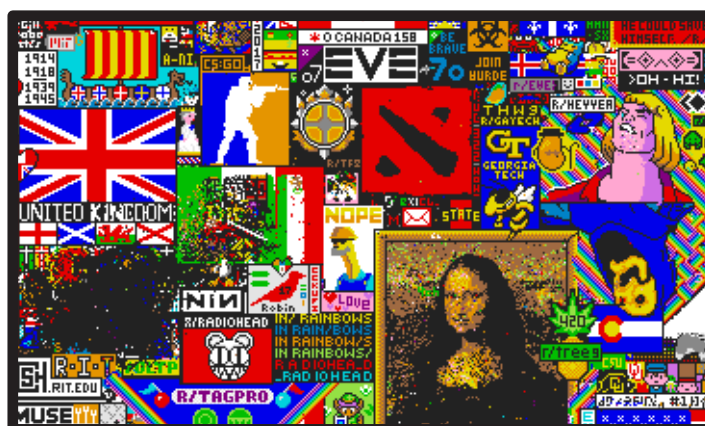


PRÉSENTATION

Le principe de Pyxelia était de reproduire r/place, un événement organisé par Reddit lors de leurs poissons d'avril. R/place était un tableau de plusieurs milliers de pixels sur lequel les utilisateurs pouvaient placer un pixel toutes les cinq minutes. Après trois jours, r/place s'arrêta et le tableau final est devenu une icône d'Internet.



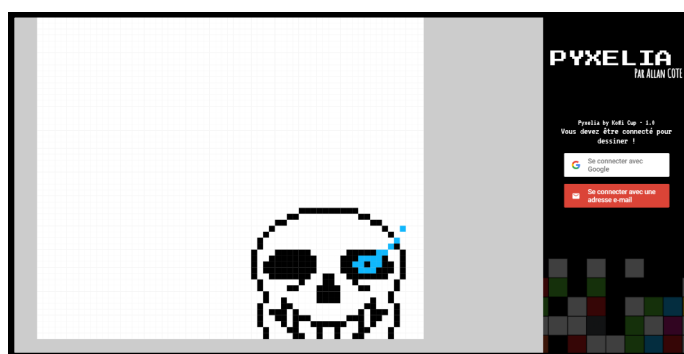
Un morceau du tableau final de r/place.

L'idée fut donc d'essayer de se rapprocher au maximum de r/place, malgré mes moindres connaissances. Après trois semaines de travail, le résultat est très satisfaisant et il fonctionne même en temps réel avec les autres joueurs !

Dirigeons-nous vers pyxelia.kofficup.fr pour découvrir comment l'utiliser !



La page d'accueil du site.



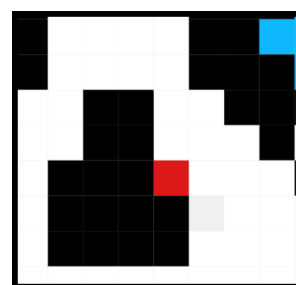
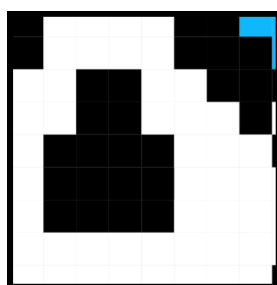
Pyxelia peu après la mise en ligne.

Rapidement, l'on découvre qu'on ne peut pas dessiner si l'on n'est pas connecté. On peut choisir de passer par un compte Google ou de créer un compte Pyxelia. Une fois connectée, la palette de couleur apparaît, l'utilisateur peut alors choisir une couleur et placer son pixel. Puisque nous venons de nous connecter pour la première fois, Pyxelia nous présente aussi les contrôles et le cooldown.

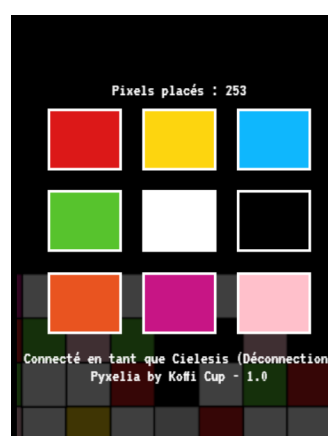


La page des contrôles

Nous sommes prêts à dessiner ! Nous pouvons donc placer un pixel toutes les 4 secondes et collaborer avec d'autres utilisateurs !



Après une petite présentation du site, Pyxelia nous emmène vers la page principale, la grille.



Pour finir, un menu latéral nous permet d'accéder aux quelques fonctionnalités supplémentaires pour enrichir notre expérience. L'un d'eux, « Stats » sera notre dernière présentation.

La page prend la forme très simple d'un classement des meilleures couleurs et des joueurs ayant placé le plus de pixels !

Vous savez quoi faire si vous voulez atteindre le top 10 !

Statistiques	
Dernière mise à jour le 24/01/2022, 11:27	
Classements des joueurs les plus actifs	Couleurs les plus utilisées
1 Cielesis - 255 pixels	1 207 pixels
2 Willem Préterre - 4 pixels	2 23 pixels
3 Baptiste LECORDIER - 0 pixels	3 20 pixels
	4 6 pixels

Et voilà ! C'était un petit tour de Pyxelia ! C'est maintenant aux artistes en herbes de créer un tableau plein de merveilleux dessins !

La suite de ce document présentera plus en détails les fonctionnalités, leurs fonctionnements et la raison de leurs existences.

FONCTIONNALITÉS

Notre présentation des fonctionnalités va être divisée en trois axes :

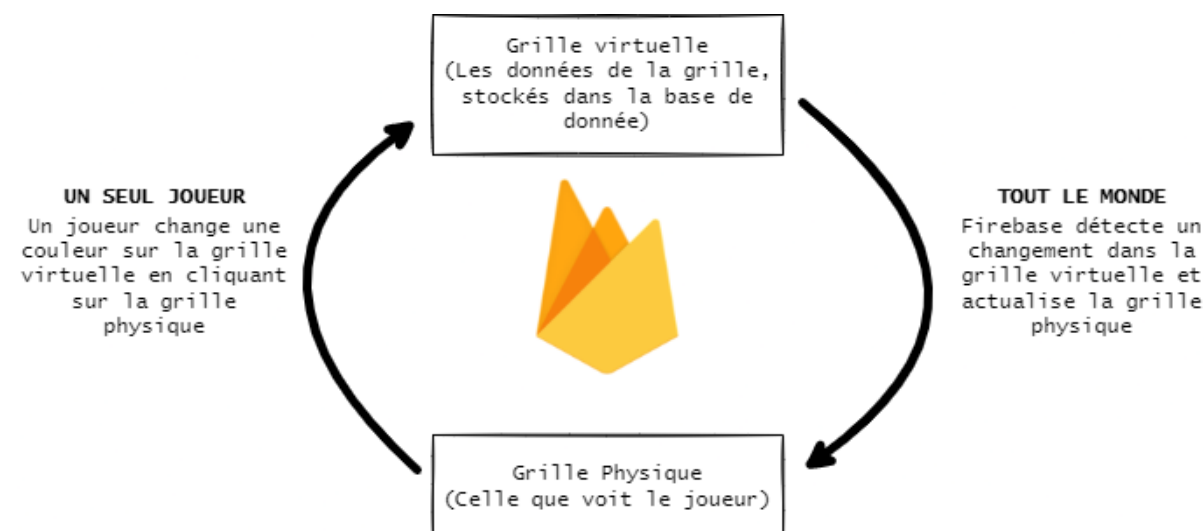
- Base de données et temps réel
- Connexion et statistiques
- Fonctionnalités annulées

Cela nous servira à guider la lecture et à naturellement emmener vers la partie « Évolution » en fin de document.

Base de données et temps réel

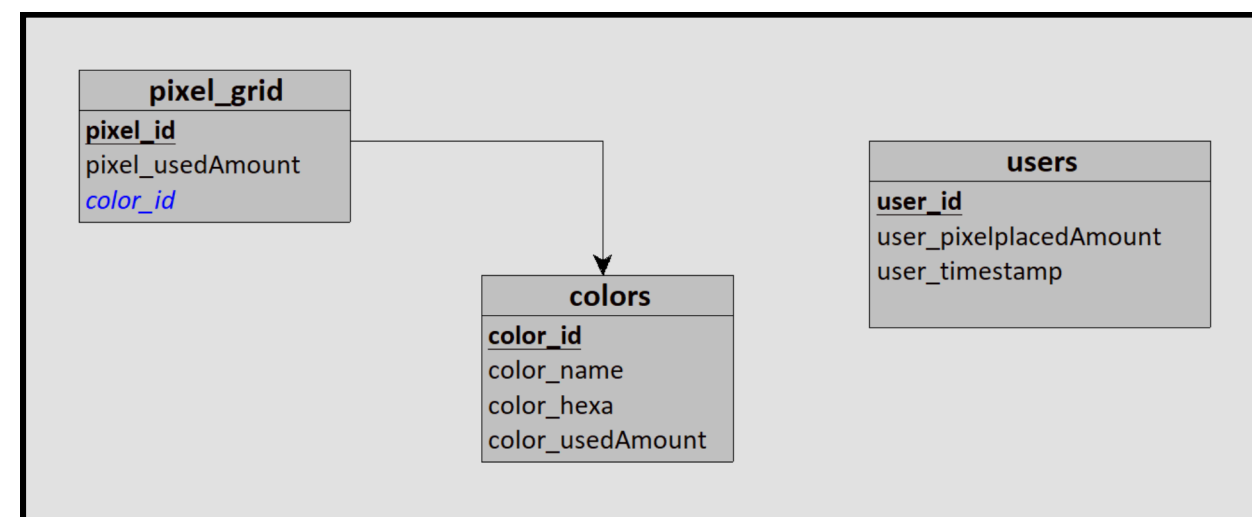
Très tôt, dans le développement de Pyxelia, il fut décidé d'utiliser la base de données Firebase, une plateforme permettant d'obtenir relativement facilement et gratuitement une base de données JSON fonctionnant en temps réel, le temps réel était l'objectif principal de Pyxelia, en effet, cela offrait une meilleure compréhension du tableau ainsi qu'un gameplay plus satisfaisant, notamment sur l'aspect collaboratif.

Firebase, donc, fonctionne avec un principe de déclencheurs, une modification dans un endroit désiré de la DB sera capable d'activer une fonction Javascript pour tous les utilisateurs, c'est le principe qui sera retenu pour Pyxelia.



La base de données stocke la grille virtuelle, mais aussi les comptes utilisateurs et les couleurs. Une partie des données est appelée en Javascript tandis qu'une autre est appelée en PHP.

Le PHP gère complètement les couleurs, en générant la feuille de style et la palette et il gère aussi entièrement la page de statistiques alors que le Javascript s'occupe de l'affichage du tableau et du système de connexion.



Une représentation visuelle de l'intérieur de la base de donnée.

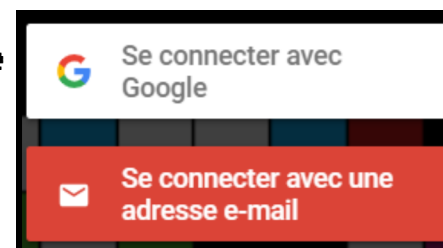
La sécurité est principalement gérée par Firebase ainsi que par un système de règles de CRUD, permettant de définir qui peut écrire, lire, mettre à jour ou supprimer des données, et ce, sur chaque morceau de la database. Un joueur ne pourra donc rien faire s'il n'est pas connecté, il aura l'autorisation de mettre à jour la grille virtuelle sinon.

Connexion et statistiques

Maintenant que le PHP génère la couleur et qu'un utilisateur connecté peut dessiner, ajoutons un moyen aux utilisateurs de se connecter justement !

Encore une fois, nous passerons par le large panel d'outils de Firebase et nous utiliserons Firebase Auth et Firebase UI. Le premier permet un système de connexion via différentes plateformes alors que le second offre une solution pour l'affichage des différents systèmes de connexions que nous allons utiliser.

Une fois installé et configuré, les connexions Google et par mail et mot de passe customisés sont disponibles.



Chaque couloir de connexion utilise Firebase UI pour créer des modales responsives et adapté à la méthode choisie.

Lors de la première connexion, un déclencheur qui ne s'active que lors de la création du compte va écrire dans la base de données le nom de l'utilisateur et un ID généré aléatoirement. Avec ça, plusieurs statistiques uniques seront créées, la première est un timestamp qui sera mise à jour chaque fois qu'un pixel est placé par l'utilisateur, servant donc à faire en sorte qu'il ne puisse placer un pixel que toutes les quatre secondes.

Une autre statistique sera la quantité de pixels placés, elle s'incrémentera dès que le joueur place un pixel. Cela incrémente au passage le nombre d'utilisation d'une couleur, valeur qui sera stockée dans la couleur en question.

En plus de tout ça, un bouton de déconnexion permet à l'utilisateur de se déconnecter manuellement, bien que Pyxelia a été configuré pour le faire automatiquement si le joueur ferme la page.

Ce sont les statistiques présentées juste avant qui sont ensuite utilisés par PHP pour fabriquer un top 10 des joueurs et le classement des couleurs.

L'INSTANT BOGUÉ

Un bug énervant empêchait le bon fonctionnement de Pyxelia après une connexion sur mobile. Ce bug, référencé nul part sur Internet, fut corrigé après plusieurs heures de tests, il s'agissait en fait d'un déclenchement prématuré de Firebase UI, avant même que Firebase ne soit lancé.

Lors des phases de tests, les joueurs subissaient des désynchronisations, ne voyant plus les mêmes choses sur leurs écrans. L'erreur ne venait en fait que d'une classe CSS qui n'avait rien à faire ici. "_(ツ)_/"

Un conflit entre deux fonctionnalités à rendu impossible le bon fonctionnement des deux simultanément, cela a causé l'annulation pure et simple et de l'une d'elles. Alors parlons-en.

Fonctionnalités annulées

Au début, le site avait une fonctionnalité très importante qui était prévue, le code avait même était pensé pour accueillir cette fonctionnalité, enfin, c'est ce que je croyais.

Je parle ici du système d'exportation en image et, grâce à ceci, de la création d'un système de timelapse, stocké en database via une fonction CRON, une fonction effectuée par la database de façon régulière.

L'objectif était de pouvoir, après quelques semaines, effectuer un timelapse de l'évolution de la grille, tout en laissant au passage la possibilité aux joueurs d'exporter le tableau en format image quand ils le souhaitaient !

Malheureusement, j'ai eu des soucis.

Le premier et le plus évident étaient que Firebase ne proposait pas CRON gratuitement, il fallait payer la version premium pour cela. J'ai donc cherché des alternatives, sans succès pour le moment.

Après ça, l'exportation en image. Bien que le programme fonctionne, il est très instable et augmente considérablement les téléchargements quotidiens permis par la version gratuite de Firebase, le problème de l'instabilité vient du fait que, exporter un SVG comme la grille en PNG n'est pas compliqué de base, mais ici, l'utilisateur a la possibilité de se déplacer et zoomer dans l'espace, une fonctionnalité primordiale pour le gameplay de Pyxelia.

Cette fonctionnalité déforme la grille pour la rendre déplaçable, créant dans l'exportation en image, un résultat dont 90% de l'image n'est que du vide avec 10% réservé à la grille, en petit dans un coin.

Alors que la date de rendu approchait, j'ai donc dû me contraindre à annuler les fonctions, bien qu'elles existent toujours, elles ne sont pas utilisables dans la version 1.0 de Pyxelia.



La seule photo réussie avec le système que j'ai développé.

Une autre fonctionnalité...

Il existe aussi une autre fonctionnalité cachée sur Pyxelia, conçu uniquement pour la blague, elle utilise plusieurs techniques visuelles et sonores et se déclenche avec un code secret mythique...

PS : activez le volume :D

ÉVOLUTIONS

Le projet a évolué de lui-même pendant le développement, avançant très vite, j'ai pu ajouter des nouvelles fonctionnalités avec aisance, j'ai toutefois plusieurs pistes d'évolutions.

La première serait bien évidemment de trouver comment faire fonctionner correctement le timelapse et l'exportation en PNG, cela semble être une tâche ardue mais pas irréalisable. Je pourrais par exemple sortir le SVG dans une autre page du site et faire le travail de transformation sur cette nouvelle page, isolant donc le code de son confrère problématique.

La seconde serait sur la quantité de couleurs et sur la taille de la grille. Actuellement, la grille fait 65*65 pixels, soit 4 225 pixels. La grille étant constituée d'un immense SVG et chaque pixel étant une balise <rect>, les performances ne sont déjà pas optimales et les tests ont montré qu'aller au delà pousse les limites de ce que Pyxelia peut supporter. L'idée serait donc d'ajouter une pagination sur les couleurs, quelque chose d'assez simple en soit, pour ajouter un nombre infini de couleur sans qu'elles ne poussent le reste du site. Pour la grille, c'est plus complexe. Peut-être faudrait-il revoir tout le système pour optimiser le tout, je redoute que ce soit ma conception qui me force à me limiter aux 4225 pixels déjà présents.

Finalement, l'ajout d'un compteur de joueurs en ligne et d'un timeout des joueurs, permettrait une meilleure régulation du flow des joueurs, en effet, Firebase ne pouvant accueillir que 100 joueurs simultanés, il est important d'indiquer cela aux joueurs, tout en forçant une déconnexion pour les joueurs restés trop longtemps connectés.

CONCLUSION

Pyxelia est un projet dont je suis très fier. Il est très avancé et présente plutôt bien ce dont je suis capable, que ce soit mon adaptation à un système inconnu comme Firebase, mon style graphique noir et blanc, mes connaissances en Javascript et en PHP et d'autres.

Il montre aussi mes faiblesses comme avec l'optimisation moindre de Pyxelia ou encore le conflit entre deux programmes pour l'exportation en image.

Merci d'avoir lu.

