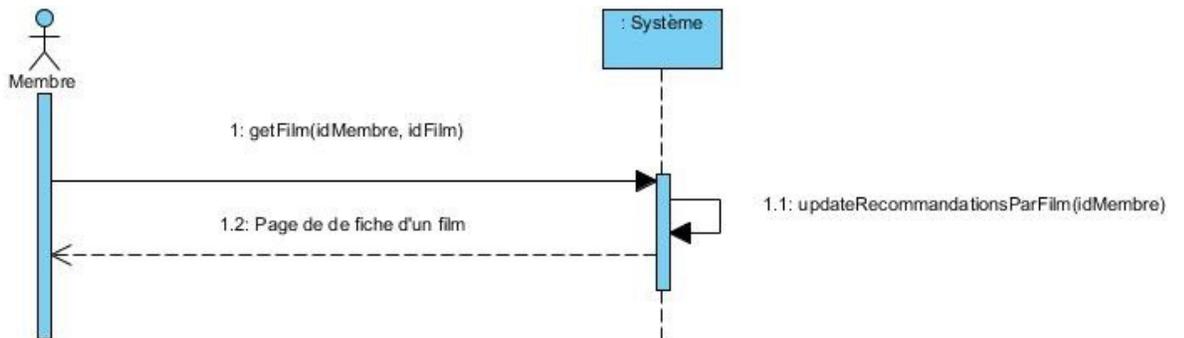


Figure 9: Diagramme de séquences : «Consulter la fiche d'un film»

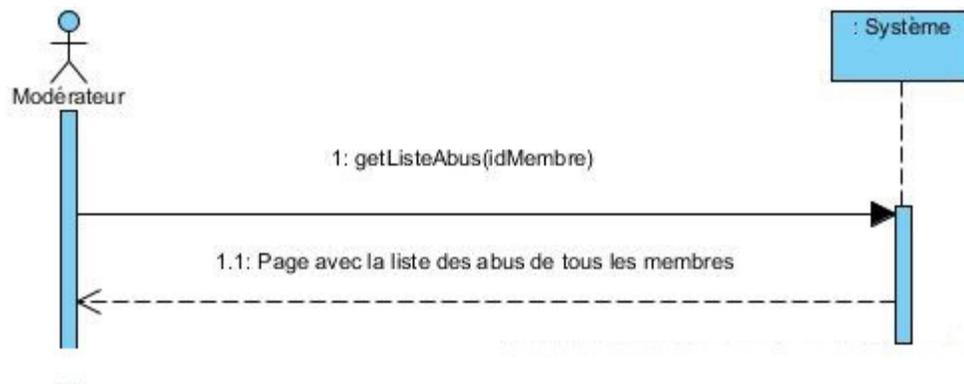


Figure 10: Diagramme de séquences : «Consulter la liste d'abus»

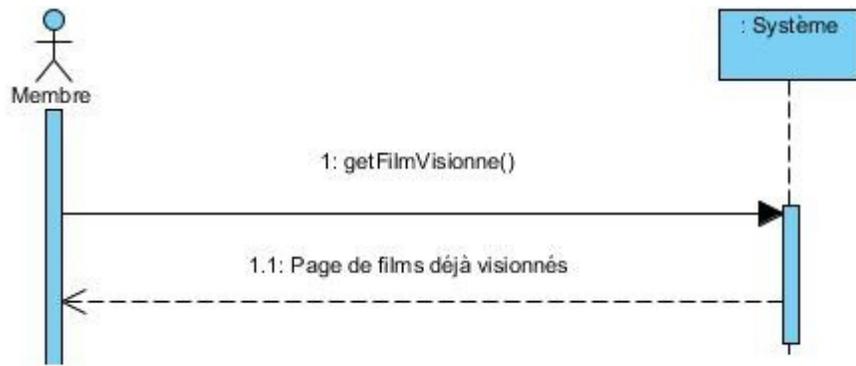


Figure 11: Diagramme de séquences : « Consulter la liste de films déjà visionnés »

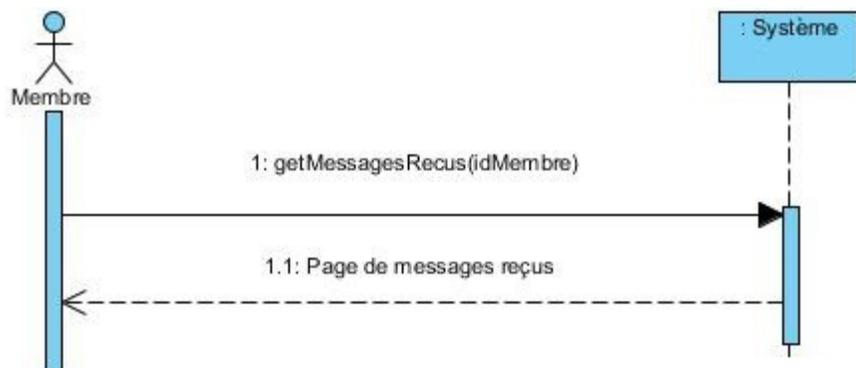


Figure 12: Diagramme de séquences : « Consulter la liste de message privés reçus »

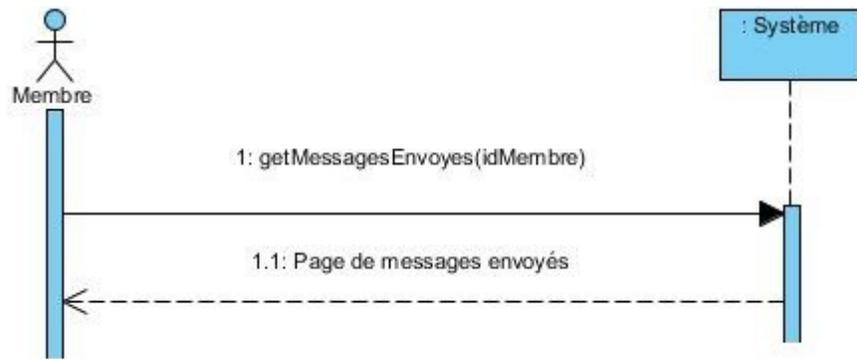


Figure 13: Diagramme de séquences : «Consulter la liste de messages privés envoyés»

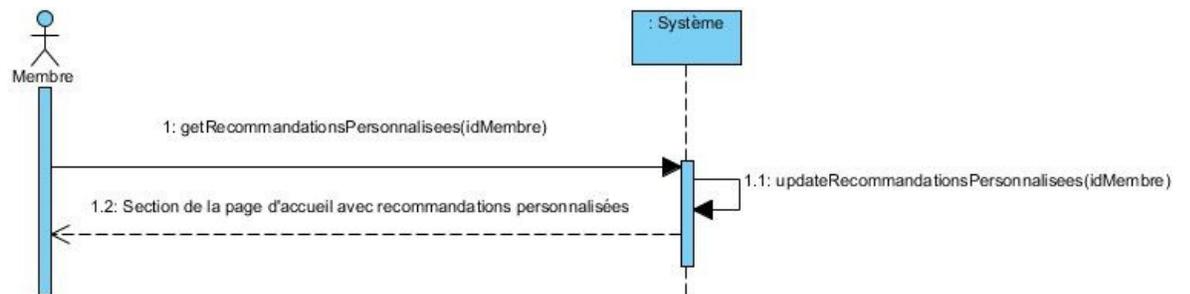


Figure 14: Diagramme de séquences : «Consulter recommandations personnalisées»

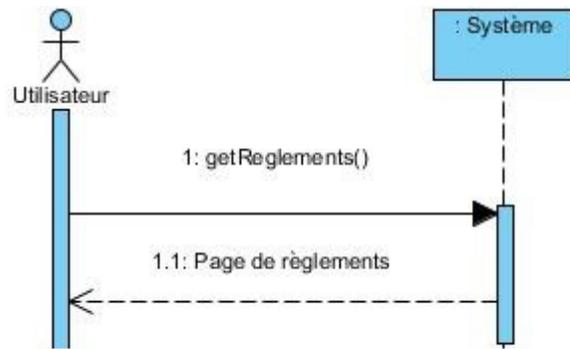


Figure 15: Diagramme de séquences :
«Consulter règlements»

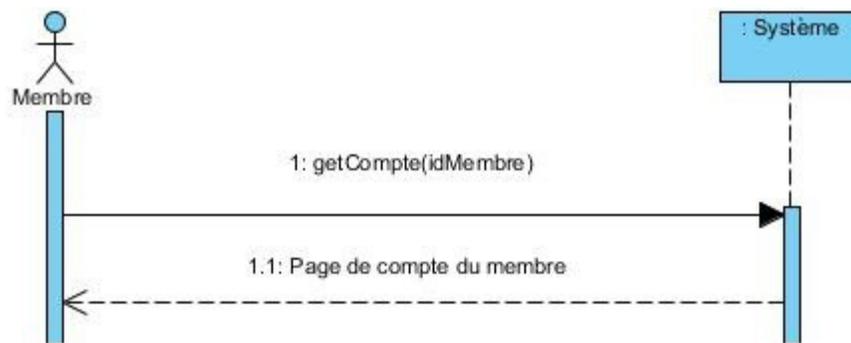


Figure 16: Diagramme de séquences : «Consulter son compte»

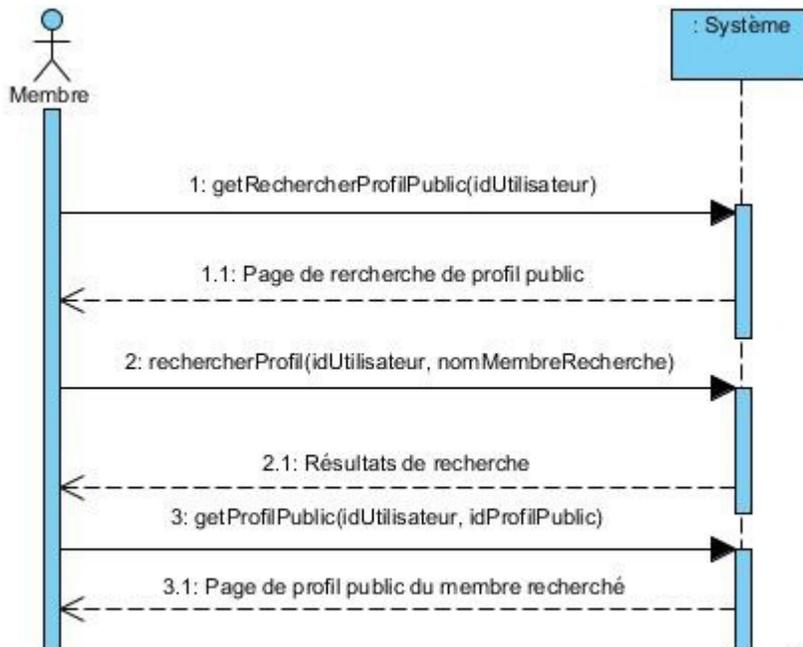


Figure 17: Diagramme de séquences : « Consulter un profil public »

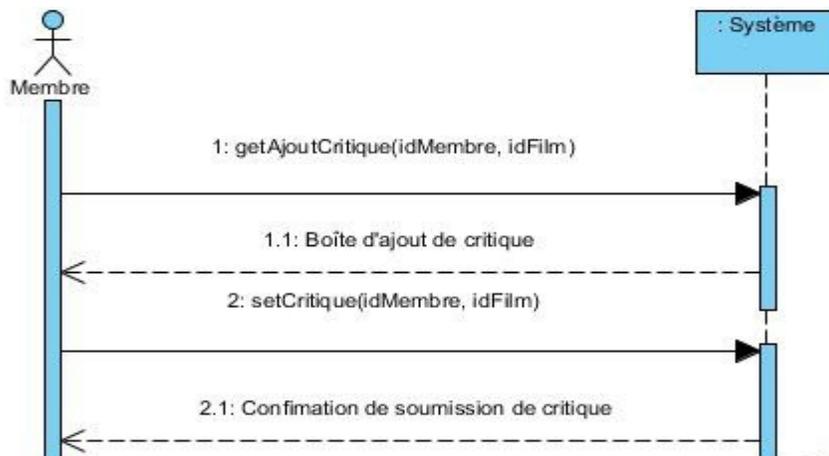


Figure 18: Diagramme de séquences : « Critiquer un film »

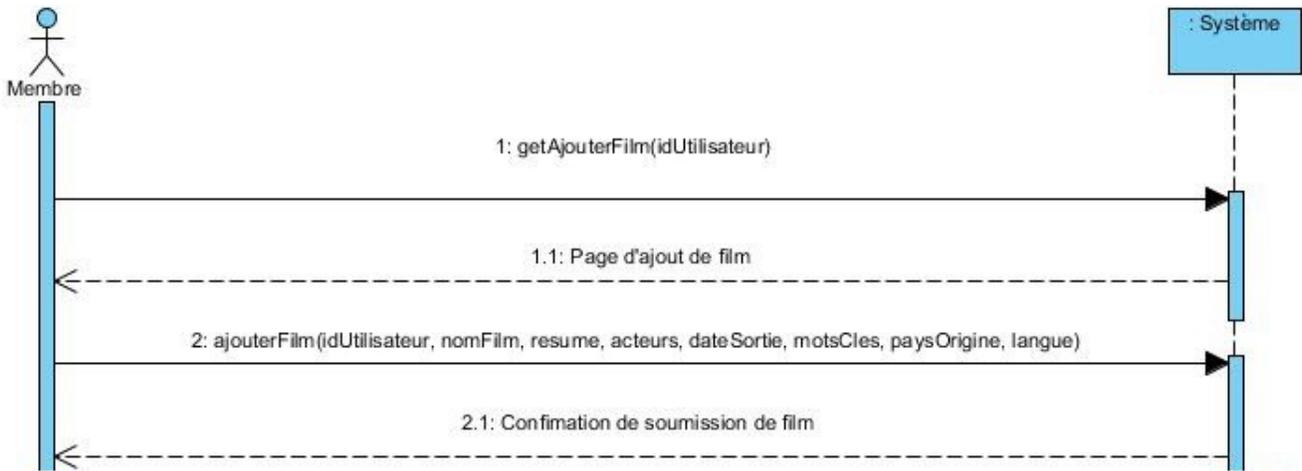


Figure 19: Diagramme de séquences : «Créer la fiche d'un film»

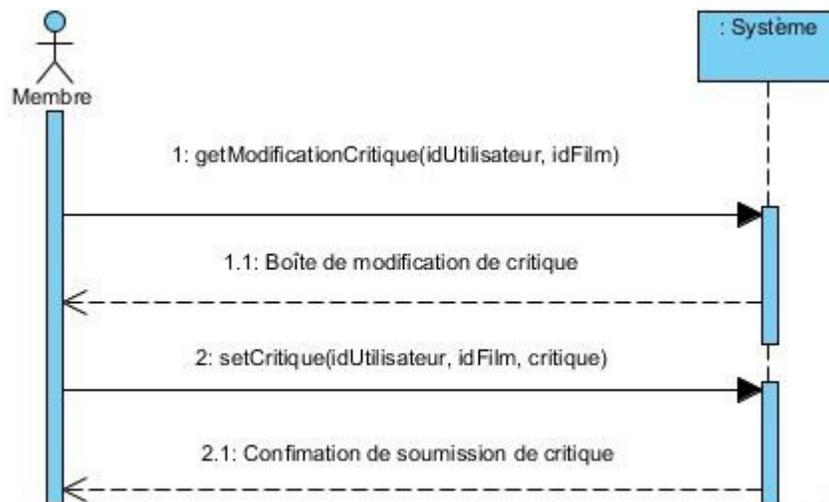


Figure 20: Diagramme de séquences : «Éditer une critique donnée à un film»

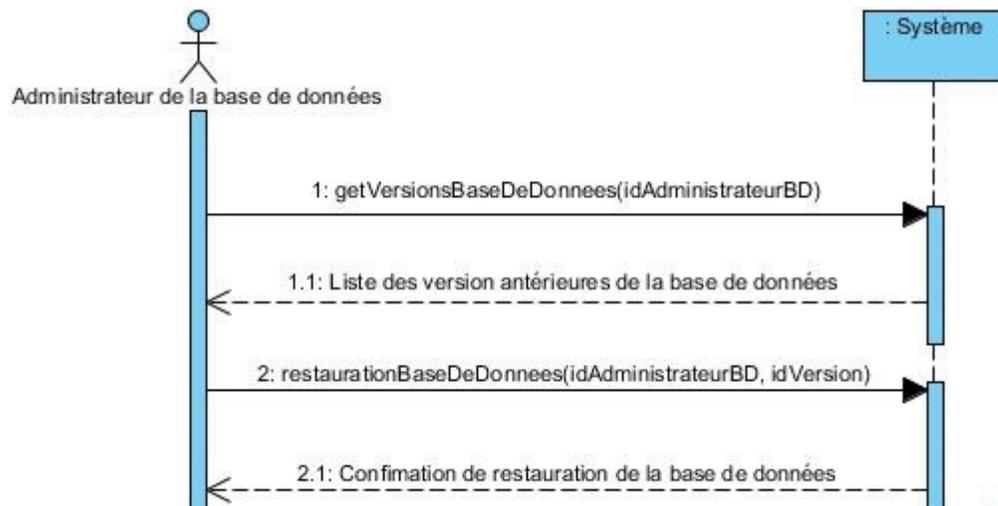


Figure 21: Diagramme de séquences : «Effectuer une restauration de la base de données»

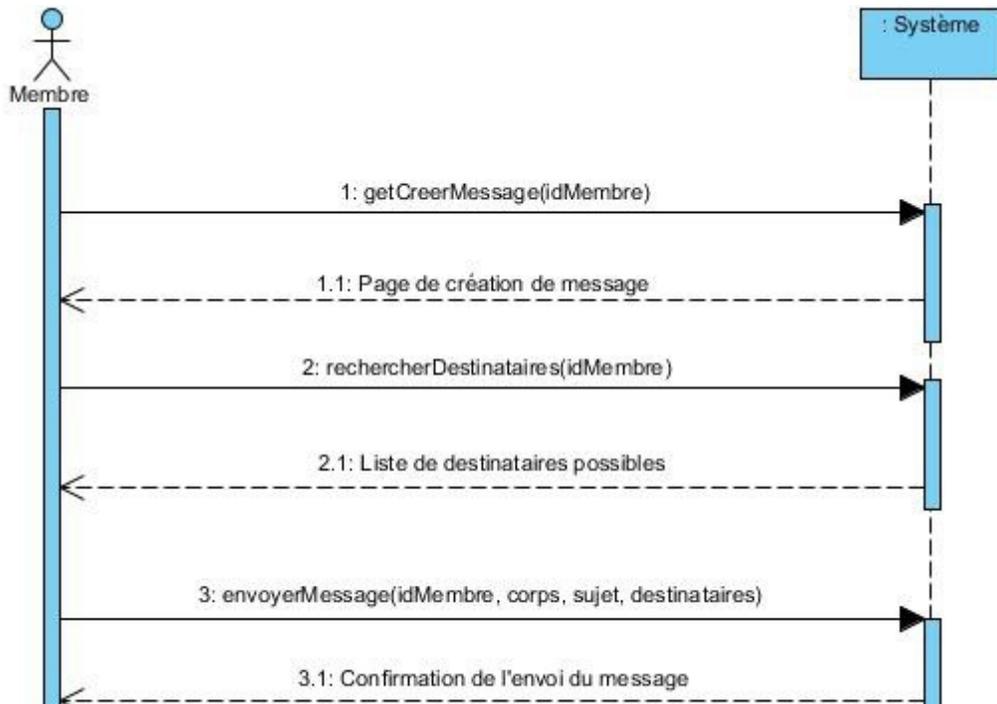


Figure 22: Diagramme de séquences : «Envoyer un message privé à un membre»

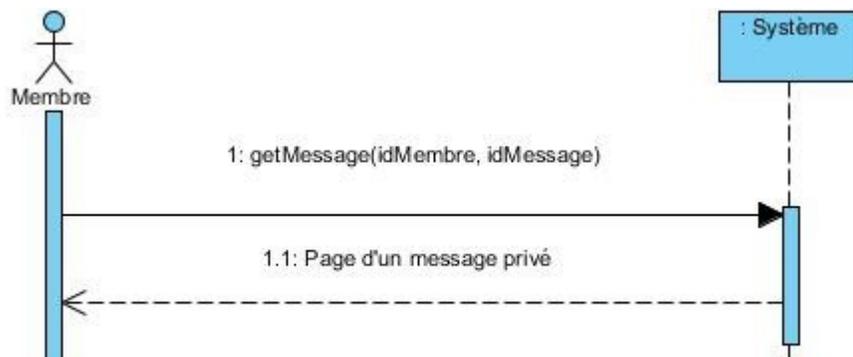


Figure 23: Diagramme de séquences : «Lire un message privé»

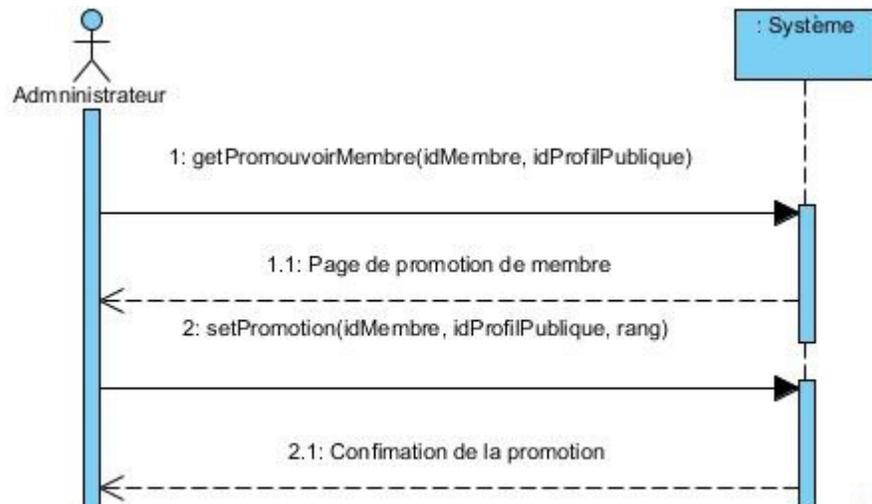


Figure 24: Diagramme de séquences : «Promouvoir un membre»

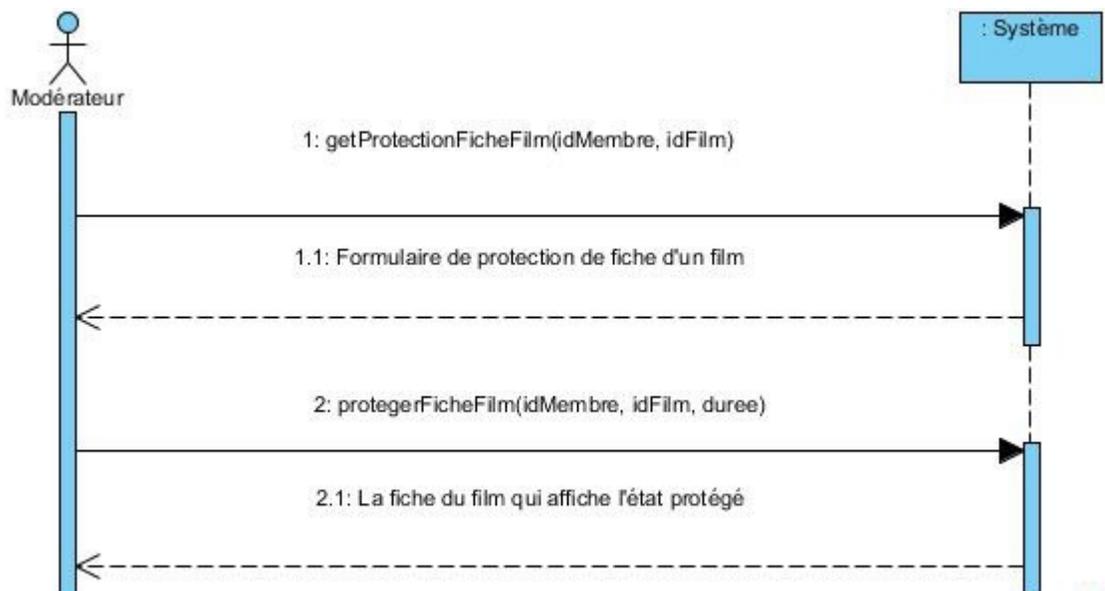


Figure 25: Diagramme de séquences : «Protéger un film contre les modifications»

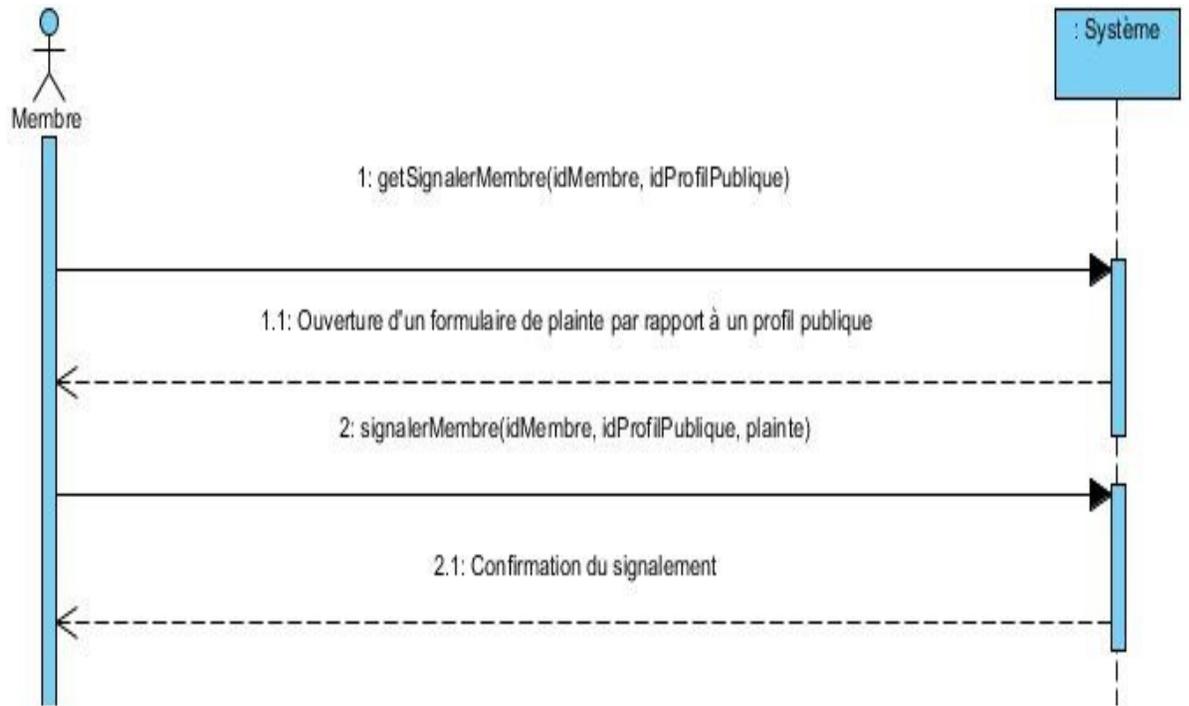


Figure 26: Diagramme de séquences : «Rapporter un abus d'un membre»



Figure 27: Diagramme de séquences : «Rapporter un abus sur un film»



Figure 28: Diagramme de séquences : «Rapporter un abus sur une critique»

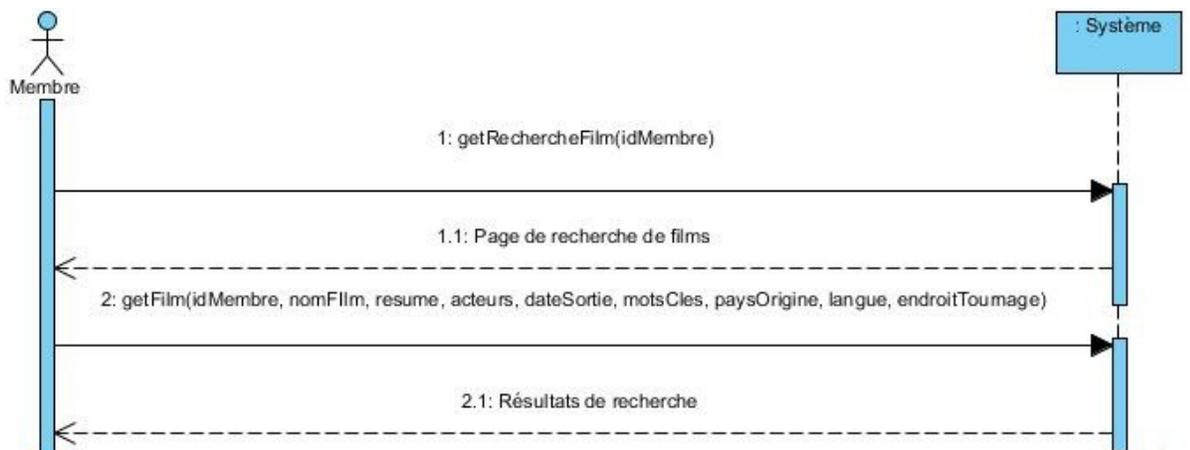


Figure 29: Diagramme de séquences : «Rechercher un film»

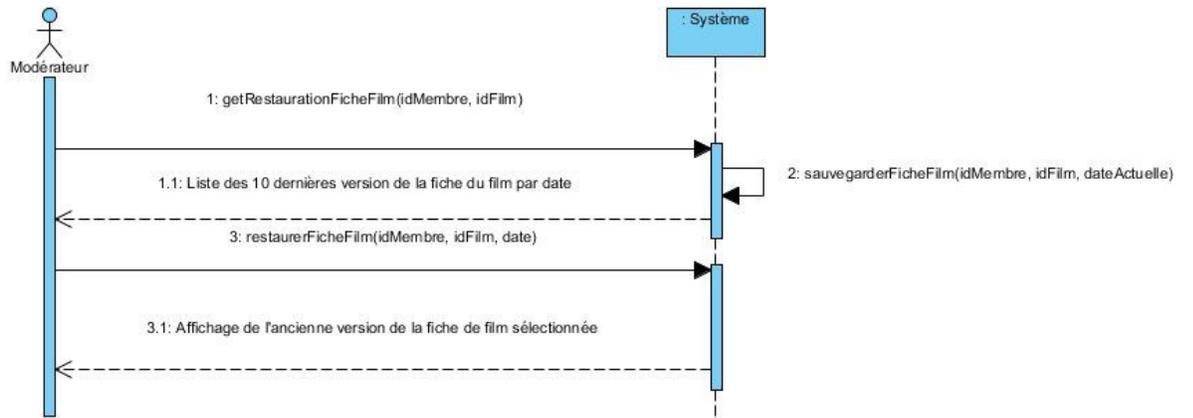


Figure 30: Diagramme de séquences : «Restaurer une version précédente de la fiche d'un film»

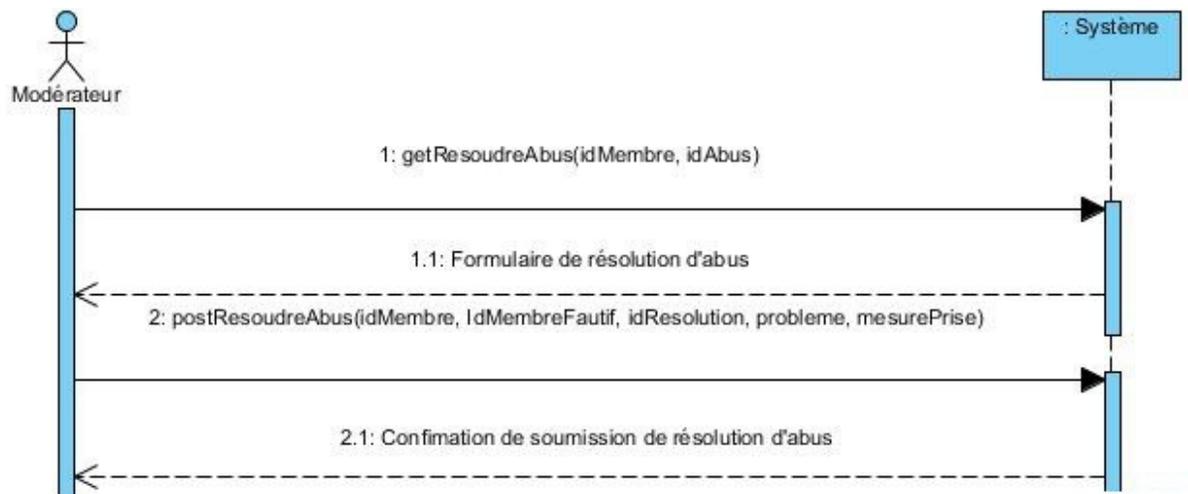


Figure 31: Diagramme de séquences : «Retirer un abus de la liste d'abus»

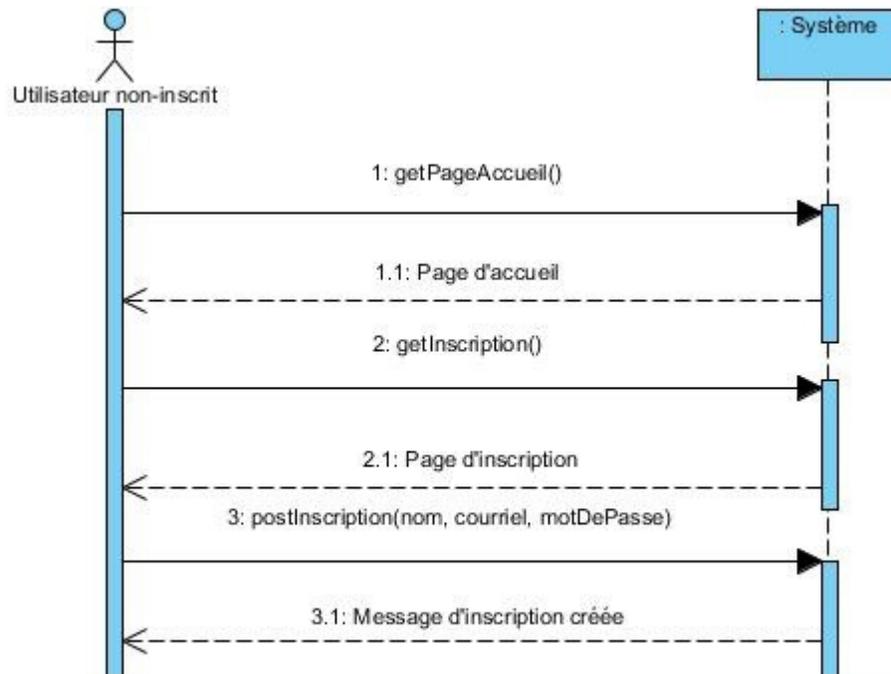


Figure 32: Diagramme de séquences : «S'inscrire»

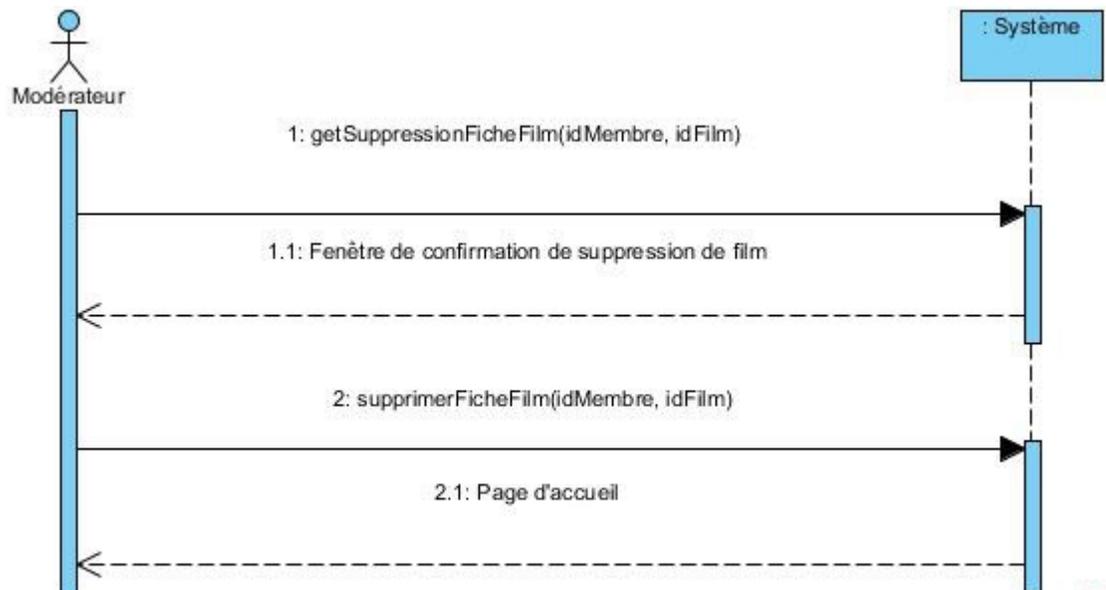


Figure 33: Diagramme de séquences : «Supprimer un film de la base de données»



Figure 34: Diagramme de séquences : «Supprimer un message privé»

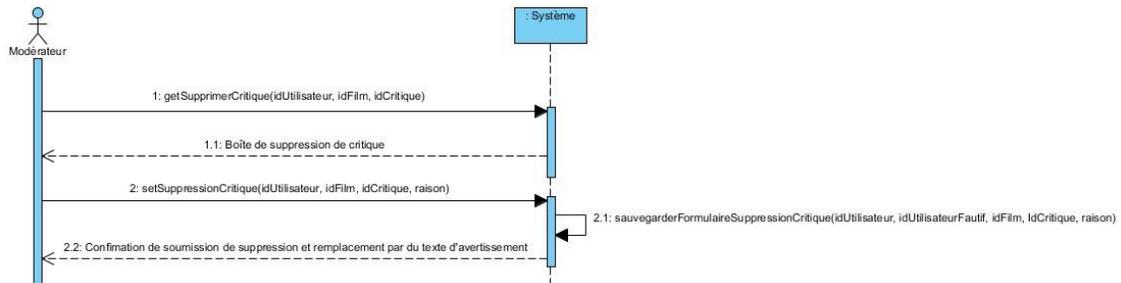


Figure 35: Diagramme de séquences : «Supprimer une critique»

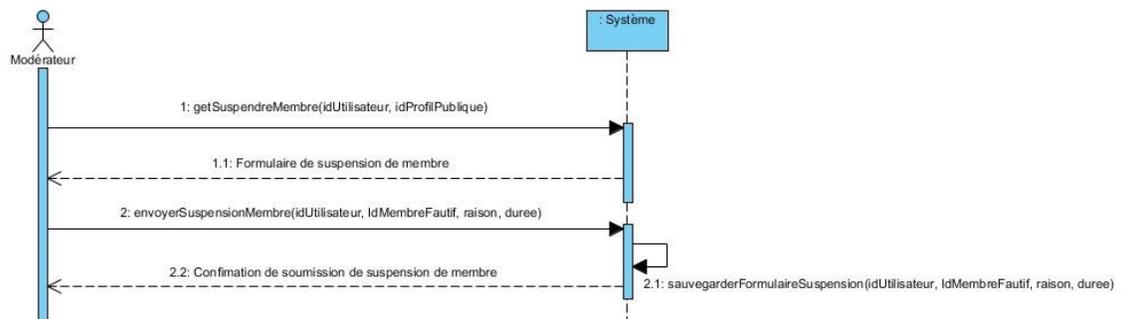


Figure 36: Diagramme de séquences : «Suspendre un membre»

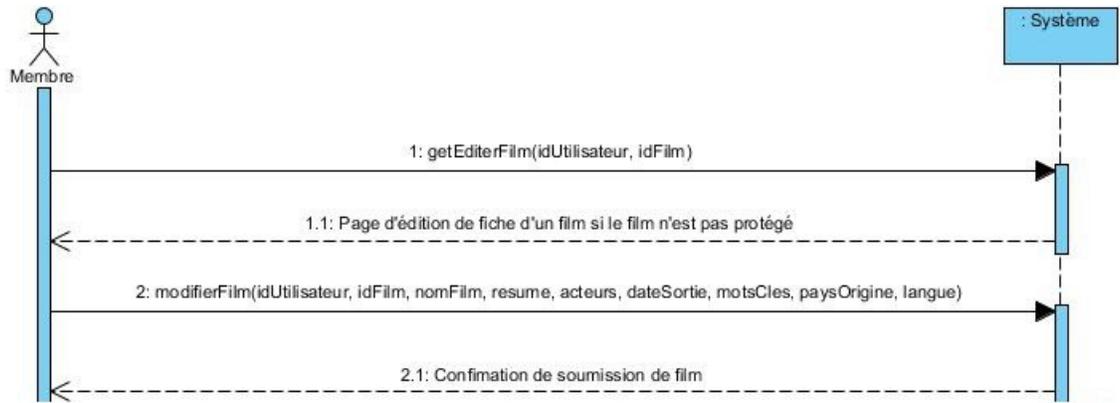


Figure 37: Diagramme se séquences : «Éditer la fiche d'un film»

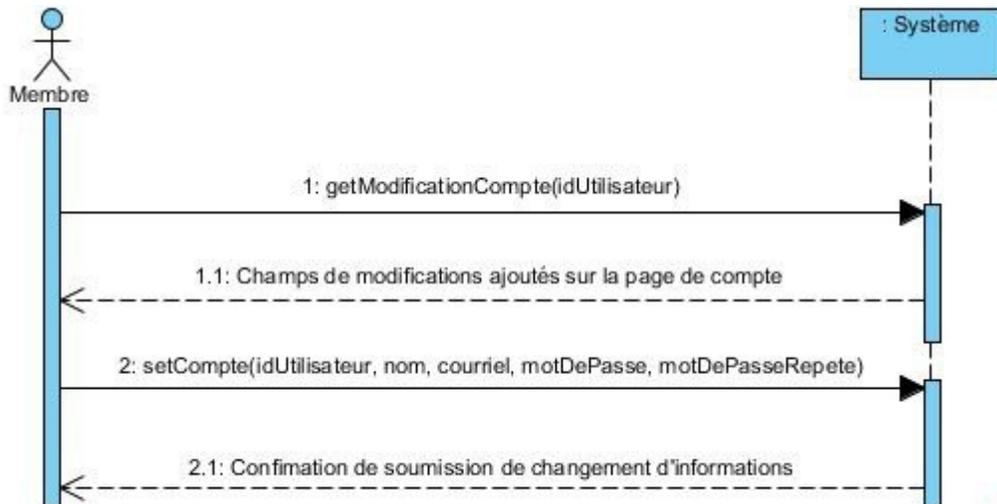


Figure 38: Diagramme se séquences : «Éditer son compte»

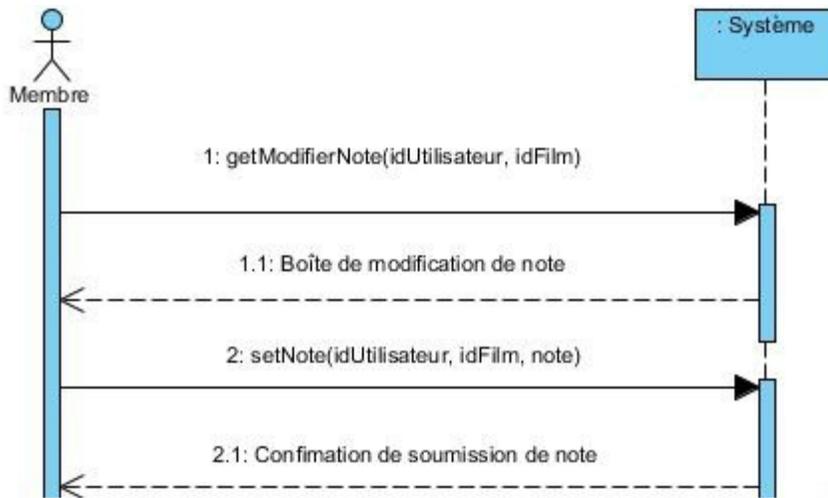


Figure 39: Diagramme se séquences : «Éditer une note donnée à un film»

3.2.2 Dictionnaire de données

Attribut	Type	Contrainte
idUtilisateur	NUMBER	Clé primaire

Tableau 37: Dictionnaire de données : Utilisateur

Attribut	Type	Contrainte
idUtilisateur	NUMBER	Clé étrangère vers Utilisateur, clé primaire

Tableau 38: Dictionnaire de données : Invite

Attribut	Type	Contrainte
idUtilisateur	NUMBER	Clé étrangère vers Utilisateur, clé primaire
banni	NUMBER	CHECK (value IN (0,1)) NOT NULL
suspendu	NUMBER	CHECK (value IN (0,1)) NOT NULL

Tableau 39: Dictionnaire de données : Membre

Attribut	Type	Contrainte
idUtilisateur	NUMBER	Clé étrangère vers Membre, clé primaire

Tableau 40: Dictionnaire de données : Modérateur

Attribut	Type	Contrainte
idUtilisateur	NUMBER	Clé étrangère vers Modérateur, clé primaire

Tableau 41: Dictionnaire de données : Administrateur

Attribut	Type	Contrainte
idProfil	NUMBER	Clé primaire
nom	VARCHAR2	NOT NULL
MembreIdUtilisateur	NUMBER	Clé étrangère vers Membre

Tableau 42: Dictionnaire de données : Profil

Attribut	Type	Contrainte
idCompte	NUMBER	Clé primaire
courriel	VARCHAR2	NOT NULL
motDePasse	VARCHAR2	NOT NULL
MembreIdUtilisateur	NUMBER	Clé étrangère vers Membre

Tableau 43: Dictionnaire de données : Compte

Attribut	Type	Contrainte
idBoiteCourriel	NUMBER	Clé primaire
courrielExterne	VARCHAR 2	NOT NULL
MembreIdUtilisateur	NUMBER	Clé étrangère vers Membre

Tableau 44: Dictionnaire de données : BoiteCourriel

Attribut	Type	Contrainte
idCourriel	NUMBER	Clé primaire
sujet	VARCHAR2	
corps	VARCHAR2	NOT NULL
dateEnvoi	DATE	NOT NULL
BoiteCourrielidBoiteCourriel	NUMBER	Clé étrangère vers BoiteCourriel
BoiteCourrielidBoiteCourriel2	NUMBER	Clé étrangère vers BoiteCourriel

Tableau 45: Dictionnaire de données : Courriel

Attribut	Type	Contrainte
idFilm	NUMBER	clé primaire
nomFilm	VARCHAR2	NOT NULL
resume	VARCHAR2	
paysOrigine	VARCHAR2	
langue	VARCHAR2	
dateSortie	DATE	
protege	NUMBER	CHECK (value IN (0,1)) NOT NULL

Tableau 46: Dictionnaire de données : Film

Attribut	Type	Contrainte
IdFilm	NUMBER	clé étrangère vers Film, clé primaire combinée à IdActeur
idActeur	NUMBER	clé étrangère vers Acteur, clé primaire combinée à IdFilm
role	VARCHAR2	NOT NULL

Tableau 47: Dictionnaire de données : Role

Attribut	Type	Contrainte
idActeur	NUMBER	clé primaire
nom	VARCHAR2	NOT NULL

Tableau 48: Dictionnaire de données : Acteur

Attribut	Type	Contrainte
IdFilm	NUMBER	clé étrangère vers Film, clé primaire combinée à IdUtilisateur
idUtilisateur	NUMBER	Clé étrangère vers Membre, clé primaire combinée à IdFilm
note	NOTE (Domaine)	NOT NULL

Tableau 49: Dictionnaire de données : NoteDonnee

Attribut	Type	Contrainte
IdFilm	NUMBER	clé étrangère vers Film, clé primaire
force	VARCHAR 2	NOT NULL

Tableau 50: Dictionnaire de données : Recommandation

Attribut	Type	Contrainte
idMotCle	NUMBER	clé primaire
titre	VARCHAR2	NOT NULL
description	VARCHAR2	
FilmIdFilm	NUMBER	clé étrangère vers Film

Tableau 51: Dictionnaire de données : MotCle

Attribut	Type	Contrainte
idCritique	NUMBER	clé primaire
critique	VARCHAR2	NOT NULL
FilmIdFilm	NUMBER	clé étrangère vers Film
MembreIdUtilisateur	NUMBER	Clé étrangère vers Membre

Tableau 52: Dictionnaire de données : Critique

Attribut	Type	Contrainte
idAbus	NUMBER	Clé étrangère vers Abus, clé primaire combinée à IdUtilisateur
idUtilisateur	NUMBER	Clé étrangère vers Modérateur, clé primaire combinée à idAbus

Tableau 53: Dictionnaire de données : AbusModérateur

Attribut	Type	Contrainte
idAbus	NUMBER	clé primaire
plainte	VARCHAR2	NOT NULL
resolu	NUMBER	CHECK (value IN (0,1)) NOT NULL
probleme	VARCHAR2	NOT NULL
mesurePrise	VARCHAR2	

Tableau 54: Dictionnaire de données : Abus

3.2.3 Contrats

Convention pour ID : CO[numéro d'acteur][numéro de contrat]

Numéro d'acteurs :

0. Utilisateur
1. Utilisateur non-inscrit
2. Membre
3. Modérateur
4. Administrateur
5. Administrateur de la base de données

ex: CO101 est le contrat 1 de l'acteur «Utilisateur non-inscrit».

Contrat CO101 : getInscription()	
Opération :	getInscription()
Références croisées :	S'inscrire
Exceptions :	-
Sorties :	Affiche la page d'inscription
Antécédents :	-
Conséquents :	-

Tableau 55: Contrat getInscription

Contrat CO102 : postInscription()	
Opération :	postInscription(nom, courriel, motDePasse)
Références croisées :	S'inscrire
Exceptions :	Affiche une erreur si le nom ou le courriel sont déjà utilisés
Sorties :	Affiche un message d'inscription créée
Antécédents :	Le nom, le courriel et le motDePasse sont valides
Conséquents :	Authentifie le nouvelle utilisateur

Tableau 56: Contrat postInscription

Contrat CO001 : getFaq()	
Opération :	getFaq()
Références croisées :	Consulter aide et FAQ
Exceptions :	-
Sorties :	Affiche la page d'aide et FAQ
Antécédents :	-
Conséquents :	-

Tableau 57: Contrat getFaq

Contrat CO002 : getReglements()	
Opération :	getReglements()
Références croisées :	Consulter règlements
Exceptions :	-
Sorties :	Afficher page de règlements
Antécédents :	-
Conséquents :	-

Tableau 58: Contrat getReglements

Contrat CO201 : getFilmVisionne()	
Opération :	getFilmVisionne()
Références croisées :	Consulter la liste de films déjà visionnés
Exceptions :	-
Sorties :	Affiche la page de films déjà visionnés
Antécédents :	-
Conséquents :	-

Tableau 59: Contrat getFilmVisionne

Contrat CO202 : ajouterFilmAListeFilmVisionne()	
Opération :	ajouterFilmAListeFilmVisionne(idUtilisateur, idFilm, note)
Références croisées :	Ajouter un film à la liste de films déjà visionnés
Exceptions :	-
Sorties :	Une confirmation d'ajout du visionnement du film
Antécédents :	L'idUtilisateur et l'idFilm sont valides
Conséquents :	-

Tableau 60: Contrat ajouterFilmAListeFilmVisionne

Contrat CO203 : getModifierNote()	
Opération :	getModifierNote(idUtilisateur, idFilm)
Références croisées :	Éditer une note donnée à un film
Exceptions :	-
Sorties :	Affiche boîte de modification de note
Antécédents :	L'idUtilisateur et l'idFilm sont valides
Conséquents :	-

Tableau 61: Contrat getModifierNote

Contrat CO204 : setNote()	
Opération :	setNote(idUtilisateur, idFilm, note)
Références croisées :	Éditer une note donnée à un film
Exceptions :	-
Sorties :	Affiche boîte d'ajout de note
Antécédents :	L'idUtilisateur, l'idFilm et note sont valides
Conséquents :	-

Tableau 62: Contrat setNote

Contrat CO205 : getAjoutCritique()	
Opération :	getAjoutCritique(idUtilisateur, idFilm)
Références croisées :	Critiquer un film
Exceptions :	-
Sorties :	Affiche boite d'ajout de critique
Antécédents :	L'idUtilisateur et l'idFilm sont valides
Conséquents :	-

Tableau 63: Contrat getAjoutCritique

Contrat CO206 : setCritique()	
Opération :	setCritique(idUtilisateur, idFilm, critique)
Références croisées :	Critiquer un film, Éditer une critique donnée à un film
Exceptions :	-
Sorties :	Affiche une confirmation de soumission de critique
Antécédents :	L'idUtilisateur, l'idFilm et critique sont valides
Conséquents :	-

Tableau 64: Contrat setCritique

Contrat CO207 : getModificationCritique()	
Opération :	getModificationCritique(idUtilisateur, idFilm)
Références croisées :	Éditer une critique donnée à un film
Exceptions :	-
Sorties :	Affiche une boîte de modification de critique
Antécédents :	L'idUtilisateur et l'idFilm sont valides
Conséquents :	-

Tableau 65: Contrat getModificationCritique

Contrat CO208 : getRecommandationsPersonnalisees()	
Opération :	getRecommandationsPersonnalisees(idUtilisateur)
Références croisées :	Consulter recommandations personnalisées
Exceptions :	-
Sorties :	Affiche la liste de films recommandés
Antécédents :	L'idUtilisateur est valide
Conséquents :	-

Tableau 66: Contrat getRecommandationsPersonnalisees

Contrat CO209 : getAjouterFilm()	
Opération :	getAjouterFilm(idUtilisateur)
Références croisées :	Créer la fiche d'un film
Exceptions :	-
Sorties :	Afficher la page d'ajout de film
Antécédents :	L'idUtilisateur est valide
Conséquents :	-

Tableau 67: Contrat getAjouterFilm

Contrat CO210 : ajouterFilm()	
Opération :	ajouterFilm(idUtilisateur, nomFilm, resume, acteurs, dateSortie, motsCles, paysOrigine, langue)
Références croisées :	Créer la fiche d'un film
Exceptions :	Afficher une erreur si le film existe déjà
Sorties :	Affiche un confirmation de soumission de film
Antécédents :	Les arguments sont valides.
Conséquents :	Affiche la fiche du nouveau film

Tableau 68: Contrat ajouterFilm

Contrat CO211 : getRechercheFilm()	
Opération :	getRechercheFilm(idUtilisateur)
Références croisées :	Rechercher un film
Exceptions :	-
Sorties :	Afficher la page de recherche de films
Antécédents :	L'idUtilisateur est valide
Conséquents :	-

Tableau 69: Contrat getRechercheFilm

Contrat CO212 : getFilm()	
Opération :	getFilm(idUtilisateur, nomFilm, resume, acteurs, dateSortie, motsCles, paysOrigine, langue)
Références croisées :	Rechercher un film
Exceptions :	-
Sorties :	Affiche les résultats de la recherche
Antécédents :	Les arguments sont valides.
Conséquents :	-

Tableau 70: Contrat getFilm

Contrat CO213 : getFilm()	
Opération :	getFilm(idUtilisateur, idFilm)
Références croisées :	Consulter la fiche d'un film
Exceptions :	-
Sorties :	Affiche la page de fiche d'un film
Antécédents :	L'idUtilisateur et l'idFilm sont valides
Conséquents :	-

Tableau 71: Contrat getFilm

Contrat CO214 : getEditerFilm()	
Opération :	getEditerFilm(idUtilisateur, idFilm)
Références croisées :	Éditer la fiche d'un film
Exceptions :	Afficher une erreur si le film est protégé
Sorties :	Affiche la page d'édition d'un film
Antécédents :	L'idUtilisateur et l'idFilm sont valides
Conséquents :	-

Tableau 72: Contrat getEditerFilm

Contrat CO215 : getEditerFilm()	
Opération :	modifierFilm(idUtilisateur, idFilm, nomFilm, resume, acteurs, dateSortie, motsCles, paysOrigine, langue)
Références croisées :	Éditer la fiche d'un film
Exceptions :	-
Sorties :	Affiche un confirmation de soumission de film
Antécédents :	Les arguments sont valides
Conséquents :	-

Tableau 73: Contrat getEditerFilm

Contrat CO216 : signalerFilm()	
Opération :	signalerFilm(idUtilisateur, idFilm)
Références croisées :	Rapporter un abus sur un film
Exceptions :	-
Sorties :	Afficher une confirmation du signalement
Antécédents :	L'idUtilisateur et l'idFilm sont valides
Conséquents :	-

Tableau 74: Contrat signalerFilm

Contrat CO217 : signalerCritique()	
Opération :	signalerCritique(idUtilisateur, idFilm, idCritique)
Références croisées :	Rapporter un abus sur une critique
Exceptions :	-
Sorties :	Afficher une confirmation du signalement
Antécédents :	L'idUtilisateur, l'idFilm et l'idCritique sont valides
Conséquents :	-

Tableau 75: Contrat signalerCritique

Contrat CO218 : getRechercherProfilPublic()	
Opération :	getRechercherProfilPublic(idUtilisateur)
Références croisées :	Consulter un profil publique
Exceptions :	-
Sorties :	Afficher la page de recherche de profil publique
Antécédents :	L'idUtilisateur est valide
Conséquents :	-

Tableau 76: Contrat getRechercherProfilPublic

Contrat CO219 : rechercherProfil()	
Opération :	rechercherProfil(idUtilisateur, nomMembreRecherche)
Références croisées :	Consulter un profil publique
Exceptions :	-
Sorties :	Afficher le résultat de la recherche
Antécédents :	L'idUtilisateur et nomMembreRecherche sont valides
Conséquents :	-

Tableau 77: Contrat rechercherProfil

Contrat CO220 : getProfilPublique()	
Opération :	getProfilPublique(idUtilisateur, idProfilPublique)
Références croisées :	Consulter un profil publique
Exceptions :	-
Sorties :	Afficher le profil publique du membre
Antécédents :	L'idUtilisateur et l'idProfilPublique sont valides
Conséquents :	-

Tableau 78: Contrat getProfilPublique

Contrat CO221 : getSignalerMembre()	
Opération :	getSignalerMembre(idUtilisateur)
Références croisées :	Rapporter un abus d'un membre
Exceptions :	-
Sorties :	Afficher un formulaire de plainte par rapport à un profil publique
Antécédents :	L'idUtilisateur est valide
Conséquents :	-

Tableau 79: Contrat getSignalerMembre

Contrat CO222 : signalerMembre()	
Opération :	signalerMembre(idUtilisateur, idProfilPublique, plainte)
Références croisées :	Rapporter un abus d'un membre
Exceptions :	-
Sorties :	Afficher une confirmation du signalement
Antécédents :	L'idUtilisateur, l'idProfilPublique et plainte sont valides
Conséquents :	-

Tableau 80: Contrat signalerMembre

Contrat CO223 : getCompte()	
Opération :	getCompte(idUtilisateur)
Références croisées :	Consulter son compte
Exceptions :	-
Sorties :	Afficher la page de compte du membre
Antécédents :	L'idUtilisateur est valide
Conséquents :	-

Tableau 81: Contrat getCompte

Contrat CO224 : getModificationCompte()	
Opération :	getModificationCompte(idUtilisateur)
Références croisées :	Éditer son compte
Exceptions :	-
Sorties :	Afficher les champs de modifications ajoutés sur la page de compte
Antécédents :	L'idUtilisateur est valide
Conséquents :	-

Tableau 82: Contrat getModificationCompte

Contrat CO225 : setCompte()	
Opération :	setCompte(idUtilisateur, nom, courriel, motDePasse, motDePasseRepete)
Références croisées :	Éditer son compte
Exceptions :	-
Sorties :	Afficher une confirmation de soumission de changement d'informations
Antécédents :	Les arguments sont valides.
Conséquents :	-

Tableau 83: Contrat setCompte

Contrat CO226 : getCreerMessage()	
Opération :	getCreerMessage(idUtilisateur)
Références croisées :	Envoyer un message privé à un membre
Exceptions :	-
Sorties :	Afficher la page de création de message
Antécédents :	L'idUtilisateur est valide
Conséquents :	-

Tableau 84: Contrat getCreerMessage

Contrat CO227 : rechercherDestinataires()	
Opération :	rechercherDestinataires(idUtilisateur)
Références croisées :	Envoyer un message privé à un membre
Exceptions :	-
Sorties :	Afficher la liste des destinataires possibles
Antécédents :	L'idUtilisateur est valide
Conséquents :	-

Tableau 85: Contrat rechercherDestinataires

Contrat CO228 : postMessage()	
Opération :	envoyerMessage(idUtilisateur, corps, sujet, destinataires)
Références croisées :	Envoyer un message privé à un membre, Avertir un membre
Exceptions :	-
Sorties :	Afficher une confirmation de l'envoi du message
Antécédents :	Les arguments sont valides.
Conséquents :	-

Tableau 86: Contrat postMessage

Contrat CO229 : getMessagesRecus()	
Opération :	getMessagesRecus(idUtilisateur)
Références croisées :	Consulter la liste des messages privés reçus
Exceptions :	-
Sorties :	Afficher la page de messages reçus
Antécédents :	L'idUtilisateur est valide
Conséquents :	-

Tableau 87: Contrat getMessagesRecus

Contrat CO230 : getMessagesEnvoyes()	
Opération :	getMessagesEnvoyes(idUtilisateur)
Références croisées :	Consulter la liste des messages privés envoyés
Exceptions :	-
Sorties :	Afficher la page de messages envoyés
Antécédents :	L'idUtilisateur est valide
Conséquents :	-

Tableau 88: Contrat getMessagesEnvoyes

Contrat CO231 : getMessage()	
Opération :	getMessage(idUtilisateur, idMessage)
Références croisées :	Lire un message privé
Exceptions :	-
Sorties :	Afficher la page d'un message privé
Antécédents :	L'idUtilisateur est valide
Conséquents :	-

Tableau 89: Contrat getMessage

Contrat CO232 : supprimerMessage()	
Opération :	supprimerMessage(idUtilisateur, idMessage)
Références croisées :	Supprimer un message privé
Exceptions :	-
Sorties :	Afficher une confirmation de la suppression
Antécédents :	L'idUtilisateur et l'idMessage sont valides
Conséquents :	-

Tableau 90: Contrat supprimerMessage

Contrat CO301 : getListeAbus()	
Opération :	getListeAbus(idUtilisateur)
Références croisées :	Consulter la liste d'abus
Exceptions :	-
Sorties :	Afficher la page avec la liste des abus de tous les membres
Antécédents :	L'idUtilisateur est valide
Conséquents :	-

Tableau 91: Contrat getListeAbus

Contrat CO302 : getResoudreAbus()	
Opération :	getResoudreAbus(idUtilisateur, idAbus)
Références croisées :	Retirer un abus de la liste d'abus
Exceptions :	-
Sorties :	Afficher le formulaire de résolution d'abus
Antécédents :	L'idUtilisateur et l'idAbus sont valides
Conséquents :	-

Tableau 92: Contrat getResoudreAbus

Contrat CO303 : postResoudreAbus()	
Opération :	postResoudreAbus(idUtilisateur, idUtilisateurFautif, idResolution, probleme, mesurePrise)
Références croisées :	Retirer un abus de la liste d'abus
Exceptions :	-
Sorties :	Afficher une confirmation de soumission de résolution d'abus
Antécédents :	Les arguments sont valides
Conséquents :	-

Tableau 93: Contrat postResoudreAbus

Contrat CO304 : getSuspendreMembre()	
Opération :	getSuspendreMembre(idUtilisateur, idProfilPublique)
Références croisées :	Avertir un membre, Suspendre un membre
Exceptions :	-
Sorties :	Afficher un formulaire de suspension de membre
Antécédents :	L'idUtilisateur et l'idProfilPublique sont valides
Conséquents :	-

Tableau 94: Contrat getSuspendreMembre

Contrat CO305 : postAvertissementMembre()	
Opération :	envoyerAvertissementMembre(idUtilisateur, idUtilisateurFautif, raison)
Références croisées :	Avertir un membre
Exceptions :	-
Sorties :	Afficher une confirmation de soumission de suspension de membre
Antécédents :	L'idUtilisateur, l'idUtilisateurFautif et raison sont valides
Conséquents :	-

Tableau 95: Contrat postAvertissementMembre

Contrat CO306 : sauvegarderFormulaireAvertissement()	
Opération :	sauvegarderFormulaireAvertissement(idUtilisateur, idUtilisateurFautif, raison)
Références croisées :	Avertir un membre
Exceptions :	-
Sorties :	-
Antécédents :	L'idUtilisateur, l'idUtilisateurFautif et raison sont valides
Conséquents :	-

Tableau 96: Contrat sauvegarderFormulaireAvertissement

Contrat CO307 : postAvertissementMembre()	
Opération :	envoyerSuspensionMembre(idUtilisateur, idUtilisateurFautif, raison, duree)
Références croisées :	Suspendre un membre
Exceptions :	-
Sorties :	Afficher une confirmation de soumission de suspension de membre
Antécédents :	Les arguments sont valides
Conséquents :	-

Tableau 97: Contrat postAvertissementMembre

Contrat CO308 : sauvegarderFormulaireSuspension()	
Opération :	sauvegarderFormulaireSuspension(idUtilisateur, idUtilisateurFautif, raison, duree)
Références croisées :	Suspendre un membre
Exceptions :	-
Sorties :	-
Antécédents :	Les arguments sont valides
Conséquents :	-

Tableau 98: Contrat sauvegarderFormulaireSuspension

Contrat CO309 : getCensurerCritique()	
Opération :	getCensurerCritique(idUtilisateur, idFilm, idCritique)
Références croisées :	Censurer une critique
Exceptions :	-
Sorties :	Afficher une boite de censure de critique
Antécédents :	L'idUtilisateur, l'idUtilisateurFautif et l'idCritique sont valides
Conséquents :	-

Tableau 99: Contrat *getCensurerCritique*

Contrat CO310 : setCensureCritique()	
Opération :	setCensureCritique(idUtilisateur, idFilm, idCritique, debutCensure, finCensure)
Références croisées :	Censurer une critique
Exceptions :	-
Sorties :	Afficher une confirmation de soumission de censure et remplacement par du texte de censure
Antécédents :	Les argument sont valides
Conséquents :	-

Tableau 100: Contrat *setCensureCritique*

Contrat CO311 : getSupprimerCritique()	
Opération :	getSupprimerCritique(idUtilisateur, idFilm, idCritique)
Références croisées :	Supprimer une critique
Exceptions :	-
Sorties :	Affiche une fiche de suppression de critique
Antécédents :	L'idUtilisateur, l'idFilm et l'idCritique sont valides
Conséquents :	-

Tableau 101: Contrat getSupprimerCritique

Contrat CO312 : setSuppressionCritique()	
Opération :	setSuppressionCritique(idUtilisateur, idFilm, idCritique, raison)
Références croisées :	Supprimer une critique
Exceptions :	-
Sorties :	Affiche une confirmation de soumission de suppression et remplacement par du texte d'avertissement
Antécédents :	Les arguments sont valides
Conséquents :	-

Tableau 102: Contrat setSuppressionCritique

Contrat CO313 : sauvegarderFormulaireSuppressionCritique()	
Opération :	sauvegarderFormulaireSuppressionCritique(idUtilisateur , idUtilisateurFautif, idFilm, idCritique, raison)
Références croisées :	Supprimer une critique
Exceptions :	-
Sorties :	-
Antécédents :	Les arguments sont valides
Conséquents :	-

Tableau 103: Contrat sauvegarderFormulaireSuppressionCritique

Contrat CO314 : getRestaurationFicheFilm()	
Opération :	getRestaurationFicheFilm(idUtilisateur, idFilm)
Références croisées :	Restaurer une version précédente de la fiche d'un film
Exceptions :	-
Sorties :	Afficher la liste des 10 dernières version de la fiche du film par date
Antécédents :	L'idUtilisateur et l'idFilm sont valides
Conséquents :	-

Tableau 104: Contrat getRestaurationFicheFilm

Contrat CO315 : sauvegarderFicheFilm()	
Opération :	sauvegarderFicheFilm(idUtilisateur, idFilm, dateActuelle)
Références croisées :	Restaurer une version précédente de la fiche d'un film
Exceptions :	-
Sorties :	-
Antécédents :	L'idUtilisateur, l'idFilm et dateActuelle sont valides
Conséquents :	-

Tableau 105: Contrat sauvegarderFicheFilm

Contrat CO316 : restaurerFicheFilm()	
Opération :	restaurerFicheFilm(idUtilisateur, idFilm, date)
Références croisées :	Restaurer une version précédente de la fiche d'un film
Exceptions :	-
Sorties :	Afficher l'ancienne version de la fiche du film
Antécédents :	L'idUtilisateur, l'idFilm et date sont valides
Conséquents :	-

Tableau 106: Contrat restaurerFicheFilm

Contrat CO317 : getProtectionFicheFilm()	
Opération :	getProtectionFicheFilm(idUtilisateur, idFilm)
Références croisées :	Protéger un film contre les modifications
Exceptions :	-
Sorties :	Afficher un formulaire de protection de fiche d'un film
Antécédents :	L'idUtilisateur et l'idFilm sont valides
Conséquents :	-

Tableau 107: Contrat getProtectionFicheFilm

Contrat CO318 : protegerFicheFilm()	
Opération :	protegerFicheFilm(idUtilisateur, idFilm, duree)
Références croisées :	Protéger un film contre les modifications
Exceptions :	-
Sorties :	Afficher la fiche du film qui affiche l'état protégé
Antécédents :	L'idUtilisateur, l'idFilm et duree sont valides
Conséquents :	-

Tableau 108: Contrat protegerFicheFilm

Contrat CO319 : getSuppressionFicheFilm()	
Opération :	getSuppressionFicheFilm(idUtilisateur, idFilm)
Références croisées :	Supprimer un film
Exceptions :	-
Sorties :	Affiche une fenêtre de confirmation de suppression de film
Antécédents :	L'idUtilisateur et l'idFilm sont valides
Conséquents :	-

Tableau 109: Contrat getSuppressionFicheFilm

Contrat CO320 : supprimerFicheFilm()	
Opération :	supprimerFicheFilm(idUtilisateur, idFilm)
Références croisées :	Supprimer un film
Exceptions :	-
Sorties :	Affiche la page d'accueil
Antécédents :	L'idUtilisateur et l'idFilm sont valides
Conséquents :	-

Tableau 110: Contrat supprimerFicheFilm

Contrat CO401 : bannirMembre()	
Opération :	bannirMembre(idUtilisateur, idUtilisateurABannir)
Références croisées :	Bannir un membre
Exceptions :	-
Sorties :	Affiche la page d'accueil
Antécédents :	L'idUtilisateur et l'idUtilisateurABannir sont valides
Conséquents :	-

Tableau 111: Contrat bannirMembre

Contrat CO402 : getPromouvoirMembre()	
Opération :	getPromouvoirMembre(idUtilisateur, idProfilPublique)
Références croisées :	Promouvoir un membre
Exceptions :	-
Sorties :	Affiche la page de promotion de membre
Antécédents :	L'idUtilisateur et l'idProfilPublique sont valides
Conséquents :	-

Tableau 112: Contrat getPromouvoirMembre

Contrat CO403 : setPromotion()	
Opération :	setPromotion(idUtilisateur, idProfilPublique, rang)
Références croisées :	Promouvoir un membre
Exceptions :	-
Sorties :	Afficher une confirmation de la promotion
Antécédents :	L'idUtilisateur, l'idProfilPublique et rang sont valides
Conséquents :	-

Tableau 113: Contrat setPromotion

Contrat CO501 : getVersionsBaseDeDonnees()	
Opération :	getVersionsBaseDeDonnees(idAdministrateurBD)
Références croisées :	Effectuer restauration de la base de données
Exceptions :	-
Sorties :	Afficher la liste des version antérieurs de la base de données
Antécédents :	L'idAdministrateurBD est valide
Conséquents :	-

Tableau 114: Contrat getVersionsBaseDeDonnees

Contrat CO502 : restaurationBaseDeDonnees()	
Opération :	restaurationBaseDeDonnees(idAdministrateurBD, idVersion)
Références croisées :	Effectuer restauration de la base de données
Exceptions :	-
Sorties :	Afficher une confirmation de restauration de la base de données
Antécédents :	L'idAdministrateurBD et l'idVersion sont valides
Conséquents :	-

Tableau 115: Contrat restaurationBaseDeDonnees

3.2.4 Modèle objets et modèle relationnel

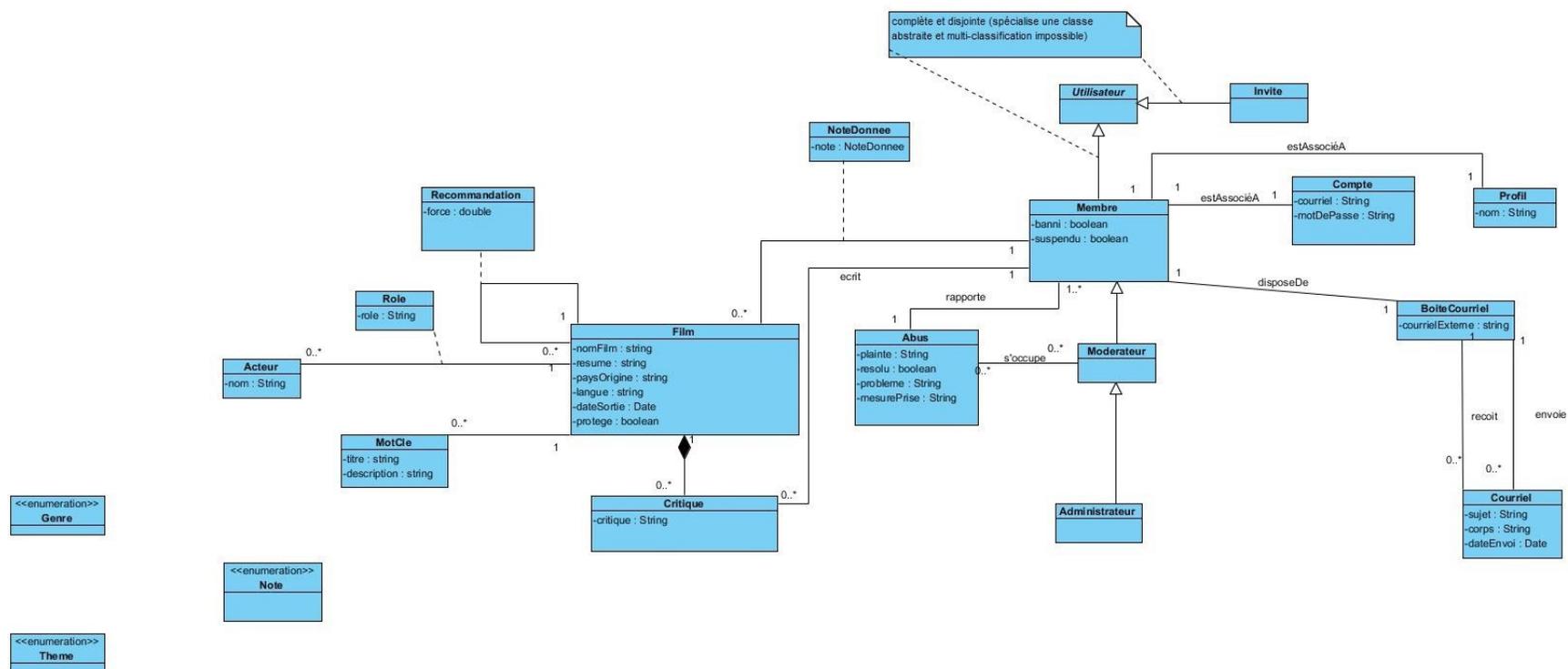


Figure 40: Modèle objet

Note : Les énumérations devront être remplies plus tard (avant la création de la base de données).

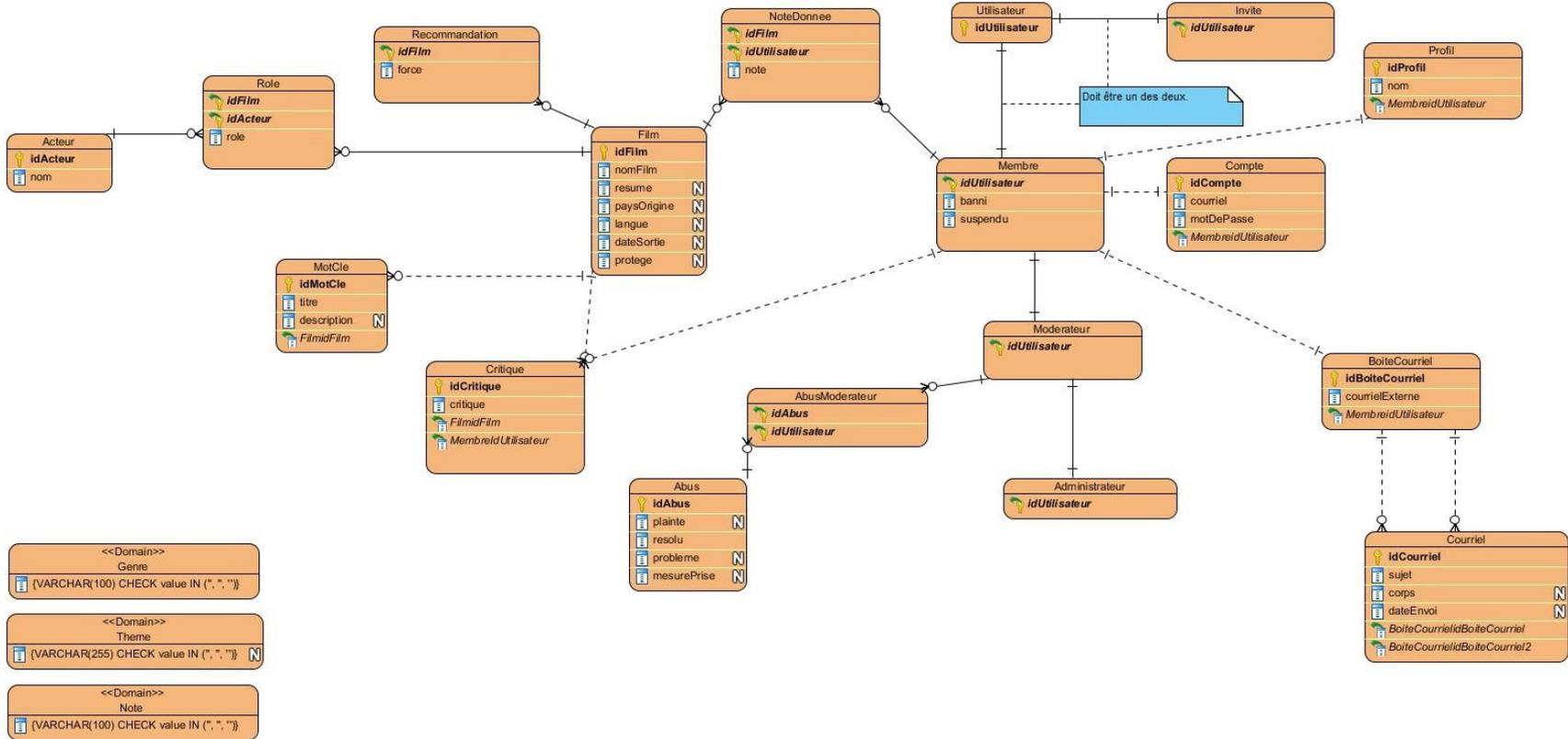


Figure 41: Schéma relationnel

3.3 Exigences d'opérations, de communications et de performance

3.3.1 Nombre de terminaux à supporter

Le système devrait pouvoir gérer un nombre de terminaux de l'ordre de quelques centaines d'utilisateurs en période de pointe sinon de quelques douzaines à une centaine.

3.3.2 Nombre d'utilisateurs simultanés

Le système doit pouvoir gérer un nombre d'utilisateur du même ordre que le nombre de terminaux.

3.3.3 Quantité et types d'information à traiter

Le système devra gérer trois types d'information :

- Un nombre élevé de requêtes seront sur les fiches de film
- Un nombre élevé au départ mais qui devrait devenir faible à un rythme qui dépendra de la popularité initiale du site web, sera pour la création de fiches de film
- Un nombre faible de demandes de recommandation, qui nécessiteront elles-mêmes un grand nombre de requête sur des fiches de film

3.3.4 Nombre de transactions à traiter par unité de temps sous conditions normales et en période de pointe

s.o.

3.4 Exigences logiques de la base de données

3.4.1 Types d'informations utilisées par les différentes fonctions

Puisque les deux fonctions principales d'un point de vue interactif sont la recherche de films et les demandes de fiche recommandations, les fonctions prendront soit les entrées des utilisateurs afin de traiter leur requêtes pour faire fonctionner le site web soit les informations sur les fiches de films pour créer la fiche de recommandations.

3.4.2 Fréquence d'utilisation

Il est fort probable que seulement une partie des utilisateurs augmente la base de données en remplissant des fiches de films dans leur temps libre. Le trafic du site sera probablement généré en grande partie par les usagers qui viennent voir des informations sur des films précis ou voir des nouvelles recommandations d'une semaine à l'autre.

3.4.3 Capacité d'accès

La base de données doit être accessible par plusieurs instances de requêtes sur diverses fiches et ce de manière concurrente. Une demande des informations sur la fiche d'un film devra se faire sans délai.

3.4.4 Exigences de rétention des données

Puisque le système se veut comme étant un Wikia avec un plus la rétention des données est primordiale. Une perte de données sur tous les films du genre science-fiction reviendrait à perdre également tous les utilisateurs qui s'attendent à se faire recommander des films de science-fiction.

3.5 Contraintes de conception

Les différents diagrammes produits lors de la documentation de ce projet seront produits à l'aide du logiciel Visual Paradigm. Le document lui-même est développé en collaboration à l'aide de l'outil en ligne Google Docs et transformé en document pdf à l'aide de LibreOffice.

À l'aide de ces outils les rencontres physiques entre les 5 membres de l'équipe peuvent ainsi être réduites au minimum et le travail peut être divisé facilement.

Au niveau de la conception de la base de données il faudra prévoir un utilisateur spécial (l'administrateur de la base de données) qui aura accès à une interface spéciale permettant de restaurer d'anciennes versions de la base de données du site.

Le code sera produit et partagé à l'aide de la technologie Git sur un répertoire privé sur bitbucket (qui permet des répertoires privés gratuitement pour 5 personnes au maximum).

3.6 Exigences non fonctionnelles

Il existe trois grandes catégories d'exigences non fonctionnelles : les exigences observables à l'exécution (donc par les utilisateurs), les exigences non observables à l'exécution (mais qui seront perçues par les développeurs) ainsi que les exigences d'affaires. Pour les besoins de ce rapport, nous nous focaliserons principalement sur les deux premières catégories d'exigences non fonctionnelles, à savoir les exigences observables à l'exécution et les exigences non observables à l'exécution.

Étant donné que les exigences non fonctionnelles s'influencent mutuellement (de manière négative ou positive), nous sommes amenés à faire des compromis parmi ces différentes exigences.

La première catégorie d'exigences non fonctionnelles, celles observables à l'exécution, comprend les aspects suivants :

- Performance : Le site devra supporter des pics de charge élevés ainsi qu'un nombre important de transactions sur sa base de données sans que cela n'affecte la réactivité du site pour l'ensemble des utilisateurs.
- Sécurité : Le modérateur a la capacité de protéger un film contre les modifications, ce qui permet au site de se prémunir contre les modifications indésirables.
- Fiabilité : Durant sa phase de développement, le site sera conçu de manière à pouvoir le tester plus aisément, ce qui permettra d'en garantir la fiabilité auprès des utilisateurs du site.
- Disponibilité : Le site devra être accessible en tout temps, ce qui signifie qu'il devra supporter des pics de charge importants tout en subissant une proportion négligeable de pannes pour les utilisateurs. Pour cela, les serveurs du site devront supporter les pics élevés de charge alors que la base de données aura à traiter un nombre élevé de transactions.
- Résilience : A l'aide des sauvegardes effectuées régulièrement sur la base de données, l'administrateur de base de données pourra restaurer une version antérieure de la base de données dans le cas où elle se retrouverait dans un état erroné, ce qui permettra au système de se rétablir dans ce cas. De même, grâce aux sauvegardes effectuées sur la fiche d'un film, le modérateur peut restaurer une version antérieure de la fiche de ce film dans le cas où la fiche se trouverait dans un état erroné.
- Utilisabilité : L'interface du site sera conçue de manière à en faciliter l'utilisation pour les utilisateurs, principalement pour les nouveaux membres qui se familiarisent avec le site.

La deuxième catégorie d'exigences non fonctionnelles, celles non observables à l'exécution, comprend les aspects suivants :

- **Modifiabilité** : le site sera conçu de manière à en faciliter la modification afin de pouvoir l'adapter aux futurs besoins des utilisateurs.
- **Portabilité** : Étant donné qu'il s'agit d'un site web, nous veillerons à ce qu'il soit accessible adéquatement sur les navigateurs de différentes plates-formes, ce qui inclut les environnements de bureau et les environnements mobiles.
- **Modularité** : La conception du site sera effectuée de manière à pouvoir réduire le couplage entre des différents composants et d'améliorer la cohésion de chaque composant afin de pouvoir garantir la modularité du site.
- **Testabilité** : Les composants du site seront conçus de manière à pouvoir les tester aisément afin de pouvoir garantir la fiabilité et la modifiabilité du site.

3.7 Organisation des exigences spécifiques

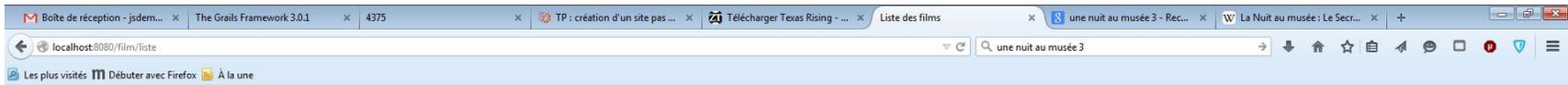
S.O.

4 Informations complémentaires

4.1 Annexes



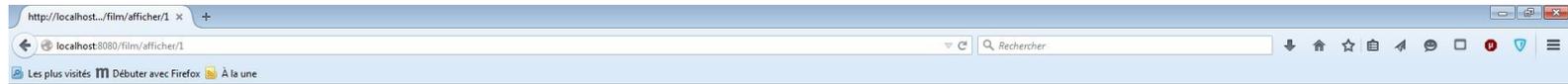
Figure 42: Page d'accueil du Wiki de films



- [Accueil](#)
- [Liste de film](#)
- [Rechercher](#)
- [Ajouter](#)
- [MesFilms](#)
- [SeDeconnecter](#)
- [FAQ](#)

Film	Année	Note
Avengers : L'Ère d'Ultron	1er mai 2014	

Figure 43: Liste de films déjà visionnés



WikiFilm

- [Accueil](#)
- [Liste de film](#)
- [Rechercher](#)
- [Ajouter](#)
- [MesFilms](#)
- [SeDeconnecter](#)
- [FAQ](#)

Fiche de Avengers : L'Ère d'Ultron

Genre: Action, Science Fiction
Theme: Super Hero
Classement du film: Général
Longueur: 142 minutes
Date: 1er mai 2014
Pays: États-Unis
Directeurs: Joss Whedon
Producteurs: MarvelStudio
Auteurs: Joss Whedon
Acteurs: Robert Downey Jr., Chris Hemsworth, Mark Ruffalo, Chris Evans, Scarlett Johansson, Jeremy Renner

Recommandation



[IronMan](#)



Vous avez vue ce film?

Notez le et ajoutez le à votre liste!

Excellent (4)

Figure 44: Fiche d'un film

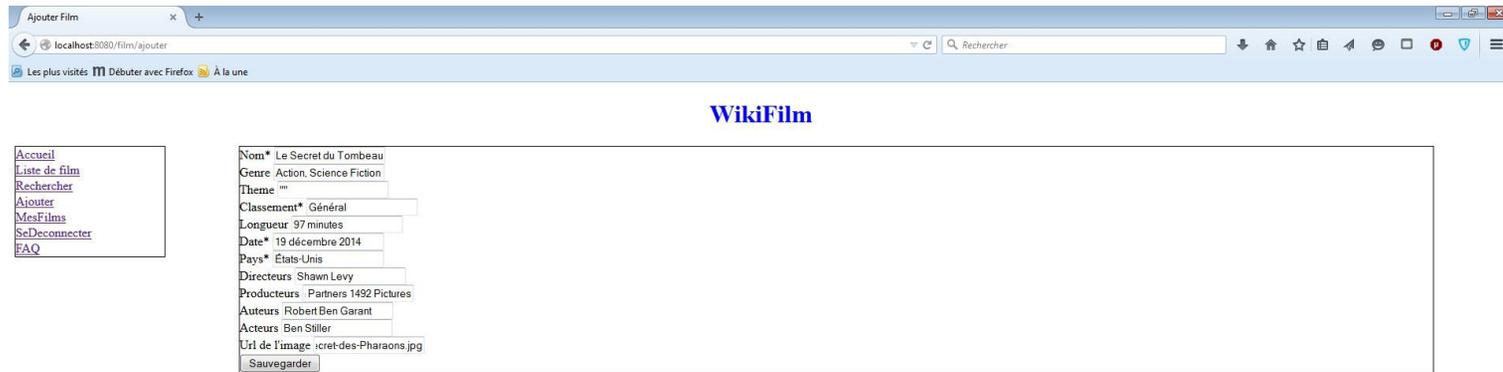


Figure 45: Création de la fiche d'un film

http://localhost.../film/afficher/2 x +

localhost:3080/film/afficher/2

Rechercher

Les plus visités Débuter avec Firefox À la une

WikiFilm

[Accueil](#)
[Liste de film](#)
[Rechercher](#)
[Ajouter](#)
[MesFilms](#)
[SeDeconnecter](#)
[FAQ](#)

Fiche de Une Nuit au Musée : Le Secret du Tombeau

Genre: Action, Science Fiction
Theme: ""
Classement du film: Général
Longueur: 97 minutes
Date: 19 décembre 2014
Pays: États-Unis
Directeurs: Shawn Levy
Producteurs: 21 Laps Entertainment Ingenious Film Partners 1492 Pictures
Auteurs: Robert Ben Garant
Acteurs: Ben Stiller



Vous avez vu ce film?

Notez le et ajoutez le à votre liste!
Excellent (4)

Figure 46: Autre fiche d'un film



Figure 47: Version mise à jour de la liste de films déjà visionnés



Inscription

Surnom*

Adresse Courriel*

Pays

Age

Sexe

Acceptez vous les [règlements](#)

Figure 48: Inscription