

II - Présentation du travail réalisé

1. Présentation des objectifs

L'objectif de ce stage était de réaliser plusieurs missions qui ont pour but de contribuer à une meilleure gestion du restaurant. Le Pont d'Asie possède depuis plusieurs années la même caisse enregistreuse, c'est pourquoi j'ai eu l'idée d'en créer une nouvelle avec les connaissances que j'ai acquises durant l'année. De plus, la caisse enregistreuse actuelle n'est pas à jour, c'est-à-dire que les prix de certains articles n'ont pas tous été mis à jour et certains articles qui n'existent plus sont toujours là.

A l'aide de VBA sur Excel, j'ai alors décidé de réaliser un début de caisse enregistreuse, avec les fonctions principales en priorité et pourquoi pas continuer à la développer dans le futur.

Au cours de ces dix semaines, j'ai donc travaillé comme serveur mais aussi comme développeur VBA. Pour cela j'ai utilisé mon ordinateur et le restaurant m'a prêté les premières semaines une tablette afin de prendre en notes les commandes réalisées avant la création de mon interface.

Ces données récoltées m'ont servi plus tard à réaliser mon analyse des données, qui est une des missions supplémentaires que je me suis fixé en accord avec ma tutrice.

Ces analyses permettront notamment de déterminer les plats les moins/plus commandés, les jours qui rapportent le plus...

Voici les différentes missions fixées dans la convention :

Réalisation d'une caisse enregistreuse de commandes dans un Fichier Excel

Amélioration du Fichier Excel en mettant en place une application interactive VBA

Gestion des Stocks des produits du restaurant

Analyse des données :

- Les meilleures ventes (pour chaque catégorie d'articles)
- Les articles les -/+ rentables

Proposition d'amélioration suite aux analyses (articles à retirer, changement des prix..etc)

La gestion des stocks étant déjà réalisée par la patronne, c'est une mission que j'ai mis de côté avant la réalisation de la caisse enregistreuse et l'analyse des données, que j'ai jugé bien plus pertinente.

2. Les outils

Durant le stage, j'avais à ma disposition mon ordinateur personnel et une tablette prêtée par le restaurant. Pour réaliser mes missions, j'ai utilisé un seul logiciel durant l'entièreté du stage : Microsoft Excel.

Ce logiciel permet de réaliser des graphiques qui m'ont été nécessaires pour mon analyse des données, et l'interface Visual Basic for Applications, intégrée dans le logiciel, m'a permis de réaliser ma caisse enregistreuse et enrichir ma base de données.



Figure 1: Logo Microsoft Excel



Figure 2: Logo Visual Basic for Applications

3. Le travail réalisé

3.1 Définition du plan de travail

Afin de mener à bien le projet, j'ai décidé le premier jour de me concerter avec ma tutrice sur les éléments de la caisse enregistreuse qu'elle souhaite intégrer, bien sûr, des modifications peuvent aboutir au fur et à mesure mais voici les éléments qui constitueraient mon application et quelques contraintes :

- Boutons de tous les articles
- Lorsqu'un bouton est appuyé, il s'affiche dans une fenêtre avec sa quantité, son nom d'article et son prix
- Le prix s'adapte à la quantité
- Affichage du total de la note, du reste à payer, du montant dû
- Boutons de paiement en carte, espèce, ticket, chèque
- Deux types de commandes : sur place et à emporter
- Une commande est considérée comme terminée lorsque le paiement total a été réalisé

Et si le temps le permet :

- Créer un reçu de commande
- Permettre l'ajout de couverts par commande sur place

Après quoi, je suis libre d'ajouter les fonctionnalités que je souhaite comme notamment réaliser un reçu si j'ai le temps. La priorité reste la création de la caisse afin de stocker les données et connaître les produits populaires du restaurant.

3.2 Le menu « InterfaceNote »

J'ai donc d'abord créé un premier UserForm que j'ai appelé « InterfaceNote » et j'ai cherché à réaliser une interface qui me plaisait esthétiquement mais qui devait aussi plaire à la patronne. Heureusement pour moi, elle n'était pas difficile sur les propositions, ce qui a rendu le travail plus simple. Je suis parti sur un modèle similaire à la caisse enregistreuse actuelle du restaurant.

J'ai également eu recours à plusieurs changements concernant les contrôles à utiliser, c'est-à-dire de choisir entre soit des TextBox, des ListBox ou des ComboBox pour faciliter au mieux la suite de la réalisation de la caisse.

The screenshot shows a software interface for a cash register titled "LE PONT D'ASIE". At the top, there's a title bar "Caisse Enregistreuse - Le Pont d'Asie". Below it, a menu bar has tabs for "Entrées", "Plats", "Desserts", "Boissons", and "Menus & Autres". The "Entrées" tab is selected, displaying a grid of food items categorized into "Nems", "Hors d'oeuvres", "Salades", "Vapeurs", and "Soupes". To the right of the menu is a payment section. It includes buttons for "Supprimer", "Remise", "Remise Note", and "Offert". Below these is a numeric keypad with digits 0-9, a decimal point, and a checkmark. To the right of the keypad are buttons for payment methods: "CB", "Espèces", "Ticket", and "Chèque". Above the keypad are fields for "Dû", "Reste", and "Total".

Figure 3 : Capture Ecran de l'Interface de la caisse enregistreuse.

Voici l'UserForm qui contient énormément d'informations, tout d'abord nous avons les différents boutons de tous les articles de la carte répartis selon leurs catégories : Entrées, Plats, Desserts, Boissons et Menus & Autres.

A droite, il y a tout ce qui concerne le paiement de la commande, un clavier de chiffres, puis plus haut des boutons pour le type de paiement : Carte Bancaire, Espèce, Ticket, Chèque

D'autres boutons supplémentaires que j'ai personnellement ajouté pour faire une remise en % sur le prix d'un article consommé ou sur la note totale. On peut également supprimer un article avec le bouton Supprimer.

Toutes les informations concernant les articles et le paiement sont affichées dans des ListBox qui seront reliées entre elles. Les trois du haut se nomment respectivement ListBoxQuantite, ListBoxArticle et ListBoxPrix.

Tandis que celles concernant le paiement se nomment ListBoxQuantitePaiement, ListBoxMontant, ListBoxTotalPaiement, ListBoxMoyenPaiement

Après validation avec ma tutrice, j'ai enfin pu commencer à coder chaque bouton, bien sûr, il a fallu les renommer un par un, ce qui a pris beaucoup de temps !

3.2.1 Les Boutons d'articles

```
''' NEM '''

'Private Sub BoutonNem_Click()

'  ' 1 article '
'  If TextBoxValeur = "" Then
'      With ListBoxArticle
'          .AddItem "Nem"
'      End With
'      With ListBoxQuantite
'          .AddItem "1"
'      End With
'      With ListBoxPrix
'          .AddItem Format(4.5, "0.00")
'      End With
'  End If
'
'  ' Plusieurs articles '
'  If Val(TextBoxValeur) >= 1 Then
'      With ListBoxArticle
'          .AddItem "Nem"
'      End With
'      With ListBoxQuantite
'          .AddItem TextBoxValeur
'      End With
'      With ListBoxPrix
'          .AddItem (Val(TextBoxValeur) * 4.5)
'      End With
'  End If
'
'  TextBoxValeur = ""
'
'End Sub
```

Voici le code sur lequel j'étais parti pour chaque bouton, celui-ci concerne l'article « Nem ».

Bien qu'efficace, ce code prend énormément de place et modifier celui-ci pour plus de 80 articles demande énormément de temps.

C'est pourquoi il était nécessaire de créer des modules, pour faire appel à des fonctions et faciliter la lecture du code.

Figure 4 : Capture Ecran du premier code pour le bouton de l'article Nem

Le module « Articles » a donc vu le jour et j'ai créé deux fonctions :
AjouterArticle et SupprimerArticle

```
''' Ajouter un Article dans la caisse '''
Sub ajouterarticle(listeArticle As MSForms.ListBox, _
    listeQuantite As MSForms.ListBox, listePrix As MSForms.ListBox, _
    nomArticle As String, quantite As Double, prix As Double)
    With listeArticle
        .AddItem nomArticle
    End With
    With listeQuantite
        .AddItem quantite
    End With
    With listePrix
        .AddItem Format(prix, "0.00")
    End With
End Sub
```

Figure 5 : Capture Ecran de la fonction ajouterarticle

Ce code permet d'ajouter le nom, le prix et la quantité souhaitée lorsqu'on fait appel à cette fonction depuis un bouton, il faut penser à bien écrire chaque argument.

```
''' NEMS '''
Private Sub BoutonNem_Click()
    If TextBoxValeur = "" Then
        Call ajouterarticle(ListBoxArticle, ListBoxQuantite, ListBoxPrix, "Nem", 1, 4.5)

    ElseIf Val(TextBoxValeur) >= 1 Then
        Dim quantite As Double
        quantite = Val(TextBoxValeur)
        Call ajouterarticle(ListBoxArticle, ListBoxQuantite, ListBoxPrix, "Nem", quantite, quantite * 4.5)
    End If

    Call Paiement.prixTotal
    Call Paiement.PrixReste
    Call MAJ.Mise_à_jour_article
    TextBoxValeur = ""
End Sub
```

Figure 6 : Capture Ecran du bouton Nem mis à jour, qui fait appel à la fonction ajouterarticle.

Comme on peut le voir sur cette capture d'écran, la structure est plus épurée, il sera plus simple de configurer chaque bouton pour les autres articles. Les seuls changements nécessaires sont le nom du bouton à changer, le nom du produit et son prix. Nous ne prendrons pas en compte les appels à la fin du code, que j'expliquerai plus tard.

Pour le bouton supprimer, il faut récupérer la ligne d'une des trois listes d'articles et supprimer l'index correspondant.

3.2.2 Les boutons de TextBox

Maintenant que les boutons des articles sont tous fait, je me suis attardé sur d'autres boutons et fonctionnalités, tout d'abord, les boutons qui permettent d'écrire des valeurs dans la TextBox nommée TextBoxValeur, celle-ci sera utile pour rentrer la quantité d'articles ou le montant d'un paiement :

```
' ' Bouton 1 ' '

Private Sub Bouton1_Click()
    If TextBoxValeur = "" Then
        TextBoxValeur.Value = "1"
    Else
        TextBoxValeur.Text = TextBoxValeur + "1"
    End If
End Sub
```

Figure 7 : Capture Ecran du bouton chiffre 1

Il est vrai qu'on pourrait directement écrire dans la TextBox mais je voulais garder cette interface similaire à une caisse enregistreuse. Suite à cela, j'ai codé les autres boutons de chiffres, la virgule pour les nombres décimaux et le x que j'expliquerai plus tard.

3.2.3 Les moyens de paiement

Concernant les moyens de paiement d'une commande, il fallait d'abord que je configure les boutons des différents moyens de paiement, en carte bancaire, espèces, ticket et chèque.

Voici le code qui est le même pour tous, sauf pour les tickets, il suffit de changer le nom du bouton et la valeur entre guillemets.

```
Private Sub BoutonCB_Click()

    Dim total As Double
    Dim montant As Double
    Dim valeur As Double
    Dim reste As Double

    If TextBoxValeur = "" Then
        reste = TextBoxReste
        Call Payer(ListBoxQuantitePaiement, ListBoxMontant, ListBoxTotalPaiement, _
        ListBoxMoyenPaiement, 1, reste, reste, "Carte Bancaire")
        Call Paiement.PrixReste
        Call Paiement.PrixDu

    ElseIf Len(TextBoxValeur) > 0 Then
        TextBoxReste.Value = "0,00"
        montant = TextBoxValeur.Value
        Call Payer(ListBoxQuantitePaiement, ListBoxMontant, ListBoxTotalPaiement, _
        ListBoxMoyenPaiement, 1, montant, montant, "Carte Bancaire")
        Call Paiement.PrixReste
        Call Paiement.PrixDu
    End If

    Call MAJ.Mise_à_jour_paiement

    TextBoxValeur = ""

End Sub
```

Figure 8 : Capture Ecran du bouton de paiement en carte bancaire

En clair, ce code permet de détecter s'il y a une valeur dans TextBoxValeur. Si cette valeur est vide, alors le paiement se réalise en une seule fois par carte bancaire. Si par exemple une valeur est inscrite, alors le code prendra en compte la valeur et l'affichera dans les listes déroulantes associées au paiement.

Pour exemple, je rentre la valeur 12 dans ma boîte de texte, puis j'appuie sur le bouton Carte Bancaire, ça affichera alors un paiement en carte bancaire d'une valeur de 12 euros.

En ce qui concerne le paiement par ticket restaurant, la tâche est plus difficile car il faut une valeur dans TextBoxValeur de la forme « nombre x nombre » par exemple 4x3.

Il faut ensuite remplacer inscrire la valeur 4 dans la quantité de paiement, 3 dans le montant du paiement, et le total du paiement qui est 4 fois 3. Après de nombreux essais, je n'ai finalement pas réussi à coder cette partie là et j'ai préféré me concentrer sur des tâches plus importantes.

Il est bon de préciser que les boutons de moyens de paiement font aussi appel à des fonctions présentes dans un autre module « Paiement ».

```
Sub Payer(listeQuantitePaiement As MSForms.ListBox, listeMontant As MSForms.ListBox, _
listeTotalPaiement As MSForms.ListBox, listeMoyen As MSForms.ListBox, quantite As Integer, _
montant As Double, total As Double, moyen As String)

    total = quantite * montant

    With listeQuantitePaiement
        .AddItem quantite
    End With
    With listeMontant
        .AddItem Format(montant, "0.00")
    End With
    With listeTotalPaiement
        .AddItem Format(total, "0.00")
    End With
    With listeMoyen
        .AddItem moyen
    End With
End Sub
```

Figure 9 : Capture Ecran de la fonction de Paiement appelée dans les boutons de paiements.

Ce code reprend ce qui a été dit plus haut : chaque valeur est ajoutée aux listes associées selon le mode de paiement.

J'ai également réalisé une fonction de suppression d'un moyen de paiement. C'est grâce au bouton Supprimer présent dans la caisse qui permet de réaliser cette action. Bien que ce bouton soit utilisé pour supprimer un article, il détecte d'abord la sélection d'un des deux groupes de listes :

Les listes d'articles ou les listes de paiements.

Lorsqu'une ligne d'une des listes de paiement est sélectionnée, les autres listes associées seront sélectionnées, et la suppression s'effectue sur le même index. Le code sera affiché en annexes.

```
If index <> -1 Then
    listeQuantitePaiement.RemoveItem index
    listeMontant.RemoveItem index
    listeTotalPaiement.RemoveItem index
    listeMoyenPaiement.RemoveItem index
End If
```

Figure 10 : Capture Ecran d'une partie du code permettant la suppression d'un mode de paiement.

3.3 Les menus « InterfaceRemise » et « InterfaceRemiseNote »

J'ai réalisé deux interfaces qui s'affichent lorsque les boutons Remise ou RemiseNote sont appuyés. Des fenêtres apparaissent avec un clavier de saisie, le prix d'un article ou de la note, et la remise s'applique en temps réel, avec le nouveau prix qui est remplacé dans le prix correspondant : soit le total de la note, soit le prix de l'article sélectionné.

On doit pour cela d'abord convertir la valeur de la remise en valeur de type double, c'est-à-dire en valeur à virgule, juste au cas où afin d'éviter les incompatibilités de type. De l'élément 0 jusqu'au dernier élément de la liste, on applique la remise un par un.

```
prixActuel = CDb1(TextBoxTotalActuel.Value)
remise = CDb1(TextBoxValeur.Value)
prixFinal = prixActuel * (1 - remise / 100)

TextBoxNouveauTotal.Value = prixFinal
```

Figure 11 : Capture Ecran d'une partie du code appliquant la remise.

La remise maintenant effectuée, il suffit de remplacer la valeur de prix de l'article par la nouvelle, pour cela il faut récupérer la ligne de l'article sélectionné avec la fonction ListIndex.

En ce qui concerne la remise de la note totale, le principe reste le même, même s'il y a une petite différence : la remise est prise en compte que si une valeur est saisie, cette remise ne s'applique alors pas seulement qu'au prix de la note mais à toutes les valeurs d'articles. Il faut alors parcourir la liste et appliquer pour chaque élément le code suivant :

```
If TextBoxValeur <> "" Then
    remise = CDb1(TextBoxValeur.Value)
    For i = 0 To InterfaceNote.ListBoxPrix.ListCount - 1
        InterfaceNote.ListBoxPrix.List(i) = (InterfaceNote.ListBoxPrix.List(i)) * (1 - remise / 100)
    Next i
End If
```

Figure 12 : Capture Ecran d'une partie du code appliquant la remise de toute la note.

3.4 Affichage des boîtes de texte des paiements

Après avoir codé l'ensemble des boutons utiles au fonctionnement de la caisse enregistreuse, il fallait désormais mettre à jour certains affichages à chaque fois qu'un bouton d'articles est appuyé.

Je devais coder l'affichage du total du prix total de la commande, le reste à payer et le montant dû après paiement.

La première étape a été de réfléchir sur quels boutons cette mise à jour d'affichage allait s'appliquer. Par la suite, j'ai conclu que les boutons d'articles, le bouton offre et les boutons de paiement devaient mettre à jour ces affichages, mais également le bouton valider des deux interfaces RemiseNote et Remise.

J'ai donc créé les procédures PrixTotal, PrixDu et PrixReste dans le module Paiement. Il m'a fallu récupérer la somme de tous les éléments de la liste ListBoxPrix et ListBoxTotalPaiement. Il suffit donc de réaliser une boucle qui parcourt tous les éléments de la liste et additionne au fur et à mesure les valeurs

```
For i = 0 To InterfaceNote.ListBoxTotalPaiement.ListCount - 1
    total = total + InterfaceNote.ListBoxTotalPaiement.List(i)
Next i
```

Figure 13 : Capture Ecran d'une partie du code des différentes boîtes de textes de paiement

Pour ce qui est du total de la commande, j'ai juste à l'afficher et à convertir la valeur en décimale dans l'UserForm « InterfaceNote ».

Le reste quant à lui est égal à la soustraction du total de la commande par la somme du total des paiements.

Et pour le montant dû, il suffit de vérifier si le total qui a été payé est supérieur au total de la commande et faire une autre soustraction du total payé par le total de la note.

3.5 Les menus « InterfaceCommande » et « InterfaceChoix »

Après avoir codé l'ensemble de la caisse, il fallait relier les données des commandes rentrées dans les tableurs pour l'analyse des données.

J'ai donc réalisé deux autre UserForm : InterfaceCommande et InterfaceChoix.

L'InterfaceCommande regroupe toutes les commandes du jour, il en existera deux types : ouvertes et fermées. Les commandes fermées sont les commandes dont le total de la note a été payé. Tandis que les notes ouvertes sont des commandes non réglées qui peuvent être modifiées par la suite.

Ainsi j'ai créé 3 boutons dans cette fenêtre qui permettent d'ajouter, modifier et supprimer une commande. C'est l'ajout d'une commande qui va activer l'ouverture de l'InterfaceChoix, qui sert à choisir entre l'ajout d'une commande sur place ou bien à emporter.

J'ai également placé deux listes déroulantes, l'une qui affichera les numéros des commandes ouvertes, et l'autre qui affichera les numéros des commandes terminées.

Mais avant toutes choses, il faut d'abord créer les éléments utiles dans le tableur afin de récupérer ces données !

3.5.1 Les éléments des feuilles de calcul

	A	B	C	D	E	F	G	H	I
1	Commandes Sur Place								
2									
3									
4							Nems		
5									
6							4,50	5,00	5,00
7	Date	Numéro	Statut	Type	Paiement	Total	Nem	Nem Poulet	Nem Légumes

Figure 14 : Capture Ecran des éléments des tableurs.

Ci-dessus, nous pouvons voir à quoi ressemblera notre tableur, j'étais premièrement parti sur un seul tableur, le type de la commande alternerait alors entre Sur Place et Emporter mais suite aux erreurs d'incompatibilités et problèmes que j'ai rencontré en route, j'ai décidé de créer deux même tableurs mais pour les deux types de commandes séparément.

Voici les précisions à prendre en compte :

- Lorsqu'une commande est créé, la date du jour est affichée
- Le numéro de commande est compris entre 1 et 19 inclus pour une commande sur place
- Le numéro de commande commence à 20 pour une commande à emporter
- Le Statut affiche soit « ouvert » ou « fermé »

Ces quatre éléments sont donc la base de la création d'une commande. Le reste des éléments affichés à partir de la colonne 5 ou colonne « E » seront visibles lorsque la caisse enregistreuse commencera à avoir des articles sélectionnés.

A partir de la colonne 7 et au-delà, nous avons l'ensemble des articles du restaurant.

3.5.2 Mise à jour des articles dans les feuilles de calcul

Avant de pouvoir réaliser mes trois boutons d'ajout, suppression et modification, il faut d'abord que je réalise une mise à jour en temps réel, c'est-à-dire que lorsque je clique sur un article dans ma caisse enregistreuse, sa quantité s'affiche dans la cellule correspondante et pour la commande associée.

Pour cela, j'ai réalisé un array de tous les articles dans le même ordre que dans ma feuille de calcul, De cette manière, je peux réaliser une boucle qui parcourt cet array et inscrit dans les cellules correspondantes les quantités affichés dans la caisse. J'ai passé énormément de temps sur cette partie du code après de nombreux problèmes d'incompatibilité et des valeurs qui ne se s'affichent pas. Il a fallu que je m'assure que tous les boutons soient écrit de la même manière dans l'array et sur ma feuille de calcul.

Je détermine donc la ligne de la commande comme vu précédemment, cette fois-ci avec le numéro de commande, la date ET le statut « ouvert ».

```
For i = LBound(articles) To UBound(articles)
    articleTrouve = False

    For j = 0 To InterfaceNote.ListBoxArticle.ListCount - 1
        If InterfaceNote.ListBoxArticle.List(j) = articles(i) Then

            ws.Cells(ligne, colonne).Value = InterfaceNote.ListBoxQuantite.List(j)
            colonne = colonne + 1
            articleTrouve = True
            Exit For
        End If
    Next j

    If Not articleTrouve Then
        ws.Cells(ligne, colonne).Value = 0
        colonne = colonne + 1
    End If
Next i
```

Figure 15 : Capture Ecran du parcours de l'array et ajout dans le tableur si l'article est trouvé

A partir de la colonne 7, je parcours l'array de la première jusqu'à la dernière valeur, si je ne trouve pas d'article dans l'array qui correspond aux articles de ma commande, alors j'inscris un 0 pour l'article correspondant, et je continue de parcourir l'array. Cependant, si dans ma liste d'article, il y a un article qui correspond, alors j'inscris dans la cellule correspondante, la quantité associée à cet article. J'affiche également le total de la commande à la colonne 6

Voici un exemple de ce que je rentre dans ma caisse enregistreuse :

Commande n° 20
 Emporter

Supprimer

Remise

Remise Note

Offert

1	Nem	4,50
1	Nem Poulet	5,00
1	Nem Légumes	5,00

Dû

Reste 14,50

Total 14,50

CB

Espèces

Ticket

Chèque

24-juin-23	20	Ouvert	Emporter		14,5	1	1	1
------------	----	--------	----------	--	------	---	---	---

Figure 16 : Exemple de mise à jour des articles

On voit donc que j'ai sélectionné 1 nem, 1 nem poulet et 1 nem légume et ils ont été ajouté au tableur pour la commande 20.

3.5.3 Mise à jour des paiements dans les feuilles de calcul

Pour ce qui concerne le paiement, c'est bien plus simple ! Si la commande est payée en plusieurs moyens de paiement différents, ça affichera Mixte dans ma colonne Paiement, si plusieurs paiements d'un même type de paiement, alors ça affichera ce type de paiement.

Et pour finir, lorsque je clique sur le bouton valider, si le reste est égal à 0, alors ma commande est clotûrée, et le statut « fermé » est remplacé dans le tableur.

3.5.4 Ajouter une commande

Le Bouton d'ajout d'une commande « Nouvelle Commande » fait appel à l'UserForm InterfaceChoix. Une fenêtre s'affiche avec deux boutons, soit il s'agit d'une commande à emporter, soit d'une commande sur place.

Le code utilisé est le même pour les deux boutons mais quelques changements se font selon le type de commande choisi. Lorsque l'on clique sur le bouton, je récupère la dernière ligne non vide.

Voici le code que j'utilisais jusqu'à maintenant trouver la première ligne vide, que j'ai ensuite remplacé par une fonction intégrée dans Excel que j'ai trouvée sur Internet et dans la documentation.

```
Function indiceProchaineLigne() As Integer
' on cherche la première ligne vide
    ligne = 2
    While Cells(ligne, 1) <> ""
        ligne = ligne + 1
    Wend
    indiceProchaineLigne = ligne
End Function
```

Figure 15 : Capture Ecran de l'ancienne fonction utilisée pour trouver la première ligne vide.

```
dernierelignePlace = ws.Cells(Rows.Count, "A").End(xlUp).Row
```

Cette ligne de code permet de travailler directement avec la cellule trouvée sans faire appel à de fonctions extérieures. La première ligne de commandes commence à la ligne 8, je détecte ensuite si la date de création de la commande correspond à celle du jour, pour savoir s'il faut recommencer les numéros de commandes pour la journée.

Pour une commande sur place, les commandes commencent à 1 et à chaque nouvelle commande créée, j'augmente la valeur, selon si la date a changé ou non. J'utilise la fonction d'Excel Max afin d'augmenter la valeur si le numéro de commande existe déjà :

```
NouvelleCommandePlace = Application.WorksheetFunction.Max(1, ws.Cells(dernierelignePlace, 2).Value) + 1
```

Pour les commandes à emporter, la première valeur est 20.

Et enfin, il suffit de rajouter la date du jour, le numéro de commande, le statut « ouvert » et le type de commande selon le choix. Une commande sur place apparaîtra dans le tableur associé et l'UserForm InterfaceCommande apparaîtra avec comme labels d'affichages le type de commande et son numéro !

3.5.5 Supprimer une commande

Pour supprimer une commande, on doit utiliser les listes de l'UserForm InterfaceCommande. Afin de réaliser cette action, des commandes doivent exister, celles-ci sont affichées dans la listebbox selon le numéro de commande. Il suffit donc de sélectionner un numéro de commande, puis grâce à deux boucles, si la date du jour et le numéro de commande correspond, alors la ligne sera effacée.

Des contraintes sont également mises en place, si rien n'est sélectionné, un message d'erreur s'affiche et si un numéro de commande de la liste des commandes fermées est sélectionné, alors ça ne marchera pas non plus.

```
For i = 8 To ws.Cells(Rows.Count, 2).End(xlUp).Row
    If ws.Cells(i, 1).Value = DateduJour And ws.Cells(i, 2).Value = numCommande Then
        ws.Rows(i).Delete
    Exit For
End If
Next i
```

Figure 15 : Capture Ecran de la boucle permettant de supprimer une commande

3.5.6 Modifier une commande

Afin de modifier une commande, j'ai d'abord rentré tous les prix des articles à la ligne 6 des tableurs, juste au-dessus du nom des articles.

De plus, le début du code est le même que pour le bouton valider de la caisse enregistreuse.

Je récupère les informations dont j'ai besoin, donc le numéro de commande, avec celui-ci je peux déterminer si c'est une commande à emporter ou sur place, après quoi les différents labels sont définis. Je récupère également la ligne, c'est très important.

Ensuite, je m'assure que les listes d'articles sont bien vides. Quant aux listes de paiements, je n'ai pas besoin de récupérer cela car dans un restaurant on ne modifie pas une commande qui a commencé à être réglé.

La partie difficile peut donc commencer, j'ai réalisé une boucle qui parcourt les colonnes de ma commande, à partir de la colonne 7 (là où les articles commencent)

J'ai défini la ligne 6 comme étant celles des prix, ensuite pour chaque itération de la boucle, j'ajoute dans la caisse enregistreuse :

- son nom, qui correspond à `ws.Cells(7, colonneArticle).Value`
- sa quantité, qui est récupérée à `ws.Cells(ligne, colonneArticle).Value`
- son prix, qui affiche le prix unitaire de la ligne 6 correspondante, où bien si la quantité est supérieure à 1, le produit de la quantité par le prix unitaire.

Voici un exemple :

J'ai ma commande ouverte en emporter qui possède seulement 1 Nem

24-juin-23	20	Ouvert	Emporter		4,5	1
------------	----	--------	----------	--	-----	---

Lorsque j'ouvre ma caisse, la commande 20 est affichée dans les commandes ouvertes, je la sélectionne et je clique sur le bouton modifier.

Commande n° 20		Emporter		
Supprimer	1	Nem	4,5	CB

Figure 16 : Exemple de modification d'une commande

Le produit avec son prix et sa quantité sont bien affichés et je peux donc continuer ma commande.

3.6 Analyse des données

Avec tous ces éléments en compte, la caisse enregistreuse est donc fonctionnelle pour créer, modifier et supprimer des commandes et enregistrer les données de chaque commande permettant ainsi de réaliser une analyse des données.

La caisse enregistreuse avec les principales fonctions ayant été finalisée fin juin, je n'ai pris que le mois de juillet dans mes données.

3.6.1 Création de la feuille de calcul « Données »

Pour réaliser une partie de mon analyse, j'ai créé une feuille de calcul « Données » qui regroupe le mois de juin et juillet avec les différents produits.

Chaque cellule de ligne « date » et de colonne « produit » affichera la quantité vendue d'article le jour correspondant. Pour cela, j'ai dû écrire une formule Excel utilisant la fonction SOMME.SI.ENS

$$=SOMME.SI.ENS('Commandes Sur Place'!G\$8:G\$980;'Commandes Sur Place'!\$A\$8:\$A\$980;\$B8) + SOMME.SI.ENS('Commandes à Emporter'!G\$8:G\$983;'Commandes à Emporter'!\$A\$8:\$A\$983;\$B8)$$

J'ai procédé à la somme de chaque produit selon le critère de date. La formule doit être écrite de la forme =SOMME.SI.ENS(plage somme, plage critère, critère)

Dans la formule que j'ai saisie, je calcule donc le total d'article par jour d'un article selon la date que je souhaite. Les \$ permettent de fixer la valeur, ici c'est la ligne 8 qui ne change pas, seulement les colonnes, j'écris donc G\$8 et non pas \$G\$8.

Cela me permet de faire défiler la formule sur les cellules de droite et me faciliter la tâche. Ensuite je réalise la même chose vers le bas.

Ventes Juin	158	196	127	85	48	22	36	31	37
Ventes Juillet	175	218	144	99	60	28	41	23	35
Articles	Nems			Hors d'œuvres					
Prix	4,50	5,00	5,00	13,00	6,00	5,50	5,50	5,50	6,50
Date	Nem	Nem Poulet	Nem Légumes	siette Mélange	gâteaux Printen	Raviolis Frits	Samossas	Beignet Poulet	Beignet Crevett
jeudi 01/06/23	9	10	5	6	1	0	0	1	0
vendredi 02/06/23	7	9	4	3	1	2	3	0	0
samedi 03/06/23	8	10	7	7	1	0	0	2	1
lundi 05/06/23	1	0	0	3	1	0	3	0	2
mardi 06/06/23	8	3	5	1	3	0	0	1	2

Figure 17 : Tableau feuille de calcul des données

Je possède ainsi la quantité de chaque article vendu sur place et à emporter par jour. Plus haut, j'ai rajouté le total de chaque produit avec la fonction =SOMME qui m'affiche le total d'articles vendus dans le mois correspondant.

956,2	samedi	24/06/23			
350,5	lundi	26/06/23		Total Emporter Juin	8992,68
455	mardi	27/06/23		Total Emporter Juillet	9713,68
454,7	mercredi	28/06/23			
633,08	jeudi	29/06/23			
1088,8	vendredi	30/06/23		Total Juin	15271,58
825,5	samedi	01/07/23		Total Juillet	17245,48
503,1	lundi	03/07/23			
378,5	mardi	04/07/23			
665,4	mercredi	05/07/23		Total Sur Place Juin	6278,9
603	jeudi	06/07/23		Total Sur Place Juillet	7531,8
789,9	vendredi	07/07/23			

Figure 18 : Tableau feuille de calcul des données concernant les totaux des deux mois

Par la suite, tout à droite de la feuille, j'ai réalisé le total de chaque journée pour chaque mois, toujours avec la formule SOMME.SI.ENS.

Ces données m'ont ainsi permis de réaliser le total du mois de juin et juillet pour réaliser une comparaison des ventes. J'ai ensuite réalisé le total de chaque mois pour les deux types de commandes.

3.6.2 Graphiques

Grâce aux données que j'ai prélevées, je peux ainsi réaliser des graphiques qui me permettent de mieux visualiser ce que représente les ventes.

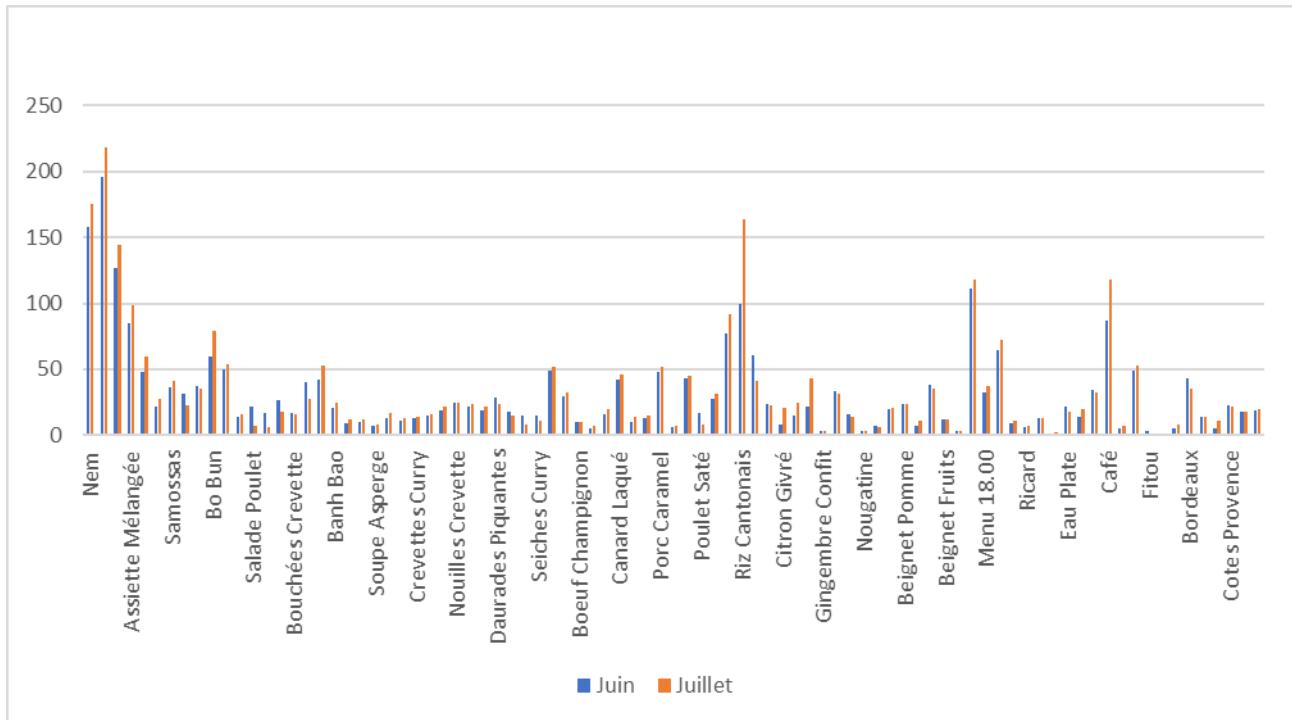


Figure 19 : Graphique affichant les ventes réalisées sur chaque article en juin et juillet

VOIR ANNEXE POUR LE TABLEAU COMPLET.

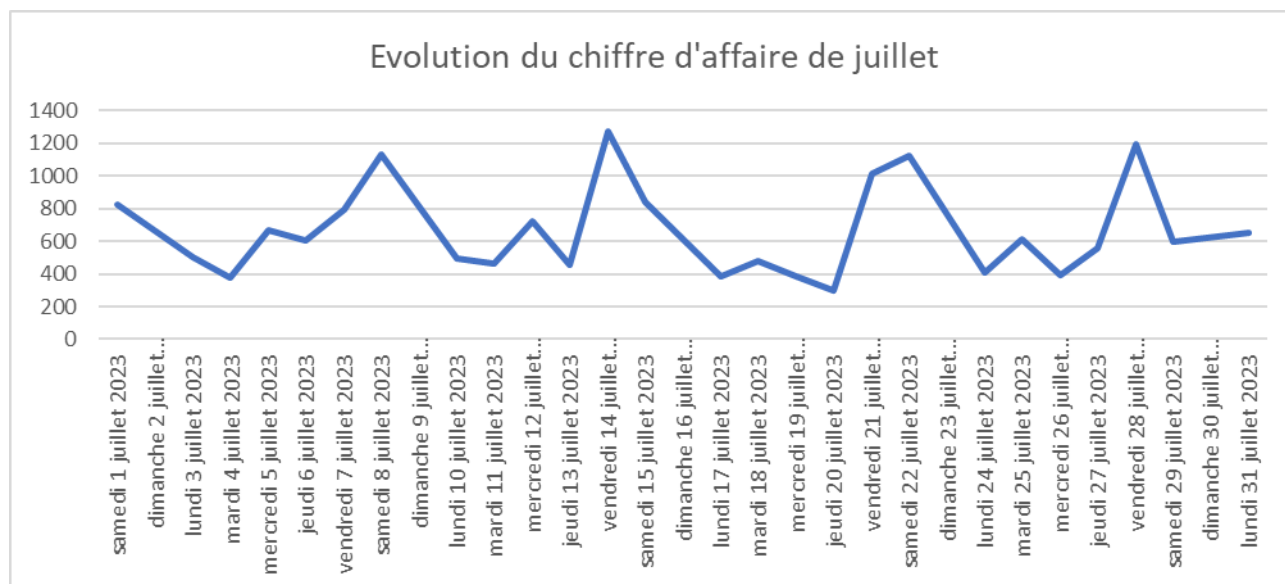
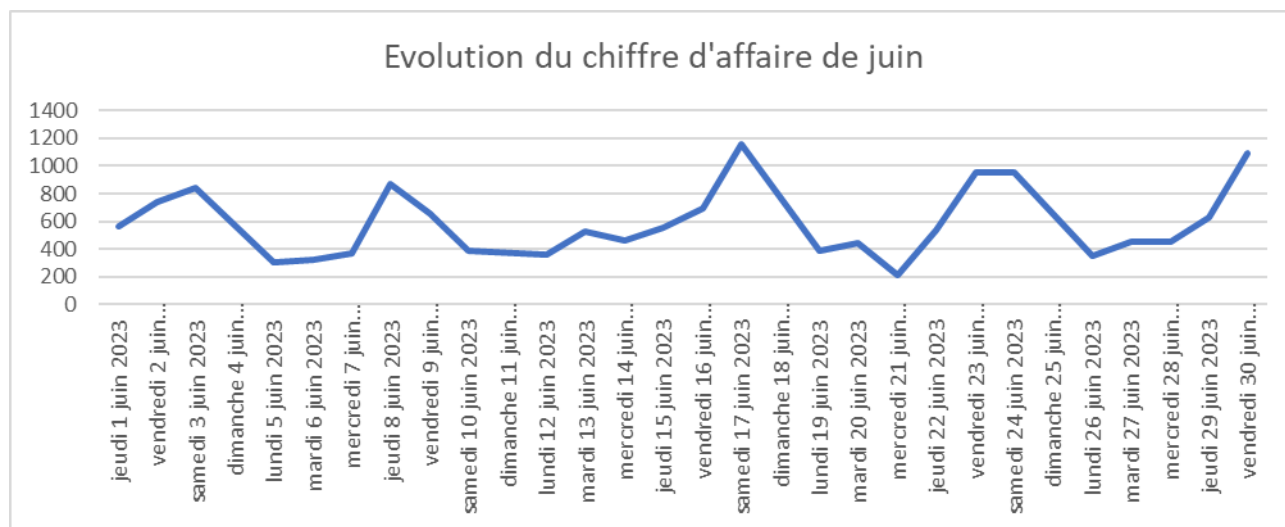
D'après les données affichées, on observe déjà que les menus sont très populaires. Les clients ont tendance à prendre le menu du midi plus généralement.

Au niveau de la carte, les entrées les plus populaires sont les Nems au poulet. Cela peut notamment s'expliquer au fait qu'il y ait 5 Nems contrairement aux Nems Porc et Légumes qui n'en proposent que 4. Ensuite, nous avons les Assiettes Mélangées qui restent très populaires pour partager à deux.

Aucune entrée ne semble délaissée hormis la salade crevette et poulet durant le mois de juillet qui sont pourtant assez vendues d'habitude. Même chose pour les vapeurs qui sont peu consommés sauf pour le mélange vapeur qui est très populaire aussi. Les soupes en ces périodes de chaleur sont peu choisies également.

Il y a donc une bonne répartition des choix des articles, notamment le porc caramel, poulet curry et bœuf piquant ! Mais surtout le riz cantonnais en accompagnement qui est le produit le plus populaire en juillet. Le saté reste tout de même très peu choisi plat.

Les produits les moins populaires comme le gingembre confit, la nougatine, les bouteilles de Badoit, de Fitou et de Cahors sont tellement peu voire pas du tout vendues que le restaurant peut envisager de revoir sa carte et retirer ces produits.

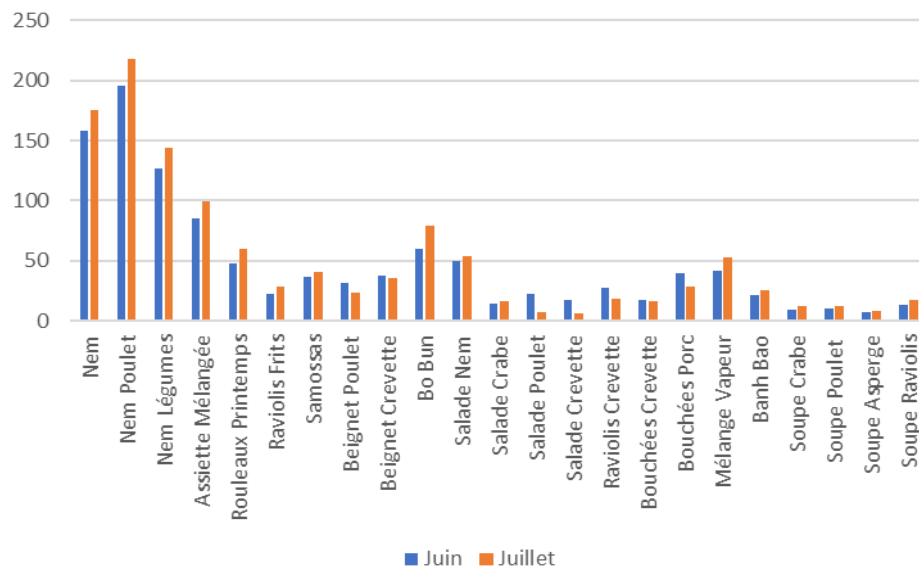


Voici l'évolution du chiffre d'affaires au cours des deux mois, on peut observer que les pics des meilleures journées sont le 3, 8, 17, 23 et 24 juin. Pour juillet, ce sont les 8, 14, 21, 22 et 28 juillet. Ces journées correspondent à des vendredis et samedis généralement et parfois le jeudi.

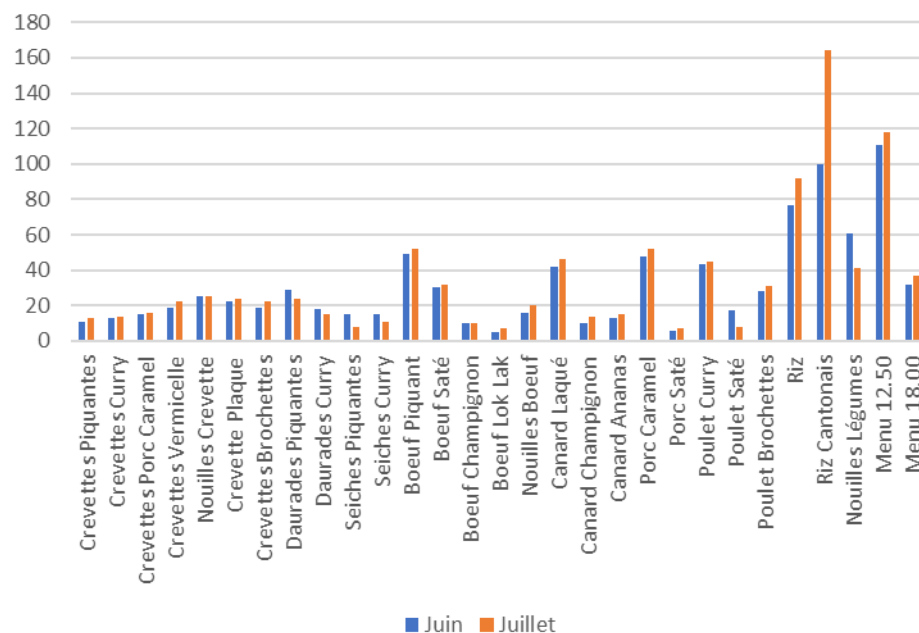
Quant aux journées les moins bonnes, ce sont souvent des mercredis ou des mardis, en début de semaine.

Quant aux graphiques des ventes, les entrées comme dit précédemment se vendent plutôt bien avec notamment les nems en produits phare. Les bouchées crevettes sont très peu vendues mais sont tout de même utilisés dans les mélanges vapeurs, il n'est donc pas envisageable de les retirer de la carte.

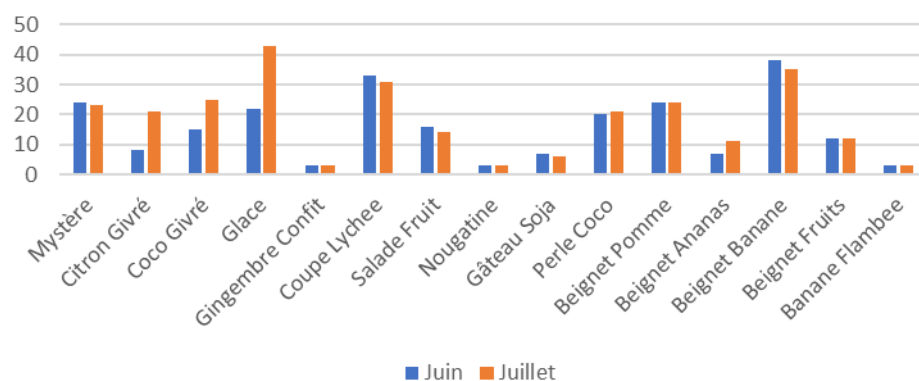
Graphique des ventes d'entrées en juin et juillet

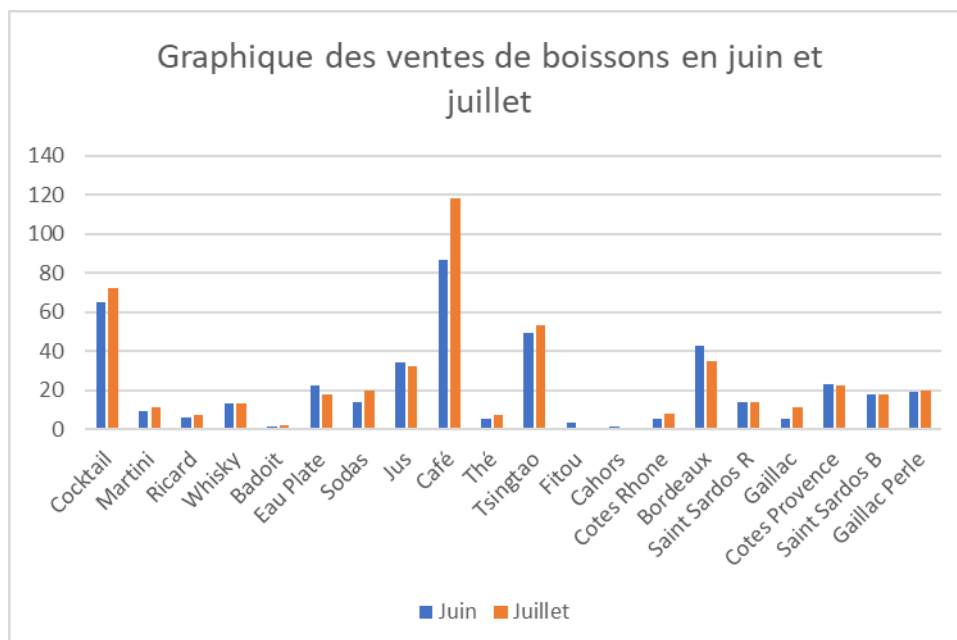


Graphique des ventes de plats en juin et juillet



Graphique des ventes de desserts en juin et juillet





A travers le travail réalisé au restaurant, le gingembre confit est un dessert très peu populaire qui pourrait être retiré de la carte, quant à certains plats comme le porc et le poulet saté, ils n'ont pas énormément la côte auprès de la clientèle et il est possible d'envisager leurs retraits de la carte.

Le chiffre d'affaires du mois de Juillet est bien plus élevé que Juin et peut notamment s'expliquer par le fait que ce soit le mois du début des vacances. Pour les deux mois consécutifs, c'est les commandes à emporter qui rapportent le plus d'argent, suivi des commandes sur place. L'écart entre chaque mois est très minime mais reste régulier

3.7 Manuels Utilisateur et Technique

Afin de permettre aux futurs utilisateurs de la caisse enregistreuse de comprendre son fonctionnement, j'ai réalisé un manuel utilisateur et un manuel technique. Le manuel utilisateur permet de savoir comment utiliser la caisse, avec les différents boutons, étapes par étapes et pour le manuel technique, il explique comment fonctionne le code au cas où celui-ci venait à être modifié par une autre personne que moi-même.

III - Conclusion

1. Synthèse du travail réalisé

Je suis très satisfait de ce que j'ai pu réaliser et des changements apportés à mon projet au cours de ces 10 semaines. J'ose espérer que ce que j'ai réalisé sera utile au restaurant dans le futur. Etant en contrat étudiant, je continuerai à travailler au sein du restaurant et je continuerai ainsi à développer ma caisse enregistreuse, qui n'est qu'un début par rapport au produit final escompté.

Ce projet a été réalisé avec des échanges réguliers avec ma tutrice concernant les attentes de la caisse enregistreuse et j'ai appris davantage de fonctionnalités Excel

J'ai passé la majeure partie de mon stage à réaliser la caisse enregistreuse et ça m'a demandé bien plus de temps que je le pensais, la réalisation d'une gestion des stocks est un objectif que je n'ai pas pu atteindre mais je suis tout de même satisfait de ce que j'ai fait. Certaines fonctionnalités comme le paiement par ticket restaurant sont inachevées et j'envisage d'en rajouter bien plus comme par exemple le nombre de couverts par commande et une page avec les produits les plus populaires dans la multipage.

Une interface bien plus ergonomique est aussi envisagée pour la suite.

2. Difficultés rencontrées

Durant cette période de stage, j'ai rencontré quelques difficultés en cours de route. Les missions qui m'ont été fixées étaient bien plus difficiles que je ne le pensais :

- Respect du plan de travail difficile car la caisse enregistreuse est affectée énormément d'éléments
- Changements constant des colonnes des feuilles que je devais adapter à mon code
- Enormément d'incompatibilité dans le code VBA pour les nombres décimaux/entiers
- Erreur de code et oublis d'articles dans les arrays d'article utilisés
- Rythme de travail trop difficile

3. Bilan professionnel et personnel

Ce stage m'a appris à mieux gérer mon temps et m'a permis d'être plus responsable. La réalisation de ces missions m'a également permis de mettre en pratique mes compétences en informatique, notamment en VBA/Excel que j'apprécie particulièrement.

Etant assez réservé, le contact avec les clients et l'équipe m'a permis de m'ouvrir plus facilement aux gens, je suis bien plus à l'aise à l'oral et j'ai plus confiance en moi. Je suis également bien plus autonome qu'avant et plus efficace ! J'ai ainsi réalisé 35h par semaines sans compter les heures supplémentaires sur l'ordinateur, ça m'a demandé beaucoup d'efforts et je pense que ce stage a été globalement une réussite que ce soit sur le plan professionnel et personnel.

IV- Bibliographie

Site restaurant Le Pont d'Asie

<https://www.restaurant-vietnamien.fr/>

Documentation Microsoft Visual Basic for Applications:

<https://learn.microsoft.com/en-us/office/vba/language/reference/functions-visual-basic-for-applications>

Cours UT2J Projet réalisation d'une application interactive en VBA Excel:

<https://iris.univ-tlse2.fr/course/view.php?id=2160>

Réaliser des graphiques sur Excel

<https://blog.hubspot.fr/marketing/graphique-excel>

Site explicatif pour le code VBA Excel

<https://excelchamps.com/vba/find-last-row-column-cell/>

Discussion Reddit sur le changement de la virgule en point

https://www.reddit.com/r/vba/comments/ikhalk/how_to_change_the_decimal_separator_for_sub_th
[at/](https://www.reddit.com/r/vba/comments/ikhalk/how_to_change_the_decimal_separator_for_sub_th)

Explication des arrays sur Excel

<https://excelchamps.com/vba/arrays/>

Cours sur Excel

<https://excelchamps.com/>

Utilisation du SUMIF ou SOMME.SI

https://www.youtube.com/watch?v=egvuzSI-JFU&ab_channel=ComputerTutoring

V- Annexes

Ajouter un article dans la caisse & Supprimer un article dans la caisse

```
''' Ajouter un Article dans la caisse '''
Sub ajouterarticle(listeArticle As MSForms.ListBox, listeQuantite As MSForms.ListBox, _
listePrix As MSForms.ListBox, nomArticle As String, quantite As Double, prix As Double)
    With listeArticle
        .AddItem nomArticle
    End With
    With listeQuantite
        .AddItem quantite
    End With
    With listePrix
        .AddItem Format(prix, "0.00")
    End With
End Sub

''' Supprimer un Article dans la caisse '''
Sub SupprimerArticle(listeArticle As MSForms.ListBox, listeQuantite As MSForms.ListBox, listePrix As MSForms.ListBox)
    Dim index As Integer

    index = listeArticle.ListIndex

    If index <> -1 Then
        listeArticle.RemoveItem index
        listeQuantite.RemoveItem index
        listePrix.RemoveItem index
    End If

    listeArticle.ListIndex = -1
    listeQuantite.ListIndex = -1
    listePrix.ListIndex = -1
End Sub
```

Ajouter un paiement dans la caisse & Supprimer un paiement dans la caisse

```
Sub Payer(listeQuantitePaiement As MSForms.ListBox, listeMontant As MSForms.ListBox, _
listeTotalPaiement As MSForms.ListBox, listeMoyen As MSForms.ListBox, quantite As Integer, _
montant As Double, total As Double, moyen As String)

    total = quantite * montant

    With listeQuantitePaiement
        .AddItem quantite
    End With
    With listeMontant
        .AddItem Format(montant, "0.00")
    End With
    With listeTotalPaiement
        .AddItem Format(total, "0.00")
    End With
    With listeMoyen
        .AddItem moyen
    End With
End Sub

Sub SupprimerPaiement(listeQuantitePaiement As MSForms.ListBox, listeMontant As MSForms.ListBox, _
listeTotalPaiement As MSForms.ListBox, listeMoyenPaiement As MSForms.ListBox)

    Dim index As Integer

    index = listeQuantitePaiement.ListIndex

    If index <> -1 Then
        listeQuantitePaiement.RemoveItem index
        listeMontant.RemoveItem index
        listeTotalPaiement.RemoveItem index
        listeMoyenPaiement.RemoveItem index
    End If

    listeQuantitePaiement.ListIndex = -1
    listeMontant.ListIndex = -1
    listeTotalPaiement.ListIndex = -1
    listeMoyenPaiement.ListIndex = -1
End Sub
```

Calcul du prix Total, prix Reste et prix Dû

```
Sub prixTotal()  
  
    Dim total As Double  
  
    InterfaceNote.TextBoxTotal.Value = "0,00"  
  
    For i = 0 To InterfaceNote.ListBoxPrix.ListCount - 1  
        total = total + InterfaceNote.ListBoxPrix.List(i)  
    Next i  
  
    If total = 0 Then  
        InterfaceNote.TextBoxTotal.Value = "0,00"  
    Else  
        InterfaceNote.TextBoxTotal.Value = Format(total, "0.00")  
    End If  
  
End Sub
```

```
Sub PrixReste()  
    Dim total As Double  
    Dim reste As Double  
  
    InterfaceNote.TextBoxReste.Value = "0,00"  
  
    For i = 0 To InterfaceNote.ListBoxTotalPaiement.ListCount - 1  
        total = total + InterfaceNote.ListBoxTotalPaiement.List(i)  
    Next i  
  
    If total = 0 Then  
        InterfaceNote.TextBoxReste.Value = InterfaceNote.TextBoxTotal.Value  
    Else  
        reste = InterfaceNote.TextBoxTotal.Value - total  
        If reste >= 0 Then  
            InterfaceNote.TextBoxReste.Value = reste  
        Else  
            InterfaceNote.TextBoxReste.Value = "0,00"  
        End If  
    End If  
  
End Sub
```

```
Sub PrixDu()  
  
    Dim valeur As Double  
    Dim total As Double  
  
    totalnote = InterfaceNote.TextBoxTotal.Value  
  
    InterfaceNote.TextBoxDu.Value = "0,00"  
  
    For i = 0 To InterfaceNote.ListBoxTotalPaiement.ListCount - 1  
        total = total + InterfaceNote.ListBoxTotalPaiement.List(i)  
    Next i  
  
    If total > totalnote Then  
        InterfaceNote.TextBoxDu.Value = total - totalnote  
    Else  
        InterfaceNote.TextBoxDu.Value = "0,00"  
    End If  
  
End Sub
```

Mise à jour des articles dans les feuilles de calcul

```

Sub Mise_à_jour_article()
    Dim articles() As Variant
    articles = Array("Nem", "Nem Poulet", "Nem Légumes", "Assiette Mélangée", "Rouleaux Printemps", _
        "Raviolis Frits", "Samossas", "Beignet Poulet", "Beignet Crevette", "Bo Bun", _
        "Salade Nem", "Salade Crabe", "Salade Poulet", "Salade Crevette", "Raviolis Crevette", _
        "Bouchées Crevette", "Bouchées Porc", "Mélange Vapeur", "Banh Bao", "Soupe Crabe", _
        "Soupe Poulet", "Soupe Asperge", "Soupe Raviolis", "Crevettes Piquantes", "Crevettes Curry", _
        "Crevettes Porc Caramel", "Crevettes Vermicelle", "Nouilles Crevette", "Crevette Plaque", _
        "Crevettes Brochettes", "Daurades Piquantes", "Daurades Curry", "Seiches Piquantes", _
        "Seiches Curry", "Boeuf Piquant", "Boeuf Saté", "Boeuf Champignon", "Boeuf Lok Lak", _
        "Nouilles Boeuf", "Canard Laqué", "Canard Champignon", "Canard Ananas", "Porc Caramel", _
        "Porc Saté", "Poulet Curry", "Poulet Saté", "Poulet Brochettes", "Riz", "Riz Cantonais", _
        "Nouilles Légumes", "Mystère", "Citron Givré", "Coco Givré", "Glace", "Gingembre Confit", _
        "Coupe Lychee", "Salade Fruit", "Nougatine", "Gâteau Soja", "Perle Coco", "Beignet Pomme", _
        "Beignet Ananas", "Beignet Banane", "Beignet Fruits", "Banane Flambee", "Menu 12.50", "Menu 18.00", "Cocktail", _
        "Martini", "Ricard", "Whisky", "Badoit", "Eau Plate", "Sodas", "Jus", "Café", "Thé", "Tsingtao", _
        "Fitou", "Cahors", "Cotes Rhone", "Bordeaux", "Saint Sardos R", "Gaillac", "Cotes Provence", _
        "Saint Sardos B", "Gaillac Perlé")

    Dim ws As Worksheet

    Dim derniereligne As Long

    Dim ligne As Long
    Dim colonne As Long
    Dim i As Long
    Dim j As Long

    Dim articleTrouve As Boolean

    Dim numeroCommande As String
    Dim DateduJour As Date
    Dim TypeCommande As String

    If InterfaceNote.LabelType.Caption = "Emporter" Then
        Set ws = ThisWorkbook.Sheets("Commandes à Emporter")
    Else
        Set ws = ThisWorkbook.Sheets("Commandes Sur Place")
    End If

    numeroCommande = InterfaceNote.LabelNumero.Caption
    DateduJour = Date
    TypeCommande = InterfaceNote.LabelType.Caption

    derniereligne = ws.Cells(ws.Rows.Count, "A").End(xlUp).Row

    For ligne = 8 To derniereligne
        If ws.Cells(ligne, 2).Value = numeroCommande And ws.Cells(ligne, 1).Value = DateduJour And ws.Cells(ligne, 3).Value = "Ouvert" Then

            colonne = 7

            For i = LBound(articles) To UBound(articles)
                articleTrouve = False

                For j = 0 To InterfaceNote.ListBoxArticle.ListCount - 1
                    If InterfaceNote.ListBoxArticle.List(j) = articles(i) Then

                        ws.Cells(ligne, colonne).Value = InterfaceNote.ListBoxQuantite.List(j)
                        colonne = colonne + 1
                        articleTrouve = True
                        Exit For
                    End If
                Next j

                If Not articleTrouve Then
                    ws.Cells(ligne, colonne).Value = 0
                    colonne = colonne + 1
                End If
            Next i

            If InterfaceNote.TextBoxTotal.Value <> "" Then
                ws.Cells(ligne, 6).Value = CDbl(InterfaceNote.TextBoxTotal.Value)
            End If

            Exit For
        End If
    Next ligne
End Sub

```

Mise à jour des paiement dans les feuilles de calcul

```
Sub Mise_à_jour_paiement()  
    Dim ws As Worksheet  
    Dim derniereligne As Long  
    Dim premierMoyenPaiement As String  
    Dim dernierMoyenPaiement As String  
  
    Set ws = ThisWorkbook.Sheets("Commandes Sur Place")  
    Set ws2 = ThisWorkbook.Sheets("Commandes à Emporter")  
  
    '***** SUR PLACE *****  
  
    If InterfaceNote.LabelType = "Sur Place" Then  
        derniereligne = ws.Cells(Rows.Count, "A").End(xlUp).Row  
  
        If InterfaceNote.ListBoxMoyenPaiement.ListCount > 1 Then  
            premierMoyenPaiement = InterfaceNote.ListBoxMoyenPaiement.List(0)  
            dernierMoyenPaiement = InterfaceNote.ListBoxMoyenPaiement.List(InterfaceNote.ListBoxMoyenPaiement.ListCount - 1)  
  
            If premierMoyenPaiement <> dernierMoyenPaiement Then  
                ws.Cells(derniereligne, 5).Value = "Mixte"  
            Else  
                ws.Cells(derniereligne, 5).Value = premierMoyenPaiement  
            End If  
        ElseIf InterfaceNote.ListBoxMoyenPaiement.ListCount = 1 Then  
            ws.Cells(derniereligne, 5).Value = InterfaceNote.ListBoxMoyenPaiement.List(0)  
        End If  
  
    '***** EMPORTEUR *****  
  
    ElseIf InterfaceNote.LabelType = "Emporter" Then  
        derniereligne = ws2.Cells(Rows.Count, "A").End(xlUp).Row  
  
        If InterfaceNote.ListBoxMoyenPaiement.ListCount > 1 Then  
            premierMoyenPaiement = InterfaceNote.ListBoxMoyenPaiement.List(0)  
            dernierMoyenPaiement = InterfaceNote.ListBoxMoyenPaiement.List(InterfaceNote.ListBoxMoyenPaiement.ListCount - 1)  
  
            If premierMoyenPaiement <> dernierMoyenPaiement Then  
                ws2.Cells(derniereligne, 5).Value = "Mixte"  
            Else  
                ws2.Cells(derniereligne, 5).Value = premierMoyenPaiement  
            End If  
        ElseIf InterfaceNote.ListBoxMoyenPaiement.ListCount = 1 Then  
            ws2.Cells(derniereligne, 5).Value = InterfaceNote.ListBoxMoyenPaiement.List(0)  
        End If  
    End If  
End Sub
```


Bouton Valider de la caisse enregistreuse

```
' ' Bouton Valider ' '
Private Sub BoutonValider_Click()
|
    Dim ws As Worksheet
    Dim derniereligne As Long
    Dim DateduJour As Date
    Dim numeroCommande As String

    numeroCommande = InterfaceNote.LabelNumero.Caption
    DateduJour = Date

    ' ' Pour la modification d'une commande ' '
    If InterfaceCommande.ListBoxOuvert.ListIndex <> -1 Then
        If numeroCommande < 20 Then
            Set ws = ThisWorkbook.Sheets("Commandes Sur Place")
        Else
            Set ws = ThisWorkbook.Sheets("Commandes à Emporter")
        End If

        derniereligne = ws.Cells(ws.Rows.Count, 1).End(xlUp).Row

        For ligne = 8 To derniereligne
            If ws.Cells(ligne, 1) = DateduJour And ws.Cells(ligne, 2).Value = numeroCommande And ws.Cells(ligne, 3).Value = "Ouvert" Then
                If TextBoxReste.Value = 0 Or TextBoxReste.Value = "0" Then
                    ws.Cells(ligne, 3).Value = "Fermé"
                End If
            End If
        Next ligne

    ' ' Pour la création d'une nouvelle commande ' '
    Else
        If numeroCommande < 20 Then
            Set ws = ThisWorkbook.Sheets("Commandes Sur Place")
        Else
            Set ws = ThisWorkbook.Sheets("Commandes à Emporter")
        End If

        derniereligne = ws.Cells(ws.Rows.Count, 1).End(xlUp).Row

        For ligne = 8 To derniereligne
            If ws.Cells(ligne, 1) = DateduJour And ws.Cells(ligne, 2).Value = numeroCommande And ws.Cells(ligne, 3).Value = "Ouvert" Then
                If TextBoxReste.Value = 0 Or TextBoxReste.Value = "0" Then
                    ws.Cells(ligne, 3).Value = "Fermé"
                End If
            End If
        Next ligne

    End If

    Call MAJ.Mise_à_jour_paiement
    Call MAJ.Mise_à_jour_article
    Unload Me
    Unload InterfaceCommande
    InterfaceCommande.Show

End Sub
```

Initialisation de l'Userform InterfaceCommande qui affiche les listes de numéros de commande

```
Private Sub UserForm_Initialize()  
    Dim ws As Worksheet  
    Dim derniereligne As Long  
    Dim currentDate As Date  
    Dim i As Long  
  
    Set ws = ThisWorkbook.Sheets("Commandes Sur Place")  
    Set ws2 = ThisWorkbook.Sheets("Commandes à Emporter")  
  
    currentDate = Date  
    LabelDate.Caption = Date  
  
    derniereligne = ws.Cells(ws.Rows.Count, "A").End(xlUp).Row  
  
    For i = 8 To derniereligne  
        If ws.Cells(i, "A").Value = currentDate Then  
            If ws.Cells(i, "C").Value = "Fermé" Then  
                ListBoxFermé.AddItem ws.Cells(i, "B").Value  
            ElseIf ws.Cells(i, "C").Value = "Ouvert" Then  
                ListBoxOuvert.AddItem ws.Cells(i, "B").Value  
            End If  
        End If  
    Next i  
  
    For i = 8 To derniereligne  
        If ws2.Cells(i, "A").Value = currentDate Then  
            If ws2.Cells(i, "C").Value = "Fermé" Then  
                ListBoxFermé.AddItem ws2.Cells(i, "B").Value  
            ElseIf ws2.Cells(i, "C").Value = "Ouvert" Then  
                ListBoxOuvert.AddItem ws2.Cells(i, "B").Value  
            End If  
        End If  
    Next i  
End Sub
```

Bouton Modifier Commande

```
Private Sub BoutonModifier_Click()  
    Dim TypeCommande As String  
    Dim ws As Worksheet  
    Dim ligne As Long  
    Dim derniereLigne As Long  
    Dim numeroCommande As String  
    Dim DateduJour As Date  
    Dim colonneArticle As Long  
  
    If ListBoxOuvvert.ListIndex >= 0 Then  
        If ListBoxOuvvert.Value < 20 Then  
            TypeCommande = "Sur Place"  
            Set ws = ThisWorkbook.Sheets("Commandes Sur Place")  
        Else  
            TypeCommande = "Emporter"  
            Set ws = ThisWorkbook.Sheets("Commandes à Emporter")  
        End If  
  
        numeroCommande = ListBoxOuvvert.Value  
        DateduJour = Date  
  
        Dim commandeTrouvee As Boolean  
        commandeTrouvee = False  
        derniereLigne = ws.Cells(ws.Rows.Count, 1).End(xlUp).Row  
  
        For ligne = 8 To derniereLigne  
            If ws.Cells(ligne, 1) = DateduJour And ws.Cells(ligne, 2).Value = numeroCommande And ws.Cells(ligne, 3).Value = "Ouvvert" Then  
                commandeTrouvee = True  
                Exit For  
            End If  
        Next ligne  
  
        InterfaceNote.LabelCommande.Caption = "Commande n°"  
        InterfaceNote.LabelType.Caption = TypeCommande  
        InterfaceNote.LabelNumero.Caption = numeroCommande  
  
        InterfaceNote.LabelCommande.Caption = "Commande n°"  
        InterfaceNote.LabelType.Caption = TypeCommande  
        InterfaceNote.LabelNumero.Caption = numeroCommande  
  
        InterfaceNote.ListBoxArticle.Clear  
        InterfaceNote.ListBoxQuantite.Clear  
        InterfaceNote.ListBoxPrix.Clear  
  
        colonneArticle = 7  
  
        Do While ws.Cells(7, colonneArticle).Value <> ""  
            Dim quantite As Double  
            quantite = ws.Cells(ligne, colonneArticle).Value  
  
            If quantite <> 0 Then  
                Dim prixUnitaire As Double  
                prixUnitaire = ws.Cells(6, colonneArticle).Value  
  
                Dim prixArticles As Double  
                If quantite > 1 Then  
                    prixArticles = quantite * prixUnitaire  
                Else  
                    prixArticles = prixUnitaire  
                End If  
  
                InterfaceNote.ListBoxArticle.AddItem ws.Cells(7, colonneArticle).Value  
                InterfaceNote.ListBoxQuantite.AddItem quantite  
                InterfaceNote.ListBoxPrix.AddItem prixArticles  
            End If  
  
            colonneArticle = colonneArticle + 1  
        Loop  
  
        InterfaceNote.TextBoxTotal.Value = ws.Cells(ligne, 6).Value  
        InterfaceNote.TextBoxReste.Value = ws.Cells(ligne, 6).Value  
  
        InterfaceNote.Show  
    Else  
        MsgBox "Veuillez sélectionner une commande à modifier."  
    End If  
End Sub
```

InterfaceNote

Caisse Enregistreuse - Le Pont d'Asie

LE PONT D'ASIE

Entrées | Plats | Desserts | Boissons | Menus & Autres

Entrées

Nems

Nem Nem Poulet Nem Légumes

Hors d'oeuvres

Assiette

Rouleaux Raviolis Frits Samossas Beignet Poulet Beignet Crevette

Salades

Bo Bun Salade Nems Salade Crabe Salade Poulet Salade Crevette

Vapeurs

Raviolis Crevette Bouchées Crevett Bouchées Porc Mélange Vapeur Banh Bao

Soupes

Soupe Crabe Soupe Poulet Soupe Asperge Soupe Raviolis

Supprimer Remise Remise Note Offert

CB Espèces Ticket Chèque

Dû Reste Total

7 8 9 ←
4 5 6 x
1 2 3
0 , ✓

Annuler Note

InterfaceCommande

Caisse Enregistreuse

LE PONT D'ASIE

Commandes du jour en cours

Commandes du jour terminées

Nouvelle Commande Modifier Commande Supprimer Commande

[illegible]