

Documentation technique de OneWay Tickets



Devaud Alan

I.IN-P4A

TPI 2014 – 2015

Table des matières

Table des matières	2
Introduction	4
Etude d'opportunité	4
Analyse fonctionnelle	6
Généralité	6
Schéma	6
Description du schéma	6
Description des fonctionnalités globales	7
Réservation des billets	7
Réservation en avance	7
Validation du paiement	7
Mode administrateur	7
Gestion des séances	7
Gestion des salles	7
Gestion des films	7
Statistique	8
Description détaillée de l'interface	8
Menu utilisateur	8
Réserver la séance	8
Réserver une séance en avance	9
Login administrateur	9
Menu administrateur	10
Gestion	10
Description des éléments de sécurité	12
Analyse organique	13
Généralités	13
Réservation d'une séance	14
Administration	14
Stockage des données	14
Description détaillée des algorithmes de résolution	16
U_ImageBouton.pas	16
U_FP.pas	19
U_OneWayTicket.pas	26
Test	34

Plan de test	34
Rapport de test	37
Conclusion	39
Bilan	39
Améliorations envisageables	39
Comparaison des plannings	40
Planning prévisionnel	40
Planning réel	41
Bibliographie	42

Introduction

OneWay Tickets est une application qui permet d'assurer la gestion, la production et la sauvegarde de billets d'entrées à une manifestation ponctuelle. Elle donne la possibilité d'avoir sa propre billetterie.

Cependant, *OneWay Tickets* est personnalisée en vue d'assurer la gestion d'un cinéma. Facile d'utilisation, *OneWay Tickets* donne à ses utilisateurs la possibilité de gérer leur propre billetterie de cinéma. Ils peuvent gérer les salles, les films, les séances et connaître le nombre de places vendues. Quand les réservations de billets sont effectuées *OneWay Tickets* génère les tickets.

L'application est divisée en deux modes. Le mode utilisateur qui donne la possibilité de faire des réservations de séances et un mode administrateur qui permet de gérer les films, les salles et les séances. L'administrateur a un privilège plus grand que l'utilisateur et a donc accès à plus de fonctionnalités.

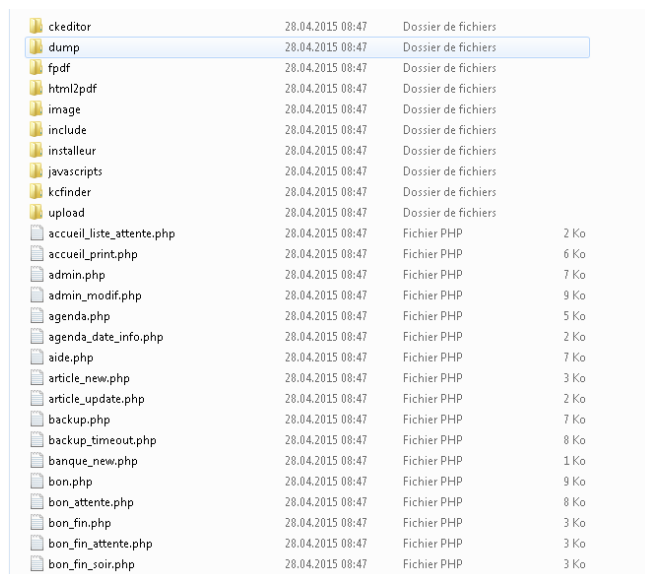
Etude d'opportunité

Il existe déjà des applications et des sites web qui permettent la gestion de la billetterie d'une manifestation. Le projet qui se rapproche le plus du mien est « *Net Billetterie* ».

La première application que j'ai trouvée est « *Net Billetterie* ». Elle permet la gestion de salles de spectacles ou de saisons culturelles. Les utilisateurs ont la possibilité de gérer les spectacles, les spectateurs, les tarifs, les abonnements et la billetterie.

L'installation de « *Net Billetterie* » est complexe. En fait, « *Net Billetterie* » est un site web que l'on utilise en local. Ce qui rend la structure confuse et subtile à comprendre pour un utilisateur qui n'a pas l'habitude de travailler avec ce genre de programme. Non seulement la structure est complexe, « *Net Billetterie* » demande la modification de certains de ses fichiers pour, par exemple, l'accès à la base de données Apache.

Une fois l'installation terminée et les connexions à la base de données faite, on peut se connecter avec nos comptes. « *Net Billetterie* » divise son site web en plusieurs sections réservées à des utilisateurs spécifiques. Il existe les sections pour les abonnements, la billetterie, la liste d'attente (enregistrer des spectateurs sur la liste d'attente), le « charger de communication » (Envoyer des mails et imprimer la liste des places vendues par



ckeditor	28.04.2015 08:47	Dossier de fichiers	
dump	28.04.2015 08:47	Dossier de fichiers	
fpdf	28.04.2015 08:47	Dossier de fichiers	
html2pdf	28.04.2015 08:47	Dossier de fichiers	
image	28.04.2015 08:47	Dossier de fichiers	
include	28.04.2015 08:47	Dossier de fichiers	
installateur	28.04.2015 08:47	Dossier de fichiers	
jscripts	28.04.2015 08:47	Dossier de fichiers	
kcfinder	28.04.2015 08:47	Dossier de fichiers	
upload	28.04.2015 08:47	Dossier de fichiers	
accueil_liste_attente.php	28.04.2015 08:47	Fichier PHP	2 Ko
accueil_print.php	28.04.2015 08:47	Fichier PHP	6 Ko
admin.php	28.04.2015 08:47	Fichier PHP	7 Ko
admin_modif.php	28.04.2015 08:47	Fichier PHP	9 Ko
agenda.php	28.04.2015 08:47	Fichier PHP	5 Ko
agenda_date_info.php	28.04.2015 08:47	Fichier PHP	2 Ko
aide.php	28.04.2015 08:47	Fichier PHP	7 Ko
article_new.php	28.04.2015 08:47	Fichier PHP	3 Ko
article_update.php	28.04.2015 08:47	Fichier PHP	2 Ko
backup.php	28.04.2015 08:47	Fichier PHP	7 Ko
backup_timeout.php	28.04.2015 08:47	Fichier PHP	8 Ko
banque_new.php	28.04.2015 08:47	Fichier PHP	1 Ko
bon.php	28.04.2015 08:47	Fichier PHP	9 Ko
bon_attente.php	28.04.2015 08:47	Fichier PHP	8 Ko
bon_fin.php	28.04.2015 08:47	Fichier PHP	3 Ko
bon_fin_attente.php	28.04.2015 08:47	Fichier PHP	3 Ko
bon_fin_soir.php	28.04.2015 08:47	Fichier PHP	3 Ko

Figure 1 : Structure du dossier "Net Billetterie"

abonnement ou réservation), le comptable et l'administrateur. Chacune de ces sections à ses propres droits et ne peut pas empiéter sur les activités des autres. Seul l'administrateur à le plein pouvoir et peut accéder à toutes ces sections en plus de la sienne.



Figure 2 : Login pour les différents utilisateurs des sections

La deuxième application est « Weezevent » qui est un site web disponible en ligne.

« Weezevent » permet de créer sa billetterie en ligne en seulement quelques minutes. Il met à disposition la possibilité de suivre ses ventes en temps réel. Il donne également la possibilité à l'utilisateur d'intégrer sur son propre site un Widget (mini logiciel intégré à la page du site web) de billetterie qui permet de ne pas rediriger le client sur un site tiers.

« Weezevent » est facile à configurer, sans connaissance et est totalement disponible en ligne. Il donne la possibilité à l'utilisateur de créer ses billets personnalisés avec les informations de la manifestation. On peut créer les billets en deux formats. En taille réduite qui donne un petit billet avec les informations essentielles. Billet qui est plutôt vendu directement sur le lieu de la manifestation. Le deuxième billet est en forma A4 et donne un peu plus d'informations. Ce dernier est plutôt vendu quand les clients achètent sur Internet.



Figure 3 : Billet vendu sur internet



Figure 4 : Billet vendu à la manifestation

Les principales différences entre mon application et celles que j'ai trouvées sur le web sont, dans un premier temps, la facilité d'installation et d'utilisation. Dans *OneWay Tickets* pour mettre en place la manifestation on doit modifier les fichiers la concernant.

Ensuite, mon application ne comprend que deux modes d'utilisation. En premier lieu, l'utilisateur est le caissier et en deuxième, l'administrateur a accès à la gestion de la manifestation, ainsi qu'aux suivis des ventes.

Pour finir, *OneWay Tickets* à l'avantage d'être intuitif pour l'utilisation tout aussi bien en simple utilisateur qu'en mode administrateur.

Analyse fonctionnelle

Généralité

Schéma

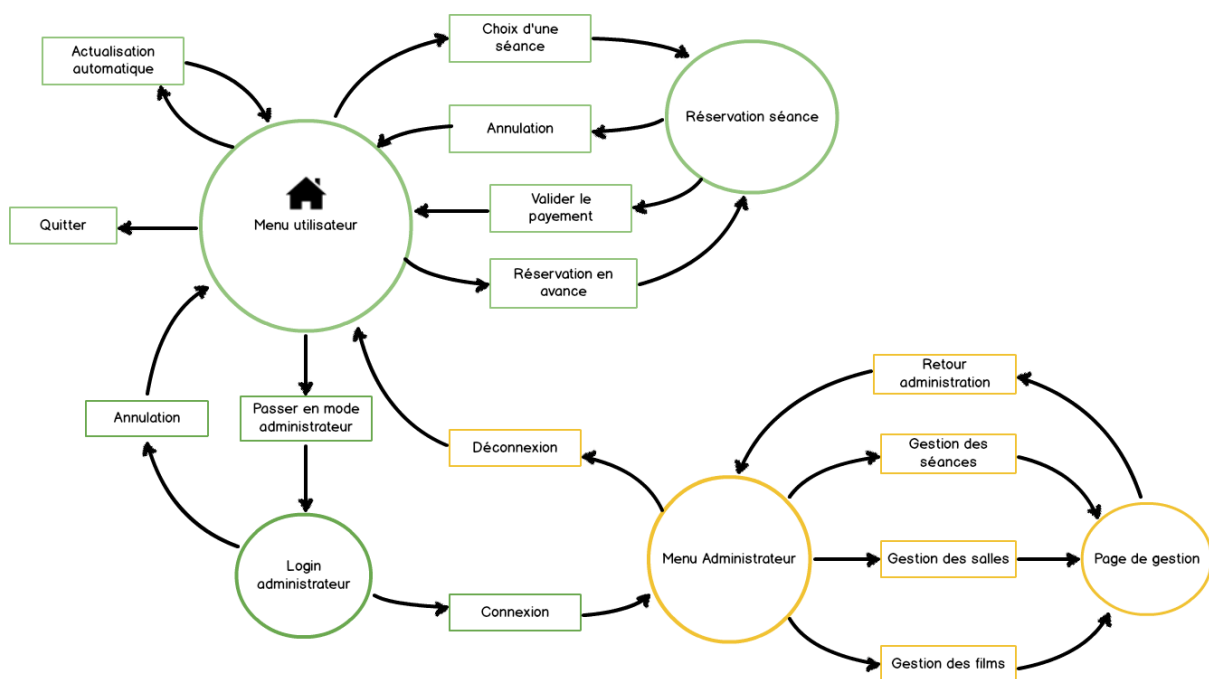


Figure 5 : Schéma général de l'application

Description du schéma

Les utilisateurs, en vert, ont accès au menu de l'utilisateur et à la connexion pour l'administration. Les utilisateurs peuvent faire le choix d'une séance, saisir les informations pour la réservation et valider le paiement. Ils ont accès à la fenêtre de connexion pour se connecter en administrateur, s'ils ont les identifiants qui leurs permettent.

Les administrateurs, en orange, ont accès au mode administrateur qui leur permet de consulter les statistiques et gérer tout ce qui concerne les salles, les films et les séances. Ils peuvent se déconnecter du mode administrateur pour revenir au mode utilisateur.

Description des fonctionnalités globales

Réservation des billets

La réservation des billets est la fonctionnalité qui permet de réserver une séance pour un certain nombre de personnes donné par le client. Elle permet donc une réservation en y ajoutant le nombre de billets que le client souhaite (ex. : 1 enfant, 2 adultes) et de valider le paiement.

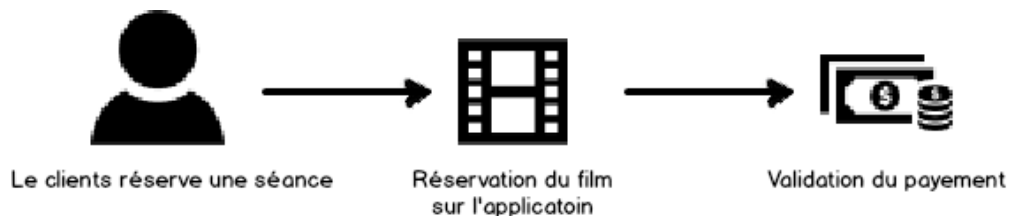


Figure 6 : Schéma de réservation

Réservation en avance

La réservation en avance permet à l'utilisateur de faire une réservation pour une séance qui se déroulera dans plusieurs jours.

Validation du paiement

La validation du paiement est la fonctionnalité qui détermine la réservation du client et procède à la génération des billets.

Mode administrateur

Le mode administrateur permet d'accéder aux fonctionnalités supplémentaires que seul l'administrateur à accès. Tels que la gestion des séances ou encore des salles.

Gestion des séances

La gestion des séances permet à l'administrateur de créer, modifier ou encore supprimer une séance en indiquant la salle et le film concerné.

Gestion des salles

Cette fonctionnalité permet à l'administrateur de gérer les salles du cinéma. C'est-à-dire d'ajouter une salle (ex. : s'il y a un agrandissement du bâtiment), de modifier une salle ou encore de supprimer une salle.

Gestion des films

L'administrateur est le seul à avoir accès à la gestion des films. Cette fonctionnalité lui permet d'ajouter, de modifier ou de supprimer un film que le cinéma possède.

Statistique

Cette fonctionnalité affiche les statistiques des ventes du cinéma.

Description détaillée de l'interface

Menu utilisateur

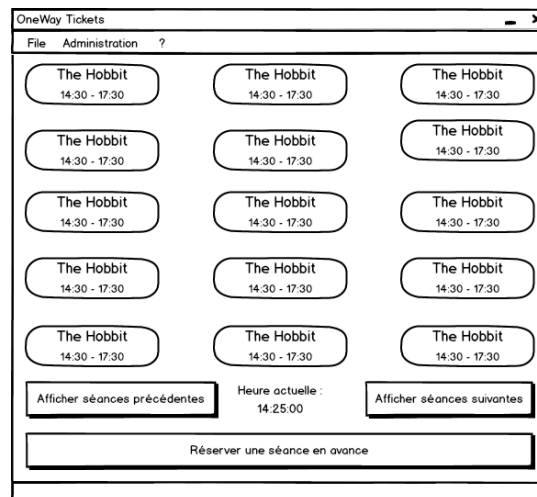


Figure 7 : Interface du menu d'utilisateur

Le menu utilisateur permet à l'utilisateur de voir les séances disponibles, de l'heure actuelle jusqu'à la fin de la journée. Il peut ainsi choisir la séance que le client veut regarder.

Cette fenêtre donne la possibilité d'afficher des séances futures et permet aux clients qui le désirent, de réserver une séance plus tardive.

L'utilisateur peut également faire une réservation pour une séance qui passe dans plusieurs jours.

Le menu utilisateur permet également à un administrateur de se connecter pour atteindre les fonctionnalités réservées aux administrateurs.

Réserver la séance

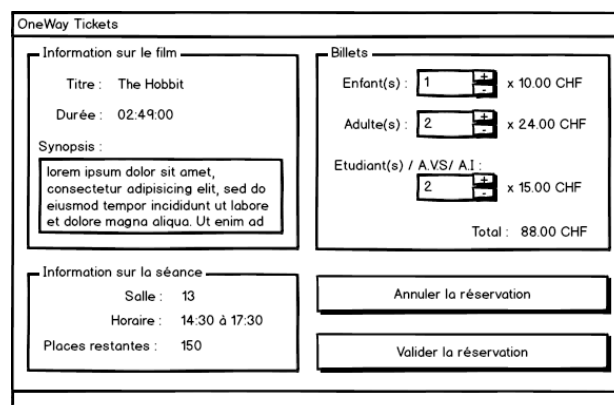


Figure 8 : Interface pour réserver une séance

Cette interface permet à l'utilisateur d'avoir de plus amples informations sur la séance sélectionnée (informations sur le film et la séance). C'est aussi ici que l'utilisateur entre le nombre de billets que le client désire.

Le tarif par billet (enfant, adulte ou étudiant) ainsi que le total, s'affichent sur cette fenêtre. Quand l'utilisateur a mis le nombre de billets souhaités, le prix est calculé et affiché pour connaître le total de la réservation.

Une fois que tout est bon l'utilisateur peut valider et procéder au paiement.

Réserver une séance en avance

OneWay Tickets

Choisir un film :
The Hobbit

Information sur le film

Titre : The Hobbit

Durée : 02:49:00

Synopsis :
lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad

Billets

Enfant(s) : 1 x 10.00 CHF

Adulte(s) : 2 x 24.00 CHF

Etudiant(s) / AVS / AI : 2 x 15.00 CHF

Total : 88.00 CHF

26.04.2015	14:00
12.05.2015	15:00
12.05.2015	18:00
13.05.2015	20:00

Information sur la séance

Salle : 13

Date : 26.04.2015

Horaire : 14:30 à 17:30

Places restantes : 150

Valider la réservation

Annuler la réservation

Figure 9 : Interface pour réserver en avance

Cette interface permet de faire une réservation en avance. C'est ici que l'utilisateur définit le film, la date et le nombre de billets que le client désire réserver à l'avance.

Login administrateur

OneWay Tickets

Veuillez entrer le mot de passe administrateur

Connexion

Annuler

Figure 10 : Interface du login

Le login permet à un utilisateur de passer en mode administrateur et d'accéder aux fonctionnalités de l'administrateur, à condition qu'il ait le mot de passe administrateur.

Menu administrateur

The screenshot shows a web interface titled "OneWay Tickets" with a sub-header "Gestions ...". It is divided into two main sections. The left section, titled "Statistique vente", contains a list of sales statistics: "Billet total vendu : 100", "Billet vendu durant le mois courant : 50", "Billet enfants vendu : 15", "Billet adultes vendu : 10", and "Billet etudiants / A.VS / A.I vendu : 25". The right section, titled "Films", displays "Film le plus vu : The Hobbit". At the bottom right, there is a "Déconnexion" button.

Figure 11 : Interface menu administrateur

Le menu administrateur affiche les statistiques de vente et le film le plus vu. Depuis ici on peut atteindre les pages de gestion concernant les films, les salles et les séances.

Gestion

The screenshot shows a web interface titled "OneWay Tickets" with a sub-header "Gestion". It features a table with three columns: "Films", "Durée (min)", and "Synopsis". The table contains seven rows, all with "The Hobbit" in the "Films" column, "169" in the "Durée (min)" column, and "lorem ipsum" in the "Synopsis" column. Below the table, there are three buttons: "Modifier", "Supprimer", and "Ajouter".

Films	Durée (min)	Synopsis
The Hobbit	169	lorem ipsum
The Hobbit	169	lorem ipsum
The Hobbit	169	lorem ipsum
The Hobbit	169	lorem ipsum
The Hobbit	169	lorem ipsum
The Hobbit	169	lorem ipsum
The Hobbit	169	lorem ipsum

Figure 12 : Interface gestion

La page gestion permet de gérer les salles, les films et les séances. La fenêtre de gestion change selon l'option de gestion choisie, mais la structure reste la même. C'est-à-dire que si à la fenêtre « Menu Administrateur » l'administrateur choisi « gestion > Salles » la liste des éléments, les modifications, les suppressions et les ajouts concerneront les salles et affichera les bonnes informations.

Modification d'un élément

La modification d'un élément s'effectue sur la même fenêtre que la gestion. L'élément qui est modifié doit être sélectionné dans la liste.

OneWay Tickets

Gestion

Films	Durée (min)	Synopsis
The Hobbit	169	lorem ipsum
The Hobbit	169	lorem ipsum
The Hobbit	169	lorem ipsum
The Hobbit	169	lorem ipsum
The Hobbit	169	lorem ipsum
The Hobbit	169	lorem ipsum
The Hobbit	169	lorem ipsum

Modifier Supprimer Ajouter

Modification

Nom du film : Synopsis :

Durées (min.) :

Annuler Valider

Figure 13 : Modification d'un élément

La modification s'affiche sous les boutons de gestion. On peut à partir de là, modifier notre élément et valider notre modification ou annuler si l'on a changé d'avis.

Ajouter un élément

OneWay Tickets

Nom du film :

Durée (min.) :

Synopsis :

Annuler Ajouter

Figure 14 : Interface pour ajouter un élément

Cette fenêtre permet d'ajouter un élément aux listes.

Créer une séance

L'ajout d'une séance se passe différemment que pour ajouter un film ou une salle.

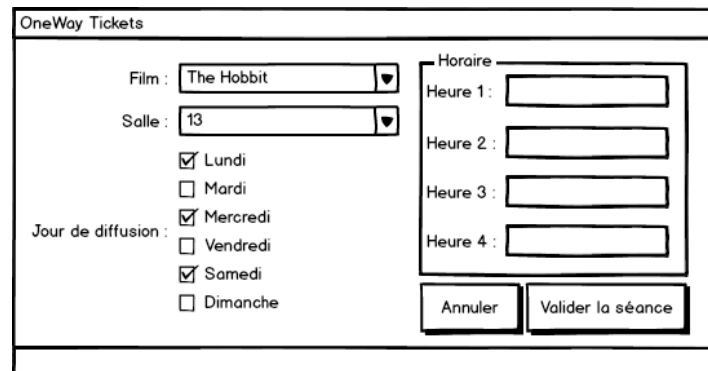


Figure 15 : Interface pour créer une séance

Pour l'ajout d'une séance, l'utilisateur choisi un film et une salle, puis il peut choisir les jours où le film est diffusé chaque semaine. L'administrateur a la possibilité de mettre jusqu'à 4 horaires de diffusion qui sont les même pour tous les jours sélectionnés.

Description des éléments de sécurité

Pour le bon fonctionnement du programme, des éléments de sécurité ont été mis en place. Notamment pour la saisie des données ou les problèmes de salles comme, par exemple, deux films en même temps dans la même salle.

Les éléments de sécurité sont les suivants :

- Les séances qui sont complètes ne sont pas cliquables.

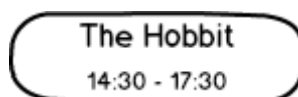


Figure 16 : séance libre



Figure 17 : séance complète

- Dans le menu utilisateur, il est impossible d'aller dans des séances déjà passées.
- Pour la réservation des billets, l'utilisateur ne peut entrer ni des lettres ni des caractères spéciaux et le champ ne doit pas être vide ou inférieur à zéro.
- Un utilisateur ne peut pas accéder aux fonctionnalités des administrateurs.
- Pour les champs textes dans la gestion, ils ne peuvent pas être vides.
- Pour les ajouts de séances, les champs d'horaires doivent correspondre au format heures/minutes.
- Dans l'ajout d'une séance, un film, une salle et au minimum un jour de diffusion avec une heure de « passage » doivent être entrés pour que la séance soit valide.
- A la validation d'une séance, la salle et les heures de diffusion sont vérifiées pour éviter d'avoir une diffusion de deux films différents dans la même salle à la même heure (voire fig. 18 et fig. 19).

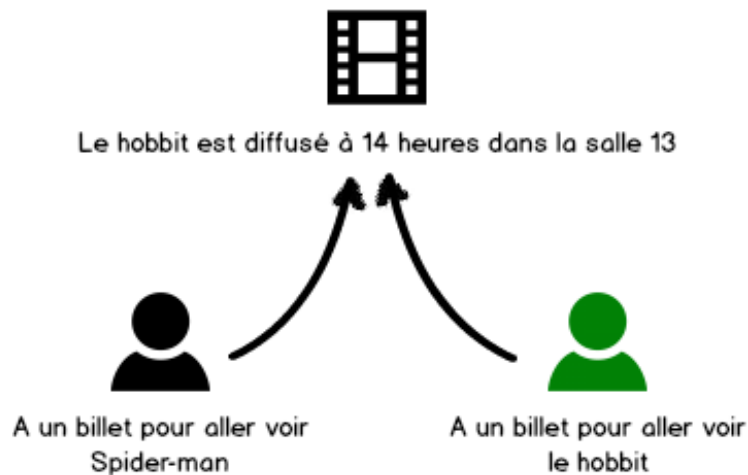


Figure 18 : Sans la vérification des séances

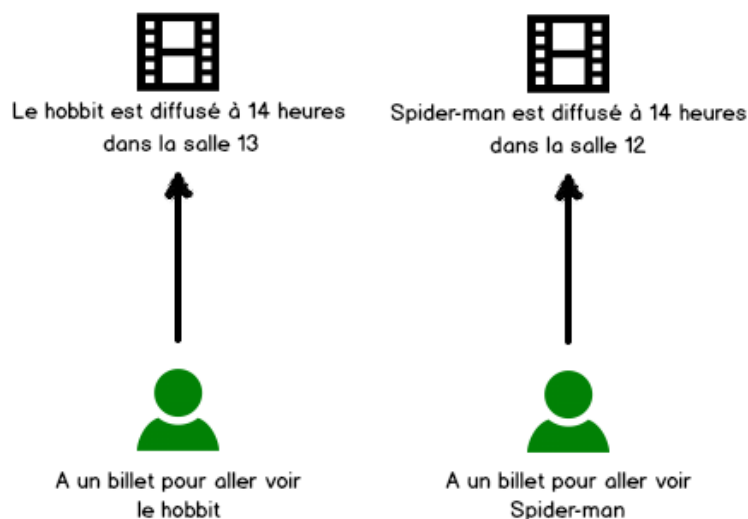


Figure 19 : Avec la vérification des séances

Analyse organique

Généralités

OneWay Tickets est une application qui est développée en Delphi, langage de programmation basé sur le Pascale.

Au démarrage, l'application charge dans un tableau, les séances disponibles, de l'heure actuelle jusqu'à la fin de la journée. Ces séances sont affichées sur le menu à l'aide de boutons qui indiquent la séance (le film et la tranche d'heure de diffusion). Si une séance est complète, le bouton est désactivé et est affiché en rouge.

Sur ce menu, nous avons la possibilité de naviguer pour voir d'éventuelles séances qui ne seraient pas encore affichées.

Réservation d'une séance

Pendant une réservation, les tarifs affichés à ce moment sont chargé depuis un fichier. Les prix peuvent varier entre deux films, car le prix n'est pas seulement calculé via le fichier des prix, il est également additionné au prix du film.

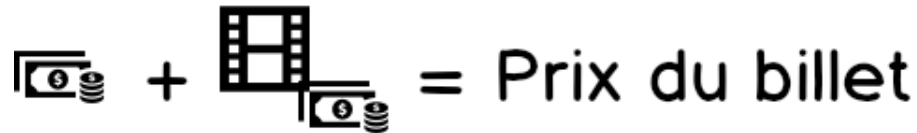


Figure 20 : Schéma pour le calcul du prix d'un billet

Pour mettre le nombre de billets, il y a un input de type texte pour chaque type de billet (enfant, adulte, etc.). Pour augmenter ou diminuer le nombre de billets, des boutons « + » et « - » sont attribués à chaque input.

Le prix total du billet est calculé chaque fois que le nombre de billets a augmenté ou diminué dans un input.

Quand l'utilisateur appuie sur le bouton pour valider, les informations sont transmises pour générer les billets.

Administration

Lors de l'identification, l'application vérifie que les informations entrées soient correctes. Si c'est le cas l'utilisateur, maintenant administrateur, peut accéder aux fonctionnalités d'administration. Si ce n'est pas le cas, un message d'erreur s'affiche et l'utilisateur est renvoyé au menu de l'utilisateur.

Dans l'administration, les statistiques sont générées par rapport aux données qui sont sauvegardées dans les fichiers de l'application.

Les pages de gestion chargent les données sur les films, les salles et les séances, depuis les fichiers les concernant.

Pour l'ajout de films ou de salles, l'administrateur doit remplir des inputs de type textes et valider son choix. A ce moment-là, le logiciel vérifie que les données sont inexistantes. Si elles sont existantes le logiciel renvoie un message d'erreur, sinon les données sont ajoutées et l'administrateur est renvoyé sur la page de gestion.

Stockage des données

Pour le stockage, les films, les salles, les prix et les statistiques sont stockés dans des fichiers CSV. Les séances sont stockées dans un fichier INI. Ces fichiers sont appelés durant le programme pour charger les données dont l'application a besoin.

Chaque fichier a sa propre structure qui se différencie selon les éléments que contient le fichier. Voici les différentes données que contiennent les fichiers.

Films : <ul style="list-style-type: none">• Nom du film• Durée du film• Synopsis• Prix du film	Salles : <ul style="list-style-type: none">• Nom de la salle• Nombre de place maximum
Prix : <ul style="list-style-type: none">• Catégorie (Ex. Enfant)• Prix	Séance : <ul style="list-style-type: none">• Section• Nom du film• Nom de la salle• Jours de diffusion• Heure de diffusion 1• Heure de diffusion 2• Heure de diffusion 3• Heure de diffusion 4• Diffusé
Réservation : <ul style="list-style-type: none">• Numéro de séance• Nombre de billet enfants• Nombre de billet adultes• Nombre de billet étudiants/a.v.s/a.i• Date de réservation• Prix total• Places totale	Statistiques : <ul style="list-style-type: none">• Nom de la statistique (ex. : BilletTotal)• Nombre de billets

Explication sur l'utilité de ces fichiers :

- *Films* : permet de renseigner sur les films qui sont disponibles dans le cinéma.
- *Salles* : permet de connaître les salles qui existent dans le cinéma.
- *Prix* : permet d'enregistrer le prix des tickets.
- *Séance* : enregistre toutes les séances qui sont possibles et permet de savoir si une séance est diffusée ou si elle ne l'est plus.
- *Réservation* : enregistre les tickets vendus.

Description détaillée des algorithmes de résolution

Dans cette section je parle et détaille les algorithmes utilisés durant le développement. Evidemment, je ne parle pas des fonctions de simple traitement de champs mais des algorithmes complexes.

U_ImageBouton.pas

U_ImageButton est un objet que j'ai créé. Il permet d'avoir un bouton de couleur avec certaines informations sur une séance. Cet objet a été créé pour avoir une couleur rouge quand une séance est complète. Cet objet est hérité de la classe TImage.

Diagramme de classe

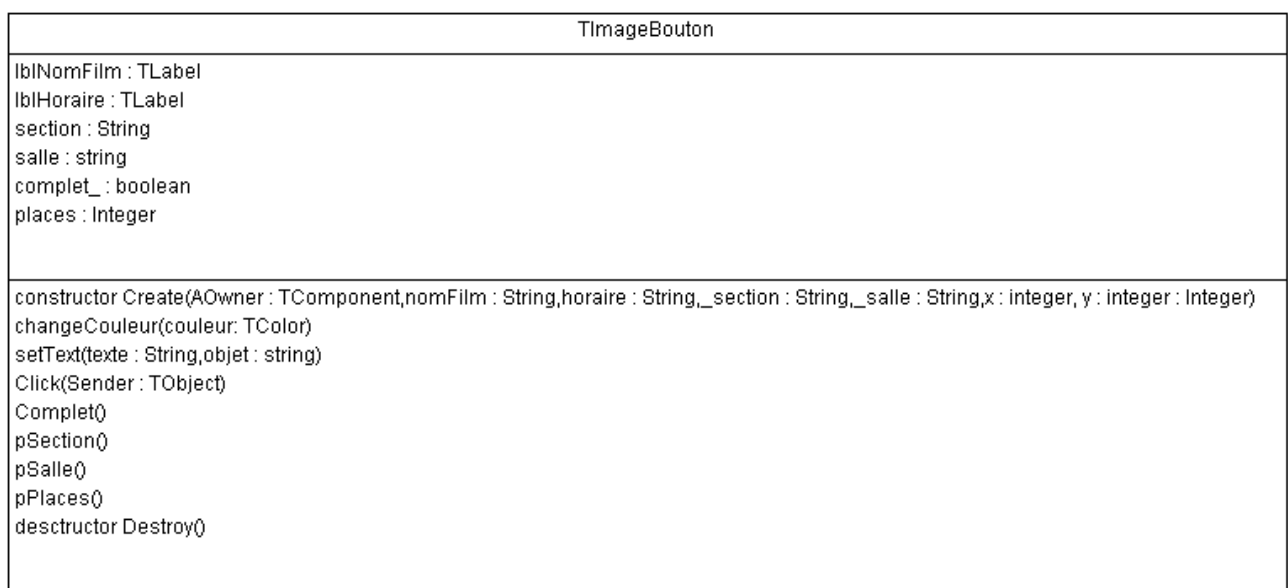


Figure 21 : Classe TImageBouto

Propriété

- *lblNomFilm* : est un label qui permet d'afficher le nom du film.
- *lblHoraire* : affiche l'heure de la séance.
- *section* : enregistre le nom de la section de la séance pour la retrouver dans le fichier des séances.
- *Salle* : l'objet enregistre le nom de la salle.
- *Complet_* : enregistre si la salle est complète.
- *Places* : enregistre le nombre de places restantes dans la salle.

Méthode

constructor TImageBouton.Create(AOwner : TComponent; nomFilm, horaire, _section, _salle: String; x, y : integer);
inherited Create(AOwner)
self.Parent ← (AOwner as TForm) self.Width ← WIDTH_IMAGE self.Height ← HEIGHT_IMAGE self.Top ← y self.Left ← x self.pSection ← _section self.pSalle ← _salle self.complet ← false
self.ChangeCouleur (clGreen)
lblNomFilm ← TLabel.Create(self.Parent)
lblNomFilm.Parent ← (self.Parent as TForm) lblNomFilm.Transparent ← true lblNomFilm.AutoSize ← false lblNomFilm.Width ← self.Width lblNomFilm.Left ← self.Left lblNomFilm.Top ← self.Top + 4 lblNomFilm.Alignment ← taCenter lblNomFilm.FontSize ← 10 lblNomFilm.FontStyle ← [fsBold] lblNomFilm.FontName ← 'Arial'
lblHoraire ← TLabel.Create(self.Parent)
lblHoraire.Parent ← (self.Parent as TForm) lblHoraire.Transparent ← true lblHoraire.AutoSize ← false lblHoraire.Width ← self.Width lblHoraire.left ← self.Left lblHoraire.Top ← lblNomFilm.Top + 24 lblHoraire.Alignment ← taCenter lblHoraire.FontSize ← 8 lblHoraire.FontName ← 'Arial'
self.setText (horaire, 'horaire') self.setText (nomFilm, 'nom')
self.OnClick ← Click lblNomFilm.OnClick ← Click lblHoraire.OnClick ← Click

Figure 22 : Constructeur de TImageBouton

Le constructeur de TImageBouton permet d'initialiser les paramètres du bouton. Il initialise le texte qui est affiché, l'horaire, la position des labels et la position du bouton. Il initialise également les événements OnClick des deux labels ainsi que celui de l'image.

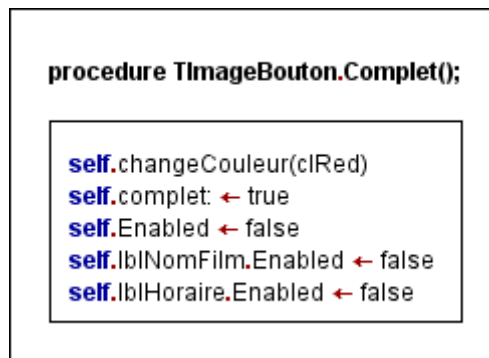


Figure 23 : procédure complet

La procédure « *complet* » change l'état du bouton, le désactive et modifie sa couleur. Il devient rouge quand la séance est complète.

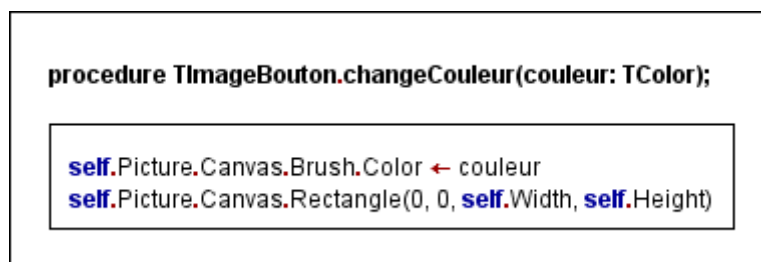


Figure 24 : procédure changeCouleur

Cette procédure permet de changer la couleur du bouton. La couleur est passée en paramètre lors de l'appel de la fonction.

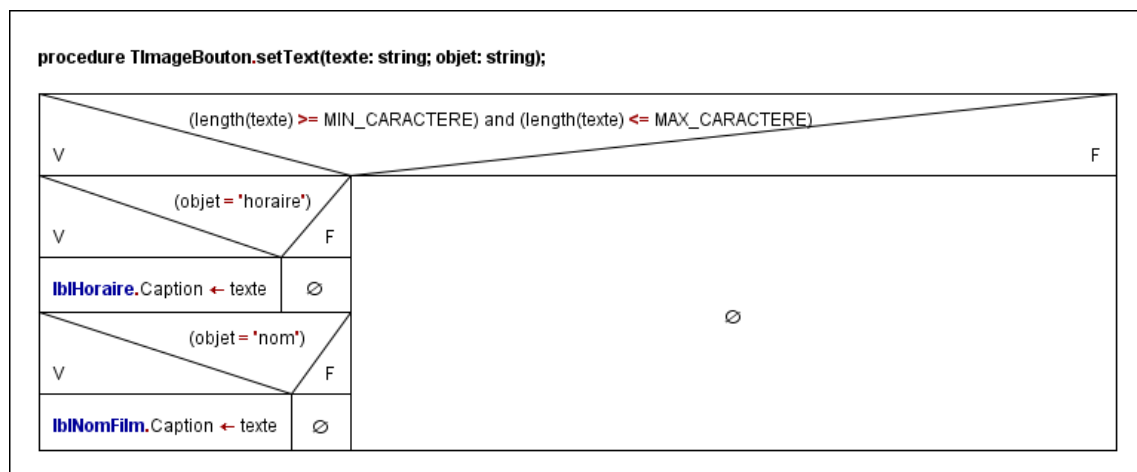


Figure 25 : Procédure setText

« *setText* » permet de vérifier si le texte est conforme aux exigences avant de l'affecter dans le bon label. Le texte à afficher dans le label est passé en paramètre. On indique également l'objet concerné. C'est-à-dire quel label est concerné, nom ou horaire.

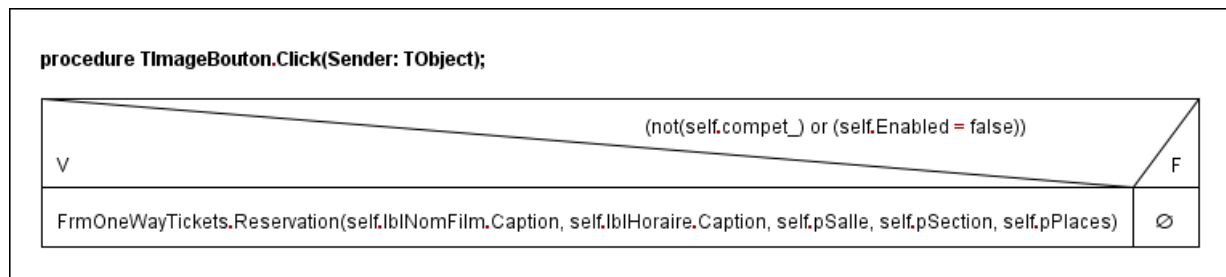


Figure 26 : Procédure de l'évènement clique

La procédure « *Click* » s'exécute lorsque l'évènement « *OnClick* » du bouton se déclenche. Elle vérifie si la propriété « *complet_* » est fausse ou que le bouton est désactivé. Si la séance n'est pas complète, la procédure de réservation démarre.

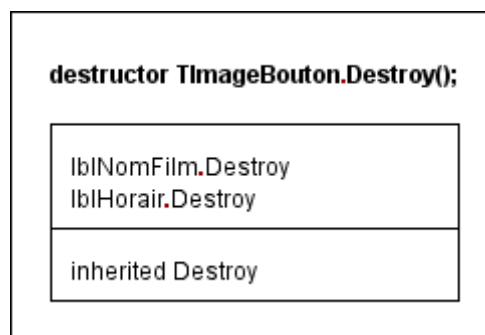


Figure 27 : Destructeur de l'objet

Le destructeur permet de détruire l'objet ainsi que ses composants.

U_FP.pas

Cette unité regroupe les variables, les constantes, les records, les procédures et les fonctions générales qui sont appelées dans la plus part des unités du programme.

Procédures et fonctions

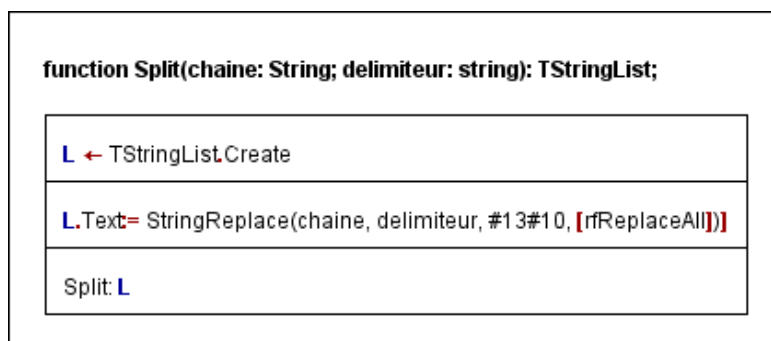


Figure 28 : Fonction Split

La fonction « *Split* » permet de couper une chaîne de caractères en segments. Elle prend en paramètre une chaîne de caractère, celle qui sera coupée, ainsi qu'un délimiteur, le caractère qui indique la fin de chaque segment. La fonction retourne un tableau qui comporte la chaîne segmentée.

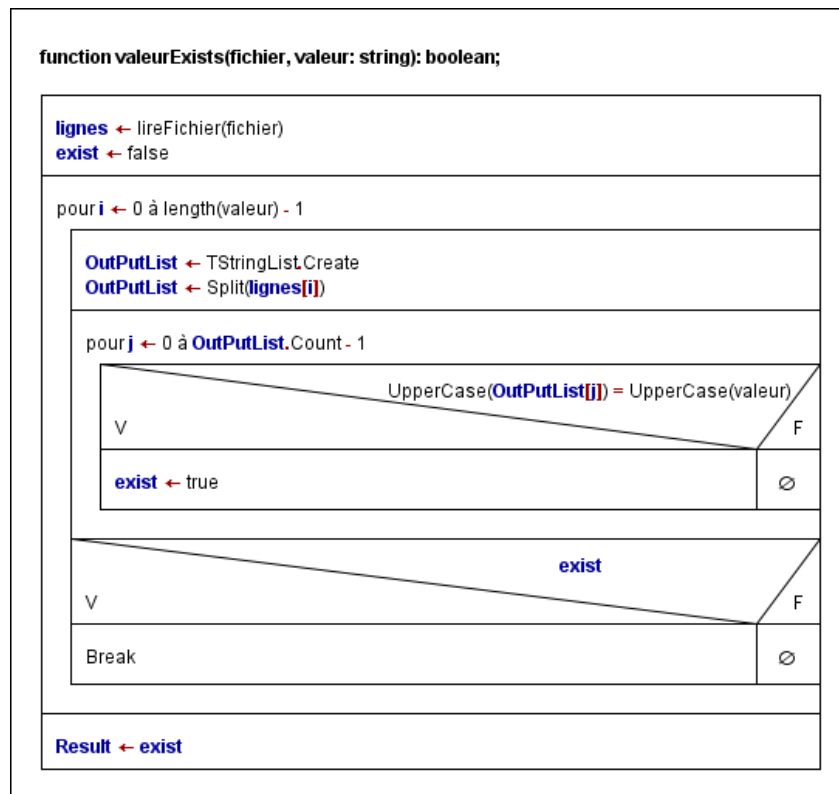


Figure 29 : Fonction valeurExiste

Cette fonction vérifie si une valeur, passée en paramètre, existe déjà dans le fichier, également passé en paramètre. Elle retourne « vrai » si la valeur existe.

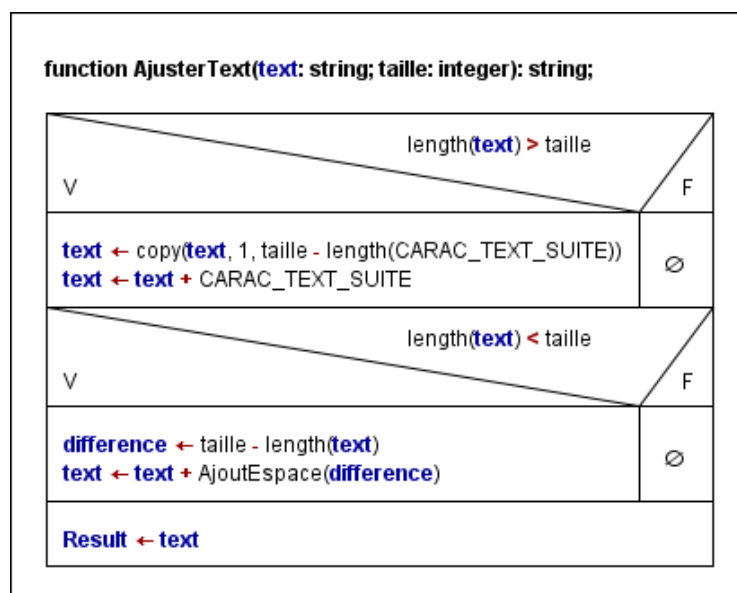


Figure 30 : fonction AjusterText

Cette fonction permet d'ajuster le nombre de caractères d'un texte passé en paramètre. Elle teste s'il est trop grand ou trop petit. Si le texte est trop grand, la fonction le coupe pour qu'il soit à la bonne taille. S'il est plus petit, la fonction rajoute le nombre d'espaces nécessaires.

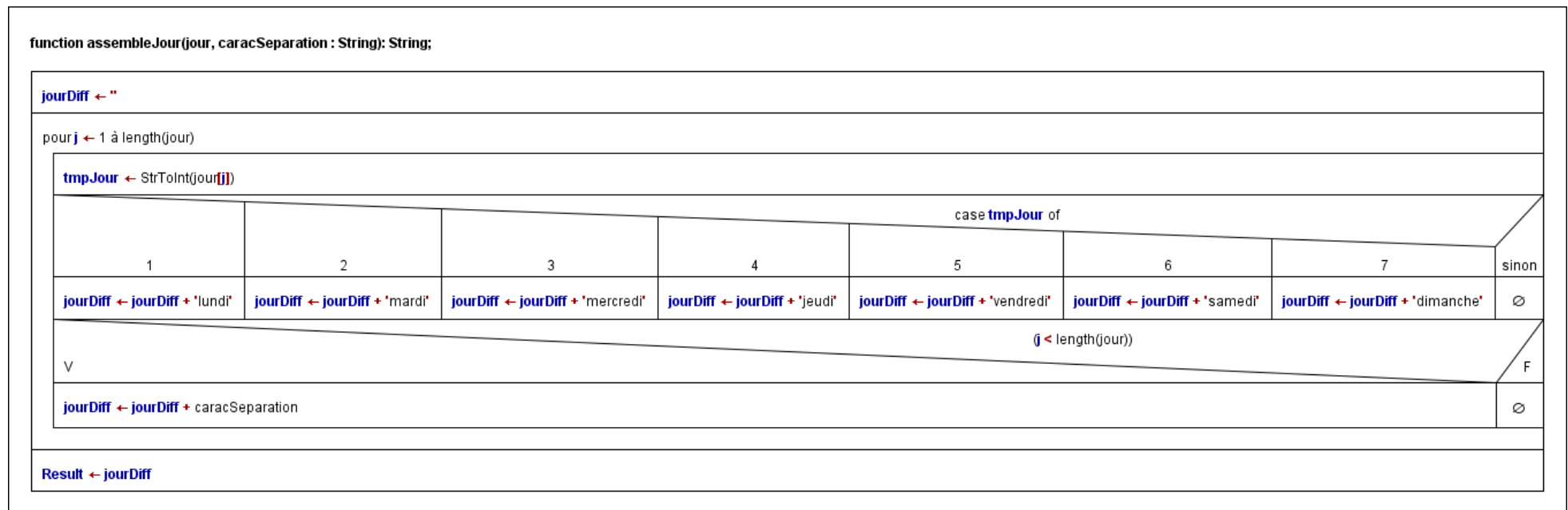
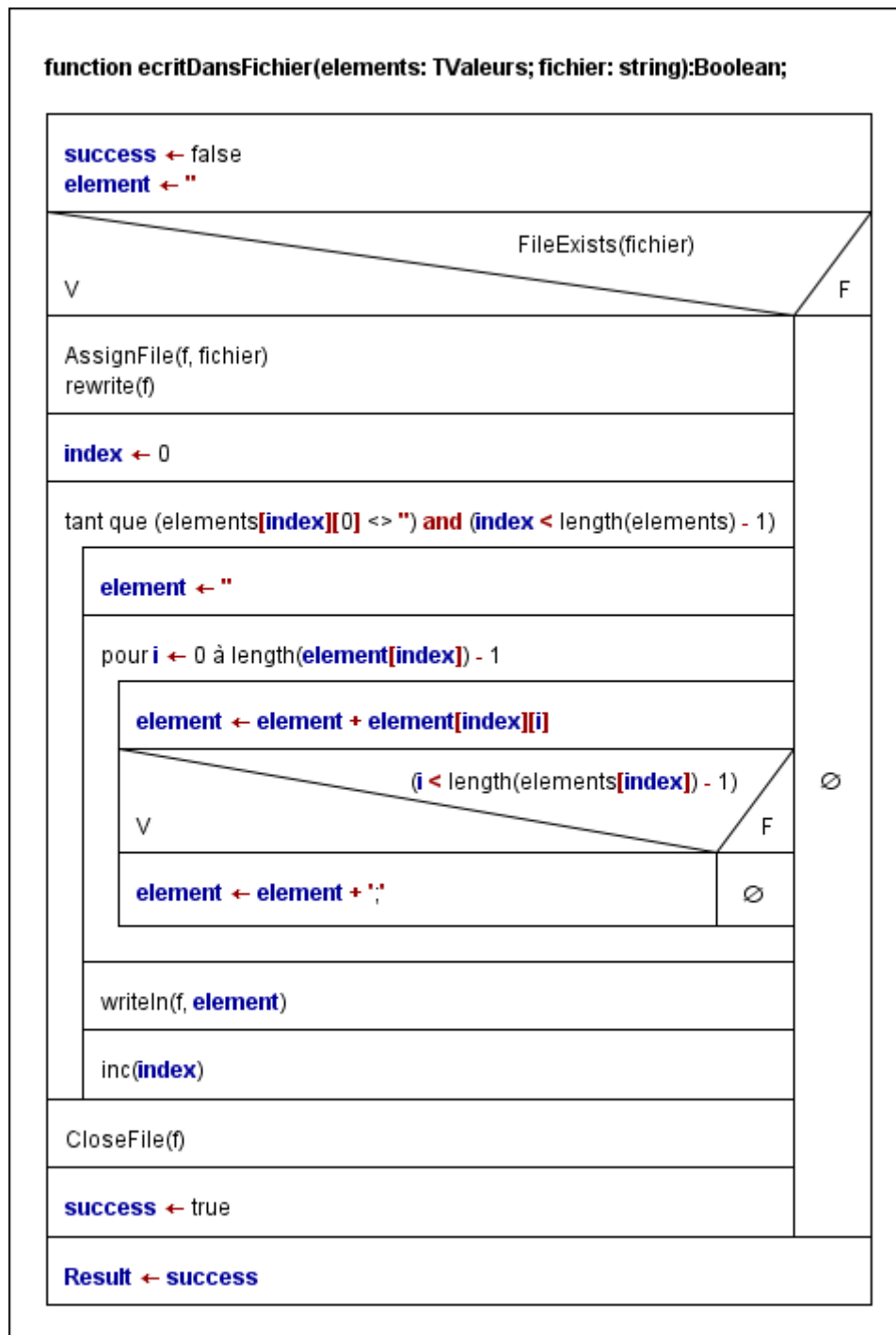


Figure 31 : Fonction assembleJour

Le jour de diffusion des séances est enregistré sous forme numérique, 1234567. Chaque nombre équivaut à un jour de la semaine. Cette fonction permet de « décoder » cette suite numérique. Elle teste chaque valeur de la chaîne et selon le numéro trouvé, elle ajoute le jour dans la chaîne. La fonction retourne une chaîne de caractères composée des jours de la semaine.

Figure 32 : fonction *ecritDansFichier*

« *ecritDansFichier* » permet d'écrire des données dans un fichier. Les données sont passées en paramètre dans un tableau à deux dimensions. Le chemin du fichier est également passé en paramètre. La fonction ouvre le fichier et supprime les données déjà existantes. Elle parcourt ensuite le tableau à deux dimensions pour assembler les données de chaque cellule qui donne une ligne dans le fichier. Elle écrit ensuite cette ligne dans le fichier. La fonction recommence cette étape autant de fois qu'il le faut. La fonction retourne une valeur vraie si le fichier a bien été écrit.

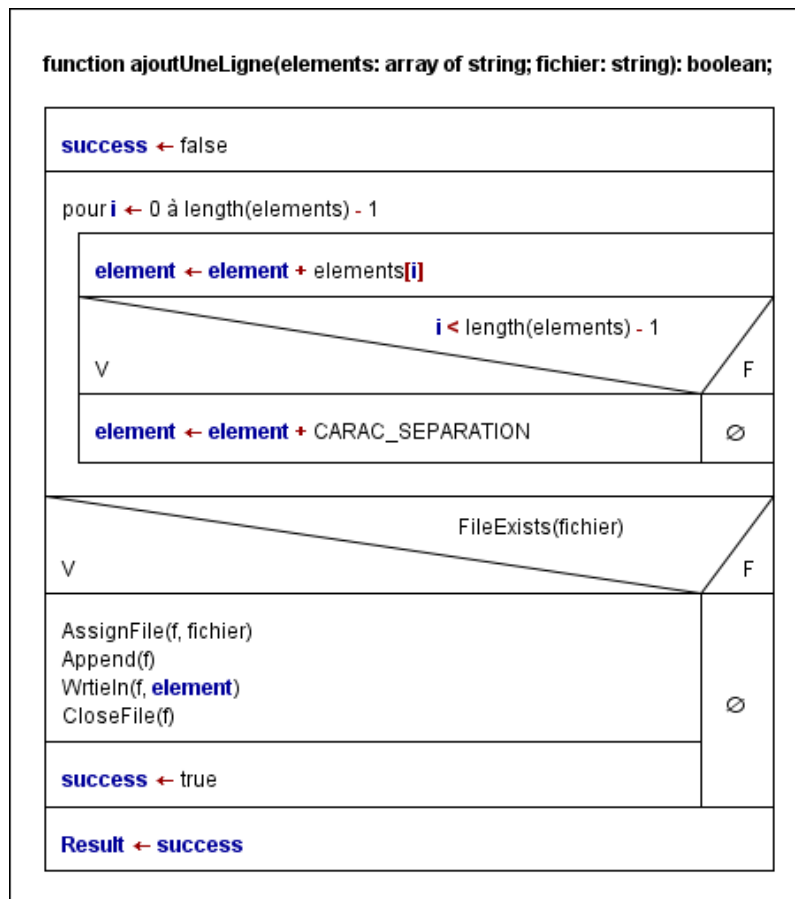


Figure 33 : Fonction ajoutUneLigne

Cette fonction permet d'ajouter une ligne dans un fichier. Les éléments à ajouter sont passés en paramètre ainsi que le chemin du fichier. En premier, la fonction rassemble les éléments du tableau pour créer une seule ligne. Ensuite, elle ouvre le fichier et écrit la ligne à la fin de celui-ci. La fonction retourne « vrai » si la ligne a bien été ajoutée.

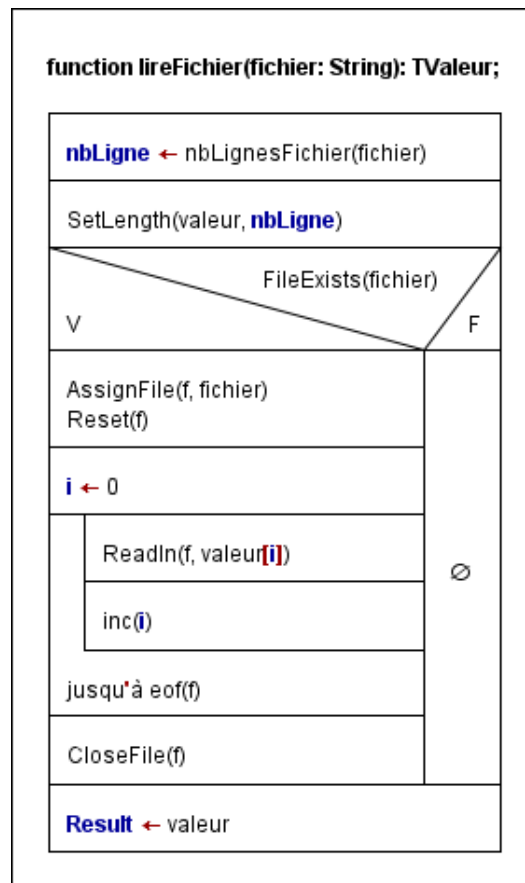


Figure 34 : fonction lireFichier

« lireFichier » permet de lire un fichier texte, dont le chemin est passé en paramètre. La fonction initialise un tableau de valeurs avec le nombre de lignes du fichier. Ensuite, le fichier est lu et les valeurs sont insérées dans le tableau. La fonction retourne le tableau de valeurs.

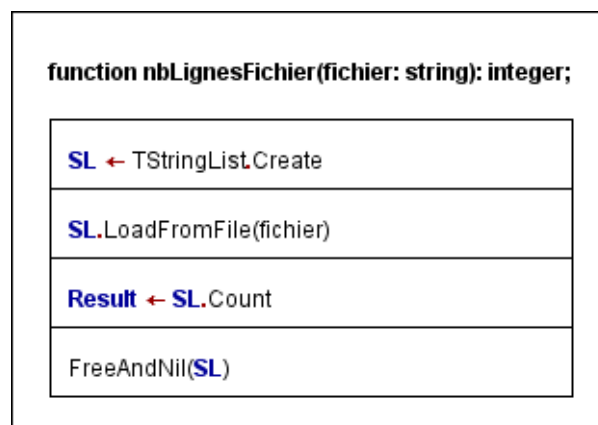


Figure 35 : fonction nbLignesFichier

Cette fonction permet de compter le nombre de lignes que possède un fichier. Elle ouvre le fichier et ajoute toutes ces lignes dans une liste. Elle retourne le nombre d'éléments que la liste comporte.

Record

Un record a été créé pour pouvoir enregistrer et traiter plus facilement les informations des séances, des films et des salles.

Ils sont structurés comme suit :

Type

```
TSeance = Record
  section : String;
  film     : String;
  salle    : String;
  jourDiff : String;
  heure1   : String;
  heure2   : String;
  heure3   : String;
  heure4   : String;
  diffuser : String;
end;
```

Type

```
TSalle = Record
  NomSalle : String;
  Places   : String;
end;
```

Type

```
TFilm = Record
  NomFilm : String;
  Duree   : String;
  Prix    : String;
  Synopsis : String;
end;
```

U_OneWayTicket.pas



Figure 36 : Fonction charge la liste des différents horaires

Cette fonction permet de séparer les différents horaires d'une séance pour obtenir celles qui doivent être affichées. Elle récupère toutes les séances qui doivent être passées dans la journée. Puis elle teste chaque horaire qui doit être mis dans la liste. Si l'horaire doit être enregistré, la fonction récupère les informations qu'elle a besoin et les met dans le tableau des séances. Une fois le traitement terminé, la fonction renvoie le tableau comportant les séances.

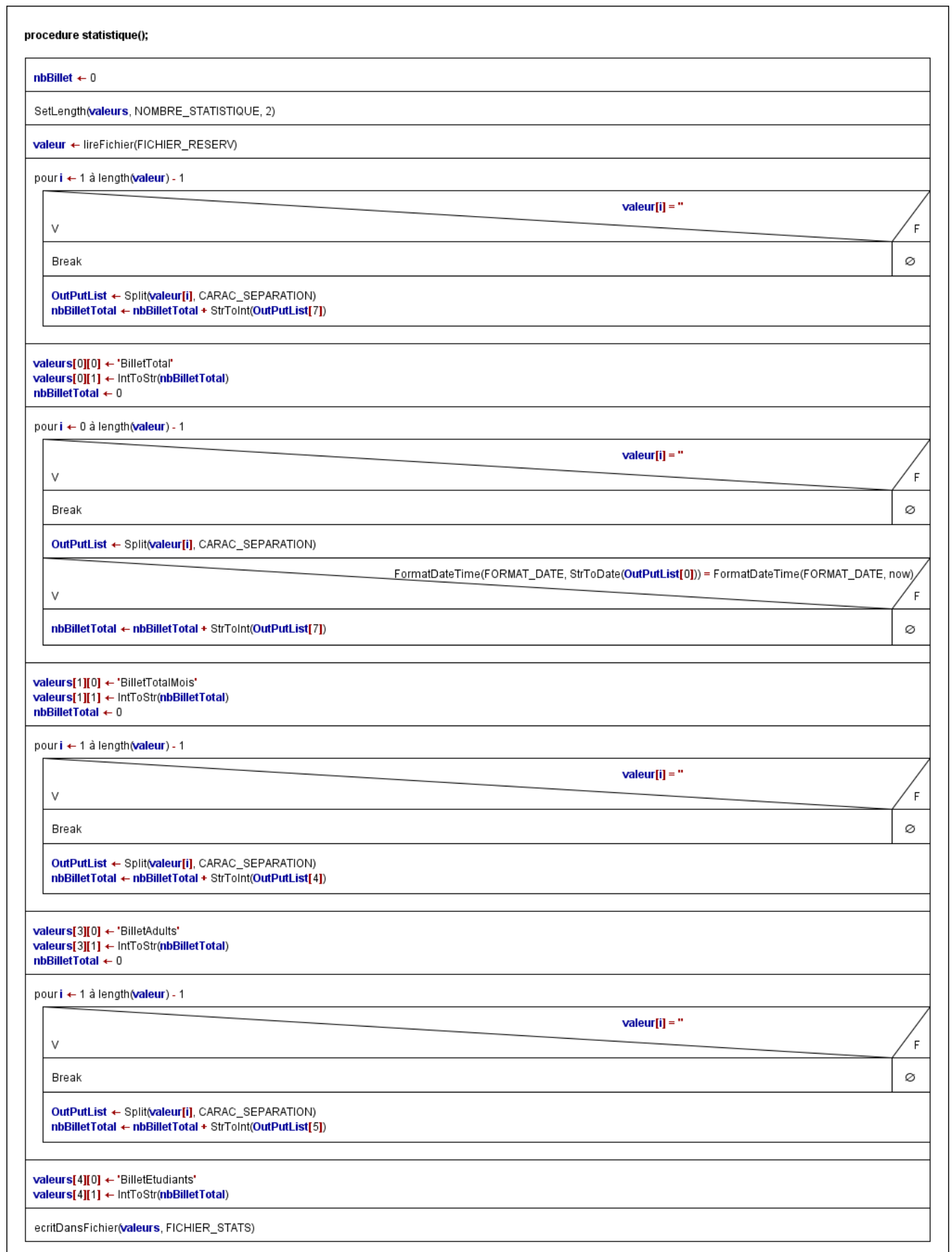


Figure 37 : procédure statistique

La procédure « *statistique* » sert à enregistrer les statistiques dans le fichier statistique. Au début, la procédure charge le fichier qui contient toutes les réservations. Puis, depuis les données qui ont été chargées, la procédure récupère les éléments nécessaires pour faire les statistiques. Elle calcule ces données puis les met dans un tableau à deux dimensions. Une fois le tableau rempli la procédure exécute la fonction « *ecritDansFichier* » avec les paramètres « *valeurs* » et « *FICHIER_STATS* ».

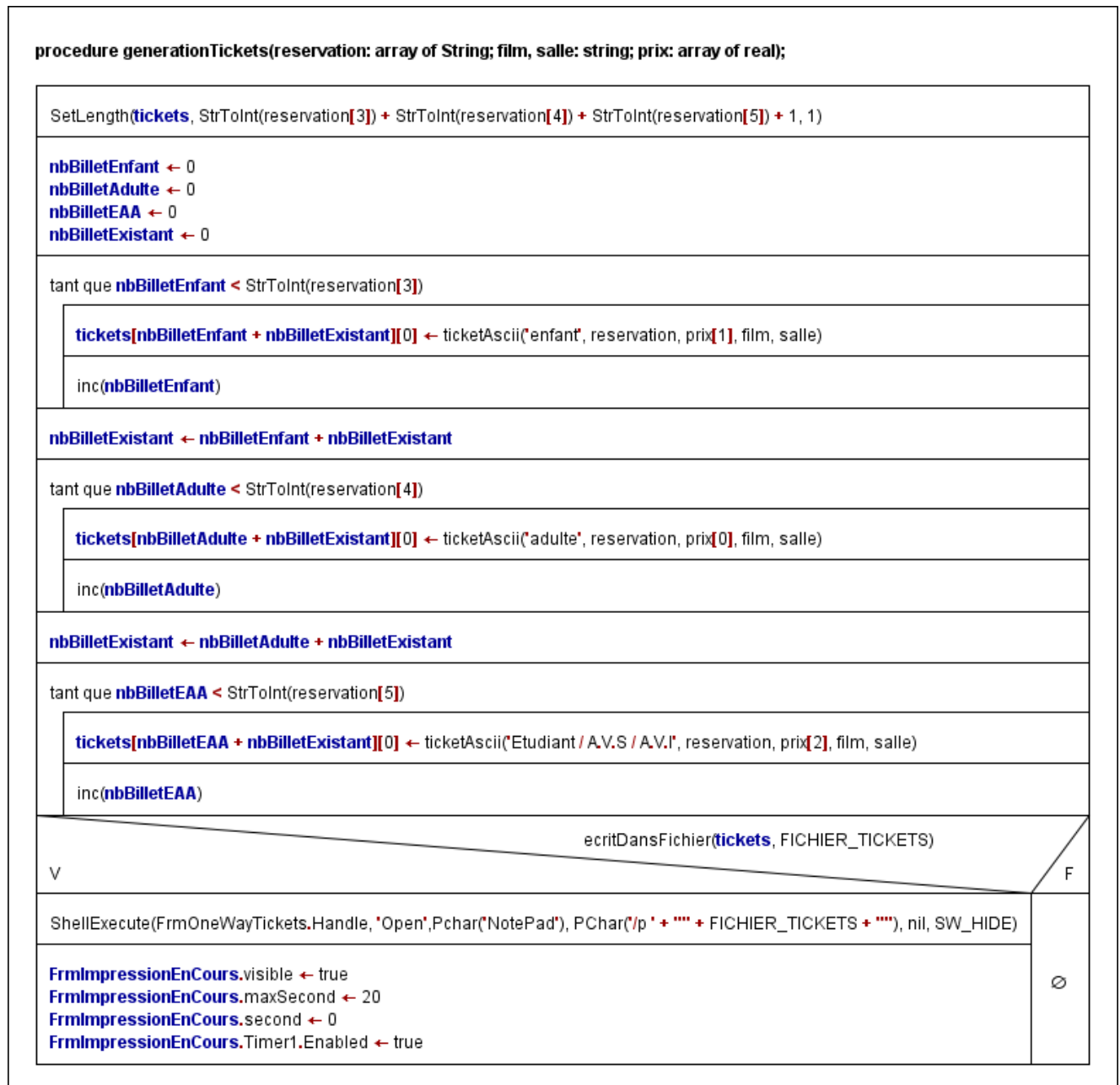


Figure 38 : Procédure de génération du ticket

« *generationTickets* » permet de générer des tickets et de les imprimer. Elle reçoit en paramètre les informations de la réservation, le nom du film, la salle et la liste des prix. La procédure passe par des boucles pour générer le bon nombre de billets. Ensuite, elle enregistre tous les billets dans un fichier texte. En dernier, la procédure ouvre le bloc note et lance l'impression.

procedure TfrmOneWayTickets.Reservation(film, horaire, salle, section : string; placesRestant: integer);

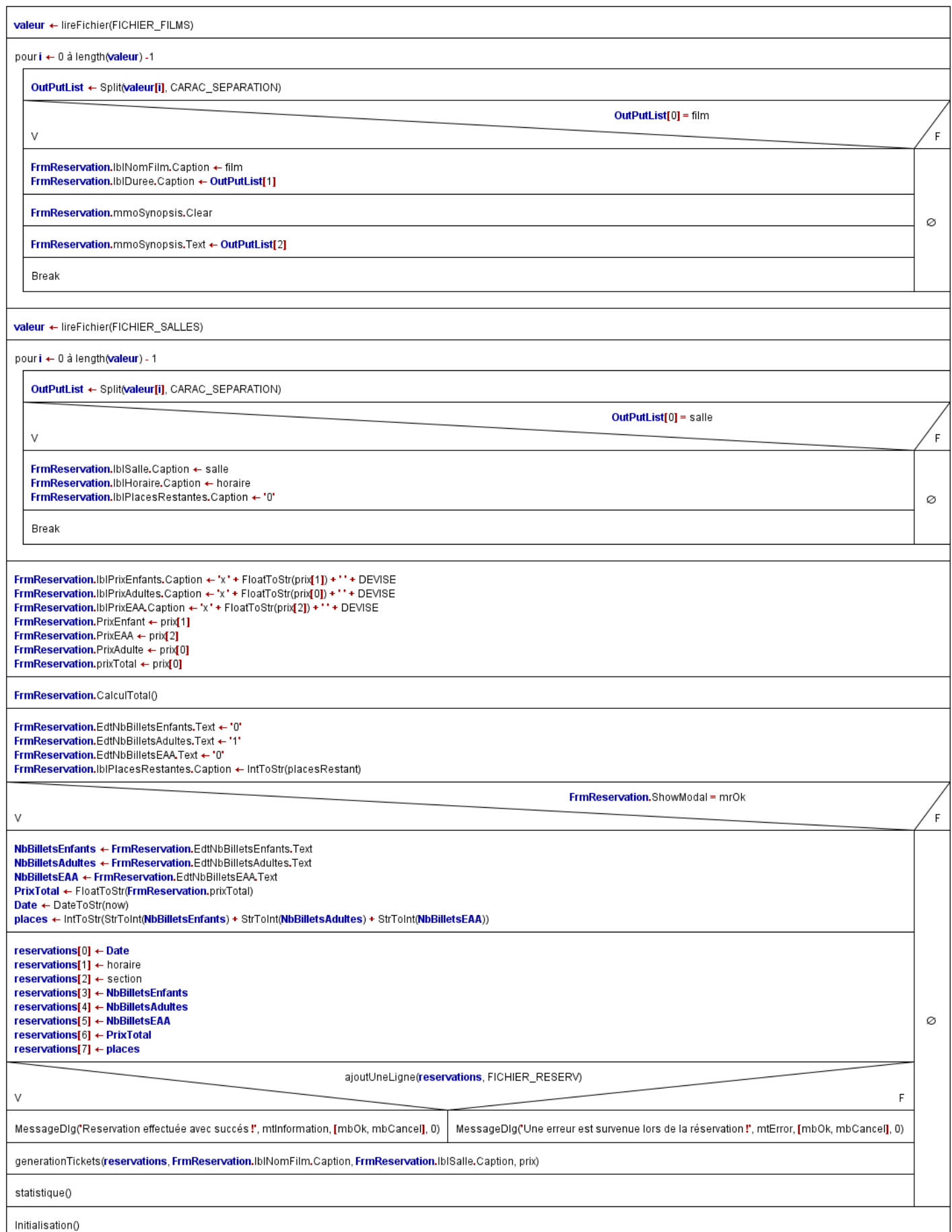


Figure 39 : procédure de réservation

La procédure « *Reservation* » prend en paramètre le nom du film, la salle, l'heure de diffusion et le nombre de places restantes. Dans un premier temps, la procédure récupère les informations du film et de la salle depuis les fichiers. Ensuite, elle traite ces données pour récupérer les valeurs qui lui sont utiles. Une fois ce traitement fait, la procédure initialise et affiche la fenêtre qui permet à l'utilisateur de faire la réservation du client.

Une fois la réservation effectuée, la procédure reçoit une valeur de la fenêtre. Si cette valeur est égale à « *mrOk* » alors la procédure récupère les données de la fenêtre de réservation. Une fois les données récupérées, elles sont enregistrées dans le fichier des réservations. Si tout se passe bien, un message s'affiche et valide le bon déroulement du processus.

En dernier, les billets sont générés, les nouvelles statistiques enregistrées et le menu utilisateur est actualisé.

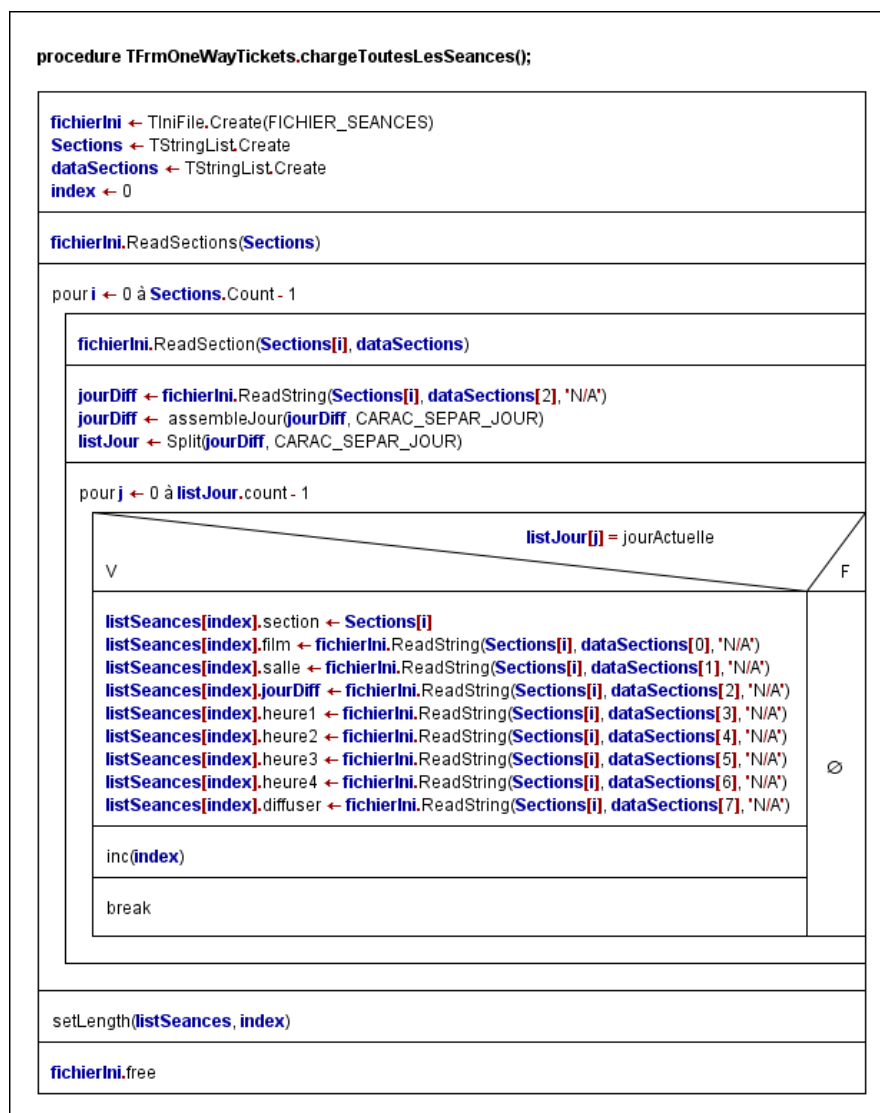


Figure 40 : Procédure chargeToutesLesSeances

« *chargeToutesLesSeances* » permet de charger les séances depuis le fichier des séances. Il charge le fichier et teste section par section, pour contrôler que le jour de la diffusion est bien le jour actuel. Si c'est le cas, la séance est enregistrée dans le tableau des séances. Une fois les bonnes séances récupérées le fichier est fermé.

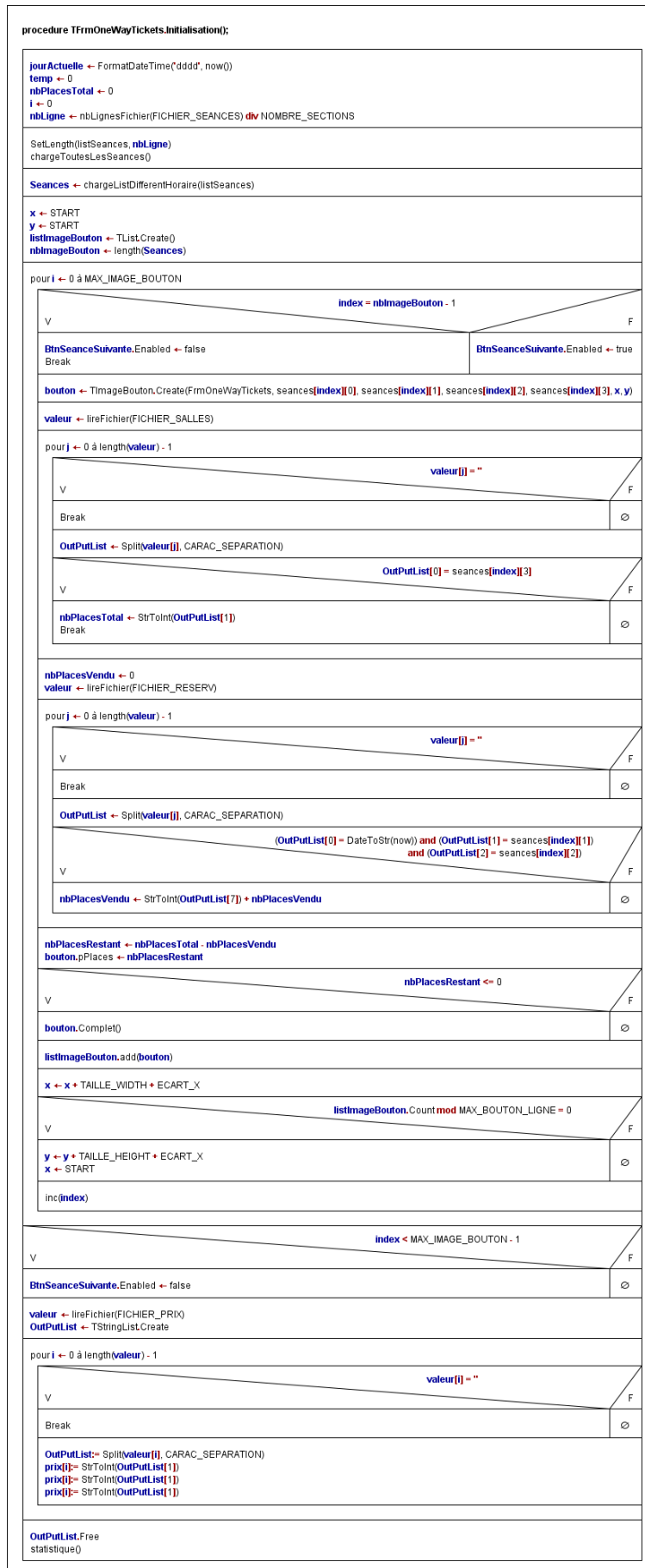


Figure 41 : procedure d'initialisation

Cette procédure est la plus importante du programme. C'est elle qui permet de générer les boutons des séances. Dans un premier temps, la procédure charge toutes les séances avec leurs différents horaires. Une fois les horaires récupérés, la procédure récupère le nombre de places dans la salle, ainsi que le nombre de places vendues pour la séance. Avec ces informations, elle calcule le nombre de places restantes. Si la séance est complète, le bouton devient rouge et est inutilisable. Les boutons générés sont enregistrés dans une liste et sont affichés sur le menu utilisateur. Une fois les boutons générés, la procédure charge le prix des billets depuis le fichier des prix. Pour finir, les statistiques sont mises à jour.

Test

Plan de test

N° Test	Auteur	Scénarios	Résultats attendus
T0001	Devaud Alan	Sur le menu utilisateur, l'utilisateur clique sur une séance rouge	Rien ne se passe
T0002	Devaud Alan	Sur le menu utilisateur, l'utilisateur clique sur le bouton « Afficher séances suivantes »	Affiche les X prochaines séances
T0003	Devaud Alan	Sur le menu utilisateur, l'utilisateur est sur les séances actuelles et il appuie sur le bouton « afficher séances précédentes »	Rien ne doit se passer
T0004	Devaud Alan	Sur une réservation, l'utilisateur valide alors que les inputs sont vides	Le bouton est désactivé
T0005	Devaud Alan	Sur une réservation, l'utilisateur clique sur le bouton « - » alors que la valeur de l'input vaut « 0 »	La valeur reste « 0 »
T0006	Devaud Alan	Sur une réservation, l'utilisateur essaye de rentrer lui-même le nombre de billets	Rien ne se passe
T0007	Devaud Alan	Au login administrateur, l'utilisateur rentre un mauvais mot de passe	Un message d'erreur s'affiche
T0008	Devaud Alan	Sur le menu administration, l'administrateur clique sur déconnexion	La déconnexion s'effectue, la fenêtre d'administration disparaît et le menu utilisateur s'affiche

T0009	Devaud Alan	Sur la fenêtre de gestion, l'utilisateur clique sur modifier alors qu'aucun élément n'est sélectionné	Un message d'erreur s'affiche (Bouton non cliquable)
T0010	Devaud Alan	Sur la fenêtre de gestion, aucun élément n'est sélectionné	Les boutons « Modifier » et « Supprimer » sont désactivés
T0011	Devaud Alan	Sur la fenêtre de gestion, l'utilisateur sélectionne un élément et clique sur le bouton « Supprimer »	Un message de confirmation s'affiche
T0012	Devaud Alan	Sur la fenêtre de gestion, lors de la modification, tous les inputs sont vides et l'utilisateur clique sur le bouton « Valider »	Rien ne se passe
T0013	Devaud Alan	Sur la fenêtre de gestion, lors de la modification, l'administrateur a modifié des champs et appuie sur annuler	Un message de confirmation s'affiche
T0014	Devaud Alan	Sur la fenêtre d'ajout d'un élément, aucun champ n'est rempli	Le bouton « Ajouter » est désactivé
T0015	Devaud Alan	Sur la fenêtre d'ajout d'un élément, tous les champs sont remplis	Le bouton « Ajouter » est activé
T0016	Devaud Alan	Sur la fenêtre pour créer une séance, l'administrateur clique sur une des deux listes déroulantes	La liste s'affiche
T0017	Devaud Alan	Sur la fenêtre pour créer une séance, aucun jour de diffusion n'est sélectionnée	Le bouton « Valider la séance » est désactivé
T0018	Devaud Alan	Sur la fenêtre pour créer une séance, aucune heure n'a été assignée	Le bouton « Valider la séance » est désactivé

T0019	Devaud Alan	Sur la fenêtre pour créer une séance, deux heures sont trop proches (ex. ; Heure 1 : 10:00, Heure 2 : 11:00, le film dure 2 heures) et l'administrateur clique sur le bouton « Valider la séance »	Message d'erreur : « Certaines heures sont trop proches »
T0020	Devaud Alan	Sur la fenêtre pour créer une séance, l'administrateur clique sur le bouton « Annuler »	Message de confirmation
T0021	Devaud Alan	Sur la fenêtre de réservation en avance, l'utilisateur ne sélectionne aucun film	Les dates de séances, les informations sur les séances et les films sont effacés
T0022	Devaud Alan	Sur la fenêtre de réservation en avance, l'utilisateur appuie sur le bouton « Annuler la réservation »	Message de confirmation
T0023	Devaud Alan	Sur la fenêtre de réservation en avance, l'utilisateur appuie sur le bouton « Valider la réservation » alors qu'aucun champ n'est rempli	Rien ne se passe (Le bouton est désactivé)

Rapport de test

Date	Testeur	N° Test	Résultats obtenus	OK/KO	Fiches Anomalies
13.05.2015	Camuglia Lucien	T0001	Rien ne se passe	OK	-
13.05.2015	Camuglia Lucien	T0002	Les prochaines séances sont affichées	OK	-
13.05.2015	Camuglia Lucien	T0003	Rien ne se passe	OK	-
13.05.2015	Camuglia Lucien	T0004	Le bouton est désactivé	OK	-
13.05.2015	Camuglia Lucien	T0005	La valeur est « 0 »	OK	-
13.05.2015	Camuglia Lucien	T0006	La saisie ne fonctionne pas	OK	-
13.05.2015	Camuglia Lucien	T0007	Le message s'affiche	OK	-
13.05.2015	Camuglia Lucien	T0008	La fenêtre disparaît et le menu utilisateur s'affiche	OK	-
13.05.2015	Camuglia Lucien	T0009	Impossible de ne pas sélectionner un élément	OK	-
13.05.2015	Camuglia Lucien	T0010	Impossible de ne pas sélectionner un élément	OK	-
13.05.2015	Camuglia Lucien	T0011	Un message d'avertissement apparaît	OK	-
13.05.2015	Camuglia Lucien	T0012	Impossible de valider si un champ est vide	OK	-
13.05.2015	Camuglia Lucien	T0013	Un message de confirmation apparaît	OK	-

13.05.2015	Camuglia Lucien	T0014	Si un input est vide le bouton est désactivé	OK	-
13.05.2015	Camuglia Lucien	T0015	Le bouton est activé	OK	-
13.05.2015	Camuglia Lucien	T0016	La liste s'affiche	OK	-
13.05.2015	Camuglia Lucien	T0017	Le bouton est désactivé	OK	-
13.05.2015	Camuglia Lucien	T0018	Le bouton est désactivé	OK	-
13.05.2015	Camuglia Lucien	T0019	La séance se crée	KO	FA0001
13.05.2015	Camuglia Lucien	T0020	Aucun message ne s'affiche	KO	FA0002
13.05.2015	Camuglia Lucien	T0021	Le bouton de réservation en avance est désactivé	KO	FA0003
13.05.2015	Camuglia Lucien	T0022	Le bouton de réservation en avance est désactivé	KO	FA0004
13.05.2015	Camuglia Lucien	T0023	Le bouton de réservation en avance est désactivé	KO	FA0005

Conclusion

Bilan

Durant ce projet, j'ai pu voir comment se déroulait un travail avec des personnes extérieures de l'école. J'ai trouvé cela très enrichissant et cela nous permet de nous mettre dans un cadre professionnel avec des objectifs et des exigences qui ne sont pas forcément les nôtres.

Avec ce projet, j'ai pu constater que faire un programme en suivant un cahier des charges n'était pas aussi facile que ce que l'on pense. Cela nous met des contraintes que l'on doit affronter et nous demande de trouver les solutions adéquates pour que le client trouve ce qu'il souhaite et qu'il soit satisfait.

Pendant le développement, j'ai eu un gros problème qui m'a pris beaucoup de temps pour le résoudre. J'ai également constaté que la gestion de fichiers avec Delphi est, à mon avis, plutôt complexe pour un projet de cette envergure. Malgré ces petits désagréments, j'ai eu de la motivation et du plaisir pour faire mon programme et satisfaire au maximum mes experts et mon enseignant.

Pendant mon projet, j'ai constaté que faire une documentation complète est un exercice long et fastidieux. Mais une fois que la documentation est faite, elle nous permet de nous retrouver dans notre projet et à savoir où l'on en est dans la progression de notre travail.

En conclusion, le projet et la documentation ont été une tâche importante dans cette formation qui nous permet de vraiment nous rendre compte à quel point les connaissances que l'on acquiert durant la formation sont conséquentes.

Améliorations envisageables

- Sur le menu utilisateur, trier les séances par ordres croissant des horaires.
- Dans la gestion, faire un meilleur affichage des informations.
- Pouvoir gérer les prix.
- Afficher un graphique pour les statistiques.
- Lors de la suppression d'un film ou d'une salle, supprimer tout ce qui les concernent.

Comparaison des plannings

Planning prévisionnel

	28.04.2015	29.04.2015	30.04.2015	05.05.2015	06.05.2015	07.05.2015	08.05.2015	12.05.2015	13.05.2015	15.05.2015
Création et structure de la documentation										
Liste des fonctionnalités disponibles										
Schématisation de l'interface										
Description des mesures de sécurité										
Description des données manipulées										
Création du plan de tests et rapport de tests										
Création du schéma global de l'application										
Création des structogrammes										
Création de l'interface										
Implémentation des structogrammes										
Exécution des tests										
Mise à jour de la documentation										

Planning réel

	28.04.2015	29.04.2015	30.04.2015	05.05.2015	06.05.2015	07.05.2015	08.05.2015	12.05.2015	13.05.2015	15.05.2015
Création et structure de la documentation										
Liste des fonctionnalités disponibles										
Schématisation de l'interface										
Description des mesures de sécurité										
Description des données manipulées										
Création du plan de tests et rapport de tests										
Création du schéma global de l'application										
Création des structogrammes										
Création de l'interface										
Implémentation des structogrammes										
Exécution des tests										
Mise à jour de la documentation										

Bibliographie

- Code de la fonction « Split » trouvée sur internet : <http://www.developpez.net/forums/d639683/environnements-developpement/delphi/debutant/fonction-split-delphi/>
- Structorizer, logiciel de structogramme trouvé sur internet : <http://structorizer.fisch.lu/>
- Install Creator 2, logiciel de création d'installateur trouvé sur internet : <http://www.clickteam.com/fr/install-creator-2>
- Certains bouts de codes trouvés dans mes cours
- Logo trouvé sur internet : <http://www.iconarchive.com/show/pretty-office-3-icons-by-custom-icon-design/tickets-icon.html>