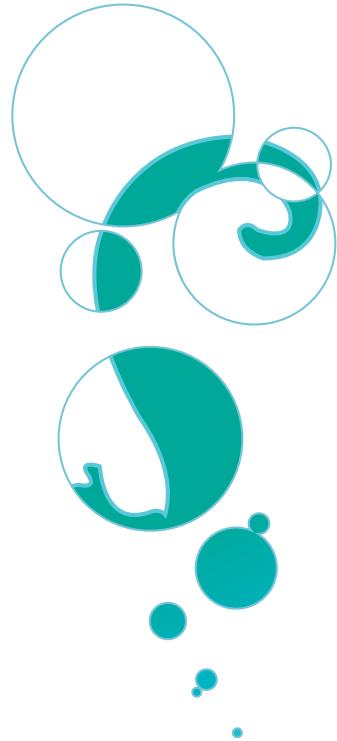


Rapport final du projet semestriel
Groupe Hippocampe | Classe 35 MM
Heig-vd | Comem†
13 juin 2008

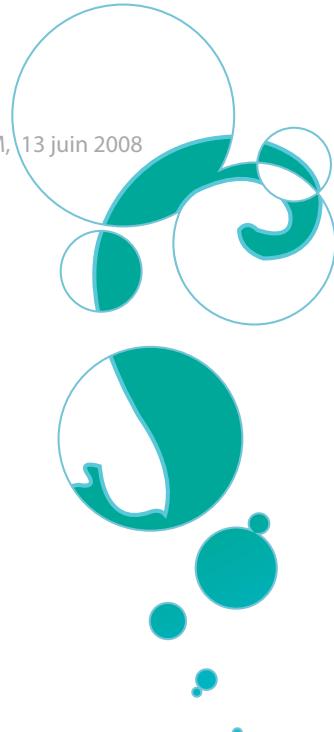


Theatre de L'hippocampe

Rapport final

Gaëlle Comment / Carine Gauthier / Joëlle Dind / Nhung Tang





Index

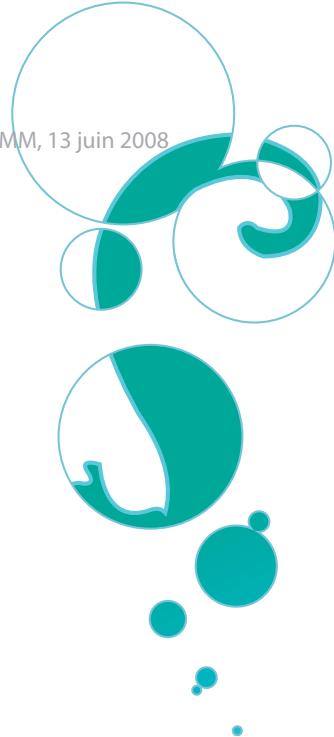
I. Gestion de projet	3
1. Proposition de projet	4
2. Dossier de planification	8
3. Document de suivi	13
.	.
II. Modélisation de traitement	16
1. Diagramme de classe	17
2. Schéma relationnel	18
3. Diagrammes de séquences	19
III. Design Document	49
1. Introduction	50
2. Approche marketing	51
3. Ergonomie, Design et Architecture	52
4. Amélioration	57
IV. Manuel d'installation	58
1. Arborescence des fichiers	59
2. Séparation traitements	65
3. Complément	66
4. Marche à suivre	66
V. Manuel d'utilisateur	67
1. Introduction	68
2. Connexion en tant qu'utilisateur	69
3. Connexion en tant qu'administrateur	84
VI. Rapport de terminaison	85
1. Les ressources	86
2. Problèmes rencontrés	87
3. Conseils et remarques	88



I. Gestion de Projet

1. Proposition de projet	4
2. Dossier de planification	8
2.1 WBS	9
2.2 Tableau des livrables et des délais	10
2.3 Schéma de PERT	11
2.4 GANTT	12
3. Document de suivi	13
3.1 Tableau des Jalons	14
3.2 Tableau Synoptique	15





1. Proposition de projet

Contexte > Idée > Enjeux > Objectifs
Livrables > Coûts > Délais > Risques



1.1 Contexte

Notre projet semestriel de deuxième année consiste en la réalisation d'une billetterie en ligne. En accord avec Monsieur Philipona nous avons décidé de nous concentrer sur un exemple concret, celui d'un théâtre à Lausanne : le théâtre de l'Hippocampe.

1.2 Idée

De nos jours, chaque minute d'une journée est comptée. Un théâtre doit se moderniser pour conserver sa clientèle actuelle et s'ouvrir à une nouvelle. L'utilisation d'un système très rependu dans l'événementiel est nécessaire : la billetterie en ligne ! La vente de billets se fait actuellement au guichet du théâtre. Les clients doivent se déplacer, plusieurs jours à l'avance. Revenir plusieurs fois afin de connaître les dates et les horaires des représentations, puis une deuxième pour réserver, et enfin assister à la représentation. La billetterie en ligne devra permettre à un utilisateur d'internet intéressé par un spectacle de :

- Consulter les différentes dates et horaires du spectacle
- Réserver et acheter des places de spectacle

D'autre part, la billetterie en ligne donnera la possibilité de :

- Modifier les informations, données, les changements.

1.3 Enjeux

Il est nécessaire de respecter le temps imparti de 2 semaines, de remplir les objectifs édicté par le projet, ainsi que de convaincre notre client de la facilité d'utilisation et de la cohérence de notre site.

1.4 Objectifs

Le principal but de notre projet est de remplir les objectifs du travail de semestre.

A la fin du projet, nous souhaitons que le site soit implanté dans un cadre réel, qui puisse être différent d'un théâtre. Nous ne traiterons pas de la publication des billets, nous proposons que ceux-ci puissent être imprimables par l'utilisateur et vérifiés au moyen d'un code barre sur place lors de la représentation.





1.5 Livrables

Les objectifs se décomposent en 4 phases principales :

- Gestion de Projet (Proposition de projet, WBS, Tableau de tâches, PERT, Gantt Ressources, Gantt Livrables, Tableau synoptique, Planning jalons, rapport de terminaison)
- La modélisation (Maquette graphique, arborescence site, design document, Domain model, diagramme classe, description attributs, diagrammes séquences)
- La programmation (Graphisme, Base de donnée, code, Tests, mise en ligne)
- La documentation (Mode d'emploi, commentaires PHP doc, Rapport final, Présentation).

1.6 Coûts

- Gestion de Projet	7'800,-CHF
- Modélisation	6'600,-CHF
- Programmation	23'400,-CHF
- Documentation	9'600,-CHF

Soit un total de **47'400,-CHF**.

Prenant en compte un salaire horaire de 150 francs par collaborateur.

1.7 Délais

Notre projet a été estimé à 64 jours/hommes, cependant nous estimons le nombre de jours réels à 41 jours hommes soit 328 heures. Nous nous engageons à rendre notre projet le 13 juin ainsi qu'à réaliser une présentation.



1.8 Risques

Les risques peuvent être classés selon 3 niveaux :

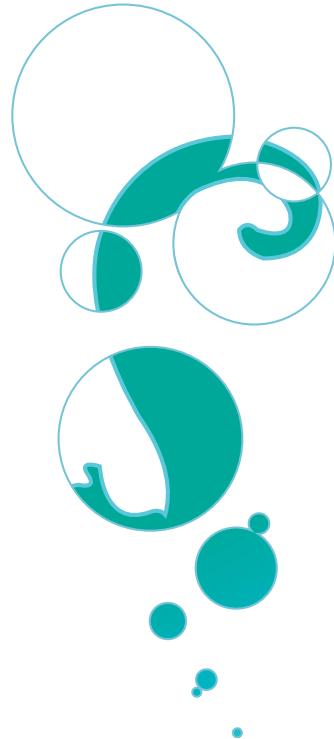
-  : une faible probabilité qu'il se produise
-  : un risque potentiel
-  : une très forte chance que le risque se produise

Tout d'abord, il est nécessaire de s'assurer que la direction du théâtre souhaite que les représentations puissent être consultées et réservées sur internet, de même que les employés seront d'accord de faire la promotion du nouveau site.

Il est important pour le théâtre d'adapter sa présentation de produit au marché actuel. D'autre part, nous supposons que le théâtre acceptera d'avoir recours à un centre de crédit afin de procéder aux vérifications des données utilisateurs concernant la validité de leur carte bancaire.

	Risques
Perte de Données	
Ressources (absence, maladie,...)	  
Mauvaise estimation de la durée des Tâches	
Mauvaise estimation de la complexité des tâches	



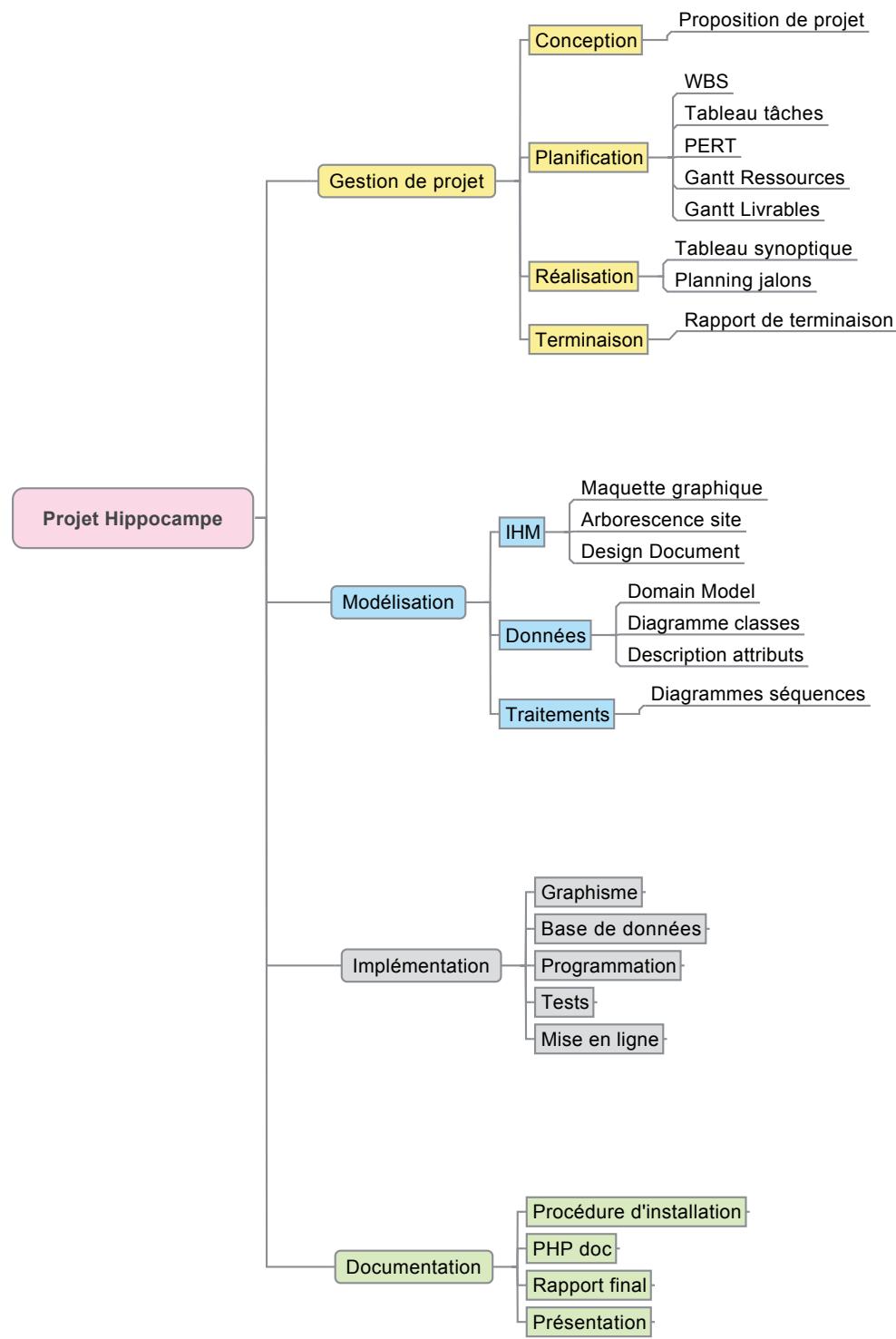


2. Dossier de planification

WBS > Table des livrables > PERT > GANTT



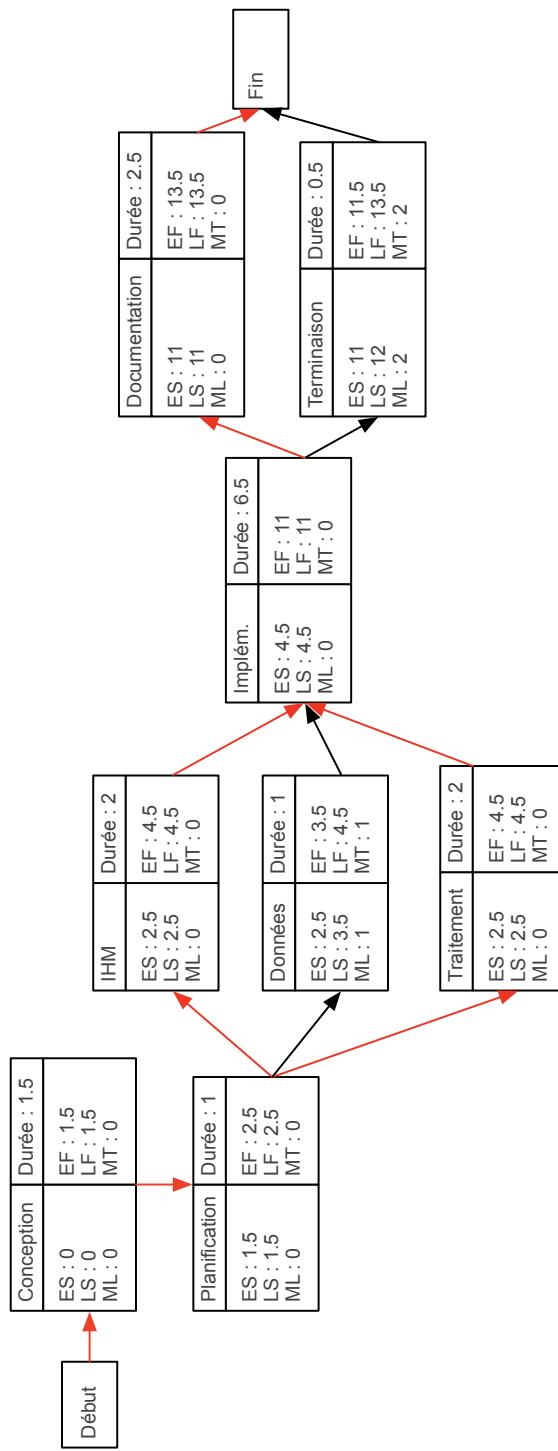
2.1 WBS



2.2 Tableau des livrables, coûts et des délais

WBS-Code	Phase	Tâche	Durée	Ressources Affectées	Coûts (jour/homme)	Coûts (CHF)
	<i>1 Gestion de projet</i>					
1.1		<i>Conception</i>				
1.1.1		Proposition de projet	1 jour 0.5 jour	Carine Gauthier Nhung Tang		
			total 1.5 jours		1.5	1800
1.2		<i>Planification</i>				
1.2.1		Documents de planifications	1 jour 1 jour 1 jour	Nhung Tang Gaëlle Comment Carine Gauthier		
			total 3 jours		3	3600
1.3		<i>Réalisation</i>				
1.3.1		Tableau synoptique	0.5 jour	Gaëlle Comment	0.5	600
1.3.2		Planning jalons	0.5 jour	Carine Gauthier	0.5	600
1.4		<i>Terminaison</i>				
1.4.1		Rapport de terminaison	0.25 0.25 0.25 0.25	Nhung Tang Carine Gauthier Gaëlle Comment Joëlle Dind		
			total 1 jour		1	1200
2	<i>Modélisation</i>					
2.1		<i>IHM</i>				
2.1.1		Maquette graphique	2 jour	Nhung Tang	2	2400
2.1.2		Arborescence site	0.5 jour	Nhung Tang	0.5	600
2.1.3		Design Document	1 jour	Carine Gauthier	1	1200
2.2		<i>Données</i>				
2.2.1		Domain Model	0.5 jour	Carine Gauthier	0.5	
2.2.2		Diagramme classes	0.5 jour	Carine Gauthier	0.5	
2.2.3		Description attributs	0.5 jour	Joëlle Dind	0.5	
2.3		<i>Traitements</i>				
2.3.1		Diagramme séquences	1 jour 1 jour	Gaëlle Comment Carine Gauthier		
			total 2 jours		2	2400
3	<i>Implémentation</i>					
3.1		Graphisme	3 jours	Nhung Tang	3	3600
3.2		Base de données	0.5 jour 0.5 jour	Nhung Tang Gaëlle Comment		
			total 1 jour		0.5	600
3.3		Programmation	4 jours 4 jours 4 jours	Gaëlle Comment Carine Gauthier Joëlle Dind		
			total 12 jours		12	14400
3.4		Mise en ligne	1 jour 1 jour	Nhung Tang Gaëlle Comment		
			total 2 jours		2	2400
3.5		Tests	0.5 jours 0.5 jours 0.5 jours 0.5 jours	Carine Gauthier Gaëlle Comment Joëlle Dind Nhung Tang		
			total 2 jours		2	2400
4	<i>Documentation</i>					
4.1		Procédure d'installation	2 jours	Joëlle Dind	2	2400
4.2		PHP doc	1 jour 1 jour	Gaëlle Comment Carine Gauthier	1	1200
			total 2 jours		2	2400
4.3		Rapport final	0.5 jour 0.5 jour 0.5 jour 0.5 jour	Carine Gauthier Gaëlle Comment Joëlle Dind Nhung Tang		
			total 2 jours		2	2400
4.4		Présentation	0.25 jours 0.25 jours 0.25 jours 0.25 jours	Carine Gauthier Gaëlle Comment Joëlle Dind Nhung Tang		
			total 1 jour		1	1200
				TOTAUX	41	47400

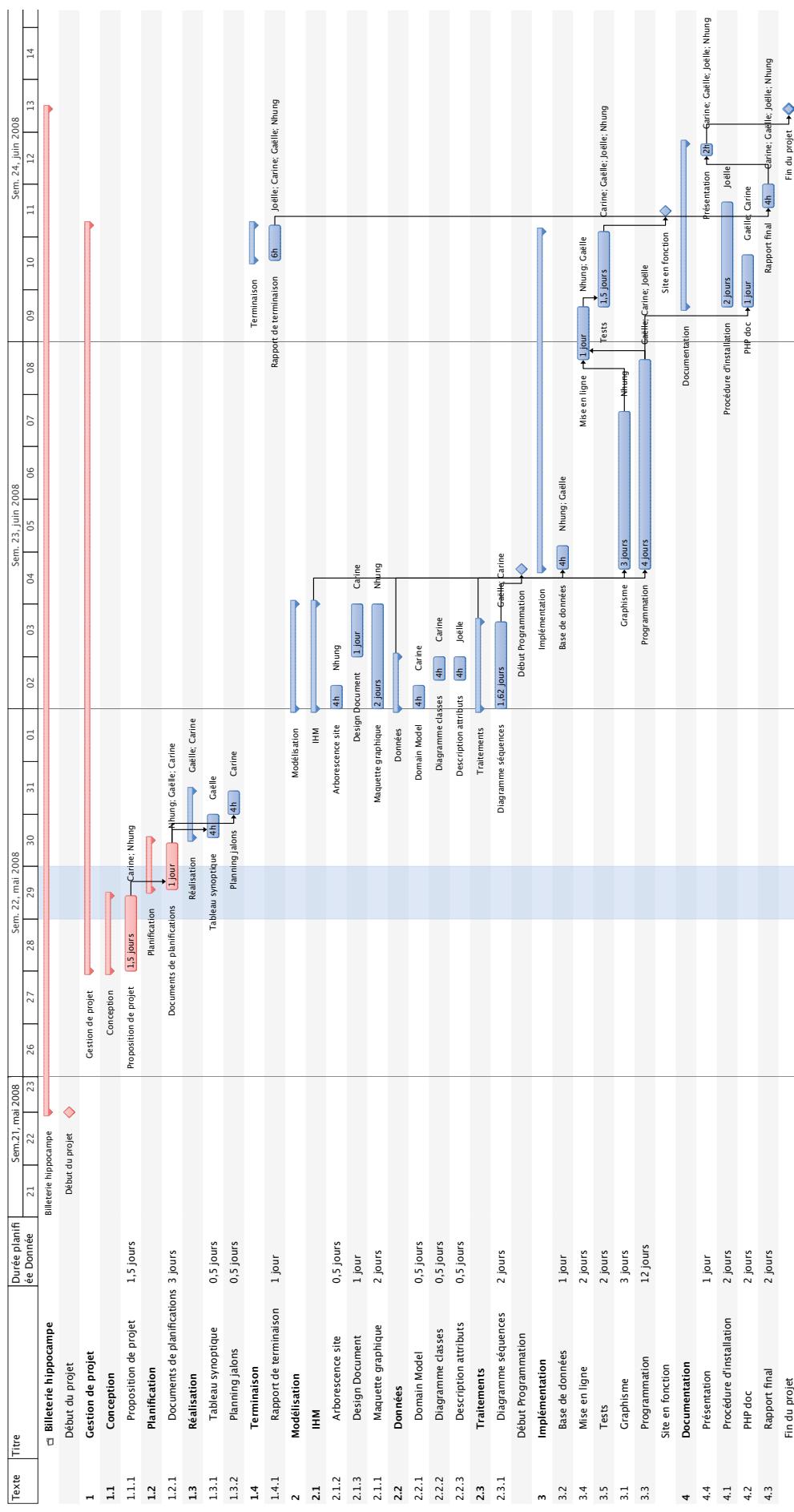
2.3 Schéma de PERT

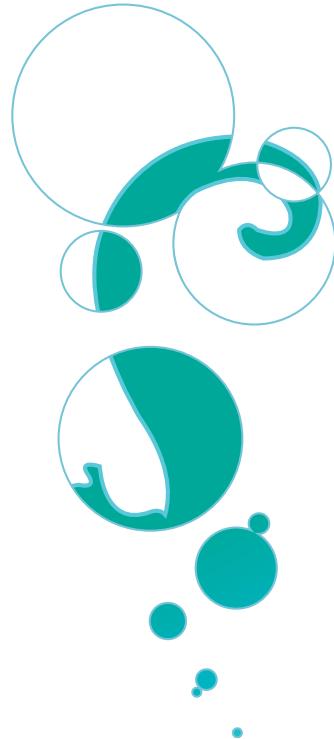


Toutes les flèches sont de relation Fin-Début



2.4 GANTT

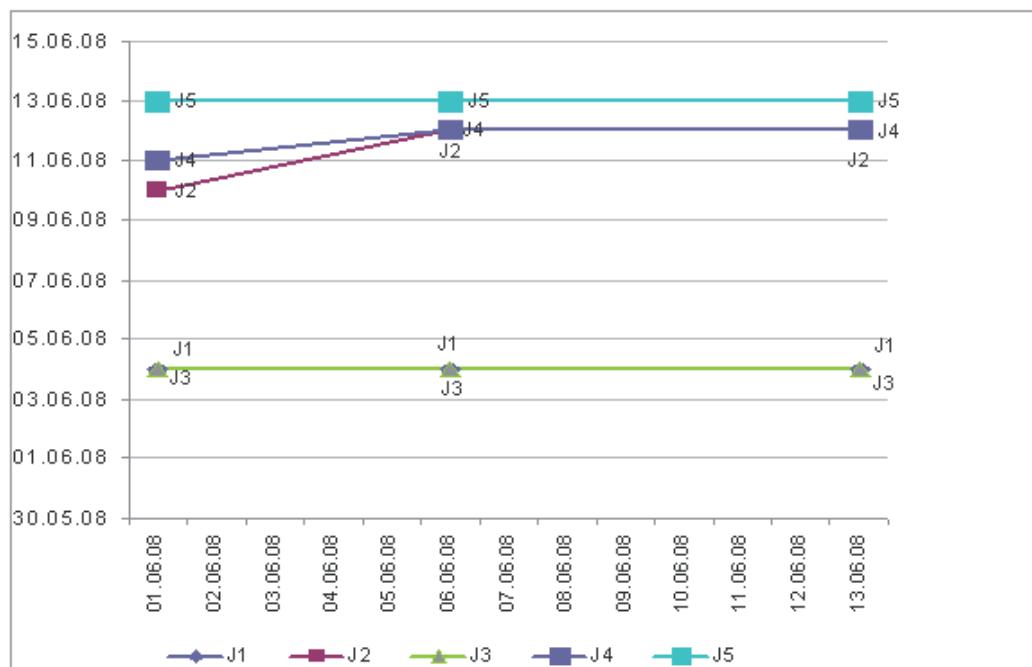




3. Document de suivi



3.1 Tableau des Jalons





3.2 Tableau Synoptique

Livrables	Responsable	Etat au 1 juin	Etat au 6 juin	Etat au 13 juin	Remarque
Projet général	Carine Gauthier	vert	orange	rouge	Du 23 mai au 4 juin, absence maladie de Joëlle. Entraîne retard dans l'implémentation
	Nhung Tang	vert	rouge	rouge	
	Joëlle Dind	vert	orange	orange	
	Gaëlle Comment	vert	orange	rouge	
Gestion de projet					
Proposition de projet	Carine Gauthier	vert	<i>finis</i>	<i>finis</i>	
Documents de planification	Nhung Tang	vert	<i>finis</i>	<i>finis</i>	
Tableau synoptique	Gaëlle Comment	vert	vert	vert	
Planning jalons	Carine Gauthier	vert	vert	vert	
Rapport de terminaison	Joëlle Dind	vert	vert	vert	
Modélisation					
IHM					
Maquette graphique	Nhung Tang	vert	vert	<i>finis</i>	
Arborescence site	Nhung Tang	vert	vert	<i>finis</i>	
Design Document	Carine Gauthier	vert	vert	<i>finis</i>	
Données					
Domain Model	Carine Gauthier	vert	vert	<i>finis</i>	
Diagramme classes	Carine Gauthier	vert	vert	<i>finis</i>	
Description attributs	Joëlle Dind	vert	orange	<i>finis</i>	
Traitements					
Diagramme séquences	Gaëlle Comment	vert	vert	<i>finis</i>	
Implémentation					
Graphisme	Nhung Tang	vert	vert	<i>finis</i>	
Base de données	Gaëlle Comment	vert	vert	<i>finis</i>	
Programmation	Gaëlle Comment	vert	rouge	rouge	Difficulté mise en route
Mise en ligne	Nhung Tang	vert	vert	vert	
Tests	Carine Gauthier	vert	vert	orange	
Documentation					
Procédure d'installation / utilisation	Joëlle Dind	vert	vert	orange	
PHP doc	Gaëlle Comment	vert	vert	orange	
Rapport final	Carine Gauthier	vert	vert	orange	
Présentation	Nhung Tang	vert	vert	orange	



II. Modélisation des données et traitements

1. Diagramme de classe 17

2. Schéma relationnel 18

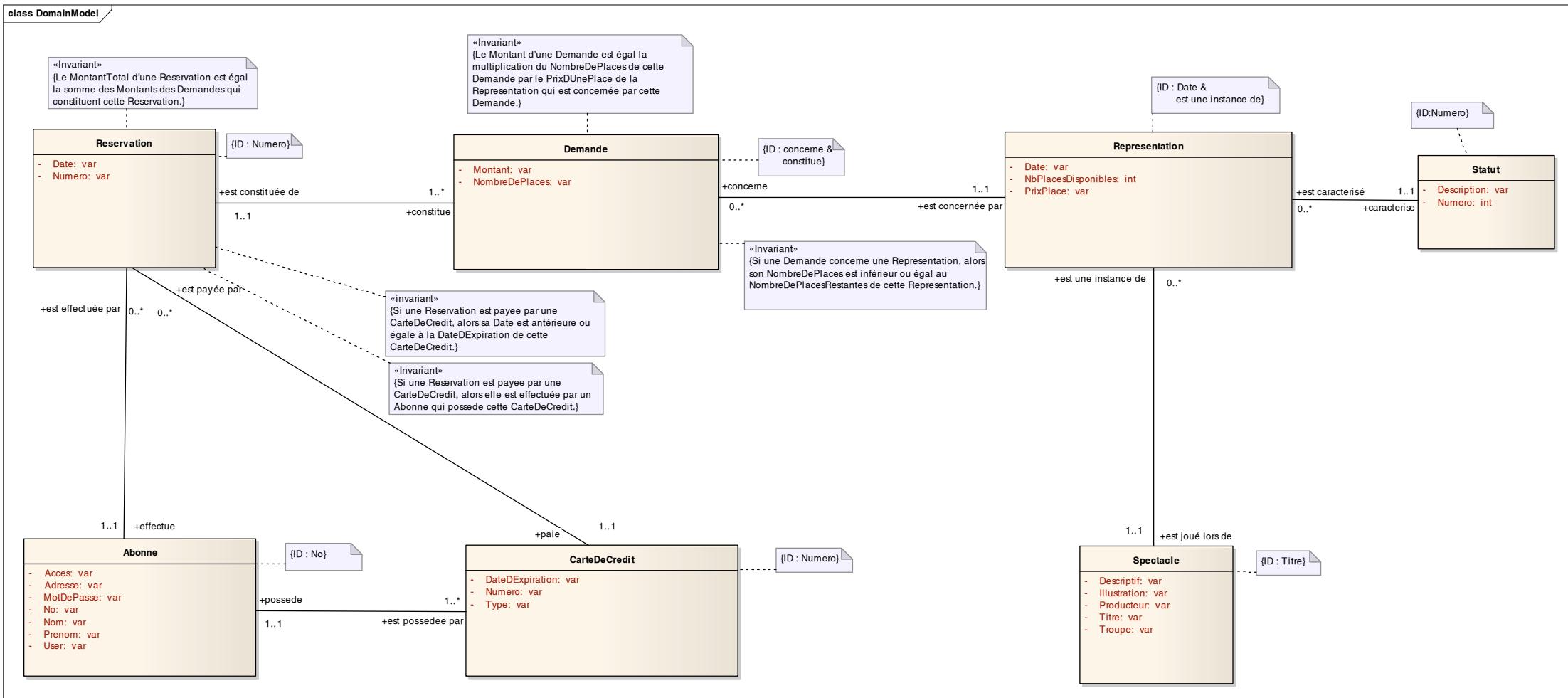
3. Diagrammes de séquences 19

Messages

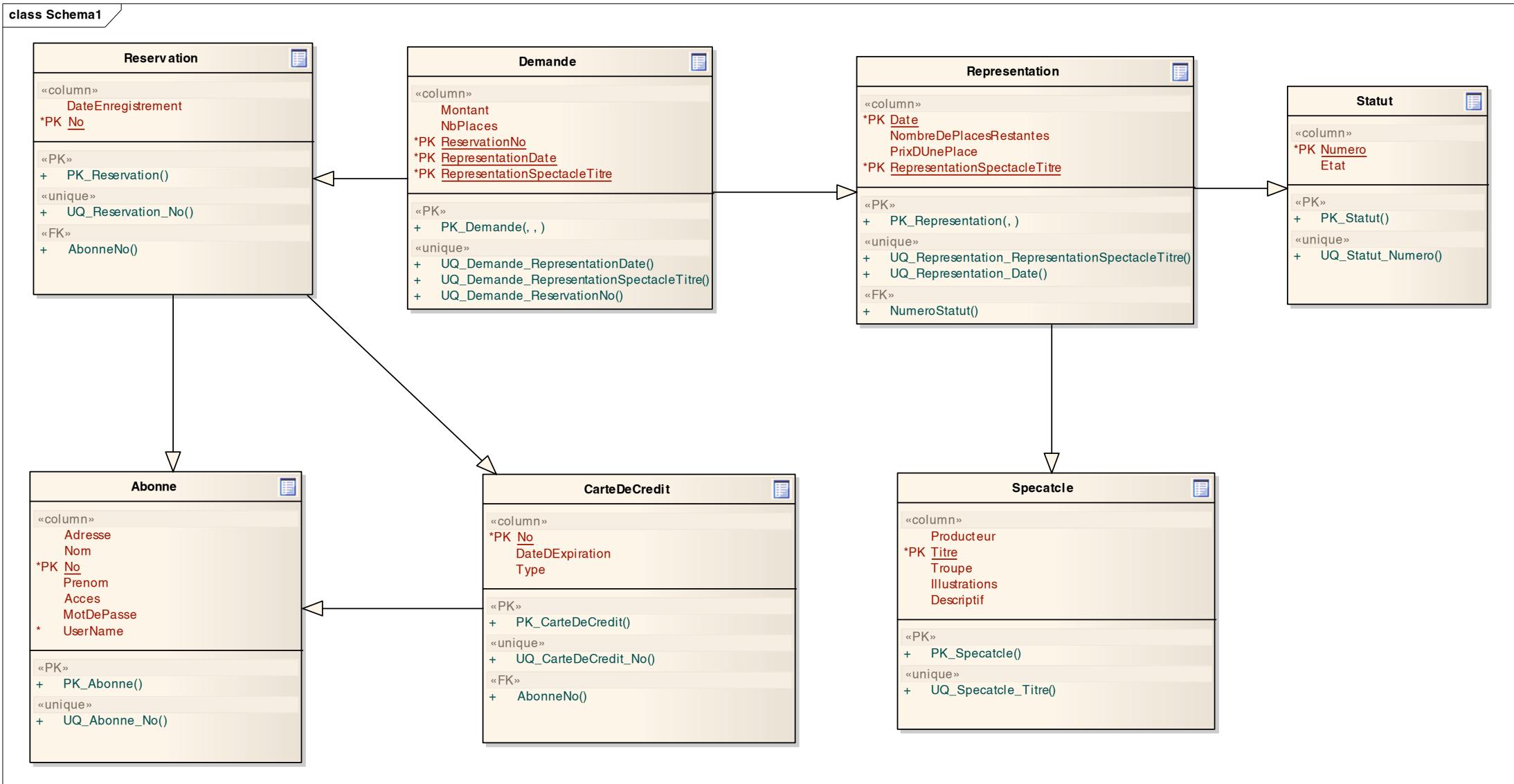
Méthodes



1. Diagramme De Classe

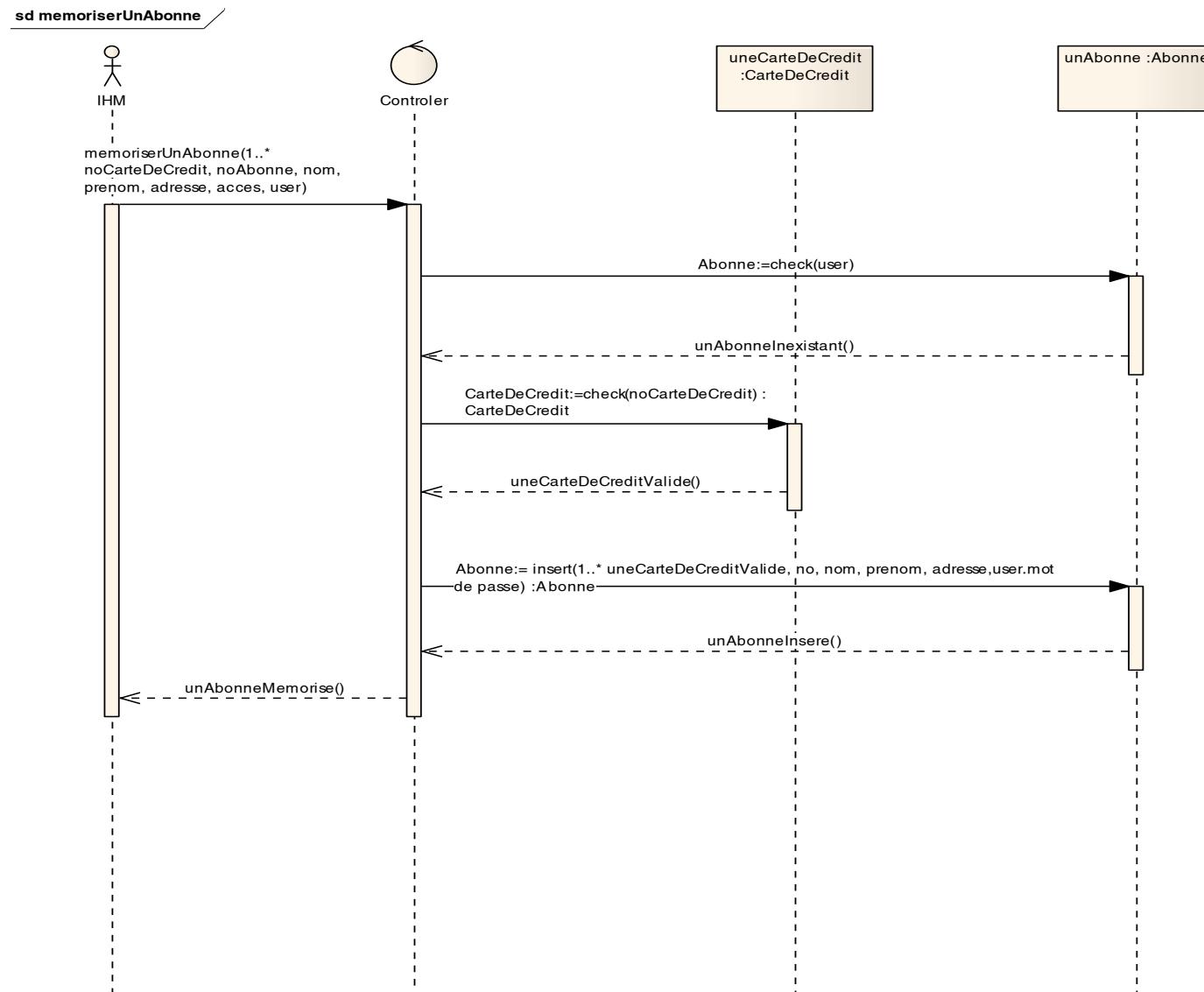


2. Schéma Relationnel



3. Diagramme De Séquence

Mémoriser unAbonné



Les Messages : mémoriserUnAbonné

MESSAGE	noCarteDeCredit
CONTIENT	un Numéro de même type que Numero de CarteDeCredit
MESSAGE	uneCarteDeCreditValide
CONTIENT	une CarteDeCredit
SUPPOSE	uneCarteDeCreditValide appartient à l'ensemble des Cartes de crédit persistantes
MESSAGE	Une CarteDeCreditInvalide
CONTIENT	Néant
SUPPOSE	uneCarteDeCreditInvalide n'appartient pas à l'ensemble des Cartes de crédit persistantes

MESSAGE	User
CONTIENT	un user de même type que user de Abonne
MESSAGE	unAbonneInexistant
CONTIENT	Néant
SUPPOSE	unAbonneInexistant n'appartient pas à l'ensemble des Abonnés persistants
MESSAGE	unAbonneExistant
CONTIENT	un Abonné
SUPPOSE	unAbonneExistant appartient à l'ensemble des Abonnés persistants

MESSAGE	nom
CONTIENT	un Nom de même type que Nom de Abonne
MESSAGE	prenom
CONTIENT	un Prenom de même type que Prenom de Abonne
MESSAGE	adresse
CONTIENT	une Adresse de même type que Adresse de Abonne
MESSAGE	mot de passe
CONTIENT	un Mot de passe de même type que Mot de passe de Abonne

MESSAGE	unAbonneInsere
CONTIENT	un Abonné

Les Méthodes : mémoriserUnAbonné

checkCarteDeCredit

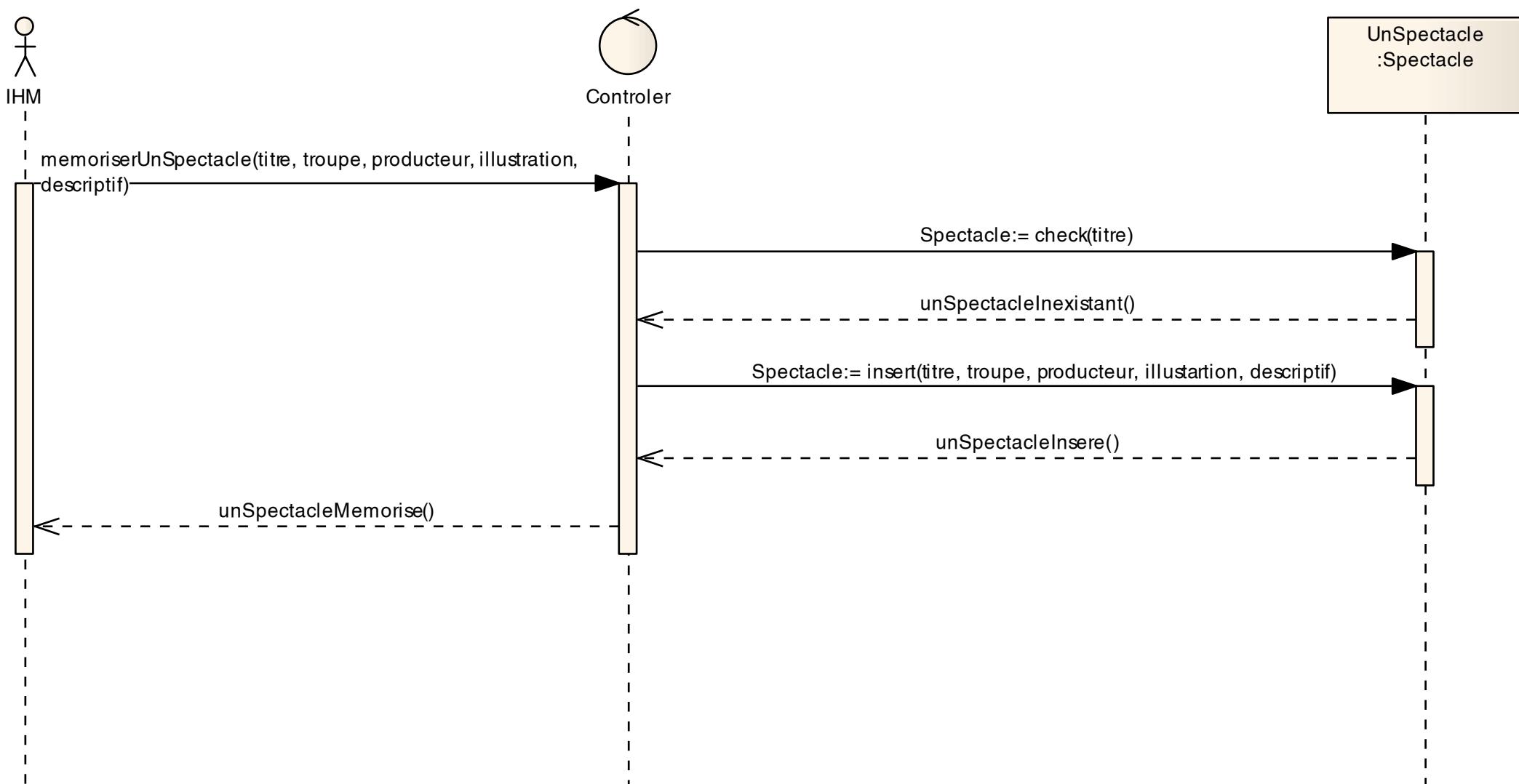
OBJECTIF	Vérifier qu'il existe un abonné persistant basé sur le numéro de carte de crédit fourni
RECOIT	noCarteDeCredit
GENERE	<p>uneCarteDeCreditValide</p> <p>ou</p> <p>uneCarteDeCreditInvalide</p>
REGLE	<p>S'il existe un Abonné dont le numéro est égal à noCarteDeCredit reçu,</p> <p>Alors génère uneCarteDeCreditValide correspondant à la Carte de Crédit consultée,</p> <p>Sinon génère uneCarteDeCreditInvalide</p>

checkUser

OBJECTIF	Vérifier qu'il existe un abonné persistant basé sur le user fourni
RECOIT	User
GENERE	<p>unAbonneInexistante</p> <p>ou</p> <p>unAbonneExistant</p>
REGLE	<p>S'il existe un Abonné dont le user est égal au user reçu,</p> <p>Alors génère unAbonneExistant,</p> <p>Sinon génère unAbonneInexistante</p>

insertAbonne

OBJECTIF	Créer un nouvel abonné
RECOIT	uneCarteDeCreditValide no nom prenom adresse user mot de passé
GENERE	unAbonneInsere
REGLE	Créer un nouvel abonné avec le user correspondant à user reçu

Mémoriser unSpectacle


Les Messages : mémoriserUnSpectacle

MESSAGE	titre
CONTIENT	un Titre de même type que Titre de Spectacle
MESSAGE	unSpectacleInexistant
CONTIENT	Néant
SUPPOSE	unSpectacleInexistant n'appartient pas à l'ensemble des Spectacles persistants
MESSAGE	unSpectacleExistant
CONTIENT	un Spectacle
SUPPOSE	unSpectacleExistant appartient à l'ensemble des Spectacles persistants

MESSAGE	troupe
CONTIENT	une Troupe de même type que Troupe de Spectacle
MESSAGE	producteur
CONTIENT	un Producteur de même type que Producteur de Spectacle
MESSAGE	illustration
CONTIENT	une Illustration de même type que Illustration de Spectacle
MESSAGE	descriptif
CONTIENT	un Descriptif de même type que Descriptif de Spectacle
MESSAGE	unSpectacleInsere
CONTIENT	un Spectacle

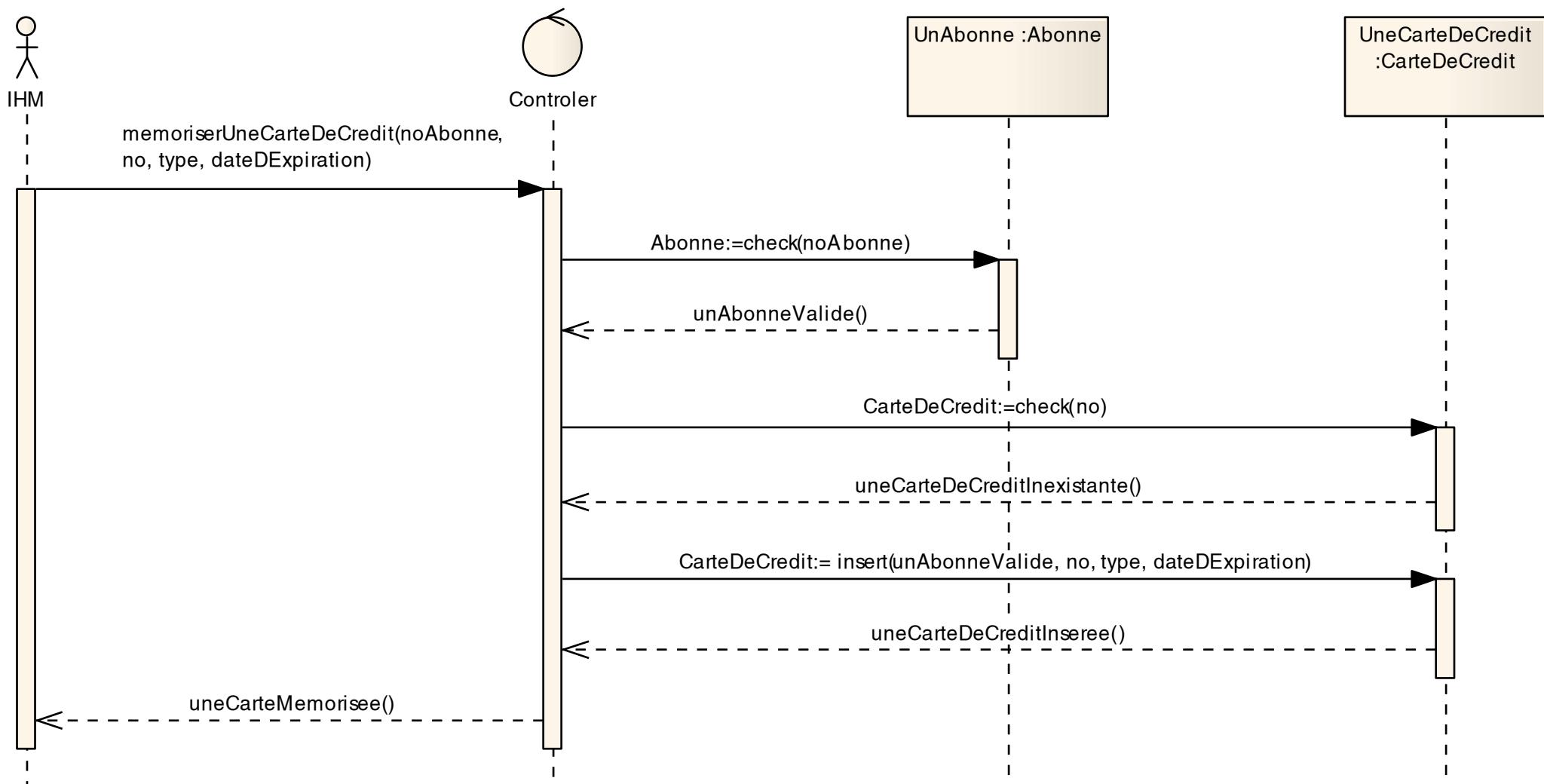
Les méthodes : mémoriserUnSpectacle

checkSpectacle

OBJECTIF	Vérifier qu'il existe un spectacle persistant basé sur le titre du spectacle
RECOIT	titre
GENERE	unSpectacleInexistant ou unSpectacleExistant
REGLE	S'il existe un Spectacle dont le titre est égal à titre reçu, Alors génère unSpectacleExistant correspondant au spectacle consulté, Sinon génère unSpectacleInexistant

insertSpectacle

OBJECTIF	Créer un nouveau spectacle
RECOIT	titre troupe producteur illustration descriptif
GENERE	unSpectacleInsere
REGLE	Créer un nouveau spectacle avec le titre correspondant à titre reçu

Mémoriser uneCarteDeCrédit


Les Messages : memoriserUneCarteDeCredit

MESSAGE	noAbonne
CONTIENT	un Numéro de même type que Numéro de Abonne
MESSAGE	unAbonneValide
CONTIENT	un Abonne
SUPPOSE	unAbonneValide appartient à l'ensemble des Abonnés persistants
MESSAGE	unAbonneInvalide
CONTIENT	Néant
SUPPOSE	unAbonneValide n'appartient pas à l'ensemble des Abonnés persistants

MESSAGE	no
CONTIENT	un Numéro de même type que Numéro de CarteDeCredit
MESSAGE	uneCarteDeCreditInexistante
CONTIENT	Néant
SUPPOSE	uneCarteDeCreditInexistante n'appartient pas à l'ensemble des Cartes de Crédit persistantes
MESSAGE	uneCarteDeCreditExistante
CONTIENT	une CarteDeCredit
SUPPOSE	uneCarteDeCreditExistante appartient à l'ensemble des Cartes de Crédit persistantes

MESSAGE	type
CONTIENT	un Type de même type que Type de CarteDeCredit
MESSAGE	dateDExpiration
CONTIENT	une Date de même type que DateDExpiration de CarteDeCredit
MESSAGE	uneCarteDeCreditInseree
CONTIENT	uneCarteDeCredit

Les Méthodes : memoriserUneCarteDeCredit

checkAbonne

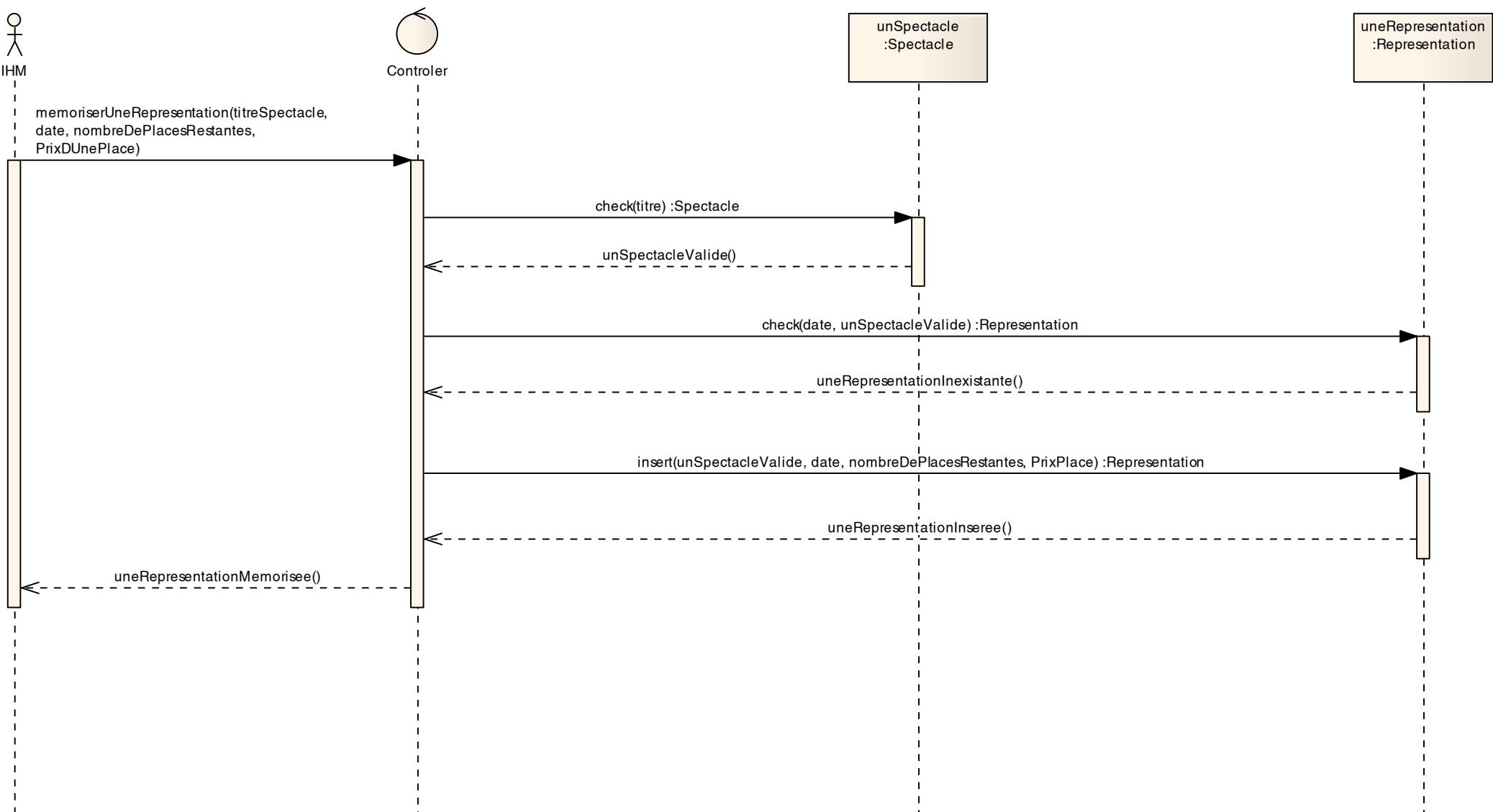
OBJECTIF	Vérifier qu'il existe un abonné persistant basé sur le numéro de l'abonné fourni
RECOIT	noAbonne
GENERE	unAbonneValide ou unAbonneInvalide
REGLE	S'il existe un Abonné dont le numéro est égal à noAbonne reçu, Alors génère unAbonneValide correspondant à l'abonné consulté, Sinon génère unAbonneInvalide

checkCarteDeCredit

OBJECTIF	Vérifier qu'il existe une carte de crédit persistante basée sur le numéro de carte de crédit fourni
RECOIT	no
GENERE	uneCarteDeCreditInexistante ou uneCarteDeCreditExistante
REGLE	S'il existe une carte de crédit dont le numéro est égal à no reçu, Alors génère uneCarteDeCreditExistante correspondant à la Carte de Crédit consultée, Sinon génère uneCarteDeCreditInexistante

insertCarteDeCredit

OBJECTIF	Créer une nouvelle carte de crédit
RECOIT	unAbonneValide no type dateDExpiration
GENERE	uneCarteDeCreditInseree
REGLE	Créer une nouvelle carte de crédit avec le numéro correspondant à no reçu

Mémoriser UneReprésentation


Les Messages : memoriserUneRepresentation

MESSAGE	titre
CONTIENT	un Titre de même type que Titre de Spectacle
MESSAGE	unSpectacleValide
CONTIENT	un Spectacle
SUPPOSE	unSpectacleValide appartient à l'ensemble des Spectacles persistants
MESSAGE	unSpectacleInvalide
CONTIENT	Néant
SUPPOSE	unSpectacleInvalide n'appartient pas à l'ensemble des Spectacles persistants

MESSAGE	date
CONTIENT	une Date de même type que Date de Representation
MESSAGE	uneRepresentationInexistante
CONTIENT	Néant
SUPPOSE	uneRepresentationInexistante n'appartient pas à l'ensemble des Representations persistantes
MESSAGE	uneRepresentationExistante
CONTIENT	uneRepresentation
SUPPOSE	uneRepresentationExistante appartient à l'ensemble des Representations persistantes

MESSAGE	nombreDePlacesRestantes
CONTIENT	un Nombre de places de même type que NombreDePlacesRestantes de Representation
MESSAGE	PrixPlace
CONTIENT	Un Prix de même type que PrixPlace de Representation
MESSAGE	uneRepresentationInseree
CONTIENT	une Representation

Les Méthodes : memoriserUneRepresentation

checkSpectacle

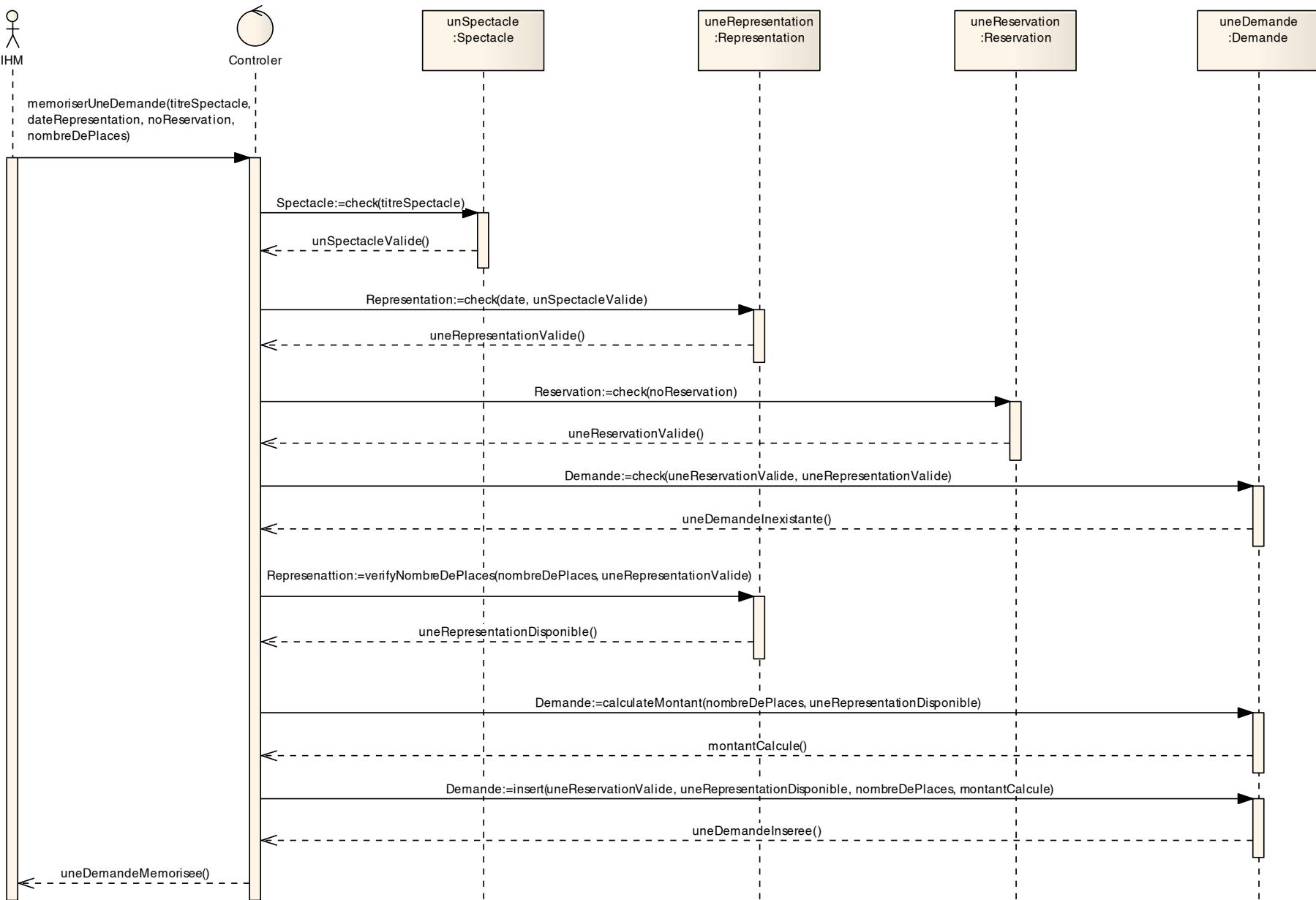
OBJECTIF	Vérifier qu'il existe un spectacle persistant basé sur le titre du spectacle fourni
RECOIT	titreSpectacle
GENERE	unSpectacleValide ou unSpectacleInvalid
REGLE	S'il existe un spectacle dont le titre est égal à titreSpectacle reçu, Alors génère unSpectacleValide correspondant au spectacle consulté, Sinon génère unSpectacleInvalid

checkRepresentation

OBJECTIF	Vérifier qu'il existe une representation persistante basée sur la date de la représentation et le spectacle fournis
RECOIT	date unSpectacleValide
GENERE	uneRepresentationValide ou uneRepresentationInvalid
REGLE	S'il existe une representation dont la date est égale à date reçue et correspondant à unSpectacleValide, Alors génère uneRepresentationValide correspondant à la représentation consultée, Sinon génère uneRepresentationInvalid

insertRepresentation

OBJECTIF	Créer une nouvelle representation
RECOIT	unSpectacleValide date nombreDePlacesRestantes PrixUnePlace
GENERE	uneRepresentationInseree
REGLE	Créer une nouvelle représentation avec la date correspondant à date reçue, Crée une association « est une instance de » entre : -la nouvelle représentation créée et -unSpectacleValideReçu



Les Messages : mémoriserUneDemande

MESSAGE	titreSpectacle
CONTIENT	un Titre de même type que Titre de Spectacle
MESSAGE	unSpectacleValide
CONTIENT	un Spectacle
SUPPOSE	unSpectacleValide appartient à l'ensemble des Spectacles persistants
MESSAGE	unSpectacleInvalide
CONTIENT	Néant
SUPPOSE	unSpectacleInvalide n'appartient pas à l'ensemble des Spectacles persistants

MESSAGE	date
CONTIENT	une Date de même type que Date de Representation
MESSAGE	uneRepresentationValide
CONTIENT	une Representation
SUPPOSE	uneRepresentationValide appartient à l'ensemble des Representations persistantes
MESSAGE	uneReprsentationInvalide
CONTIENT	Néant
SUPPOSE	uneReprsentationInvalide n'appartient pas à l'ensemble des Representations persistantes

MESSAGE	noReservation
CONTIENT	un Numéro de même type que Numéro de Reservation
MESSAGE	uneReservationValide
CONTIENT	une Reservation
SUPPOSE	uneReservationValide appartient à l'ensemble des Réservations persistantes
MESSAGE	uneReservationInvalide
CONTIENT	Néant
SUPPOSE	uneReservationInvalide n'appartient pas à l'ensemble des Reservations persistantes

MESSAGE	uneDemandeInexistante
CONTIENT	Néant
SUPPOSE	uneDemandeInexistante n'appartient pas à l'ensemble des Demandes persistantes
MESSAGE	uneDemandeExisteante
CONTIENT	une Demande
SUPPOSE	uneDemandeExisteante appartient à l'ensemble des Demandes persistantes

MESSAGE	nombreDePlaces
CONTIENT	un Nombre de places de même type que nombreDePlaces de Demande
MESSAGE	uneRepresentationDisponible
CONTIENT	une Representation
SUPPOSE	nombreDePlaces de uneDemandeExisteante est inférieur ou égal à NombreDePlacesRestantes de uneRepresentationValide
MESSAGE	uneRepresentationIndisponible
CONTIENT	uneRepresentation
SUPPOSE	nombreDePlaces de uneDemandeExisteante est supérieur à NombreDePlacesRestantes de uneRepresentationValide

MESSAGE	montantCalcule
CONTIENT	un Montant de même type que Montant de Demande
SUPPOSE	montantCalcule est égal à NombreDePlaces de uneDemandeValide multiplié par PrixPlace de uneRepresentationDisponible

MESSAGE	uneDemandeInseree
CONTIENT	une Demande

Les Méthodes : mémoriserUneDemande

checkSpectacle

OBJECTIF	Vérifier qu'il existe un spectacle persistant basé sur le titre du spectacle fourni
RECOIT	titreSpectacle
GENERE	unSpectacleValide ou unSpectacleInvalide
REGLE	S'il existe un spectacle dont le titre est égal à titreSpectacle reçu, Alors génère unSpectacleValide correspondant au spectacle consulté, Sinon génère unSpectacleInvalide

checkRepresentation

OBJECTIF	Vérifier qu'il existe une representation persistante basée sur la date de la représentation et le spectacle fournis
RECOIT	date unSpectacleValide
GENERE	uneRepresentationValide ou uneRepresentationInvalide
REGLE	S'il existe une representation dont la date est égale à date reçue et correspondant a unSpectacleValide, Alors génère uneRepresentationValide correspondant à la représentation consultée, Sinon génère uneRepresentationInvalide

checkReservation

OBJECTIF	Vérifier qu'il existe une réservation persistante basée sur le numéro de la réservation fourni
RECOIT	noReservation
GENERE	uneReservationValide ou uneReservationInvalide
REGLE	S'il existe une réservation dont le numéro est égal à noReservation reçu, Alors génère uneReservationValide correspondant au spectacle consulté, Sinon génère uneReservationInvalide

checkDemande

OBJECTIF	Vérifier qu'il existe une demande persistante basée sur la réservation et la représentation de la demande fournie
RECOIT	uneReservationValide uneRepresentationValide
GENERE	uneDemandeExistante ou uneDemandeInexistante
REGLE	S'il existe une demande dont la réservation correspond à uneReservationValide reçue et la représentation correspond à uneRepresentationValide reçue, Alors génère uneDemandeExistante correspondant à la demande consultée, Sinon génère uneDemandeInexistante

verifyNombreDePlaces

OBJECTIF	Vérifier la contrainte que le nombre de places disponibles de représentation est supérieur ou égal au nombre de places de demande
RECOIT	nombreDePlaces uneRepresentationValide
GENERE	uneRepresentationDisponible
GENERE	ou uneRepresentationIndisponible
REGLE	Si nombreDePlacesDisponibles de uneRepresentationValide est supérieur ou égal à nombreDePlaces reçu, Alors génère uneRepresentationDisponible correspondant à uneRepresentationValide, Sinon génère uneRepresentationInvalide

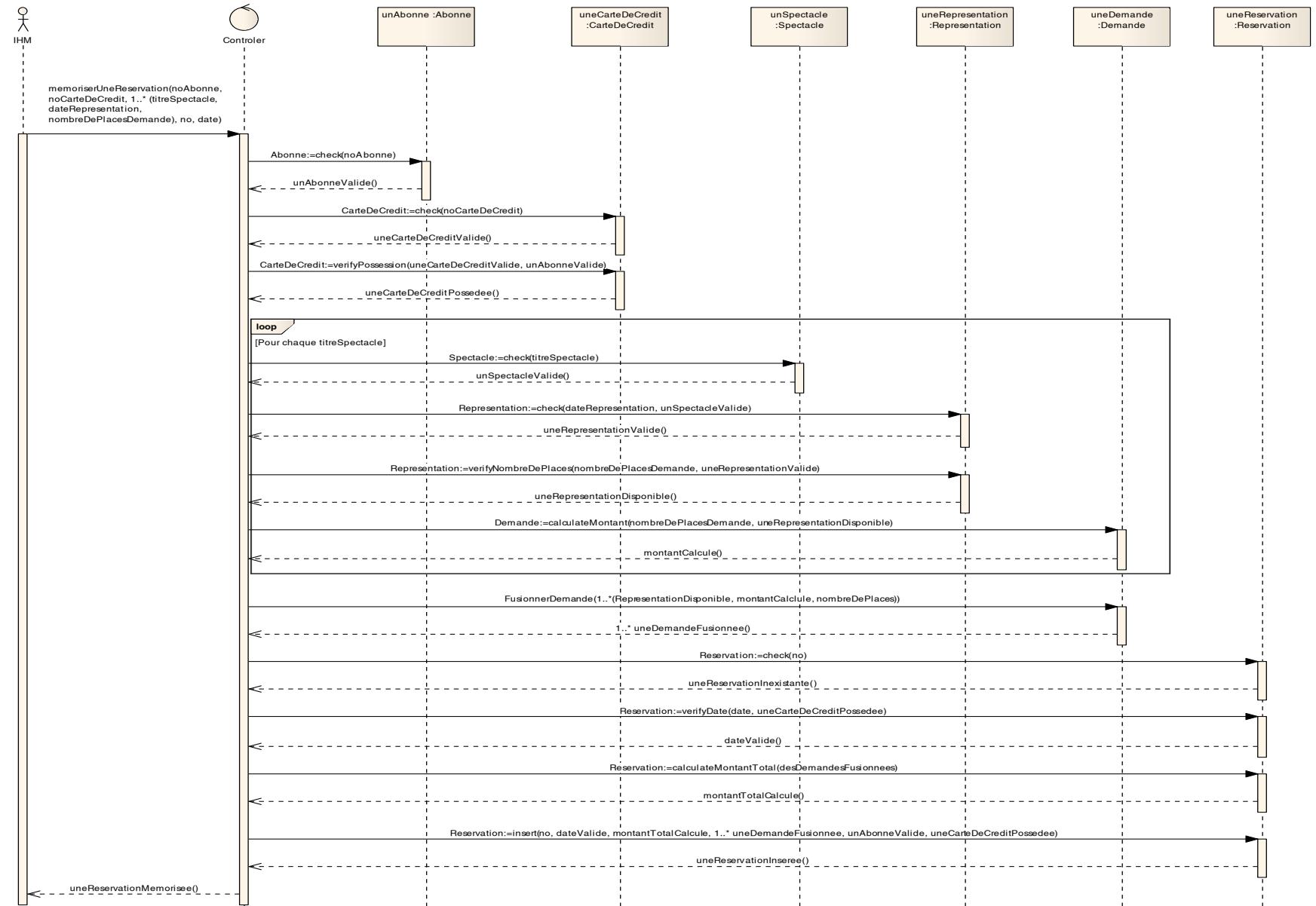
calculateMontant

OBBJECTIF	Calculer le montant de la réservation
RECOIT	nombreDePlaces uneRepresentationDisponible
GENERE	montantCalcule
REGLE	Le montant calculé est égal à la multiplication du nombreDePlaces de uneDemandeValide par le PrixPlace de uneRepresentationDisponible

insertDemande

OBJECTIF	Créer une nouvelle demande
RECOIT	uneReservationValide uneRepresentationDisponible nombreDePlaces montantCalcule
GENERE	uneDemandeInseree
REGLE	Crée une nouvelle demande

	<p>Crée une association « concerne » entre :</p> <ul style="list-style-type: none">-la nouvelle demande créée et-uneRepresentationDisponible reçue <p>Crée une association « constitue » entre :</p> <ul style="list-style-type: none">-la nouvelle demande créer et-uneReservationValide reçue
--	---

**Mémoriser
UneRéservation**


Les Messages. : mémoriserUneReservation

MESSAGE	NoAbonne
CONTIENT	un Numéro de même type que Numéro de Abonne
MESSAGE	unAbonneValide
CONTIENT	un Abonne
SUPPOSE	unAbonneValide appartient à l'ensemble des Abonnés persistants
MESSAGE	unAbonneInvalide
CONTIENT	Néant
SUPPOSE	unAbonneInvalide n'appartient pas à l'ensemble des Abonnés persistants

MESSAGE	NoCarteDeCredit
CONTIENT	un Numéro de même type que Numéro de CarteDeCredit
MESSAGE	uneCarteDeCreditValide
CONTIENT	une Carte De Credit
SUPPOSE	uneCarteDeCreditValide appartient à l'ensemble des Cartes de Crédit persistantes
MESSAGE	uneCarteDeCreditInvalide
CONTIENT	Néant
SUPPOSE	uneCarteDeCreditInvalide n'appartient pas à l'ensemble des Cartes de Crédit persistantes

MESSAGE	uneCarteDeCreditPossedee
CONTIENT	une Carte De Credit
SUPPOSE	uneCarteDeCreditPossedee appartient à unAbonneValide
MESSAGE	uneCarteDeCreditNonPossedee
CONTIENT	une Carte De Credit
SUPPOSE	uneCarteDeCreditPossedee n'appartient pas à unAbonneValide

MESSAGE	titreSpectacle
CONTIENT	un Titre de même type que Titre de Spectacle
MESSAGE	unSpectacleValide
CONTIENT	un Spectacle
SUPPOSE	unSpectacleValide appartient à l'ensemble des Spectacles persistants
MESSAGE	unSpectacleInvalide
CONTIENT	Néant
SUPPOSE	unSpectacleInvalide n'appartient pas à l'ensemble des Spectacles persistants

MESSAGE	dateRepresentation
CONTIENT	une Date de même type que Date de Representation
MESSAGE	uneRepresentationValide
CONTIENT	une Representation
SUPPOSE	uneRepresentationValide appartient à l'ensemble des Representations persistantes
MESSAGE	uneRepresentationInvalide
CONTIENT	Néant
SUPPOSE	uneRepresentationInvalide n'appartient pas à l'ensemble des Representations persistantes

MESSAGE	nombreDePlacesDemande
CONTIENT	un Nombre de places de même type que NombreDePlaces de Demande
MESSAGE	uneRepresentationDisponible
CONTIENT	une RepresentationValide
SUPPOSE	unNombreDePlaceDemandé appartient à une représentationValide
MESSAGE	uneRepresentationIndisponible
CONTIENT	une RepresentationValide
SUPPOSE	uneReprsentationIndisponible n'appartient pas à une représentationValide

MESSAGE	montantCalcule
CONTIENT	un Montant de même type que Montant de Demande
SUPPOSE	montantCalcule est égal à NombreDePlacesDemande de uneDemandeValide multiplié par PrixPlace de uneRepresentationDisponible

MESSAGE	nombreDePlacesDemande
CONTIENT	un Nombre de places de même type que NombreDePlaces de Demande
MESSAGE	uneDemandeFusionnee
CONTIENT	uneDemandeValide
SUPPOSE	Le MontantTotal de Reservation est égal à la somme des Montants des Demandes qui constituent uneReervationValide

MESSAGE	no
CONTIENT	un Numero de même type que Numero de Reservation
MESSAGE	uneReservationInexistante
CONTIENT	Néant
SUPPOSE	uneReservationValide n'appartient pas à l'ensemble des Reservation persistantes
MESSAGE	uneReservationExisteante
CONTIENT	uneReservation
SUPPOSE	uneReservationExisteante appartient à l'ensemble des Reservation persistantes

MESSAGE	date
CONTIENT	une Date de même type que Date de Reservation
MESSAGE	dateValide
CONTIENT	une Date
SUPPOSE	Date de uneReservation est antérieure ou égale à DateDExpiration de uneCartdeDeCreditPossedee
MESSAGE	dateInvalide
CONTIENT	une Date

SUPPOSE	Date de uneReservation n'est pas antérieure ou égale à DateDExpiration de uneCarteDeCreditPossedee
----------------	--

MESSAGE	uneReservationInseree
CONTIENT	uneReservation

Les Méthodes : memoriserUneReservation

checkAbonne

OBJECTIF	Vérifier qu'il existe un abonné persistant basé sur le numéro de l'abonné fourni
RECOIT	noAbonne
GENERE	unAbonneValide ou unAbonneInvalide
REGLE	S'il existe un Abonné dont le numéro est égal à noAbonne reçu, Alors génère unAbonneValide correspondant à l'abonné consulté, Sinon génère unAbonneInvalide

checkCarteDeCredit

OBJECTIF	Vérifier qu'il existe une carte de crédit persistante basée sur le numéro de carte de crédit fourni
RECOIT	noCarteDeCredit
GENERE	uneCarteDeCreditValide ou uneCarteDeCreditInvalide
REGLE	S'il existe une carte de crédit dont le numéro est égal à noCarteDeCredit reçu, Alors génère uneCarteDeCreditValide correspondant à la Carte de Crédit consultée, Sinon génère uneCarteDeCreditInvalide

verifyPossession

OBJECTIF	Vérifier la contrainte que la Carte de crédit valide fournie correspond à l'abonné valide fourni
RECOIT	uneCarteDeCreditValide unAbonneValide
GENERE	uneCarteDeCreditPossedee ou uneCarteDeCreditNonPossedee
REGLE	Si numero de uneCarteDeCreditValide fournie correspond à numero de unAbonneValide fourni, Alors génère uneCarteDeCreditPossedee correspondant à uneCarteDeCreditValide, Sinon génère uneCarteDeCreditNonPossedee

checkSpectacle

OBJECTIF	Vérifier qu'il existe un spectacle persistant basé sur le titre du spectacle fourni
RECOIT	titreSpectacle
GENERE	unSpectacleValide ou unSpectacleInvalide
REGLE	S'il existe un spectacle dont le titre est égal à titreSpectacle reçu, Alors génère unSpectacleValide correspondant au spectacle consulté, Sinon génère unSpectacleInvalide

checkRepresentation

OBJECTIF	Vérifier qu'il existe une representation persistante basée sur la date de la représentation et le spectacle fournis
RECOIT	dateRepresentation unSpectacleValide

GENERE	uneRepresentationValide ou uneRepresentationInvalidé
REGLE	S'il existe une representation dont la date est égale à dateRepresentation reçue et correspondant a unSpectacleValide, Alors génère uneRepresentationValide correspondant à la représentation consultée, Sinon génère uneRepresentationInvalidé

verifyNombreDePlaces

OBJECTIF	Vérifier la contrainte que le nombre de places disponibles de représentation est supérieur ou égal au nombre de places de demande
RECOIT	nombreDePlacesDemande uneRepresentationValide
GENERE	uneRepresentationDisponible ou uneRepresentationIndisponible
REGLE	Si nombreDePlacesDisponibles de uneRepresentationValide est supérieur ou égal à nombreDePlaces reçu, Alors génère uneRepresentationDisponible correspondant à uneRepresentationValide, Sinon génère uneRepresentationInvalidé

calculateMontant

OBJECTIF	Calculer le montant de la réservation
RECOIT	nombreDePlacesDemande uneRepresentationDisponible
GENERE	montantCalcule
REGLE	Le montant calculé est égal à la multiplication du nombreDePlaces de uneDemandeValide par le PrixPlace de uneRepresentationDisponible

fusionnerDemande

OBJECTIF	S'assurer qu'il n'existe pas plusieurs lignes concernant la même réservation, et le cas échéant, fusionner ces lignes en additionnant le montant de chaque demande respective.
RECOIT	Un ou plusieurs montantCalcule, une ou plusieurs représentationdisponible, et un ou plusieurs nombreDePlaces.
GENERE	Des lignes fusionnées
REGLE	<p>Pour chaque Demande reçue,</p> <p>S'il existe un autre montant que celui reçu avec la même demande,</p> <p>Alors</p> <ul style="list-style-type: none"> - Fusionne les deux montants reçus - Additionne les deux nb de places * Prix Place correspondantes. <p>Génère uneDemandeFusionnée comprenant :</p> <ul style="list-style-type: none"> - Chaque nb de Place reçu ou fusionné - Chaque Prix de place reçu ou fusionné-

checkReservation

OBJECTIF	Vérifier qu'il existe une réservation persistante basée sur le numéro de la réservation fourni
RECOIT	no
GENERE	<p>uneReservationInexistante</p> <p>ou</p> <p>uneReservationExistante</p>
REGLE	<p>S'il existe une réservation dont le numéro est égal à no reçu,</p> <p>Alors génère uneReservationExistante correspondant au spectacle consulté,</p> <p>Sinon génère uneReservationInexistante</p>

verifyDate

OBJECTIF	Vérifier la contrainte que la date de la réservation est antérieure ou égale à la date d'expiration de la carte de crédit
RECOIT	date

	uneCarteDeCreditPossedee
GENERE	dateValide ou dateInvalide
REGLE	Si date de uneRevervationValide est antérieure ou égale à DateDExpiration de uneCarteDeCreditPossedee, Alors génère dateValide correspondant à date reçue, Sinon génère dateInvalide

calculateMontantTotal

OBJECTIF	Vérifier que le montant est le montant des demandesFusionnees
RECOIT	desDemandesFusionnees
GENERE	montantTotalCalcule
REGLE	SI le montant reçu de Demande est égal au montantTotalCalcule desDemandesFusionnees, Alors génère unMontantTotalCalcule correspondant à l'addition des montants de Demande .

insertReservation

OBJECTIF	Créer une nouvelle reservation
RECOIT	no dateValide montantTotalCalcule uneDemandeFusionnee unAbonneValide uneCarteDeCreditPossedee
GENERE	uneRevervationInseree
REGLE	Crée une nouvelle reservation avec le numéro correspondant au no reçu

III. Design Doc

1. Introduction	50
2. Approche Marketing	51
2.1 Définition du thème (objectifs et limites)	51
2.2 Analyse de sites concurrents	51
2.3 Public cible	51
3. Ergonomie, Design et Architecture	52
3.1 Design Graphique	52
3.2 Ergonomie d'interface	54
3.3 Architecture du Contenu et Cohérence de la Navigation	55
4. Améliorations	57



1. Introduction

Ce document a pour but de présenter l'ergonomie de notre publication électronique, le design graphique et la structure de l'information de notre site. Dans le cadre de notre projet semestriel, nous avons conçu une publication WWW. Ce site sera utilisé afin de mettre en ligne une billetterie permettant l'achat de billet de spectacle.



2. Approche Marketing

2.1 Définition du thème (objectifs et limites)

Notre site a pour but de permettre à un utilisateur lambda de consulter les spectacles actuels et futurs du théâtre Hippocampe, dans cet exemple : le théâtre de l'hippocampe. Celui-ci permet de consulter les descriptifs de chaque spectacle, de connaître les différentes représentations, de procéder à la réservation de billets, au paiement et de connaître l'historique des achats de chaque abonné. D'autre part, il permet à l'administrateur du site de mettre à jour les nouveautés du théâtre.

2.2 Analyse de sites concurrents

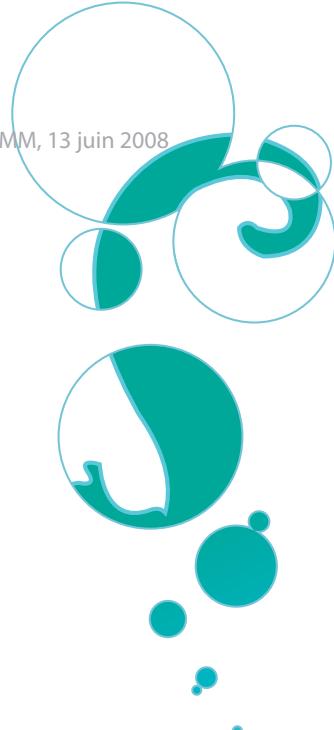
Afin de réaliser les maquettes nous nous sommes inspirées de différents sites déjà existants. tels que :

www.ticketcorner.com/eventstore/control/home
www.co2-spectacle.ch
www.geneveopera.ch/
www.vidy.ch/saison.htm
www.theatredevevey.ch/
www.theatredupassage.ch/
www.theatre-arsenic.ch/site_arsenic/accueil.htm
www.theatredujorat.ch/

2.3 Public cible

Le public concerné directement par le théâtre est un public aisé qui peut consacrer du temps et de l'argent aux loisirs. Dans le cas de notre théâtre les adultes sont visés en priorité. La moyenne d'âge se situe entre 25 et 60 ans. Le public visé n'a pas besoin d'être spécialiste de l'informatique. Nous avons tenté d'adapter le graphisme et la navigation afin que l'interface soit la plus intuitive possible.





3. Ergonomie, Design et Architecture

3.1 Design Graphique

Couleurs

Dès notre page d'accueil, nous avons tenté de recréer un univers ludique, et féérique. Notre graphisme est inspiré des décors de théâtre de l'époque Louis XIV. Ce style baroque nous a permis de nous démarquer des sites classiques et sobres que nous avons l'habitude de rencontrer.

Le dégradé des couleurs froides donne un aspect calme et sérieux à notre outil de vente. Les pointes de jaune donnent un côté joyeux, dynamique et rappelle la magie du théâtre.

Nous souhaitons nous placer dans une gamme de théâtre moderne, et avant-gardiste.

Logo

Dans le thème de l'océan choisi par la classe (pingouin, cachalot, medusa,...), nous avons choisi l'hippocampe comme nom de groupe. Nous avons donc décidé de l'utiliser dans notre logo, afin de différencier symboliquement notre équipe.

L'apparence graphique du logo reste dans un esprit baroque et magique.

Logo : Color : rouge dégradé, Font : Bolton Sans, Icon : Hippocampe en dégradé rouge également



3. Ergonomie, Design et Architecture (suite)

Typographie

1. Police utilisée : Trebuchet MS

2. Texte :

2.1 Couleur : gris 818181

2.2 Font : Trebuchet Regular 12pt, Interline : 14pt

3. Titres :

3.1 Color : bleu 208e9d,

3.2 Font : Trebuchet Bold 14 pt

4. Liens :

4.1 Color : rouge 890120

4.2 Font : Trebuchet Bold ou Regular 12pt

4.3 Roll-over : Souligné en rouge

5. Menu :

5.1 Color : gris 818181

5.2 Font : Trebuchet Regular 14pt

5.3 Roll-over : rouge 208e9d

5.4 Lien actif : gris clair

6. Menu niveau 2:

6.1 Color : gris 818181

6.2 Font : Trebuchet Bold 12pt

6.3 Roll-over : rouge 208e9d

6.4 Over : rouge 208e9d

7. Messages erreur:

7.1 Color : rouge e81b00

7.2 Font : Trebuchet Regular ou Bold



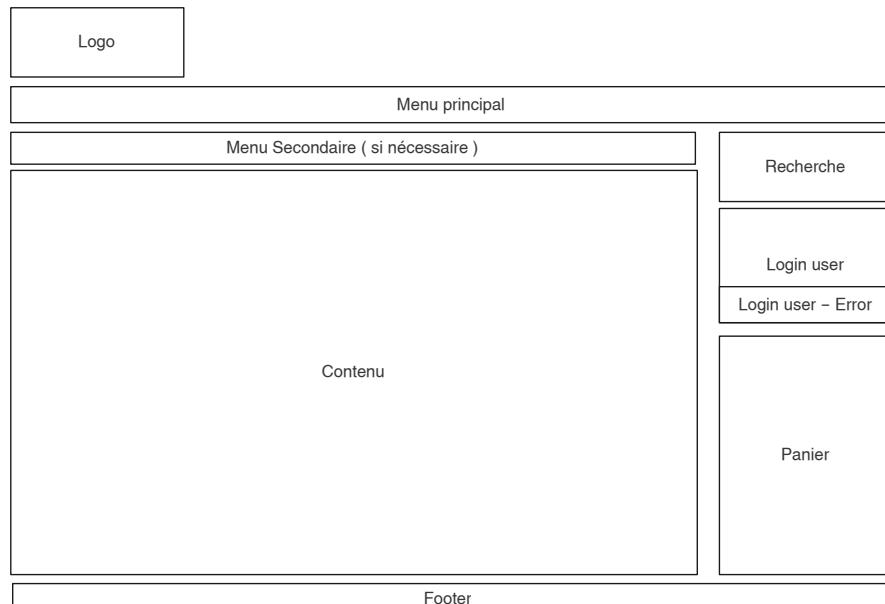
3. Ergonomie, Design et Architecture (suite)

Images, Photographies

Nous avons utilisé des images au format Jpg, les couleurs sont indexées. Nos images et photographies sont tirées d'internet. Nous les avons adaptées du format A4 selon nos besoins. Autrement dit, une diminution de l'image de 50% en moyenne.

3.2 Ergonomie d'interface

Réalisation graphique des blocs



3. Ergonomie, Design et Architecture (suite)

Accessibilité

Grâce aux <div>, il est possible de se placer au travers des blocs au moyen des raccourcis claviers. D'autre part, des outils de navigation supplémentaires spécifiques pour les non-voyants permettent, au moyen de commandes vocales, de se déplacer sur notre site.

3.3 Architecture du Contenu et cohérence de la navigation

Impression de pages WWW : Style CSS

Nous avons utilisé des feuilles CSS pour toute la mise en page. Cependant il n'est prévu aucune interface spécifique concernant la mise en page pour l'impression.

Multimédia, interactivité

Nous avons utilisé du JQuery afin de réaliser le menu déroulant sur la page des spectacles. Du JavaScritp a été nécessaire pour une partie des formulaires (onFocus).

Variétés des navigateurs

En raison de la grande variété d'utilisateurs potentiels, nous avons dû tenir compte des différents navigateurs qui sont sur le marché actuel. Nous avons donc adapté la feuille de style CSS aux différents types de navigateur : Firefox, Internet explorer 7, Safari, Opéra,...



3. Ergonomie, Design et Architecture (suite)

Navigation

Une barre de menu permet de se diriger sur n'importe quelle page du site tout au long de la navigation. Sur chaque page la section activée est grise et le titre de la page est inscrit. Un bloc à droite est réservé pour le login, le panier, l'état de connexion, le profil de l'abonné pendant sa connexion.

De même, le champ de Recherche reste en permanence sur toutes les pages.

Le plan du site permet d'avoir une vue globale de toute l'architecture du site, et de se déplacer directement au travers des pages.

Aucun retour aux pages précédentes n'a été prévu. En effet, ce lien n'aurait pas de réelle utilité dans notre cas et de plus, beaucoup d'internautes utilisent le *RETOUR A LA PAGE PRECEDENTE* de leur navigateur. La barre de menu permet de changer de pages et les achats doivent se poursuivre de la consultation des spectacles à la réservation. Un accès direct au panier permet de modifier ses achats.

Le rollover est marqué en rouge sur le menu. Dans le cas où c'est un lien il est souligné et en rouge.

Fonction de recherche

La fonction de recherche est un outil destiné à trouver un spectacle en saisissant son nom en majuscule ou non. En cas de recherche infructueuse un message d'erreur apparaît.

Temps de réponse

Le site ne présente aucune grosse image en haute résolution ou une quelconque application multimédia très lourde. Notre temps de réponse reste donc dans la moyenne des sites.



4. Améliorations

Lors de la conception nous avons pensé à certaines améliorations graphiques que nous développerons dans le futur. Telles que :

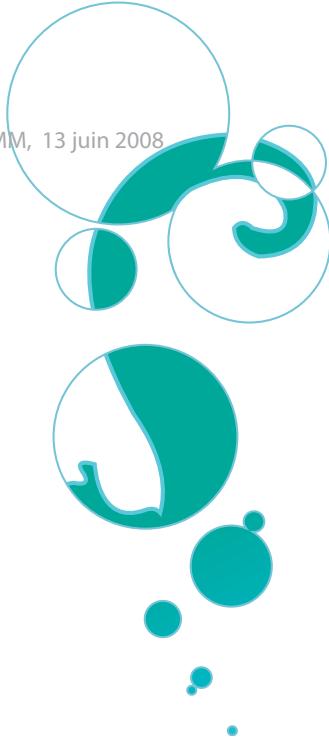
- Intégrer une googleMap, dans la partie contact.
- Nous souhaitons ajouter un module d'agenda de tous les spectacles, qui permettrait de voir immédiatement les représentations par jour.
- Au moment de modifier son login dans l'hippocompte, une vérification de l'ancien mot de passe serait nécessaire.



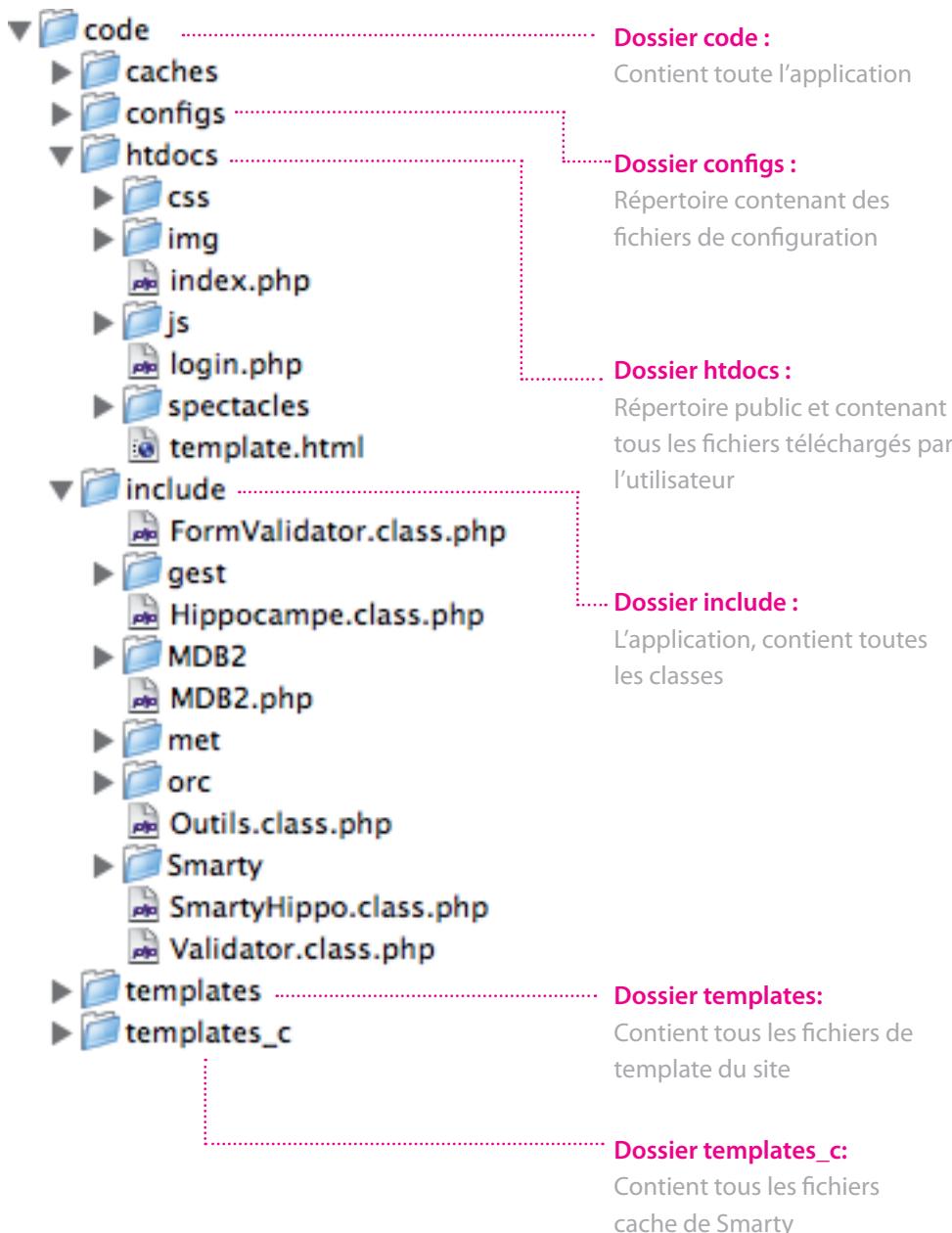
IV. Manuel d'installation

1. Arborescence des fichiers / Documents	59
2. Séparation traitements-données-présentation	65
3. Complément	66
4. Marche à suivre pour l'Installation	66
4.1 Pré requis	66
4.2 Installation	66

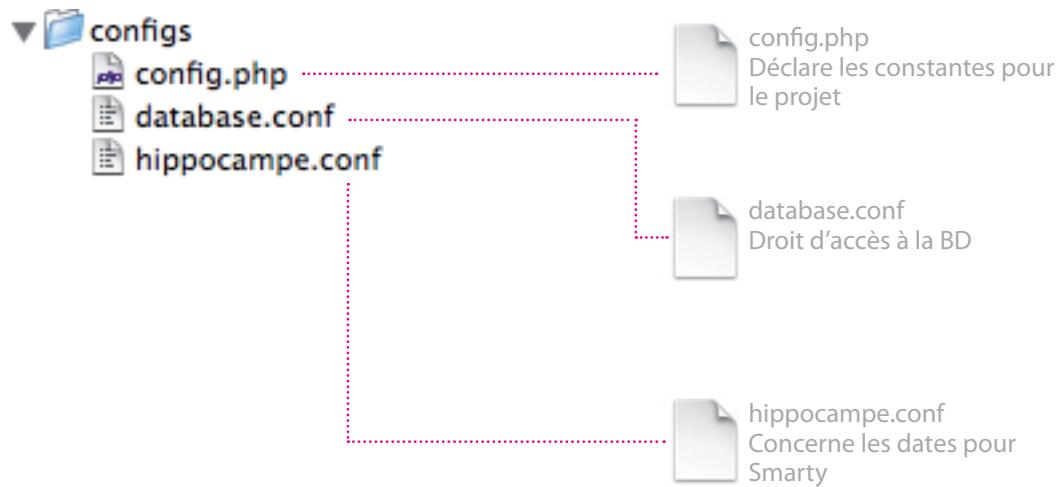




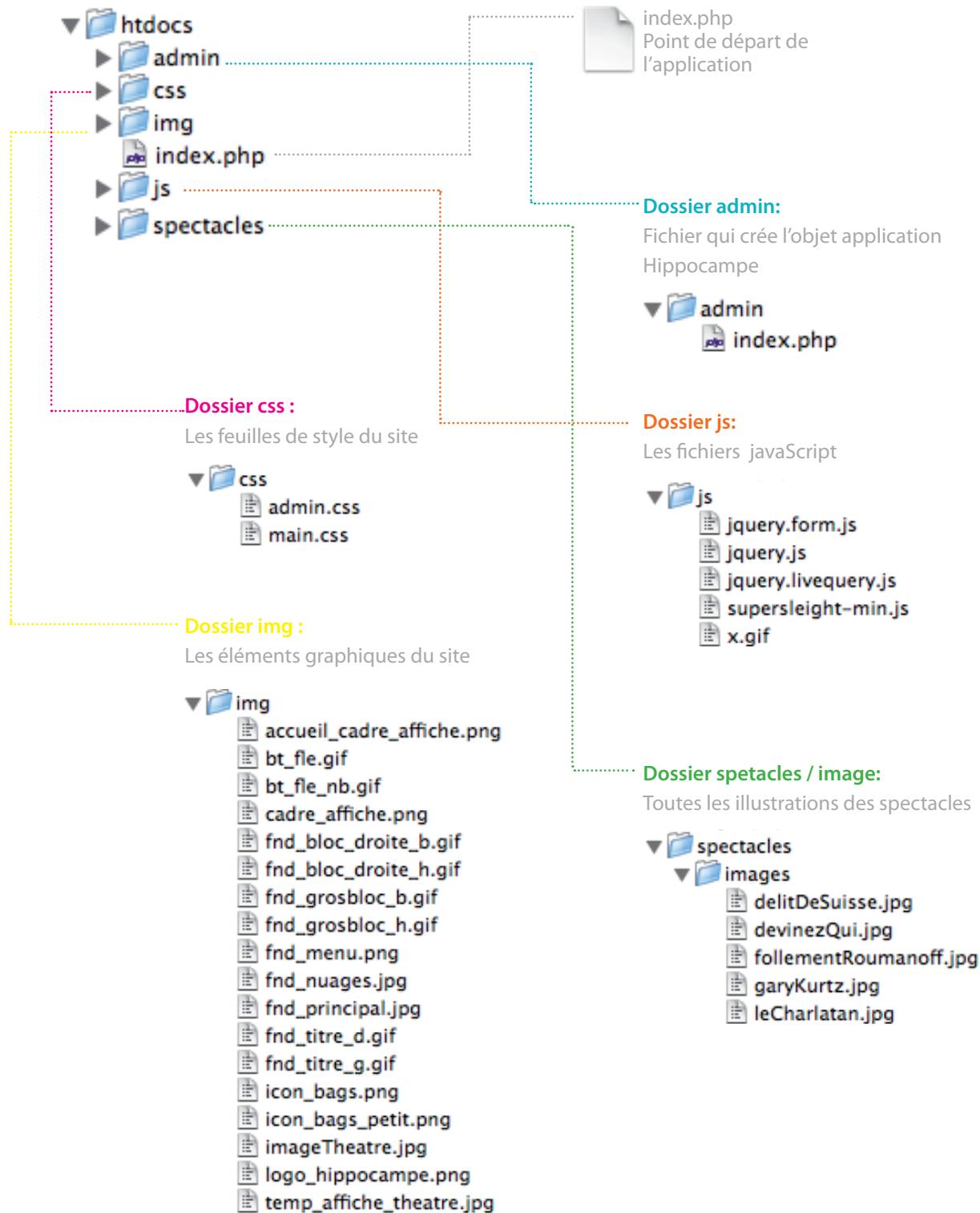
1. Arborescence des fichiers / Documents



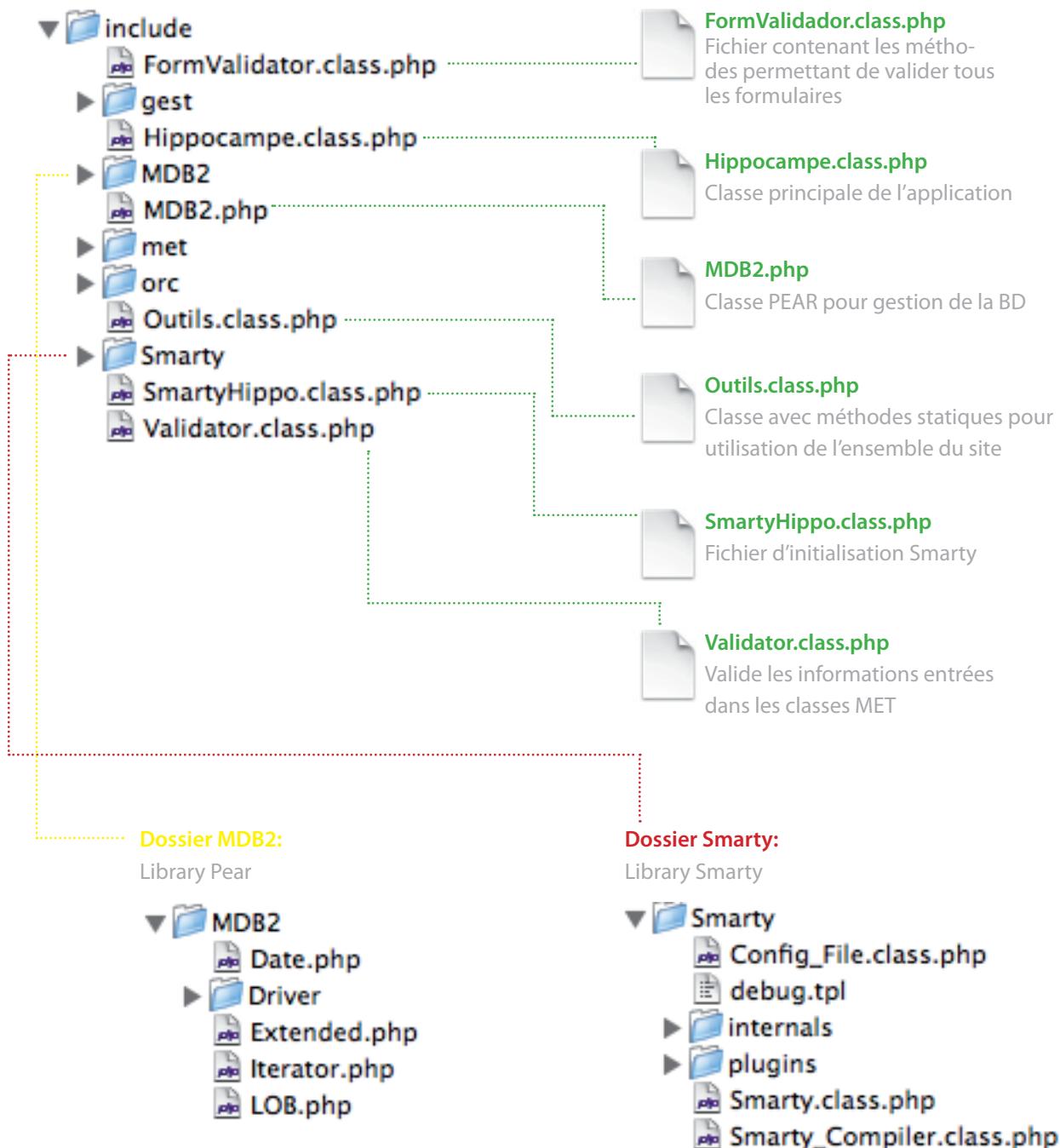
1. Arborecence des fichiers / Documents



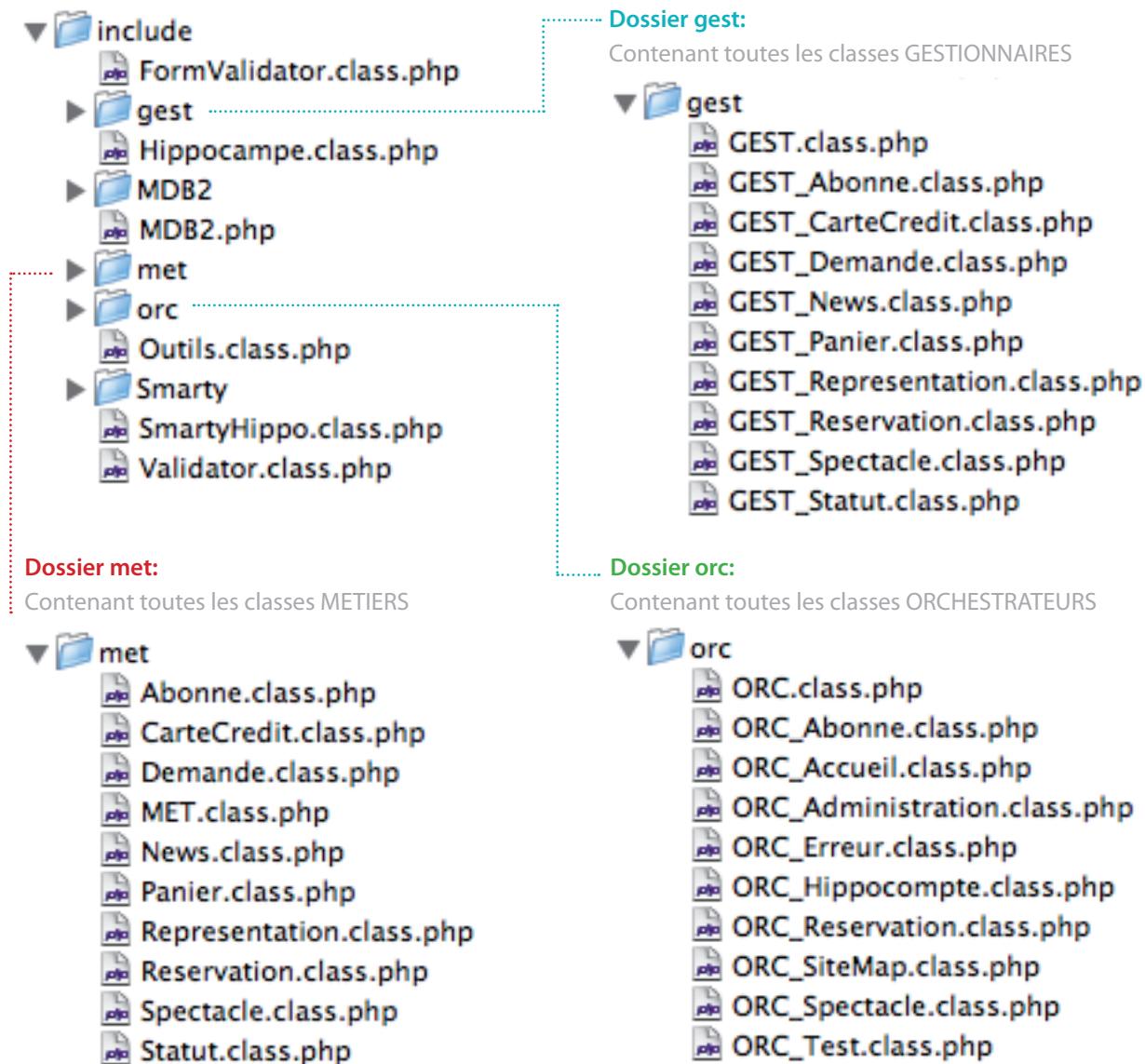
1. Arborecence des fichiers / Documents



1. Arborecence des fichiers / Documents



1. Arborecence des fichier / Documents



Selon l'adaptation du langage de Mr.Maksay, nous avons utilisé la modélisation n-tiers.

GEST : Réunit toutes les méthodes permettant de manipuler des objets appartenant à la classe concernée; on trouvera en particulier les méthodes permettant de valider l'existence d'un objet basé sur son identifiant. GEST inclut les méthodes de base du CRUD qui font appel à la librairie Pear::MDB2 pour la connexion à la base de données.

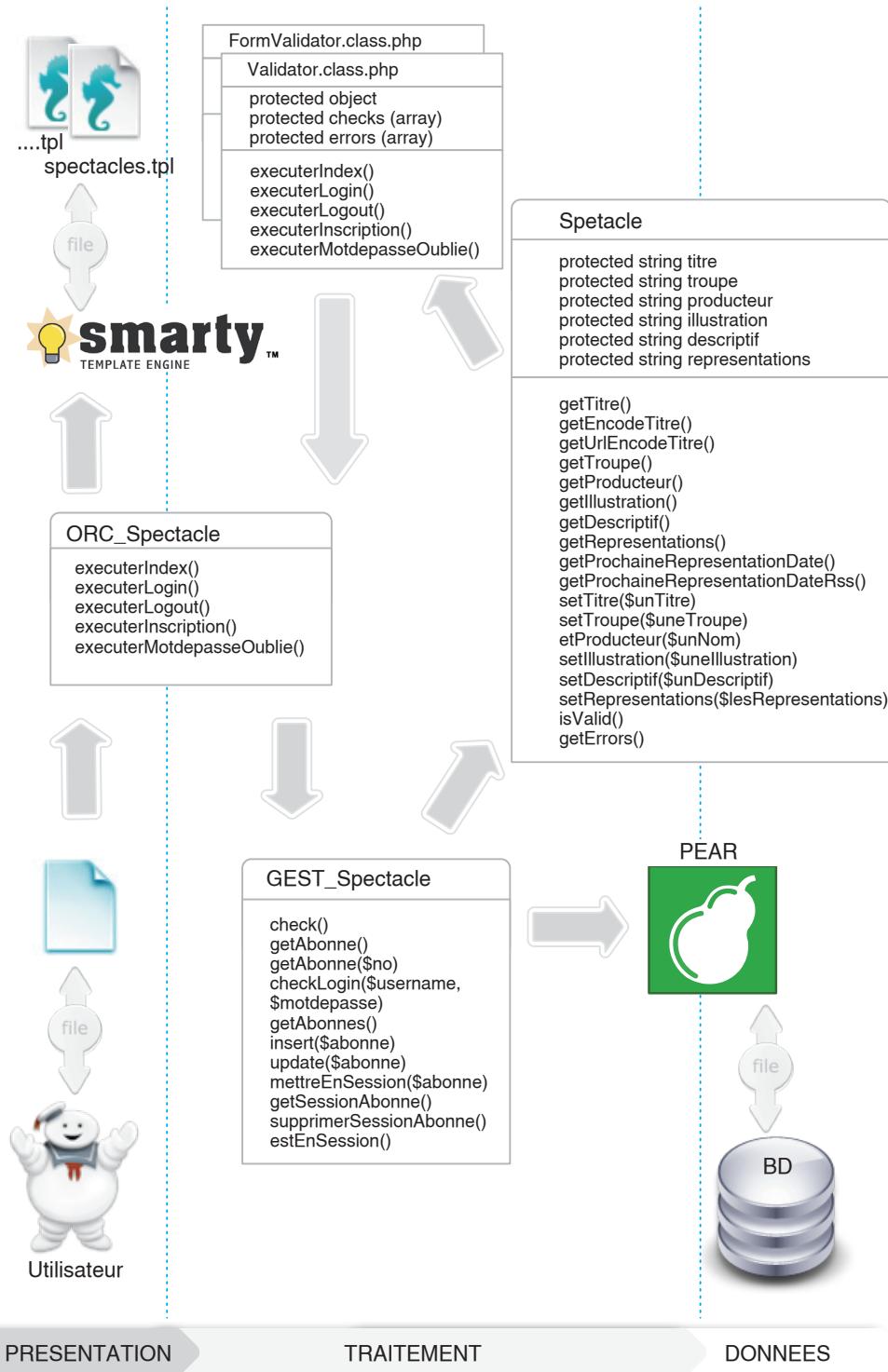
MET : Décrit une classe métier avec ses propriétés, son constructeur ses getteurs et setteurs. Les classes MET peuvent également contenir des méthodes permettant d'effectuer des calculs sur les propriétés d'un objet.

ORC : Décrit la portion de code qui va invoquer, dans un ordre précis, les méthodes qui se trouvent dans la partie GEST et/ou MET.

1. Arborecence des fichier / Documents

▼  templates	Dossier templates :
 footer.tpl		Tous les templates des différents
 header.tpl		parties / blocs du site
 hippocompte.tpl		
 layoutAdmin.tpl		
 layoutHome.tpl		
 menu.tpl		
 metas.tpl		
▼  pages	Dossier pages :
 accueil.tpl		Tous les templates des différentes
 contact.tpl		pages du site
 erreur.tpl		
 form_inscription.tpl		
 form_login.tpl		
 inscription_confirmation.tpl		
 le_theatre.tpl		
 liste_reservation.tpl		
 monhippocompte.tpl		
 motdepasse_oublie.tpl		
 reservation_confirmation.tpl		
 resume_panier.tpl		
 sitemap.tpl		
 spectacles.tpl		
 spectacles_rss.tpl		
 panier.tpl		
 recherche.tpl		
 sousMenu.tpl		

2. Séparation données-traitements-présentation



3. Compléments

Pour le développement de notre application, nous avons utilisé PEAR et Smarty. PEAR (pour PHP Extension and Application Repository) est un ensemble de bibliothèques logicielles de composants PHP. Smarty est, quant à lui, un moteur de templates pour PHP. Rapide, il permet la gestion des caches et facilite la séparation entre la logique applicative et la présentation.

D'autre part, pour éviter la suppression de données, nous avons ajouté une table statut dans notre Diagramme de Classe. Lors de la suppression d'un spectacle les objets ne se suppriment pas, mais leur statut passe en inactif. Les données sont donc conservées dans la base de données, ce qui permet de les récupérer dans l'historique des spectacle des abonnés.

4. Marche à suivre pour l'Installation :

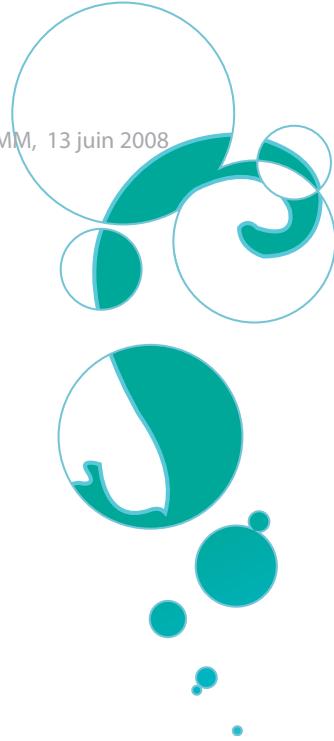
3.1 Pré requis

- Serveur http (ex. Apache 5)
- Php 5
- SGBD postgresql

3.2 Installation

- Tout d'abord, il faut créer une Base de données, au moyen de notre fichier dump.
- Puis configurer l'application Hippocampe, fichier configs/database.conf (permet de modifier les accès à la Base de Données)
- Déploiement de l'application :
Copier tous les répertoires code sur votre serveur.
- Pour Smarty, ajouter des droits d'écriture sur les répertoires templates_c et caches.
- Tester

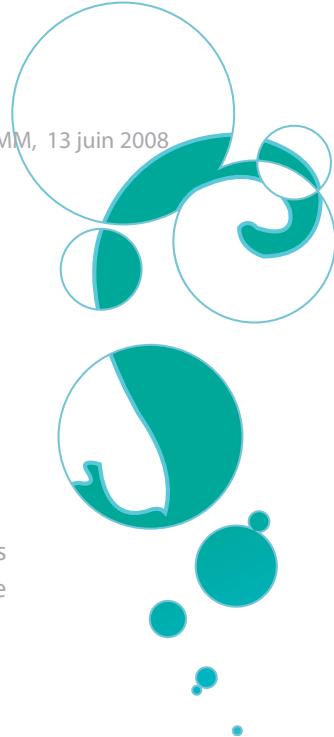




V. Manuel d'utilisation

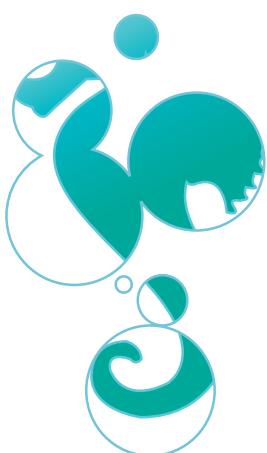
1. Introduction	68
2. Connexion en tant qu'utilisateur	69
2.1 Lancement et accueil	69
2.2 Commun à toutes les pages	70
2.3 Spectacles	76
2.4 Prochainement	80
2.5 Théâtre	81
2.6 Mon Hippocompte	82
2.7 Contact	83
3. Connexion en tant qu'administrateur	84





1. Introduction

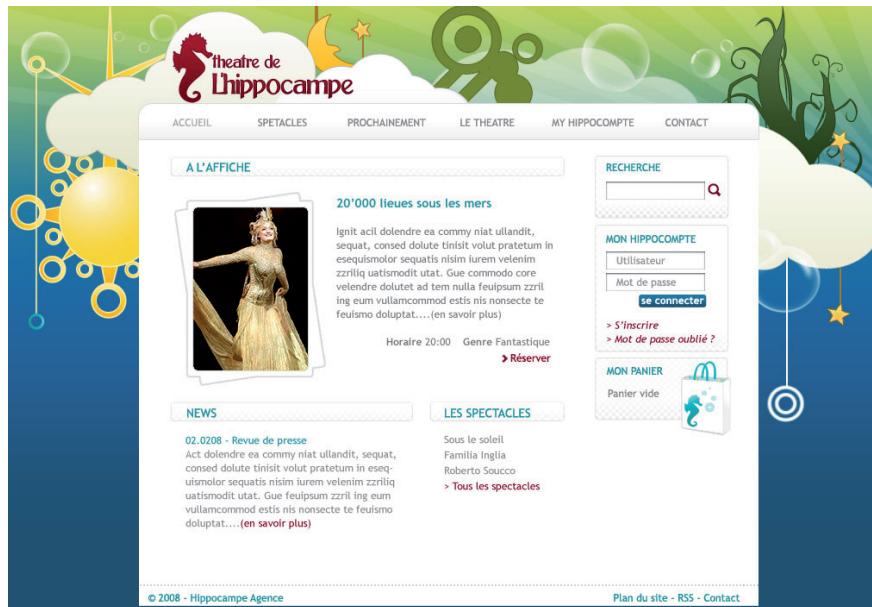
Le but principal du mode d'emploi suivant est de présenter et d'expliquer les fonctions utilisateur et administrateur de notre plateforme de réservation de spectacles en ligne. Et ce, afin de faciliter son utilisation par des personnes n'ayant pas de connaissances en informatique et d'expliquer les différentes tâches que l'on peut accomplir via cette plateforme.



2. Connexion en tant qu'utilisateur

2.1 Lancement et accueil

Pour visionner le site, vous devez ouvrir votre navigateur et saisir l'adresse suivante : <http://pingouin.heig-vd.ch/hippocampe/>
 Vous arrivez alors sur la page ci-dessous :



Sur cette page, vous pouvez voir le principal spectacle à l'affiche avec un accès direct à ses détails, les news, ainsi qu'une liste des spectacles réservables.



2. Connexion en tant qu'utilisateur

2.2 Commun à toutes les pages

Sur chacune des pages du site, il est possible d'effectuer 3 manipulations. Premièrement, si vous êtes déjà inscrit, il vous suffit de saisir votre nom d'utilisateur et votre mot de passe dans les champs appropriés et de cliquer sur le bouton *SE CONNECTER* pour continuer et effectuer des réservations via le site. Si vous avez oublié votre mot de passe, vous pouvez également le récupérer. Deuxièmement, vous pouvez créer votre compte. Et enfin, effectuer une recherche par mot-clé. Pour ce faire, il vous suffit de taper un mot dans le champ *RECHERCHE* et simplement cliquer sur le bouton *OK*.

Les informations pratiques, telles que l'état actuel de votre panier est également accessible sur toutes les pages du site, ainsi que le *PLAN* du site, le *RSS*, *CONTACT* et les boutons.

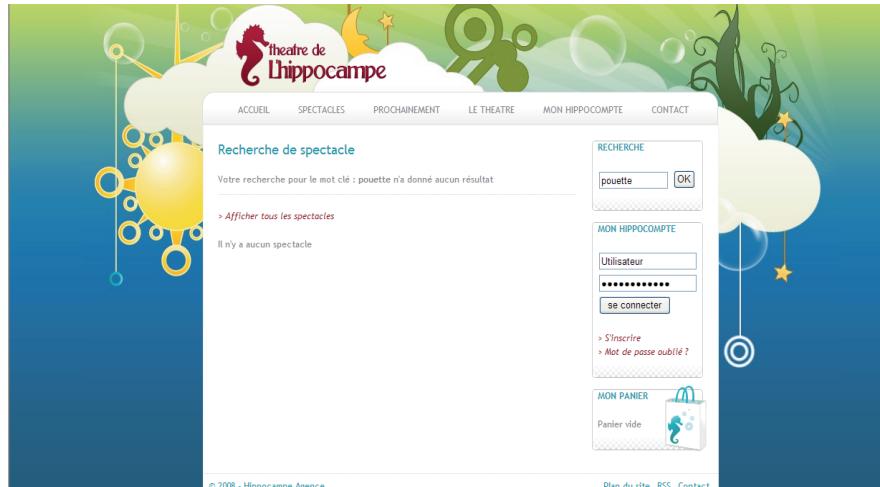
Recherche

Le moteur de recherche développé pour ce site est case insensitive, ce qui signifie qu'il ne tient pas compte des majuscules. Il vous suffit d'entrer un mot et il cherche si ce mot existe dans le site et correspond à un spectacle. Si c'est le cas, il vous propose la liste de spectacle correspondant à votre recherche directement avec un bouton *RESERVER* et le message « Votre recherche pour le mot clé : XXX a donné un résultat ».



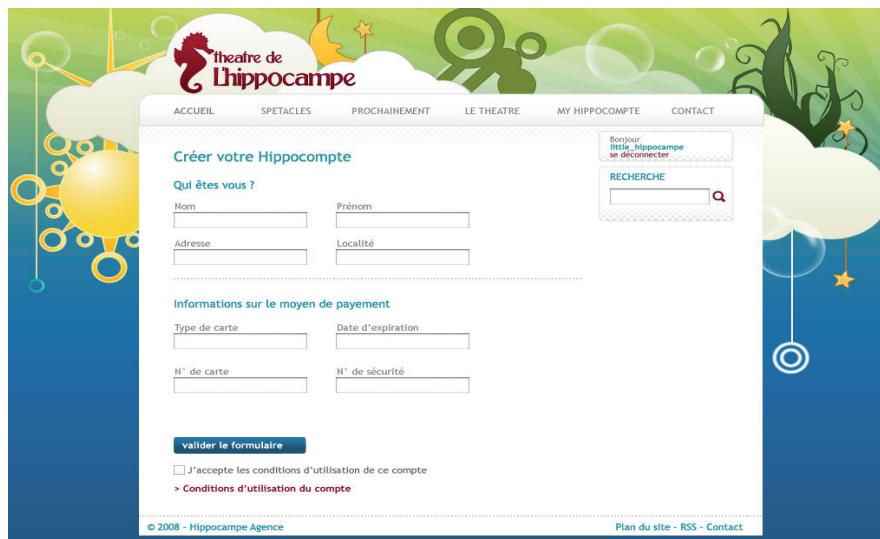
2. Connexion en tant qu'utilisateur (suite)

Sinon, il affiche « Votre recherche pour le mot clé : XXX n'a donné aucun résultat avec un lien « Afficher tous les spectacles » qui renvoit à la page *SPECTACLES*, ainsi que la phrase « Il n'y a aucun spectacle ».



S'inscrire

Sous la rubrique *MON HIPPOCOMPTE*, notre plateforme vous donne la possibilité de créer un nouveau compte. En effet, lorsque vous cliquez sur *S'INSCRIRE*, une page d'inscription qui sert à créer son propre Hippocompte apparaît. Sur cette page, vous devez *CONFIRMER* votre mot de passe en l'entrant une nouvelle fois dans le champ réservé à cet effet.



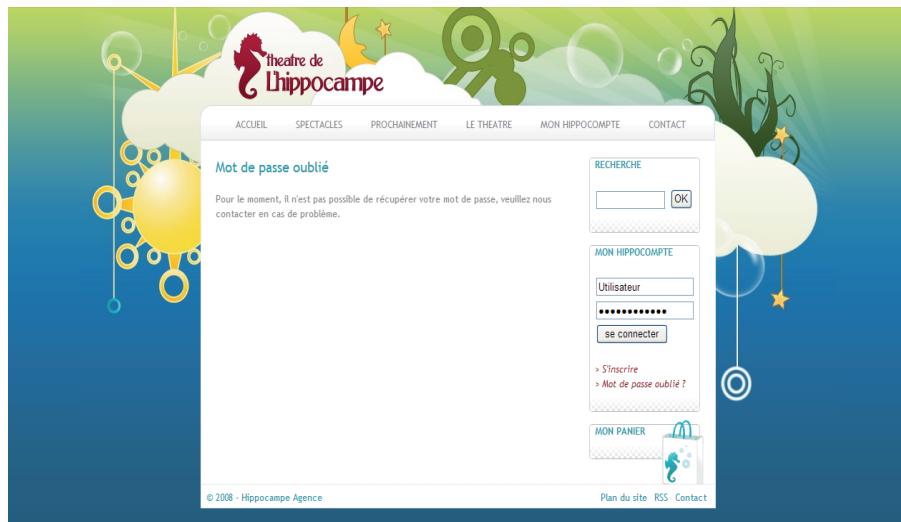
2. Connexion en tant qu'utilisateur (suite)

Il vous suffit alors de remplir le formulaire de manière appropriée et de cliquer sur le bouton *VALIDER LE FORMULAIRE*.

Tout au long des pages, vous êtes informé de votre état de connexion dans la partie de droite, « Bonjour+ nom User ».

Mot de passe oublié

Sous la rubrique *MON HIPPOCOMPTE*, La page « mot de passe oublié » n'a pas été développée pour l'instant et affiche « Pour le moment, il n'est pas possible de récupérer votre mot de passe, veuillez nous contacter en cas de problème », mais vous pouvez nous contacter par e-mail et nous vous renverrons votre mot de passe en cas d'oubli.



2. Connexion en tant qu'utilisateur (suite)

Mon Panier

« Mon panier » représente l'état actualisé de votre panier d'achat. Le calcul du total se fait au fur et à mesure de vos réservations.

Une fois logué, en cliquant sur le panier, l'utilisateur accédera au détail de ce qu'il aura commandé.



Titre de la pièce	Date	Heure	Nbr place	Prix/Billet
Sous le soleil	jeudi 01.01.08	20:30	1x	10.-
20000 lieus sous la mer	vendredi 02.01.08	20:30	2x	35.-

Total : 80.-

Reculer Réservé d'autre billet Commander

© 2008 - Hippocampe Agence Plan du site - RSS - Contact



2. Connexion en tant qu'utilisateur (suite)

Le Footer

En bas de chaque page du site, des liens renvoient vers le *PLAN DU SITE*, le *RSS* et notre page *CONTACT*.

© 2008 - Hippocampe Agence

[Plan du site](#) [RSS](#) [Contact](#)

Il y a également un lien RSS (Really Simple Syndication) qui est un fichier XML mis à jour en temps réel. Il reprend automatiquement les titres et le texte intégral des descriptions de nos spectacles. On peut donc s'abonner un flux RSS et recevoir les informations sur les spectacles directement par e-mail.

L'Hippocampe, les spectacles
 Vous affichez un flux RSS qui contient un contenu fréquemment mis à jour. Lorsque vous êtes abonnés à un flux RSS, il est ajouté à votre liste de flux RSS commune. Les informations mises à jour à partir du flux RSS sont automatiquement téléchargées sur votre ordinateur et peuvent être affichées dans Internet Explorer et d'autres programmes. [En savoir plus sur les flux RSS.](#)

 [M'abonner à ce flux](#)

Le Charlatan
mardi, 12. août 2008, 01:00:00 

Quand deux escrocs se rencontrent, qu'est-ce qu'ils se racontent ? Des histoires d'escrocs, bien sûr ! Albert et Emile n'échappent pas à la règle.

Gary Kurtz
vendredi, 10. mars 2005, 01:00:00 

Laissez-vous emporter par les prouesses de Gary Kurtz. Il devine ce que les spectateurs ont dans leurs poches, les numéros de série des billets pliés dans leur portefeuille, un événement qui a...

Follement Roumanoff
dimanche, 13. mars 2005, 01:00:00 

Déclats de rires en émotions, Anne ROUMANOFF confirme dans ce nouveau spectacle 'Follement Roumanoff' son talent d'humoriste. Elle incarne avec brio : une bouchère obsédée par...

Délit de Suisse
vendredi, 31. décembre 2004, 01:00:00 

Est-ce un délit d'être Suisse ? Y a-t-il une vie après le 1er août ? Yann Lambiel répond à toutes ces questions et bien d'autres par le biais de nos hommes politiques préférés.

Affichage 5 / 5

Tout 5

Trier par : ▼ Date Titre



2. Connexion en tant qu'utilisateur (suite)

Le Footer

Il y a également un permalien vers notre page *CONTACT*.



Les boutons

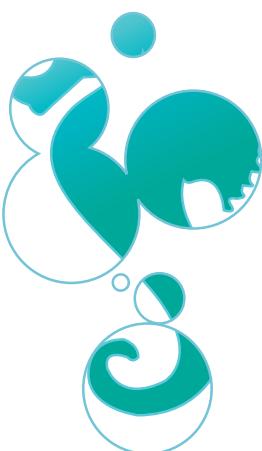
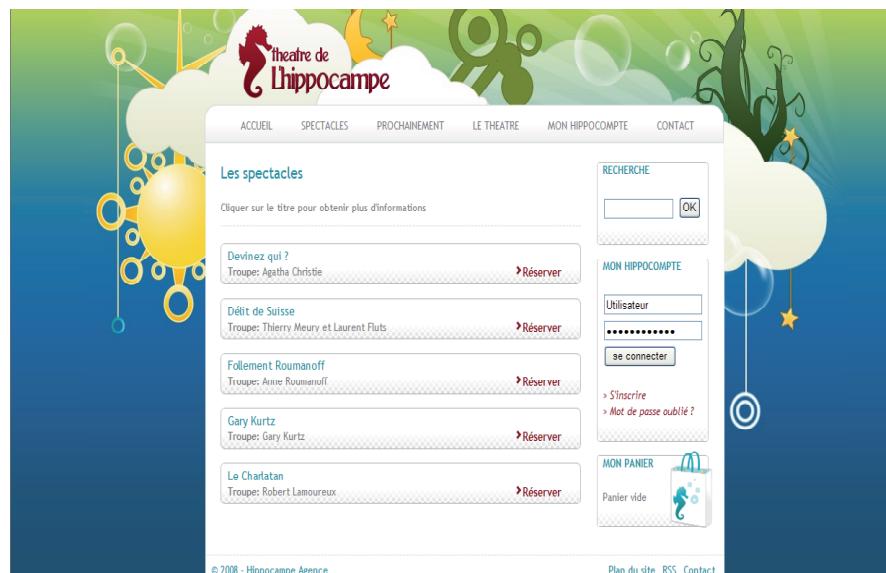
La barre de menu contenant les boutons est présente sur chaque page et permet d'accéder directement à l'*ACCUEIL*, aux *SPECTACLES* à l'affiche du moment, à un agenda des spectacles, au *THEATRE* ainsi qu'à notre page *CONTACT*.



2. Connexion en tant qu'utilisateur (suite)

2.3 Spectacles

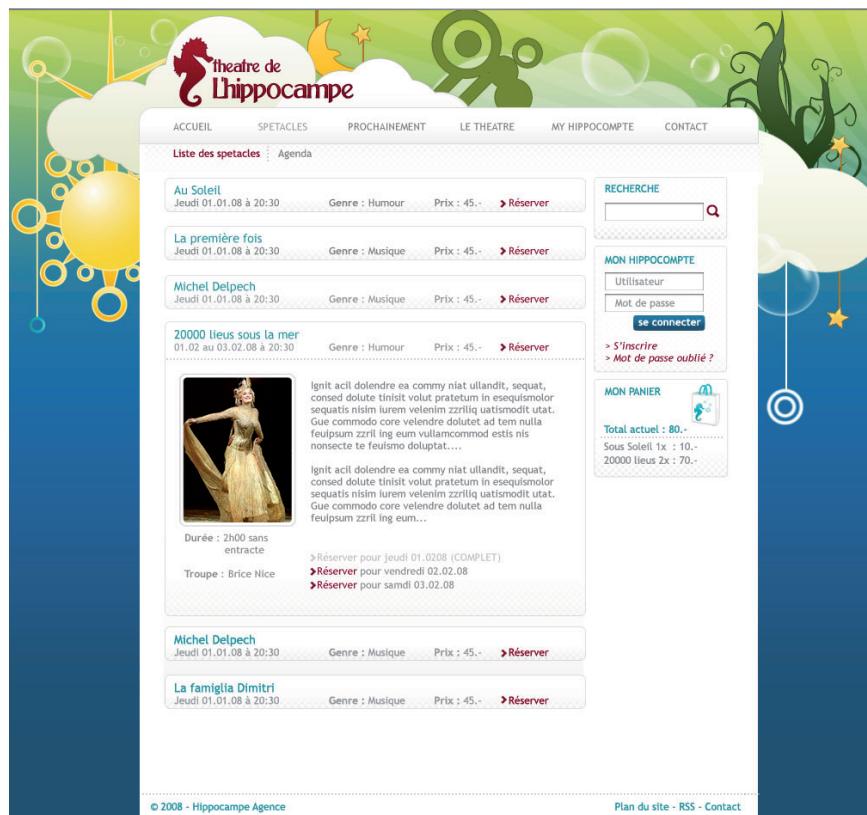
En cliquant sur le bouton *SPECTACLES*, vous accéderez directement à la liste des spectacles à l'affiche et qu'il est possible de réserver.



2. Connexion en tant qu'utilisateur (suite)

2.3 Spectacles (suite)

Sur cette page, en cliquant sur l'un des spectacles, une description apparaît par le biais d'un menu déroulant. La Description apparaît également lorsque l'on clique sur le *RESERVER* qui se trouve à côté du titre du spectacle.



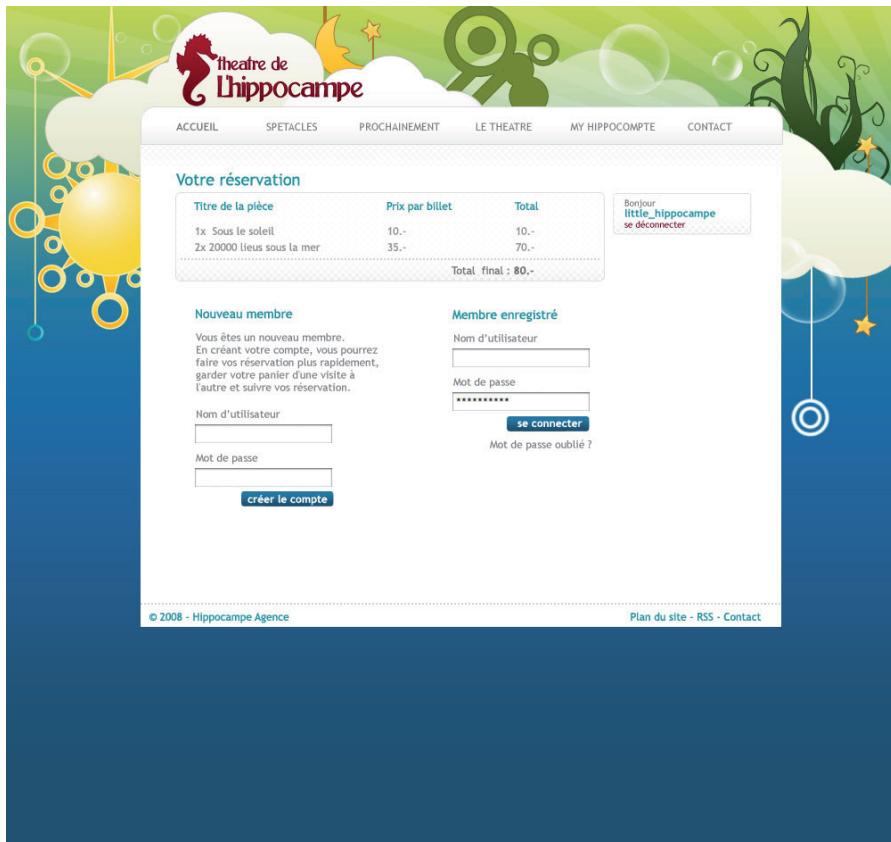
Spectacle	Date	Genre	Prix	Action
Au Soleil	Jeudi 01.01.08 à 20:30	Genre : Humour	Prix : 45,-	Réserver
La première fois	Jeudi 01.01.08 à 20:30	Genre : Musique	Prix : 45,-	Réserver
Michel Delpech	Jeudi 01.01.08 à 20:30	Genre : Musique	Prix : 45,-	Réserver
20000 lieus sous la mer 01.02 au 03.02.08 à 20:30				
Ignit acil dolendre ea commy niat ullandit, sequat, consed dolute tinitis volut pratetum in esequismolor sequatis nism lurenim velenim zzriliq utsmmodit utat. Gue commodo core velendre dolutet ad tem nulla feulpsum zzril ing eum vullamcommod estis nis nonsecte te feusimo dolupat...				
Ignit acil dolendre ea commy niat ullandit, sequat, consed dolute tinitis volut pratetum in esequismolor sequatis nism lurenim velenim zzriliq utsmmodit utat. Gue commodo core velendre dolutet ad tem nulla feulpsum zzril ing eum...				
Durée : 2h00 sans entracte Troupe : Brice Nice Réserver pour jeudi 01.02.08 (COMPLET) Réserver pour vendredi 02.02.08 Réserver pour samedi 03.02.08				
Michel Delpech Jeudi 01.01.08 à 20:30				
La famiglia Dimitri Jeudi 01.01.08 à 20:30				

Lorsque l'on décide de réserver un spectacle, il suffit de cliquer sur *RESERVER* qui se trouve sous la description et qui correspond à la représentation de votre choix.

2. Connexion en tant qu'utilisateur (suite)

2.3 Spectacles (suite)

On accède ensuite à une page intitulée « Votre Réservation » sous forme de facture. Si l'utilisateur n'est pas encore logué, il peut le faire via cette page et s'il ne possède pas encore de compte, il peut également le créer via cette page .



Titre de la pièce	Prix par billet	Total
1x Sous le soleil	10.-	10.-
2x 20000 lieus sous la mer	35.-	70.-
Total final : 80.-		

Nouveau membre
 Vous êtes un nouveau membre.
 En créant votre compte, vous pourrez faire vos réservations plus rapidement,
 garder votre panier d'une visite à l'autre et suivre vos réservations.

Membre enregistré

Bonjour little_hippocampe
[se déconnecter](#)

Nom d'utilisateur

Mot de passe
 [Mot de passe oublié ?](#)

se connecter

[créer le compte](#)

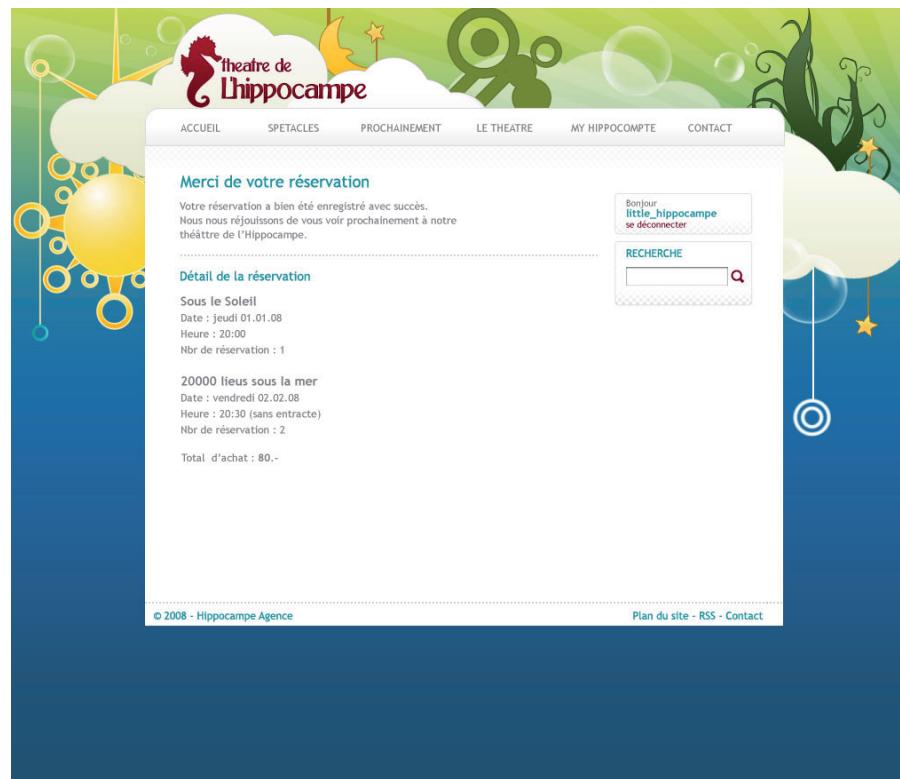
© 2008 - Hippocampe Agence [Plan du site - RSS - Contact](#)



2. Connexion en tant qu'utilisateur (suite)

2.3 Spectacles (suite)

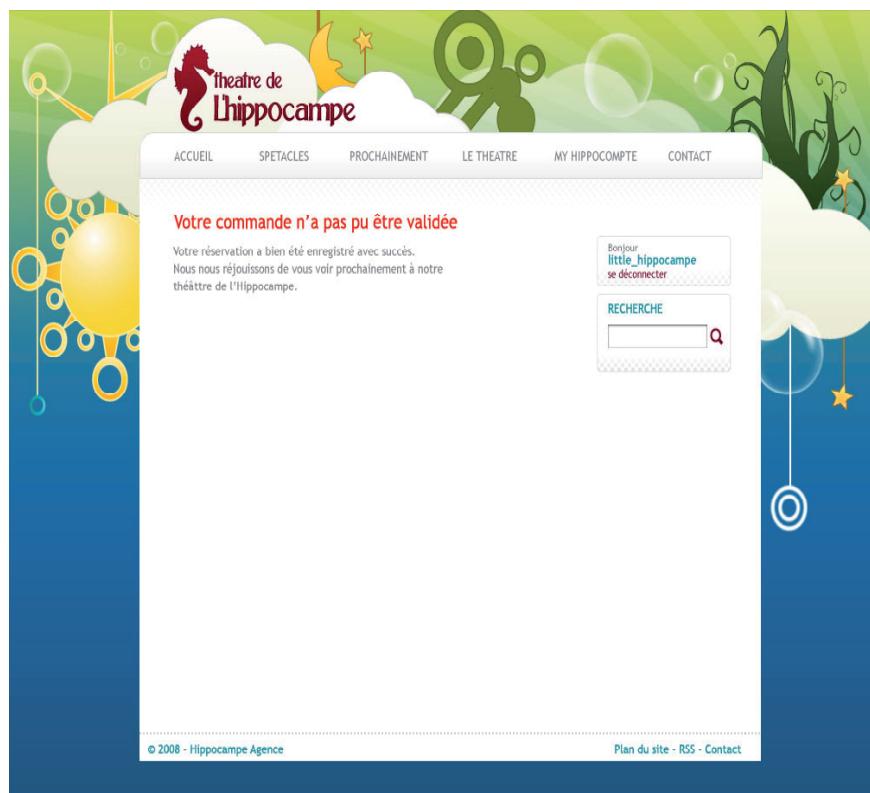
Une fois la réservation effectuée et confirmée, l'utilisateur sera redirigé sur une page de remerciement et de confirmation contenant également les détails du spectacle réservé.



2. Connexion en tant qu'utilisateur (suite)

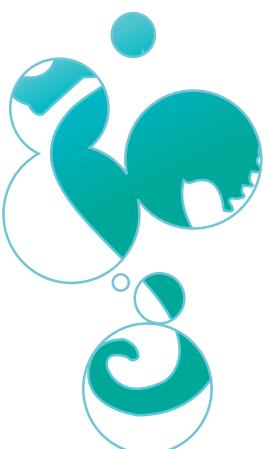
2.3 Spectacles (suite)

En cas de problème lors de la réservation, l'utilisateur verra un message d'erreur.



2.4 Prochainement

Le lien de ce bouton n'a pas été développé pour des questions de temps, mais il amènera sur une page contenant l'agenda de tous les spectacles à l'affiche (réservables) et également de ceux qui ne sont pas encore à l'affiche (pas encore réservables).



2. Connexion en tant qu'utilisateur (suite)

2.5 Théâtre

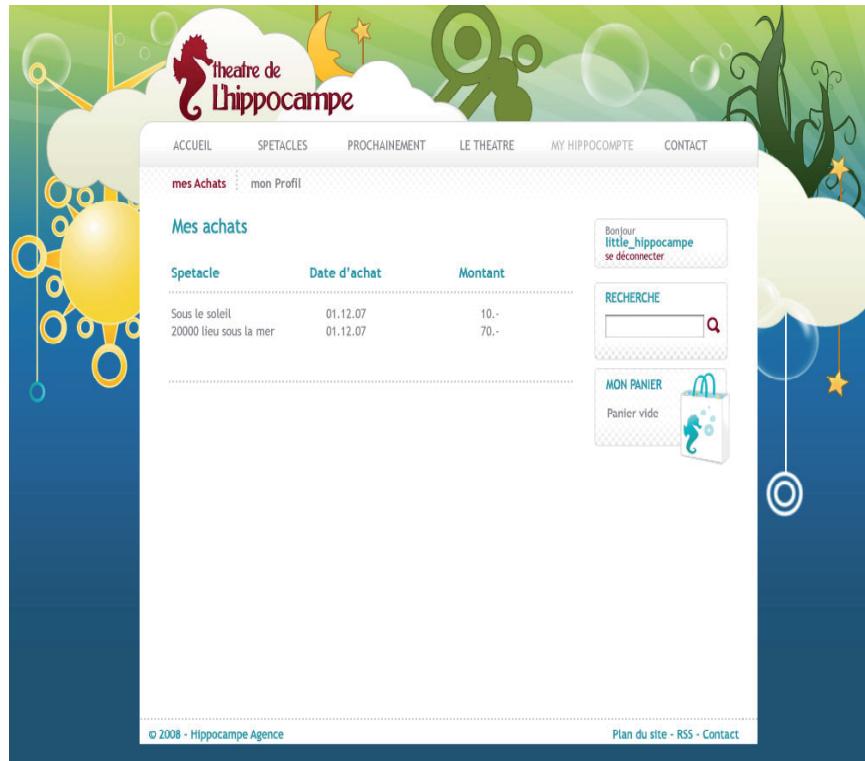
En cliquant sur le bouton *THEATRE*, vous accéderez à une présentation du théâtre de l'hippocampe.



2. Connexion en tant qu'utilisateur (suite)

2.6 Mon Hippocompte

En cliquant sur le bouton *MON HIPPOCOMPTE*, l'utilisateur verra ses pages personnelles. Il aura alors accès à un historique de «mes Achats» et pourra également remplir «mon Profil». Cette section permet également de consulter, modifier ou supprimer des informations du profil.



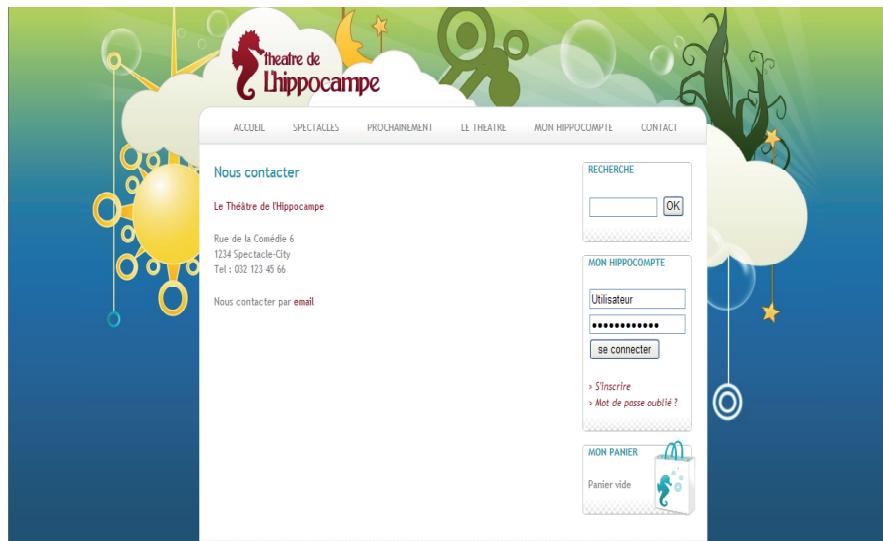
Spectacle	Date d'achat	Montant
Sous le soleil 20000 lieu sous la mer	01.12.07 01.12.07	10.- 70.-



2. Connexion en tant qu'utilisateur (suite)

2.7 Contact

Le bouton *CONTACT* sert, tout comme le *CONTACT* du Footer, à accéder aux coordonnées du théâtre.

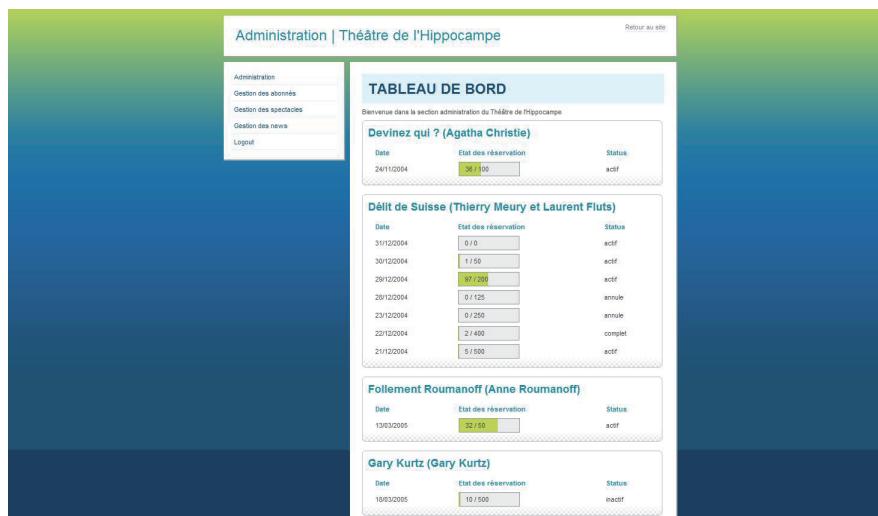


3. Connexion en tant qu'administrateur

En se connectant en temps qu'administrateur via les champs de connexion comme le ferait un utilisateur normal en utilisant «admin» comme username et «admin» comme mot de passe, ce dernier pourra gérer les comptes des utilisateurs (supprimer, modifier, consulter) en accédant à toutes leurs données ainsi qu'aux représentations qu'ils ont réservées (passées ou futures). D'autre part, l'administrateur peut gérer l'ensemble de ses spectacles (ajouter, modifier) ainsi que ses représentations. Enfin, l'administrateur devra modifier les statuts des représentations à l'affiche, en cas d'annulation ou de représentation complète. Un spectacle passé sera automatiquement inactif, donc plus à l'affiche.

Pour le moment, en se connectant en temps qu'administrateur, celui-ci a accès à un « tableau de bord » duquel il peut consulter le statut des différents spectacles et voir, sous forme graphique, le nombre de places déjà réservées par date de représentation. De même, il est possible de modifier, ajouter et supprimer des news. Les autres onglets doivent encore être développés. Il s'agit de la gestion des abonnés, de la suppression et de l'ajout de la gestion des spectacles. La plateforme offre déjà la possibilité de modifier des spectacles.

Pour différencier l'espace administrateur de l'espace utilisateur, nous avons décidé d'utiliser un design plus sobre et plus fonctionnel.



Date	Etat des réservation	Status
24/11/2004	36 / 00	actif
31/12/2004	0 / 0	actif
30/12/2004	1 / 50	actif
29/12/2004	07 / 200	actif
28/12/2004	0 / 125	annulé
23/12/2004	0 / 250	annulé
22/12/2004	2 / 400	complet
21/12/2004	5 / 500	actif

Date	Etat des réservation	Status
13/03/2005	22 / 100	actif

Date	Etat des réservation	Status
18/03/2005	10 / 500	inactif





VI. Rapport de terminaison

1. Les Ressources **86**

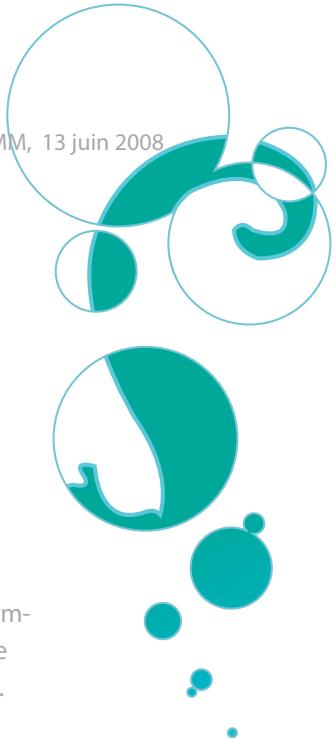
1.1 Communication	86
1.2 Coordination	86
1.3 Imprévus	86

2. Problèmes rencontrés **87**

2.1 Base de données	87
2.2 Programmation	87

3. Conseils et remarques **88**





1. Les Ressources

1.1 Communication

En dehors de l'école, nos principaux moyens de communications furent notamment les téléphones portables, les messageries en ligne, les mails ainsi que le système SVN qui fut un grand avantage lorsqu'il y a plusieurs programmeurs.

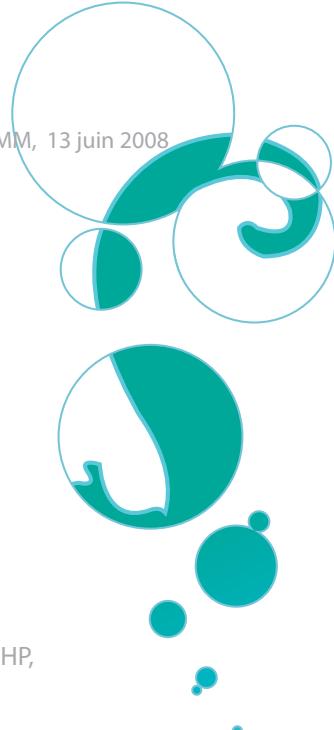
1.2 Coordination

Chacune d'entre nous s'est proposée spontanément et naturellement afin d'effectuer les différentes tâches. De plus, une très bonne collaboration entre les membres du groupe nous a permis d'affronter les difficultés. Devant un problème, chacune essayait d'apporter son aide et des solutions afin d'éviter qu'une personne soit ralentie par une complication. Les tâches étaient donc clairement définies et chacune a trouvé sa place dans le groupe et savait exactement ce qu'elle devait faire notamment pendant les soirées et les weekends.

1.3 Imprévus

L'entente entre les membres de notre groupe a été très bonne. C'est un point qu'il est important de relever, en raison de la maladie durant une semaine de l'une de nos ressources, Joëlle Dind. Grâce à la bonne entente entre les membres du groupe, la répartition des tâches a été évidente et chacune a pu communiquer l'état du projet et les tâches à faire à la personne absente. Lors de chacune de nos rencontres, nous effectuions un point sur l'avancement de notre travail et définissions les prochaines tâches à exécuter. Pour travailler ensemble, nous avions revu l'aménagement de nos bureaux, afin d'être plus proches. Cette disposition facilitait beaucoup la communication.





2. Problèmes rencontrés

Lors du développement, nous avons été confrontées à plusieurs difficultés.



2.1 Base de données

Dans certaines tables, les ID n'étaient pas numériques. Dans notre langage PHP, une manipulation d'ID numérique aurait simplifié la programmation.

La table REPRESENTATION n'avait pas d'ID propre. A la place, elle dépendait de deux autres tables. Il a donc été très difficile de récupérer les données afin de lister les spectacles avec leurs dates de représentations.

Il n'existe pas, dans la BD, de champ permettant d'indiquer si une réservation a été payée ou non.

Dans les dates de représentation de la table REPRESENTATION de la BD, les dates ne sont pas à jour (passées au lieu d'être futures). Le changement de ces dates implique des changements dans la base de données toute entière.

Le mélange de français et d'anglais a géné notre communication. Il aurait été judicieux de choisir une langue et de s'y tenir pour des questions de synchronisation.

Une différence entre les appellations des champs de la base de données et du Domain model est une difficulté supplémentaire.

2.2 Programmation

Le plus important de nos problèmes a été de définir la structure de l'application.

Certaines notions peu utilisées ont été difficiles à réaliser (gestion panier, login,...)

La gestion du format de date a été compliquée au niveau de l'affichage (timeStamp) et jj/mm/aa. De plus, les données récupérées dans la base de données ne sont pas actuelles avec l'année 2008. Nous avons donc du simuler une date antérieure qui coïncide avec les données fournies.

Problème Objet en Session : utilisation de sessions pour la gestion du panier et des utilisateurs.

Problème rencontré : PHP doit connaître le nom des classes avant d'initialiser les sessions, car le « include » des classes est placé avant l'initialisation.



3. Conseils et remarques

Lors de la réalisation de ce travail, nous nous sommes rendues compte qu'une bonne modélisation est essentielle afin d'éviter de mauvaises surprises lors de la programmation.

Nous avons également décidé de mettre la Base de donnée directement en ligne et de programmer à partir du résultat et non pas en local. Cela évite de multiplier les bases de données sur les machines et chacun travaille à partir de la même BD. Un des désavantages de cette méthode : nos idées sont facilement accessibles et il existe un risque de plagiat.

Enfin la gestion du temps nous a fortement handicapées. Le fait d'avoir des cours tout au long du travail de semestre empêche de s'y consacrer complètement. De plus, même si une partie de la modélisation nous était fournie il est très difficile de suivre un cadre définit. Il a été nécessaire de revoir la modélisation, ce qui nous a également pris un certain temps. Dans ce manque de temps, La visite de la Drupa, proposée par l'école, nous a également privées de deux de nos ressources pendant trois jours entiers.

Pour conclure, notre groupe a apprécié l'utilisation des diverses notions apprises tout au long de notre deuxième année. De plus, le travail en équipe fut une expérience très enrichissante pour chacune d'entre nous.

