

# Toppics

CANTA Carmelo - DEHONDT Léo - REYES Adam

## Rapport de fin de projet



Votre nouveau réseau social Toppics est en marche ! Vous trouverez dans ce document toutes les informations concernant l'état du projet lorsque nous vous l'avons rendu.

Le précédent cahier des charges est aussi fourni dans le GitHub que vous avez reçu, puisqu'il possède tous les éléments UX/UI, les choix techniques ainsi que les architectures de notre projet.

## I) Description de l'application

Le fonctionnement de notre application est le suivant : lorsque l'utilisateur est inscrit et connecté sur Toppics, il peut accéder à la page permettant de poster une publication. Sur cette page, il peut uploader une image puis choisir les tags correspondant à son image.

Les tags publics qui ont été créés préalablement possèdent un contour noir lorsqu'ils sont ajoutés à la publication. Cela veut dire que la publication sera visible par tout le monde dans ce tag.



Figure 1 : Tag publique chien

Si l'utilisateur crée son propre tag, il sera alors privé et celui-ci sera affiché avec un contour orange. Seules les personnes autorisées qui sont ajoutées lors de la création du tag privé pourront voir cette publication.

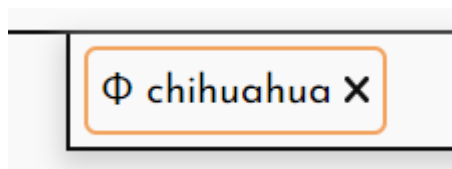


Figure 2 : Tag privé chihuahua

Une fois sa publication mise en ligne, il est possible de voter pour elle à travers notre système de jeu. En allant sur la page de jeu (à partir de l'icône manette du menu), on peut choisir un tag et le jeu commence alors.



Which one is best ?



*Figure 3 : Vote entre deux chiens*

Deux images sont tirées au hasard dans le tag sélectionné, et l'utilisateur doit alors choisir celle qu'il préfère parmi les deux. En la choisissant, cette image va recevoir +1 point alors que l'autre va perdre -1 point. C'est de cette façon que les publications sont votées, et le nombre de points de la publication est visible sur la page de détails de la publication.

Ce système va évidemment évoluer dans les prochaines mise à jour, avec l'ajout d'une limitation de temps pour pouvoir voter. Chaque semaine, les utilisateurs publieront dans les tags correspondants et la semaine suivante, leurs publications seront disponibles dans la partie jeu pour être votées. Lorsque la semaine est terminée, un classement sera alors mis en place selon le nombre de points, mettant ainsi en avant les publications ayant reçu le plus de points. Pour les trois premières publications, une récompense sera alors donnée à chaque utilisateur pour mettre en avant leur profil.

Nous souhaitons également gérer une certaine équité d'apparition entre chaque photo grâce à la valeur de "nbVotes" de chacune. À chaque fois qu'une image est votée, elle-même et celle présentée en face sont incrémentée de 1 sur leur "nbVotes". Ensuite, nous ferons en sorte que deux images ayant un "nbVotes" le plus bas soient affichées en même temps.

## II) Architecture

Les technologies utilisées pour notre application n'ont pas changé par rapport à ce que nous avons décidé dans le cahier des charges. Je vous laisse donc vous y diriger pour revoir nos spécifications.

L'architecture de la partie Front n'a pas changé non plus, utilisant toujours le `BrowserRouter` de `React router` pour changer facilement de page et en affichant le menu sur la plupart des pages.

L'architecture de la partie Back a quant à elle été quelque peu modifiée. En effet, nous avions au départ prévu d'avoir une base `MongoDB` en local sur chacun de nos postes pour faire nos tests. Mais on s'est vite rendu compte qu'il était beaucoup plus pratique pour nous d'avoir une base hébergée sur le cloud pour que les données soient partagées, ce que nous avons fait sur le site cloud de `MongoDB` alias `Mongo Atlas`.

Il a fallu également mettre en ligne la sauvegarde des images publiées. Pour cela, nous avons mis en place un `Cloud` qui héberge nos images et nous retourne le lien de l'image sauvegardée pour pouvoir la sauvegarder à son tour dans notre `BDD`.

## III) Wireframes

Si vous comparez nos wireframes (disponible en `.pdf` dans la partie docs du github) avec le projet actuel, vous verrez quelques petites différences.

En effet, la page permettant de choisir la photo parmi notre galerie de photo était une vue dirigée vers l'utilisation sur téléphone (avec `React Native` par exemple), nous avons donc simplifié cela en réalisant la vue comme sur un navigateur web.

La même chose est visible concernant la vue du jeu en mode paysage, cette partie n'a pas été réalisée puisqu'il s'agit d'une fonctionnalité disponible uniquement sur téléphone.

Comme nous l'avons dit à la fin de la partie une, le système de jeu n'est pas encore tout à fait complet, ce qui fait que la maquette concernant la recherche de publication par tag n'est pas tout à fait réalisée, puisque la partie gérant les publications par semaine n'est pas encore mis en place.

## IV) Organisation

Pour nous organiser nous avons mis en place dès le lancement du projet un Trello. Cela nous a permis de diviser les tâches entre nous et de suivre l'avancement global du développement de l'application.

Nous avons également trié les tâches en fonction de leur utilité (docs, archi, DB, ...).

Les tâches de développement étaient elles aussi réparties sous les catégories suivantes : DEV BACK, DEV FRONT, TO DO, DOING et DONE afin de nous permettre de bien nous organiser.

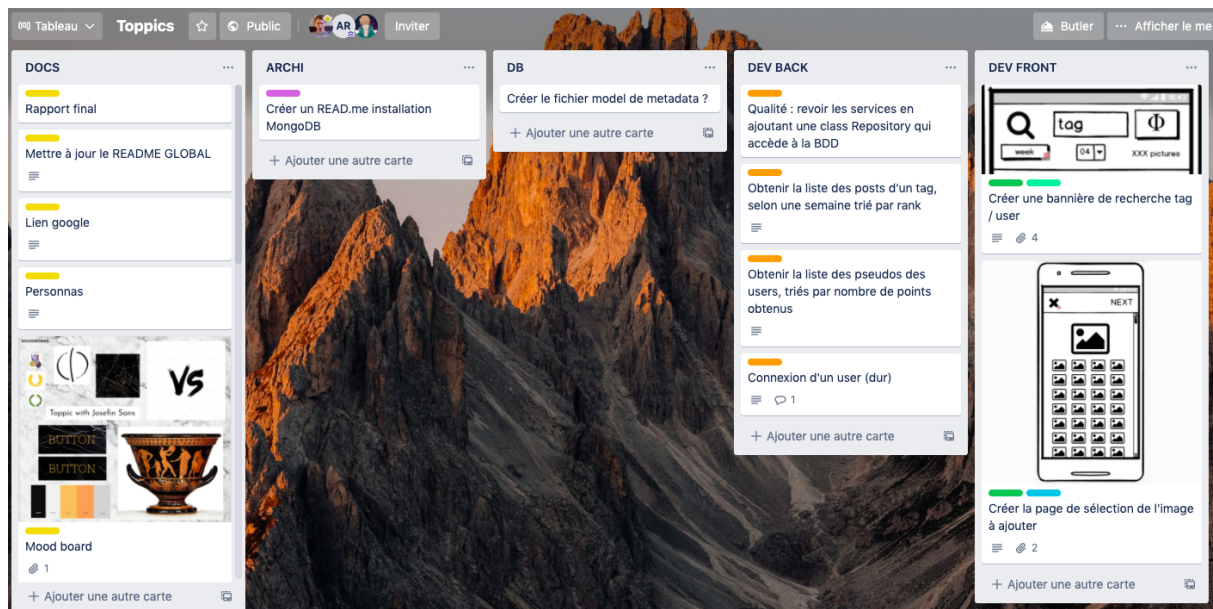


Figure 4 : Screenshot [Trello](#)