

### Directives :

- Le travail est individuel. Il vaut pour 35% de la note finale de la session.
- Le travail doit être remis par GitHub avant la date de remise indiquée sur Lea
- Vous devez vous créer un dépôt GitHub pour ce projet avec ce lien <https://classroom.github.com/a/cWcuycsF>
- Renommez votre dépôt de la façon suivante :  
a22-tp2-*VotreDA* par exemple a22-tp2-1234567
- **ATTENTION : Le fichier taches.xml doit être utilisé pour réaliser ce travail**

### Description du travail :

Vous devez créer un logiciel qui permet de gérer une liste de tâches dans un style de tableau Kanban. Ce modèle permet de suivre le déroulement de certaines tâches. Dans ce travail, les tâches seront composées de différentes étapes de réalisation. Une tâche pourra être soit planifiée, en cours ou encore terminée.

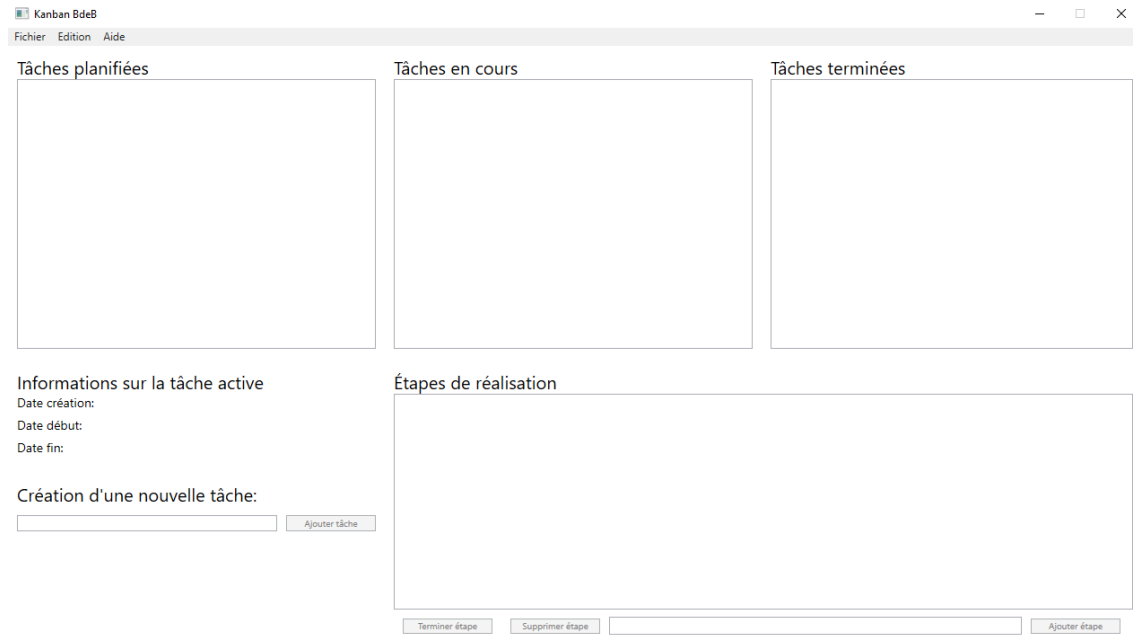
À partir de votre logiciel, il sera possible de créer et supprimer des tâches, de créer et de supprimer des étapes de réalisation, de charger et sauvegarder des fichiers de tâches dans un format .xml imposé. Un fichier initial contenant des données est fourni avec l'énoncé de travail.

Le format de fichier suivant doit être respecté :

```
<taches>
  <tache creation="2022-10-31" debut="" fin="">
    <description>Premiere tâche</description>
    <etapes>
      <etape termine="False" no="1">Première étape</etape>
    </etapes>
  </tache>
</taches>
```

Les différents éléments seront expliqués dans la section sur les tâches.

Au démarrage, vous aurez cette interface :

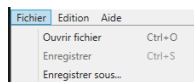


Règles à respecter sur la fenêtre :

- Le titre doit être Kanban BdeB
- La fenêtre doit être de 1280x720
- Il ne doit pas être possible de la redimensionner la fenêtre

Vous êtes libre de placer les différents éléments autrement, mais ceci ne doit pas rendre votre logiciel inutilisable. Au démarrage, il n'y a aucune donnée à afficher. Il faudra en créer ou encore ouvrir un fichier de tâches.

### *Présentation du menu Fichier*



### *Ouvrir fichier*

Le fichier Ouvrir fichier doit permettre d'ouvrir un fichier en format .xml. Une fois ce fichier ouvert, il faut charger les différentes tâches dans les trois listes de la fenêtre (Tâches planifiées, Tâches en cours et Tâches terminées).

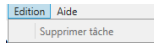
### *Enregistrer*

Les modifications ne seront pas sauvegardées automatiquement. Si on veut les conserver, il faudra sauvegarder le fichier préalablement ouvert. L'option sera réalisable seulement si un fichier a préalablement été ouvert ou qu'un Enregistrer sous... a été fait.

### *Enregistrer sous...*

Cette option permet de sauvegarder les données en spécifiant un autre nom de fichier.

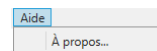
### *Présentation du menu Edition*



### *Supprimer tâche*

Permet de supprimer la tâche sélectionnée. Pour que l'option soit active, il faut préalablement qu'une tâche soit sélectionnée.

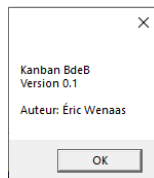
### *Présentation du menu Aide*



### *À propos...*

Permet d'afficher une boîte de message contenant le nom du logiciel, sa version et votre nom.

Par exemple :



### *Notes générales sur les menus :*

- Toutes les options doivent être réalisées avec des RoutedCommand.
- Le menu Fichier doit s'ouvrir avec le raccourci clavier Alt+F.
- Le menu Édition doit s'ouvrir avec le raccourci clavier Alt+D.
- Le menu Aide doit s'ouvrir avec le raccourci clavier Alt+A.
- Les raccourcis clavier pour ouvrir et sauvegarder (Ctrl+O et Ctrl+S) doivent être fonctionnels.

### *Ouverture du fichier*

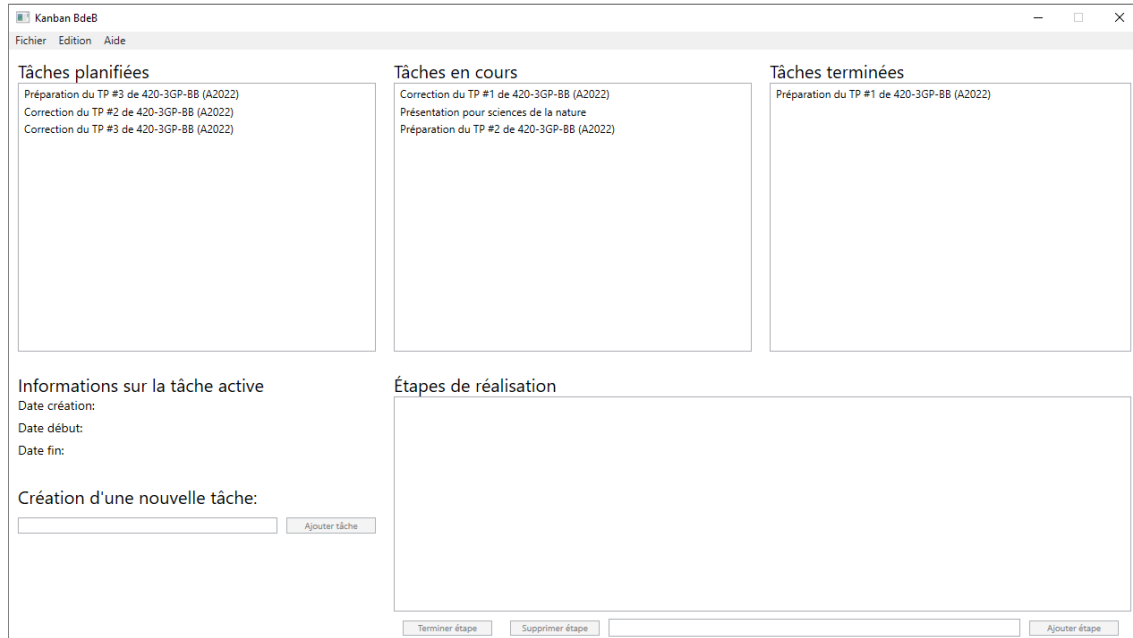
Lorsqu'un fichier de tâches sera ouvert, vous devrez l'interpréter et placer les différentes tâches en trois listes différentes. Une tâche contient les valeurs suivantes :

- Une description
- Une liste d'étapes de réalisation.
- Une date de création (le jour où la tâche a été créée)
- Une date de début (le jour où la première étape a été complétée)
- Une date de fin (le jour où toutes les étapes ont été complétées)

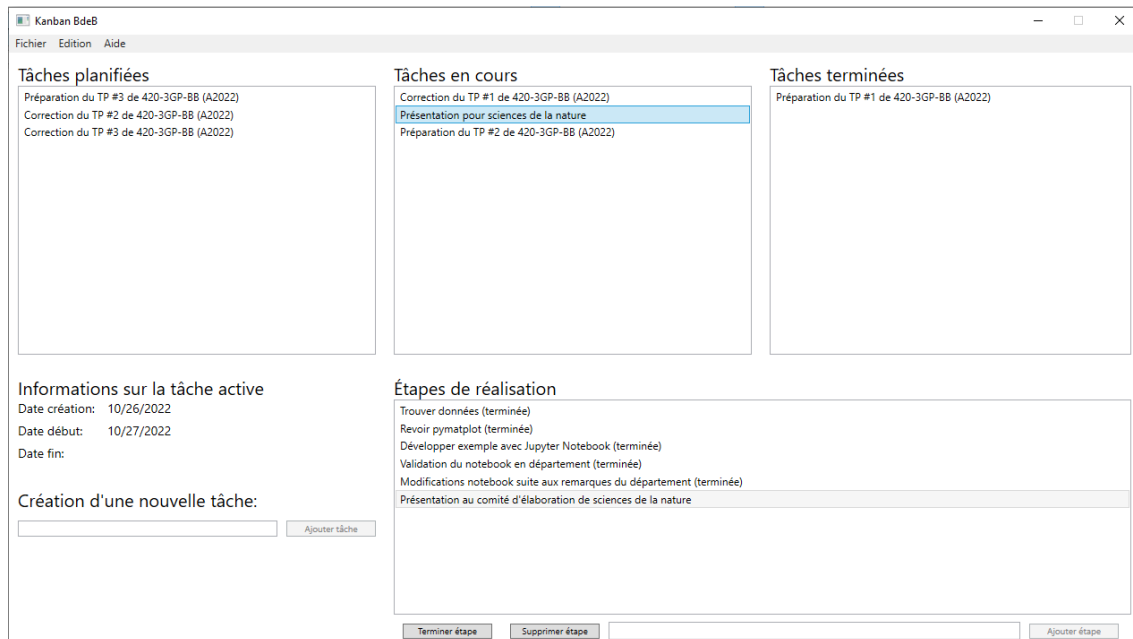
Une étape de réalisation contient les valeurs suivantes :

- Une valeur qui décrit l'étape
- Une valeur qui identifie si l'étape est terminée (True ou False)
- Un numéro

Après avoir chargé le fichier taches.xml, vous devriez obtenir le résultat suivant :



Les tâches sont placées selon leur statut (planifiée, en cours et terminée). Si on sélectionne une tâche, on obtiendra les résultats suivants :



Quand on sélectionne une tâche, ses informations doivent être affichées<sup>1</sup>.

<sup>1</sup> Vous devrez utiliser la liaison de données pour afficher les informations sur la tâche active (Data Binding)

Vous devrez indiquer les dates de création, de début et de fin s'il y a lieu :

#### Informations sur la tâche active

Date création: 10/26/2022

Date début: 10/27/2022

Date fin:

Vous devrez indiquer les différentes étapes de réalisation de la tâche :

### Étapes de réalisation

Trouver données (terminée)
Revoir pymatplot (terminée)
Développer exemple avec Jupyter Notebook (terminée)
Validation du notebook en département (terminée)
Modifications notebook suite aux remarques du département (terminée)
Présentation au comité d'élaboration de sciences de la nature

Les étapes de réalisation doivent être en ordre croissant de numéro. La première étape non complétée doit être sélectionnée par défaut. Si toutes les étapes sont complétées, alors aucune étape ne sera sélectionnée. Trois options sont ici possibles :

#### *Terminer étape*

L'étape sélectionnée sera indiquée comme terminée. Ce doit être la première étape non terminée. Sinon, la commande est désactivée. Lorsqu'on termine la première étape d'une tâche, cette tâche passe de planifiée à en cours. Lorsqu'on termine la dernière étape, la tâche est alors complétée.

#### *Supprimer étape*

L'étape sélectionnée sera supprimée. Une étape complétée ne peut pas être supprimée. La suppression d'une étape peut faire en sorte que la tâche devient complétée. Ceci se produira si on supprime la dernière étape alors que toutes les étapes avant sont complétées.

#### *Ajouter étape*

On indique une description pour l'étape et on ajoute l'étape à la fin de la liste d'étapes. Une étape ne peut pas être ajoutée dans une tâche complétée<sup>2</sup>.

<sup>2</sup> Une tâche complétée ne peut jamais être modifiée. Elle peut seulement être supprimée.

ATTENTION : Une seule tâche doit apparaître comme sélectionnée à la fois, et ce même si vous avez trois listes dans l'interface. Cette tâche doit être utilisée comme *DataContext* pour la liaison de données.

### *Création d'une nouvelle tâche*

Cette section permet de créer une nouvelle tâche :

Création d'une nouvelle tâche:

Ajouter tâche

Il suffit d'indiquer une description de tâche et de cliquer sur Ajouter tâche. La tâche sera ajoutée dans la liste des tâches planifiées même si elle ne possède encore aucune étape. Cette nouvelle tâche deviendra votre tâche active pour la liaison de données.

### *Styles*

Vous devrez définir et utiliser au moins deux styles dans votre fichier .xaml.

### *Tests unitaires*

Vous devrez faire trois tests unitaires automatisés :

- Un test qui vérifie que lorsque vous créez une tâche, son statut est : Planifiée.
- Un test qui vérifie que lorsque la première étape est complétée, son statut est : En cours.
- Un test qui vérifie que lorsque toutes les étapes d'une tâche sont complétées, son statut est : Terminée.

### *Notes additionnelles*

- Vous devrez conserver des dates pour les tâches. Utilisez pour ce faire la structure `DateOnly` de C#<sup>3</sup>.
- La mise à jour pour les étapes de réalisation ne se fera pas automatiquement suite à la commande permettant de terminer une étape. Ce n'est pas grave, il faudrait utiliser l'interface `INotifyPropertyChanged`. Nous verrons cette interface dans le troisième bloc du cours<sup>4</sup>.

### **Contraintes :**

- Le travail doit être fait en C# avec .NET 6.0 sur Visual Studio 2022. Il s'agit d'un projet WPF.
- Les boutons et les options de menus doivent être réalisés avec des *RoutedCommand*.
- Le fichier XML sert à charger et sauvegarder les données. Les différentes données doivent être conservées dans des structures de données et des classes appropriées. Les différentes opérations se font sur cette représentation des données. Elles seront ensuite sauvegardées à la demande de l'utilisateur.

<sup>3</sup> <https://learn.microsoft.com/en-us/dotnet/api/system.dateonly?view=net-6.0>

<sup>4</sup> <https://learn.microsoft.com/en-us/dotnet/api/system.componentmodel.inotifypropertychanged?view=net-6.0>

- Les informations sur les tâches doivent être affichées avec du *Data Binding* (cette contrainte n'est pas requise pour les autres éléments).
- Vous devrez adopter une approche orientée-objet :
  - classes
  - encapsulation
  - propriétés
  - méthodes

## Barème de correction

◦	<b>Qualité de la programmation</b>	<b>20 %</b>
▪	Commentaires	5 %
▪	Découpage fonctionnel	5 %
▪	Optimisation	5 %
▪	Clarté générale du code	5 %
◦	<b>Qualité de l'interface</b>	<b>10 %</b>
▪	Présence et alignement des éléments	6 %
▪	Utilisation de styles (minimum de 2 styles)	4 %
◦	<b>Tests unitaires</b>	<b>5 %</b>
◦	<b>Fonctionnement</b>	<b>65 %</b>
	<b>(Un programme qui ne compile peut se mériter 0 dans cette partie)</b>	
▪	Lecture des tâches dans le fichier	3 %
▪	Partage des tâches selon le statut	6 %
▪	Sélection d'une seule tâche à la fois	3 %
▪	Affichage des dates sur la tâche active	3 %
▪	Affichage des étapes de la tâche active	3 %
▪	Sélection de la bonne étape de réalisation	3 %
▪	Terminer étape	3 %
▪	Supprimer étape	3 %
▪	Ajout d'une étape	3 %
▪	Changement de statut d'une tâche	6 %
▪	Détermination des dates	3 %
▪	Suppression des tâches	3 %
▪	À propos	3 %
▪	Création d'une nouvelle tâche	3 %
▪	Sauvegarder	3 %
▪	Sauvegarder sous...	3 %
▪	Activation et désactivation des commandes	6 %
▪	Fonctionnement général de l'application	5 %

La qualité de la programmation et le fonctionnement sont deux critères différents. Toutefois, pour courir la chance d'avoir tous vos points pour la qualité de la programmation, vous devez tenter d'avoir fait l'ensemble du travail. Si des parties du travail sont absentes, vous serez pénalisé dans la qualité de la programmation en proportion de la partie du travail qui n'a pas été faite.

La qualité de l'interface n'est pas directement évaluée dans le fonctionnement. Toutefois, si une mauvaise qualité de votre interface nuit aux fonctionnalités de votre logiciel, vous serez pénalisé en fonction des éléments entravés par votre interface.

Le non-respect des contraintes entraînera une sévère pénalité non prévue à cette grille.