

Labo 3 - Documentation

Expliquer brièvement l'architecture de l'application

D'un point de vue technique, l'application a une architecture assez simple.

Pour nous faciliter le développement, nous avons par exemple mis toutes nos importations de modules ainsi que les déclarations de nos composants dans ``app.module.ts``. Pour une application complexe, il aurait fallu les séparer. Ici, ce n'était pas nécessaire.

Pour ce qui est des routes, c'est également simple. Il n'y a pas de sous-routes ou bien de variables compliquées. Uniquement des routes simples qui pointent vers des composants simples. Pour nos composants, notre architecture de l'application a déjà été réfléchi dans le MVP et dans nos wireframes du labo 2. En effet, nous avons un dashboard qui est la page centrale de notre application. Une sidebar sur le côté qui utilise nos différentes routes pour naviguer dans celle-ci.

Pour ce qui est des autres composants, l'affichage était le plus important. Nous avons mis en place un JSON server avec des données fictives pour simuler une API REST. Étant donné que ce module est sur l'UI et l'UX et non sur le back-end le CRUD n'est pas respecté, car nous voulions avoir une version V1 tirée de notre MVP pour pouvoir potentiellement la vendre à un client. Nous avons jugé qu'il n'était donc pas nécessaire de faire un affichage dynamique en ce qui concerne l'ajout et la modification de note. Seulement l'affichage était important pour le groupe. Par la suite, si le client est intéressé par notre application, le CRUD représente la prochaine étape de notre projet pour avoir quelque chose de réellement fonctionnel. Nous avons également choisi de ne pas mettre les descriptions des modules dans notre applications. En effet, par manque de temps, nous avons dû faire des sacrifices et cette partie n'était pas primordiale et ne correspondait plus vraiment à un MVP.

En gardant en tête notre vision d'aller droit au but vers quelque chose de visuel, nous avons utilisé les composants angular de Material avec un peu de css custom.

Critiquer le travail réalisé

Décrire ce qui a bien été réalisé :

Tout d'abord, une des principales forces de notre groupe était notre entraide. Comparé au labo 2 où la communication pouvait faire défaut, nous avons cette fois-ci fait le maximum pour tous s'entraider si nous étions dans le besoin. Le partage de connaissances a été incroyable

tout au long de notre projet. Cela nous a permis de vraiment simuler un travail en équipe comme si nous étions dans une vraie entreprise. Une expérience qui est à chérir.

Nous avons vraiment simulé une petite entreprise qui utilise la méthode agile pour fonctionner. Avec un Product Owner et une équipe de développement. Ceci nous a permis de prendre des décisions tous ensemble et cela a grandement aidé à l'avancement de notre application.

Décrire ce qui pourrait être amélioré et expliquer comment

Pour la plupart des points d'améliorations, nous nous sommes rendus compte des défauts de notre application durant son développement. Par exemple, les routes de notre application et le choix des boutons de la sidebar ne sont pas parfaites. En effet, il est compliqué de naviguer convenablement dans notre application. Cependant, nous voulions respecter la maquette le plus possible et nous avons donc décidé de ne pas améliorer cet aspect et de laisser la SideBar telle quelle. C'est en revanche, quelque chose à améliorer lors d'une V2 par exemple.

De plus, le développement sur angular s'est avéré particulièrement compliqué. En effet, le temps mis à disposition pour faire le labo 3 était assez court et personne de notre groupe ne connaissait angular. Il a donc fallu tout apprendre de zéro et créer exactement ce qu'on voulait était très difficile. Des sacrifices pour le bien de l'application ont dû être faits. Par exemple, la modification des notes et l'ajout de celles-ci n'ont pas été liés au back-end. Ou encore, les moyennes ne sont pas calculées dynamiquement. Tout ce qui n'a pas été implémenté dans le code a été choisi par l'équipe entière. Notre but était de créer une application « modèle » qui pourrait être présentée à un potentiel client pour la vente de celle-ci. Le calcul des moyennes n'était, par exemple, pas indispensable. Seulement le visuel l'était.

En plus de tout ça, le développement sur git, étant nouveau pour certains, nous a demandé de prendre beaucoup de temps pour pouvoir mettre tout en place. Un tutoriel a été créé (mis en annexe) pour ceux dans le besoin, mais la complexité de git nous a fait perdre beaucoup de temps. Ce n'est pas la même chose de travailler seul qu'à 5 ! Il y a eu énormément de conflits de code à gérer. Bien qu'ils étaient simples, la non-connaissance de notre support pouvait en faire peur à plus d'un !

Notes supplémentaires

Nous tenons malgré nos difficultés, à noter que nous sommes tous reconnaissant d'avoir pu travailler tous ensemble. Cela nous a permis de voir la réalité du travail en face et d'apprendre plein de choses que ce soit d'un point de vue technique ou humain. Nous sommes fiers de notre application et du chemin parcouru.