



UNIVERSITÉ  
CAEN  
NORMANDIE



## RAPPORT DE STAGE DE LICENCE L3 INFORMATIQUE

**Sujet : Réalisation d'une application d'aide à la formation au traitement thermique.  
Site industriel de STELLANTIS CAEN  
Du 4 avril au 27 mai 2022**

Stagiaire :  
Malaubier Léo  
N° étudiant : 21903309  
21903309@etu.unicaen.fr  
Malaubier.leo@gmail.com

Tuteur d'entreprise :  
Alain Gauthier  
[alain.gauthier@stellantis.com](mailto:alain.gauthier@stellantis.com)

# STELLANTIS



## Table des matières

Remerciements.....	3
Résumé.....	4
1. Introduction.....	5
2. Le contexte.....	5
2.1 L'entreprise.....	5
2.2 L'équipe Projet.....	6
2.3 Communication au sein de l'équipe.....	6
2.4 Ressources fournies et/ou à utiliser.....	7
3 Conception et réalisation de l'application.....	8
3.1 objectif et avancement .....	8
3.2 Étude du besoin.....	8
3.2.1 Le contexte .....	8
3.2.2 Les besoins principaux.....	8
3.2.3 Choix de développement.....	9
3.2.4 Choix des technologies .....	10
3.2.4.1 Choix du langage de programmation .....	10
3.2.4.2 Comparaison des langages.....	11
3.2.4.3 Choix des bibliothèques .....	12
3.3 Cahier des charges.....	13
3.4 Modélisation de l'application.....	15
3.4.1 schéma interface (étape de recherche).....	15
3.4.2 Interface principale (la simulation première version).....	16
3.4.3 Diagramme de classe du logiciel .....	17

4 Phase de réalisation .....	20
4.1 Description pour utilisation : .....	20
4.1.1 Version 1 : .....	20
4.1.1.1 LA NAVIGATION ENTRE LES MENUS .....	21
4.1.1.2 LA SIMULATION .....	25
4.1.2 Version 2 .....	26
4.2 PROGRAMMATION DETAILLEE: .....	28
4.3 Amélioration possibles .....	32
5 CONCLUSION .....	33
6 ANNEXES .....	34
6.1 Sources .....	34
6.2 Lexique.....	34

## Remerciements

Je tiens, en premier lieu, à remercier le personnel de STELLANTIS du site de CAEN pour l'accueil chaleureux qui m'a été fait, pour l'accompagnement et les échanges instructifs dont j'ai bénéficié tout au long de mon stage.

Je remercie Alain Gauthier, mon maître de stage, de m'avoir accompagné durant les deux mois de stage, trop court à mon avis.

Je tiens aussi à remercier toute l'équipe du Laboratoire, pour leur accueil et leur bonne humeur, ainsi que Roseline Monnier et Christophe Rivère pour leur conseils éclairés.

## Résumé

Le site de CAEN fabrique pour toutes les marques automobiles du Groupe STELLANTIS des transmissions et des berceaux en acier.

L'activité « développements informatiques » au sein de STELLANTIS est très majoritairement sous-traitée, et sujette à financement. Les techniciens du laboratoire de métrologie du site ont donc tout naturellement fait appel à un stagiaire du domaine Informatique pour créer et développer un logiciel pédagogique sur le « traitement thermique » à destination des nouveaux arrivants sur le site STELLANTIS de CAEN.

L'objet de ce rapport est de présenter les différentes étapes de développement de ce logiciel, de cette application, qui devait avoir pour caractéristiques principales :

- ⇒ Être ergonomique, les utilisateurs « cible » étant des métallurgistes n'ayant pas de compétence dans le fonctionnement d'un ordinateur ni le développement d'applications.
- ⇒ Être modulable, afin de pouvoir être adapté à plusieurs formations, notamment à pouvoir décliner les différentes méthodes de traitements thermiques applicables aux transmissions fabriquées sur le site.

Ces principes constituaient la charpente de ce « cahier des charges » de développement du logiciel, les différentes étapes des traitements métallurgiques devant pouvoir y être greffées en fonction des besoins de mon tuteur. Nous verrons dans la suite qu'effectivement les besoins ont été évolutifs, tout au long du stage.

Dans ce déroulé, nous verrons également qu'un stade fonctionnel du logiciel a été atteint et figé sous le titre de « logiciel V1 », deux semaines avant la fin du stage, répondant aux besoins alors exprimés. Le langage utilisé est « python ». Les lignes de programmation sont explicitées en annexe de ce rapport.

La version « Logiciel V2 » correspond à une autre direction souhaitée par la suite par le maître de stage. Cette seconde version n'est pas fonctionnelle mais constitue une bonne base de reprise du développement du logiciel, si nécessaire.

## 1. Introduction.

La 3<sup>e</sup> et dernière année de Licence informatique à l'Université de Caen se conclut par un stage de 2 mois en immersion en entreprise. L'objet principal de ce stage est de mettre en pratique concrètement les connaissances acquises à l'Université, dans un cadre professionnel. Un autre objet de ce stage est d'appréhender un début d'insertion professionnelle.

La mission de deux mois qui m'a été confiée au sein du Groupe STELLANTIS a été de réaliser un logiciel didactique en langage python pour support à la formation en traitement thermique des nouveaux arrivants.

Aucun synoptique ni support papier n'existait sur le sujet. L'outil répondant au besoin exprimé a donc été créé « à partir de 0 ».

(\*) les mots suivis de (\*) trouvent leur définition dans le lexique en fin de rapport.

## 2. Le contexte

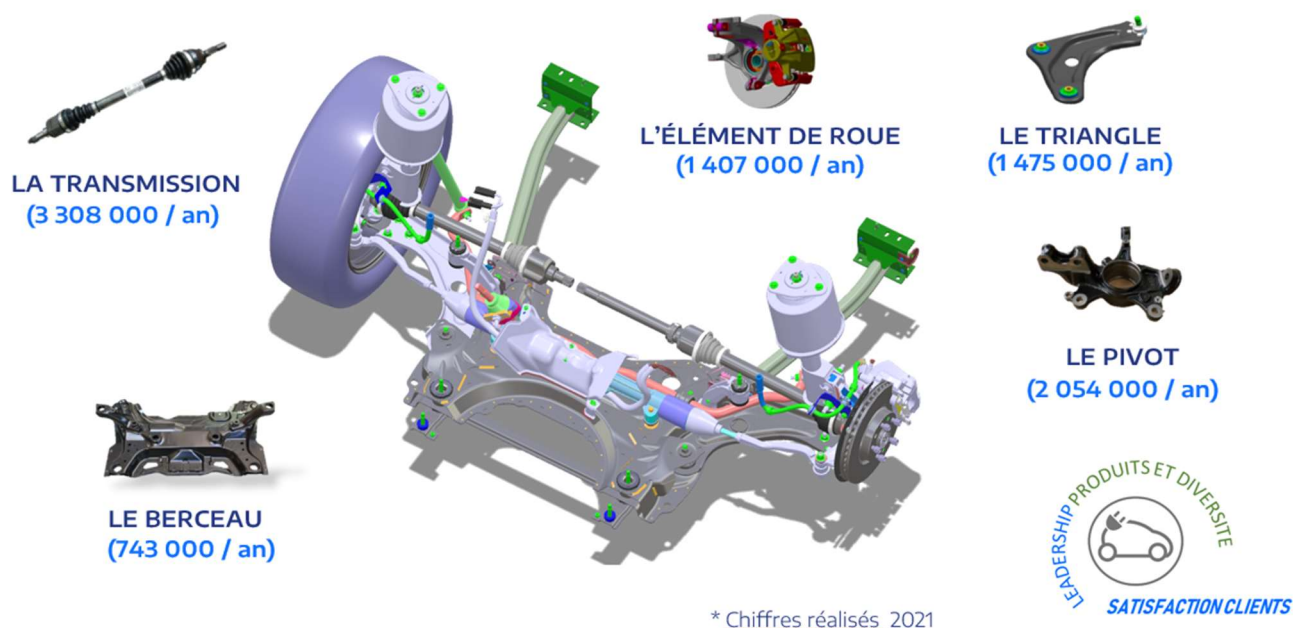
### 2.1 L'entreprise



STELLANTIS est un constructeur automobile multinational, présent sur les 5 continents et ayant des sites industriels dans 30 pays. STELLANTIS a été fondé le 16 janvier 2021 par la fusion de PSA Peugeot-Citroën et de Fiat Chrysler Automobiles. Le Groupe représente plus de 281 500 salariés pour un chiffre d'affaires de 152 milliards d'euros. C'est le 6<sup>ème</sup> producteur mondial de voitures.

Le Groupe s'engage à façonner la mobilité du futur en se basant sur les performances environnementales, la sécurité et la connectivité des véhicules. Ainsi, STELLANTIS vise à conquérir désormais le marché de l'électrique et des véhicules propres.

L'usine du site de Caen a pour vocation de produire des organes mécaniques équipant les véhicules de l'ensemble du Groupe. Avec un effectif de 1200 salariés, l'usine de Caen reste compétitive face aux autres équipementiers automobiles. Cette usine propose 5 types de pièces, produites et assemblées sur place : transmission, berceau, pivot, triangles, éléments de roue.



Différents traitements thermiques sont utilisés dans le processus de fabrication, avec pour objectif d'améliorer la solidité des matériaux.

Le Laboratoire métallurgique du site, auquel j'ai été affecté, a pour mission de vérifier la conformité des pièces testées et ainsi de garantir la représentativité des outils de production, notamment les fours pour traitement thermique.

## 2.2 L'équipe Projet

L'équipe est composée de 2 membres :

- ⇒ Alain Gauthier, technicien du laboratoire métallurgique STELLANTIS, mon maître de stage ;
- ⇒ Léo Malaubier, stagiaire et auteur de ce rapport, Étudiant en Informatique L3, Université de CAEN. Je référais directement à Alain Gauthier.

Madame Roseline Monnier et Monsieur Christophe Rivere, tous deux « Professionnels » formés à l'analyse métallurgique ont répondu à mes questions sur la métallurgie. Ils n'ont pas pris part à la création du programme.

Un autre stagiaire informaticien, Martin Rouault, étudiant en IUT d'informatique de Clermont Ferrand en Auvergne, a su m'apporter parfois un regard extérieur. Son sujet de stage était différent du mien.

## 2.3 Communication au sein de l'équipe

Tous les collaborateurs du laboratoire, dont moi-même, étions regroupés en open-space. L'équipe était très disponible, les échanges verbaux possibles. Ils m'ont permis d'obtenir les informations indispensables à la compréhension du fonctionnement des moyens de fabrication.

Autre mode de communication, les échanges de mails avec Mr Gauthier, à des fins

d'échange de fichier lorsque c'était nécessaire.

## 2.4 Ressources fournies et/ou à utiliser

Les ressources principales matérielles étaient personnelles : ordinateur sous linux/windows (en dual-boot\*). J'ai pu utiliser l'accès internet du réseau « Stellantis Guest », réseau parallèle au réseau principal, afin de garantir la sécurité informatique.

Pour l'aspect « informatique », j'ai dû me baser sur mes seules connaissances personnelles. Pour l'aspect « métallurgie », un support papier et pdf de la « pré-formation » au traitement thermique par induction m'a été fourni.



## 3 Conception et réalisation de l'application

### 3.1 objectif et avancement

La première phase du stage a consisté à expliciter les attentes de mon maître de stage. La seconde a été de réaliser seul une application fonctionnelle, sans application pré existante. Le tout en un laps de temps relativement court (2 mois).

Le besoin principal réside en une application qui contienne suffisamment d'éléments constitutifs d'une formation « métallurgique » complète (fonctionnement optimum) ou au moins qui permettent la poursuite future du développement.

Pour ce faire, j'ai provoqué un certain nombre de réunions avec Mr Gauthier pour bien cerner ses attentes, pour établir un cahier des charges et faire des recherches sur des bibliothèques existantes afin de gagner du temps.

Puis j'ai pu commencer le développement de l'application.

### 3.2 Étude du besoin

#### 3.2.1 Le contexte

La formation des nouveaux arrivants est jusqu'à présent principalement théorique : elle se pratique sur la base d'un document papier et pdf. Puis par l'apprentissage et l'expérience de l'utilisation des machines.

Mr Gauthier a, par le passé, déjà tenté une modernisation de cette formation via la demande de développement d'outils par des stagiaires. Cependant, les attentes en termes de formation ont évoluées, et les outils alors développés ne correspondent plus au besoin d'une part et ne sont pas adaptatifs d'autre part pour coller à la nouvelle demande.

Le nouveau moyen de formation à développer devait tenir compte de cet aspect « évolutif ».

#### 3.2.2 Les besoins principaux

Après quelques entretiens avec Mr Gauthier, il est apparu rapidement que le besoin n'était pas « une application spécialisée pour une seule formation », mais bien une application polyvalente permettant d'aborder plusieurs formations différentes.

L'application devait également permettre de représenter les conséquences d'actions diverses sur le métal, tout en affichant des aides et indications à suivre.

Cette application devait d'être :

=> Modulable pour recevoir différents types de formation, avec une interface qui s'adapte selon l'utilisation.

=> Ergonomique, tant pour l'utilisateur que pour le professeur ;

=> Ne pas prendre trop de place pour qu'elle soit facilement utilisable sur des clefs USB

=> Pouvoir fonctionner sur des machines peu puissantes et/ou des ordinateurs type raspberry. Bref, de pouvoir fonctionner sous n'importe quel support.

En outre, il m'a été demandé de faire un logiciel capable de simuler une machine qu'elle qu'elle soit, chacune possédant différents paramètres, tout en concevant une utilisation



instructeur / élèves simple.

Le tout, devant être développé dans les temps qui impartis.

### 3.2.3 Choix de développement

Pour pouvoir permettre de s'adapter au mieux aux besoins principaux, j'ai dû faire des choix de développement, qui ont été décidés en début de stage, avec accord du maître de stage.

Les solutions que j'ai proposées sont telles que :

#### MODE DE CODAGE

- ⇒ Dans la mesure où les formations proposées sont diverses, il n'est pas codé en brut les informations que doivent contenir les simulations. De plus, comme je ne suis pas expert en formation « métallurgie », malgré toute l'aide apportée par l'équipe, le codage en brut m'aurait pris beaucoup trop de temps.
- ⇒ En observant le travail de Mr Gauthier, je me suis aperçu que l'utilisation de tableur Excel prédominait dans l'échange d'information avec le Laboratoire et les autres équipes de l'usine. Aussi en me référant aux systèmes de cartouche des anciens jeux console, j'ai pensé que créer un logiciel capable de lire un tableur contenant toutes les informations nécessaires (comme une base de données ou une sauvegarde), permettrait de gagner du temps et d'avoir une application qui puisse faire n'importe quelle formation.

Ainsi, pour avoir une application modulable, **un système de lecture de fichier de cours me parut être une bonne solution.**

**ERGONOMIE :** pour ce qui est de l'ergonomie, j'utilise la règle des 3 clics pour déterminer si une interface est trop compliquée ou pas. Et veille à ce qu'il n'y est pas trop d'éléments affichés à l'écran.

**TAILLE :** L'utilisation de logiciel sur clef USB n'est plus un problème de nos jours. Car la taille des clef USB s'exprime facilement en Go. Mais pour tout de même respecter cette demande, j'ai formulé l'idée de **ne pas intégrer d'image pré-faite qui pourrait prendre de la place.**

**SUPPORT :** Enfin, pour ce qui est du fonctionnement du logiciel sur des machines peu puissantes, là aussi nous n'avons plus trop de problèmes.

Cependant, une solution m'est venue : de la même manière que l'on lirait un fichier Excel (xlsx\*), l'interface graphique n'aurait pas besoin de se générer à chaque utilisation. C'est-à-dire que si l'on arrive à générer un fichier qui sauvegarderait le contenu du cours sans avoir à relire le cours à chaque fois et à simplifier la recherche d'élément, on aurait alors un logiciel plus performant.

C'est pourquoi j'ai décidé d'incorporer au logiciel un générateur XML \*

Pour gérer l'interface graphique.

Cela a pour avantage une lecture de donnée plus simple, et de pouvoir changer l'interface graphique facilement sans avoir à refaire toute la partie gestion des fichiers Excel.

Ainsi, si l'on décide de vouloir changer de librairie graphique, **alors il n'y aura pas à refaire l'entièreté du programme. Et le programme gagne en temps de chargement.**

### 3.2.4 Choix des technologies

#### 3.2.4.1 Choix du langage de programmation

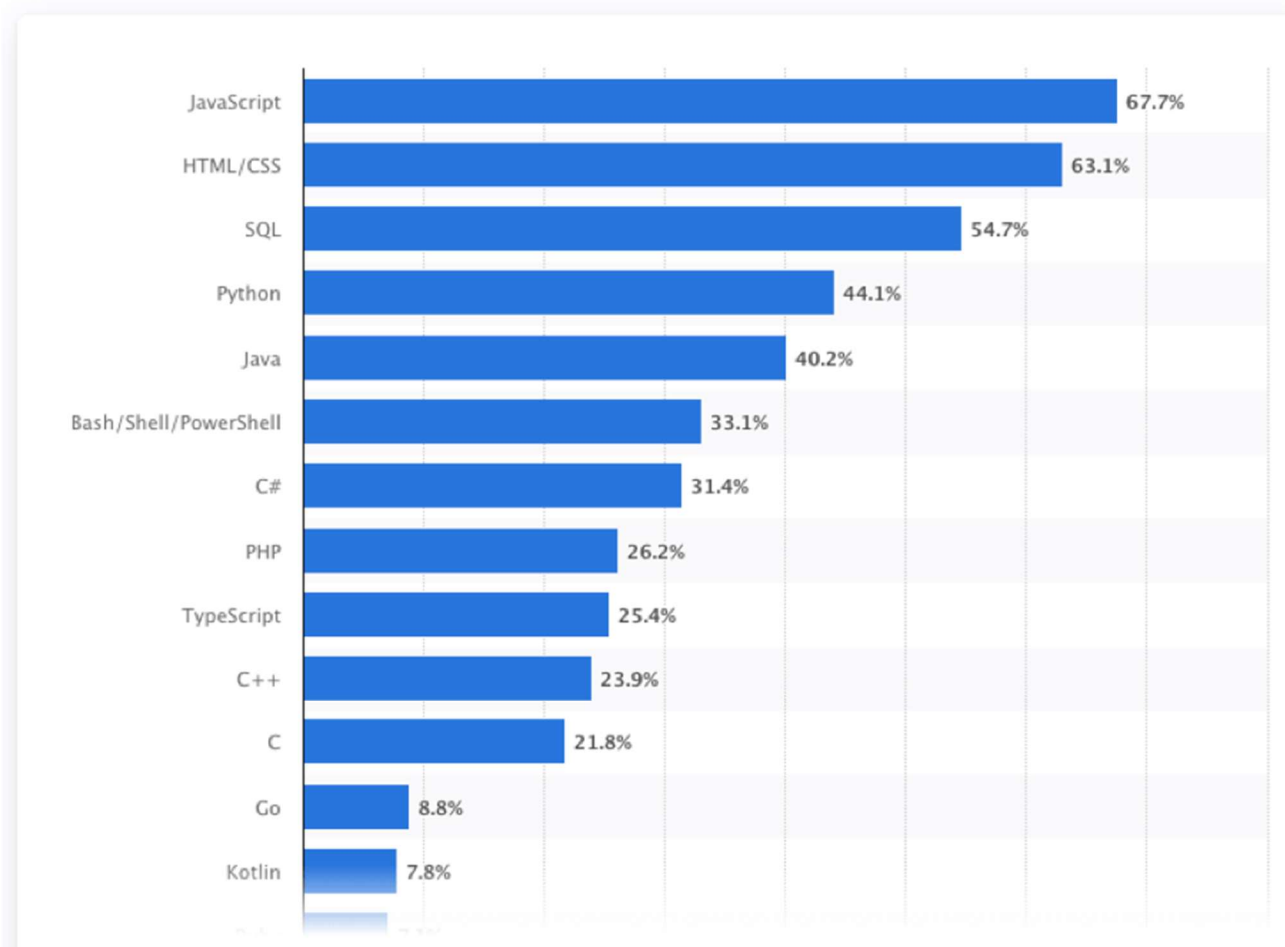
Pour ce projet, le choix du langage de programmation est libre. Il n'y a pas de contrainte sur l'utilisation d'un langage précis.

Cependant c'est un choix qui reste important, car il va influencer sur la manière dont on devra développer le projet.

Sur le site de Back4App, lorsque l'on recherche quel langage est le plus populaire pour les développeurs, nous retrouvons le graphique ci dessous.

(représentation graphique des langages les plus utilisés)

#### Most used programming languages among developers



D'après ce graphique, les langages web sont les plus utilisés dans le top 3 du classement. J'aurais pu faire une application web possédant une base de donnée SQL.

Cependant, le but de l'application est que l'enseignant (Mr Gauthier) puisse facilement rajouter des cours sans avoir à programmer quoi que ce soit. Et créer un formulaire pour remplir une base de données m'aurait pris beaucoup trop de temps et n'aurait pas pu faire la partie principale de l'application en elle-même.

**Donc j'ai retenu les langages qui permettent de faire une application rapide c'est-à-dire : Python, Java, C#, C++ et C.**

Choisir un langage très utilisé permet de trouver une communauté de développeurs et un support qui propose plusieurs solutions lors du codage de l'application, pour moi, ou pour un futur développeur.

#### 3.2.4.2 Comparaison des langages.

**PYTHON** : créé par Guido van Rossum, avec une première version en 1991, python est un langage de programmation interprété et multiplateformes. Il permet le codage en orienté objet\* et propose une gestion automatique de la mémoire. Et inclut un système de gestion d'exceptions. Il est idéal pour l'écriture de scripts et le développement rapide d'applications.

**JAVA** : créé par James Gosling et Patrick Naughton, présenté officiellement java en 1995. Java est désormais détenu par l'entreprise Oracle. C'est un langage qui permet d'être compilé. Qui est exécuté dans une machine virtuelle. Il est similaire au C++ mais plus lent. Permet aussi une gestion simplifiée de la mémoire.

**C#** : est un langage de programmation orientée objet, commercialisé par Microsoft depuis 2002 et destiné au développement sur les plateformes Microsoft. C'est un dérivé du C++ et proche de java dont il reprend la syntaxe. Il possède des notions telles que la surcharge des opérateurs, les indexeurs et les délégués. Il est utilisé notamment pour développer des applications web.

**C++** : est un langage de programmation compilé dont la programmation procédurale, la programmation orientée objet. Ses bonnes performances, et sa compatibilité avec le C en font un des langages de programmation les plus utilisés dans les applications où la performance prime.

**C** : est un langage bas niveau, NON orienté objet, c'est-à-dire que l'on ne peut faire que des fonctions, pas de classe. Inventé au début des années 1970, C est devenu le langage le plus utilisé. Il est à la base de C++, C# et Java. Permet une gestion de la mémoire avancée, peut créer certains risques si la mémoire est mal gérée.

**Faire une application en Python me paraît une bonne solution**, puisque c'est un langage auquel je me suis intéressé, que j'ai étudié à l'Université. Python me permettrait d'avoir un résultat rapide, avec une interface graphique et une utilisation de librairie facile à trouver et à installer.

Pour ce qui est de Java, C#, C++, ce sont des langages de programmation orientés objet\* que je connais moins (pour le C# et le C++) et pour lesquels je ne connais pas de librairie graphique. Ce qui me retarderait pour avoir un rendu rapidement.

Le C# étant plus destiné à une utilisation sous Windows, j'ai préféré ne pas m'y intéresser. Le C n'étant pas orienté « objet », j'ai préféré ne pas l'utiliser. Et cela m'aurait pris trop de temps.

J'ai donc choisi de coder en Python.

### 3.2.4.3 Choix des bibliothèques

Python possède une multitude de bibliothèques permettant d'avoir des outils simples à utiliser pour avancer rapidement dans le développement d'une application.

Pour les parties graphiques, il existe une multitude de bibliothèques comme pyOpenGL, PyQt, pygame et tkinter.

Par soucis de simplicité, j'ai utilisé tkinter qui fournit des outils simples pour créer des fenêtres. Ce qui m'a permis de créer rapidement un rendu graphique. J'avais déjà des connaissances dans l'utilisation de tkinter ce qui m'a permis d'être plus efficace.

Pour la lecture et l'écriture de fichiers tableurs, je vais utiliser pandas. C'est un outil qui regroupe le traitement des tableur Excel (xlsx) et d'autres formats comme le CSV\*.

Pour quoi l'utilisation d'autre format ?

Excel est un format propriétaire à Microsoft qui peut être modifié drastiquement par l'entreprise. Ce qui peut produire des problèmes de compatibilité entre le logiciel et les nouveaux formats. (il y a eu plusieurs formats pour Excel comme le xls, xlt)

Donc, je propose d'incorporer au logiciel un convertisseur Excel xlsx vers un format open source qui met plus de temps à chanter, le CSV. (utilisé par Libre office).

Le logiciel est donc entièrement compatible pour les fichiers au format xlsx (Excel) et csv.

Pour la lecture de fichier xml (pour l'affichage), je vais utiliser la bibliothèque lxml qui permet des recherches efficaces dans des arbres xml\*.

Par soucis de bien faire, je vais utiliser la bibliothèque logging. Logging me permet remplacer les print qui servent à suivre l'exécution du logiciel. En les hiérarchisant et en les stockant dans un fichier .log.

On retrouve principalement les informations : info, warning et debug lors du suivi du logiciel. (debug, info, warning créer la hiérarchie des informations dans le fichier log).

Cela pourra permettre de sauvegarder l'exécution d'un programme et de savoir à quel moment le programme a eu une erreur lors d'une utilisation normale.

Aussi la bibliothèque Numpy est utilisée pour la gestion de listes d'informations.

Cette bibliothèque permet la simplification de calculs et de gestion de listes.

### 3.3 Cahier des charges

La préparation du cahier des charges est l'une des parties importantes du projet. Elle permettra d'organiser le travail à effectuer et permet de ne pas s'égarer dans sa conception. Et permet également de mettre par écrit le fonctionnement et les fonctionnalités du logiciel.

Dans mon cas, partant de 0, il m'a été difficile de créer dès le début un cahier des charges complet. C'est pourquoi il a évolué au long de ces 2 mois.

Voici en dessous un cahier des charge rempli (regroupe tout le contenu du stage) :  
Les priorités vertes ont été prévues dès le début ;  
Les priorités en bleu sont les fonctionnalités rajoutées au cours de la programmation ;  
Les lignes grisées sont celles qui n'ont pas été faites ;  
Les lignes en rouge ont été annulées ;

Les ajouts dûs au « logiciel V2 » sont précisés.

Les priorités sont définies en bleu et vert. Vert étant « Obligatoire » et bleu « facultatif »

N°	fonctionnalité	Description	priorité
1	Switch	Permettre de naviguer facilement entre les pages tkinter.	bleu
2	Accueil de sélection	Page de sélection pour ajouter différents programmes ou exercices.	vert
3	Recherche de cours	Recherche dans les fichiers de l'ordinateur des cours que l'on veut sélectionner	vert
4	Conversion de fichier xlsx	Convertir les fichier xlsx en csv	bleu
5	Création des dossiers manquants Génération automatique de fichier	Vérifier si les dossiers de stockage sont présents ou pas, sinon les créer	bleu
6	Lecture de fichier	Pouvoir lire les fichiers donnés selon des colonnes voulus (colonnes du fichier tableur	vert
7	System de paramétrage	Pouvoir paramétrer à l'avance les éléments que l'on veut connaître de notre fichier tableur afin de préciser la lecture	Bleu
8	Page de sélection de cours	Page qui permet de sélectionner les cours voulus	vert
9	Page d'ajout de cours	Page pour ajouter des cours qui ne sont pas présents dans les fichier de cours	vert
10	Liste de cours dans la sélection des cours	Il faut établir une liste des cours existants pour les afficher lors de la sélection	vert
11	Générateur xml	Générer un fichier xml correspondant au cours voulu pour ne pas à avoir à lire en top les fichier tableurs. Permet aussi de séparer la lecture des fichiers tableurs et l'affichage.	Vert

12	Fichier de suivi	Un fichier log permettant de connaître le déroulé du logiciel lors d'une utilisation normale	Bleu
13	Une page de « simulation »	Créer une page d'affichage de la simulation	Vert
14	Rangement automatique des fichiers	Faire en sorte que les fichiers mis dans les cours soient triés et rangés dans leurs dossiers respectifs	Bleu
15	Affichage des Erreurs commise par l'utilisateur	Objet d'erreur qui indique avec un commentaire l'erreur produite par les modifications des paramètres prévues dans le cours	Vert
16	Boutons de réglage	Objet boutons qui permet les interactions avec le simulateur	Vert
17	Utilisation du simulateur avec clavier	Permettre de régler les boutons affichés grâce au clavier et/ ou potentiomètre.	Vert
18	Modification paramètre (logiciel V2)	Modifier la lecture des paramètres pour ajouter des paramètres de lecture (fichier settings)	Vert
19	Modification de l'écriture de fichier xml	Modifier l'écriture de fichier xml pour avoir de nouveaux paramètres (annulation cause : on peut lire les paramètres grâce au dictionnaire d'indice dans xml_lecture)	
20	Ajout d'un dictionnaire d'indice	Pour simplifier la lecture du fichier xml, ajouter un dictionnaire de la colonne indice (définis dans settings.txt)	
21	Ajout Baromètre (logiciel V2)	Sorte de baromètre qui affiche le total des valeurs produites par l'interaction des boutons avec le cours	Vert
22	Graphique de production (logiciel V2)	Afficher une courbe représentative de la chauffe des pièces lors de la simulation	Bleu
23	Affiché des images/vidéo (logiciel V2)	Afficher les images/vidéo lorsque l'utilisateur se trompe ou demande de l'aide lors d'une simulation	Vert
24	Affichage de commentaires (logiciel V2)	(rajouté car il était contenu dans les Erreurs affichées mais pour la nouvelle version du programme (V2) il doit être affiché dans un nouvel élément)	vert
25	<b>Affiché une animation de la production de la pièce (logiciel V1)</b>	<b>Animation centrale qui s'anime lorsque on lance la production d'une pièce avec les paramètres saisis avec les boutons</b>	vert
26	<b>Page de changement de paramètre</b>	<b>Pouvoir changer les paramètres en étant dans l'application sans à avoir aller dans le fichier txt</b>	Bleu

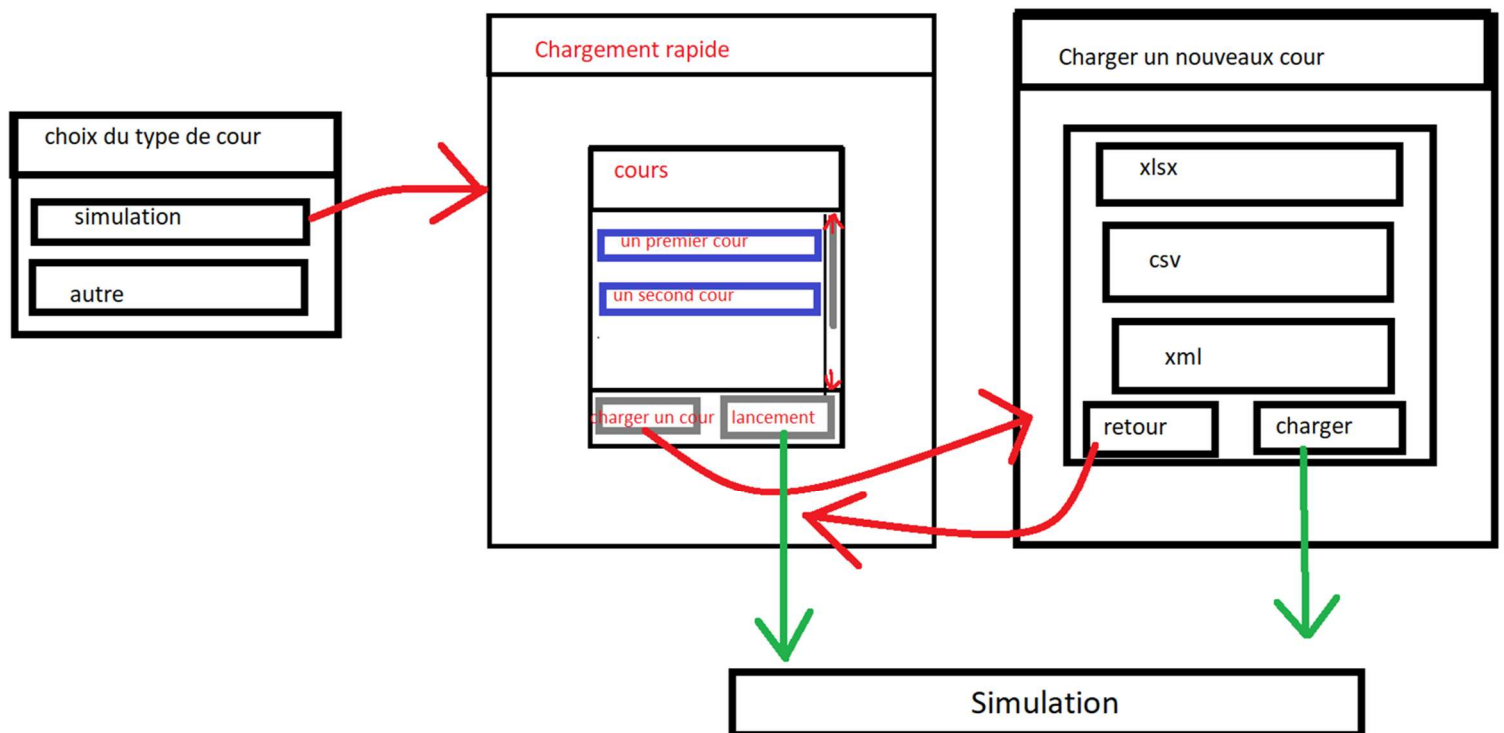
## 3.4 Modélisation de l'application

### 3.4.1 schéma interface (étape de recherche)

Lors de la création de ce logiciel, pour pouvoir représenter ce que le client (Mr Gauthier) veut, la schématisation de l'interface avant la phase de programmation est utile. Et permet aussi de savoir comment l'on va gérer les différents éléments du programme.

La plupart du temps, la schématisation est faite sur papier à la main. Pour des soucis de lisibilité, les voici refaites avec un outil numérique.

Menus principal et secondaire



Ici nous avons donc le fonctionnement et l'affichage de, « à quoi va correspondre les menus » pour arriver à la simulation.

En premier lieu, nous voyons le choix du type de cours que l'on va suivre.

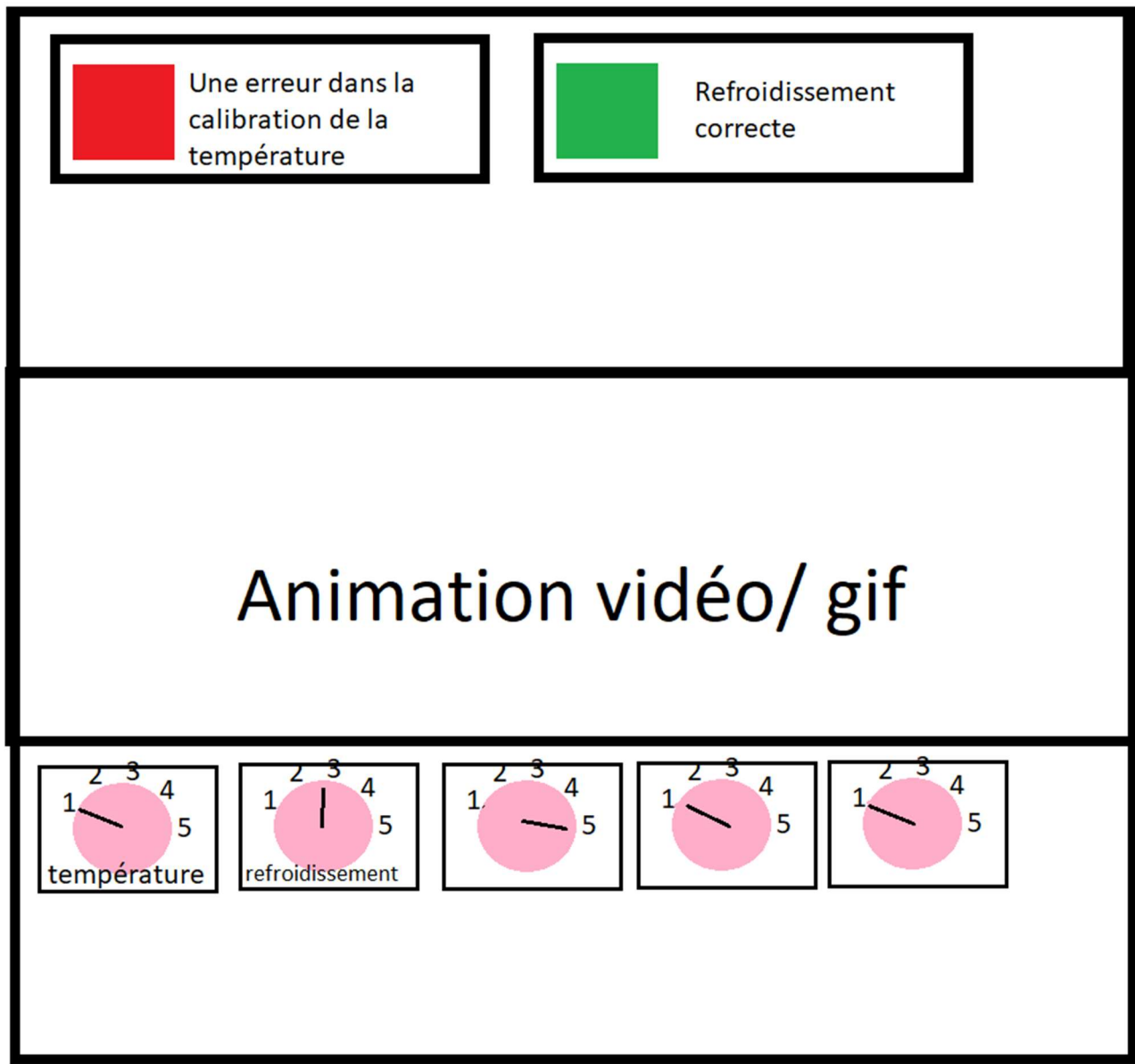
On voit :

- ⇒ le chargement des cours,
- ⇒ la position des boutons et la liste des cours, qui possède un assesseur vertical.
- ⇒ Et le chargement d'autre cours, non présents dans la liste.

On voit que « lancement et charger » envoient vers la simulation.



### 3.4.2 Interface principale (la simulation première version)



Voici la représentation de ce que le programme donne lors d'une simulation d'un cours.

La fenêtre serait séparée en 3 frames :

La première contenant des messages, (les objet erreur) pour indiquer comment est l'état de la machine et la progression de fabrication.

La seconde frame quant à elle contiendrait les images, vidéo ou gif contextualisant la situation. Cela permettrait d'avoir un logiciel interactif, pouvant imager des réactions possibles au modification des paramètres.

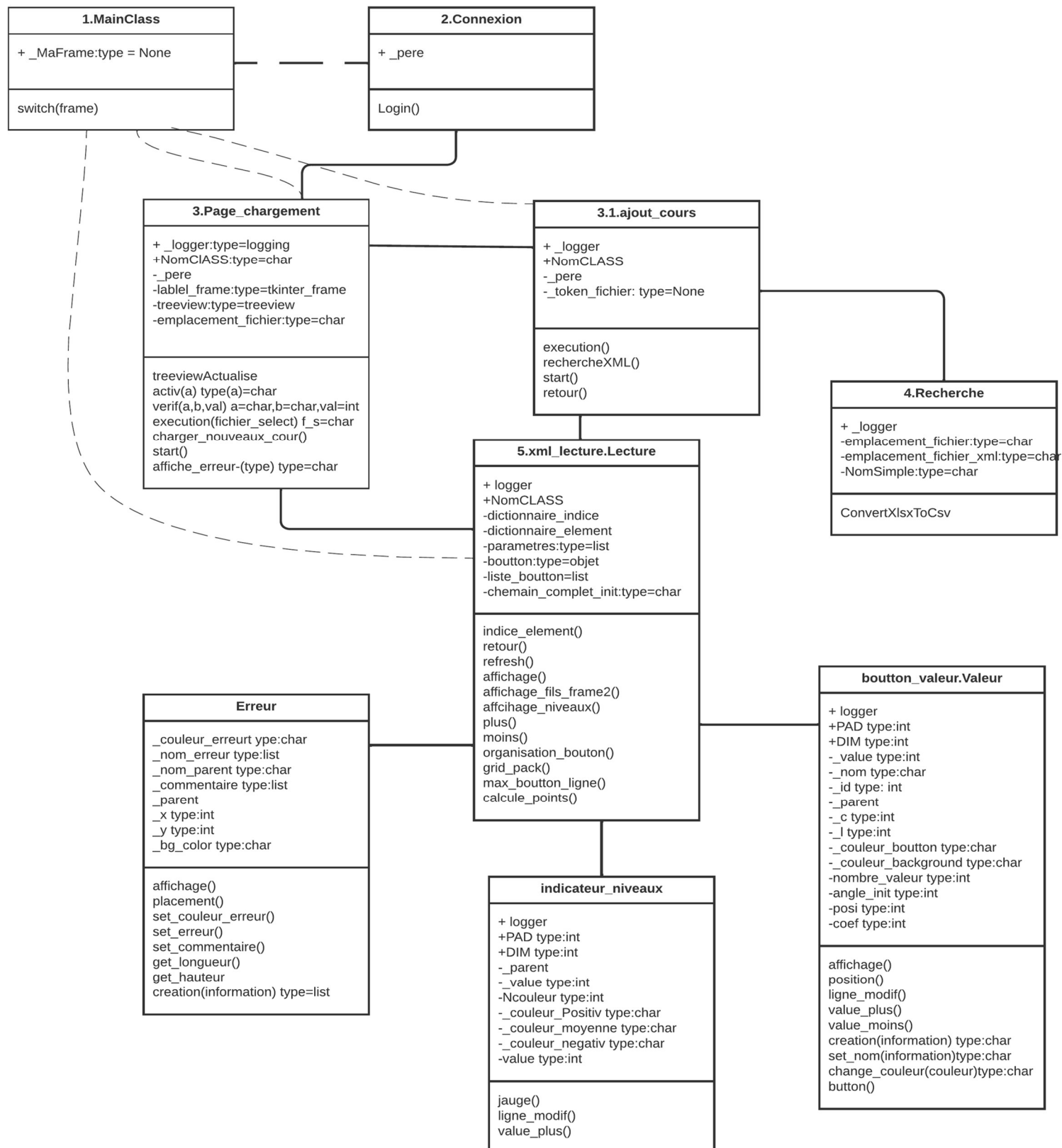
Aussi le but de la 3 eme frame est de contenir un ensemble de bouton pour interagir dans le simulateur. (le nombre de bouton et d'erreur peut être modulable)

Donc avec l'accord de Mr Gauthier, et les représentations graphiques, la production a pu commencer.

### 3.4.3 Diagramme de classe du logiciel

#### Diagramme de class

Léo Malubier



À noter que dans ce diagramme de classe, il n'y a pas les fichiers contenant uniquement des fonctions (souvent petits) et qui n'ont pas un impact important dans le diagramme.

À noter aussi que la classe MainClass sert de switch et donc pour passer d'un élément à un autre. (page Connexion à Page\_chargement ou ajout\_cours ou encore xml\_lecteur.Lecture) , on passe par MainClass pour pouvoir changer d'interface.

#### Apport personnel supplémentaire

Les apports supplémentaires que j'ai réalisés dans ce logiciel sont principalement les fonctionnalités bleues dans le cahier des charges. C'est-à-dire des fonctionnalités qui n'étaient pas demandées, et qui n'était pas forcément nécessaires au bon fonctionnement de l'application mais tout de même me semblaient utiles.

Je vais rapidement re définir ce que j'appelle les cours :

Les cours sont les tableurs, contenant les informations nécessaires à faire fonctionner le simulateur.

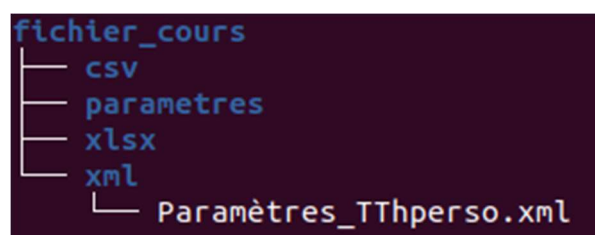
Pour commencer la génération automatique des dossiers des cours. Initialement, on aurait dû uniquement rechercher les cours à chaque fois que l'on utilisait le logiciel. Pour simplifier l'utilisation, on peut toujours rechercher les cours, mais un dossier appelé fichier\_cours est créé. **Ce qui permet de ranger les cours et de pouvoir créer des listes de cours facilement** en scannant le contenu de ce fichier.

Pour améliorer ce fichier, j'ai aussi ajouté le rangement automatique des cours à l'intérieur. C'est à dire que l'utilisateur a juste besoin de glisser ses fichiers à l'intérieur du dossier fichier\_cours et le logiciel les rangera dans un sous-dossier approprié.

Par exemple : je mets un fichier tableur xlsx dans fichier\_cours et il sera automatiquement rangé dans le sous dossier xlsx.

Toujours dans ce dossier, la possibilité de changer le format de fichier xlsx en csv, comme expliqué plus haut, pour des raisons d'utilisation du logiciel dans le temps.

Voici à quoi ressemble le dossier contenant les cours, appelé fichier\_cours pour montrer à l'utilisateur qu'il faut y mettre ses fichiers :



#### Aussi la génération de fichier xml, qui apporte un plus au projet :

Ayant séparé l'interface graphique de simulation du reste du logiciel, la capacité à pouvoir modifier cette interface permet une maintenance du logiciel. **Si l'interface ne convient plus, qu'il doit y avoir des retouches à faire ou carrément un changement, seule l'interface graphique de la simulation est à refaire.**

La génération du XML utilise des balises qui permettent de classer et d'organiser les informations des tableurs. Ce qui, sans cette étape, risque de compliquer la création d'interface en intégrant l'interface à l'algorithme de tri des informations.

La lecture des informations contenues dans les tableurs est relativement souple et permet à l'utilisateur de pouvoir paramétrer cette dernière. Grâce au fichier settings.txt, j'ai pu mettre en place une possibilité de paramétrer le lecteur du tableur. (ce paramétrage sera expliqué plus clairement dans la présentation du logiciel)

Le système de log est un apport utilisé pour la vérification du bon fonctionnement de l'application. Ce logiciel étant réalisé pour être repris plus tard par quelqu'un d'autre, le système de logging lui permettra de vérifier que le logiciel a fonctionné correctement jusqu'à présent, et si des erreurs non prévues arrivent, alors il pourra en avoir une trace.

J'ai récupéré le système de Switch utilisé lors de mon projet de fin d'année. Son fonctionnement est simple et permet de pouvoir changer la page sur laquelle on se trouve, sans avoir à supprimer chaque élément d'une fenêtre, un par un, au risque d'en avoir oublié et de provoquer des erreurs.

## 4 Phase de réalisation

En préambule, je tiens à préciser que :

La première version du logiciel est fonctionnelle en l'état. Toutes les fonctions, méthodes ou classes fonctionnent. Cette version est la plus aboutie, sauf pour ce qui concerne l'affichage « images, vidéo, gif ».

La seconde version est en fait principalement une refonte graphique. Cependant, des modifications (pour la génération xml en particulier) ont été apportées suite à la demande de Mr Gauthier, visant à changer la forme du tableur.

Dans la seconde version, l'application propose de pouvoir calculer et définir de coefficient et non de choisir un élément du tableur directement. (ce sera précisé dans l'explication de la Version2).

Ce logiciel a été codé sous un environnement Linux. Toutes les versions ont été testées sous Linux et Windows, les deux fonctionnent correctement.

Certains caractères sous Linux ont du mal à être reconnus sous Windows. Il y a des adaptations dans le code pour empêcher ce type d'erreur. Cependant il se peut que des erreurs non détectées lors de mes tests se présentent sous Windows.

### 4.1 Description pour utilisation :

#### 4.1.1 Version 1 :

En premier lieu, il est tout de même intéressant de savoir comment est formé le Tableur. Ainsi que les paramètres pouvant être modifiés par l'utilisateur PROFESSEUR.

**TABLEUR :** Pour ce faire je vais prendre l'exemple d'une ligne du tableur utilisée dans la simulation qui suivra.

Une ligne est composée de 13 colonnes. Ces colonnes peuvent être séparées les unes des autres, si l'utilisateur a bien rempli le fichier des paramètres.

La première colonne représente le groupe d'appartenance de la ligne. Ainsi, nous pouvons avoir plusieurs ligne Chauffage.

La seconde colonne représente la ligne en elle-même. On pourrait dire que la première colonne est un Nom de Famille et la seconde le prénom de la ligne. Ici comme exemple Puissance.

Puis les états sont représentés de la 3eme colonne à l'avant-dernière colonne. Entre autres exemples, « l'état de transformation d'une pièce dans un four » .

Autre exemple, pour une pièce que l'on chauffe trop peu : la colonne 1 donne l'info de Brûlures et de Tapures (tapures : petit strie dans le métal)

On peut donc avoir 2 titres dans cette seule case, ici, « Tapures » et « Brûlure ».

Aussi, on est en mesure de pouvoir mettre un commentaire dans une case pour l'afficher lors de la simulation : Il suffit de mettre un « # » pour que le commentaire s'affiche comme dans l'exemple de la colonne 9

On peut mettre autant de commentaires et de titres dans une case que l'on le souhaite. Tant que le Titre n'a pas de symbole spécial et qu'il est le seul sur sa ligne (s'il suit un commentaire, il faut qu'il y ait un retour à la ligne) : exemple colonne 6

Enfin la dernière colonne représente la valeur des colonnes supérieures à 5.  
Par exemple : mes valeurs de chauffe sont trop élevées quand elles sont supérieures à 5 et trop faibles quand elles sont inférieures à 4. Dans ce cas, mon indice de ligne est +  
A noter qu'il est important de bien définir un symbole avant la valeur positive ou négative, car le tableur Excel ne permet pas d'écrire un + ou un – dans une case.

Enfin, il ne sert à rien de remplir les colonnes 4 et 5 puisqu'elles sont considérées comme étant à l'état neutre (on vise un score de 4 ou 5 en additionnant chaque élément d'une même famille).

Exemple d'une ligne du tableur:

		0	1	2	3	4	5	6		7	8	9	indice
Chauffage	Puissance		Tapures Brûlure					Titre #com Titre Titre #com #com Titre				#un commentaire	'+

PARAMETRES : Pour ce qui est des paramètres de lecture, le fichier txt est fait en sorte que l'on comprenne ce que l'on remplit : Chaque paramètre possède un commentaire expliquant à quoi il sert. Il est important de bien le remplir puisqu'il définit les colonnes utiles à lire.

Aussi, chaque variable possède un nom et son paramètre entre {} pour bien définir le dernier comme par exemple:

colonnes{A:M} ou encore indice{N} (les lettres étant le nom des colonnes dans un tableur).

#### 4.1.1.1 LA NAVIGATION ENTRE LES MENUS

La prise en main est relativement facile et efficace.

Les différents menus ne sont pas surchargés d'information et permettent une prise en main rapide, tout en respectant la règle des 3 cliques pour ne pas perdre l'utilisateur.

Au lancement, nous avons une interface demandant le type d'exercice que l'on souhaite réaliser.

A noter :

Le choix de cette première page, est motivé du fait que l'on m'ait proposé 2 choix de sujets. Un QCM qui afficherait des questions, ainsi que des images et un simulateur.

J'ai choisi de faire le simulateur de machines, tout en gardant à l'esprit qu'un autre type d'exercice pouvait être rajouté. Donc une première page semble utile pour une utilisation future.

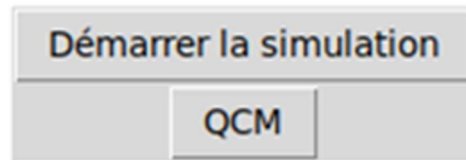


Image: Sélection du type d'exercice.

Une fois le type d'exercice sélectionné, on a accès à la sélection du cours. Une sélection peut être faite dans une liste d'éléments qui apparaît alors à l'écran, et permet de lancer le cours sélectionné, sous forme d'exercice ou simulation.

En cliquant sur « Démarrer la simulation », l'utilisateur lance l'exercice.

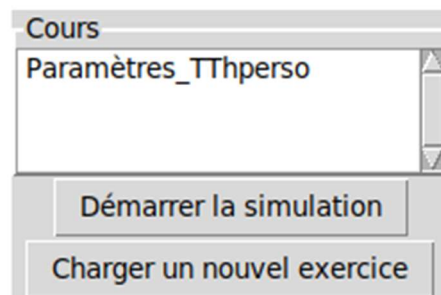


Image: menu de sélection des cours.

Cependant, il est possible que l'utilisateur n'ait pas rangé ses fichiers dans l'emplacement qui est prévu à cet effet. Alors il est possible de pouvoir charger un exercice qui n'est pas affiché dans la liste ci-dessus. Ceci, en cliquant sur « Charger un nouvel exercice », un nouveau menu s'affiche alors.

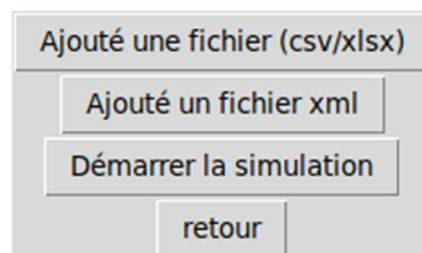


Image : menu de sélection de fichier de cours.



Une fois sur ce menu, plusieurs choix sont à notre disposition :

- Nous pouvons sélectionner un fichier tableur au format csv et xlsx.
- Nous pouvons également sélectionner un fichier xml si le programme en a déjà généré un et que l'utilisateur a modifié l'emplacement des fichiers ;
- Puis nous pouvons Démarrer la simulation ou revenir dans le menu précédent.

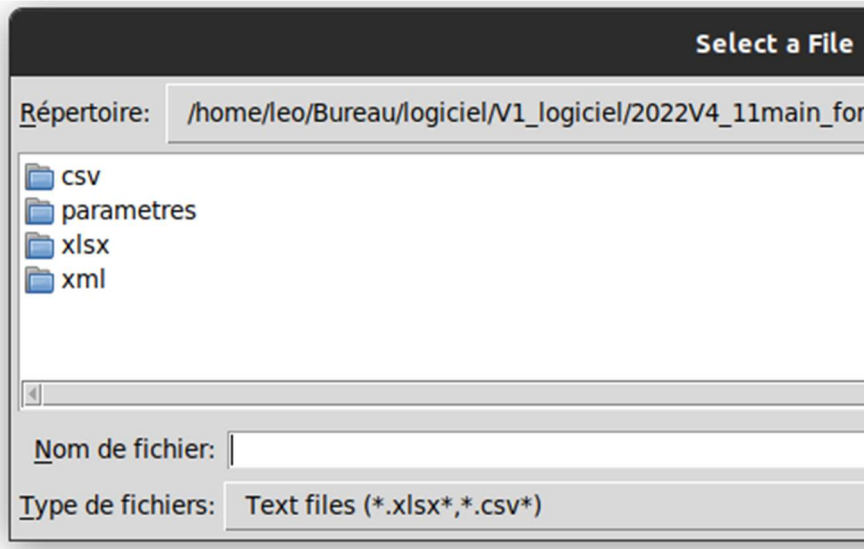


Image : Sélection de fichier

La sélection de fichier se fait comme dans n'importe quelle application existant.

À noter 1 :

-si utilisateur clic sur « Démarrer la simulation » sans avoir sélectionné de fichier au préalable, une erreur s'affiche à l'écran et demande de sélectionner un cours/ exercice.

À noter2 :

-Il est probable que le logiciel ne trouve aucun cours dans ces dossiers (si l'utilisateur n'en a pas mis, ou lors du premier lancement de l'application puisque les dossiers ne sont pas créés).

Alors une erreur s'affiche à l'écran, avec message indiquant qu'il faut trouver des exercices grâce au second menu en cliquant sur « charger un nouvel exercice ».

Il est aussi possible de redémarrer le logiciel en ayant ajouté les tableurs dans les fichiers alors apparus.

À noter 3 :

Lors de la sélection de fichier xlsx, un choix est donné à l'utilisateur. Il peut convertir son fichier xlsx en csv pour pouvoir garder ce format.

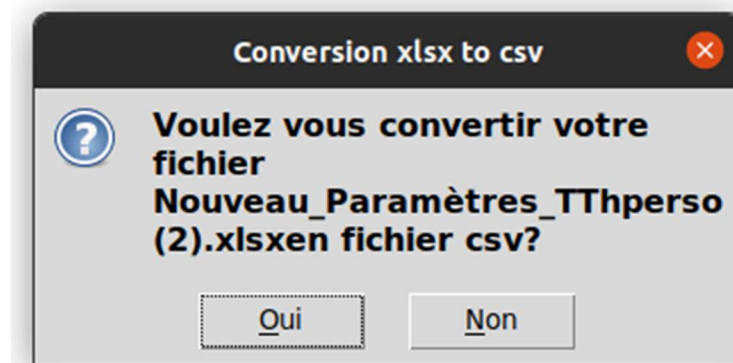


Image : proposition  
conversion xlsx à csv

Si l'utilisateur clique sur « oui » alors un fichier au format csv apparaîtra dans le sous-dossier csv du dossier fichier\_cours.

Ainsi en cliquant sur « Démarrer la simulation » depuis n'importe quel menu, l'utilisateur démarre la simulation.

### 4.1.1.2 LA SIMULATION

Ici, la mise en page n'est pas entièrement finie. On peut encore voir le nom des frames, puis les boutons qui permettent les interactions dans la frame2.

Le bouton 'actualise', permet de re-définir largement des boutons de la frame3 lorsque la taille de la fenêtre change.

Le bouton 'retour' permet de revenir au menu.

Le bouton 'print' permet de recharger les erreurs affichées dans la frame1.

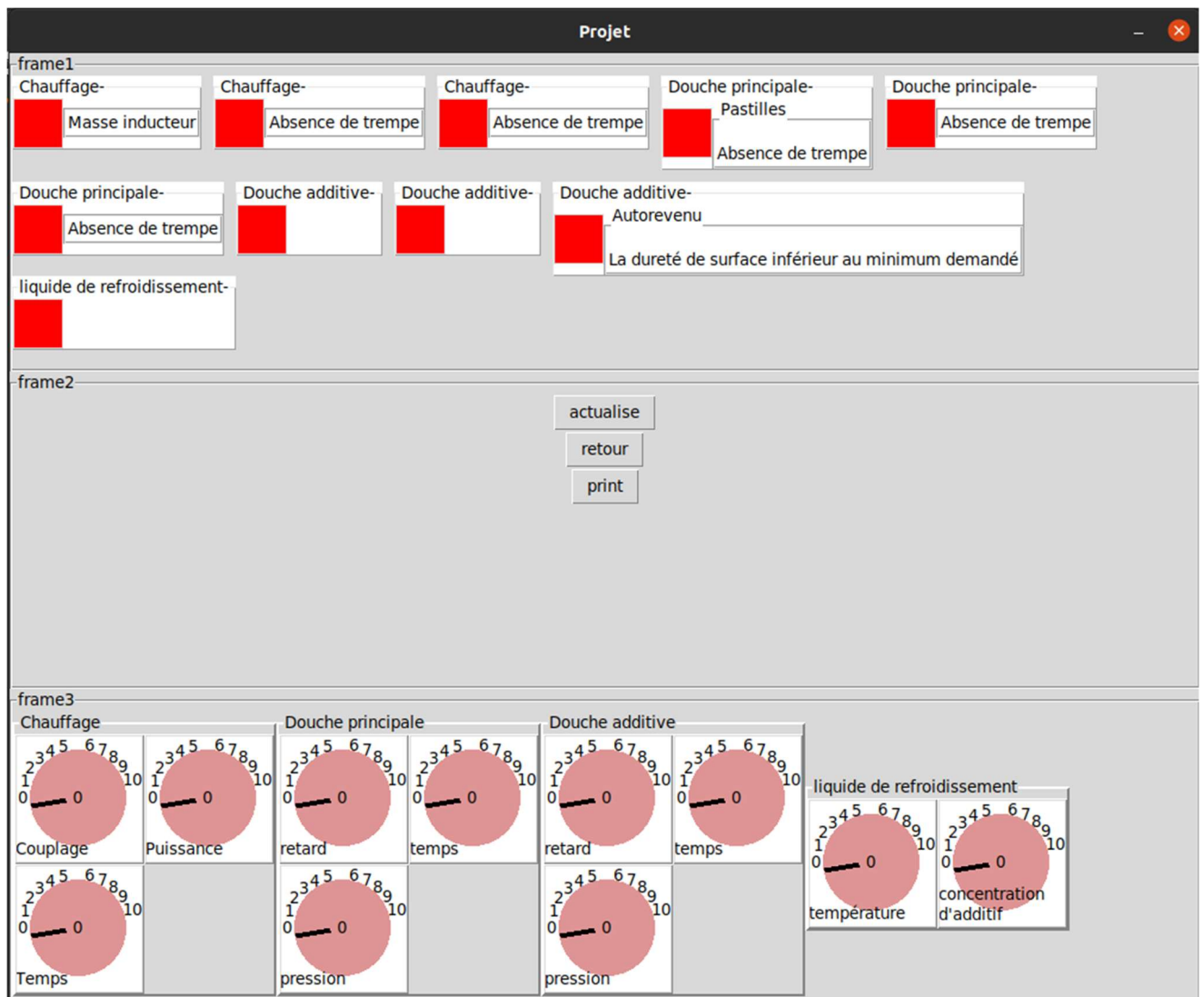


Image : fenêtre de simulation.

Ainsi chaque bouton de la frame3 peut augmenter sa valeur de +1 par un clic droit et de -1 par un clic gauche. Chaque le nom de chaque famille est indiqué, entourant tous les boutons appartenant à cette famille.

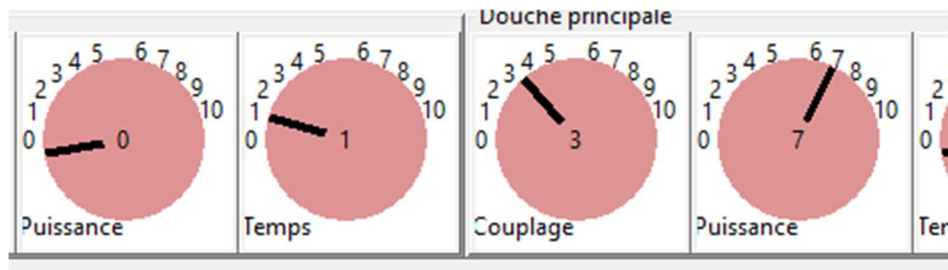


Image : exemple de variation des boutons

En modifiant les valeurs de ces boutons, et en actualisant les résultats, nous pouvons donc avoir des variations de ces résultats.

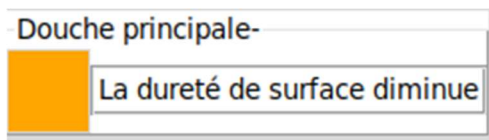


Image : résultat moyen

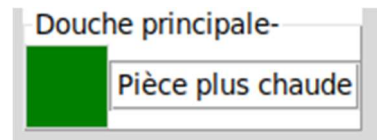


Image: résultat positif.

Ainsi la simulation permet de définir des boutons apportant des résultats différents à chaque variation et en avoir un résultat.

#### 4.1.2 Version 2

Les menus et leur fonctionnement n'ont pas été modifiés. Il n'est pas nécessaire de les redéfinir. Cependant le fonctionnement de notre tableur ainsi que l'affichage ont, quant à eux changé.

La seconde version entraînée par la modification des informations contenues dans les tableurs révèle plusieurs changements au sein du programme. Et étant donnée le temps imparti relativement court du stage, les changements apportés ne sont pas aboutis. De ce fait, la seconde version de ce logiciel ne fonctionne pas.

Cependant, la plupart des modifications importantes ont été réalisées correctement : la lecture et l'écriture des fichiers xml ont été modifiées, ainsi que les paramètres de lecture des tableurs. Et un nouvel objet est apparu, entraînant une modification de l'affichage.

Pour commencer, voyons la nouvelle architecture de notre tableur.

On retrouve nos 13 colonnes avec notre Nom et Prénom.

Le changement important qui a été apporté se trouve dans le contenu des case 3 à 12.

Ainsi que le contenu de la colonne « d'indice ».

Exemple :

0	1	2	3	4	5	6	7	8	9	indice
P2	P-1	P2H4	P3H2 #com		P=2;H=-1					P→ puissance H→ hateur

Comme montré dans cet exemple, la composition des cases est différente.

Elle contient des valeurs qui influent sur une variable définie dans la colonne 5.

Les indices définissent la signification de ces variables. Ces variables peuvent être négatives ou positives.  
 Quand une variable équivaut à P2 alors la variable « puissance » gagne +2 en valeur.  
 Inversement, une variable P-1 perd un point.  
 Le but étant d'atteindre les valeurs définies dans la colonne 5 pour être optimale.

Ainsi voici à quoi ressemble la nouvelle version :

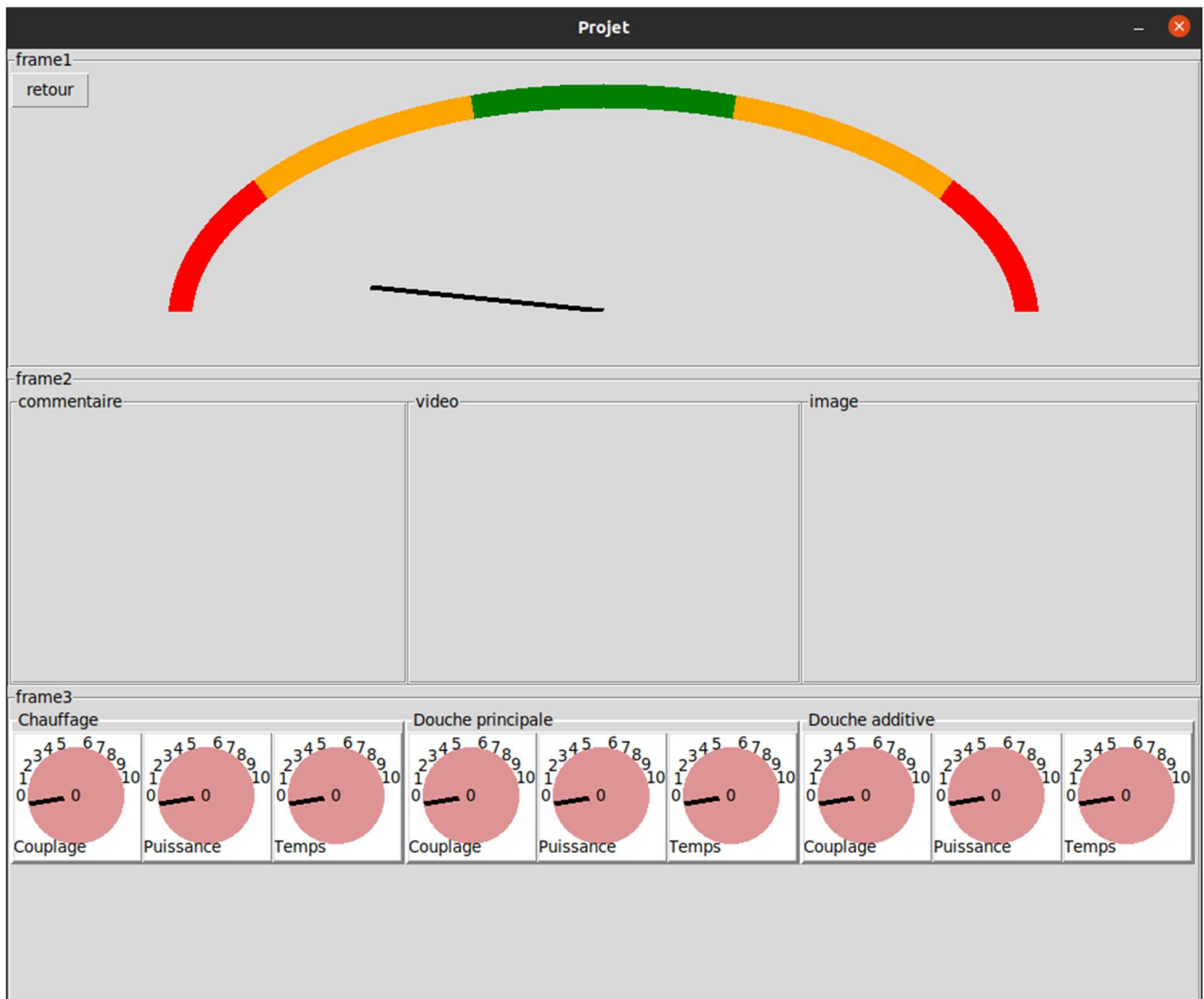


image : fenêtre de simulation version 2

3frames sont apparues pouvant contenir des commentaires, des images et des vidéo.  
 Ainsi qu'un baromètre fonctionnel qui indique si l'on se rapproche de la valeur voulue ou non.  
 Les boutons d'actualisation et 'print' ont disparu pour devenir automatiques.  
 Le bouton « retour » est quant à lui toujours présent en haut à gauche.

## 4.2 PROGRAMMATION DETAILLEE:

xml\_lecture.py:

Ce fichier permet la gestion de la partie graphique de la 'simulation' .  
On affiche plusieurs objet(Différent selon la version) comme bouton\_valeur.py,

Erreur.py , indicateur\_niveaux.py.

Elle est composé de 6 frame différentes.

3 principales

frame1 : l'affichage graphique des variation engendré par l'utilisation des bouton et la modification des valeur entré.

frame2 : l'affichage de 3 frames différentes : commentaire, vidéo et images. (cette partie n'est par réalisé pour le logiciel V1 mais l'est pour V2.les frame sont présentes dans V2, mais ne sont pas utilisé.)

frame3 : l'affichage des bouton généré. (sous forme de listes).

Les listes de boutons utilisés des objets Valeur de bouton\_valeur.py

Ce fichier permet ainsi le calcul des points et la lecture du fichier XML et de l'afficher à l'utilisateur

(pour la v2) La méthode indice\_element permet de créer un dictionnaire d'indice contenant la signification des éléments contenus dans valeur trouvée dans le xml.

La méthode refresh permet d'actualiser la position des boutons (de la liste des boutons) et d'afficher le maximum d'éléments sur une ligne. (appelle la méthode organisation\_bouton

La méthode affichage est l'une des plus importante. Elle permet d'afficher les 3 frames de contenus. (F1,f2,f3) et initie la création d'une liste de boutons. (les boutons qui sont des objet de bouton\_valeur.py)

La méthode affichage\_fils\_frame2 affiche les frames image, comentaire et vidéo.

La méthode affiche\_niveaux affiche le baromètre, qui est un objet de indicateur\_niveaux.py

Les méthodes « plus » et « moins » permettent de capter les évènements « clic » des souris pour monter ou baisser la valeur des boutons.

La méthodes organisation\_bouton permet de calculer la position de chaque bouton de la liste. Chaque bouton étant placé par grid, on détermine comment il est placé.

grid\_pack permet d'afficher les frames principales en les forçant à ce partager l'écran en 3 parties et à prendre tout l'espace disponible.

max\_boutton\_ligne est utilisé par organisation\_boutons pour savoir combien de boutons l'on peut positionner sur une seul ligne.

(v1) `calcule_points` permet de définir les erreurs que l'on affiche. En leur attribuant une valeur, ainsi que le nom et les commentaires qu'ils doivent avoir.

(v1) `organisation_erreur` fonctionne de la même manière que `organisation_bouton` mais affiche les erreurs

PS : à partir d'ici, et afin d'alléger la lecture du rapport, je détaille un peu moins les fonctions, tout en maintenant la description de leur fonctionnement général.

`indicateur_niveaux.py`:

Objet utilisé par `xml_lecture.py`

Ce fichier permet d'afficher un « compteur » ou « une jauge ». Cet affichage est fait grâce à la fonction `create_arc` de `tkinter`. Pour avoir une variance de la couleur on produit des « segments » de la courbe que l'on place les uns à la suite des autres.

L'aiguille qui indique la couleur est la même que dans `'bouton_valeur.py'`. On utilise le cercle trigonométrique (dans son fonctionnement classique) ce qui permet de calculer efficacement la position de l'aiguille.

Les fonctions « `value_plus` » et « `value_moins` » sont les fonctions qui permettent de faire varier l'aiguille en appelant la fonction `ligne_modif`.

Méthode `jauges` : permet de calculer les différentes positions des couleurs

« `ligne_modif` » : modifie la ligne d'indication de niveau de couleur

« `plus` » et « `moins` » appellent `ligne_modif` pour faire baisser ou augmenter sa valeur

`Erreur.py`:

Objet utilisé par `xml_lecture.py`

Ce fichier était utilisé dans la première version du programme.

Ce fichier permettait l'affichage des erreurs sous forme de `label_frame` composé d'un indicateur de couleur (`canvas`).

Les `label_frame` possédaient le nom des différents éléments auxquels ils se rattachent (d'où vient l'erreur, avec le nom du père et du fils.)

`bouton_valeur.py`:

Objet utilisé par `xml_lecture.py`

Ce fichier sert de potentiomètre graphique. Cet objet est créé grâce à la librairie graphique `tkinter`. Composé d'un disque dont on peut changer la couleur grâce à la variable `self.couleur_bouton` et changer la couleur de la frame grâce à la méthode « `change_couleur` ».

Dans la méthode `button`, l'affichage des valeurs (0 à 10) est faite en suivant la courbe supérieure du disque (180° à 0°)

avec une correction pour que les écritures 0 à 6 / 7 soit en dehors du disque. (sur la partie 180° à 90°).

L'affichage des lignes indicatrices de la valeur sélectionnée peut être modifiée par `ligne_modif` appelée par `value_plus` ou `value_moins` (similaire à `indicateur_niveaux.py`)

« méthode `position` » permet de positionner l'objet dans la frame parente.

« `ligne_modif` » : modifie l'indicateur de niveaux

« `value_plus` » et « `moins` » font changer la valeur positivement ou négativement la ligne d'indication de niveaux

« méthode `bouton` » : calcule la position de l'affichage des valeurs 0 à 9 autour du disque.



xml\_generateur.py:

Ce fichier s'occupe de la création de fichier xml. Elle utilise les paramètres définis dans le fichier parametres.txt (encore dans le /logiciel/parametres)

Ce générateur est fait à la main avec la méthode write à l'aide de liste et de boucle.

### Exemple XML V1 :

```
<?xml version="1.0" encoding="UTF-8"?>
- <Famille>
  - <Pere name="Chauffage">
    - <Fils name="Couplage">
      <Petit_Fils couleur="red" value="eli" commentaire_0="Masse inducteur" titre_0="None" id="0"/>
      <Petit_Fils couleur="red" value="eli" titre_0="Brûlures (peau d'orange)" id="1" titre_1="Tapures"/>
      <Petit_Fils couleur="orange" value="3" commentaire_0="Grossissement de grain" titre_0="None" id="2"/>
      <Petit_Fils couleur="green" value="2" commentaire_0="La profondeur de trempe augmente" titre_0="None" id="3"/>
      <Petit_Fils couleur="green" value="1" titre_0="test" id="4"/>
      <Petit_Fils couleur="green" value="0" titre_0="test" id="5"/>
      <Petit_Fils couleur="green" value="-1" commentaire_0="La profondeur de trempe diminue" titre_0="None" id="6"/>
      <Petit_Fils couleur="green" value="-2" commentaire_0="La profondeur de trempe diminue" titre_0="None" id="7"/>
      <Petit_Fils couleur="orange" value="-3" commentaire_0="La dureté de surface diminue" titre_0="None" id="8"/>
      <Petit_Fils couleur="red" value="eli" commentaire_0="La dureté de surface inférieur au minimum demandé" titre_0="None" id="9"/>
      <Petit_Fils couleur="red" value="eli" commentaire_0="Absence de trempe" titre_0="None" id="10"/>
    </Fils>
    + <Fils name="Puissance">
    + <Fils name="Temps">
  </Pere>
  + <Pere name="Douche principale">
  + <Pere name="Douche additive">
  + <Pere name="liquide de refroidissement">
</Famille>
```

### Exemple XML V2 :

```
<?xml version="1.0" encoding="UTF-8"?>
<Famille>
  - <Pere name="Chauffage">
    - <Fils name="Couplage">
      <Petit_Fils couleur="red" value="eli" commentaire_0="Commentaires" valeur_0="C3" id="0"/>
      <Petit_Fils couleur="red" value="eli" commentaire_0="Commentaires" valeur_0="P3" id="1"/>
      <Petit_Fils couleur="orange" value="-3" commentaire_0="Commentaires" valeur_0="P2" id="2"/>
      <Petit_Fils couleur="orange" value="-2" commentaire_0="Commentaires" valeur_0="P1" id="3"/>
      <Petit_Fils couleur="green" value="-1" commentaire_0="Commentaires" valeur_0="P, H" id="4"/>
      <Petit_Fils couleur="green" value="0" valeur_0="P=4;H=4" id="5"/>
      <Petit_Fils couleur="green" value="1" commentaire_0="Commentaires" valeur_0="P1,H2" id="6"/>
      <Petit_Fils couleur="orange" value="2" commentaire_0="Commentaires" valeur_0="P2,H2" id="7"/>
      <Petit_Fils couleur="orange" value="3" commentaire_0="Commentaires" valeur_0="P3,H3" id="8"/>
      <Petit_Fils couleur="red" value="eli" commentaire_0="Commentaires" valeur_0="P3,H4" id="9"/>
      <Petit_Fils couleur="red" value="eli" commentaire_0="Commentaires" valeur_0="P3,H5" id="10"/>
    </Fils>
    + <Fils name="Puissance">
    + <Fils name="Temps">
  </Pere>
  + <Pere name="Douche principale">
  + <Pere name="Douche additive">
</Famille>
```

recherche.py:

Ce fichier permet l'affichage d'une nouvelle fenêtre de recherche de fichier. Elle permet à l'utilisateur de trouver ses fichiers xml, xlsx, csv. Elle propose la conversion xlsx vers csv.

recherche\_element.py:

Ce fichier permet de ranger les fichiers (csv,xlsx,xml, v2(png,jpg,mp4)) mis dans fichier\_cours. Et renvoie une liste des éléments csv xlsx xml.

Page\_chargement.py:

Ce fichier permet l'affichage des listes de cours trouvés dans fichier\_cours. Si un fichier xml est trouvé et porte le même nom qu'un fichier csv ou xlsx, alors on le fichier affiché dans la liste des cours est le fichier xml. Cela permet de ne pas refaire un fichier xml à chaque lancement de cours.

En cliquant sur « charger un cours » on accède donc à la « ajout\_cours.py » qui permet de rajouter un cours dans la liste des cours existant s'il n'est pas trouvé automatiquement précédemment.

Des erreurs prévues s'affichent à l'utilisateur lorsque le programme n'a pas trouvé de fichier ou qu'il n'y a pas de fichier de cours sélectionné

« méthodes verif »: vérifie les listes entre elles pour savoir si un fichier xml a le même nom que des fichier csv ou xlsx. (dans le but de ne gardé que les fichier xml)

« treeviewActualise » : affiche en priorité les fichiers xml puis csv et xml.

« Méthode affiche\_erreur » : affiche les informations d'erreurs à l'utilisateur.

Lecture\_Settings.py:

Objet servant à récupérer les informations des paramètres (paramètre.txt)

Lecture\_Donnee.py:

Fichier permettant de lire les données d'un CSV ou XLSX selon les colonnes données.

FileExist.py:

Fichier permettant de vérifier l'existence des dossiers nécessaires à l'exécution du programme. Permet aussi de créer les dossiers s'ils n'existent pas.

converXlsx.py:

convertit les fichier xlsx en csv (car le fichier csv évolue moins c'est open source)

Ajout\_cours.py:

Ce fichier est appelé par Page\_chargement.py et peut rappeler « Page\_chargement.py »

Permet d'ouvrir un fichier xlsx / csv qui n'est pas dans les fichiers de cours.

Permet de convertir un fichier xlsx en csv.

Permet de choisir l'utilisation d'un fichier xlsx, csv ou xml

Accueil.py:

Permet d'orienter l'utilisateur dans les exercices qu'il veut faire.

(QCM pour un future exercice sous forme de QCM)

main.py:

Le Switch :Le fonctionnement du Switch est simple. On initialise une frame vide à la taille de la fenêtre. On appelle une class contenant une frame ou des objets que l'on doit affichés.

On récupère les paramètres de la frame parent pour indiquer à l'élément fils la frame de destination.

Puis quand on veut changer de page, on appelle à nouveau « le Switch » avec la nouvelle page à afficher, ce qui détruit la frame pere pour en re crée une.

Ce qui permet de simplifier l'utilisation des objet tkinter tout en sachant que lorsque l'on veut changer de fenêtre, nous ne sommes pas obligés de supprimer un à un tous les éléments de la fenêtre.

Au démarrage de l'application, on initie une fenêtre, on active le switch possédant une fram qui se réinitialise à chaque nouvelle page appelée.

Permet une interaction tkinter plus simple dans une seule fenêtre.

verification\_file.py:

Vérifie que le fichier des log existe bien et qu'il ne prend pas trop de place.

Arrivé à 1Mo, le fichier est réinitialisé.

Treeview des de l'ensemble des fichier python :

```
Logiciel/
├── documentation
│   └── information_debug.txt
├── GenXml
│   ├── Accueil.py
│   ├── ajout_cours0.py
│   ├── ajout_cours.py
│   ├── converXlsx_to_csv.py
│   ├── FileExist.py
│   ├── Lecture_Donnee.py
│   ├── Lecture_Settings.py
│   ├── Page_chargement.py
│   ├── recherche_element.py
│   ├── recherche.py
│   └── xml_generateur.py
├── __init__.py
├── log
│   └── log.log
├── main
│   └── main.py
├── parametres
│   └── setting.txt
├── Simulation
│   ├── bouton_valeur.py
│   ├── Erreur.py
│   └── xml_lecture.py
├── temp
│   └── temporaire.txt
└── verification_file_log.py
```

### 4.3 Amélioration possibles

Les améliorations restent possibles : elles consistent principalement à faire aboutir les parties non réalisées du cahier des charges. D'autre part, il me paraît dommage de ne pas avoir eu le temps de conclure sur les parties bleues (qui sont les parties que j'ai rajoutée en supplément du projet principal).

Le détail de ce qui a été fait, ce qui reste à faire, et des ajouts personnels au projet figure au « cahier des charges » détaillé dans ce rapport.

## 5 CONCLUSION

Ce stage m'a paru extrêmement enrichissant. Avoir pour mission de répondre aux attentes d'un client m'a donné l'envie et le dynamisme de créer et de coder. De plus, pouvoir mettre en pratique le savoir acquis au long de ces années d'études est gratifiant et me permet de faire un point sur mes connaissances.

Grâce à ce stage, j'ai pu appréhender le fonctionnement d'entreprise, et le travail que devait fournir un informaticien. J'ai pu côtoyer différents corps de métier, tous différents du mien, ce qui constitue l'avantage d'une mission au sein d'un grand Groupe comme STELLANTIS.

Le projet m'a également plus dans son ensemble, notamment la partie « programmation » pure, et moins celle de l'interface. Ce projet m'a permis d'apprendre beaucoup en peu de temps. La création de ce logiciel durant ces 2 mois m'a permis de réaliser une synthèse de mon année d'Université.

Au bilan, j'ai pu créer une bonne base d'application de simulation de machine, tout en prévoyant les futurs ajouts d'autres types d'exercices, simplifié par le switch. En y apportant des outils qui me semblaient pratiques, comme le rangement automatique de fichiers ou bien encore des log.

L'application V1 est fonctionnelle, sans pour autant disposer de tous les éléments d'affichage. Et l'application V2 possède tous les éléments nécessaires à son bon fonctionnement, quoique le temps imparti ait été trop court pour finaliser l'algorithme.

Je termine donc ce stage avec un léger goût d'inachevé. Même s'il était prévu avec mon maître de stage que l'application nécessiterait beaucoup de temps dans sa finalisation.

L'ensemble du programme et des versions ne contient pas d'élément sensible appartenant au groupe STELLANTIS. Suite à l'accord de mon maître de stage, Mr Alain Gauthier technicien du laboratoire de métallurgie, l'intégralité de mon code est disponible librement à l'adresse suivante.

<https://github.com/Leo-Malaubier>

## 6 ANNEXES

### 6.1 Sources

site Stellantis :

<https://www.stellantis.com/fr/groupe/a-propos-de-nous>

site Stellantis de Caen :

<https://site.groupe-psa.com/caen/fr/a-propos/microduromètre>

<https://www.analyses-surface.com/laboratoire-mesure-durete-materiaux.html#:~:text=Le%20microdurom%C3%A8tre%20permet%20d'effectuer,de%20la%20microstructure%20des%20grains.>

Traitement thermique par induction :

<https://www.bodycote.com/fr/services/traitement-thermique/trempe-et-revenu/trempe-par-induction/#:~:text=La%20trempe%20par%20induction%20est%20un%20processus%20de%20traitement%20thermique,les%20caract%C3%A9ristiques%20de%20r%C3%A9sistance%20m%C3%A9canique.>

Dual-boot :

<https://fr.wikipedia.org/wiki/Multiboot>

back4App :

<https://blog.back4app.com/fr/les-10-principaux-langages-de-programmation/>

C# :

[https://fr.wikipedia.org/wiki/C\\_sharp](https://fr.wikipedia.org/wiki/C_sharp)

Java :

[https://fr.wikipedia.org/wiki/Java\\_\(langage\)](https://fr.wikipedia.org/wiki/Java_(langage))

Python :

<https://docs.python.org/fr/3/tutorial/>

C++ :

<https://fr.wikipedia.org/wiki/C%2B%2B>

C :

[https://fr.wikipedia.org/wiki/C\\_\(langage\)](https://fr.wikipedia.org/wiki/C_(langage))

arbre xml :

Introduction à XML. (developpez.com)

Table des figures

### 6.2 Lexique

-Transmission : La Transmission est l'organe qui permet de transmettre l'énergie produite par le moteur aux 2 roues avant de la voiture. Elle relie la boîte de vitesses à chacune des roues en permettant les débattements de suspension et le braquage dans les virages.

-Pivot :Le pivot de suspension préparé (PSP) est un organe de sécurité assurant le freinage, la liaison avec la direction, la suspension et l'Anti Blocage Roue.

-Berceau : Le berceau est un composant du train avant du véhicule, il assure l'interface entre la caisse et les liaisons au sol.

-triangle :Le triangle est un bras de suspension, il fait partie des liaisons au sol du train avant.

-Traitement thermique :Les pièces usinées subissent un traitement thermique (traitement par induction ou four à atmosphère contrôlée). Ces opérations consistent à traiter en

surface ou en profondeur la matière des différents composants pour les rendre plus résistants aux efforts mécaniques et à l'usure.

-microduromètre :Le microduromètre permet d'effectuer des tests de dureté Vickers rapidement, précisément et de façon fiable, sur des métaux, des matériaux frittés, des produits céramiques, des circuits intégrés, des revêtements, et permet l'observation de la microstructure des grains.

-traitement thermique par induction :a trempe par induction est un processus de traitement thermique permettant d'améliorer les propriétés mécaniques dans une zone spécifique d'un composant ferreux. La zone durcie résultante améliore les résistances à l'usure et à la fatigue, ainsi que les caractéristiques de résistance mécanique.

Dual-boot :Le terme *dual-boot* (en français, double amorçage ou amorçage double) désigne la possibilité de démarrer deux systèmes d'exploitation sur le même ordinateur.

Xlsx ou fichier xlsx : Format de tableur utilisé par Excel de Microsoft

CSV ou fichier csv : Format open source de tableur.

XML : L'**Extensible Markup Language**, généralement appelé XML, « langage de balisage extensible » en français, est un métalangage informatique de balisage générique.

Orienté objet :La programmation orientée objet (POO) est un paradigme informatique consistant à définir et à faire interagir des objets grâce à langages. On appelle objet, un ensemble de variables complexes et de fonctions, comme par exemple un bouton ou une fenêtre sur l'ordinateur, des personnes (avec les noms, adresse...)

-arbre xml : Un document XML est hiérarchisé sous forme d'arbre. Cette représentation est logique et permet de faire des recherches très pointues à l'intérieur d'un document XML