# Date : 31 juillet 2023

## Description du problème ou de l’enjeu

Je veux fonctionner avec Material. Je dois donc faire mon design d’application au maximum avec des thèmes et informations material dans Figma. Sauf que Figma seul n’offre pas les composantes material de base, il me faut une autre manière de les importer.

## Description de la solution ou du contournement

Il se trouve que Material fournit une page Figma avec toutes leurs composantes utiles. Je n’ai pas pu les importer dans mon propre Figma, mais je peux faire des copier coller entre les 2 pages pour avoir ce que je veux.

# Date : 31 juillet 2023

## Description du problème ou de l’enjeu

J’ai besoin de pouvoir avoir les couleurs et thèmes de textes de mon Figma dans mon application flutter. Le faire à bras fonctionnerait, mais prendrait énormément de temps. J’aimerais donc trouver une solution qui me prendrait moins de temps pour importer les thèmes sombres et clairs de mon Figma dans mon Flutter.

## Description de la solution ou du contournement

En cherchant, j’ai découvert que depuis quelques mois, Figma offre maintenant la possibilité d’exporter son thème en Flutter. Cela donne un fichier contenant une classe de thème avec toutes les couleurs et tous les thèmes de textes de mon Figma directement. Ça va me demander du reformatage, mais pas une tâche vraiment compliquée.

C’est la meilleure solution que je pouvais espérer. Ça va me sauver énormément de temps de retranscription et faciliter ma gestion des thèmes.

# Date : 1er août 2023

## Description du problème ou de l’enjeu

La compilation de mon app en version android ne fonctionne pas. J’ai des erreurs diverses comme quoi la version du gradle ne serait pas suffisante. Pourtant, j’utilise les paramètres par défaut de Flutter, ça devrait donc marcher.

## Description de la solution ou du contournement

Déjà de 1, j’avais pré-importé un package pour la connexion avec Firebase. Ce package demande une version de gradle supérieure à celle de Flutter par défaut. J’ai donc augmenté à la version 25 pour pouvoir supporter le package.

# Date : 1er août 2023

## Description du problème ou de l’enjeu

J’ai une erreur de compilation Android qui me dit qu’il manque le support « multidex ». Pourtant, encore une fois, j’ai les paramètres par défaut de Flutter pour une app mobile et rien de spécial, ça devrait donc marcher.

## Description de la solution ou du contournement

Les paramètres par défaut de flutter pour Android ont des options manquantes, du moins pour la version de Flutter. Il m’a donc fallu ajouté des permissions pour Firebase dans le fichier « build.gradle » ainsi qu’activer le fameux support « multidex » dans le même fichier. Maintenant la compilation pour Android fonctionne !

# Date : 2 août 2023

## Description du problème ou de l’enjeu

Je dois entrer toutes les configurations Firebase pour mon app flutter. Ces configurations sont longues et complexes, et il y en a une par plateforme supportée. J’aimerais trouver une manière de ne pas y passer 6h à faire de la retranscription.

## Description de la solution ou du contournement

En fouillant la documentation de Firebase, j’ai trouvé l’outil Flutterfire. Cet outil console se connecte directement à mon compte Firebase et écrit tout seul les configurations nécessaire dans mon projet Flutter.

Par contre, il est extrêmement dur à configurer en Windows. J’ai dû d’abord installer la dernière version de Node sur ma machine pour avoir accès à npm.

Ensuite, j’ai dû installer les firebase-tools à partir de npm.

Ensuite, j’ai dû m’arranger pour rendre les firebase=tools disponibles en console car l’installation n’a pas été capable de le faire elle-même. J’ai donc cherché le fichier source et ajouté son chemin dans les variables d’environnement.

Ensuite, j’ai dû installer le firebase-cli, un fichier qui exécute une console Firebase. J’ai tout essayé pour le rendre disponible en console, mais rien ne fonctionnait, j’ai donc exécuté le fichier directement à la racine du projet en droits d’administrateur.

Finalement, j’ai dû installer flutterfire en global à l’aide de la console dart.

Une fois toutes ces étapes franchies, j’ai finalement pu faire fonctionner ma configuration automatique. Ça a été long mais au moins je suis sûr qu’il n’y a pas d’erreur.

# Date : 2 août 2022

## Description du problème ou de l’enjeu

J’ai en théorie réussi à me connecter à la base de données Firebase. Les requêtes get oint l’air de fonctionner sans erreur. Par contre, je n’arrive absolument pas à ajouter des données à la base de données, donc je n’ai aucune confirmation de si ça fonctionne ou pas.

## Description de la solution ou du contournement

Quand on crée une base de données pour une application dans la console Firebase, les droits données à tous les utilisateurs sont par défaut à lecture seulement. C’est pourquoi je ne pouvais pas écrire dans la base de données.

Ça a pris du temps à fouiller dans les paramètres de Firebase pour trouver ça. Sachant que j’ai d’abord supposé que le problème était de mon côté et testé plusieurs possibilités avant.

# Date : 3 août 2023

## Description du problème ou de l’enjeu

J’ai une option dans ma page de profil qui me permet de changer le nom de l’utilisateur s’il le veut. Ça l’envoie à la base de données, il n’y a aucun problème là-dessus, mais mon widget principal dans lequel il y a toute ma navigation ne se met pas à jour lui. Et je n’ai pas de moyen de l’atteindre à partir du widget de la page de profil, car même en méthode statique je ne pourrais pas appeler le setstate (méthode de mise à jour d’un widget dans flutter.

## Description de la solution ou du contournement

En cherchant, j’ai trouvé un package sur le site [www.pub.dev](http://www.pub.dev) qui s’appelle Eventify. Ce package permet de communiquer entre plusieurs widgets séparés à l’aide d’un event. Ça me permet de faire une communication entre mon widget de navigation et ma page.

# Date : 3 août 2023

## Description du problème ou de l’enjeu

Je dois être capable de changer le thème selon une sélection de boutons dans la page de profil (thème sombre, clair ou système). Malheureusement, Flutter ne me donne pas d’option pour changer directement le thème au runtime. Il est définit dans le fichier main.dart dans le MaterialApp (App principale) au moment de démarrer l’application.

## Description de la solution ou du contournement

J’ai changé le widget principal qui contient la MaterialApp dans le fichier main.dart pour un widget Stateful, c’est-à-dire un widget dynamique qui peut être modifié au runtime avec une commande setState. Ensuite, je me suis créé une enum BrightnessSelection pour pouvoir faire efficacement la sélection entre mes 3 modes.

Je voulais garder en mémoire sur l’appareil et non pas pour l’utilisateur, pour donner une expérience personnalisée à l’appareil utilisé. J’ai donc utilisé le package Shared Preferences de flutter qui peut stocker des informations en local sur l’appareil ou le navigateur. J’ai créé un LocalStorageService pour permettre de lire et stocker des informations en local.

J’ai ensuite créé le BrightnessService, qui fait simplement des appels au LocalStorageService pour aller chercher la valeur de thème stocké en local.

Pour finir, quand je clique sur un bouton de sélection de thème, j’envoie un signal avec le même package Eventify que pour le nom du profil au widget principal dans main.dart. Ce signal fait rafraîchir au complet l’application. J’ai aussi changé le thème constant pour que l’app aille le chercher par le BrightnessService. De cette manière, quand j’appuie sur le bouton, le thème est changé dans la mémoire interne, puis le signal est envoyé au main, qui rafraîchit toute l’application avec le nouveau thème choisi.

Cette solution a pris plus de temps à mettre en place que je pensais, mais cela permettra à long terme d’avoir déjà les préférences utilisateurs dans le système et d’ajouter une mémoire cache si jamais il y a des chargements trop longs.

# Date : 7 août 2023

## Description du problème ou de l’enjeu

Je veux faire afficher mes budgets, projets et modèles dans des carrousels pour que ce soit intéressant visuellement. Je connais un package en dart qui normalement fonctionnait assez bien, mais il refuse de fonctionner dans mon projet peu importe ce que je fais. En regardant dans les logs d’erreur, je peux voir des logs de « dimension double.infinite » ou des trucs ressemblants.

## Description de la solution ou du contournement

Il se trouve qu’en cherchant dans le code source, on s’aperçoit que si on ne fournit pas une hauteur au carrousel dans le code, celui-ci fonctionne avec un Expanded, soit un widget fait pour prendre toute la place disponible. Sauf que j’ai rendu la page sur laquelle sont les carrousels scrollable, donc avec une dimension en hauteur théoriquement infinie. Ce qui faisait que le carrousel se retrouvait avec une hauteur infinie et donc n’affichait pas dû à l’erreur. J’ai simplement ajouté une hauteur au carrousel pour régler le problème.

# Date : 7 août 2023

## Description du problème ou de l’enjeu

Quand je navigue d’une page à l’autre, soit de la page d’accueil au profil, j’ai toujours une erreur qui s’affiche en base de l’écran. Cela m’indique qu’une « Global Key » a été utilisée en double et compromet l’application. Cela ne semble pas affecter l’app en développement, mais j’aimerais mieux me débarrasser de cette erreur vu que j’ai éventuellement l’intention de compiler l’app en production.

## Description de la solution ou du contournement

Selon les forums que j’ai lu sur le sujet, les widgets dans Flutter posséderaient une clé. Celle-ci fait en quelque sorte office d’Id dans l’app et permet au flutter de générer plus efficacement les graphiques. Quand on ne leur donne pas de clé, celle-ci est générée automatiquement, il n’y a donc normalement pas d’erreur possible.

Sauf que mon erreur se faisait non pas sur un widget, mais sur ma navigation. J’utilise un package nommé Beamer pour faire la navigation entre mes pages, et j’ai dans mon cas plusieurs « navigateurs » : un principal pour rediriger aux autres, un pour la connexion et un pour l’app en tant que tel. Cela me permet d’ajouter des sécurités plus efficacement sur des sections de l’app tout en ayant des « layouts » prédéfinis pour chaque section.

Par contre, il semblerait que Flutter ne génère pas automatiquement des clés différentes pour les navigateurs de ce package. Ce qui faisait que mon navigateur principal et mon sous-navigateur se retrouvaient avec la même clé en étant tout les deux présents dans l’app, d’où l’erreur.

Pour la corriger, j’ai défini des clés manuellement aux navigateurs avec une clé unique en String pour chacun. Cela permet à Flutter de les identifier séparément et donc d’éviter l’erreur.

# Date : 8 août 2023

## Description du problème ou de l’enjeu

La base de données Firebase est une base de données sans en être une. C’est plutôt un gros document en JSON dans lequel je peux naviguer et aller chercher des données. Cela le rend extrêmement rapide, mais m’empêche de faire des requêtes complexes avec des paramètres ou des liaisons entre tables comme je pourrais le faire sur d’autres bases de données.

Sauf que pour les besoins de l’app, j’ai besoin que des projets puissent être liés à des budgets et des modèles de budget pour savoir quand je met de l’argent sur un projet et combien j’en met. Sauf que je ne peux pas stocker ça dans des tables pivots ou autres car je ne peux pas faire directement des liaisons. Du coup la question est : comment je fais ça ?

## Description de la solution ou du contournement

J’ai décidé de fonctionner un peu sur le même principe que les MoneyEntry que j’ai créé pour les revenus et dépenses d’un budget et d’un modèle de budget. J’ai créé des ProjectEntry, qui contiennent l’id du projet et le montant mis dedans. Les seules données stockées dans la base de données sont donc l’id du projet et les montants mis sur l’entrée. Cette entrée est stockée directement dans un budget ou un modèle.

Ensuite, pour aller chercher les informations manquantes, comme la date d’entrée et le projet lui-même, je vais chercher les budgets et les projets dans la base de données pour populer les entrées directement dans l’app par la suite. Ça me rajoute un appel à la base de données par requête, mais étant donné que j’ai peu de données et que Firebase est rapide, ça ne se voit pas dans l’app.

De plus, ça me permet de stocker moins de données dans la base de données et donc d’économiser mon stockage vu que j’ai un format gratuit du service Firebase.

# Date : 8 août 2023

## Description du problème ou de l’enjeu

Depuis que j’ai ajouté la gestion des entrées de projet dans l’app, mes communications avec la base de données semblent ne plus fonctionner. Changer le montant dans le compte fonctionne, mais toutes les requêtes pour les modèles, les budgets et les projets me donnent des chargements infinis. Ça refonctionne si j’enlève les entrées de projet, mais j’ai besoin de ces entrées dans mon application.

## Description de la solution ou du contournement

Dans mon application, j’ai 2 méthodes pour aller chercher mes budgets. Une pour aller chercher les anciens budgets, et une pour aller chercher les nouveaux. Ces 2 méthodes appellent une méthode de base getAll, car je ne peux pas faire de requête avec des paramètres avec Firebase. Je fais donc un getAll, puis je filtre la liste directement dans l’app selon ce que je veux.

Dans toutes ces méthodes, il y a la méthode populateProjectEntries qui est appelée pour aller chercher les données des entrées projet. Sauf que vu que je l’appelle déjà dans mes 2 méthodes séparées, l’appeler dans la méthode getAll est un appel inutile. Par contre, cela ne devrait en théorie pas causer de bug.

Je ne sais pas pourquoi, mais le fait de l’appeler dans la méthode principale et dans les 2 secondaires a créé une sorte de boucle d’appel à la base de données. Vu que ce sont les budgets qui sont chargés en premier dans l’app, la base de données se retrouvait bloquée et incapable de passer tout autre appel que les budgets, d’où le fait que les modèles et projets ne fonctionnaient pas. Le changement du montant au compte fonctionnait car il est chargé avant les budgets, donc avant la boucle.

Le problème s’est réglé simplement en enlevant la ligne fautive du code, mais m’a pris beaucoup de temps à trouver.

# Date : 9 août 2023

## Description du problème ou de l’enjeu

Quand j’ajoute une ligne dans les revenus ou les dépenses et qu’il y a déjà une ligne d’entrée, la nouvelle ligne prend les valeurs de l’ancienne sans que je fasse quoi que ce soit pour lui donner. Quand je regarde dans le débogueur avec des points d’arrêts, les valeurs de la nouvelle ligne sont vides, comme supposé. Pourtant, l’app affiche les valeurs de l’ancienne ligne malgré tout. La validation des formulaires et tout ce qui y est lié semble fonctionner selon l’affichage et non la valeur dans le code.

## Description de la solution ou du contournement

Comme je l’ai dit précédemment, les widgets dans Flutter ont une clé qui leur sert d’Id. Cette clé est normalement générée automatiquement par Flutter lorsque non fournie, ce qui normalement empêche l’app d’avoir des widgets identifiés pareil dans une même page.

Eh bien il semblerait que l’attribution automatique de clés dans Flutter fonctionne plus ou moins bien dans certaines exceptions. Quand j’ajoute un widget de nature identique (même classe) dans une liste sur une page, la clé automatiquement générée par flutter semble référer à la classe du widget et non au widget en tant que tel. Ce qui fait que je me retrouvais avec 2 widgets pas pareils mais ayant le même Id, et donc flutter les affiche de la même sans rafraîchir l’affichage.

Pour contrer le problème, j’ai donc encore une fois ajouté une clé moi-même. J’ai fabriqué cette clé avec le hashcode du budget lié (valeur unique) et les millisecondsSinceEpoch de la valeur DateTime.now (change à chaque rafraichissement de page). De cette manière, vu que l’Id du widget est toujours différent, je force Flutter à toujours le regénérer et donc à ne plus me faire des affichages bizarre comme il m’a fait.

Pour le coup, je travaille souvent avec Flutter, mais celle-là je ne la connaissais pas. Ça va énormément me servir dans le futur.

# Date : 9 août 2023

## Description du problème ou de l’enjeu

Je me suis fait un logo pour mon app avec une IA sur internet. J’aimerais bien qu’il s’affiche dans l’app, mais surtout changer les logos de l’app compilée. Donc le favicon en web et l’icône de l’app en mobile.

## Description de la solution ou du contournement

Ici le problème se sépare en 2.

Déjà, pour le favicon en web. Il m’a fallu générer plusieurs dimensions de mon logo avec un site fait pour ça pour avoir les différentes dimensions android, apple, etc. Ensuite, il m’a fallu mettre ces logos dans l’app et les référer de la bonne manière dans le fichier index.html de mon projet Flutter (dans le dossier web, là où Flutter génère sa version compilée web). En faisant ça, mon app s’est mise à afficher le bon favicon en web.

Pour le mobile, il m’a fallu utiliser le package flutter\_launcher\_icons. Ce package me permet de faire rouler une commande de terminal qui génère les logos nécessaires à la version mobile selon l’image que je lui fournis.