

Reciplease



PROJET 10

Document traitant des fonctionnalités présentes dans l'application iOS Reciplease.

Mickaël HORN

Étudiant sur le parcours « Développeur d'application iOS » - OpenClassrooms

FONCTIONNALITÉS

Table of Contents

Contexte	2
Ajout des ingrédients.....	3
Recherche des recettes.....	5
Détail de la recette	6
Liste des favoris	8
Détail du favori.....	9
Get Directions.....	10
Settings	11
VoiceOver	11

Contexte

Reciplease est une application iOS, développée en Swift permettant de trouver des recettes à partir d'une liste d'ingrédient préalablement établie.

Les fonctionnalités sont réparties grâce à une UITabBar contenant 3 onglets : « Search », « Favorites » et « Settings ».

A la suite des ingrédients saisis par l'utilisateur, il a alors accès à une liste de recettes, avec pour chacune d'entre elles, l'image associée, son titre, le temps de préparation ainsi que la liste des ingrédients principaux.

Lorsque l'utilisateur a choisi une recette, il appuie dessus, ce qui le mène à une autre interface, celle du détail de la recette.

Ici, une plus grande image est affichée, toujours accompagnée du titre de la recette. Cependant, il a maintenant accès aux ingrédients détaillés de cette recette.

Il peut également s'il le désire, appuyer sur le bouton « Get Directions », ce qui le mènera vers la page web de la recette contenant les instructions pour la réaliser.

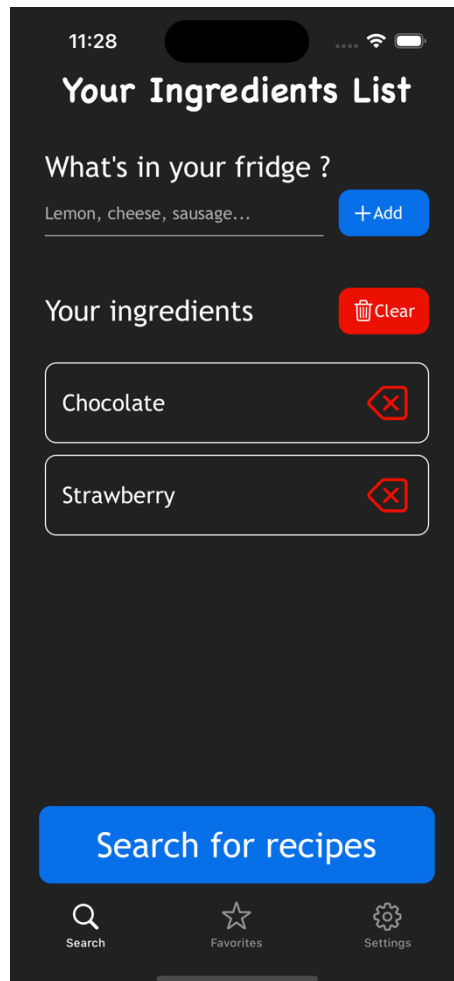
Pour finir, un bouton en forme d'étoile, schématisant les favoris est présent dans le coins supérieur droit de l'écran.

Une pression permet d'ajouter en favoris la recette, que l'utilisateur pourra retrouver dans l'onglet « Favorites ».

La liste de ses favoris s'affiche alors, ou il pourra, tout comme quand il recherche des recettes, appuyer sur l'une d'entre elles, et afficher ses détails ou consulter la page web associée.

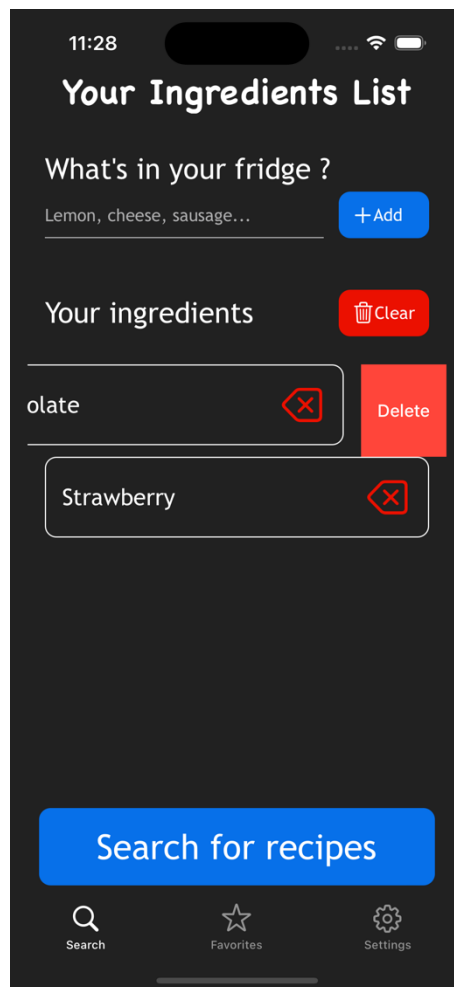
Ajout des ingrédients

La première étape consiste à insérer les ingrédients de notre choix.
Pour ce faire, une zone de texte nous est fournie, accompagnée de son bouton d'ajout.



On constate que deux ingrédients ont été ajoutés, « Chocolate » ainsi que « Strawberry ». Le bouton « Clear » permet de vider intégralement la liste.

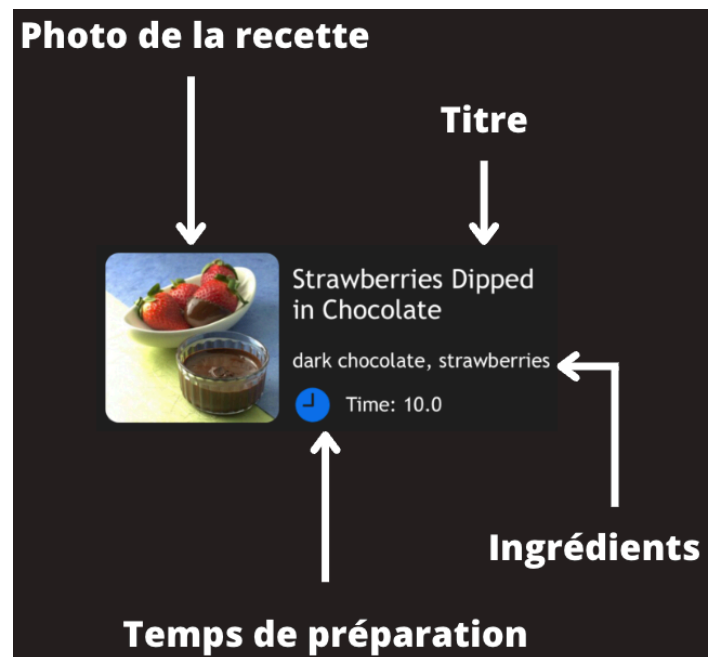
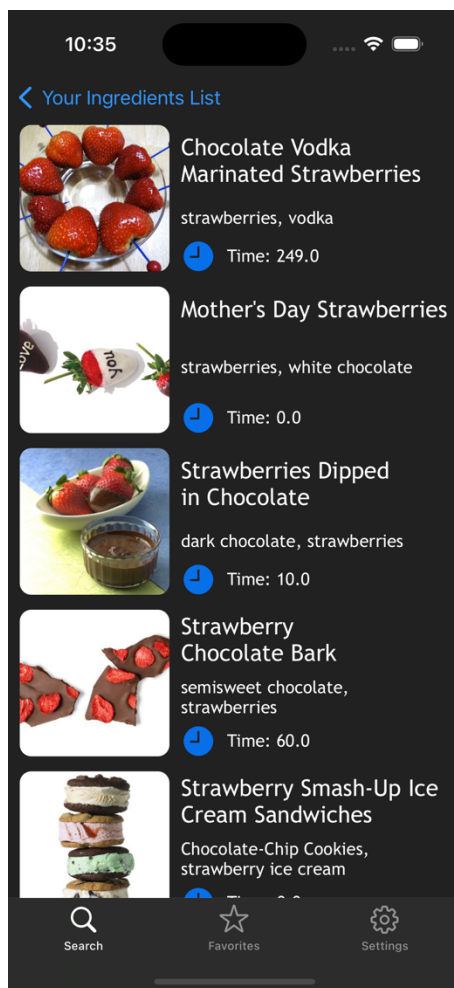
Une flèche pointant vers la gauche indique à l'utilisateur qu'il peut swiper dans cette direction, afin de supprimer un des ingrédients.



Une fois que les ingrédients ont été saisis, nous pouvons passer à la deuxième étape, la recherche des recettes !

Recherche des recettes

L'écran suivant présente le résultat de la recherche.



Si le nombre de recette excède le nombre présenté sur l'écran, l'utilisateur peut alors scroller vers le bas afin d'afficher le reste des recettes.

Une fois son choix fait, l'utilisateur sélectionne une recette, ce qui nous conduit à l'étape suivante.

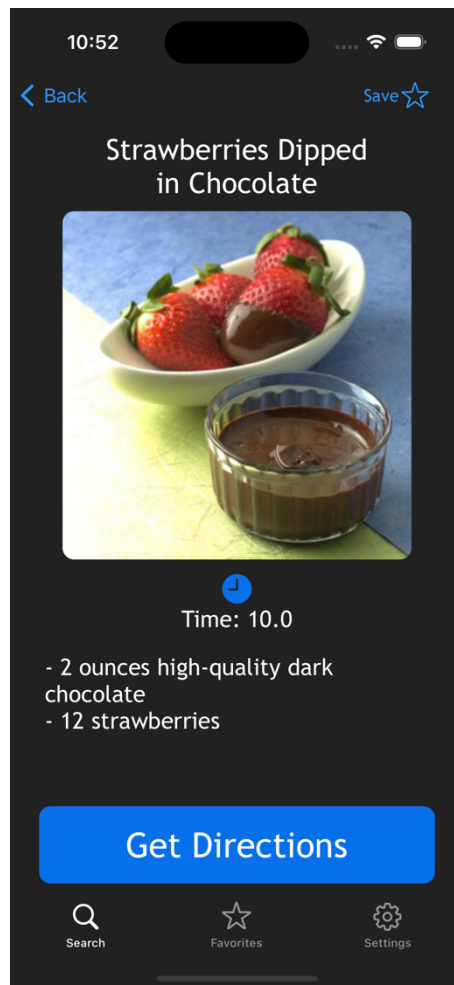
Note :

Pour mes appels API, j'utilise le framework Alamofire.

Les images sont également stockées en cache, afin de ne pas avoir à les retélécharger à chaque fois.

Pour cela, j'utilise le framework SDWebImage.

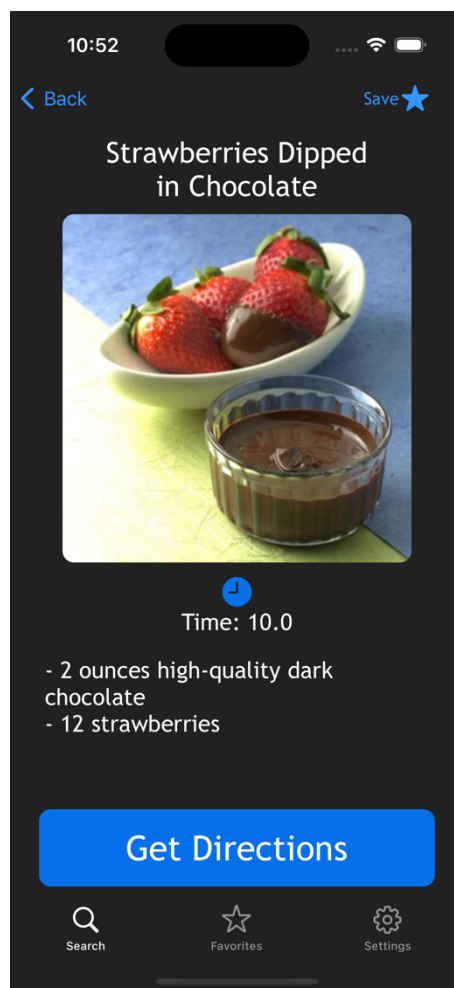
Détail de la recette



Dans cet écran, on a à notre disposition une image plus grande de la recette que l'on a choisie, une liste détaillée des ingrédients et un bouton « Get Directions » pour consulter les instructions détaillées.

Dernier point ici, le bouton des favoris, présent en haut à droite sous forme d'étoile.

Quand on appuie dessus, plusieurs choses se passent :



Tout d'abord, le bouton change d'apparence, passant d'une étoile vide à une étoile remplie. C'est un indicateur visuel permettant de savoir si la recette est en favoris ou non.

Bien entendu, si on ne veut plus de la recette, on a juste à appuyer de nouveau sur le bouton, vidant ainsi l'étoile et retirant la recette des favoris.

Note :

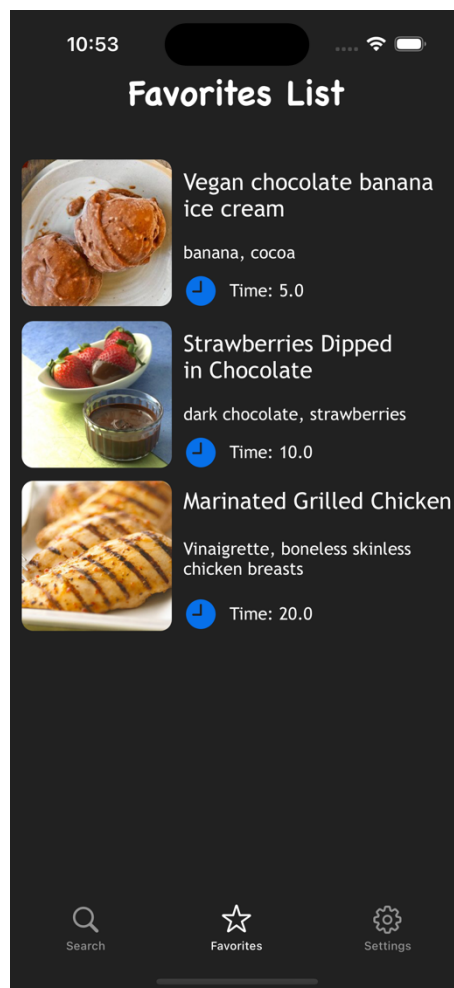
Afin de sauvegarder les recettes, j'utilise la technologie CoreData, fournie par Apple. Je dispose de deux entités, Recipe et Ingredient.

La relation « ingredients » lie la table Recipe à Ingredient, et la relation « recipe », quant à elle, lie Ingredient à Recipe.

Le mode de suppression est réglé sur « Cascade », afin que si on supprime une recette, ses ingrédients sont également supprimés.

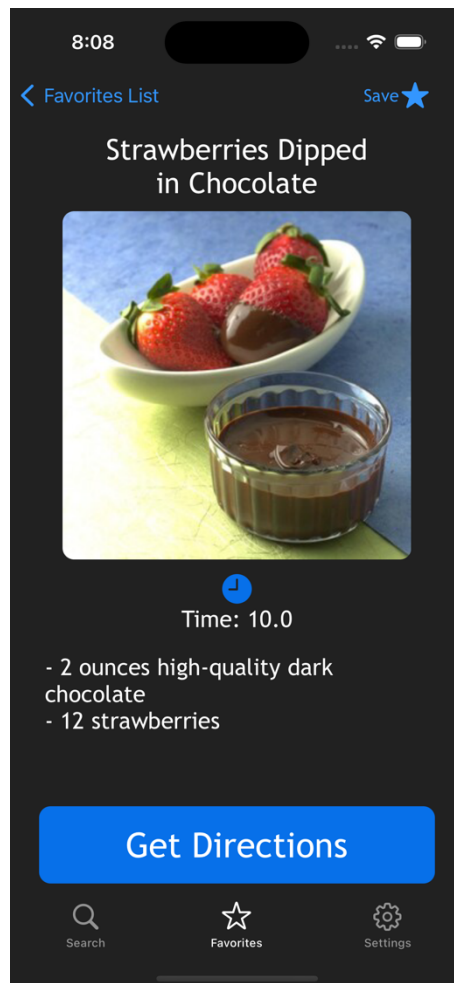
Liste des favoris

Il regroupe l'ensemble des recettes que l'utilisateur a décidé d'ajouter à ses favoris.



On retrouve la même interface que pendant la présentation du résultat des recettes. De la même manière, il suffit d'appuyer sur l'une des recettes afin d'accéder à l'écran de détails.

Détail du favori



Cet écran doit également vous être familier.

Il s'agit de la même interface que celle du détail d'une recette, mais pour nos favoris.

Une différence mineure subsiste.

Quand on arrive sur cette page, il est naturel que l'on retrouve l'étoile remplie, car après tout, cette recette est en favoris !

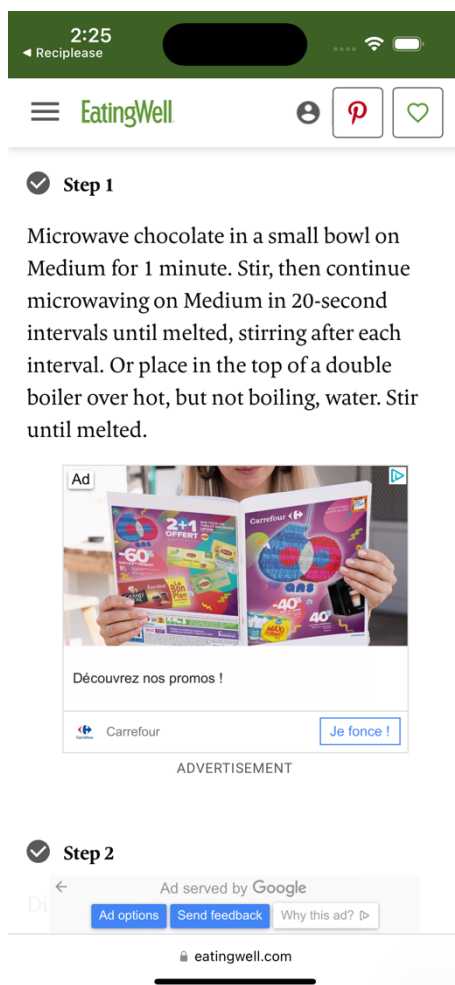
Get Directions

La faim commence à se faire ressentir...

Nous avons les ingrédients, c'est bien.

Les ingrédients détaillés, encore mieux, mais nous voulons les instructions claires et précises pour réaliser notre recette !

En appuyant sur le bouton « Get Directions », nous allons pouvoir avoir accès à la liste des instructions de réalisation de notre recette, comme ceci :



Settings

Dernier onglet de notre UITabBar, les paramètres de l'application.

Pour l'instant, on ne dispose que d'un UISegmentedControl, permettant de changer de mode parmi le mode clair et le mode sombre.

J'ai pris la décision d'ajouter cette fonctionnalité pour deux raisons principales.

La première, dans le but de proposer à l'utilisateur une autre interface s'il n'aime pas celle qui lui est présentée.

La deuxième, c'est que le mode sombre est important pour le green code.

En effet, cela permet d'économiser la batterie sur les écrans de type AMOLED.

VoiceOver

Un des enjeux de ce projet était également la mise en place de l'accessibilité pour les personnes en ayant besoin.

Une de mes implémentations a consisté à mettre en place VoiceOver, l'assistant vocal d'Apple permettant aux utilisateur aveugles ou malvoyants de mieux naviguer sur l'application.

Il nous faut configurer, si besoin, les accessibilityLabel, accessibilityValue et les accessibilityHint pour les éléments de l'interface, tels que les UILabel, UIButton, etc.

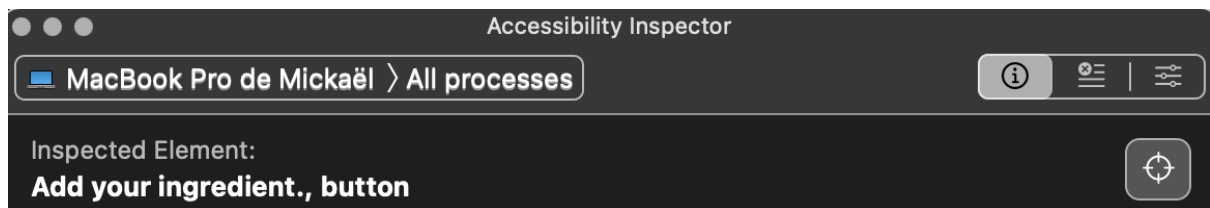
- accessibilityLabel : ajoute une description détaillée de l'élément graphique.
- accessibilityValue : annonce la valeur que contient l'élément. Par exemple, pour un UITextField, on pourrait annoncer le place holder.
- accessibilityHint : quel effet va produire l'appui sur un bouton par exemple.

Voici mon implémentation pour l'un des écrans (la liste des ingrédients) :

```
private func setupVoiceOver() {  
    // accessibilityLabel  
    ingredientTextField.accessibilityLabel = "Put your ingredients here."  
    addIngredientButton.accessibilityLabel = "Add your ingredient."  
    clearIngredientsButton.accessibilityLabel = "Clear your ingredient list."  
    ingredientList.accessibilityLabel = "Your ingredient list."  
  
    // accessibilityValue  
    ingredientTextField.accessibilityValue = "Example : lemon, cheese, sausage."  
  
    // accessibilityHint  
    addIngredientButton.accessibilityHint = "Your ingredient will be displayed on the list."  
    searchRecipesButton.accessibilityHint = "Available recipes will be displayed."  
}
```

Voici les éléments que j'ai jugé pertinent de préciser.

Prenons l'exemple du bouton pour ajouter un ingrédient :



VoiceOver, grâce à la fonction `setupVoiceOver()` pourra alors lire, pour le bouton d'ajout :
« Add your ingredient ».

Il indiquera également à l'utilisateur qu'il s'agit un bouton en disant : « button ».

Afin d'apporter encore plus de détail, propriété `accessibilityHint` nous a permis de préciser à l'utilisateur l'indication suivante :

Help: Your ingredient will be displayed on the list.