

Scrum-o-wall Journal de bord

06.04

Tâches à faire

- Créer le versionning
- Commencer la structure de la documentation
- Redéfinir le cahier des charges

Liens utiles et idées

Recherche de ressources pour faire du multi touch <https://docs.microsoft.com/fr-fr/dotnet/framework/wpf/advanced/walkthrough-creating-your-first-touch-application?redirectedfrom=MSDN> : Permet de changer la taille, tourner et déplacer en utilisant plusieurs doigts Il faut mettre tout ça dans une classe pour faire des post-it avec et pas que ce soit la fenêtre qui s'en occupe

<https://thedigitalprojectmanager.com/fr/meilleurs-outils-scrum/> : Pour l'étude d'opportunité

Tâches accomplies

- Versioning créé
- Première structure de la documentation effectuée
- Nécessite encore une restructuration de la base de données

07.04

Tâches à faire

- Restructuration de la base de données
- Création de la base de données
- Documentation de la base de données

Liens utiles et idées

Je pourrais intégrer un Mindmap en créant une table à chaque mindmap et en utilisant une table pour lier les tables avec le projet. Cependant, l'optimisation risque d'être horrible. C'est soit ça, soit avoir une méga table qui contiendrais tous les mindmap

Tâches accomplies

- Restructuration accomplies
- Base de donnée créée dans un fichier accdb
- Documentation mise à jour pour la base de données

08.04

Tâches à faire

- Création des maquettes
- Documentation des maquettes

Liens utiles et idées

J'ai fais les maquettes et la documentation de toutes les maquettes. J'ai même fais les maquettes si jamais je réussissais à faire le mindmap et le gantt (qui s'avèrera plutôt être une visualisation des tâches accomplies durant le sprint plutôt qu'un vrai gantt)

Tâches accomplies

- Maquettes créé
- Maquettes documentées sauf gantt et mindmap

09.04 (Vacances)

Pendant les vacances, je vais continuer à travailler mais à un rythme moins élevé. J'ai pas mal de trucs à faire chez moi donc je vais en profiter pour effectuer ces tâches ménagères.

Tâches à faire

- Début de la création du poster
- Création d'un icône

Liens utiles et idées

J'ai du raccourcir l'écriture du projet afin qu'il rentre dans un icône carré sans pour autant avoir de préjudices sur la lisibilité.

Tâches accomplies

- Icône créé pour github et pour l'application
- Poster commencé

14.04

Tâches à faire

- Finir le poster
- Créer le résumé

Liens utiles et idées

J'ai décidé de mettre des post-it sur le poster afin de représenter la méthode agile ainsi qu'un mindmap comme image d'arrière plan afin de faire penser à la réflexion qu'on doit avoir en permanence lors du développement agile. J'ai voulu garder le poster assez sobre et ne pas rajouter d'informations pour ne pas surcharger l'image. Je ne sais pas si le résumé est correct. Je devrais demander conseil à Mme. Terrier.

Tâches accomplies

- Poster
- Premier résumé fini

15.04

Tâches à faire

- Recherche sur les existants

Liens utiles et idées

- monday.com : https://monday.com/lp/mb/dpm/scrum/?utm_source=mb&utm_campaign=dpm&utm_medium=scrum%20
- JIRA : <https://www.atlassian.com/software/jira?r=sct>

Spécifique aux écrans interactifs <https://www.speechi.net/fr/2019/06/04/comparatif-des-outils-de-visual-management-pour-ecran-interactif/> - Ubikey : <https://www.ubikey.fr/>

- Trello : <https://trello.com/fr>
- Kantree : <https://kantree.io/fr/>
- Wrike : <https://www.wrike.com/fr/>

Tâches accomplies

- Trouver plein d'applications pareils

20.04

Tâches à faire

- Analyser les existants
- Créer les classes

Liens utiles et idées

J'ai réduit pas mal les existant pour garder que les plus pertinents. J'ai déjà créer les fonctions pour altérer les listes des classes.

Tâches accomplies

- Existant analyser
- Classes créées

21.04

Tâches à faire

-Faire une vue

Liens utiles et idées

En fait il faudrait que je fasse le controller avant de faire les vues. Je vais donc commencer le controller J'ai eu une erreur "Le fournisseur 'Microsoft.ACE.OLEDB.12.0' n'est pas inscrit sur l'ordinateur local". Je pense que je ne dois pas avoir les drivers. Après recherches, j'ai trouvé un lien en ligne : <https://www.microsoft.com/fr-FR/download/details.aspx?id=39358> De plus, Visual Studio à choisit de lancer l'application de préférence en 32 bits alors que l'engine de base de données est en 64 bits. J'ai donc du désactivé la préférence aux 32 bits dans les options de Build de l'application

Tâches accomplies

- Début de la vue (Manque la création dynamique des projets et de leurs liens respectifs)
- Singlotron de la Database fonctionnelle

22.04

Tâches à faire

- Finir le controlleur pour la vue des projets
- Finir la vue des projets

Liens utiles et idées

Je galère a trouver un algorithme pour centre verticalement les frame des projets. De plus, je risque de rencontrer un problème si il y a vraiment trop de projets dans la database. Je suis donc parti du haut de la fenêtre et j'ai mis le tout dans un ScrollViewer pour qu'une scrollbar s'affiche automatiquement quand il y a trop de contenu. Il faut encore que je crée le popup de création et j'aurais fini pour la vue des projets.

Tâches accomplies

- Fais la création et la récupération des projets
- Première vue créée avec le popup de création

23.04

Tâches à faire

- Créer la forme principale des autres vues
- Créer une méthode controller pour avoir toutes les infos des projets
- Mise à jour du planning actuel

Liens utiles et idées

Il n'existe pas de numeric updown par défaut dans WPF. Mais il existe un package nuget qui l'ajoute (Source : <https://stackoverflow.com/questions/841293/where-is-the-wpf-numeric-updown-control>) Je sais pas si je vais utilisé cela car il ne permet que peu de choses de par sa licence. Je vais chercher une autre solution. Finalement je vais juste filtrer les entrées utilisateurs et mettre un textbox. C'est le plus simple à faire sans plugin. J'aimerais quand même avoir un petit clavier virtuel qui respecterait le multi-point.

Tâches accomplies

- Planning mis à jour
- Le controlleur récupère toutes les infos au lancement de l'application
- Les vues du sprint et du backlog ne sont pas finies (sans compter le gantt et le mindmap)

24.04

Tâches à faire

- Finir les vues
- Connecter les vues entre elles

Liens utiles et idées

J'ai pensé à peut-être une protection de projet par mot de passe.

Tâches accomplies

- Vues finies
- La connection avec la vue du sprint a un problème. Il ne trouve pas le groupbox

27.04

Tâches à faire

- Régler le problème de la vue du sprint
- Faire les méthodes d'ajout

Liens utiles et idées

Le problème était que je cherchais le nom sans avoir enregistré le nom : <https://docs.microsoft.com/en-us/dotnet/api/system.windows.framework.element.registername?view=netcore-3.1> Je viens de penser au rajout de colonne et je remarque qu'il me manque des vues. Il me faut un popup de création de colonnes et un popup de gestion de colonnes

Tâches accomplies

- Problème de la vue du sprint réglé
- Export des méthodes d'accès à la DB dans la classe DB
- Modification de l'orthographe de la doc
- Analyse et priorisation des fonctions du trello

28.04

Tâches à faire

- Approfondir l'analyse approfondie de Trello
- Restructurer la documentation
- Modifier la base de données et la documentation pour s'adapter à l'analyse de Trello.

Liens utiles et idées

Tâches accomplies

- Analyse sur Trello approfondie
- Documentation restructurée
- Changement du Modèle Conceptuel de Données et documentation des changements

29.04

Tâches à faire

- Modifier la base de données
- Créer/Modifier les classes

Liens utiles et idées

Je ne trouvais pas le diagramme de classe mais après une recherche il fallait l'installer par le Visual Studio Installer

Tâches accomplies

- Base de données modifiée
- Création des classes et modification des existantes
- Création du diagramme de classe

30.04

Tâches à faire

- Modification des maquettes existantes
- Refaire des maquettes pour les fenêtres manquantes
- Faire un plan de tests

Liens utiles et idées

Tâches accomplies

- Modification des maquettes
- Maquettes pour les fenêtres manquantes
- Début du plan de test

01.05

Tâches à faire

- Finir le plan de test

- Faire la réunion avec Mme. Terrier
- Faire le visuel des maquettes

Liens utiles et idées

Pour la gestion des listes il faudra que je reproduise ça en code :

```
<Border BorderBrush="Black" BorderThickness="1" Width="408" >
  <Grid x:Name="list1" >
    <Grid.ColumnDefinitions>
      <ColumnDefinition Width="30*"></ColumnDefinition>
      <ColumnDefinition Width="67*"></ColumnDefinition>
    </Grid.ColumnDefinitions>
    <TextBlock VerticalAlignment="Top" TextWrapping="Wrap" Margin="10,10,10,0" TextAlignment="Right"></TextBlock>
    <ListView Grid.Column="1" VerticalAlignment="Top" x:Name="listItems" Margin="10,10,10,10" Height="76">
      <CheckBox IsChecked="True">asdasd</CheckBox>
      <CheckBox IsChecked="False">asdasd</CheckBox>
      <CheckBox >asdasd</CheckBox>
    </ListView>
  </Grid>
</Border>
```

Tâches accomplies

- Plan de test des vues
- Réunion avec Mme.Terrier
- Fais la moitié des visuels

04.05

Tâches à faire

- Finir les visuels
- Créer les constructeurs des vues
- Placer les valeurs dans le visuel

Liens utiles et idées

Trouvé de quoi faire la ligne pour le burndown chart <https://www.c-sharpcorner.com/UploadFile/mahesh/polyline-in-wpf/>

Tâches accomplies

- Finir les visuels
- Fais les vues suivantes ; Burndown Chart, ChecklistCreate, ActivitesMenus, CheckListItemCreate, ChecklistMenu,CommentCreate, CommentsMenu, MindMapCreate, CommentCreate, CommentMenu

05.05

Tâches à faire

- Faire une partie des vues restantes

Liens utiles et idées

Tâches accomplies

- Fini les vues : FileCreate, FileEdit, FileMenu, ProjectEdit, StateCreate, StateMenu, UserCreate, UserEdit, UserStoryCreate
- Il reste UserStoryEdit et UserMenu

06.05

Tâches à faire

- Faire les vues restantes
- Corriger le poster

Liens utiles et idées

Afin de pouvoir utiliser le menu des utilisateurs j'ai du créer une interface car plusieurs classes différentes l'utiliseront. J'ai découvert grâce à cela que la version 8.0 de C# permet d'implémenter les propriétés. Cependant, vu que j'utilise la version 7.3, j'ai dû créer des méthodes pour altérer la liste d'utilisateurs.

Tâches accomplies

- Fini les vues : UserStoryEdit et UserMenu
- Création d'une interface
- Rajout du langage de programmation sur le poster

07.05

Tâches à faire

- Refaire la documentation sur les maquettes écrans
- Refaire la documentation des classes

Liens utiles et idées

Important de citer que j'ai voulu utiliser du déconnecté Expliquer la stratégie Expliquer la notice de connection à Access Pourquoi j'ai fait une classe d'interface, pourquoi j'ai fait ci et ça. Les challenges du WPF. Ce qui est lié à l'écran tactile.

Tâches accomplies

- Documentation sur les maquettes écrans refaite
- Les diagrammes de classes ont été ajoutés et nécessitent une documentation plus poussée
- Correction de bugs empêchant la compilation

08.05

Tâches à faire

- Implémenter les méthodes de la base de données

Liens utiles et idées

J'ai eu un problème sur la base de données. Il y avait une erreur lors de l'ajout d'un champ pouvant être nul. Après une recherche, j'ai découvert grâce à cette source : <https://www.c-sharpcorner.com/article/enter-null-values-for-datetime-column-of-sql-server/> qu'une classe DBNull possédait une constante pour les valeurs nulles des bases de données.

Tâches accomplies

- Implémentation des méthodes de la base de données

11.05

Tâches à faire

- Rajouter la gestion de l'écran tactile.
- Inclure le drag'n'drop sur ProjectMenu

Liens utiles et idées

J'ai remarqué qu'il me manquait une vue la dernière fois. Il me manque ChecklistItemEdit.

En plus, il faut que je commence à m'intéresser au drag'n'drop en wpf. J'ai trouvé ce post sur stackoverflow qui pourrait m'en apprendre plus : <https://stackoverflow.com/questions/11306194/wpf-dragenter-between-two-canvas-not-firing> Je vais essayer de faire un drag and drop sur le menu du projet.

Le post stackoverflow renvoie sur un tutoriel de wpftutorial : <http://wpftutorial.net/DragAndDrop.html>.

J'ai un problème pour la liaison du sprint avec les user stories "INSERT INTO TUserStoriesSprint (IdUserStory,IdSprint,Order) VALUES (2,3,1);" ne fonctionne pas avec erreur de syntaxe mais "INSERT INTO TUserStoriesSprint VALUES (2,3,1);" fonctionne.

Après avoir cherché beaucoup trop de temps pour que ce soit raisonnable, j'ai trouvé le problème. Le mot order est réservé par SQL pour le "ORDER BY". j'ai donc renommé le nom du champ par "OrderUserStory".

Tâches accomplies

- Gestion de l'écran tactile accomplies
- Le drag'n'drop a été ajouté sur ProjectMenu

12.05

Tâches à faire

- Inclure le drag'n'drop sur SprintMenu
- Rendre le drag'n'drop plus évident

Liens utiles et idées

J'ai effectué le drag'n'drop mais les user controller ne sont pas effacés. Il faudrait que je mette à jour uniquement le bon UserControl ou que je remette à jour correctement tous les éléments de la fenêtre.

J'ai choisi la deuxième option et j'ai du refaire totalement la mise à jour de la vue. Cela m'a pris plus de temps que prévu et je n'ai pas le temps de faire ma deuxième activité.

Tâches accomplies

- Drag'n'drop sur sprint menu effectué

13.05

Tâches à faire

- Faire le résumé
- Faire l'abstract
- Rendre le résumé et l'abstract

Liens utiles et idées

Ayant un vocabulaire en français plus fourni que mon vocabulaire anglais, j'ai fait d'abord l'anglais puis j'ai traduit de l'anglais au français.

Tâches accomplies

- Faire le résumé
- Faire l'abstract
- Rendre le résumé et l'abstract

14.05

Tâches à faire

- Faire des tests
- Régler les problèmes
- Créer les fonctionnalités visuels

Liens utiles et idées

Problèmes réglés : - Le wrap du texte dans des textbox multiligne - Création de fichier (Mauvais nom de table) - Création de commentaires (DateTime = mot réservé) - Création de checklist (Mauvais nom de table) - Checkbox to ChecklistItem - Changement de la taille de la police de base pour plusieurs fenêtres - Liste de ChecklistItem non initialisée

Fonctionnalités visuels : - Couleur sur les sprints (Rouge = déjà passé, Bleu = en cours, Gris = à venir) - Augmentation de la taille des User Stories et des Sprints - Limiter les caractères

Autres : - Mise en ordre de la classe DB

Idées : - Suppression de fichier dans l'user story - Visuel des sprints sous forme de GANTT

Tâches accomplies

- Tests effectués
- Problèmes réglés
- Rajout de couleur pour les sprints
- Rajout d'une limitation sur l'interface utilisant
- Augmentation de la hauteur des éléments
- Création de la fenêtre ChecklistItemEdit

15.05

Tâches à faire

- Retoucher le résumé et l'abstract
- Rendre la nouvelle version
- Implémenter la fenêtre ChecklistItemEdit dans le fonctionnement :
 - Accès
 - Modification après fermeture validée

Liens utiles et idées

J'ai d'abord commencé à faire le résumé et l'abstract puis j'ai rendu sur moodle. Cependant, l'heure de rendu était mise à minuit alors qu'elle était annoncée à 17h. J'ai donc envoyé un message à M. Garcia.

J'ai implémenter la fenêtre d'edition de la checklistItemEdit. J'en ai profité pour modifier quelques fonctions redondantes d'évènement en acceptant l'interface EventArgs au lieu d'accepter les évènements de type plus particulier.

Tâches accomplies

- Retoucher le résumé et l'abstract
- Rendre la nouvelle version
- Implémenter la fenêtre ChecklistItemEdit dans le fonctionnement : -- Accès -- Modification après fermeture validée
- Ajout de limitations sur les utilisateurs assignés

18.05

Tâches à faire

- Implémenter l'ajout d'activités
- Refactoriser le code

Liens utiles et idées

Pour implémenter l'ajout des activités, j'ai pensé qu'il serait mieux que rien ne soit lié à l'interface utilisateur. Cependant, j'ai jugé plus intéressant que ce soit le contrôleur qui s'occupe d'appeler la création des activités. J'ai donc rajouter dans chaque fonction changeant les User Stories un appel à la fonction rajoutant un enregistrement dans les activités.

Après cela, j'ai refactoriser le code en supprimant les fonction redondantes. J'ai ensuite séparé les fonction d'évènement de chaque vue des autres fonctions plus utiles et j'ai regrouper les fonctions dans le controller par leur but tel que la création, mise à jour ou suppression d'enregistrement à la base de données.

J'ai commencé à vouloir faire des tests unitaires. Cependant, pour définir la base de données j'ai dû changé le fonctionnement de la classe DB. Il faut absolument pouvoir définir une base de données sans interface graphique. C'est pourquoi j'ai créer une propriété qui peut être définie pour ouvrir une base de données sans avoir à utiliser l'interface utilisateur

Demain, je m'occuperai de la création des tests unitaires

Tâches accomplies

- Implémenter l'ajout d'activités
- Refactoriser le code
- Définition des tests unitaires

19.05

Tâches à faire

- Implémenter les tests unitaires

Liens utiles et idées

J'ai eu mon fameux problème de moteur de base de données en voulant lancer les tests. J'ai donc dû forcer l'application à se lancer en 64 bits pour que cela fonctionne.

J'ai décider de tester uniquement la classe Controller et la classe DB.

Tâches accomplies

- Tous les tests du controller
- Les tests de lecture de la classe DB.

20.05

Tâches à faire

- Finir les test unitaires
- Rendre le drag'n'drop multipoint

Liens utiles et idées

J'ai fini de faire les tests unitaires pour la DB et j'ai corrigé quelques problèmes sur le controller.

Afin de rendre le drag'n'drop multipoint, plusieurs posts stackoverflow (<https://stackoverflow.com/questions/3191084/wpf-4-multi-touch-drag-and-drop> , <https://stackoverflow.com/questions/3044834/drag-and-drop-as-multitouch-event-in-wpf> , <https://stackoverflow.com/questions/8444998/multi-touch-screen-and-wpf-listbox> , et bien d'autres) me disent d'utiliser le "Surface Toolkit for Windows Touch". Je vais donc faire un git push et essayer en installant le surface toolkit.

Résumé de la visioconférence : Faire un flag pour supprimer quelque chose plutôt que de le supprimer totalement

J'ai essayer de trouver comment importer le Surface Toolkit mais après un moment à essayer de trouver sans succès comment l'importer, je suis parti sur une solution personnalisée.
Je garde en mémoire les informations de l'objet à drag'n'drop dans des listes. J'ai également rajouté une petite boite qui suit le curseur pour avoir un support visuel.

J'ai eu des soucis sur le PreviewTouchDown qui ne s'activaient pas. Après des recherches, le post de stackoverflow : <https://stackoverflow.com/questions/36656709/wpf-touchscreen-events-not-working-properly> m'a permis de découvrir que le problème était dû au fait que WPF s'attend à la réaction d'un stylet et non à réagir rapidement sur une touche d'utilisateur. Pour changer cela, il a fallut que je désactive cette propriété sur le contrôle voulu.

J'ai également rajouté un style au drag and drop en affichant une petite boite à l'endroit ou on bouge la boite. Le message de confirmation s'affiche maintenant de manière asynchrone pour ne pas perturber le drag'n'drop des autres gens.

Tâches accomplies

- tests unitaires finis
- Drag'n'drop multipoint sur ProjectMenu
- Drag'n'drop stylisé

21.05

Tâches à faire

- Changer le drag'n'drop sur SprintMenu en multipoint
- Retoucher la documentation

Liens utiles et idées

Pour pouvoir changer le drag'n'drop, j'ai refais comme dans la fenêtre ProjectMenu. Pour ce qui est du déplacement, j'ai tout mis dans l'évènement de la fenêtre directement. Cependant, la zone qui est détectée par l'UI comme étant la zone du groupbox est uniquement sur les bords. C'est pourquoi j'ai récupéré les limites du groupbox et j'ai vérifié si le drag se situait dans la limite du groupbox.

De plus, relacher le drag'n'drop avec un autre drag'n'drop actif recharge bien l'emplacement des userStories sans pour autant casser le drag'n'drop actuel.

Pour la documentation, j'ai vérifié que toutes les images possédaient bien une légende et j'ai rajouté les fonctionnalités et le résultat de mes tests unitaires. Il faudra que je mette à jour le diagramme de classes demain.

Tâches accomplies

- Changer le drag'n'drop sur SprintMenu en multipoint
- Retoucher la documentation

22.05

Tâches à faire

- Mettre à jour les figures
- Mettre à jour la doc

Liens utiles et idées

En mettant à jour les différentes figures, il m'est venu à l'idée de commencer l'implémentation dont j'avais parlé avec Mme. Terrier. J'ai donc rajouté un champ "deletedFlag" dans les tables qui me semblaient pertinentes. J'ai donc omis de le mettre dans les tables de liaison. Je n'ai pas encore implémenter l'utilisation de ce flag mais je le ferais Lundi.

J'ai refais une base de données entièrement vierge et je la garde de côté pour ne pas avoir trop de données.

J'ai restructuré le github car il était un peu trop brouillon. J'ai créé un dossier Documentation pour tout ce qui est documentation et illustrations. Un dossier Installation pour la base de données vierge et, qui sera rajouté à la fin, l'exécutable de l'application. Le dossier Scrum'o'Wall est toujours mon projet C#.

Tâches accomplies

- Mise à jour des figures
- Mise à jour de la doc
- Changement dans la base de données (Rajout de flag pour la suppression)
- Restructuration du github

25.05

Tâches à faire

- Implémenter l'utilisation du flag de suppression (Modification de DB - GET, Liaison et DELETE)

Liens utiles et idées

Le problème principal que je trouve avec l'implémentation de flag contre la suppression dans la base, je dois donc changer les méthodes du controller pour faire des suppressions en cascade. Je vérifie également que les objets existent bien avant d'effectuer les liaisons au lancement de l'application.

Pour ce qui est de l'acquisition des données, j'ai rajouté pour chaque table pertinente une condition me permettant de récupérer uniquement les bonnes valeurs.

Comme il me restait du temps, j'ai commencé à réaliser l'interface utilisateur pour la suppression. J'ai déjà effectué ce qu'il faut sur la fenetre d'edition des fichiers en rajoutant un bouton supprimer. J'ai trouvé plus pertinent de rajouter un champ "UserStory" contenant l'objet parent du File dans le fichier plutôt que de passer le userStory en paramètre à travers toutes les vues.

J'ai retapé un peu la documentation. Cependant, il y a quelques petites modification que j'ai préféré ne pas changé. Par exemple: Les faiblesses dans le SWOT sont obligatoires sinon le SWOT ne serait pas complet. Pour le commentaires sur le Modèle de classe (description de la communication) je pense rajouter un schema afin d'exemplifier plus facilement.

J'ai donc: supprimer le manuel utilisateur, déplacer le manuel d'installation, déplacer le planning

Tâches accomplies

- Implémentation du flag de suppression dans DB
- Implémentation de la suppression des fichiers
- Retouche de la documentation

26.05

Tâches à faire

- Rajouter la suppression sur les vues
- Faire le diagramme de communication vers la base de données
- Faire le point avec Mme. Terrier

Liens utiles et idées

J'ai commencé par faire le diagramme de communication. J'ai choisi de faire un diagramme de séquence. Cela permet de mieux montrer quels actions provoquent un accès à la base de données.

J'ai ensuite eu une réunion avec Mme Terrier.

Mettre en évidence des points subtils : - La communication avec Access - Expliquer le WPF - Drag'n'drop

Chapitre sur l'apport personnel : - Préciser ce qui est récupérer et pas

Conclusion : - Personnel de comment vécu le travailler - Points d'améliorations

J'ai finalement travaillé sur l'ajout de suppression dans les vues. J'ai eu le temps de créer tous les boutons sur toutes les vues et de créer les liaisons entre les vues. Il me manque à implémenter les fonctions dans le controller

Tâches accomplies

- Rajouter la suppression sur les vues
- Faire le diagramme de communication vers la base de données
- Faire le point avec Mme. Terrier

27.05

Tâches à faire

- Implémenter les fonctions de suppression dans le controller.

Liens utiles et idées

J'ai implémenter les fonctions en les reliant aux fonctions correspondantes et j'ai créé les fonctions manquantes dans DB. J'ai également changé la structure des classes afin de contenir directement un objet parent (La user story pour le commentaire ou le projet pour le user story)

J'ai également rajouter un attribut "Deleted" dans les vues de modifications pour pouvoir savoir à la fermeture si l'utilisateur a appuyé sur le bouton de confirmation ou le bouton de suppression. En effet, j'avais commencé par utiliser l'attribut DialogResult des fenêtres car il utilise un booléen nullable ce qui, dans ma première idée permettait de renvoyer 3 types de données différentes avec "Vrai,Faux et Null". Cependant, à la fermeture et avec un dialogResult égal à null, les

Tâches accomplies

- Implémenter les fonctions de suppression dans le controller.
- Implémentation des fonctions manquantes dans la DB.
- Modification des vues pour intégrer la suppression.

28.05

Tâches à faire

- Refaire les tests unitaires

- Régler les problèmes de boutons qui s'interclickent entre les vues

Liens utiles et idées

Grâce au site <https://wpf.2000things.com/tag/touchdown/> j'ai découvert l'ordre de lancement des événements liés au Touch ce qui me permet de déterminer quel événement il faut appeler.

J'ai réglé le problème des cliques non-attendus en utilisant la méthode TouchUp à la place de TouchDown pour les boutons.

Refaire les tests m'a pris bien plus de temps que prévu. Je ne réussirais pas à terminer à temps. J'ai renommé certaines fonction et je les ai retravaillées pour qu'il n'y ait plus d'erreur peu importe le cas. Je dois encore repasser sur les méthodes de la classe Controller.

Tâches accomplies

- Problèmes des boutons régler
- Tests unitaires de la classe DB refait et documentés.

29.05

Tâches à faire

- Finir les tests unitaires pour le Controller

Liens utiles et idées

J'ai commencé directement par changé les tests unitaires du Controller. Ce qui m'a fait perdre beaucoup des méthodes que j'avais avant en passant le nombre de tests de plus d'une trentaine à 17 pour le controller. J'ai également fait attention à ce que les tests ne puissent pas échouer si aucun enregistrement non nécessaire (autre que les type et le priorité) n'est présent dans la base.

Après cela, j'ai mis à jour différents éléments de la documentation tel que les illustrations.

Tâches accomplies

- Finir les tests unitaires pour le Controller et les documenter
- Mise à jour des illustrations de la doc

01.06

Tâches à faire

- Trouver une solution pour le clavier virtuel
- Implémenter les méthodes pour le mindmap et les node

Liens utiles et idées

Alors que je cherchais un moyen de créer un clavier visuel, je suis tombé sur une astuce pour Windows 10 qui permet d'afficher le clavier visuel si aucun clavier n'est connecté sans que l'ordinateur soit en mode tablette dès que le focus sur un champ de type texte est lancé. <https://blog.mzkmund.com/2015/09/how-to-show-touch-keyboard-on-touch-interaction-with-wpf-textboxes/> Pour cela, il faut aller dans les options > Périphériques > Saisie et activer le champ "Afficher le clavier tactile lorsque vous n'êtes pas en mode tablette et qu'aucun clavier n'est connecté".

Cependant, le mode tablette ne peut pas être activé avec plus d'un écran ce qui rend le réglage de cette option obligatoire et qui rend la connexion d'un clavier non-recommandée. (<https://answers.microsoft.com/fr-fr/windows/forum/all/le-mode-tablette-est-gris%C3%A9e-impossible-de/cff5b45b-e229-48dc-ad16-e85335189b45?auth=1>)

J'ai rajouté un chapitre dans le manuel d'installation pour le clavier virtuel.

Pour le mindmap et les nodes, j'ai d'abord commencé par créer les méthodes dans la classe DB. J'ai ensuite continué par le Controller et j'ai fait le CRUD dans le test unitaire pour les deux classes. Après avoir fait les tests unitaires et qu'ils soient valide, j'ai changé la documentation pour coller aux résultats.

Tâches accomplies

- Trouver une solution pour le clavier virtuel
- Implémenter les méthodes pour le mindmap et les node
- Création des tests unitaires pour le mindmap et les node

02.06

Tâches à faire

- Rajouter un accès au mindmap depuis le menu du projet

Liens utiles et idées

Pour rajouter l'accès au mindmap, j'ai préféré rajouter une colonne dans la fenêtre du menu du projet. En effet, les mindmaps sont liés aux projets et il me semble

logique que ce soit leur place. Après cela, j'ai créé les liaisons entre les différentes fenêtres afin d'afficher la fenêtre de création de noeud quand il faut, etc...

Après avoir placé les contrôles correctement sur la fenêtre de projet, j'ai décidé de refaire un passage sur les différentes vues afin de normaliser un peu l'apparence. Maintenant, les pages pleines contiennent des boutons ronds et les pages pop-up contiennent des boutons carrés.

Tâches accomplies

- Accès aux mindmaps créés.
- Création de noeud créé
- Normalisation de l'interface utilisateur
- Événements de la fenêtre du mindmap créés.

03.06

Tâches à faire

- Dessiner les Node
- Faire le chapitre de l'apport personnel au projet
- Visio avec Mme Terrier

Liens utiles et idées

Pour dessiner les nodes, je pense faire une arborescence comme les fichiers de répertoires d'une commande tree. Je mettrais donc le noeud "root" en haut à gauche et je vais créer des colonnes pour chaque profondeur.

Reunion avec Mme Terrier Questions: - Pour le rendu c'est quoi un export git ?

Rendre quelque chose de non exécutable aux experts

Défense à blanc le mercredi 10 juin. Montrer la structure d'un test unitaire, la structure du drag'n'drop multipoint.

Pendant que j'étais entrain d'essayer de créer correctement les node, j'ai remarqué que les bordures du drag'n'drop multipoint restaient parfois sur place lors du relâchement et lorsqu'un user story était ouvert. J'ai réussi à enlever la zone lors du relâchement mais pas quand l'user story s'ouvre. Cependant, j'ai essayé de supprimer l'apparition de la boîte à l'édition d'un user story mais je n'ai pas réussi. Je n'ai pas eu d'autres choix que d'enlever l'aspect visuel du drag'n'drop

Tâches accomplies

- Vision avec Mme Terrier
- Dessin des Node
- Correction de l'aspect du drag'n'drop multipoint
- Chapitre sur l'apport personnel réalisé

04.06

Tâches à faire

- Corriger des bugs
- Faire un test de l'interface utilisateur

Liens utiles et idées

Afin d'avoir une interface de mindmap plus simple à créer et modifier, j'ai décidé de changer l'utilisation du canvas par une grid. Je crée ensuite des lignes pour chaque node dessiner et je rajoute une colonne à chaque fois que le nombre de colonnes requises est insuffisant.

J'ai ensuite profité d'une opportunité que mes camarades m'ont proposé de leur montrer mon application pour faire un test en live. J'ai grâce à cela découvert des bugs dans la modification des fichiers et l'obtention des commentaires au chargement de l'application.

J'ai corrigé ces problèmes et j'ai suivi un conseil qui m'a été donné pour certains aspects. Par exemple, afficher la date et l'heure d'une activité ou attribuer une couleur aux user stories possédant une date limite.

Tâches accomplies

- Corriger les problèmes de dessin des node
- Corriger la modification des fichiers
- Corriger l'obtention des commentaires
- Ajout de couleurs aux user stories avec une date limite
- Modifications sur l'affichage de certaines classes

05.06

Tâches à faire

- Commenter les endroits manquants
- Faire le planning final

Liens utiles et idées

N'ayant pas commenté entièrement mon projet, il a fallu que je repasse sur certains fichiers pour rajouter des précisions sur certaines lignes de code. J'ai également mis un entête sur les fichiers n'en possédant pas.

Après cela, j'ai normalisé les variables et les méthodes du projet en mettant les variables en readonly ou en mettant des majuscules au début des fonctions.

J'ai finalement repassé sur le planning avant de m'atteler à des morceaux de documentation que je n'avais pas encore créés. J'ai rempli la section sur le drag-and-drop

Tâches accomplies

- Commenter les endroits manquants
- Faire le planning final
- Remplir le glisser-déposer dans la doc

06.06

Tâches à faire

- Documentation. chapitre conclusion

Liens utiles et idées

J'ai commencé par les difficultés rencontrées, j'ai continué sur les améliorations possibles et finalement mon ressenti sur le projet.

Tâches accomplies

- Documentation. chapitre conclusion

07.06 - 08.06

Tâches à faire

- Récupération du code source
- Création du manuel utilisateur
- Mettre le journal de bord en pdf
- Rendu du projet

Liens utiles et idées

J'ai commencé par créer le manuel utilisateur. j'ai décidé de faire une description très détaillée des menus principaux puis une description plus brève des différents pop-up. J'ai donc utilisé GIMP afin d'identifier les principales composantes des vues pour les expliquer au mieux.

Tout en faisant le manuel utilisateur, j'ai découvert certains problèmes que j'ai corrigés en même temps.

Problèmes trouvés : - Delete checklistItem, checklist, file, - Création commentaires - Modification sprint + Etat (Texte)

Après cela, j'ai utilisé les indications de M. Maréchal sur le moodle afin de rendre mon code source en un fichier pdf avec LaTeX.

Finalement, je vais récupérer le journal de bord que voici et le mettre au format pdf afin de le rendre avec la documentation.

Tâches accomplies

- Bug fix
- Récupération du code source
- Création du manuel utilisateur
- Journal de bord en pdf + rendu (futur instantané)