

## STAGE Solti




## Sommaire

Introduction : .....	3
Mes missions à Solti : .....	4
Facture de sorti de produit : .....	4
Supprimer un document : .....	6
Conclusion.....	7
Niko Niko.....	7

# Introduction :

Solti est une entreprise basée à Seyssinet-Pariset qui commercialise différents logiciels concernant la gestion de stock. Le logiciel nommé GSM (Gestion de stock modulaire) est le plus utilisé, il permet aux entreprises de gérer leur stock. Elle possède actuellement plus de 1550 client à travers 75 pays différents.

Durant mes semaines de stages mes mission seront l'ajout de nouvelles fonctionnalités sur leurs différents logiciels en fonction des besoins des clients et de l'entreprise.

 Version 5.004 Copyright © Solti 2022

Journal

Produits

Fournisseurs

Clients


Affaires

Journal

Inventaire

Etiquettes

Outils

 AIDE

Famille

Reference

Designation

Fournisseur

Affaire

Client

Commentaire

Type

Utilisateur

Emplacement

Tous critères

Total

☐ Période

15/01/2022

15/01/2022

Période

	ID	Dates	Type	Comn	Famille	ID_prc	Refere	Design	Fourni	Quant	Mouvé	Mini	Empla	PUA	PTA	PUV	PTV	Utilisat	Affaire	C
▶	810	14/0...	Modif		SILIC...	210	1001...	DRO...	LDM...	2	0	0	D10 ...	0	0	0	0	PCS...		
	809	14/0...	Modif		SILIC...	209	1001...	DRO...	LDM...	1	0	0	D10 ...	0	0	0	0	PCS...		
	808	14/0...	Modif		SILIC...	208	1001...	DRO...	LDM...	1	0	0	D10 ...	0	0	0	0	PCS...		
	807	14/0...	Cre...		SILIC...	0	1001...	DRO...	LDM...	1	1	0	D10 ...	0	0	0	0	PCS...		
	806	14/0...	Cre...		SILIC...	0	1001...	DRO...	LDM...	1	1	0	D10 ...	0	0	0	0	PCS...		
	805	14/0...	Cre...		SILIC...	0	1001...	DRO...	LDM...	1	1	0	D10 ...	0	0	0	0	PCS...		
	804	14/0...	Modif		SILIC...	213	1001...	DRO...	LDM...	2	0	0	D10 ...	0	0	0	0	PCS...		
	803	14/0...	Modif		Exh...	47	1001...	Join...	LDM...	2	0	0		5,83	0	9,04	0	PCS...		
	802	14/0...	Entree		Exh...	47	1001...	Join...	LDM...	3	1	0		5,83	5,83	9,04	9,04	PCS...		
	801	14/0...	Modif		Exh...	47	1001...	Join...	LDM...	2	0	0		5,83	0	9,04	0	PCS...		
	800	14/0...	Cre...		SILIC...	0	1001...	DRO...	LDM...	1	1	0	D10 ...	0	0	0	0	PCS...		
	799	14/0...	Cre...		SILIC...	0	1001...	DRO...	LDM...	1	1	0	D10 ...	0	0	0	0	PCS...		
	798	14/0...	Cre...		SILIC...	0	1001...	DRO...	LDM...	1	1	0	D10 ...	0	0	0	0	PCS...		
	797	14/0...	Modif		SILIC...	180	1001...	DRO...	LDM...	2	0	0	D10 ...	2,02	0	2,68	0	PCS...		
	796	14/0...	Cre...		SILIC...	0	1001...	DRO...	LDM...	2	2	0	D10 ...	0	0	0	0	PCS...		
	795	14/0...	Cre...		SILIC...	0	1001...	DRO...	LDM...	1	1	0	D10 ...	0	0	0	0	PCS...		
	794	14/0...	Cre...		SILIC...	0	1001...	DRO...	LDM...	1	1	0	D10 ...	0	0	0	0	PCS...		
	793	13/0...	Modif		CAR...	177	1001...	Plaq...		1	0	0	C3	43,3	0	57,66	0	PCS...		
	792	13/0...	Modif		SILIC...	191	1001...	DRO...	LDM...	1	0	0	D10 ...	3,94	0	5,25	0	PCS...		

810

Rafraichir

Export

Imprimer

Facture

# Mes missions à Solti :

## Semaine 4 :

### Facture de sorti de produit :

Pour ce début de semaine on m'a demandé à nouveau de travailler sur une fonctionnalité relative aux factures. Cette fois-ci l'objectif était de pouvoir créer des factures à partir d'une sortie de produits.

#### *Onglet produits :*

Version 5.023 Copyright © Solti 2022													
Produits													
Id Famille Reference Designation Fournisseur Emplacement Total													
Id Famille Reference Designation Fournisseur Quantite Mini Maxi Emplacen PUA PTA PTV PUV													
2	CUIVRE	1235896...	MANCH...	VAN MA...	3	3	0	1,19	3,57	45000	15000		
3	CUIVRE	1256998...	COUDE ...		4	3		0	0	0	0		
4	CUIVRE	1235689...	COUDE ...	VAN MA...	8	5		1,24	9,92	16	2		
5	CUIVRE	1456998...	COUDE ...	VAN MA...	12	5		1,28	15,36	24	2		
6	CUIVRE	4456982...	COUDE ...	CEDEO	8	2		1,77	14,16	24	3		
8	CUIVRE	1255639...	REDUCTI...		6	3		0	0	0	0		
9	CUIVRE	1258998...	COULISS...		7	2		10,58	74,06	112	16		
10	CUIVRE	1256698...	MAMEL...	CEDEO	4	2		4,65	18,6	28	7		
11	CUIVRE	7458236...	MAMEL...	VAN MA...	6	3		5,15	30,9	48	8		
12	CUIVRE	2365889...	DIVERS ...		1	0		5	5	8	8		
13	CUIVRE	1256366...	MAMEL...		4	3		0	0	0	0		
14	CUIVRE	2569963...	REDUCTI...	VAN MA...	10	3		3,58	35,8	50	5		
15	CUIVRE	1256987...	MAMEL...		7	3		0	0	0	0		
16	CUIVRE	1235689...	TE DIAM ...	CEDEO	6	3		5,86	35,16	54	9		
17	CUIVRE	4125636...	MANCH...	VAN MA...	10	3		1,32	13,2	20	2		
18	CUIVRE	1255639...	COUDE ...	VAN MA...	7	3		3,54	24,78	35	5		
19	CUIVRE	1256987...	COUDE ...	VAN MA...	12	3		2,21	26,52	36	3		
20	CUIVRE	1452569...	COUDE ...	VAN MA...	25	3		1,87	46,75	75	3		
21	CUIVRE	1256369...	COUDE ...	VAN MA...	19	3		1,37	26,03	38	2		
22	CUIVRE	1256987...	TE DIAM ...	VAN MA...	16	3		2,42	38,72	64	4		
23	CUIVRE	1254895...	MAMEL...		16	3		1,93	30,88	48	3		
24	CUIVRE	1452369...	MAMEL...	VAN MA...	11	3		6,25	68,75	99	9		
25	CUIVRE	1258874...	MAMEL...	VAN MA...	10	5		2,41	24,1	40	4		
26	CUIVRE	1458236...	MAMEL...	VAN MA...	15	3		2,68	40,2	60	4		
28	CUIVRE	1425698...	COULISS...		6	2		8,47	50,82	78	13		

Commandé

Paniers

Rafraichir

Export

Alarmes

Entrée

Sortie

Nouveau

Modifier

Pour remplir les factures j'avait besoin de récupérer les informations de la société et du client. 2 classes existent qui permettent de récupérer les informations de chacun, CCompany et CContact.

```

522
523
524 1 référence
525 void MakeYodaDoc()
526 {
527     // Pour Matthieu
528     LibSQL_YODA m_GP = new LibSQL_YODA();
529
530     CCompany Comp = new CCompany();
531     CContact Cont = new CContact();
532     string sCompany = userControlProperties1.GetClient();
533
534     if (!Tools.IsValidString(sCompany))
535     {
536         bool resp = Tools.MyQuestion(this, "Pas de client. Continuer");
537         if (!resp) return;
538     }
539     else
540     {
541         int err1 = m_GP.Open();
542         if (err1 == 0)
543         {
544             Comp = m_GP.GetCompanyByName(sCompany);
545             if (Comp == null) MessageBox.Show("Client introuvable dans YODA: " + sCompany);
546             Cont = m_GP.GetCompanyContact(Comp.id);
547             if (Cont == null) MessageBox.Show("Contact introuvable dans YODA: " + sCompany);
548             m_GP.Close();
549         }
550         m_GP.Error(err1);
551     }
552
553     F_DocumentChoice MyDocForm = new F_DocumentChoice();

```

Ensuite je vais récupérer les informations de chaque produit grâce à une boucle foreach sur les lignes sélectionnées dans le tableau.

```

List<List<string>>> DevisTab1 = new List<List<string>>>();
foreach (DataGridViewRow row in DGV1.Rows)
{
    m_GSM.Open();
    CProduct Prod = m_GSM.GetProductByReference((string)row.Cells[0].Value);
    m_GSM.Close();
    if (Prod == null) break;
    List<string> rowTab = new List<string>()
    {
        Prod.Reference,
        Prod.Designation,
        (string)row.Cells[1].Value,
        Tools.ConvToString(Prod.PUA)
    };
    DevisTab1.Add(rowTab);
}

```

Pour finir je vais réutiliser les fonctions de la class CExcelDoc qui m'avait déjà servi les semaines précédentes afin de remplir la facture.

```
CExcelDoc ET = new CExcelDoc();

string src = Environment.CurrentDirectory + @"\DOC\" + FileName;

ET.OpenExcelDocument(src);

List<int> Réf = ET.FindInAllSheet("Designation", true);
if (Réf.Count == 0) return 1;

ET.FillTable(Réf[0] + 2, Réf[1], DevisTab1);
ET.DeleteTabRow("TOTAL HT");
ET.SaveAndClose();

SoltiProcess.Start(src);
return 0;
}
```

## Supprimer un document :

Une collègue à fait remarque qu'elle ne pouvait pas supprimer de facture/devis de client depuis le logiciel ont m'a donc confié la tâche de rendre ça possible.

Dans une boucle foreach je récupère l'ID correspondant à chaque ligne sélectionner pour le placer en paramètre d'une fonction permettant de supprimer un document de la base de données à partir de son ID puis je récupère le chemin du fichier et le supprime du pc.

```
private void dataGridViewExchanges_KeyDown(object sender, KeyEventArgs e)
{
    if (e.KeyCode != Keys.Delete) return;

    foreach (DataGridViewCell cell in dataGridViewExchanges.SelectedCells)
    {
        int row = cell.RowIndex;
        int index = dataGridViewExchanges.Columns["ID"].Index;
        string Value = dataGridViewExchanges.Rows[row].Cells[index].Value.ToString();

        if (cell == null) return;

        string Name = cell.Value.ToString();
        bool Resp = Tools.MyQuestion(this, "Voulez-vous vraiment supprimer " + Name + "?");
        if (!Resp) return;
        Resp = Tools.MyQuestion(this, "Toutes les données seront également supprimées.\r\nContinuer ? ");
        if (!Resp) return;

        int ID = Tools.ConvToInt(Value);
        DeleteDocument(ID);

        string FilePath = GetFilePath(Name);
        if (File.Exists(FilePath))
        {
            File.Delete(FilePath);
        }
        RefreshDocuments();
    }
}
```

## Conclusion

Encore une semaine très agréable à SOLTI, j'ai toujours quelque chose à faire et j'ai vraiment le sentiment d'en apprendre de plus en plus sur la programmation objet et le développement en c#.

## Niko Niko

Lundi	Mardi	Mercredi	Jeudi	Vendredi
