

IUT Informatique 2009-2010

# Rapport de Stage

Développement d'une application web Intranet

Quentin GAUTIER



Responsable de stage : Gilles NACHOUKI  
Maitre de Stage : Sébastien BURGAUD

2010

## **Remerciement**

*Je remercie d'abord Mr Le Biez (Président Directeur Général) pour m'avoir accueilli durant les dix semaines de stage au sein du groupe SOFIA.*

*Je tiens particulièrement à remercier Mr Burgaud (Pôle informatique) et Mme Liman (Direction contrôle de gestion) pour m'avoir accueilli comme un membre à part entière de leur service et pour m'avoir accordé leur confiance sur le projet à mener.*

*Un grand merci va à Eric Moureaux (Contrôleur de Gestion) qui n'a cessé de me prodiguer ses conseils et ses suggestions justes et pertinentes.*

*Merci aussi aux responsables des douze magasins qui m'ont donné de leur temps pour tester et donner leur avis sur l'application.*

*Mes remerciements aux membres des autres services du groupe SOFIA que j'ai côtoyé quotidiennement et dont j'ai apprécié la gentillesse et la bonne humeur.*

*Je remercie également mon tuteur de stage, Mr Nachouki pour ses conseils avisés.*

## Résumé

Rapport de stage de fin d'études de deuxième année d'IUT Informatique qui s'est déroulé du 12 avril 2010 au 18 juin 2010 au sein de la holding SOFIA.

**Thème** : Optimisation d'un procédé de vérification de données comptables via l'utilisation d'un intranet.

Ce rapport porte sur la mission du développement d'une application s'inscrivant dans l'Intranet d'une société. L'application sera utilisée par les magasins franchisés Douglas de l'Ouest de la France afin de renseigner des données comptables qui seront ensuite traitées par la société SOFIA.

Ce rapport évoque les conditions de mise en place, d'étude et de développement de l'application. Son évolution et ses tests sont aussi traités et analysés.

Un bilan clôture ce document et synthétise l'expérience et le ressenti durant ces 10 semaines de stage.

En annexe, le code de l'application, un exemple d'existant, le cahier de bord et le manuel d'utilisateur à disposition des magasins franchisés Douglas pour l'utilisation de l'application.

## Abstract

Internship report of end study second year on computer sciences license. It's which took place from April 12th, 2010 to June 18th, 2010 in the SOFIA holding company.

**Subject:** Optimization of a process of check of accounting data through the use of an intranet.

This report focuses on the mission of developing an application that fall within the intranet of a company. The application will be used by franchise stores Douglas's' of west of France to learn the accounting data which will then be processed by the company SOFIA. This report discusses the conditions of establishment, development and application development. Its evolution and its tests are also discussed and analyzed.

A summary concludes this document and summarizes the experience gained and felt during these 10 weeks of training.

In annex, the application's source, an example of the actual product, a logbook and user manual available in franchise Douglas stores for the use of the application

## Sommaire

<i>Remerciement</i> .....	2
Résumé - Abstract .....	3
1 Le groupe SOFIA et la filiale Clin d'œil.....	6
1.1 SOFIA : Fiche d'identité .....	6
1.2 SOFIA : Son histoire .....	7
1.3 SOFIA : Son organisation .....	8
1.4 SOFIA : Présentation de sa filiale Clin D'œil. ....	9
2 Introduction .....	10
2.1 Le Cadre .....	10
2.2 La Description de la tâche confiée .....	10
2.3 Le travail de recherche .....	10
3 Analyse des besoins.....	11
3.1 L'existant.....	11
3.1.1 Introduction.....	11
3.1.2 Présentation de la structure du système existant.....	11
3.1.3 Automatismes existants .....	11
3.1.4 Finalité de l'ancien système .....	12
3.2 Analyse des fonctions principales demandées .....	12
3.2.1 Au niveau de l'interface homme-machine.....	12
3.2.2 Au niveau de l'utilisateur principal.....	12
3.2.3 Au niveau de l'administrateur .....	13
3.2.4 Pour tout le monde .....	13
4 Structure de l'application et évolution .....	14
4.1 Application finale.....	14
4.1.1 L'identification.....	14
4.1.2 Le procès verbal de caisse .....	14
4.1.3 Le chiffre d'affaires.....	15
4.1.4 Le fond de caisse .....	15
4.1.5 Le détail des remises en banque .....	15
4.1.6 La fiche de synthèse .....	16
4.2 Evolutions de l'application dans le temps.....	16
4.2.1 Premier essai .....	16

4.2.2 Les modifications apportées .....	17
4.2.3 Structure du code .....	18
5. Le développement .....	19
5.1 Solutions techniques apportées .....	19
5.1.1 Langages utilisés, spécificités .....	19
5.1.2 Cookies, Sessions et url longue .....	19
5.1.3 Structure de la base de données .....	20
5.2 Difficultés rencontrées .....	21
5.2.1 La modélisation de l'application.....	21
5.2.2 Prévoir et s'adapter .....	22
6. La Phase de tests et de validations.....	23
6.1 Tests et validation en interne .....	23
6.1.1 Validation de l'application par rapport aux exigences demandées .....	23
6.1.2 Tests internes .....	24
6.2 Choix des magasins tests et lancement du test à plus grande échelle.....	25
6.2.1 Echantillon des magasins et critères de sélection.....	25
6.2.2 Déroulement du lancement de l'application sur les magasins tests.....	25
6.3 Erreurs détectées via les phases de test et remonté d'information.....	26
6.3.1 Erreurs de calculs de l'écart de caisse et approximation du résultat.....	26
6.3.2 Retours des magasins.....	26
7. Cahier des charges.....	27
7.1. Présentation générale du problème.....	27
7.1.1 Projet .....	27
7.1.2 Contexte .....	27
7.1.3 Énoncé du besoin (finalités du produit pour le futur utilisateur tel que prévu par le demandeur).....	28
7.1.4 Environnement du produit recherché.....	29
7.2. Expression fonctionnelle du besoin.....	29
7.2.1 Fonctions de service et de contrainte .....	29
7.2.2 Niveaux des critères d'appréciation et ce qui les caractérise .....	30
7.3. Cadre de réponse .....	31
7.3.1 Pour chaque fonction, solution proposée.....	31
7.3.2 Pour l'ensemble du produit.....	31
Synthèse du stage .....	33

# 1 Le groupe SOFIA et la filiale Clin d'œil

## 1.1 SOFIA : Fiche d'identité

Raison Sociale:	Société Financière Atlantic
Statut juridique:	Société Anonyme à Directoire et Conseil de surveillance au capital de 3 125 000 €
Code APE	741 J: Administration de sociétés
Date de création:	1990 = création de SOFIA 1961 = création de la société Atlantic Distribution
Adresse:	2 rue Georges Clémenceau 44612 SAINT-NAZAIRE
Effectif:	Société mère = 25 Groupe = 1007 (751 CDI )
Chiffres clés :	Résultat d'exploitation 2009= 4 230 902 € Bénéfice exercice 2009 = 3 727 511 € Résultat consolidé 2009 = 2 267 798 €  C.A. consolidé 2009= 104 422 000 € C.A. consolidé 2008 = 98 919 000 € Taux de croissance = + 5, 6 %  Capitaux Propres consolidés 2009 = 22 164 185 € Capitaux propres consolidés 2008 = 21 022 786 €  Total Bilan consolidé 2009 = 78 552 336 € Total Bilan consolidé 2008 = 87 057 966 €

## 1.2 SOFIA : Son histoire

1961	Création de la société Atlantic Distribution Distribution de fioul domestique Secteur très localisé autour de Saint-Nazaire
1970	Développement des activités : négoce de robinetterie industrielle Distribution des marques Butagaz et Camping Gaz
1973	Création du premier magasin de bricolage Franchise à l'enseigne OBI
1978	Création du premier magasin de jardinage Franchise à l'enseigne JARDILAND
1982 à 1988	Stratégie de croissance interne, investissements très importants Extension des secteurs d'activités Création de la SAS Loire Entrepôts Transports (1985)
1990	Filialisation des activités Atlantic Distribution devient la société holding SOFIA
1994 à 1995	Acquisition et développement du groupe des sociétés CLIN D'ŒIL Activité de commerce de parfumerie (6 parfumeries à fin 95)
1998	Ouverture du premier magasin sous l'enseigne Elytis (Nantes)
2005-2006	40 parfumeries franchisées ou labellisées Elytis en France, de Cherbourg à Lourdes  Nouveau contrat de franchisage Magasins de Bricoland passent sous l'enseigne Leroy Merlin
2010	50 parfumeries franchisées ou labellisées Douglas en France, de Cherbourg à Biarritz

### 1.3 SOFIA : Son organisation

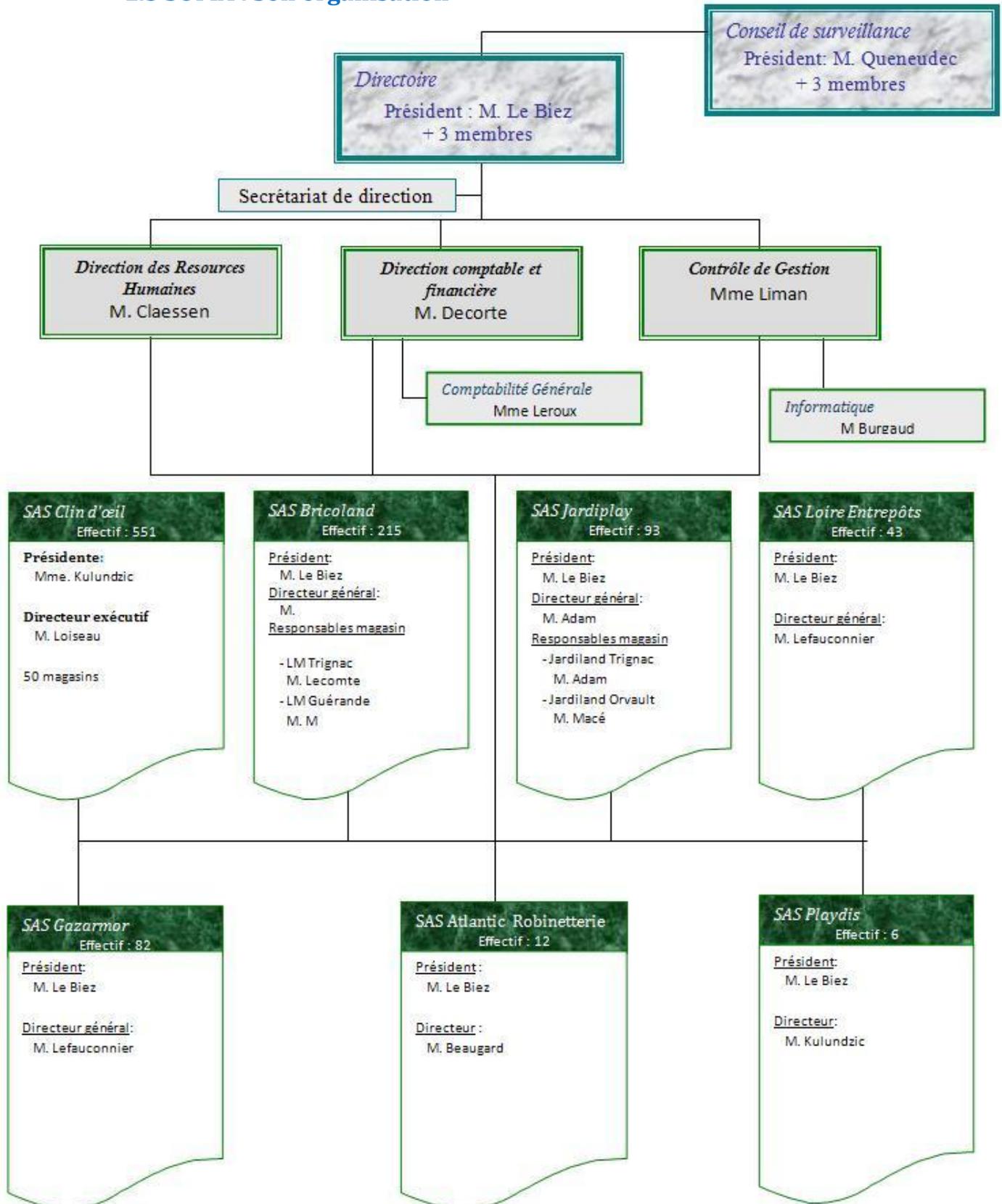


Figure1 : Organisation du groupe SOFIA

## 1.4 SOFIA : présentation de sa filiale Clin D'œil.

La Société Financière Atlantic n'est pas une holding pure. Elle ne se contente pas de percevoir les bénéfices de ses filiales. En effet SOFIA assure la gestion administrative, comptable, financière et informatique de ses filiales. Elle est en quelque sorte une société prestataire de services. Ces services font l'objet d'une facturation interne tous les mois. De ce fait, ses filiales peuvent se concentrer à leur métier de base : le commerce de gros ou de détails.

Le groupe SOFIA s'est développé depuis sa création. Elle a mené une stratégie de croissance externe et s'est diversifié dans différents secteurs d'activité. Cette diversification et cette croissance ont abouti à la création d'une société holding et à la filialisation de ses activités (sept sociétés filles + quelques sociétés immobilières) La filiale la plus importante est la filiale Clin d'œil avec 50 magasins franchisés Douglas sur l'Ouest de la France et un effectif de 551 personnes pour 2010.

La société Clin d'œil intervient sur le marché du bien être (parfumerie, cosmétique, soins du corps, ...). Le marché national est très concentré avec quelques groupes importants tels que Marionnaud (environ 600 magasins), Sephora (environ 400 magasins) et quelques parfumeries indépendantes telles qu'Anthynéa à Saint-Nazaire. Au niveau national, la société se classe quatrième rang en termes de chiffre d'affaires réalisé. Cependant la société couvre principalement l'Ouest de la France et est présente à Rochefort. Ses magasins se situent en centre ville ou centres commerciaux sur des emplacements stratégiques.

Constituée en 1991, la S.A.S Clin d'œil a un capital de 5 000 000€ et la participation de Sofia est de 95.01%.

## **2 Introduction**

### **2.1 Le Cadre**

Le stage s'est déroulé du 12 avril 2010, au 18 juin 2010 pour un total de 10 semaines de stage au sein du groupe SOFIA. Il s'est déroulé pour la filiale de SOFIA Clin D'œil pour les magasins franchisés Douglas. La société SOFIA, dans un souci d'amélioration de procédés, a donc mis en place ce stage pour la création d'une nouvelle application. SOFIA étant une société familiale, son fonctionnement est très basé sur le dialogue, une chose qui a joué de façon très importante sur le cours de mon stage ou l'information transitait très rapidement et influait donc le développement de l'application.

### **2.2 La Description de la tâche confiée**

Dans un souci d'amélioration et d'optimisation de procédés comptables, il m'a été confié la tâche de développer une application web, disponible via l'intranet Clin d'œil. La réalisation de l'application s'est déroulée avec l'aide de divers intervenant de la société, contrôleur de gestion (Magali LIMAN et Eric Moureaux), ou magasins Douglas (pour les tests et les réactions). L'application vient remplacer un système existant en place dans la quasi-totalité des magasins du groupe et apportera une solution technique à des problèmes de traitement de données rencontrés par SOFIA.

### **2.3 Le travail de recherche**

Le travail de recherche s'est effectué en plusieurs parties. Premièrement la recherche et la spécification de l'application ainsi que le cahier des charges se sont déroulés sur une dizaine de jours, afin de commencer à créer une structure de l'application stable et une structure pour la base de données.

Un autre travail de recherche s'est effectué sur les langages de programmation. Ne maîtrisant pas ou très peu le JavaScript il m'a fallu, à l'aide de site web et d'un manuel, apprendre le fonctionnement du langage et son utilisation. Tout ceci est bien détaillé dans le rapport de stage qui suit avec une analyse des besoins de l'entreprise, un détail du développement et l'application allant jusqu'aux tests de celle-ci. Des recherches aussi sur les algorithmes à utiliser pour optimiser au mieux l'application ont été réalisées.

## 3 Analyse des besoins

### 3.1 L'existant

#### 3.1.1 Introduction

Actuellement la société SOFIA récupère, sur des documents imprimés Excel, différentes données nécessaires à la comptabilité. (Voir en annexes les documents de l'existant). On retrouve donc les feuilles de caisse et procès verbaux remplis par chaque magasin le soir ou lendemain matin. Les magasins franchisés de la filiale Clin d'œil s'étalant sur tout l'Ouest de la France, réceptionner les fiches peut parfois mettre une semaine. D'un autre côté, les différents instruments présents en magasin envoient directement dans une base de données le détail précis des ventes. Ceci étant récupéré par la comptabilité de Sofia, ils peuvent commencer à calculer des chiffres d'affaire et autres données essentielles à la comptabilité. Cependant sans les informations entrées par les magasins, il est difficile de faire ressortir les erreurs. C'est pourquoi il est nécessaire d'accélérer l'arrivée d'information pour les formulaires de feuille de caisse.

#### 3.1.2 Présentation de la structure du système existant

L'actuel système utilisé demande aux magasins de remplir sur des documents Excel ou Open Office deux fiches :

Le premier, appelé « Procès Verbal de Caisse » regroupe les informations des chèques dit différés possédés par le magasin. Les chèques différés sont une offre faite aux clients pour acheter des produits tout en payant à une date ultérieure fixée. Le problème posé par ces chèques différés c'est qu'ils doivent être renseignés tous les jours sur la fiche jusqu'à la remise en banque. C'est un problème que devra donc gérer l'application. Sur cette même feuille, le magasin reporte le total des fonds de caisse au soir, c'est-à-dire après la journée de vente.

Le second, appelé « Feuille de Caisse » est le formulaire qui reprend les valeurs du fond de caisse du matin, des remises clients que ce soit espèces, chèques ou cartes bancaires. Les magasins renseignent aussi les valeurs des différents bons d'achats. Chose qui deviendra inutile car ces renseignements ne sont pas assez précis pour la trésorerie de SOFIA. C'est donc un champ qui sera revu dans l'application finale.

Enfin un champ « ventes » assez controversé qui au final gardera la structure mise en place sur le document Excel et non celui prévu par le cahier des charges.

En effet, le cahier des charges spécifie que l'application devait prendre en compte une plus grande précision sur les ventes. Une ventilation par taux de TVA et par type de ventes avait été spécifié. Détail qui au final sera supprimé.

#### 3.1.3 Automatismes existants

Une fois les champs remplis, l'application Excel ou Open Office calculait automatiquement ce qui est appelé un « écart du jour », qui vient, s'il existe, soit en crédit, soit en débit selon sa valeur calculée. Un écart du jour apparaît s'il y a une différence entre la valeur reportée depuis le procès verbal et la différence entre les sommes crédits et débits enregistrées dans le formulaire. Si une erreur apparaît, elle est

directement affichée en rouge, ce qui permet à l'utilisateur de voir une erreur et d'essayer de la trouver. Un champ pour justifier l'erreur est aussi prévu.

### **3.1.4 Finalité de l'ancien système**

Le tout est imprimé et est joint avec tous les justificatifs tels que les remises de chèque, les remises d'espèces, les bons d'achats, etc. Les feuilles sont distinguées grâce à un code magasin (ici CEX pour Douglas), une date, et un numéro de Folio qui prend le numéro de section comptable du magasin concaténé à un numéro qui s'incrémente chaque jour d'ouverture du magasin et est remis à 1 chaque début du mois. On a donc pour le magasin de Lourdes, un folio 2825 pour 28 étant le numéro comptable et 25 le 25eme jour d'ouverture du mois. Tout ce système d'identification devra être conservé pour gérer la base de données de l'application web.

## **3.2 Analyse des fonctions principales demandées**

### **3.2.1 Au niveau de l'interface homme-machine**

Première exigence vis-à-vis de l'application, celle-ci doit en plus de reprendre ce que faisait l'ancien système existant, elle doit être d'une très grande simplicité d'utilisation. En effet, le personnel qui sera utilisateur de l'application ne connaît pas toujours très bien l'environnement informatique. L'application doit donc répondre à des exigences de lisibilité, d'ergonomie et de facilité.

L'application devra être présentée sous la forme d'une succession de formulaires dans des tableaux afin de rendre ça simple et accessible. La mise en forme devra garder celle utilisée dans le document Word pour que l'utilisateur n'ait pas à chercher l'information à l'écran.

L'application est développée pour fonctionner sous Internet Explorer 8, ce qui entraîne une mise à jour des machines de chaque magasin.

### **3.2.2 Au niveau de l'utilisateur principal**

L'application web devra, via une interface composée d'une succession de formulaires web, remplir une base de données des différents champs définis dans le « 3.1 l'existant ». Il est donc demandé de créer cette base de données et d'y faire figurer tous les champs nécessaires pour qu'ensuite SOFIA puisse les récupérer et les traiter avec leur logiciel « ReportOne ». Les différents calculs ne seront pas systématiquement enregistrés dans la base de données mais seront là à valeur informative pour l'utilisateur. Par exemple, les totaux crédits et débits seront calculés par l'application, mais n'auront pas d'enregistrement dans la base de données.

Les chèques différés seront traités dans une page dédiée, avec un bouton « ajouter chèque différé » qui amènera à un formulaire qui exigera de l'utilisateur la saisie de tous les champs, c'est-à-dire nom et prénom de la personne ayant remis le chèque, la date d'émission, et de remise en banque, et enfin le montant. La date de remise en banque sera vérifiée par l'application et devra bien être supérieure à la date de remise en banque. De plus, ces dates seront sélectionnées via un calendrier dynamique en JavaScript. Une fois validé, l'utilisateur se retrouvera sur la page dédiée aux chèques différés et verra apparaître sous forme de tableaux la totalité des chèques en attente de remise en banque. Si la date de remise en banque de l'un des chèques est dépassée, l'utilisateur sera immédiatement prévenu, et il pourra à se moment là l'enlever de la liste en le remettant en banque.

Pour éviter des erreurs de calculs, l'application proposera pour la feuille de caisse, autant de champs que de caisses disponibles dans le magasin. Sachant que ce nombre sera disponible dans la base de données en tant que donnée fixe, et est donc variable d'un magasin à l'autre. L'utilisateur remplira donc ses champs là et l'application calculera automatiquement la valeur totale qu'elle enregistrera dans la base de données contrairement aux valeurs des caisses.

Un récapitulatif global sera à disposition de l'utilisateur sur une seule page avec la possibilité de modifier les champs si besoin. Une version dite « brouillon » pourra alors être enregistrée et consultée le matin de la prochaine ouverture du magasin. Cette version pourra être éditée et devra être validée pour passer à la feuille de caisse suivante.

Une aide accessible depuis chaque page et pour chaque champ important devra être disponible. En effet pour les champs à remplir une bulle d'aide sera disponible permettant d'aider l'utilisateur à chercher au bon endroit l'information. Une page entière sur le fonctionnement de l'application et des conditions exigées pour remplir correctement les formulaires sera aussi disponible sur un lien « aide ».

### **3.2.3 Au niveau de l'administrateur**

Pour un utilisateur de rang administrateur, l'application proposera de pouvoir interagir sur la base de données en proposant de modifier les tables à valeur fixe comme la table Tab\_Magasin qui répertorie les numéros de comptes associés aux différents libellés pour chaque magasin. Un magasin pouvant remettre ses chèques ou espèces à une certaines banques à une date donnée peut très bien les donner à une autre banque à une autre date en fonction des offres de celle-ci. C'est pourquoi les administrateurs de l'intranet pourront modifier l'affectation d'une banque à tel ou tel magasin via une interface qui leur sera prévue.

### **3.2.4 Pour tout le monde**

Enfin tous les utilisateurs, magasins ou administrateurs, pourront aller voir sur l'application un historique des feuilles de caisse par magasin et par date de journée de traitement. Via la page de l'identification, il sera possible de choisir l'historique plutôt que de traiter une nouvelle feuille de caisse, en inscrivant la journée que l'on souhaite visualiser sous le format de la page de synthèse habituelle.

## 4 Structure de l'application et évolution

### 4.1 Application finale

L'application dans sa forme finale se découpe en six parties principales visibles. L'identification, le procès verbal de caisse, le chiffre d'affaires, le fond de caisse, le détail des remises en banque, et la fiche de synthèse. Ces six éléments constituent un formulaire répondant aux exigences demandées.

#### 4.1.1 L'identification

L'identification est la première page que l'utilisateur rencontrera dans la navigation de l'application web. Elle permet de récupérer le nom et le prénom de la personne qui saisie les informations de la feuille de caisse. Cela sert juste à titre informatif. La page de l'identification est séparée en deux parties. La première partie est l'identification, nécessaire pour la saisie de la feuille de caisse intranet, et la seconde partie concerne l'historique enregistré sur une base de données. Cette page est primordiale pour pouvoir identifier le magasin, et avoir la journée à traiter pour les pages suivantes qui se basent sur ces informations pour créer la page et pour enregistrer les informations au bon endroit.

Une vérification de la base de données sur le champ « Brouillon » est effectuée à l'identification. En effet si le champ est actif dans la base de données, cela signifie qu'un brouillon de données a été enregistré depuis la synthèse précédente, et est automatiquement restauré. (Voir 4.1.6)

#### 4.1.2 Le procès verbal de caisse

Le procès verbal de caisse regroupe les chèques différés en attente dans le magasin. Un magasin pouvant offrir aux clients une créance par chèque remis en banque à une date ultérieure, ils sont stockés dans le procès verbal de caisse. L'application va donc directement chercher dans la base de données en fonction de la date et du numéro du magasin stocké via l'identification, les chèques à faire apparaître. Trié par date de remise en banque, les magasins ne peuvent oublier d'en remettre à la banque. De plus, si la date de remise en banque est égale ou antérieure à la date de journée traitée, les informations du chèque apparaissent avec une police d'écriture rouge. A partir de cette page, l'utilisateur a le choix entre, ajouter de nouveaux chèques différés, remettre en banque les chèques différés au choix, ou directement passer à la suite.

L'ajout d'un chèque différé se fait via un bouton « ajouter un chèque différé » qui amène sur une nouvelle page demandant le remplissage de cinq champs obligatoires qui sont les dates d'émission, de remise en banque, le nom et le prénom de l'émetteur, et le montant du chèque. L'ajout se fait simplement en pressant le bouton « ajouter chèque » avec un message validant ou non l'ajout du chèque et réinitialisant les champs de la page pour l'ajout éventuel d'un nouveau chèque. Une fois la totalité des chèques différés saisis, un retour au procès verbal fait apparaître les chèques ajoutés à ce jour dans un tableau séparé, avec possibilité de supprimer un chèque.

La remise en banque d'un chèque se fait en sélectionnant simplement les chèques que l'on désire retirer en cochant une boîte de sélection

apparaissant sur la ligne du chèque dans le tableau. Il en est de même pour les rares cas où un client viendrait payer le chèque différé à l'aide d'un autre moyen de paiement. Une fois sélectionné, il faut soit appuyer sur « remettre les chèques en banque » ce qui fera apparaître une page récapitulative des chèques remis, soit appuyer sur valider, qui sauvegardera automatiquement les chèques remis et passera à la page du chiffre d'affaires.

#### 4.1.3 Le chiffre d'affaires

Longtemps nommée « Détail des ventes », ce formulaire a subi un changement de dernière minute le simplifiant de façon brutale. Alors qu'il était demandé un détail des ventes de marchandises et de prestation, ainsi qu'un taux de TVA à 5.5% représentant les quelques rares produits ayant ce taux. A présent cette page ne contient plus que le libellé « chiffre d'affaires » qui correspond donc à la totalité des ventes sans remises et toutes taxes comprises. Il n'y a qu'à mettre le détail par caisse pour que l'application additionne et passe à la suite.

Une aide présente à la fois sur cette page, sur le fond de caisse et sur la page du détail des remises en banque, sous la forme d'un point d'interrogation permet en sélectionnant l'image d'afficher un message d'aide pour déterminer où aller chercher les informations.

#### 4.1.4 Le fond de caisse

Le fond de caisse comme son nom l'indique viendra demander à l'utilisateur de rentrer son fond de caisse, caisse par caisse, le soir même, ainsi qu'un chiffre d'affaires en espèce. La page récupérant automatiquement le fond de caisse du matin via la base de données, et grâce aux informations de la page suivante, il sera possible de calculer un fond de caisse théorique grâce au calcul :

$$\begin{aligned} & \text{Fond de caisse du matin} + \text{C.A. espèces} - \text{Remises en banques des espèces} \\ & = \text{Fond de caisse théorique du soir.} \end{aligned}$$

#### 4.1.5 Le détail des remises en banque

Sans doute la partie la plus longue à renseigner, le détail des remises en banque demande à l'utilisateur de renseigner un certain nombre de champs en fonction du nombre de caisses dans le magasin, et de l'utilisation ou non de cartes Cofinoga ou Accord. Les autres champs apparaissent pour tous les magasins, c'est-à-dire, les libellés « carte bleu », « chèque », « espèces », « monnaie », « American Express », « bons d'achat » et en ce qui concerne les chèques cadeau Douglas. C'est sur cette page là que les informations importantes sont saisies, car l'application a pour but de vérifier ces informations. Une fois la totalité saisie, une validation amène la dernière page récapitulative, la page de synthèse.

(Un exemple du code utilisé pour le traitement de la remise en banque est joint en annexe)

#### 4.1.6 La fiche de synthèse

La synthèse regroupe la totalité des informations renseignées depuis l'identification. Apparaissent ainsi dans des tableaux juxtaposés les détails remplis pour les fonds de caisse, des ventes, des remises et les valeurs des mouvements des chèques différés. A chaque tableau, un bouton de retour vers l'une des pages vues précédemment afin de modifier les informations ou renseigner les champs oubliés a été mis en place.

Sur le bas de la page apparaît après calcul, un « écart du jour » qui correspond à une différence entre les débits et les crédits renseignés dans les formulaires. Si cette différence est nulle, rien n'apparaît, si un écart est détecté, l'application fait apparaître la valeur de l'écart, et un champ justificatif en dessous. A partir de cela, l'utilisateur peut rechercher son erreur, retourner sur le formulaire où il a ciblé la faute et la corriger. Si l'erreur persiste et qu'il ne peut éviter l'écart de caisse avec par exemple une erreur de rendu monnaie. Il pourra indiquer cette erreur dans le justificatif. Si rien n'est renseigné dans le justificatif, l'application ne laisse pas l'utilisateur valider. De plus, pour aider l'utilisateur à trouver son erreur, un message apparaît ici si les fonds de caisse du soir et du soir théorique ne concordent pas. Si ce message apparaît, il saura directement qu'une erreur se situe soit sur le chiffre d'affaires en espèce, soit sur le fond de caisse du soir, soit sur le fond de caisse du matin.

Afin de pouvoir corriger leur erreur ultérieurement, un bouton, sauvegarder brouillon est accessible. Ce qui enregistre entièrement le travail fait et renvoie directement à l'accueil de l'intranet. A l'identification, si un brouillon est détecté, il est automatiquement restauré et l'application s'ouvrira sur la fiche de synthèse dans l'état où elle était avant la sauvegarde. Cette méthode permet en cas d'erreur, de reprendre ce travail le lendemain par exemple.

Pour finir, la validation de la synthèse n'est possible que si deux conditions sont remplies. La première est qu'un chiffre d'affaires des ventes doit être renseigné. Si le chiffre d'affaires indique zéro, il sera impossible de valider. L'autre condition concerne la justification en cas d'erreur, il faudra que le magasin entre un texte quelconque pour au moins valider et confirmer cet écart. Si les deux conditions sont validées, une dernière page de validation demande à l'utilisateur s'il est certain de vouloir valider. Si tel est le cas, la totalité des données seront sauvegardées et envoyées dans la base de données.

## 4.2 Evolutions de l'application dans le temps

### 4.2.1 Premier essai

Au lancement de la programmation de la feuille de caisse, j'ai effectué un premier essai en me basant sur la feuille Excel existante. C'est-à-dire que l'application était découpée en deux parties plus l'identification. Un procès verbal qui est plus ou moins resté le même sur l'application finale, et une partie feuille de caisse qui regroupait la totalité des autres informations. On retrouvait alors sur une seule et même page le détail de toutes les informations demandées en saisie que ce soit chiffres d'affaires, fonds de caisses et remises en banque. On avait aussi à cet endroit le justificatif et l'écart de caisse qui apparaissait. La structure de la page a finalement donné ce qui est aujourd'hui la fiche de synthèse, les champs du formulaire en moins. Ce premier essai

d'application a été abandonné. En effet l'apparition du détail crédit et débit n'avait que peu d'utilité pour des personnes pas ou peu sensible à la comptabilité, de plus certaines informations étaient à revoir, d'où un changement radical de la structure du système.

De plus cette première version possédait trois enregistrements en base de données, un au niveau du procès verbal pour les chèques différés, un au niveau de la feuille de caisse de chaque caisse traitée une à une, et enfin un enregistrement à la validation finale. Ce procédé posait de multiples problèmes et n'était vraiment pas sécuritaire vis-à-vis des informations erronées à corriger. Ces erreurs là ont permises une progression vers un système plus sure afin d'écarter toutes erreurs possible une fois l'enregistrement de la base de données.

## **4.2.2 Les modifications apportées**

### **4.2.2.1 Détails par caisse**

Avant d'apporter un détail par caisse, l'application était conçue pour fonctionner sur toutes les caisses en même temps. On identifiait le magasin, puis le numéro de la caisse traitée pour enfin valider les calculs quand toutes les caisses étaient traitées. Il aurait fallu mettre un système de synchronisation ou de vérification pour utiliser cette méthode, mais elle a vite été abandonnée pour laisser place au système actuel qui sur une seule et même page est demandé le détail des informations par caisse. Les magasins ayant de une à trois caisses, l'application s'adapte au magasin et fait apparaître le nombre de champs nécessaire.

### **4.2.2.2 Rajout de scripts**

Divers scripts ont été ajoutés au fur et à mesure du développement de l'application pour apporter un certain confort à l'application. Ne pouvant servir de sécurité à cause d'une désactivation possible du JavaScript, celui-ci apporte tout de même une certaine aide pour l'utilisation de l'application. Grâce à ses scripts, l'utilisateur n'a pas besoin d'envoyer un formulaire pour passer dans les structures de contrôle en PHP pour savoir s'il a bien rempli les champs demandés. Une erreur sera détectée avant l'envoi et un message apparaîtra prévenant l'utilisateur. De plus, grâce à ce procédé, il n'est pas obligatoire de ressaisir les champs. En effet si aucun script ne vérifiait le contenu envoyé, le traitement généré par le code PHP derrière ferait une redirection automatique vers la page de saisie en effaçant les champs remplis.

De même, les scripts JavaScript permettent l'utilisation du calendrier pour la saisie des champs « dates ». Le format des dates demandées étant du jj/mm/aaaa, il pourrait arriver que certains saisissent un format différent et soit refusé à la validation. Le calendrier permet donc de saisir directement au bon format la date.

(Un exemple des scripts utilisés est présent en annexe.)

### 4.2.3 Structure du code

Pour finir voici comment est agencé le code. Six principaux dossiers regroupent les fichiers par type.

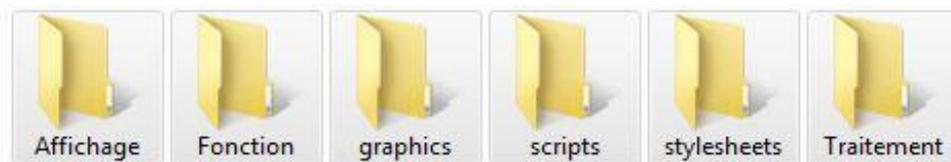
**Affichage :** Contient tous les fichiers qui représentent des pages qui affichent quelque chose. C'est le cas par exemple des pages Identification.php, ou encore Synthese.php

**Fonction :** Toutes les fonctions créées pour l'application sont regroupées dans ce dossier. Elles sont séparées en deux catégories, les fonctions qui permettent de construire les différents tableaux HTML pour mes pages de formulaire, et les fonctions qui permettent la non redondances de code pour le calcul sur les dates par exemple, ou encore pour l'affichage de l'entête.

**Graphics :** Contient toutes les images utilisées. La bannière, le fond de page, ou les différents éléments qui constituent le calendrier comme les flèches de déplacement ou la croix de fermeture.

**Scripts :** Dans ce dossier est stocké tout le code JavaScript pour les différents scripts utilisés dans le code.

**Traitement :** Le dossier le plus important et le plus volumineux, il contient tout le code avec le calcul entre les pages. Les pages de « Traitement » servent à récupérer les variables entrées dans les pages « Affichage » et à effectuer les calculs nécessaires. On trouvera aussi dans ces pages traitement les redirections nécessaires en fonction de l'état de l'application. Enfin les pages traitement exécutent les mises à jour de la base de données et les insertions dans celles-ci.



Structure interne de l'application

## 5. Le développement

### 5.1 Solutions techniques apportées

#### 5.1.1 Langages utilisés, spécificités

Plusieurs langages de programmation ont été utilisés pour le développement de l'application web. Premièrement le HTML et le CSS utilisés tous les deux pour le côté affichage d'information. Ils ne servent qu'à afficher des pages web statiques. Le CSS m'a permis de définir les fonds de l'application et la couleur des tableaux ainsi que la structure principale des pages avec des blocs. Une page est composée d'une bannière identique à chaque page, d'un corps de page, d'un pied de page et d'une partie entête pour certaines pages affichant les informations entrées à l'identification. Le HTML m'a servi, cependant, à construire la mise en page des informations avec des tableaux, et à structurer le reste du code.

Le PHP est un langage qui dynamise les pages web. Grâce à cela il a été possible de faire construire les tableaux affichés pour le formulaire de façon dynamique avec le nombre de caisses par magasin par exemple. Ou encore de pouvoir faire apparaître automatiquement l'entête affichant les noms, prénoms de la personne saisissant l'information, mais aussi le numéro de magasin et le calcul du numéro de folio. C'est avec ce langage aussi que chaque calcul est possible. Les totaux, les écarts de caisse ou de fond de caisse et autres calculs sont réalisés via ce langage de programmation. Enfin le PHP, permet l'enregistrement dans une base de données, ici MySQL, et de traiter les données lues dans la base de données.

JavaScript est un langage de programmation de scripts principalement utilisé dans les pages web interactives. Ce langage est optionnel à la programmation de système web mais il est tout de même fortement conseillé pour des raisons de confort d'utilisation.

Ce sont donc ces quatre langages que j'ai utilisés tout long du développement avec en plus un cinquième langage de base de données, le SQL pour ce qui concernait les requêtes d'ajout, de suppression ou de mise à jour de la base de données.

#### 5.1.2 Cookies, Sessions et url longue

Le principal objectif dans le développement de formulaire est de faire transiter les informations d'une page à l'autre. Pour cela, il y a plusieurs méthodes.

Premièrement les variables cookie. Un cookie est un fichier temporaire qui s'enregistre sur la machine d'un utilisateur avec un numéro qui le caractérise. On peut y enregistrer différentes informations pour la récupérer à n'importe quel moment tant que le cookie est encore actif, car certains ont une durée de vie très courte. J'ai utilisé les cookies pour récupérer depuis l'intranet développé avec le CMS<sup>1</sup> « joomla » l'identifiant du magasin. Dans la page

---

<sup>1</sup> CMS : Système de gestion de contenu

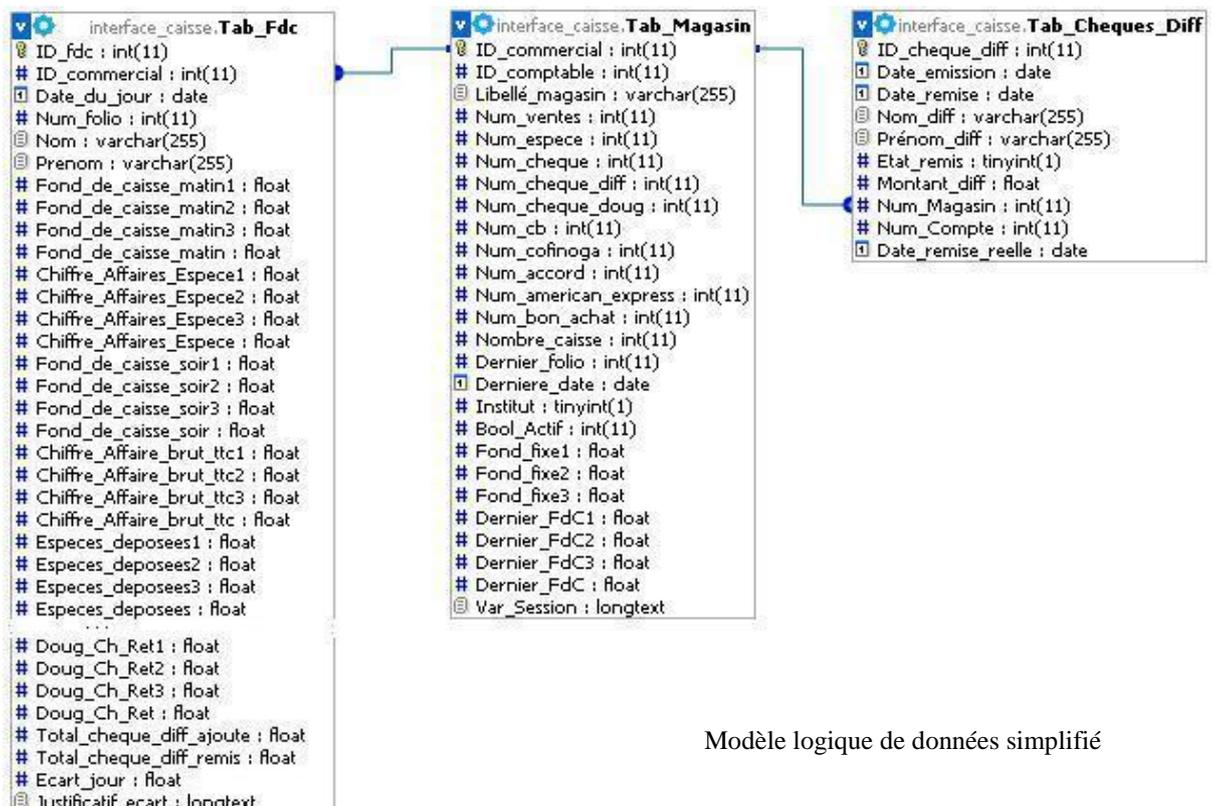
d'index de l'accueil de l'intranet, diverses fonctions vérifient constamment l'identification, c'est alors que j'y ai donc récupéré l'identifiant dans un cookie pour ensuite pouvoir le récupérer de n'importe où dans mon application.

L'url longue, est une méthode que j'ai peu utilisée car les variables sont transmises directement dans la barre d'adresse, et il est donc possible de les modifier. Je n'ai utilisé cette méthode que pour une seule raison. C'était pour la navigation dans l'application à partir de la feuille de synthèse. Pour mémoriser le passage par la feuille de synthèse, une petite variable du nom de « synt » prend la valeur « 1 » et donc à l'arriver sur une page, si cette variable a cette valeur, automatiquement la page réagit en fonction. Par exemple si de la page « synthèse », nous retournons sur la page du « détail de remises en banque », la variable étant à « 1 », la page affichera un bouton « retour synthèse » à la place des boutons « retour » et « valider ».

La dernière méthode, et celle que j'ai le plus utilisée, les variables de sessions. Ces variables là se comportent sous la forme d'un « array »<sup>2</sup> stockent chaque variable dans un tableau lié à une clé qui l'identifie. Les variables de sessions sont stockées ensuite dans un cookie sur la machine de l'utilisateur. Quand un formulaire quelconque de l'application est validé, les informations sont directement traitées

### 5.1.3 Structure de la base de données

La base de données a subi de multiples modifications en fonction des besoins énoncés au cours du stage et se retrouve finalement à 3 tables. « Tab\_magasin », « Tab\_FdC » et « Tab\_cheque\_différé » suivant un modèle logique de données assez simple au vue du nombre de tables.



<sup>2</sup> Array : Tableau de données

La table « Tab\_Cheques\_Diff » enregistre à chaque nouvelle insertion dans cette table un nouveau chèque. La clé primaire est un numéro d'identification donné automatiquement à l'ajout du chèque. C'est un numéro unique qui permet par la suite à l'application de différencier les chèques lors d'une remise en banque notamment. Les chèques possèdent tous un numéro de magasin associé et un état de remise en banque sous forme booléen. La table est nettoyée automatiquement, dès que la date de remise réelle dépasse un an et que l'état remis est bien à 1, le chèque est supprimé de la base. Ceci pour ne pas alourdir le système.

La table « Tab\_Magasin » est une table fixe, à moins d'un ajout ou suppression de magasin, il n'y aura aucune opération sur cette table mise à part de la lecture et la mise à jour de certains champs qui sont, « dernier\_folio », « Bool\_Actif », qui représentent la présence ou non d'un brouillon, les « Dernier\_FdC » et la « Var\_Session » qui est une variable texte où sont noté la totalité des variables sessions enregistrées dans la base de données.

La table « Tab\_FdC » est la plus imposante, elle était conçue à la base en plusieurs parties, mais comme leur clé primaire était identique, les tables ont fusionnées et ont donné la table feuille de caisse. Composée de soixante six champs, elle regroupe la totalité des saisies d'un magasin avec 3 caisses. Afin d'avoir le détail et aucun calcul à refaire, il a été décidé de sauvegarder le détail saisi et totaux dans cette table. Une journée de ventes correspond alors à une feuille de caisse et donc à une ligne de la table. Pour l'extraction des données, il suffira alors de choisir les champs voulu dans la table en fonction du magasin sélectionné et d'autres critères. De même le détail servira à afficher l'historique sur l'application.

L'extraction des données par la société SOFIA se fait via un logiciel « Report One » pouvant requêter différents types de bases de données comme ici « MySQL ».

## 5.2 Difficultés rencontrées

### 5.2.1 La modélisation de l'application

La principale difficulté que j'ai pu rencontrer au début du stage aura été la modélisation et la conception de l'application. Il m'aura fallu imaginer de quelle façon les magasins devaient saisir les informations, en respectant une idée principale de garder une interface simple et dans la continuité de ce que les magasins connaissaient. On s'est donc directement dirigé vers un développement similaire à l'existant qui servait de base ce qui créa une première application bancaire et rapidement abandonnée pour un changement vers ce qui donnera le système final.

L'autre chose à modéliser qui posa certains problèmes, est la structure de la base de données utilisée. J'avais commencé par créer un modèle de données lié à un début de spécification. Une spécification non gardée car les besoins changeaient rapidement et il fallait donc créer et adapter le code plutôt que d'adapter le modèle UML créé. Le premier modèle UML disposait sous forme d'objets l'application. On retrouvait donc un objet « système »,

« opération », « débit et crédit », « procès verbal », « Magasin », « Folio » et « Feuille de caisse ». On retrouve quelque peu cette structure grandement simplifiée dans la création de la base de données. De même des diagrammes de cas d'utilisation et d'activité ont été créés au début du développement, mais ne concordent plus totalement avec les besoins finaux et le système final. Ils ne sont donc pas intégrés à ce rapport.

### **5.2.2 Prévoir et s'adapter**

Au cours des dix semaines de développement, certains éléments validés du cahier des charges rédigé fin avril sur lequel devait s'appuyer l'élaboration du système devinrent désuet. En effet, au fur et à mesure que le développement avançait, des présentations étaient faites et de nouvelles idées surgissaient pour soit améliorer, soit changer quelques éléments. Il fallait donc être très réactif sur la modification du code pour adapter les nouveaux éléments à prendre en compte.

Ca a été le cas notamment pour la répartition du détail par caisse qui revoyait la structure principale de l'application. Mais aussi pour un changement début juin de l'abandon de la saisie de certaines données. Des conséquences plus ou moins lourdes sur le développement qui auraient permis la mise en place de certaines applications plus hâtivement et donc la mise en place d'ajout non traité dans l'application final, faute de temps.

La dernière grande modification dans le développement concerne le procès verbal. Jusqu'à début juin, le procès verbal sauvegardait de façon définitive les chèques différés dans la base de données, ne laissant ainsi peu de place aux erreurs. Même si une page de vérification lors de la remise demandait la validation de la liste de chèque, l'utilisateur n'était pas à l'abri d'une erreur. Il a fallu donc, lancer très rapidement une modification sur la structure même du traitement du procès verbal pour mettre à jour en même temps l'application sur l'intranet qui fonctionnait avec les magasins tests en même temps.

## 6. La Phase de tests et de validations

### 6.1 Tests et validation en interne

#### 6.1.1 Validation de l'application par rapport aux exigences demandées

L'application a été présentée à plusieurs reprises devant divers parties de la société.

Une présentation au Directeur Exécutif de la société Clin d'œil s'est déroulée fin mai, pour qu'il puisse donner son avis sur celle-ci et pour apporter un retour terrain. Si dans l'ensemble il était satisfait, il a été demandé de fixer le champ « fond de caisse du matin » en allant requêter la base de données sur le fond de caisse de la veille. Cette demande a eu pour conséquence l'ajout d'un champ « chiffre d'affaires espèces » qui m'a permis de calculer un fond de caisse théorique du soir qui se compare avec le fond de caisse saisie le soir et affiche un message s'il est différent. Cela pour éviter le report systématique d'une erreur de fond de caisse.

Une réunion du « Comité de direction » composé de la direction générale de clin d'œil, du Directeur Exécutif, des directrices régionales commerciales, d'un responsable achat, d'un contrôleur de gestion (SOFIA) et d'un responsable des ressources humaines (SOFIA). Cette réunion s'est déroulée fin Mai avec une présentation de l'application, la soumettant donc aux critiques sur le fond comme sur la forme que ce soit sur le design, la façon dont est découpé l'application ou encore sur les différents champs à remplir demandés. L'application a passé avec succès cette première validation qui semblait convenir aux membres présents durant la réunion.

Une nouvelle réunion début juin avec l'équipe trésorerie et comptable de la société s'est déroulée et était totalement centrée sur ce nouveau produit à disposition des magasins. Avec deux heures de discussions, il a été mis en évidence que certains champs, et plus précisément, la demande de saisie du montant de la TVA à 5.5%, et de la ventilation du chiffre d'affaires par type de ventes, soit de la marchandise soit de prestations de l'institut posaient problèmes. En effet, les magasins doivent reporter depuis un ticket qu'ils éditent ces différents taux, caisse par caisse. Une erreur dans le report de ses chiffres est impossible à déceler via l'application à moins de rajouter de nouveaux champs de saisie. La demande concernant l'application étant de ne pas ajouter une masse de travail supplémentaire aux magasins, il a donc été décidé de supprimer les champs qui posaient problèmes. Au niveau de l'application, cette décision de supprimer une partie de traitement importante puisque des calculs de TVA à 19.6%, de calculs de chiffres d'affaires net et brut, toutes taxes et hors taxes et même de calculs de TVA à 2.10% et à 0% étaient pris en compte. Autre conséquence de cette suppression, l'affichage automatique des champs relatifs à l'institut en fonction de sa présence ou non dans le magasin est devenu inutile, ce qui a fait un champ en moins dans la base de données.

## 6.1.2 Tests internes

Avant de faire tester l'application par les principaux intéressés, j'ai du réaliser une série de tests. Nous allons voir étape par étape les différents champs testés.

### 6.1.2.1 Le procès verbal

Les tests réalisés autour du procès verbal concernent surtout ce qui concerne l'ajout de nouveaux chèques différés en magasin. Afin de tester les structures de contrôle mises en place, dans un premier temps l'application vérifiait le format des dates saisies qui devait répondre au format jj/mm/aaaa. Puis pour plus de souplesse dans l'application, un calendrier a été ajouté permettant de cliquer sur la date, excluant ainsi toutes erreurs de format. Ensuite une autre structure permet de vérifier si les champs sont bien tous saisis. Que ce soit au niveau du nom, du prénom du montant ou des dates d'émission et de remise en banque du chèque, tous les champs doivent être renseignés avec en plus une vérification au niveau de la date, une date d'émission antérieure ou égale une date de remise en banque bloque le processus d'ajout. Il en est de même si la date d'émission du chèque dépasse la date de traitement de la caisse. Pour ce qui est du champ « montant », l'application vérifie si c'est bien un chiffre qui est saisi, dans le cas contraire, le processus est à nouveau bloqué jusqu'à une saisie correcte. Si le montant est un nombre à plus de deux chiffres après la virgule, il est automatiquement arrondi à la décimale près.

Sur la page du procès verbal, la remise de chèques en banque se déroule de la façon suivante. Le magasin sélectionne les chèques apparaissant en rouge sous le libellé « Remettre en banque » ou « Solder par un autre moyen de paiement » et appuie sur le bouton remettre en banque. Afin d'éviter des erreurs, que ce soit au niveau de l'enregistrement dans la base de données, ou dans le calcul de l'écart de caisse sur la feuille de synthèse, il y a fallu tester tous les scénarii possibles. Sur un formulaire comprenant 4 chèques tests en attente, différents tests ont été effectués. La remise en banque et l'acquittement du chèque par un autre moyen de paiement sur un même chèque décoche automatiquement les deux cases puisqu'un chèque ne pourra être réglé que d'une seule façon. De plus lors de la navigation dans l'interface, il a fallu vérifier que la liste de chèques remis en banque et son mode de remise était bien à jour, quelque soit le parcours emprunté dans la feuille de caisse pour arriver à la synthèse.

### 6.1.2.2 L'Identification du magasin

Dans un premier temps, l'identification se déroulait ainsi, le magasin allait sur l'intranet, entrait son identifiant, et allait sur la feuille de caisse. Il devait ensuite taper son nom, prénom, sélectionner la date du jour à traiter, toujours avec le calendrier mis en place pour le procès verbal et sélectionner dans un menu déroulant le numéro du magasin. Ainsi, sur la totalité des pages, une fonction permet de vérifier si le numéro du magasin est bien mémorisé. Dans le cas contraire un renvoie automatique à la page d'identification est exécuté perdant toutes les données saisies. Début juin, un rajout permettant de récupérer le numéro du magasin directement de la page d'authentification de l'intranet permet une saisie en moins et sécurise plus l'application empêchant

un magasin d'accéder à la feuille d'un autre magasin. Il a fallu alors rajouter une fonction qui renvoie à la page d'identification de l'intranet en cas de perte du numéro du magasin. .

### **6.1.2.3 La synthèse**

Derniers éléments de l'application web à tester, la feuille de synthèse. Sur cette feuille, était à tester la bonne transition des informations lors de retours vers les formulaires précédents qui doivent donc repasser dans les structures de calcul de l'application. De plus, lors d'un retour aux différents formulaires que sont, les ventes, le détail des remises, le fond de caisse ou le procès verbal, il fallait s'assurer qu'un bouton retour synthèse remplaçait bien le mode de navigation classique.

## **6.2 Choix des magasins tests et lancement du test à plus grande échelle**

### **6.2.1 Echantillon des magasins et critères de sélection**

Lors d'une réunion avec les différents responsables régionaux des magasins, une présentation de l'application a été faite pour démontrer l'utilité de celle-ci et sa continuité avec l'existant. Il a été sélectionné 12 magasins tests parmi les 50 magasins franchisés du groupe pour tester l'application. Le choix des magasins a été fait selon plusieurs critères. En premier lieu leur habilité à utiliser l'outil informatique. Si certains magasins ont un personnel très à l'aise avec cet outil, d'autres ont cependant beaucoup de mal à l'utiliser. Il y a donc parmi les douze magasins un échantillon assez représentatif de l'ensemble des magasins vis-à-vis de l'informatique. Ensuite, certains magasins testant d'autres systèmes, comme le système du fond de caisse fixe, ont été choisis pour tester en même temps la nouvelle feuille de caisse.

Les magasins choisis sont les magasins de Nantes Boileau, La Roche sur Yon, Cognac, Glacerie, Saumur, Pontivy, Challans, Biarritz, St Lo, Guérande, Avranches et Limoge. Cet échantillon a donc commencé les phases de test de l'application en temps réelle à partir de début juin.

### **6.2.2 Déroulement du lancement de l'application sur les magasins tests**

Afin d'obtenir un maximum de tests avec des chiffres réels, l'application a été lancé simultanément sur les 12 magasins test. Après un mail d'information, chacun des magasins a été appelé pour faire une présentation à distance de la nouvelle méthode de feuille de caisse qu'ils auront à utiliser tout le long de la période de test, et si la méthode fonctionne bien, en remplacement de leur feuille de caisse Excel. Le lancement de l'application sur les magasins tests s'est effectué sur environ dix jours, entre les rendez-vous assez tardifs, et les responsables à former, pas toujours présents. Les tests se sont réellement lancés à deux semaines de la fin du stage. J'ai alors appelé un à un les magasins, leur faisant une démonstration à distance via le logiciel pcAnywhere me permettant de voir sur l'écran de leur caisse mère. Récupérant la liste des chèques différés en attente dans chaque magasin, et la valeur de leur fond de caisse, j'avais ainsi une initialisation parfaite pour leur faire une démonstration en leur montrant toutes les possibilités au niveau du procès verbal, de la modification des données à partir de la feuille de synthèse et de la sauvegarde en brouillon en cas d'erreur.

## 6.3 Erreurs détectées via les phases de test et remonté d'information.

### 6.3.1 Erreurs de calculs de l'écart de caisse et approximation du résultat.

L'écart de caisse, calculé et expliqué dans le « 5. » n'avait jusqu'à la phase de test avec les magasins révélé aucune erreur. En effet, jusqu'à ce moment là, tous les chiffres testés dans l'application étaient soit repris sur les totaux d'une archive magasin, soit des chiffres arbitraires. Lors des phases de tests avec les magasins, une erreur est remontée systématiquement dans la base de données permettant de mettre à jour une inversion dans les flux de débit et de crédit. L'erreur a rapidement été corrigée et mise à jour sur l'application.

Une erreur révélée à plusieurs reprises, mais pas de façon systématique, est une erreur au niveau du calcul globale. Que ce soit la somme des chèques différés, l'écart de caisse ou juste le total d'un champ précis, il arrivait que certains magasins découvrent un écart de caisse de un centime inexplicable, voir même dans certains cas, des erreurs avec des chiffres aberrant allant avec vingt chiffres après la virgule. Il a été rapidement mis en évidence que le problème venait du format des valeurs des variables. Les chiffres subissant parfois une variation très faible créaient des erreurs de calcul. Pour y remédier, dans la totalité du code, chaque variable est traitée une par une avec un arrondi au centième d'euro près.

Les tests réalisés par les magasins par la suite ne remontaient plus cette erreur à partir de sa correction. Il avait été vu avec chaque magasin de noter dans le champ justificatifs, le montant de leur écart de caisse sur feuille Excel si celui-ci était différent sur l'application afin de pouvoir localiser au plus vite les erreurs de calcul.

### 6.3.2 Retours des magasins

Certaines responsables des magasins test ont pu exprimer leur avis sur l'application. L'avis général est plutôt positif, les responsables magasins qui n'étaient pas à l'aise avec l'informatique ont réussi aisément à s'adapter à leur nouvel outil. Les craintes émises au début du test sur le temps de saisie se sont estompées et certains responsables affirment même une saisie plus sûre et plus rapide. En effet, ils retrouvent plus rapidement une erreur sur le formulaire en ligne qui affiche bien tout le détail sur leur feuille Excel ou le calcul se fait parfois via la feuille elle-même, parfois via une calculatrice.

En plus de ces avis, quelques retours d'améliorations esthétiques et ergonomiques ont été relevés. Que ce soit pour la disposition globale des pages ou l'utilisation même de l'application, j'ai pu tenir compte de leurs réactions. Nous avons eu cependant des retours non exploitables sur le fond de l'application, comme la volonté de pouvoir modifier des informations de chèques en attente dans le procès verbal, requête refusé par SOFIA qui désire bien voir la différence entre la date de remise réelle du chèque et la date définis à son dépôt.

## 7. Cahier des charges

Norme : **AFNOR NF - X 50 151**

### 7.1. Présentation générale du problème

#### 7.1.1 Projet

##### *7.1.1.1 Finalités*

Une application web doit remplacer un existant au format Excel. Sa finalité étant de transmettre à l'entreprise SOFIA, des informations concernant chacun des magasins du groupe Clin d'Oeil S.A.S. Un formulaire s'intégrant à l'intranet de Clin d'Oeil regroupant deux fiches, la feuille de caisse sur lesquels apparaissent les recettes et les dépenses du magasin, ainsi que le solde. Et un procès verbal de caisse détaillant les chèques différés et la trésorerie.

##### *7.1.1.2 Espérance de retour sur investissement*

Grâce à cette application, l'entreprise espère gagner du temps pour gérer les erreurs de saisie qui ralentisse considérablement la comptabilité et la trésorerie. Avec ce système, il sera plus facile de retrouver une erreur de saisie dans l'application « Winparf » qui remonte les données de ventes et de paiement dans une base de données du groupe SOFIA.

#### 7.1.2 Contexte

##### *7.1.2.1 Situation du projet par rapport aux autres projets de l'entreprise*

Le projet devra proposer une alternative à une solution existante qui n'est plus conforme aux spécifications et aux attentes du client (fichier Excel). Actuellement, deux systèmes sont en place pour gérer la comptabilité, les feuilles de caisse remplies par chaque magasin et un fichier Excel automatique généré par les différentes caisses et envoyé directement à Sofia. Le projet devra donc remplacer les feuilles de caisse qui peuvent mettre jusqu'à une semaine à parvenir jusqu'à SOFIA. Il devra automatiser les calculs et proposer des options pour enrichir le système de base. Il devra proposer une interface qui facilitera la saisie des données et devra être utilisable par du personnel familiarisé ou pas avec l'environnement informatique et intranet. Le projet devra aussi être plus précis que l'existant en détaillant certaines entrées dans le formulaire.

##### *7.1.2.2 Études déjà effectuées*

Le calcul des différentes sommes a déjà été informé et automatisé dans le document Excel existant.

De même les calculs mettant à jour les éventuels écarts entre les différents justificatifs renseignant les champs de recettes et de dépenses ont été effectués et indique d'une couleur différente si un écart apparaît.

Un système de message informatif avait été mis en place sur la feuille Excel dans le procès verbal de caisse pour informer les chèques à remettre à la banque.

#### *7.1.2.3 Suites prévues*

Récupération dans la Base de Données des différentes données de chaque magasin tel que le chiffre des ventes, les espèces remises à la Banque, le montant des chèques reçus, le montant des paiements par Carte Bancaire, par Carte Cofinoga, par American Express, ou par Bons d'achat. Récupérer aussi les totaux, le solde du procès verbal et s'il existe un écart du jour avec justificatif. Ces informations seront traitées au niveau de la comptabilité de SOFIA via un document « journal de caisse ».

#### *7.1.2.4 Nature des prestations demandées*

Développement d'une application web s'intégrant à l'Intranet Clin d'œil et de son interface. L'application doit améliorer un système manuel existant (Excel).

Concernant le formulaire Feuille de caisse, il devra être enrichi par rapport à la version de base, un descriptif plus détaillé des ventes est demandé et une saisie par caisse plutôt que par globalité permettrait d'avoir toutes les informations nécessaires à la comptabilité de SOFIA.

L'authentification d'un magasin doit automatiquement mettre à jour le formulaire pour ce magasin même. C'est-à-dire reporter les informations des magasins, proposer le formulaire type de celui-ci avec l'apparition ou non d'informations à remplir tel que « Institut » pour les ventes et le nombre de caisse à traiter. L'authentification de la caisse mettra à jour aussi automatiquement la liste des chèques différés.

#### *7.1.2.5 Parties concernées par le déroulement du projet et ses résultats (demandeurs, utilisateurs)*

Le demandeur est la S.A. SOFIA par l'intermédiaire d'Eric Moureaux (au service contrôle de gestion) et de Sébastien Burgaud (au service Informatique).

Les utilisateurs seront les collaborateurs des magasins sous la direction de la S.A. SOFIA, et plus exactement pour la S.A.S Clin d'œil, filiale parfumerie de l'entreprise.

### **7.1.3 Énoncé du besoin (finalités du produit pour le futur utilisateur tel que prévu par le demandeur)**

Le besoin de l'entreprise est de remplacer un existant par un développement interne de l'application intranet. L'application permettra aux caisses des différents magasins du groupe Clin d'Oeil de remplir des informations concernant les ventes de la journée dans un formulaire qui alimentera une base de données avec les informations recueillies. Le but étant de gagner du temps entre la saisie manuelle d'une feuille de caisse et d'un procès verbal sous Excel le soir ou lendemain matin de la vente afin que la société SOFIA puisse traiter le plus rapidement possible les données et traiter en cas de problème les erreurs de saisie le plus tôt possible.

Il doit être possible de modifier dans la base de données (via une interface) les valeurs liées à chaque magasin comme des numéros de comptes liés à une banque ou l'ajout de nouveaux libellés. Des actions possibles uniquement pour un administrateur.

L'application ne doit pas fournir plus de travail au magasin que le système actuellement en place.

#### 7.1.4 Environnement du produit recherché

##### *7.1.4.1 Listes exhaustives des éléments (personnes, équipements, matières...) et contraintes (environnement)*

Serveur incorporant l'intranet déjà en place

L'utilisateur qui utilisera l'application : le collaborateur des différents magasins du groupe franchisé Douglas de Clin d'Oeil, ainsi que l'administrateur de l'Intranet pour mettre à jour manuellement des bases de données, sachant que l'utilisateur pour un même magasin n'est pas la même personne tous les jours à utiliser l'application.

##### *7.1.4.2 Caractéristiques pour chaque élément de l'environnement*

Serveur : Apache 2 exécuté dans une virtualbox <sup>3</sup> Ubuntu <sup>4</sup> d'environnement KDE.

Intranet développé sous un environnement « Joomla <sup>5</sup> », outil de développement et de gestion de contenu web. Il y a donc une base de données gérer via phpMyAdmin <sup>6</sup> en MySQL <sup>7</sup> à réutiliser.

L'utilisateur peut être un novice en informatique, l'application devra donc être adaptée.

## 7.2. Expression fonctionnelle du besoin

### 7.2.1 Fonctions de service et de contrainte

#### *7.2.1.1 Fonctions de service principales (qui sont la raison d'être du produit)*

Etablir pour chaque magasin, une feuille de caisse sous la forme d'un formulaire et un procès verbal de caisse relatif à l'ID du magasin.

Calculer à partir des champs du formulaire des feuilles de caisse et procès verbaux de caisse pour mettre en avant des écarts de caisse ou des totaux de vente.

Faire un récapitulatif global en prenant les valeurs des formulaires de chaque caisse.

---

<sup>3</sup> VirtualBox : Logiciel permettant de simuler dans l'environnement Windows un système d'Exploitation, ici Linux

<sup>4</sup> Ubuntu : Distribution GNU/Linux

<sup>5</sup> Joomla! : est un système de gestion de contenu

<sup>6</sup> phpMyAdmin : est une interface de gestion pour système de gestion de base de données MySQL

<sup>7</sup> MySQL : Système de gestion de base de données

Sur les procès verbaux de caisse, prévoir une fonction qui reporte chaque chèque différé tant qu'ils n'ont pas été remis à la banque et indique les chèques à remettre lors du dépassement de date.

Sur les procès verbaux de caisse, donner la possibilité à la caisse de remettre un chèque à la banque et donc de changer le statut du chèque en « remis » ce qui l'enlèvera de la liste des chèques différés et reportera automatique son montant sur la feuille de caisse.

Pour valider le procès verbal, intégrer au formulaire une liste de champs obligatoires comme la date d'émission et de remise. Sans cela, l'application ne validera pas le procès verbal et ne passera pas aux formulaires suivants.

#### *7.2.1.2 Fonctions de service complémentaires (qui améliorent, facilitent ou complètent le service rendu)*

A chaque remise (chèque, espèce et carte bancaire) prévoir un champ pour remplir les numéros de remises. Cela permet d'améliorer la recherche d'erreur et la mise en parallèle avec les documents fournis par les banques.

Une page d'aide à la recherche pour les éventuels écarts de caisse pourrait être proposée en cas d'erreur. La page type indiquerait les différents justificatifs à vérifier et les numéros à prendre en compte, ainsi qu'une aide pour remplir les champs du formulaire à la façon d'un manuel d'utilisateur. Prévoir aussi des bulles d'aides avec un « ? ».

#### *7.2.1.3 Contraintes (limitations à la liberté du concepteur-réalisateur)*

Temps : 12 avril au 18 juin 2010

Application s'intégrant à l'Intranet développé via « Joomla ». Garder une interface qui permette de revenir à tout moment sur celui-ci.

Réaliser une application simple à prendre en main par n'importe quel utilisateur, les pages devront proposer toutes les informations demandées mais en afficher le moins possible pour ne pas que l'utilisateur se perde dans l'information.

Ne pas imposer une surcharge de travail supplémentaire à l'utilisateur.

### 7.2.2 Niveaux des critères d'appréciation et ce qui les caractérise

#### *7.2.2.1 Niveaux dont l'obtention est imposée*

A la date du 18 juin, l'application web devra effectuer les tâches principales misent en 2.1.1, intégrée à l'Intranet de l'entreprise et à son système d'authentification.

#### *7.2.2.2 Niveaux souhaités mais révisables*

Il serait souhaitable que l'application soit terminée pour la semaine 8 du stage (début juin) afin de pouvoir procéder à l'élaboration de tests, d'un manuel d'utilisateur et de pouvoir peaufiner en ajoutant les fonctions secondaires développées en 6.2.1.2.

## 7.3. Cadre de réponse

### 7.3.1 Pour chaque fonction, solution proposée

Pour la réalisation de l'application, le langage utilisé sera le PHP<sup>8</sup> pour la partie traitement dynamique, le MySQL pour la base de données et le HTML<sup>9</sup> et CSS<sup>10</sup> pour la mise en forme.

Les tableaux Excel seront mis en forme grâce à des formulaires qui enregistreront leurs données directement dans une base de données.

La spécification sera réalisée grâce aux outils proposés par le site « win-design.com »

Les chèques différés seront gérés par une variable booléenne à leur création. Si le chèque n'a pas été remis à la banque, la variable sera dans un état « non remis » et si le chèque est remis, la variable sera dans l'état « remis ». Grâce à cela, on pourra afficher et traiter que les chèques dont la variable booléenne sera sur « non remis » via une requête simple de la base de donnée des chèques différés.

De même, afin de remettre un chèque à la banque, une solution via un bouton ou autre widget<sup>11</sup> permettra de modifier la variable et donc de supprimer à l'affichage les chèques remis.

### 7.3.2 Pour l'ensemble du produit

#### 7.3.2.1 Prix de la réalisation de la version de base

Application non soumis à la vente, donc pas de prix associé.

Charges humaines et matériels : 1 développeur avec un ordinateur a disposition et un serveur web de l'Intranet. Ajouter à cela des intervenants (clients)

#### 7.3.2.2 Mesures prises pour respecter les contraintes et leurs conséquences économiques

Développement avec base MySQL pour réutiliser les bases de données fournis par l'Intranet (notamment l'authentification des caisses des magasins.)

#### 7.3.2.3 Outils d'installation, de maintenance ... à prévoir

L'installation se fera par une copie sur le serveur des fichiers et seront donc directement interprétés.

Pour la maintenance, prévoir une modification manuelle de la base de données (via PHP MyAdmin) si aucune interface dans l'intranet n'est développée.

#### 7.3.2.4 Prévisions de fiabilité

L'application entre dans un ensemble déjà développé, les problèmes de fiabilité qu'elle pourrait rencontrer pourraient être liés aux problèmes de

---

<sup>8</sup> PHP : c'est un langage de scripts libre principalement utilisé pour produire des pages Web dynamiques

<sup>9</sup> HTML : Hypertext Markup Language est le format de données conçu pour représenter les pages web.

<sup>10</sup> CSS : Cascading Style Sheets : feuilles de style en cascade, c'est un langage qui sert à décrire la présentation des documents HTML

<sup>11</sup> Widget : contraction de Window et Gadget, c'est le nom d'un composant d'interface graphique, un élément de base d'une interface graphique (bouton, ascenseur, liste déroulante, etc.)

fiabilité de l’Intranet lui-même. Il faudra tout de même gérer au niveau de la sécurité des données que les magasins pourront modifier avec la base de donnée comme une vérification qu’aucun code malicieux ne soit entré dans le formulaire et que le format des données soit bien respecté.

L’application devra aussi être utilisable avec différents navigateurs dont Internet Explorer 6 qui n’est pas à jour sur les normes w3c<sup>12</sup>.

A vérifier tout de même dans la procédure de test à la fin du cycle de développement. Tests à mettre en place :

-Test HTML

-Validation de navigateur en cas de changement de navigateur.

#### *7.3.2.5 Perspectives d’évolution technologique*

L’application utilisera les langages courants pour développer un site dynamique.

Cependant des ajouts ou des évolutions des langages pourraient amener à refaire ou à modifier le code.

---

<sup>12</sup> w3c ou World Wide Web Consortium : est un organisme de standardisation chargé de promouvoir la compatibilité des technologies du World Wide Web.

## Synthèse du stage

Après une période de quelques jours d'essai, on a pu constater que l'application en elle-même fonctionnait bien grâce aux retours de chiffres vérifiés chaque jour. Elle répond aux principales exigences fixées au début du stage, et même à d'autres demandes ajoutées au fur et à mesure. Cependant, certains points auraient pu être améliorés. La partie de l'application pour la partie de l'administration, c'est-à-dire la modification des tables et des valeurs inscrites dans celles-ci sans avoir à passer par PHPMyAdmin qui propose une interface un peu trop complète et complexe. L'application pourrait être plus facile d'utilisation pour l'entretien des bases de données. Cela ne pose pas de problèmes pour le fonctionnement global de l'application, mais il faudra quelqu'un connaissant l'outil PHPMyAdmin pour pouvoir entretenir l'application en cas de problème. De même la gestion des chèques différés dans le procès verbal ayant été revu à la fin du stage, toutes les idées n'ont pas pu être intégrées.

Au final, ce stage a été très intéressant et instructif sur un certain nombre de points. Premièrement, j'ai eu la chance de pouvoir accomplir la tâche de réaliser une application de son commencement à sa mise en ligne, comportant les phases de créations, de développements, de tests et de formations de l'utilisateur. J'ai pu voir que la communication était très importante pour que l'application puisse répondre le plus fidèlement possible aux besoins qui eux même peuvent être amenés à changer au fil du temps. C'est ce qui est arrivé au cours du stage avec une suppression de plusieurs jours de travail due au remaniement des informations demandées. Ce stage m'a aussi permis de mettre en application mes connaissances accumulées durant mes deux années à l'IUT, que ce soit en technologie web avec les langages HTML, CSS, PHP et JavaScript, ou de base de données via la base MySQL, ou encore de Gestion avec la manipulation de numéro et données comptables. J'ai pu affiner mes connaissances dans ces secteurs et voir même apprendre de nouvelles



# Annexes Rapport de Stage

Développement d'une application web Intranet

Quentin GAUTIER



Responsable de stage : Gilles NACHOUKI  
Maitre de Stage : Sébastien BURGAUD

2010

## Sommaire

Journal de Bord .....	36
Semaine 1 .....	36
Semaine 2 .....	37
Semaine 3 .....	38
Semaine 4 .....	40
Semaine 5 .....	41
Semaine 6 .....	42
Semaine 7 .....	43
Semaine 8 .....	43
Semaine 9 .....	44
Semaine 10 .....	45
Modèle logique de données.....	46
Exemple existant .....	47
Capture d'écran de quelques pages de l'application .....	49
Webographie .....	51
Bibliographie.....	51
Code source .....	52
Manuel d'utilisateur .....	56

## Journal de Bord

### Semaine 1

*12 avril 2010*

Pour ne pas faire mauvaise impression le premier jour, je suis arrivé plus tôt, et la journée a débuté par une présentation de l'entreprise et du personnel de SOFIA. Ensuite a commencé une description des besoins, Eric Moureaux m'a apporté un peu en vrac les différentes informations que j'avais besoin pour réaliser ce qu'ils ont besoin. J'ai à disposition le document Excel qui doit être remplacé, et des documents utilisés ensuite par Eric (au service contrôle de gestion) qui fait la synthèse des feuilles envoyé par les magasins. Ma tâche consiste donc à automatiser les feuilles partant des magasins en créant un formulaire sur l'intranet qui alimenterait une base de données qu'il pourra réutiliser par la suite. J'ai commencé à apporter des solutions théoriques d'une structure que je pourrais mettre en place. Début de rédaction brouillon de l'analyse des besoins.

Premier constat de l'ambiance qui règne dans l'entreprise, c'est détendu, les employés se tutoies et parle librement, très bonne ambiance générale.

*13 avril 2010*

Rédaction du cahier des charges. Apport de solutions théoriques pour gérer les procès verbaux. Mise en place d'un schéma de traitement des pages du formulaire. Rapide apprentissage du fonctionnement de Joomla, logiciel de gestion de contenu permettant la création de site par morceau de code préconstruit. Sébastien Burgaud (maitre de stage) a utilisé ce logiciel pour mettre au point le site intranet Clin d'Oeil et mon application doit donc s'intégrer à elle. L'accès à l'intranet se fait via l'IP du serveur : 192.168.1.20. Le pôle administration se joint en 192.168.1.20/administrator

Difficultés rencontrées : Difficile de ne pas se lancer directement dans le développement pour tester des solutions qui viennent à l'esprit. Il n'est pas évident de faire une analyse des besoins très précise.

*14 avril 2010*

Sites consultés ce jour :

<http://www.joomla.fr/>

<http://www.phpsources.org/scripts67-PHP.htm>

[http://openweb.eu.org/articles/css\\_impression/](http://openweb.eu.org/articles/css_impression/)

<http://www.win-design.com/fr/index.htm>

Découverte du logiciel win-design pour la spécification de l'application. Rédaction du cahier des charges. Brouillons de diagrammes d'activité. Mise en place de problématiques concernant la gestion des dates des documents.

Mise au point avec Eric Moureaux (personne connaissant les besoins de l'entreprise et du groupe Clin d'Oeil remplaçant Sébastien Burgaud en son absence) pour valider les besoins de l'utilisateur. Spécification concernant les bons d'achats avec une liste exhaustive à mettre au point. Modifier par rapport à l'existant du libellé ventes qui doit être plus précis avec

apparition du détail avec « parfumerie » « institue » et les différents taux de TVA. A faire apparaitre donc dans le formulaire final (19.6%, 5.5%, 2.1% et 0%). Voir aussi la possibilité d'afficher automatiquement le formulaire spécifique au magasin par rapport au compte d'identification. Penser aussi à gérer dans la BDD des booléens qui afficheront ou non les chèques différés dans les procès verbaux et les bon d'achats sur la feuille de caisse.

*15 avril 2010*

Commencement de la journée par la rédaction du cahier des charges avec une analyse plus fine des besoins. Certains besoins pas obligatoires mais pratiques ont été évoqués notamment l'ajout d'un champ pour la futur base de données avec les numéros de remise pour chacune d'entre elle.

SOFIA n'étant pas une boîte d'informatique, et que seul mon maitre de stage a le statut d'informaticien, aucun logiciel de développement n'est disponible sur les machines. J'ai donc procédé à l'installation de divers outils et demandé un accès à la base de donnée de PHP MyAdmin de Joomla pour avoir une vue d'ensemble de la base existant et émettre des solutions théoriques aux problèmes posés.

*16 avril 2010*

Matinée passée à peaufiner le cahier des charges et à compléter les oublies. Vérifier s'il correspond bien aux attentes de la société.

Représenter un diagramme d'activité permet de bien visualiser le fonctionnement de l'application, j'ai donc commencé à voir comment on les réalise et commencé à en griffonner un. La visualisation est directe, on peut voir comment fonctionnera le site d'un coup d'œil. Et cela permet de voir comment coder ensuite l'application.

Rédaction de certaine partie du rapport de stage comme la présentation de l'entreprise plus poussée et de l'Introduction.

## **Semaine 2**

*19 avril 2010*

Sites consultés ce jour :

<http://g-rossolini.developpez.com/tutoriels/php/formulaires/>

<http://fr.php.net/manual/fr/language.oop5.basic.php>

<http://www.vulgarisation-informatique.com/php-poo.php>

[http://www.editeurjavascript.com/scripts/scripts\\_formulaires\\_3\\_595.php](http://www.editeurjavascript.com/scripts/scripts_formulaires_3_595.php)

JavaScript

1er problème technique, comment automatiser le formulaire pour afficher les bons éléments dans un tableau.

Rédaction d'un premier algorithme répondant à ce problème.

Mise en avant de solutions techniques comme l'utilisation de variable session pour conserver les champs remplis par l'utilisateur. Prévoir la sécurité par la suite de l'application.

*20 avril 2010*

Sites consultés ce jour :

<http://g-rossolini.developpez.com/tutoriels/php/formulaires/>

<http://fr.php.net/manual/fr/language.oop5.basic.php>

J'ai pu mettre en place quelques scripts Javascript sur des pages que je sais fixe et finale afin de pouvoir montrer à Eric ce qu'il est possible de faire au niveau de l'application pour aider l'utilisateur au niveau de la saisie et l'empêcher de mettre de mauvaises informations.

*21 avril 2010*

Sites consultés ce jour :

<http://g-rossolini.developpez.com/tutoriels/php/formulaires/>

<http://fr.php.net/manual/fr/language.oop5.basic.php>

Les informations de chaque formulaire soumis sont intégralement conservées sur le serveur et je les réutiliserais chaque fois que l'utilisateur en a besoin. Le plus simple est d'utiliser des sessions, mais le plus flexible est bien sûr d'utiliser une base de données. Quoi qu'il en soit, la session est nécessaire pour passer les informations d'un formulaire à l'autre.

*22 avril 2010*

J'ai débuté à construire un modèle de base de donnée et j'ai soumis mon cahier des charges à Eric pour vérifier si notre vision du projet était la même.

*23 avril 2010*

Modification du cahier des charges en prenant compte de nouveaux éléments. J'ai passé la journée ou presque à voir avec Eric comment doit être l'application pour que ça arrange tout le monde que ce soit la partie « Compta » de SOFIA, que les magasins ainsi que la Trésorerie. On est donc partie sur une application plus précise que l'existant, afin d'avoir le maximum d'informations dans la base de donnée et empêcher les erreurs de calcul par les caisses.

### **Semaine 3**

*26 avril 2010*

Mise en place de la VirtualBox pour faire tourner Linux. Il a fallu relancer Apache et les différents services sur une IP différentes pour pouvoir travailler.

Réunion avec la responsable informatique de SOFIA pour lui présenter la future application.

Un mot d'ordre, ne pas ajouter de travail aux magasins. Il faut donc repenser ce qu'on avait vu avec Eric concernant les bons d'achats, revoir l'application pour une feuille de caisse par magasin et non par caisse.

*27 avril 2010*

Sites consultés ce jour :

<http://www.toutjavascript.com/main/index.php3>

→ Envoyer à [gilles.nachouki@univ-nantes.fr](mailto:gilles.nachouki@univ-nantes.fr) le cahier des charges - FAIT -

On a revu la façon de gérer les fonds de caisse, on a avec Eric passé en revue plusieurs façons de traité le problème afin d'avoir une continuité comptable du fond de caisse, ce qui pose beaucoup de problèmes.

J'ai pu utiliser un peu ma version de l'Intranet pour faire des modifications dessus pour voir le fonctionnement de Joomla!.

*28 avril 2010*

Code de la base de données MySQL sous PHPMyAdmin, et de l'intégration dans le code PHP. -> PB rencontrés : problèmes de dates à régler grâce à un calendrier à cliquer.

*29 avril 2010*

Sites consultés ce jour :

<http://www.javascriptfr.com/dlzip.zipnix?ID=34725&accept=1> -> Calendrier java

Après récupération d'un code javascript de calendrier, j'ai incorporé ce code dans le mien pour faire apparaître un calendrier aux endroits nécessaires, comme à l'identification ou lors d'un procès verbal de cheque différé.

J'étudie maintenant comment entrer ces dates formaté dans la base de donnée.

J'essaye aussi la mise en place de l'update automatique de l'état de remise d'un chèque.

*30 avril 2010*

Sites consultés ce jour :

<http://www.phpfrance.com/>

<http://www.phpsources.org/>

J'ai été bloqué un petit moment sur une fonction d'update de la base de données pour une erreur de syntaxe visiblement mais qui n'avait pas l'air de gêner le serveur l'hors de la demande d'exécution de la page. Je faisais afficher les variables sélectionnés par checkbox avant et après l'Update de la BDD et ça s'affichait correctement mais ne modifiait pas la BDD. J'ai donc essayé en remplaçant manuellement la valeur de la variable pour vérifier la syntaxe de ma fonction et cela fonctionnait. Le problème venait donc au passage de paramètre. Problème réglé en utilisant une formulation préparé de la fonction. J'ai ensuite construit les fonctions qui changent les dates au format jj/mm/aaaa au format mm-jj-aaaa pour l'intégrer à la BDD MySQL. La fonction inverse existe aussi.

J'ai ensuite commencé à mettre en place la table Tab\_Magasin renseignant les champs des numéros de compte etc. On a décidé avec Eric de rendre l'iD plus simple en demandant de sélectionner le magasin avec libellé lors de l'iD.

## Semaine 4

*03 mai 2010*

J'ai changé la structure de vérification des identifiants. J'ai fait l'erreur de vérifier sur la page même du procès verbal alors qu'il fallait que je le fasse entre la page d'identification et la page du procès verbal. J'ai donc tout condensé sur une page php qui ne fonctionne pas encore très bien, il faut que je pousse un peu plus les tests.

D'autre part j'ai pu mettre en place une page récapitulative des chèques sélectionnés par la caisse qui si elle est validée modifiera la base de données et sinon retournera au procès verbal. Il faudrait aussi que je mette dans une autre page la partie qui risque de revenir souvent sur l'affichage des noms prénoms, numéros magasin etc.

J'ai aussi mis dans une autre page php le code récurrent de la présentation de la personne et du magasin.

J'ai commencé la page des feuilles de caisse.

*04 mai 2010*

J'ai regroupé dans une même page php mes différentes fonctions que j'appellerais aux différents endroits. J'ai créé la fonction qui permettra de voir si les identifiants ont été perdu par le système de retourner la page à l'identification pour ne pas avoir d'erreurs dans la BDD en enregistrant n'importe quoi dans un champ vide.

De plus j'ai analysé le nouveau procédé pour avoir le moins d'entrée possible au niveau du formulaire de ventes pour alléger le travail des caisses et diminuer le risque d'erreurs. Il faut prendre maintenant en compte le produit brut, net, ttc et la répartition TVA 19.6 et 5.5.

*05 mai 2010*

Eric Moureaux (contrôleur de Gestion) m'annonce que le 27 Mai je dois présenter devant le comité mon application en précisant bien l'objectif de mon application, qu'elle est faite pour gagner du temps et pour ne pas donner une charge de travail supplémentaire aux différents magasins. Il faudra aussi quelques diapo' pour présenter suivis d'une présentation de l'application. Présentation de 30 minutes avec la démonstration plus des questions éventuelles.

Problème au niveau du passage des paramètres. A voir pour les régler. (Régulé en utilisant la BDD)

J'ai ensuite fini toute la structure squelettique html du site. J'ai aussi créé toutes les variables sessions nécessaire, et il y en a un paquet. Je ne sais pas s'il y a une meilleure solution.

*06 mai 2010*

Finissant la structure du site, j'ai mis en place toutes les variables de sessions et j'ai commencé à développer une structure pour les traiter dans une boucle pour que ce soit plus propre. Pour cela j'essaye d'utiliser un tableau de tableau mais l'appel dans le tableau ne fonctionne pas...

Problème résolu en découpant les tableaux en sous tableaux et en faisant de multiples tests. Voilà une soixantaine de lignes de code récursif d'économisé !

On a commencé avec Eric a posé sur papier la partie calcul par taux de TVA, et pour les remises à amputer sur les montants brut pour avoir en net selon le poids de la partie institut / parfumerie.

*07 mai 2010*

Rédaction du rapport le matin et reprise des calculs en rapport avec la TVA et les remises, pour gérer les chiffres après la virgule, pour les arrondis pour ne pas avoir de chiffres faussés.

<http://fr.selfhtml.org/html/tableaux/construction.htm>

Fonction pour créer mes tableaux en fonction du nombre de caisse.

## Semaine 5

*10 mai 2010*

J'ai mis au point les différentes fonctions qui me permettent de créé une ligne supplémentaire dans mes tableaux en fonction du nombre de caisse. Cela m'économise des lignes de codes assez lourdes. D'autre part, suivant l'actualité de la crise, le changement du taux de TVA en Grèce fait questionner Eric vis-à-vis de la gestion de la TVA dans l'application. Il n'était de toute évidence pas prévue de l'inscrire en dur dans l'application, mais on rajoutera quand même les taux 2.1 et 0 dans les calculs si un jour ils deviennent utiles pour l'application. Conséquence, à rajouter dans une table TVA, les taux, et une variable booléenne qui permet d'activer ou non le taux dans l'application.

J'ai commenté et ordonné un peu mon code pour plus de lisibilité vu que je commence à en accumuler.

*11 mai 2010*

Mise en place de la fonction « brouillon » qui sauvegarde les variables de sessions. La fonction permettra de sauvegarder les variables dans la base de données de chacun des champs remplis par l'utilisateur afin de pouvoir récupérer son travail la fois d'après. J'ai donc mis un bouton « brouillon » sur la feuille de synthèse qui redirige vers l'accueil en sauvegardant dans la BDD. Une variable booléenne est mise en place. A 0 le brouillon n'est pas actif, à 1 il l'est. Quand on sauvegarde le brouillon, la variable passe soit de 1 à 1 dans le cas d'un nouvel enregistrement, soit de 0 à 1. Si la variable est à 1, à l'identification du magasin le brouillon est aussitôt récupéré sans préavis et l'utilisateur est à nouveau à la synthèse. La validation de la synthèse passe la variable brouillon à 0. J'ai mis en place un type de sauvegarde qui me paraît lourd ... je cherche donc une solution pour remplacer un tableau dans la BDD de 60 par 50 soit 3 000 cases à gérer.

*12 mai 2010*

<http://fr.php.net/manual/fr/function.serialize.php>

J'ai réussi à finaliser le brouillon, j'ai pu découvrir comment était sauvegardé les variables \$\_SESSION, et c'est un simple array. J'ai donc voulu l'enregistrer dans directement dans la base de données. Pour cela j'ai eu quelques soucis car la BDD n'acceptait pas mon tableau, il a fallu donc que je cherche un formatage pour l'enregistrer et le relire après. Ce que j'ai trouvé après des recherches.

J'ai donc obtenu une chaîne de caractère que je mets à jour dans la BDD dans un champ par magasin soit 50 champs.

*13 mai 2010*

Férié

## **Semaine 6**

*17 mai 2010*

Amélioration de la fonction pour créer automatiquement le formulaire avec les différents champs afin d'afficher la variable de session si elle existe.

Afficher ou pas le total ? Information utile ? Non

Présentation à Magali, après quelques tests le matin sur l'application, celle-ci ne fonctionnait plus tout à fait pour la présentation. Quelques fautes d'orthographe et des libellés pas très parlant, mais dans l'ensemble c'était satisfaisant. A corriger donc pour la présentation du mercredi.

Début de mise en place de la structure pour effacer la BDD au fur et à mesure que les chèques différés sont ajoutés. Les chèques différés ne seront donc pas conservés plus d'un an.

*18 mai 2010*

Grosse optimisation du code en supprimant dans les constructions de formulaire les différents champs pour 1 et 2 caisses, maintenant géré après avec le numéro de session dans la variable nombre caisse de ma fonction

*19 mai 2010*

→ Présentation à Jean Charles (président exécutif Clin d'œil) de l'application à son état d'avancement.

Ce qu'il en est ressorti, l'application aidera bien la société, cependant il a butté sur le fond de caisse à saisir le matin. Après discussion avec Magali et Eric qui avait deux idées opposées sur la question, il a été dit que le fond de caisse du matin serait repris automatiquement de la veille. Et qu'un fond de caisse théorique serait calculé à partir du chiffre d'affaire net en espèce et de la remise en banque des espèces. Théoriquement il est coïncide, sinon, il y a erreur de caisse à faire apparaître.

J'ai réparé le script JavaScript qui ne fonctionnait plus pour l'ajout de chèques différés, j'ai mis en place celui de l'authentification. On s'est aussi posé la question comment valider les données.

*20 mai 2010*

Un bug énorme est apparu dans un test de calcul. Une erreur affichant « 158.28999999991254638 » est apparue dans mon calcul d'écart. Une erreur due au format

des variables utilisées dans les calculs. J'ai essayé plusieurs méthodes pour essayer de corriger cette faille assez importante de l'application.

*21 mai 2010*

Bug de la veille encore actif, j'ai donc essayé la mise en place d'une solution multipliant dans les calculs les variables par 100, les castant en « Integer » et redivisant par 100 par la suite. L'objectif étant de supprimer ces chiffres qui apparaissent derrière les centimes qui faussent l'application. En effet ça pourrait créer un petit écart de caisse provoquant une erreur injustifiée. Pire les chiffres sauvegardé et remontant dans la base de données pourraient être erronés. Pour pallier à ce problème, j'ai mis en place une structure entourant l'application d'affichage des variables et des calculs à différents stade de l'application.

## **Semaine 7**

*24 mai 2010*

*Lundi pentecôte, férié.*

*25 mai 2010*

La présentation « CODIR » approchant, je devais rapidement mettre en place quelque chose de plus propre visuellement. La mise en page de l'application fut donc revue pour présenter une application propre visuellement. J'ai aussi mis en place un système de contrôle JS sur la page de synthèse.

*26 mai 2010*

J'ai continué à préparer l'application et à la tester pour ne pas avoir de surprise le lendemain avec la présentation. Améliorations visuels donc et tests en tout genre pour cette journée.

*27 mai 2010*

→ Présentation au comité de mon application, à préparer quelque chose.

Résultat de la présentation du 27 mai devant les responsables régionaux des magasins, du président exécutif Clin d'œil et d'un contrôleur de gestion :

L'application a plu, il n'y a pas eu beaucoup de question sur l'application en elle-même, elle répond à des besoins. Une liste des magasins qui devront la tester cet été à été faites regroupant magasins à l'aise avec l'informatique et moins à l'aise.

*28 mai 2010*

Le problème des décimales flottantes persiste malgré la solution de traitement mise en place. J'ai donc décidé de revoir la façon dont étaient faits mes calculs.

## **Semaine 8**

*31 mai 2010*

Les phases de tests ayant été fixées à l'été, Magali (contrôleur de gestion) a émit l'idée de lancer les tests tant que j'étais encore présent afin de me donner la possibilité de pouvoir suivre le processus de développement du début à la fin.

Côté programmation, avancement sur la remontée des informations dans la base de données.

### *01 juin 2010*

Un mail a été envoyé à chaque magasin test : Boileau, La Roche/Yon, Cognac, Glacerie, Saumur, Pontivy, Challans, Biarritz, St Lo, Guérande, Avranches, Limoges. Chacun de ses magasins a été prévenu de la nouvelle procédure qui allait se mettre en place, et que j'allais les appeler chacun pour leur expliquer le fonctionnement de l'application

### *02 juin 2010*

#### > Appel des premiers magasins pour tests

Avec une liste des noms des responsables magasins tests, j'ai commencé à lancé la phase de test. J'ai donc appelé pour la journée d'aujourd'hui 5 magasins leur présentant en direct via pcAnywhere l'application et recueillant leurs questions. Si aucune crainte n'est à déclarer concernant l'application elle-même. Les magasins ont peur de l'ajout de travail du à la phase de test qui demande une saisie via les feuilles Excel (l'existant) et via l'intranet. Lancement prévu pour 2 magasins demain soir.

### *03 juin 2010*

Phase de test sur les magasins continué avec appel des autres magasins. Lancement ce soir pour 3 magasins, j'ai donc initialisé l'application pour ces magasins niveau fond de caisse et procès verbal.

### *04 juin 2010*

Premier bug découvert, les chèques différés ne se calculent pas de la bonne façon, ça a donc créé une erreur à tous les magasins tests. Matinée passé à les rappeler pour leur expliquer et pour relancer la procédure.

Appel des autres magasins pour mettre au point et continuer la formation téléphonique. Lancement le soir pour de nouveaux magasins et initialisation de l'application pour ceux là.

## **Semaine 9**

### *07 juin 2010*

#### → Présentation devant l'équipe comptabilité et trésorerie.

Beaucoup d'inquiétudes vis-à-vis de l'application via un souci de compréhension de son utilité. L'équipe comptabilité et trésorerie ayant peu de confiance dans les magasins, ils ne voulaient pas que les magasins aient trop de responsabilités vis-à-vis des champs à saisir.

L'après midi au téléphone pour dépanner les magasins ayant besoin d'aide et pour former les magasins non encore formé.

### *08 juin 2010*

Mise en place de détails dans le code, comme la casse sur les noms et prénoms saisies pour obtenir un nom au format NOM et prénom au format Prénom.

Appel des magasins pour régler les soucis quand il n'y a pas eu de remonté de feuille.

09 juin 2010-10 juin 2010-11 juin 2010

Rédaction du rapport de stage entre deux mises à jour du code et coup de téléphones avec les magasins.

## **Semaine 10**

*14 juin 2010*

Connexion via ReportOne à la BDD, on y récupère les tables sans problèmes, les croisements avec d'autres tables se feront sans soucis.

Rédaction du rapport de stage

*15 juin 2010*

Rédaction du rapport de stage. Mise en place de l'historique.

*16 juin 2010*

→ Envoyer Rapport à [gilles.nachouki@univ-nantes.fr](mailto:gilles.nachouki@univ-nantes.fr) en format PDF et 2 versions imprimées et reliées.

Rédaction du rapport de stage et Annexes. Impression, relié et envoyé

*17 juin 2010*

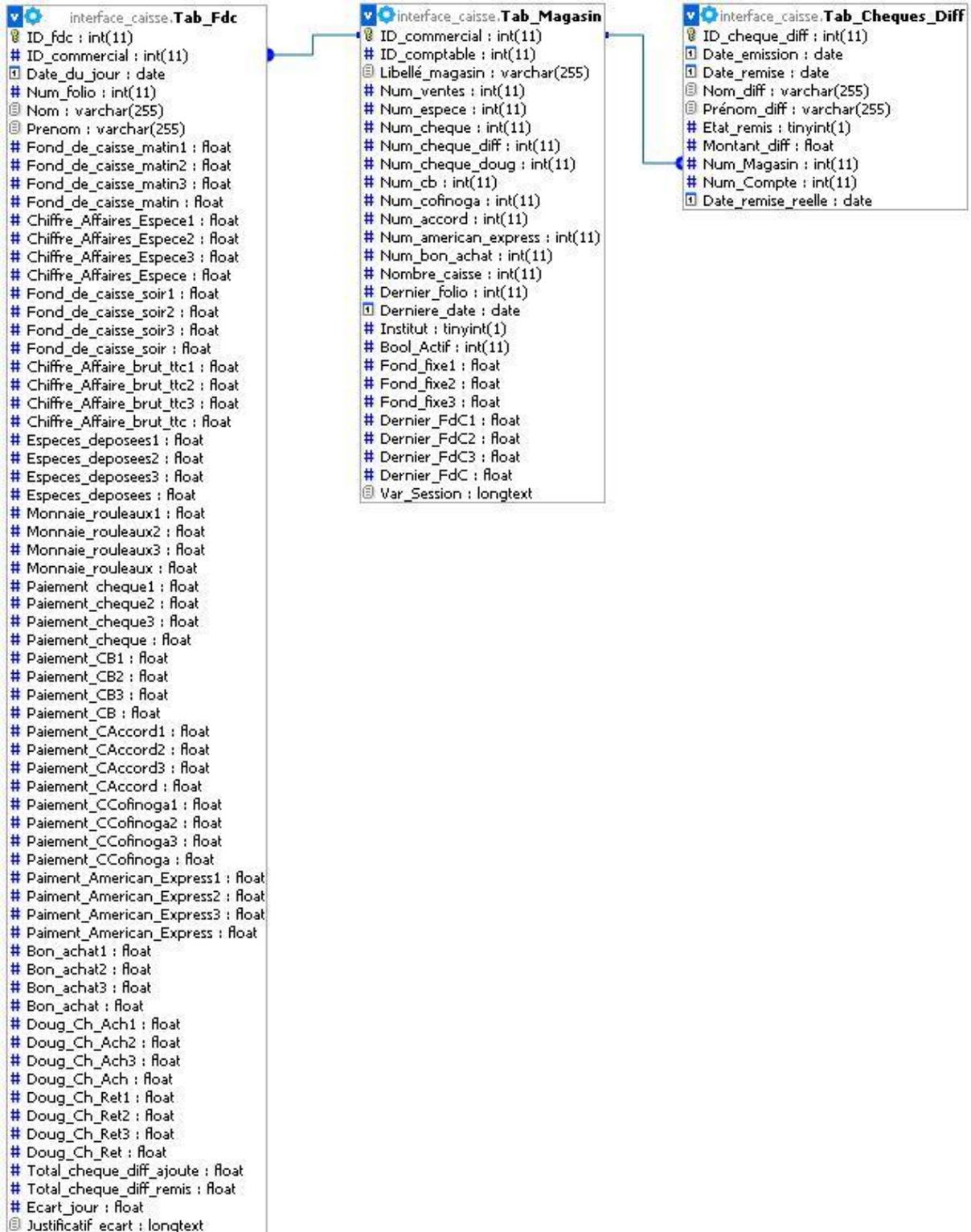
→ Finir ajout des chèques procès verbal avec option de suppression

*18 juin 2010*

→ Passer l'application finale sur l'intranet et donner les instructions à Sébastien pour l'entretien des données.

## Modèle logique de données

Modèle de données complet :



## Exemple existant

DOUGLAS PONTIVY

**CODE : CEX**

**TRESORERIE JOURNAL LUNDI 07 2010**

**JOURNAL : 22**

**FOLIO : 0906**

LIGNE	LIBELLE	N° COMPTE	SA	RECETTES CREDIT	DEPENSES DEBIT	SOLDE
1	Report	531009		<b>455,15</b>		
2	VENTES	707009	09	6 763,98		
3	ESPECES N° remise :	512250	CIO		760,00	
4	CHEQUES N° remise :	512250	CIO		1 263,57	
5	C.B. N° remise :	513720	BPO		4 626,93	
6	COFINOGA N° remise :	513720	BPO		0,00	
7	AX N° remise :	513720	BPO		0,00	
8	ECART DU JOUR	654120	09		<b>62,48</b>	
9	BON D'ACHATS CADHOC	580090			40,00	
10	BON D'ACHATS TIRGROUPE	580090			0,00	
11	BON D'ACHATS COMP ACCOR	580090			90,00	
12	Autres BON D'ACHATS C.E	580090				
13	CEZAM ACEVA				0,00	
14	Autres BON D'ACHATS C.E				0,00	
15	HAVAS				0,00	
16						
17						
18	DOUG <b>VENTE</b> CH CADEAU (76)	411100/10000		258,00		
19	DOUG <b>RETOUR</b> CH CADEAU (76)	411100/10000			179,00	
20						
21	<b>Justif ecart du jour</b>					
22						
23						
24						
25	<u>Signature et Date du Responsable :</u>	14 juin 2010				
26	<i>Pièces à joindre obligatoirement à la caisse</i>					
27	1 Bordereau Espèces					
28	1 Bordereau Chèques					
29	1 Bordereau Cartes bancaires					
30	1 Bordereau Cartes Cofinoga					
31	1 Bordereau Américain express					
32	Originaux Bons achats CE					
33	Originaux Chq cadeaux Elytis-Doug					
34	Originaux Chq Fidelytis - Doug 10€					
	Total Folio			7 477,13	6 959,50	<b>455,15</b>
	Contrôle					

*Feuille de caisse exemple*

S.A.S CLIN D'ŒIL  
PROCES VERBAL DE CAISSE

5007

**JOURNEE DU :** 16 juin 2010**DETAIL CHEQUES DIFFERES**

<i>Date d'émission</i>	<i>Nom</i>	<i>Prénom</i>	<i>Montant</i>	<i>Date de remise en banque</i>
20/05/10	THOMAS	MURIEL	44,90	20/06/10
18/05/10	OUARTINDI	ZOUBIDA	45,50	18/06/10
25/05/10	SEMPERE	MARTINE	77,40	25/06/10
31/05/10	CRUSE	ROLAND	156,6	15/06/10
22/05/10	MONTEAU	CELIA	32,5	22/06/10
20/05/10	MONTPION	MARIE CLAUDE	225,28	20/06/10
25/05/10	SEMPERE	MARTINE	77,40	25/07/10
20/05/10	BARON	JOCELYNE	232	28/06/10
04/06/10	AUDEBERT	ANNIE	55,36	03/07/10
21/05/10	MERIGUET	STEPHANIE	50,64	25/06/10
25/05/10	MURO	KATIA	1 366,56	15/06/10
28/05/10	BEZIAUD	MURIEL	1601,17	15/06/10
05/06/10	TEXIER	MONIQUE	124	05/07/10

A REMETTRE

A REMETTRE

A REMETTRE


Total Chèques Différés (1)  
Fond de Caisse Espèces (2)

4 089,31

Total à Reporter sur votre feuille de TRÉSORERIE (1+2)

4 089,31

Je certifie que le montant correspond à l'existant en caisse ce jour.

NOM :

PRENOM :

SIGNATURE :

*Procès verbal de caisse*

## Capture d'écran de quelques pages de l'application



la beauté complice

### Intranet Clin d'Oeil



Ajout chèque différé

Bonjour **Quentin GAUTIER**, vous traitez la journée du **14 Juin 2010**  
 Numéro Magasin : **1 - Nantes Boileau**  
 Folio numero **1017**

Veuillez entrer les dates d'émission et de remise en banque du chèque différé.

Date d'émission	Nom	Prénom	Montant	Date de remise en banque
10/06/2010	Gautier	Quentin	155.25	

Ajouter cheque    Retour Procès Verbal

juin							2010	
Lun	Mar	Mer	Jeu	Ven	Sam	Dim		
	1	2	3	4	5	6		
7	8	9	10	11	12	13		
14	15	16	17	18	19	20		
21	22	23	24	25	26	27		
28	29	30						

**FEUILLE DE CAISSE VERSION TEST**

Ajout d'un nouveau chèque différé et calendrier pour le saisie des dates



la beauté complice

### Intranet Clin d'Oeil



Bonjour **Quentin GAUTIER**, vous traitez la journée du **14 Juin 2010**  
 Numéro Magasin : **1 - Nantes Boileau**  
 Folio numero **1017**

Detail Remise

Libellés	Caisse 1	Caisse 2	Caisse 3
Espèces déposées en Banque ?	<input type="text"/>	<input type="text"/>	<input type="text"/>
Monnaie ?	<input type="text"/>	<input type="text"/>	<input type="text"/>
Chèques encaissés ?	<input type="text"/>	<input type="text"/>	<input type="text"/>
CB encaissés ?	<input type="text"/>	<input type="text"/>	<input type="text"/>
American Express encaissés ?	<input type="text"/>	<input type="text"/>	<input type="text"/>
Bons d'achats encaissés ?	<input type="text"/>	<input type="text"/>	<input type="text"/>
Douglas Chèques Vente ?	<input type="text"/>	<input type="text"/>	<input type="text"/>
Douglas Chèques Retour ?	<input type="text"/>	<input type="text"/>	<input type="text"/>

**IDENTIFICATION > PROCES VERBAL > FOND DE CAISSE > DETAIL VENTES > DETAIL REMISES > SYNTHESE**

**FEUILLE DE CAISSE VERSION TEST**

Détail des remises en banques avec détail sur 3 caisses

Bonjour **Quentin GAUTIER**, vous traitez la journée du **14 Juin 2010**  
 Numéro Magasin : **1 - Nantes Boileau**  
 Folio numero **1017**

Synthese

Ventes	Caisse1	Caisse2	Caisse3	Total
Chiffre d'affaires net ttc	3693.99	2038.78	0	5732.77
<a href="#">Modifier C.A. Ventes</a>				
Fond de Caisse	Caisse1	Caisse2	Caisse3	Total
Précédent fond de caisse	0	0	0	0
Chiffre d'affaires espèces	330.74	564.83	0	895.57
Fond de caisse du soir	45.5	44.46	0	89.96
<a href="#">Modifier Fond de Caisse</a>				
Détail Remises en Banque	Caisse1	Caisse2	Caisse3	Total
Espèces déposées en Banque	330	540	0	870
Monnaie	0	0	0	0
Chèques encaissés	631.4	136.03	0	767.43
CB encaissées	2610.33	1080.92	0	3691.25
American Express encaissés	47.52	43	0	90.52
Bons d'Achat encaissés	110	12	0	122
Douglas Chèques Vente	79	0	0	79
Douglas Chèques Retour	0	245	0	245
<a href="#">Modifier Détail Remises</a>				

Détails chèques différés	Total
Total chèques différés ajoutés	0
Total chèques différés remis	0
Total chèques différés soldés par un autre moyen de paiement	0
<a href="#">Modifier Procès Verbal</a>	

Ecart du jour

Ecart du jour	64.39
<p style="text-align: center;">Erreur fond de caisse</p> <p style="text-align: center;">Le fond de caisse du soir saisi ne concorde pas avec le fond de caisse théorique calculé.</p>	

[Valider](#) [Sauvegarder en Brouillon](#)

[IDENTIFICATION](#) > [PROCES VERBAL](#) > [FOND DE CAISSE](#) > [DETAIL VENTES](#) > [DETAIL REMISES](#) > [SYNTHESE](#)

**FEUILLE DE CAISSE VERSION TEST**

Page de synthèse avec écart du jour

## Webographie

<http://www.joomla.fr/> consulté le : 14 avril 2010  
<http://www.phpsources.org/scripts67-PHP.htm> consulté le : 14 avril 2010  
[http://openweb.eu.org/articles/css\\_impression/](http://openweb.eu.org/articles/css_impression/) consulté le : 14 avril 2010  
<http://www.win-design.com/fr/index.htm> consulté le : 14 avril 2010  
<http://g-rossolini.developpez.com/tutoriels/php/formulaires/> consulté le : 19 avril  
<http://fr.php.net/manual/fr/language.oop5.basic.php> consulté le : 19 avril  
<http://www.vulgarisation-informatique.com/php-poo.php> consulté le : 19 avril  
[http://www.editeurjavascript.com/scripts/scripts\\_formulaires\\_3\\_595.php](http://www.editeurjavascript.com/scripts/scripts_formulaires_3_595.php) consulté le : 19 avril  
<http://g-rossolini.developpez.com/tutoriels/php/formulaires/> consulté le : 20 avril  
<http://fr.php.net/manual/fr/language.oop5.basic.php> consulté le : 20 avril  
<http://www.toutjavascript.com/main/index.php3> consulté le: 27 avril  
<http://www.javascriptfr.com/dlzip.zipnix?ID=34725&accept=1> consulté le : 29 avril  
<http://www.phpfrance.com/> consulté le: 30 avril  
<http://www.phpsources.org/> consulté le: 30 avril  
<http://fr.selfhtml.org/html/tableaux/construction.htm> consulté le : 07 Mai  
<http://fr.php.net/manual/fr/function.serialize.php> consulté le : 12 mai  
<http://openweb.eu.org/css> consulté le: 28 Mai

## Bibliographie

Christine Jacquin

**Cours Technologie web 1**  
2009-2010

Jean Grelier

**Cours Gestion 1-3**  
2008-2010

Sébastien Cazalas

**Cours Informatique & Société**  
2010

[Guillaume Ponçon](#) , [Liberio Maesano](#)

**Best practices PHP 5**  
Edition Eyrolles

Thierry Templier, Arnaud Gougeon

**JavaScript pour le Web 2.0**  
Edition Eyrolles

## Code

Sélection parmi les 30 pages de code des solutions techniques misent en avant pour répondre à des problèmes.

```
<?php
/*****
/* Feuille qui gère le formulaire de remise en banque */
/* Date : 2010 */
/* Auteur : Quentin Gautier */
/*****
session_start();

//Création de tableaux avec les variables de chaque champs de caisses pour 3 caisses + un otal

/* 0 - */ $espece = array ('especes1','especes2','especes3','especes');
/* 1 - */ $monnaie = array ('monnaie1','monnaie2','monnaie3','monnaie');
/* 2 - */ $cheque = array ('cheque1','cheque2','cheque3','cheque');
/* 3 - */ $c_b = array ('c_b1','c_b2','c_b3','c_b');
/* 4 - */ $c_accord = array ('c_accord1','c_accord2','c_accord3','c_accord');
/* 5 - */ $c_cofinoga = array ('c_cofinoga1','c_cofinoga2','c_cofinoga3','c_cofinoga');
/* 6 - */ $american_express = array
('american_express1','american_express2','american_express3','american_express');
/* 7 - */ $bon_achat = array ('bon_achat1','bon_achat2','bon_achat3','bon_achat');
/* 8 - */ $Doug_Ch_Ach = array
('Doug_Ch_Ach1','Doug_Ch_Ach2','Doug_Ch_Ach3','Doug_Ch_Ach');
/* 9 - */ $Doug_Ch_Ret = array ('Doug_Ch_Ret1','Doug_Ch_Ret2','Doug_Ch_Ret3','Doug_Ch_Ret');

// création d'un tableau avec pour variable les valeurs des tableaux précédents.
$tableau_sessions =
array($espece,$monnaie,$cheque,$c_b,$c_accord,$c_cofinoga,$american_express,$bon_achat,$Doug_Ch_Ach,
$Doug_Ch_Ret);

//Pour chaque variables , calcul du total + création de la variable de session correspondante.
for ($i=0; $i<=9 ; $i++)
{
    $_SESSION[$tableau_sessions[$i][0]]=round($_POST[$tableau_sessions[$i][0]],2);
    $_SESSION[$tableau_sessions[$i][1]]=round($_POST[$tableau_sessions[$i][1]],2);
    $_SESSION[$tableau_sessions[$i][2]]=round($_POST[$tableau_sessions[$i][2]],2);

    $_SESSION[$tableau_sessions[$i][3]]=round(round($_SESSION[$tableau_sessions[$i][0]],2)+round($
    _SESSION[$tableau_sessions[$i][1]],2)+round($_SESSION[$tableau_sessions[$i][2]],2),2);
}

/* Redirige le client vers la fiche de synthèse */
header("Location: ../Traitement/CalculSynthese.php");
/* Garantie que le code ci-dessous n'est jamais exécuté. */
exit();
?>
```

Code de la page traitement détail remise

```

/Récupère les valeurs dans les tableaux de données
$cheque_differe = $_SESSION['$cheque_differe'];

if(!empty($_SESSION['$cheque_differe']))
{
    foreach ($cheque_differe as $key => $value)
    {
        $libelle = $value ;
        $nombre_champs++;
        foreach ($libelle as $value )
        {
            $nombre_variables++;
            if($key=="date_emission")
                $date_emission[]=$value;
            else if($key=="nom_diff")
                $nom_diff[]=$value;
            else if($key=="prenom_diff")
                $prenom_diff[]=$value;
            else if($key=="montant_diff")
                $montant_diff[]=$value;
            else if($key=="date_remise")
                $date_remise[]=$value;
        }
    }
    if($nombre_variables!=0&&$nombre_champs!=0)
        $nombre_chèques = $nombre_variables / $nombre_champs;

    for($i=0; $i<$nombre_chèques ; $i++)
    {
        //Insertion dans la BDD préparé des chèques différés ajouté le jour via une requête préparée.
        $req = $bdd->prepare("INSERT INTO Tab_Cheques_Diff(Date_emission, Date_remise, Nom_diff,
Prénom_diff, Etat_remis, Montant_diff, Num_Magasin, Num_Compte) VALUES(:date_emission,
:date_remise, :nom_diff, :prenom_diff, :etat_remis, :montant_diff, :num_magasin, :num_compte)");
        $req->execute(array(
            'date_emission' => convdatephp($date_emission[$i]),
            'date_remise' => convdatephp($date_remise[$i]),
            'nom_diff' => $nom_diff[$i],
            'prenom_diff' => $prenom_diff[$i],
            'etat_remis' => 0,
            'montant_diff' => $montant_diff[$i],
            'num_magasin' => $num_magasin,
            'num_compte' => 0
        ));
    }
}

```

Méthode d'ajout de chèque différé pensée après multiples modifications.

Ici, un tableau avec des clés correspondantes aux noms des libellés d'ajout de chèque est récupéré et traité. Pour chaque valeur il ajoute dans la base de données les enregistrements présents dans le tableau pour chaque champ.

```

// Enleve le '0' des nombres < 10
function ConvNum(tabDeDate)
{
    for (i=0; i<tabDeDate.length; i++)
        tabDeDate[i] = (tabDeDate[i].charAt(0)=='0')?tabDeDate[i].charAt(1):tabDeDate[i];
    return tabDeDate;
}

// Retourne vrai si valeur_date est postérieure ou egale à la date du jour
function DateFuture(valeur_date)
{
    var tabDate = valeur_date.split('/');
    var datAujourdhui = new Date();
    tabDate = ConvNum(tabDate);
    if (valeur_date.length > 0)
    {
        var datTest_Date = new Date(parseInt(tabDate[2]), parseInt(tabDate[1])-1,
parseInt(tabDate[0]));
        if (datTest_Date < datAujourdhui) return false;
    }
    return true;
}

// Retourne 1 si valeur_date1 < valeur_date2
// 0 si valeur_date1 = valeur_date2
// -1 si valeur_date1 > valeur_date2
function Compare_Dates(valeur_date1, valeur_date2)
{
    var tabDate1 = valeur_date1.split('/');
    tabDate1 = ConvNum(tabDate1);
    var datTest_Date1 = new Date(parseInt(tabDate1[2]), parseInt(tabDate1[1])-1, parseInt(tabDate1[0]));
    var tabDate2 = valeur_date2.split('/');
    tabDate2 = ConvNum(tabDate2);
    var datTest_Date2 = new Date(parseInt(tabDate2[2]), parseInt(tabDate2[1])-1, parseInt(tabDate2[0]));

    return (datTest_Date2-datTest_Date1==0)?"0":(datTest_Date2-datTest_Date1<0)?"-1":"1";
}

function Verifier()
{
    var strMsg="";
    if ((document.getElementById('date_emission').value !=
"")&&(document.getElementById('nom_diff').value != "")&&(document.getElementById('prenom_diff').value
!= "")&&(document.getElementById('montant_diff').value !=
"")&&(document.getElementById('date_remise').value != ""))
    {

        if(parseFloat(document.getElementById('montant_diff').value)==document.getElementById('montant_
diff').value)
            {
                switch
(Compare_Dates(document.getElementById('date_emission').value,
document.getElementById('date_remise').value))
                {
                    case "-1" :
                        alert('Date de remise antérieur à la date de d\'émission,
vérifiez les dates! '); return false; break;
                    case "0" :
                        alert('Erreur ! Les dates sont identiques. '); return false;
                }
            }
        }
}

```

```

break;

                                case "1" :
                                return true; break;
                                default :
                                alert('Comparaison impossible');break;
                                }
                                }
                                // les données sont ok, on peut envoyer le formulaire/
else
{
    alert('Le montant du chèque doit être une valeur numérique. ');
    return false;
}
else
{
    // sinon on affiche un message
    alert("Saisissez les champs du procès verbal");
    // et on indique de ne pas envoyer le formulaire
    return false;
}
}

```

Scripts JS utilisés sur l'ajout des chèques pour vérifier la validité des champs.

**Annexe – Manuel d'utilisateur**  
**Procédure feuille de caisse**