

# **Programmation Web avec PHP**

**Laurent Iacroy**

**Éditions Eyrolles**

**ISBN : 2-212-09113-3**

**2000**

## Étude de la mise en œuvre d'un site web marchand

---

### Contexte du projet

#### *Présentation*

Fondées en 1918, les éditions Eyrolles sont aujourd'hui l'un des premiers éditeurs scientifiques et techniques français. Spécialisées à l'origine dans le bâtiment et les travaux publics, les éditions Eyrolles ont élargi progressivement leur spécialisation à l'ensemble des sciences et techniques : informatique, sciences fondamentales, techniques de l'ingénieur, bâtiment-travaux publics, artisanat et vie pratique constituent le cœur d'un catalogue d'un millier de titres. S'orientant naturellement vers les nouvelles technologies, les éditions Eyrolles ont décidé de s'ouvrir au Web et offrent l'accès à leur catalogue à travers leur site de commerce électronique.

La Librairie Eyrolles a décidé de permettre aux internautes de rechercher ses ouvrages puis de les commander par l'intermédiaire de son site web.

*Dans ce chapitre, nous décrivons, étape par étape, l'ensemble des phases nécessaires à la réussite du projet, depuis le cahier des charges – qui récapitule les besoins fonctionnels – jusqu'à la livraison de l'application sur le serveur de production.*

Objectif : créer un site commercial, qui regroupe toutes les fonctionnalités de recherche, de sélection et d'achat d'ouvrages.

#### **Cahier des charges**

*Le cahier des charges a pour objectif de délimiter le périmètre fonctionnel de l'application. Il décrit les principaux besoins et traitements à inclure dans le site. Il donne une vision d'ensemble, ce qui permet de cerner l'infrastructure technique adéquate.*

Un site de cette catégorie répond à des besoins fonctionnels bien précis :

**1. Répertorier le plus rapidement possible les livres qui répondent au mieux à l'attente du visiteur.**

La première étape pour l'internaute est de trouver l'ouvrage souhaité, à travers l'ensemble du catalogue. Les références de cet ouvrage étant plus ou moins précises, différentes méthodes de recherche doivent être conçues. Une méthode va porter sur une recherche multicritère qui définit les références précises du livre recherché (titre, auteur, ISBN, etc.), une autre va utiliser une classification thématique (à l'aide d'Unix, de la programmation Java, de systèmes d'exploitation, etc.).

**2. Sélectionner un ouvrage et le mettre dans son panier.**

Dans un véritable magasin, un client flâne dans les rayons, choisit ses articles les uns à la suite des autres, les dépose dans son panier, puis se rend à la caisse pour régler le tout. Les sites marchands tentent de reproduire ces habitudes d'achat le plus fidèlement possible. Ainsi, une fois un article sélectionné, il est mémorisé dans un panier virtuel. L'internaute peut ainsi sélectionner plusieurs articles. Ce n'est qu'une fois son choix arrêté sur une liste d'ouvrages qu'il est amené à passer sa commande.

**3. Régler sa commande simplement, et en toute sécurité.**

À tout moment, le client peut accéder au formulaire du bon de commande, dans lequel il saisit ses coordonnées et son numéro de carte de crédit. Pour garantir la sécurisation et la confidentialité des échanges, il est impératif que l'envoi des données se fasse de manière cryptée. Le protocole HTTP ne remplit pas cette condition. Il faut donc utiliser le mode SSL (Secure Socket Layer) ou HTTPS, qui crypte les échanges vers le serveur, et offre toutes les garanties.

### Un site de qualité

Pour obtenir un site de qualité, quelques règles doivent être respectées :

**1. Construction graphique sobre et efficace.**

Acheter un livre sur Internet demande du temps et un nombre important de clics. La mise en page du site doit faciliter au maximum la démarche à l'aide d'une présentation claire et intuitive. Les sites trop complexes, dont la mise en page est trop dense, n'incitent pas à l'achat, car ils demandent un effort de compréhension.

**2. Formulaire de commande simple.**

Très souvent, l'acte d'achat s'arrête au moment de la commande pour une raison simple : l'effort le plus important à fournir est le renseignement du bon de commande ! La conception et la présentation de celui-ci doivent donc être particulièrement soignées.

## Environnement technique

Nous allons décrire, un à un, tous les éléments techniques qui composent l'architecture retenue, répondant aux besoins décrits dans le cahier des charges. Voici les principaux : système d'exploitation, serveur HTTP gérant le protocole SSL (pour le paiement sécurisé),

serveur de base de données, langage de script, moteur de recherche. L'architecture retenue résulte du meilleur rapport sécurité/fiabilité/qualité de réalisation, au vu de l'ensemble des offres disponibles sur le marché.

### **Système d'exploitation : Linux**

Linux est un système d'exploitation dont le source est produit par différents groupes de travail, répartis dans le monde, orchestrés par Linus Torvald. À la base, Linux ne constitue que le code source d'un noyau (kernel), compilable sur la plus grande majorité des plates-formes. De multiples extensions, logiciels, langages de programmation (Apache, Samba, Perl, Xfree86, etc.) viennent s'y ajouter, et permettent ainsi à Linux d'être le système d'exploitation le plus complet du marché.

Autour de Linux est née l'idée de travail en collaboration. Des milliers de personnes, de l'étudiant au chercheur, joignent leurs efforts pour faire de Linux une plate-forme fiable et performante, tout en mettant à la disposition de tout un chacun le source de leurs travaux. Toutes les briques logicielles ainsi réalisées sont estampillées au label OSS (Open Source Software).

À l'inverse des logiciels propriétaires, un logiciel OSS est libre de diffusion et d'exploitation (aucune licence n'est requise) et, pour la plupart, totalement gratuit.

### **Serveur http : Apache**

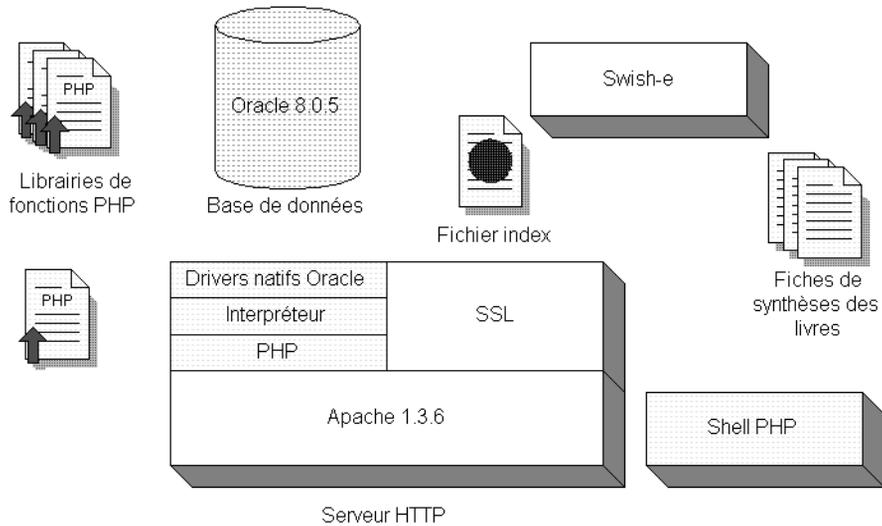
Apache est la référence en matière de serveur HTTP. Selon un sondage réalisé par Netcraft en mai 1999, il était utilisé par plus de 57 % des sites web. Il fait partie des logiciels OSS du monde Linux. Son nom vient du fait qu'il est constitué de codes existants ainsi que de certains modules qui lui sont adjoints (patches).

Le serveur Apache s'adapte selon les besoins, par l'ajout ou la suppression de modules spécifiques. Encore récemment, le changement de modules nécessitait une recompilation du serveur HTTP ; il est maintenant possible de charger des modules dynamiquement.

Pour notre projet, les modules suivants sont nécessaires :

- **mod\_ssl**. Permet l'utilisation du système de chiffrement SSL (Secure Socket Layer). Avec ce système, toutes les requêtes sont chiffrées afin d'améliorer la confidentialité, c'est le protocole HTTP sécurisé (HTTPS). Le chiffrement s'effectue sur 128 bits.
- **Open\_ssl**. Ce module réalise une implémentation des protocoles TLS (Transport Layer Security) et SSL.
- **mod\_perl**. Permet au serveur Apache d'interpréter le langage Perl, nécessaire au fonctionnement d'un serveur d'indexation.
- **mod\_php**. Ce module est nécessaire pour que le langage PHP soit interprété par le serveur HTTP. Php aurait pu être installé en tant que CGI, mais l'installation en module améliore nettement les performances par rapport au fonctionnement CGI standard, grâce à une interconnexion plus efficace entre le serveur HTTP et les environnements d'exécution des scripts. Avec le CGI standard, lorsqu'on exécute un script, le serveur HTTP crée un nouveau processus, puis lance l'interpréteur PHP, ce qui constitue une opération très coûteuse en temps et en ressources machines. Grâce à ce module, ce n'est plus le cas, tout est intégré dans le serveur.

**Figure 12.1**  
Schéma récapitulatif de la plateforme.



### Base de données : Oracle 8

Dans ce type de projet, comme dans bien d'autres, le SGBDR est la pierre angulaire de l'application. Quel que soit le type de technologie employée, client-serveur, intranet, L4G, langage de scripting, il demeure le centre vital qui distribue ses données à qui les réclame.

Oracle est l'un des premiers éditeurs de bases de données à s'être tourné vers Linux, en y apportant tous ses SGBDR depuis la version 8.0.5. C'est pourquoi, alors que de nombreux autres SGBDR existent sur Linux (par exemple, MYSQL), Oracle a été choisi pour les prestations de suivi garanties par son éditeur.

### Moteur de recherche : Swish-e

Un moteur de recherche est un logiciel qui permet d'effectuer une recherche rapide sur un grand nombre de documents indexés. Un moteur de recherche efficace est également un moteur d'indexation.

Le moteur d'indexation le plus répandu à l'heure actuelle est Swish-e (Swish-Enhanced) ; c'est tout naturellement qu'il a été retenu pour la conception de ce site. Swish-e est un moteur d'indexation de pages web ou autres fichiers texte (documents Word, par exemple). Il permet d'effectuer des recherches en mode « Full text » sur la globalité du document, ou de restreindre ces recherches à des balises HTML (les métatags). Il est gratuit, et livré avec des programmes Perl qui permettent à quiconque de créer et de maintenir ses propres fichiers d'index. Voici les caractéristiques de cet outil :

- **rapidité** : une recherche sur un millier de documents ne prend que quelques secondes ;
- **flexibilité** : de nombreuses options de configuration permettent de le paramétrer et de l'utiliser à sa convenance ;

- **puissance** : il permet les recherches en mode « Full text » ou restreintes à des balises HTML (métatags). Il accepte les opérateurs ET, OU et NON, le parenthésage et la troncature droite (le début d'un mot suffit pour trouver ce mot).

## Spécifications fonctionnelles

*Les spécifications fonctionnelles décrivent de manière précise chacun des traitements effectués par l'application. Elles permettent de dégager une structure générale du modèle conceptuel de données, en définissant les tables majeures et leurs interactions. En voici les grandes lignes.*

### Architecture des traitements

#### Mise à jour automatique des données de référence

Les informations relatives aux ouvrages présentés sur le site proviennent essentiellement (80 à 90 %) de deux sources de données complémentaires. L'une sert à alimenter la base avec tous les nouveaux ouvrages, l'autre sert à mettre à jour toutes les données qui concernent le prix et l'état du stock des livres. Ces deux sources proviennent d'un site central. Elles se présentent sous forme de fichiers structurés, qui seront automatiquement chargés dans la base de données Oracle de la façon suivante : une fois par jour pour le premier, deux fois par jour pour le second. Toutes les autres informations seront saisies manuellement au moyen d'une petite application intranet, dédiée à l'enrichissement des données relatives aux ouvrages.

#### Mise à jour depuis les formulaires du site

Les données saisies depuis le site web, et enregistrées dans la base, décrivent les coordonnées des utilisateurs, ainsi que les caractéristiques de leurs commandes.

Les coordonnées de l'utilisateur sont mémorisées. Dans un premier temps, elles permettent d'envoyer un colis. Dans un second temps, cela épargne de les saisir de nouveau lors des prochaines commandes.

Les commandes sont enregistrées, puis traitées ultérieurement manuellement. L'utilisateur peut consulter l'historique de toutes ses commandes.

#### Le caddie

L'application laisse à l'utilisateur la possibilité de choisir plusieurs ouvrages et de les placer dans un caddie, ou panier, avant de passer commande. L'utilisateur a la possibilité de gérer lui-même son caddie, en y ajoutant ou en y supprimant les ouvrages de son choix.

Le caddie n'est en aucune façon sauvegardé dans la base. Sa durée de vie n'excède pas celle de la visite de l'utilisateur.

#### Étude de la transaction du processus de paiement sécurisé

La saisie du numéro de carte de crédit doit s'effectuer de manière sécurisée, en cryptant le transfert HTTP, via le protocole SSL. La commande et le numéro de la carte sont stockés dans

la base, jusqu'au traitement de la commande. Le traitement s'effectue au moyen d'un PAD relié à une banque. Cette dernière valide la transaction. À cette étape, le numéro de la carte de crédit est supprimé de la base de données.

## Le maquetage

*Il est temps de représenter de manière concrète ce qui est suggéré par les spécifications fonctionnelles. La maquette, au format HTML, va représenter toutes les pages de l'application, elle n'est en aucun cas un prototype de l'architecture. Elle ressemble plutôt à un décor cinématographique, sans aucun lien avec des traitements ou une base de données. Elle permet de dresser la liste exhaustive de tous les types de pages rencontrées, de valider l'enchaînement des écrans(cinématique), ainsi que leur apparence, et d'affiner le dictionnaire des données.*

## Interface et ergonomie

Une ergonomie adaptée est indispensable pour attirer des internautes de plus en plus exigeants, habitués à naviguer à travers des interfaces intuitives, qui permettent de trouver rapidement l'information dont ils ont besoin. Aussi, un site doit-il être conçu en ayant à l'esprit deux points fondamentaux :

1. Les pages ne doivent pas être surchargées ; chacune doit contenir une seule information majeure, clairement identifiée.
2. Le nombre de navigations entre les pages pour atteindre l'information recherchée doit être réduit au maximum.

Afin d'obtenir des pages claires et lisibles, la maquette doit combiner ces deux principes.

## La maquette

On confond souvent le prototype de la maquette avec la maquette elle-même.

Le prototype permet de valider toute la chaîne technique, depuis le navigateur jusqu'à la base de données. En général, il est constitué de deux pages. La première fait appel à un formulaire (avec envoi de données vers le serveur) ; la seconde au résultat attendu (retour de la réponse du serveur vers le navigateur client). Ce prototype permet de vérifier que toutes les couches techniques sont bien traversées, dans un sens comme dans l'autre (par exemple, navigateur → serveur HTTP → page PHP → middleware → pilote de la base de données → procédures stockées → données).

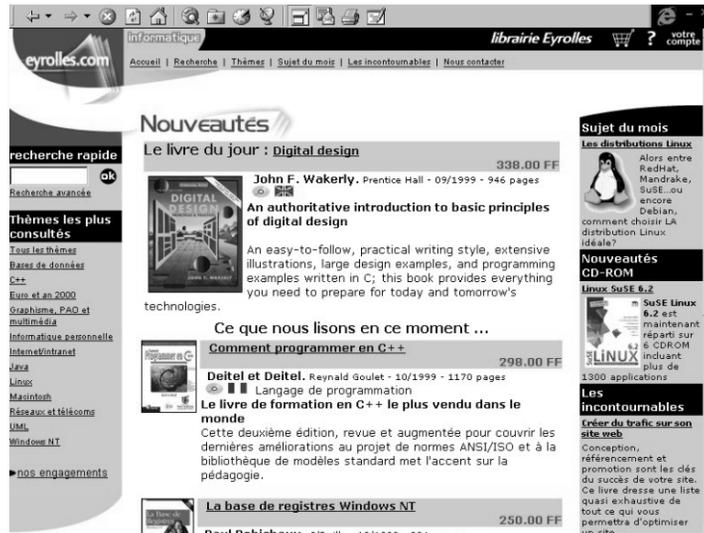
La maquette a pour but de valider l'ensemble des données et des traitements, la navigation (cinématique) et l'aspect graphique. En aucun cas, elle ne valide l'aspect technique du site. Ainsi, aucune des pages présentes dans une maquette n'est dynamique, aucune ne fournit un accès à une base de données. C'est le décor. Ce qu'il y a derrière est laissé à la charge du prototype.

## Exemples de pages

Cette page permet de décrire la structure générale de l'ensemble des pages du site.

Figure 12.2

La page d'accueil du site marchand des éditions Eyrolles.



Elle est composée des éléments suivants :

- un bandeau horizontal, qui permet l'accès à l'ensemble des principaux menus ;
- un bandeau situé à gauche, qui permet un accès rapide aux ouvrages, par recherches multicritères ou thématiques ;
- un bandeau situé à droite, qui renseigne le visiteur sur le contenu éditorial. L'éditorial apparaît uniquement sur la page d'accueil ; son contenu, à la fois sobre et riche, doit retenir le visiteur ;
- le corps de la page, ou contenu. Ici, la page présente les dernières nouveautés parues.

Figure 12.3

La page de recherche d'un livre.



## Page de recherche

Dans toutes les pages suivantes, on retrouve les bandeaux haut et gauche, grâce auxquels on peut, d'un clic, changer d'orientation de navigation, soit en modifiant la recherche ou la sélection thématique, soit en changeant de menu principal :

- le contenu présente tous les critères de recherche pertinents pour la recherche d'ouvrages ;
- un accès rapide par code ISBN ou par code Article est également prévu.

Figure 12.4

*La page de résultat d'une recherche de livres.*



## Page de résultat de recherche

Si la recherche précédente a produit plusieurs réponses, son résultat est présenté sous forme de liste, qui résume les principales caractéristiques des ouvrages.

Notez que les deux bandeaux de navigation haut et gauche sont toujours présents (figure 12.5).

## Page de description d'un ouvrage

Cette page décrit un ouvrage de manière plus précise.

Outre les caractéristiques qui figurent dans la liste de résultat, elle présente un résumé de l'ouvrage, sa couverture, ainsi que son thème. Elle permet également d'ajouter le livre dans son panier, et d'en passer immédiatement commande (figure 12.6).

## Page de description du contenu du panier

Enfin, on trouve la page qui répertorie les ouvrages retenus dans le panier.

Elle permet de modifier le contenu du panier (ajout ou suppression). Elle ne constitue pas la commande en tant que telle, mais l'avant-dernière étape.

Figure 12.5

La page de description d'un ouvrage.

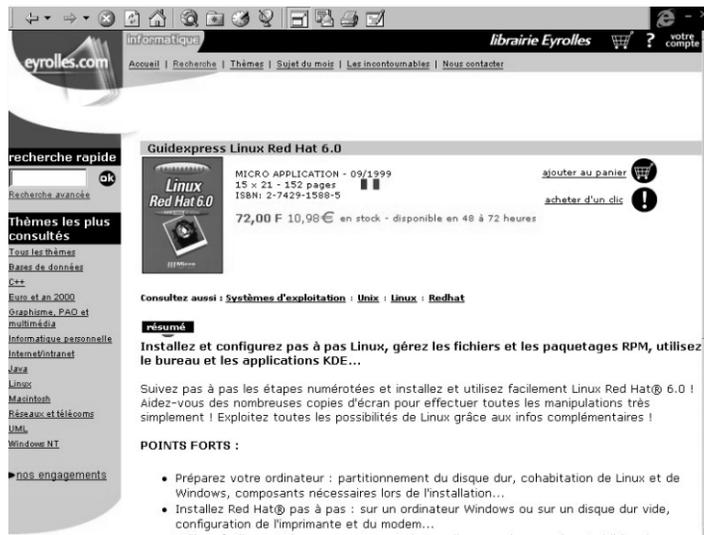


Figure 12.6

La page de description du panier de l'utilisateur.



## Itérations

Pour arriver à ce résultat, plusieurs présentations successives (itérations) sont indispensables, afin d'affiner tous les concepts mis en œuvre.

Deux, trois itérations au maximum, sont nécessaires pour aboutir à la création d'un site. Au-delà, on peut considérer que leur organisation n'a pas fonctionné.

La première itération consiste à définir les exigences du client, en s'appuyant sur le cahier des charges. En fonction de cela, le fournisseur, fort de son expérience, doit rapidement concevoir toute la cinématique du site (mais pas encore son ergonomie).

Une équipe de concepteurs et de graphistes se met ensuite au travail. Ils doivent se conformer à la charte graphique du client, si elle existe. Ils produisent l'ensemble du site avec des pages HTML, qui intégreront la totalité des données ainsi que la simulation des traitements.

La seconde itération est la présentation du résultat obtenu. Elle est elle également très importante, car elle permet de valider tous les concepts mis en œuvre : traitements, cinématique, ergonomie et données.

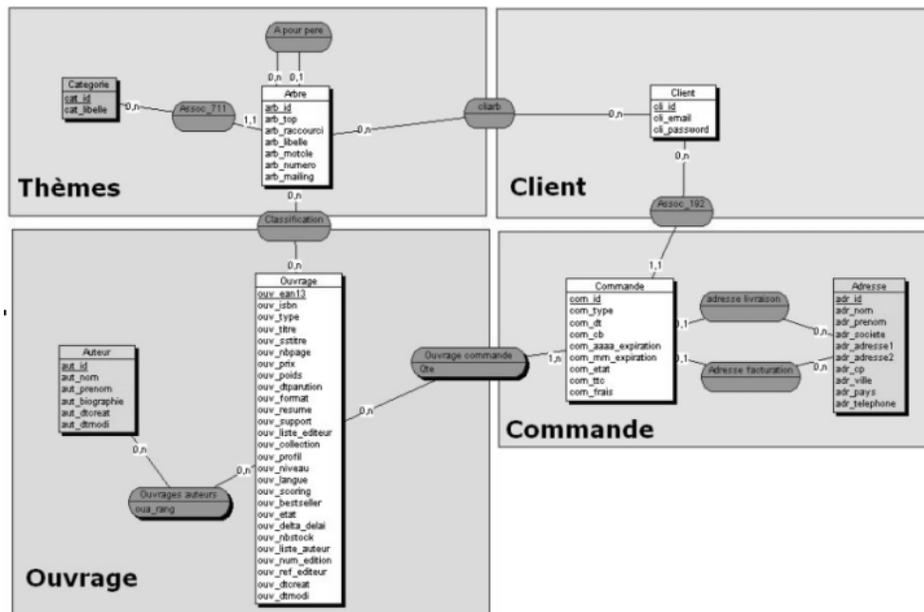
La maquette est enfin validée pour la réalisation de l'ensemble du site.

## Le Modèle conceptuel de données

Tous les éléments sont à présent réunis pour constituer le Modèle conceptuel de données (MCD). Les spécifications générales ont permis de comprendre comment chaque élément interagit avec les données présentes dans la maquette.

### Le MCD

Figure 12.7  
Le modèle  
conceptuel  
des données.



### Entités majeures

On appelle entités majeures toutes les tables qui correspondent aux grandes notions que le projet invoque, et qui sont indispensables. Ici, ces tables sont au nombre de quatre. Une table permet de gérer les ouvrages, une autre les clients, une autre les commande, et la dernière les thèmes associés aux ouvrages. Ces quatre tables constituent le cœur du MCD. Toutes les autres ne serviront qu'à préciser chacune d'entre elles.

## **Entités mineures**

Les entités mineures sont les tables qui permettent de compléter les descriptions des entités majeures. Ces tables sont au nombre de deux : la table des auteurs (liée à la table ouvrages) et la table des adresses de livraison (liée à la table commandes).

## **Entités de paramétrage**

Les entités de paramétrage contiennent les données qui permettent de paramétrer l'application. Leur utilisation autorise une certaine souplesse. Ainsi, un module d'administration peut en modifier la valeur. Mais, en aucun cas, la modification ne doit porter sur le code des pages réalisées. Une seule table de ce type est présente dans le MCD. Elle décrit les grandes catégories de thèmes, par exemple l'informatique ou le management.

## **Spécifications détaillées**

La maquette validée et le MCD terminé, nous allons nous appliquer à préparer la phase de développement. Les spécifications détaillées ont pour objectif de décrire toutes les données, tous les traitements et tous les événements de l'ensemble des pages qui seront soumis aux développeurs

## **Enjeu**

Avant de débiter la réalisation, la description des spécifications détaillées est indispensable pour lever toutes les ambiguïtés qui peuvent survenir lors du développement. Ces documents sont également l'occasion de recueillir l'accord de toutes les parties concernées (client, fournisseur, maître d'ouvrage et maître d'œuvre) sur ce qui va être livré à la fin du projet, car ils décrivent toutes les règles de gestion mis en œuvre.

## **Que décrire ?**

La description des spécifications doit répondre à toutes les attentes du programmeur, afin qu'il puisse procéder au développement de la page. La page doit contenir toutes les données présentes, toutes les règles de gestion de la page, tous les traitements effectués ainsi que leurs algorithmes, et tous les noms des pages auxquelles il est possible d'accéder depuis cette page.

## **Modèle de document**

### **Titre**

Titre de la page.

### **Image**

Insertion de l'image de la page.

## Navigation

<b>Bouton Bxx</b>	Description, nom de la page appelée
<b>Lien Lx</b>	Description, nom de la page appelée

## Association MCD

Champ de la page	Champ de la table	M	O	Valeurs d'initialisation
Libellé Lxxx	Table. Colonne	O	N	V1
M Modifiable, O Obligatoire				

## Règles de gestion

Règle	Description
R1	Si valeur saisie = XX, alors champ YY obligatoire

## Contrôles de saisies

Champ	Contrôle
Montant visé	La valeur saisie doit être numérique (de type 14,4)
Date de début	La date saisie doit être inférieure à la date de fin

## Traitements

Traitement	Description
T1	Mise à jour de l'enregistrement...
T2	Tri par ordre alphabétique...