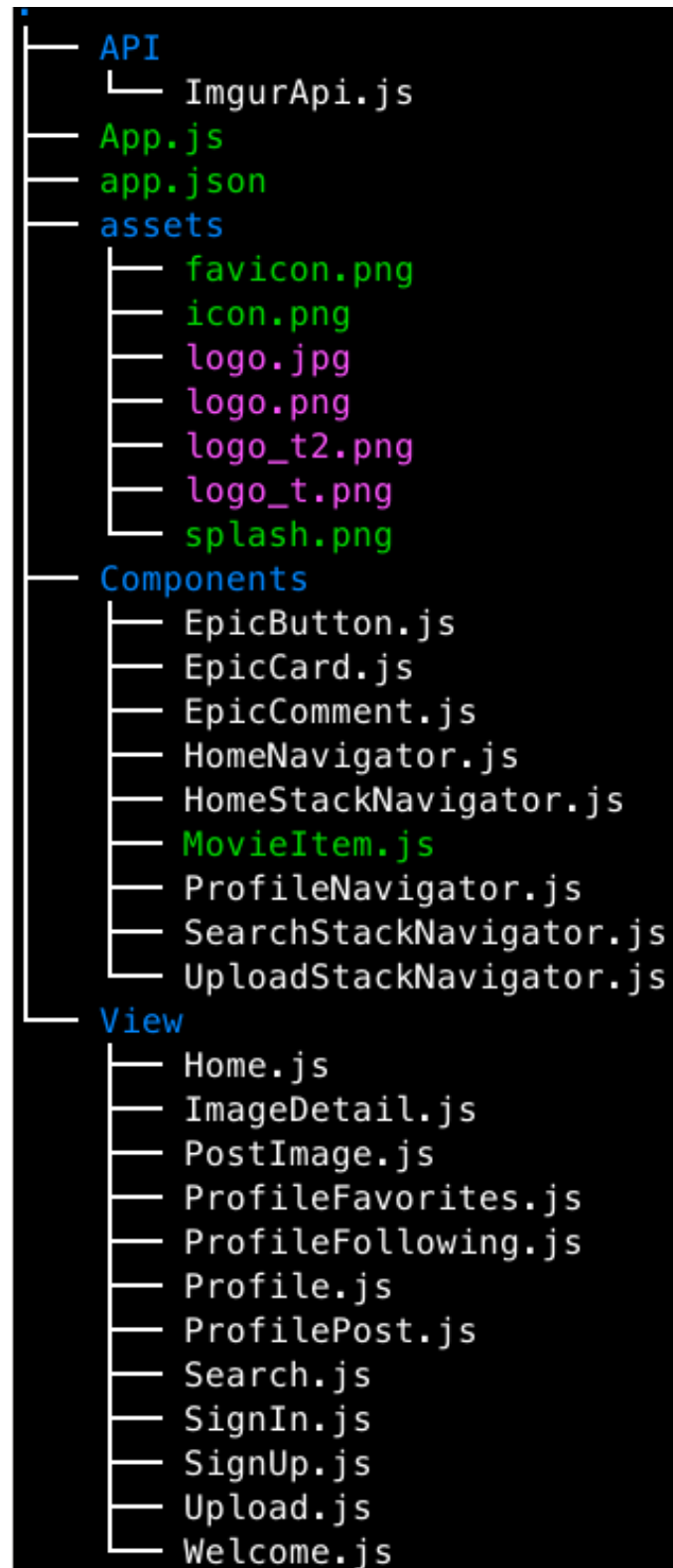


# DOCUMENTATION TECHNIQUE

Chimgurna

Wang Zhiwen  
Zhou Tony

# ARCHITECTURE



# ARCHITECTURE

↳ API

↳ ImgurApi.js

## Une page contenant tous les calls API.

- **const CLIENTID:**

Constante avec la data de la clé API récupérée sur Imgur.

- **getAccountFavorites(username):**

Retourne un objet contenant les images favorites de l'utilisateur.

- **getAccountPost(username, userAccessToken):**

Retourne un objet contenant les images postées de l'utilisateur, prend en paramètre le nom de l'utilisateur et le token d'accès.

- **searchByInput(input):**

Retourne un objet contenant les images avec le mot clé indiqué par l'utilisateur dans le paramètre input.

- **getAuthorizationUserUrl():**

retourne une string contenant le lien pour l'autorization.

- **getHomeViralFeed():**

Retourne la tendance actuelle la plus virale.

# ARCHITECTURE

↳ API/

↳ ImgurApi.js

- **getComments(galleryId):**  
Retourne un objet contenant les commentaires spécifiques à une gallery
- **getAccountFavorites(username):**  
Retourne un objet contenant les images favorites de l'utilisateur.
- **getUserAvatar(username):**  
Retourne un objet contenant l'image de profil de l'utilisateur spécifié.
- **postImage(userAccessToken, data):**  
Envoie d'une image. Les paramètres du body sont en JSON.
- **postComment(userAccessToken, commentMessage, imageId):**  
Envoie un commentaire sur l'image spécifié
- **voteGallery(userAccessToken, galleryId, type):**  
Envoie un vote "up", "down" ou "veto" sur la gallery spécifié
- **favorite(userAccessToken, galleryId):**  
Met en favoris la gallery spécifiée

## ARCHITECTURE

### ↳ App.js

Main page, la navigation entre les pages du début sont spécifiés aussi.

## ARCHITECTURE

### ↳ ASSETS/

Dossiers contenant les assets et images

## ARCHITECTURE

### ↳ Components/

Dossiers contenant les composants personnalisés utilisé dans react native

## ARCHITECTURE

### ↳ COMPONENTS/

### ↳ EpicCard.js

Composants affichant les cards avec titre, image, bouton de vote et de favoris.

Il prend en paramètre:

- data: la dataSource
- userInfo: une structure avec les informations de l'utilisateur
- navigation: du module de react-native-navigation

## ARCHITECTURE

↳ COMPONENTS/

↳ EpicComment.js

Components affichant les cards avec titre, image, bouton de vote et de favoris.

Il prend en paramètre:

- comment: object d'un comment de Imgur

## ARCHITECTURE

↳ COMPONENTS/

↳ HomeNavigator.js

Composent de la barre de menu, fais les liens entre chaque tabs. C'est la qu'on personnalise l'aspect du bouton

## ARCHITECTURE

↳ COMPONENTS/

↳ ProfileNavigator.js

Composent d'une view de la barre de menu, qui mène a la vue du profil utilisateur. C'est la qu'on personnalise l'aspect du bouton

## ARCHITECTURE

### ↳ COMPONENTS/

### ↳ SearchStackNavigator.js

Composant d'une view de la barre de menu, qui mène à la vue de recherche d'image. C'est là qu'on personnalise l'aspect du bouton

## ARCHITECTURE

### ↳ COMPONENTS/

### ↳ UploadStackNavigator.js

Composant d'une view de la barre de menu, qui mène à la vue de l'upload d'image. C'est là qu'on personnalise l'aspect du bouton

## ARCHITECTURE

↳ **VIEW/**

Dossier contenant toutes les views.  
Une View est une page que l'utilisateur peut voir.

## ARCHITECTURE

↳ **VIEW/**

↳ **Home.js**

La view du Home, c'est à dire la page d'accueil une fois  
loggée

## ARCHITECTURE

↳ **VIEW/**

↳ **ImageDetail.js**

La view lorsqu'on accede à une image ou vidéo en  
particulier.

## ARCHITECTURE

↳ **VIEW/**

↳ **PostImage.js**

La view ou on peut choisir son image à upload



## ARCHITECTURE

↳ VIEW/

↳ ProfileFavorites.js

La view qui permet de voir les images mit en favoris par l'user

## ARCHITECTURE

↳ VIEW/

↳ Profile.js

La view qui mène à la page de profile

## ARCHITECTURE

↳ VIEW/

↳ ProfilePost.js

La view qui mène aux images postées par l'user

## ARCHITECTURE

↳ VIEW/

↳ Search.js

La view ou l'utilisateur peut recherche une image

## ARCHITECTURE

↳ VIEW/

↳ SignIn.js

La view sur l'écran de connection

## ARCHITECTURE

↳ VIEW/

↳ Upload.js

La view qui permet d'upload son image après l'avoir choisit

## ARCHITECTURE

↳ VIEW/

↳ Welcome.js

La premiere page de l'application, demande à l'utilisateur de se log ou de s'inscrire ou alors de continuer en tant qu'invité.