

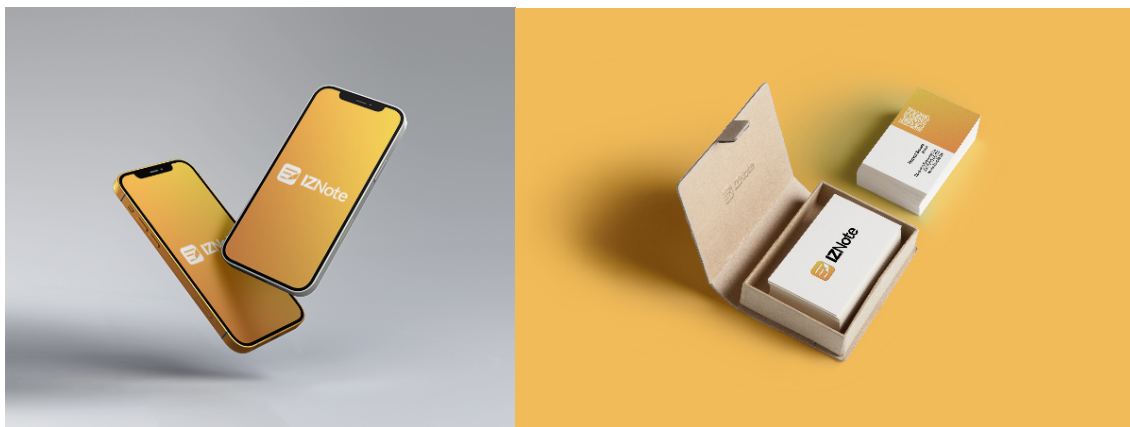
# Atelier de développement mobile

## Introduction

Dans le cadre de notre cours d'atelier de développement mobile avec M. Ernotte, il nous est demandé de réaliser une application mobile de note ergonomique. Nous avons décidé de nommer notre app : IZNOTE (easy note en anglais) car notre application est facile à utiliser.

## Charte graphique et logo

Nous avons fait la charte graphique et avons réalisé le logo de notre app. Nous avons choisi des couleurs vives et colorées qui se différencient des autres applications de notes que nous pouvons voir sur le store. De plus, nous avons intégré un stylo de type minimaliste dans le logo afin distinguer rapidement l'application.



## CSS

Le CSS de notre application se base bien évidemment sur l'identité visuelle du logo et donc sur sa palette de couleur. La page d'accueil se compose d'une en-tête qui reprend notre logo de base pour que celui-ci puisse parfaitement rentrer en harmonie avec les couleurs des divers contenus présents dans la page en question. En effet, le fond reprend la couleur de base du logo mais de manière plus subtile afin que les divers champs et logos puissent ressortir d'avantage.

Les divers champs et notes déjà présentes dans le « localStorage » sont encadrés par des bordures grises afin de rentrer en contraste avec la couleur de base.

Les divers boutons, quant à eux, sont constitués d'un léger dégradé de couleur et d'une couleur de texte « gris foncé » afin d'annoncer clairement les principales fonctionnalités. Toutes ces indications de couleurs sont utiles afin de faciliter la navigation de l'utilisateur au travers de l'application, mais également, pour créer quelque chose de cohérent et agréable au possible.

Lorsqu'une note est sélectionnée, l'utilisateur peut alors aisément la modifier grâce à l'implémentation d'un « text editor » qui lui permet alors de personnaliser son texte selon ses désirs. Il peut alors mettre son texte en gras, en italique, en souligné et peut même décider de le « barrer » si une tâche n'est plus d'actualité ( si la note représente une « to do list » par exemple ).

La couleur de texte du bouton « Effacer » est mise en rouge pour lui indiquer plus explicitement son signifiant de même que pour le bouton « Retour » mis en bleu et représenté par une flèche.

## Faire fonctionner le bouton supprimer

- Fonction deleteNote dans le js.
- Gestionnaire de l'évènement click sur le bouton Effacer de la page de détail dans le js.

## Permettre la recherche aisée d'une note avec une search bar

- Ajout de la Search Bar dans le index.HTML avec l'aide de la Documentation.
- Ajout d'une ligne « note introuvable » dans le HTML au cas où la recherche n'aboutit à rien.
- Ajout d'un chemin d'accès dans le index.js grâce à la documentation.

## Une suppression plus facile avec swipeout

- Ajout du swipeout en l'intégrant dans le template et ajout du swipeout-right avec data-confirm dans l'html.
- Gestionnaire du swipeout dans le js.

## Une confirmation de suppression

Nous avons décidé de faire la confirmation de suppression seulement pour le swipeout et non dans la page de détail pour ne pas que cela soit trop lourd pour l'utilisateur.

- Ajout du Dialog dans le gestionnaire du swipeout.

## Permettre la modification d'une note

- Lorsqu'une note a été ajoutée, nous pouvons accéder à une zone de modification via la page details.html. L'utilisateur récupère les données dans un nouveau formulaire.
- Il peut décider de changer les données du titre tout seul, du mémo tout seul ou les deux à la fois. Quand on clique sur le bouton « modifier », l'ancienne note est supprimée via un champ caché qui contient le titre de l'ancienne note. Les données des deux champs input sont sauvegardées dans une nouvelle clé du localStorage.

## Gérer des catégories

- Mise en place d'une catégorie assez compliquée à réaliser.
- Test effectué sur deux classes de type list-group, avec un list-group-title pour chaque catégorie. Dans le formulaire d'ajout de note, nous avons intégré un formulaire toggle, avec un champ qui permettrait à l'utilisateur de cocher ou pas le champ « urgent ».

## Une modification de la méthode d'ajout de notes

- Création d'un floating action bouton.
- Création d'une page ajout.html, liée au floating action bouton.
- La page principale montre ainsi la liste des notes, pour ajouter une nouvelle note, l'utilisateur est renvoyé sur une nouvelle page reprenant le formulaire d'ajout de note.

## Test sur nos machines

Nous avons été lire la documentation de Cordova et nous avons testé l'application sur un de nos smartphone.

- Installation du SDK sur la machine.
- Installation du kit de développement Java.
- Configuration d'un émulateur.
- Création du projet.
- Test sur la machine.

## GitHub

Voici le lien de notre projet sur GitHub : <https://github.com/francoislaflleur/atelierdevmobile>