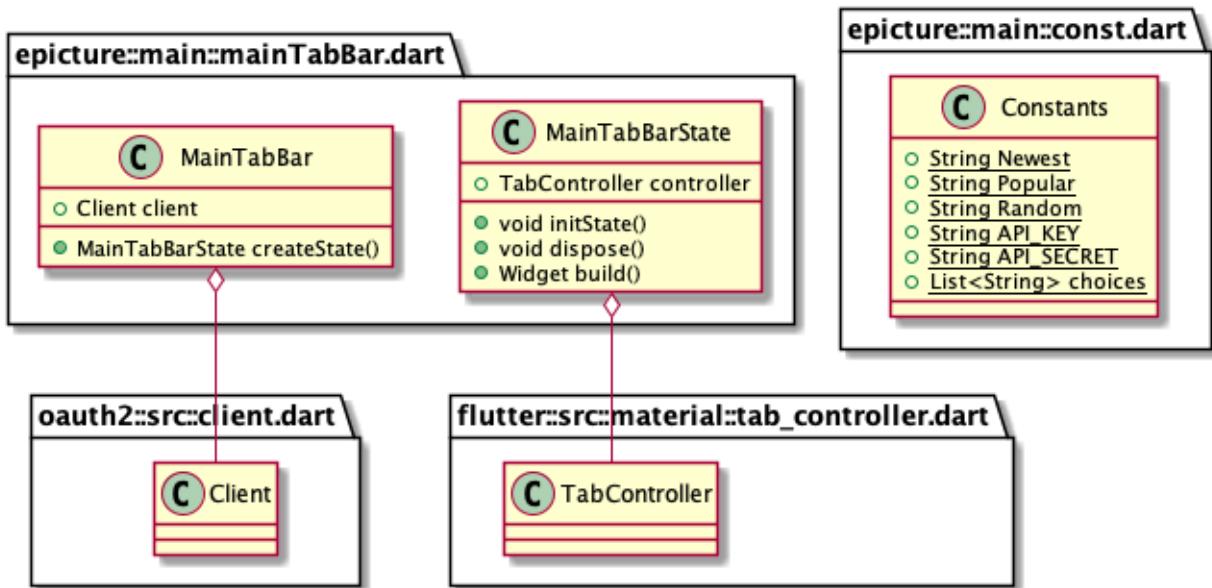


Epicture documentation



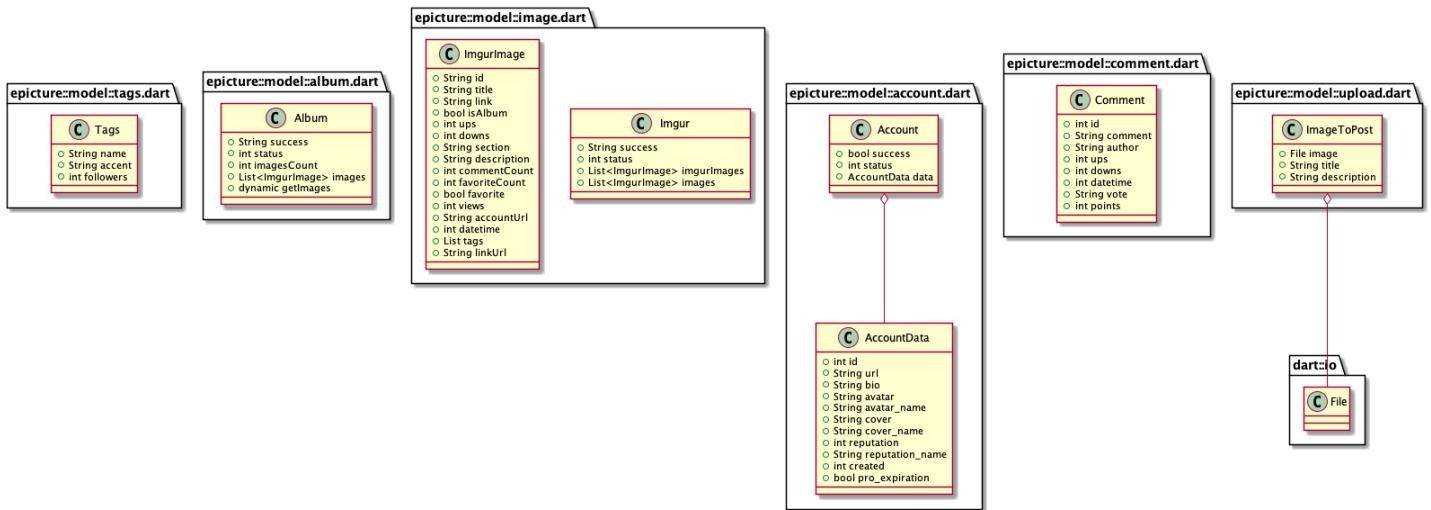
The application is based on a **TabController**. It allows to add pages easily to the application. There is one directory for each page.

You can find all the constants in the file **const.dart**.

If you want to connect your Imgur account to the application you need to modify the constant **API_KEY** and **API_SECRET**.

API

All the models recovered by the API are in the folder models.
The API request are in the **requests.dart** file.



Home



epicture::model::image.dart

C ImgurImage

- String id
- String title
- String link
- bool isAlbum
- int ups
- int downs
- String section
- String description
- int commentCount
- int favoriteCount
- bool favorite
- int views
- String accountUrl
- int datetime
- List tags
- String linkUrl

C Imgur

- String success
- int status
- List<ImgurImage> imgurImages
- List<ImgurImage> images

The Home page is used to display the user's feed. The content of the feed is stored in a list of **ImgurImage** object.

C HomeBarState

- BuildContext context
- Widget homeBar
- void choiceAction()
- Widget build()

C HomeBar

- BuildContext context
- Client client
- State createState()

C ImageCardState

- ImgurImage photo
- String renderUrl
- Color favColor
- bool isFav
- Color upColor
- bool isUp
- Widget imageCard
- Widget build()

C ImageCard

- ImgurImage photo
- Client client
- bool isCommentPage
- State createState()

The first Button allow to add filter on the feed.

If you want to add filter choice for the feed, you need to modify the function **choiceAction** and add the filter to the list choices in **const.dart**.

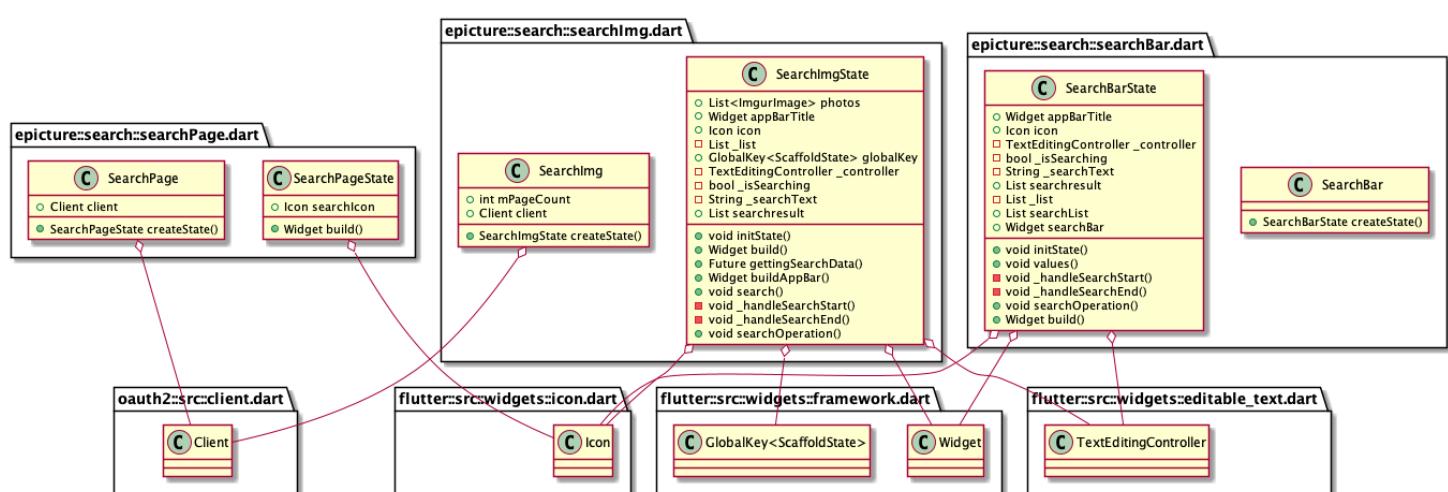
C PhotosList

- List<ImgurImage> photos
- Client client
- PhotosListState createState()

The Homepage is a **PhotosList** object.

It's a infinite list of **ImageCard**, to create one you need to give a list of **ImgurImage** as argument.

Search



Profile



The profile show information about the user.

The class contains a TabBarView where you can add tabs.

profilePage class display the appBar with avatar and name of the user. The rest of the data is displayed in the tabs.

There is one class for each Tab in the profile.

The posts are **ImageGrid** object and page are **photosGrid** object. It's the same as **PhotosList** in Home page.

