

Sommaire

1. Expression des besoins : pages 3 à 4
2. Conception : pages 5 à 7
3. Développement : pages 8 à 9
4. Mise en production : page 10
5. Bilan : page 11

1. Expression des besoins

1.1. Contexte, domaine, processus métier :

Vertech' est un acteur incontournable dans les solutions logicielles destinées aux verriers (fabricants de verre). Grâce à son large éventail de produits, le SIL est un package de logiciels, qui fournit aux acteurs d'une usine des indicateurs très précis et en temps réel sur la performance du site (nombres de bouteilles cassées, nombre de défaut par machines etc...). Ce partage de données et la traçabilité des produits mènent à une augmentation des cadences de production, une baisse des pertes et une amélioration de la satisfaction client. Il y a seulement un site de Vertech' en France et quelques acteurs à l'étranger.

Vertech' et leur SIL sont présent dans le monde entier :



1.2. Demandeur, acteurs, utilisateurs :

Le projet que m'a donné l'équipe dans lequel je travaille est d'intégrer un envoi de rapport de favoris récurrent par mail en arrière-plan dans leur application web SILXtractorV2. Certains clients qui avaient accès à cette fonctionnalité dans la première version de SILXtractor et ont demandé un retour de cette fonctionnalité dans la V2. L'équipe m'a donc donné ce projet puisqu'il ne représente pas une urgence ou une priorité.

1.3. Etude de l'existant, diagnostic :

L'ancienne version de SILXtractor utilisait Hangfire pour effectuer cette tâche, j'ai donc réutilisé le même outil.

1.4. Description de la demande, objectifs, bénéfices attendus :

L'objectif est d'intégrer un envoi de mail, en arrière plan de SILXtractorV2 avec comme contenu un rapport d'un favori enregistré par le client, et récurant.

1.6. Contraintes ou exigences (matérielles, techniques, délais, budget, ...) :

J'ai utilisé Microsoft Visual Studio pour développer mon API, ainsi que MongoDB pour ma base de données, Bitbucket pour m'aider à gérer mon développement et Git pour pouvoir travailler sur le logiciel SILXtractorV2 sans le modifier et pouvoir intégrer ma fonctionnalité dedans une fois terminée. J'ai aussi utilisé le package Hangfire pour pouvoir créer des actions récurrentes en arrière-plan de mon navigateur.

2. Conception

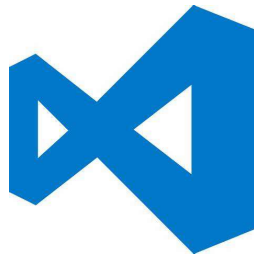
2.1. Description de la solution :

Spécification n°1 : En tant qu'utilisateur, je veux pouvoir paramétrer un envoi de rapport récurrent de mon/mes favoris.

Spécification n°2 : L'API doit s'intégrer correctement au logiciel.

2.2. Outils logiciels de la solution :

Logiciel éditeur de texte : Visual Studio Code



Logiciel de base de données : MongoDB



Service web d'hébergement et de gestion de développement logiciel :
Bitbucket



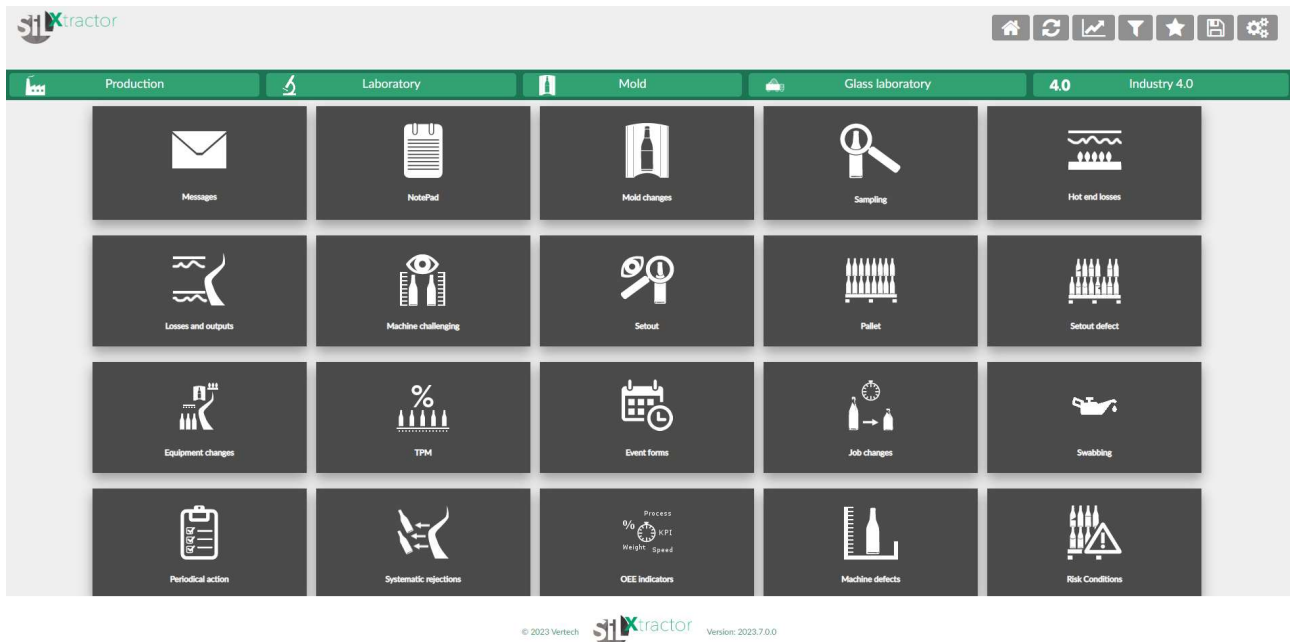
Logiciel de gestion de versions décentralisé : Git



NuGet Package permettant d'effectuer des tâches en arrière-plan dans les applications .NET et .NET Core : Hangfire

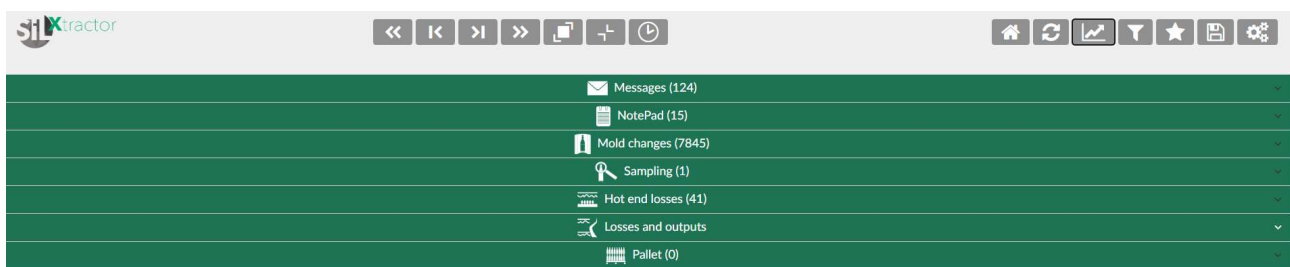
2.6. Interface de SILXtractor :

Vertech' développe des solutions pour les fabricants de verre mondiaux et est leader dans ce domaine, SILXtractor permet aux clients de visionner et d'effectuer des traitements sur toutes les données récupérées par leurs machines.



La solution SILXtractorV2 est disposée sous forme de tuiles à sélectionner.

Ensuite il est possible de les afficher sous forme de liste déroulantes :



The screenshot shows the SILXtractor V2 interface with a data table for 'Mold changes (7845)'. The table has columns for Plant, Line, Production machine, Job order, Article, Article reference, Section, Gob, Gob I, Cavity, Plunger, Mold number, Mold reference, Serial number, Mold type label, Mounting date, and Unmounting date. The table contains 10 rows of data.

Plant	Line	Production machine	Job order	Article	Article reference	Section	Gob	Gob I	Cavity	Plunger	Mold number	Mold reference	Serial number	Mold type label	Mounting date	Unmounting date	Re
Plant	S01	IS Machine	201912	ART1	ART1	1	1	B	S1Int.		10		D	Blow	11/19/2019 2:54:05 PM		
Plant	S01	IS Machine	201912	ART1	ART1	1	2	A	S1Ext.		11		D	Blow	11/19/2019 2:54:05 PM		
Plant	S01	IS Machine	201912	ART1	ART1	2	1	B	S2Int.		12		D	Blow	11/19/2019 2:54:05 PM		
Plant	S01	IS Machine	201912	ART1	ART1	2	2	A	S2Ext.		13		D	Blow	11/19/2019 2:54:05 PM		
Plant	S01	IS Machine	201912	ART1	ART1	3	1	B	S3Int.		14		D	Blow	11/19/2019 2:54:05 PM		
Plant	S01	IS Machine	201912	ART1	ART1	3	2	A	S3Ext.		15		D	Blow	11/19/2019 2:54:05 PM		
Plant	S01	IS Machine	201912	ART1	ART1	4	1	B	S4Int.		16		D	Blow	11/19/2019 2:54:05 PM		
Plant	S01	IS Machine	201912	ART1	ART1	4	2	A	S4Ext.		17		D	Blow	11/19/2019 2:54:05 PM		
Plant	S01	IS Machine	201912	ART1	ART1	5	1	B	S5Int.		18		D	Blow	11/19/2019 2:54:05 PM		
Plant	S01	IS Machine	201912	ART1	ART1	5	2	A	S5Ext.		19		D	Blow	11/19/2019 2:54:05 PM		
Plant	S01	IS Machine	201912	ART1	ART1	6	1	B	S6Int.		20		D	Blow	11/19/2019 2:54:05 PM		

Il dispose aussi d'ajout de filtre et de sélection ainsi que d'un téléchargement des données, en respectant les traitements effectués dessus, dans un fichier Excel :



2.7. Conduite de projet:

Pour la conduite de projet, chez Vertech' Jira est utilisé. Il permet de décomposer le travail en tâches qui peuvent contenir des sous-tâches (ci-dessous les sous-tâches de mon projet) :

Sous-tâches							+ ...	
1.	✓ Analyse faisabilité	↔	TERMINÉ	▼	SILXtractorV2	Non attribuée	100%	<div><div></div></div>
2.	Ossature	↔	RC / QUALIF	▼	SILXtractorV2	Non attribuée	56%	<div><div></div></div>
3.	Adapter la méthode pour récupérer les favoris	↔	EN COURS	▼	SILXtractorV2	Louis BOURBON	0%	<div><div></div></div>
4.	Adapter la méthode pour générer les fichiers Excel	↔	BACKLOG	▼	SILXtractorV2	Non attribuée	0%	<div><div></div></div>
5.	Adapter l'envoi des emails	↔	BACKLOG	▼	SILXtractorV2	Non attribuée	0%	<div><div></div></div>
6.	Modifier le Powershell	↔	BACKLOG	▼	SILXtractorV2	Non attribuée	0%	<div><div></div></div>
7.	Documentation	↔	BACKLOG	▼	SILXtractorV2	Non attribuée	0%	<div><div></div></div>

3. Développement

Répondre Répondre à tous Transférer
jeu. 25/05/2023 13:20

 louis.bourbon@verttech.eu

À louis.bourbon@verttech.eu

 ScheduledReports.xml
365 octets

```
public void SendMail()
{
    //Instanciation du client
    SmtplibClient smtpClient = new SmtplibClient("192.168.1.1", 25);
    //On indique au client d'utiliser les informations qu'on va lui fournir
    smtpClient.UseDefaultCredentials = true;
    //Ajout des informations de connexion
    smtpClient.Credentials = new System.Net.NetworkCredential("louis.bourbon@verttech.eu", "Ta1chocol@t");
    //On indique que l'on envoie le mail par le réseau
    smtpClient.DeliveryMethod = SmtplibDeliveryMethod.Network;
    //On active le protocole SSL
    smtpClient.EnableSsl = false;

    MailMessage mail = new MailMessage();
    //Expéditeur
    mail.From = new MailAddress("louis.bourbon@verttech.eu");

    mail.Attachments.Add(new Attachment("ScheduledReports.xml", "application/xml"));
    //Destinataire
    mail.To.Add(new MailAddress("louis.bourbon@verttech.eu"));

    smtpClient.Send(mail);
}
```

Code C# de mon envoi de mail intégrant un fichier XML.

Ligne de code permettant d'ajouter un job récurant dans Hangfire, ici l'envoi de mon mail toute les minutes.

```
RecurringJob.AddOrUpdate("myrecurringjob", () => SendMail(), Cron.Minutely);
```

louis.bourbon@verte...	jeu. 13:20
louis.bourbon@verte...	jeu. 12:03
louis.bourbon@verte...	jeu. 12:02
louis.bourbon@verte...	jeu. 12:01
louis.bourbon@verte...	jeu. 12:00
louis.bourbon@verte...	jeu. 11:59
louis.bourbon@verte...	jeu. 11:58
louis.bourbon@verte...	jeu. 11:57
louis.bourbon@verte...	jeu. 11:56

J'ai réussi à faire un exemple fonctionnel intégré dans SILXtractorV2, mais avec des liens en dures.

 lun. 05/06/2023 16:22
louis.bourbon@verttech.eu

À louis.bourbon@verttech.eu

 Changement-de-moules_20230605_162129.xlsx
14 KB

```
public string CallExcelExport()
{
    string datatype = "MouldChange";
    string dateFormat = "M/d/yyyy h:mm:ss tt";
    string shortDateFormat = "M/d/yyyy";
    string culture = "en-US";
    string startDate = "1685484000000";
    string endDate = "1685570399000";
    string chartData = "plantName";
    string chartType = "column";
    string chartDataSort = "fieldCount";
    string chartSortType = "desc";
    string spcMode = null;
    string[] hiddenColumns = { "eqmcleunik", "mouldId" };
    long[] JobOrder = { 3246, 55980 };
    GridSortItem[] sort = new GridSortItem[] { };
    GridFilterItem[] filter = new GridFilterItem[] { };
    string[] dataToRequest = new string[] { };
    string eventsFilters = null;

    string filename = ExcelController.ManageExport(datatype, dateFormat, shortDateFormat, culture, startDate, endDate, chartData, chartType, chartDataSort, chartSortType, spcMode, hiddenColumns, JobOrder, sort, filter, dataToRequest, eventsFilters);

    return filename;
}
```


J'ai remplacé la plus part des liens en dur par des fonctions et des objet permettant de récupérer les informations nécessaire pour accéder aux favori. J'arrive toujours à envoyer mon mail avec mon fichier Excel contenant les informations récupérées.

```
string datatype = favorite.CategoryData.categories[0].Name;
string dateFormat = "M/d/yyyy h:mm:ss tt";
string shortDateFormat = "M/d/yyyy";
string culture = "en-US";
string startDate = favorite.FilterData.Date.Start.ToString();
string endDate = favorite.FilterData.Date.End.ToString();
string chartData = favorite.CategoryData.categories[0].Grids[0].Chart.chartData.ToString();
string chartType = favorite.CategoryData.categories[0].Grids[0].Chart.chartType.ToString();
string chartDataSort = favorite.CategoryData.categories[0].Grids[0].Chart.chartDataSort.ToString();
string chartSortType = favorite.CategoryData.categories[0].Grids[0].Chart.chartSortType.ToString();

Dictionary<bllSpecificFilter.FilterType, List<object>>> ao_SpecificFilter = new Dictionary<bllSpecificFilter.FilterType, List<object>>>();
ao_SpecificFilter.Add(bllSpecificFilter.FilterType.FILTER_PRODUCTION_TOOL_ORDER, favorite.FilterData.Order.ToList<object>());

bllProductionToolManagement.GetProductionTool(Convert.ToDateTime(startDate), Convert.ToDateTime(endDate), ao_SpecificFilter, 0);

string spcMode = null;
string[] hiddenColumns = { "eqmcleunik", "mouldId" };
long[] JobOrder = { 3246, 55980 };
GridSortItem[] sort = new GridSortItem[] { };
GridFilterItem[] filter = new GridFilterItem[] { };
string[] dataToRequest = new string[] { };
string eventsFilters = null;

string filename = ExcelController.ManageExport(datatype, dateFormat, shortDateFormat, culture, startDate, endDate, chartData, chartType, chartDataSort,
chartSortType, spcMode, hiddenColumns, JobOrder, sort, filter, dataToRequest, eventsFilters);

return filename;
```



mer. 07/06/2023 15:45

louis.bourbon@verttech.eu

À louis.bourbon@verttech.eu



Messages_20230607_154429.xlsx
13 KB

```
foreach (Category items in favorite.CategoryData.categories)
{
    foreach (Grid item in items.Grids)
    {
```

J'ai rajouté des foreach pour pouvoir traiter toute les données enregistré dans les favoris (précédemment le rang 0 était inscrit en dur dans les paramètres pour pouvoir tester le programme).



jeu. 15/06/2023 14:58

louis.bourbon@verttech.eu

À louis.bourbon@verttech.eu



Palette_20230615_145801.xlsx
19 KB



Palette_grille_groupee_20230615_145810.xlsx
14 KB

```
string elementToInsert = ExcelController.ManageExport(datatype, dateFormat, shortDateFormat, culture,
JobOrders, sort, filter, dataToRequest, eventsFilters);
int insertIndex = G;

string[] newArray = new string[filename.Length + 1];
Array.Copy(filename, newArray, insertIndex);
newArray[insertIndex] = elementToInsert;
Array.Copy(filename, insertIndex, newArray, insertIndex + 1, filename.Length - insertIndex);

filename = newArray;

G++;
```

Jeu. 15/06/2023 14:58
 L louis.bourbon@verttech.eu
 À louis.bourbon@verttech.eu

Palette_20230615_145801.xlsx 19 KB
 Palette_grille_groupee_20230615_145810.xlsx 14 KB

Dans mes foreach j'ai modifié la variable filename pour qu'elle comporte maintenant une liste de fichiers à envoyer dans un même mail.

Certaines catégories de favoris doivent-être gérées autrement, par exemple ici avec les grids pivot (type de grille) :

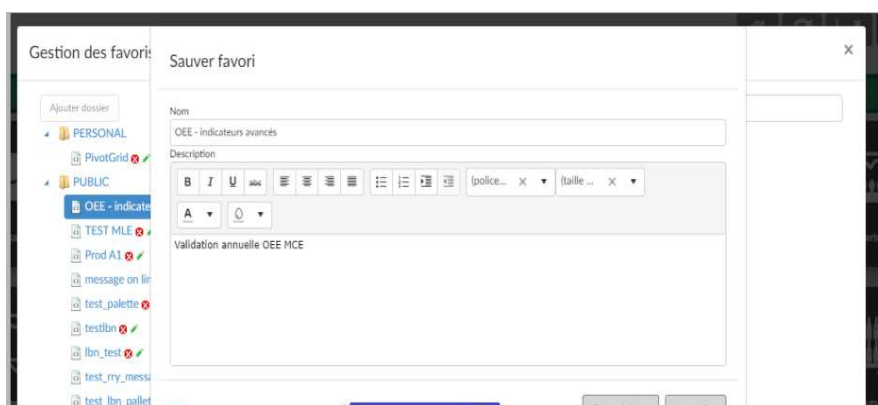
```
if (item.Name.Contains("_Pivot_"))
{
    string elementToInsert = GenerateExcelReportForPivotGrid(pathExcelFolder, datatype, domain, favpath);
    int insertIndex = G;

    string[] newArray = new string[filename.Length + 1];
    Array.Copy(filename, newArray, insertIndex);
    newArray[insertIndex] = elementToInsert;
    Array.Copy(filename, insertIndex, newArray, insertIndex + 1, filename.Length - insertIndex);

    filename = newArray;
}
```

3.3. Difficultés rencontrées (Bugs, Reste à faire) :

Il reste maintenant l'interface utilisateur à développer (maquette ci-dessous) :



4. Mise en production

4.1. Tests (unitaires, fonctionnels, intégration) :

La solution que j'ai développée est totalement fonctionnelle, mais elle n'est pas encore intégrée au code de SILXtractorV2 pour le moment.

Je suis en train de supprimer petit à petit tout les liens en durs pour automatiser ma solution.

Pratiquement tout les liens en durs ont été remplacé par des fonctions récupérant les données.

5. Bilan

Pour l'instant le projet se déroule bien, j'effectue des recherches pour mener à bien mon travail.

La semaine dernière, j'ai touché à des fonctions déjà programmées dans le logiciel SILXtractorV2, ce qui m'a permis de découvrir un peu plus la façon de faire de l'entreprise et m'a appris comment faire des recherche dans un code plutôt complexe.

Cette semaine j'ai bien avancé sur mon projet, il ne me reste plus que quelques petits détails à terminer et il me reste aussi l'interface utilisateur à créer.