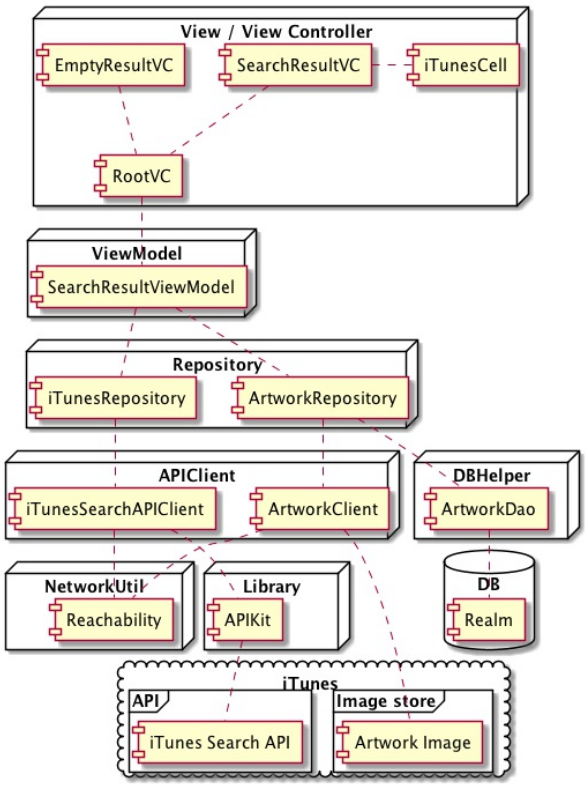
iTunes検索アプリ

クラス関連図



主なクラス

View

名前	説明
iTunesCell	iTunes Search APIの検索結果を表すTableViewCell

VC

名前	説明	備考
	検索結果を表示する	

RootVC	TableViewのVC	
SearchResultVC	検索結果を表示する TableViewのVC	
EmptyResultVC	検索結果が無かった場合に表示するVC	「該当の音楽が見つかりません。」を表示

ViewModel

名前	説明
SearchResultViewModel	検索結果画面のViewModel

Repository

名前	説明
iTunesRepository	iTunes Search APIの検索結果を供給する
ArtworkRepository	検索結果の画像を供給する

APIクライアント

名前	説明
iTunesSearchAPIClient	ツイッターのAPIとの通信を行う
ArtworkClient	検索結果の画像を取得する

DB

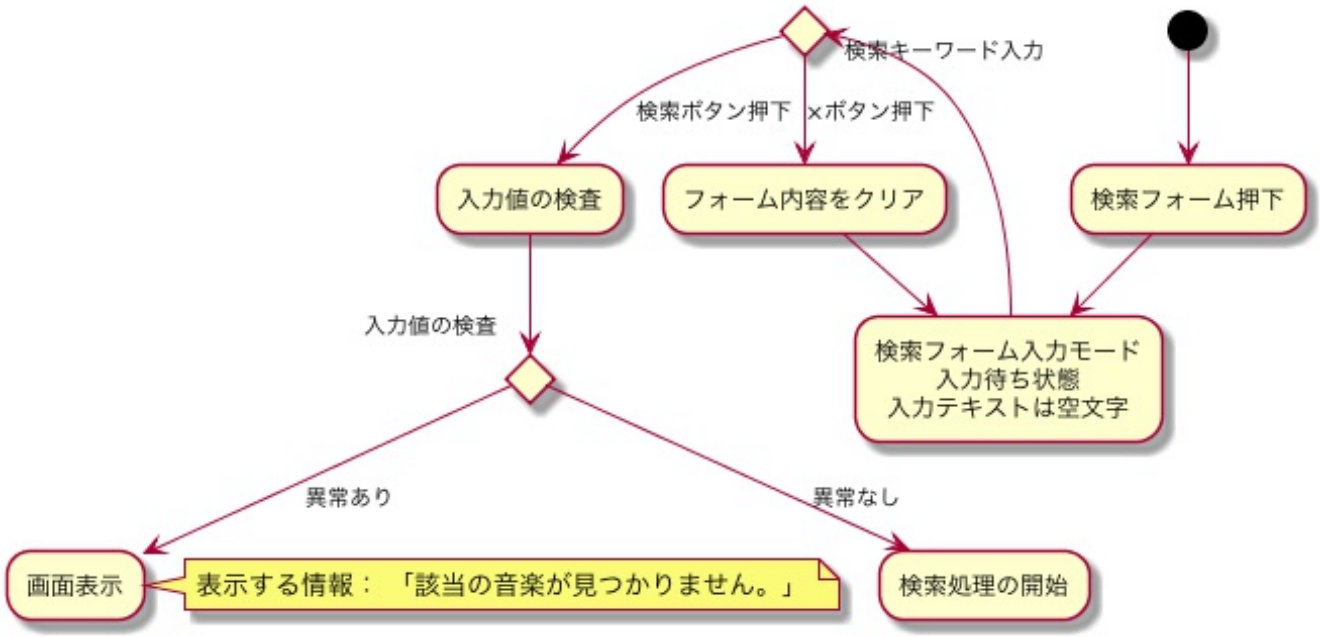
名前	説明
ArtworkDao	Realmとの情報の入出力を担当する

モデル

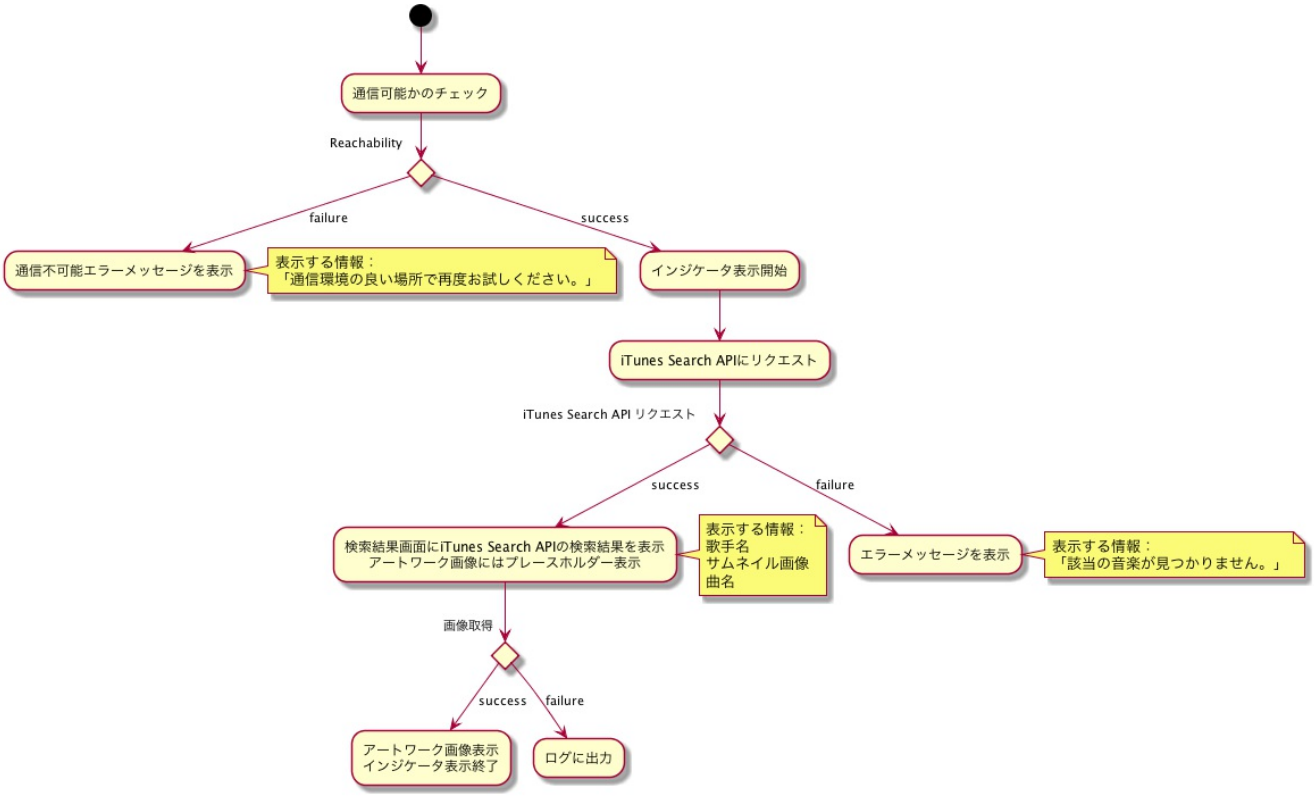
名前	説明
iTunesTrack	iTunes Search APIの検索結果を表す
Artwork	画像情報
ArtworkDto	Realmとの入出力に使用

フローチャート

ユーザー検索フォーム入力時



検索処理



表示項目

APIレスポンスの項目のうち、表示に使用するのは以下の通り。 表示順は、返却された際の順番をそのまま用いるものとする。

表示項目	対応するレスポンスJSONの項目名	備考
アーティスト名	artistName	
サムネイル画像	artworkUrl100	
曲名	trackName	

モデル

エンティティ

検索結果を表す構造体、画像情報を表す構造体が必要。

Artwork

画像情報を表す構造体。

変数名	型	説明	備考
id	Int	ID	
url	String	アートワーク画像のURL	
image	UIImage	アートワーク画像のイメージ	

ArtworkDto

画像情報を表す構造体。RealmのObject型を継承。

変数名	型	説明	備考
url	String	アートワーク画像のURL	
data	Data	アートワーク画像のバイナリデータ	検索結果と別途にダウンロードする必要がある
image	UIImage?	アートワーク画像のイメージ	artworkDataから変換して渡す計算型プロパティ

iTunesTrack: Codable

iTunes 検索 APIのレスポンスのうち曲情報を表す構造体。

変数名	型	説明	備考
artworkUrl	String	サムネイル画像のURL	JSONからのデコード時、artworkUrl100とのマッピングが必要
artistName	String	アーティスト名	

trackName　String　曲名

永続化

Realmを使用して、画像と画像URLの組を保存する。

ArtworkDto: Realm.Object

Realmに入出力するアートワークの情報。画像のキャッシュとして動作する。

変数名	型	説明	備考
id	Int	ID	primaryKeyに設定
url	String	アートワーク画像のURL	
_image	UIImage?	アートワーク画像のイメージ	デフォルト値をnilで設定
imageData	Data?	アートワーク画像のバイナリデータ	デフォルト値をnilで設定
image	UIImage?	アートワーク画像のイメージ	計算型プロパティ

```
class ArtworkDto: Object {
    @objc dynamic var id = 0
    @objc dynamic var url = ""

    @objc dynamic private var _image: UIImage? = nil
    @objc dynamic private var imageData: Data? = nil
    @objc dynamic var image: UIImage?{
        set{
            //imageにsetすると、自動的に_imageに値が保持され、imageDataにも変換&setされる。
            self._image = newValue
```

```
            if let value = newValue {
                self.imageData =
UIImagePNGRepresentation(value)
            }
        }
        get{
            if let image = self._image {
                return image
            }
            if let data = self.imageData {
                self._image = UIImage(data: data)
                return self._image
            }
            return nil
        }
    }

    override static func primaryKey() -> String? {
        return "id"
    }

    override static func ignoredProperties() ->
[String] {
        return ["image", "_image"]
    }
}
```

その他

ATS対応

指定されたアートワークのURLがHTTPSでないときがある。そのため、ATSの除外設定が必要。 [iOS 9] iOS 9 で追加された App Transport Security の概要 | Developers.IO

ATSを無効にする

```
<key>NSAppTransportSecurity</key>
<dict>
    <key>NSAllowsArbitraryLoads</key>
    <true/>
</dict>
```

ATSの対象外となるドメインを指定する

```
<key>NSAppTransportSecurity</key>
<dict>
    <key>NSExceptionDomains</key>
    <dict>
        <key>insecure.example.com</key>
        <dict>

<key>NSTemporaryExceptionAllowsInsecureHTTPLoads</key>

            <true/>
        </dict>
    </dict>
</dict>
```

基本的にATS を無効にし、ATS の対象にするドメインを Info.plist に記載する

```
<key>NSAppTransportSecurity</key>
<dict>
    <key>NSAllowsArbitraryLoads</key>
    <true/>
    <key>NSExceptionDomains</key>
    <dict>
        <key>secure.example.com</key>
```

```
        <dict>
<key>NSTemporaryExceptionAllowsInsecureHTTPLoads</key>
            <false/>
        </dict>
    </dict>
</dict>
```

Reference

第十回スキルアップ　実践編 · stv-ekushida/iOSTraining Wiki

iTunes Search API - Affiliate Resources

View

StoryboardのUIView一部切り替えについて

Library/スニペット

sahara-ooga/ios-realm-demo: iOS Realmを利用したCRUD操作のサンプルです。(Swift4)

sahara-ooga/RepositoryTool: library to use repository pattarn in Swift.

sahara-ooga/ios-uikit-uisearchbar-demo: iOS UISearchBarのサンプル (Swift4)

ドキュメント作成

PlantUML Cheat Sheet - Qiita Visual Studio CodeでMarkdown目次(TOC)を作成する便利Plugin - Qiita

Realm

RealmがRealmが対応しているNSPredicateのクエリ

NSPredicate Cheatsheet

[iOS] Realmを使ってみた ～環境構築からCRUDまで～ | Developers.IO

その他

vanilla-di-manifesto/swift at master · vanilla-manifesto/vanilla-di-manifesto

Reachability.swiftで通信状況を確認する - しめ鯖日記

Reachability.swift/ViewController.swift at master · ashleymills/Reachability.swift