

# Projet Android Mamazone

## Quoi ?

Mamazone est notre projet de M2 MIAGE dans le cadre du module de développement Android. Il s'agit d'une application permettant d'explorer les livres d'une API, et de constituer une watchlist. Plusieurs options de recherches permettent de filtrer par genre, par auteur, et de trouver des résultats similaires.

## Fonctionnalités

Mamazone permet principalement de rechercher et marquer des livres en favoris, depuis l'API Google Books. La liste des livres est stockée localement dans un localStorage, l'utilisateur ne perd donc pas sa liste à la fermeture de l'application.

En plus de la recherche par auteur et par titre, l'utilisateur peut aussi scanner l'ISBN en code-barre d'un livre simplement avec l'appareil photo du téléphone, afin de retrouver la fiche en ligne.

Pour les personnes malvoyantes, ou bien pour un confort supplémentaire dans la même idée qu'Audible par exemple, il est également possible d'utiliser la fonctionnalité TTS d'Android pour lire à haute voix le résumé d'un livre.

## Langages

Nous utilisons VueJS pour développer une application web, et Capacitor pour la porter et l'intégrer dans une application pour Android.

La majorité du code est en Javascript.

## Librairies

[VueJS](#) pour le développement web

[Capacitor](#) pour la conversion en application Android

[Bulma](#) pour le style

## APIs

Nous utilisons l'API REST Google Books afin d'obtenir des informations sur les livres. L'API permet une grande efficacité dans les recherches, que ce soit par auteur, par titre, voire les recherches multi-critères.

Le gros désavantage de cette API est la structure de l'objet JSON retourné, qui est particulièrement chargée, proposant parfois des imbrications inutiles, et des informations redondantes.

## Installation

### Sur serveur web

Vous allez avoir besoin de NodeJS et de NPM pour commencer :

```
apt install nodejs npm
```

Puis d'installer l'application depuis npm :

```
npm install
```

Pour build l'application :

```
npm run build
```

Pour exécuter l'application (\*serve\* fait un \*build\* au préalable, si ce n'est pas déjà buildé):

```
npm run serve
```

### Sur mobile

Nous utilisons Capacitor pour déployer sur smartphones. Par conséquent, il est nécessaire de faire les manipulations précédemment décrites au préalable.

Une fois l'application web installée depuis NPM, lancer la commande :

```
npx cap sync
```

Cette commande génère un projet Android avec Gradle, qu'il est ensuite possible d'ouvrir avec Android Studio. Ce dernier peut alors générer un APK pour installer l'application sur téléphone.

### Sur mobile via APK

Un APK compilé et comportant les dernières fonctionnalités est stocké dans le répertoire :

```
/build/release/release-app.apk.
```