

Androidアプリ開発における デザインパターンの提案

リコーITソリューションズ株式会社

若松和憲

wakamatsu.kazunori@gmail.com

Androidアプリ開発における問題

以下のような問題が頻発する

- Activityクラスの肥大化
- UIとビジネスロジックの混同
- DB接続周りの複雑化

しかしながら、
上記の問題を解決できるようなデファクトスタンダードな実装パターンは存在しない。

解決方法

問題を解決する実装のデザインパターンを提案する。
パターンを利用する事で、誰でも一定以上の品質のアプリが開発できるようになる事を目指す。

デザインパターン作成プロセス

世にリリースされている複数のアプリをソースコードレベルで解析し、共通して見つかった良い設計を実装のデザインパターンとして提案した。

調査対象のアプリケーション

以下を条件として、8つのAndroidアプリを選出した。

- ソースコードが公開されている
- 開発期間が1年以上
- アプリがGooglePlayに公開されている
- アプリがGooglePlayで1000人以上にダウンロードされている

デザインパターン作成手順

1. 解決する問題を選択
2. 調査対象のAndroidアプリを選出
3. 問題の解決に関心を絞って、Androidアプリのソースコードを解析
4. 3で良い設計が見つければ、抽象化してクラス図に起こす
5. 共通して複数のAndroidアプリで良い設計が見つければ、それをパターンとする

作成したデザインパターン

今回、解決する問題を「Activityクラスの肥大化」に絞ってデザインパターンの作成を行い、4つのパターンを作成する事ができた。そのうちの1つを紹介する。

【パターン名】
SplitLayoutPattern

【概要】
1つの画面に複数の並列するLayout要素がある場合、1つ1つのLayout要素に対して、ビジネスロジックの実装クラスを用意する。

