

# CR du 04/02/2021 - Structuration des idées et diagramme de Gantt



Salomé Guinaudeau, Paul Bertho, Julien Priam, Théo Prigent

Réunion précédente: CR du 01/02/2021 - Réorganisation des versions, timings\*

## Ordre du jour

- ☐ Établir un diagramme de Gantt prévisionnel
- ☐ Répartition des tâches
- ☐ Réflexions sur l'organisation du logiciel

## Contenu de la réunion

Le but principal de cette réunion étant l'élaboration du diagramme de Gantt, il en découle forcément de nombreuses réflexions quant à l'organisation du projet, et aux méthodes de réalisation.

## Répartition des tâches

Dans un premier temps, le choix a été fait de séparer l'équipe en deux pour avancer en parallèle. Ainsi, Salomé et Paul s'occupent de la partie gestion des données, et Julien et Théo de l'interface graphique. Cette organisation pourra très bien être remise en cause si la base de données est finalisée tôt dans le projet.

La partie interface entre la base de donnée (BDD) et l'interface graphique (UI) sera gérée par les deux équipes, en fonction de l'avancement et des facilités.

## Organisation du logiciel

Cette réunion a donné lieu à une première réflexion sur l'organisation de l'interface entre l'UI et la base de données. L'équipe a discuté de plusieurs stratégies.

En ce qui concerne les données patient (**anamnèse** dans le monde du médical):

- Solution 1: Créer une structure *Patient* qui contiendrait tous les attributs présents dans l'anamnèse. Cette structure serait remplie par les informations de la BDD à l'aide d'une fonction contenant toutes les requêtes, et ce, à l'ouverture du dossier patient. Ainsi, l'UI n'aura qu'à chercher les informations qui lui sont utiles dans cette structure, lors de l'affichage de la vue. De même, lors d'éventuelles modifications de ces informations, l'interface graphique récupérerait les nouveautés et les placerait dans la structure, à l'appui du bouton "enregistrer", la base de données n'aurait plus qu'à faire appel à une requête qui recopie tous les champs de la structure dans la BDD, sans distinction
- Solution 2: Ne pas utiliser de structure, et faire les requêtes SQL directement là où on définit le texte à afficher dans l'interface graphique. Cette solution paraît moins avantageuse en terme d'organisation du code et de performances.

En ce qui concerne les séances:

La question se pose de savoir comment gérer un nombre potentiellement élevé de séances. Pour faire le lien entre la BDD et l'UI, l'équipe a pensé à utiliser une structure de données sous forme de liste. À la sélection du dossier, la liste serait d'abord complétée par la BDD, et l'UI pourrait ainsi l'avoir à disposition pour l'interpréter comme elle le souhaite, tout comme la gestion de l'anamnèse.

Cependant, la question se pose de savoir si cette stratégie serait bénéfique dans un souci de lourdeur de code, et si simplement faire les requêtes sur signal de l'interface graphique ne serait pas suffisant.

## UI

Il a été relevé qu'il allait falloir attacher une importance particulière à la segmentation du code et trouver des moyens de créer des fonctions afin qu'elles puissent être réutilisées, ou au moins déportées afin de proposer un code qui soit le plus léger et compréhensible possible.

## Conclusion

Il est difficile au début d'un tel projet d'établir précisément l'enchaînement et les dates de chaque phase de développement. L'équipe a cependant réussi à proposer une prévision qui semble pouvoir tenir la route, même s'il est évident que ce diagramme de Gantt sera amené à évoluer en fonction des difficultés et facilités rencontrées.



Pour la suite, l'apprentissage des langages de BDD et d'UI sont au programme jusqu'à la fin de cette semaine. L'objectif est de commencer à coder la semaine prochaine.