drivemode

Keishin Yokomaku / Startup Android #1

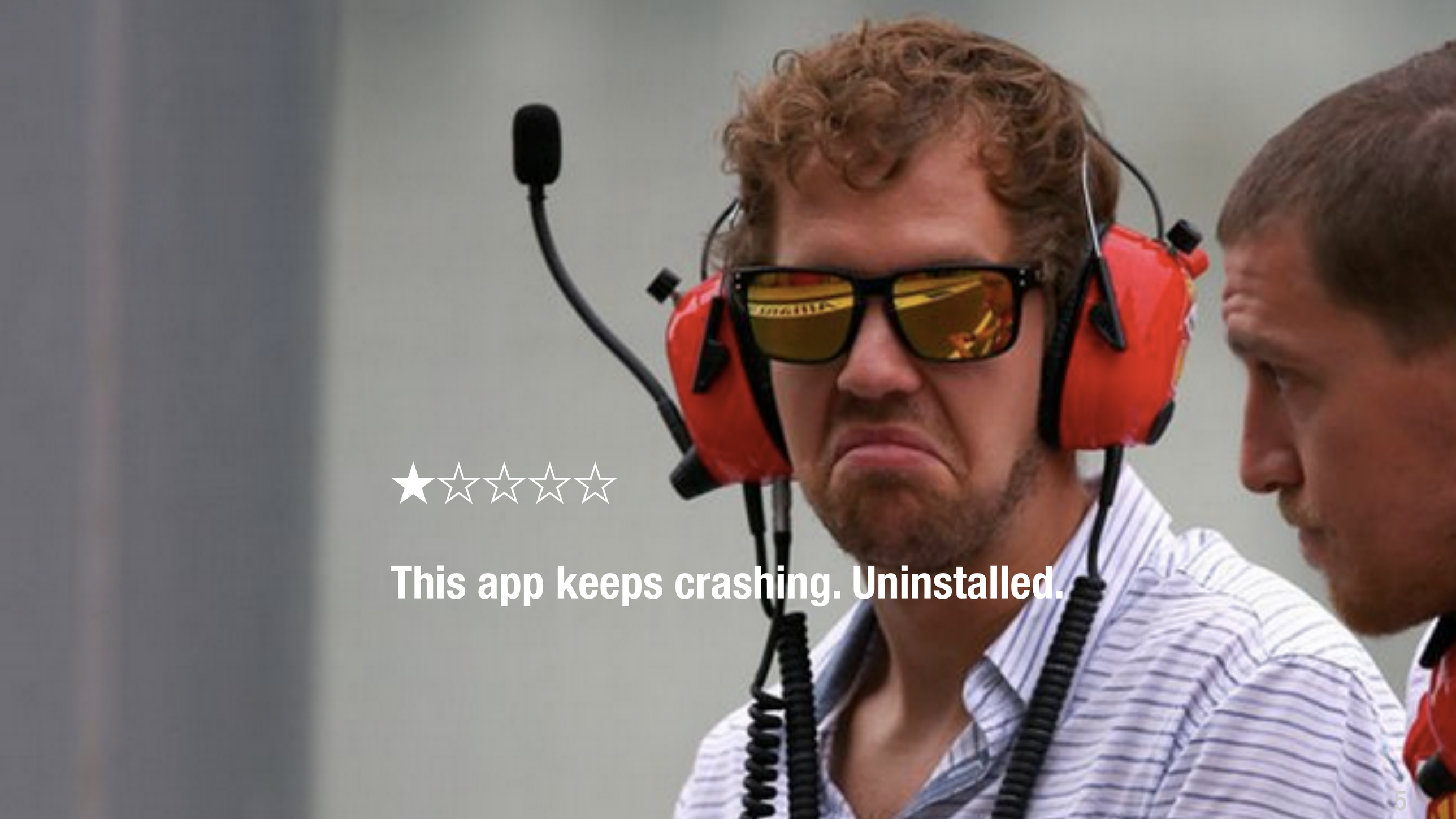
致命傷を避けるための例外実例集

About Me

- Keishin Yokomaku
-  Drivemode, Inc. / Principal Engineer
-  KeithYokoma: GitHub / Twitter / Qiita / Tumblr / Stack Overflow
  - Books: Mobile App Dev Guide / Android Academia / Grimoire of Android
  - Fun: Gymnastics / Cycling / Photography / Motorsport
  - Today's Quote: "時代は筋肉"

例外


18:08
Android Wear
Audible
Automatic
Calendar
Chrome
Chromecast
DeployGate
Drivemode
問題が発生したため、Fitを終了します。
OK
Endomondo
Events
Evernote
Expensify
Facebook
Fit
Fit
freee


★☆☆☆☆
This app keeps crashing. Uninstalled.

傷は浅いうちに塞ぐ

- リリース戦略とクラッシュ検知の仕組みは必須テクニック
  - Fabric(Crashlytics) でクラッシュログの収集
  - Google Play Developer Console で段階的リリース
- ユーザフィードバックの変化に気付く
  - 「最近こんなフィードバックがよく来てる」
  - 「あのフィードバック減った」

不条理と戦う

- 明らかにクラッシュすると分かるものだけが原因ではない
  - ユーザの使用状況によってクラッシュしたりしなかったりするものもある
- 端末の違い、使い方の違い、他にインストールしているアプリの違い
  - 様々な要因が予想外のクラッシュを誘発する
- ときには見知らぬ例外が原因であることもある
  - 例外の名前やメッセージから状況や理由を読み取るスキルを身に着けよう

Let’s get started!

- 本日の例外レシピ
  - 有名どころからどマイナーなものまでモリモリコース

NullPointerException	StackOverflowError
OutOfMemoryError	Illegal**Exception
CursorWindowAllocationException	TransactionTooLargeException
DeadObjectException	

NullPointerException

NullPointerException

- 一番よくあるクラッシュの原因
- 起きる理由は様々
  - コード上での検査漏れ
  - Nullable でない場所に null を代入
  - オブジェクトのライフサイクルに対する考慮漏れ

NullPointerException

- むやみに対策を取ってはいけない
  - e.g. あらゆるメソッドで null チェックを入れる
  - 防御的になりすぎると後々のメンテナンスで足を引っ張ってしまう

NullPointerException

- 対策の方法
  - 設計による対策
    - メソッドが null を許容するかどうか表明することで、
      誰がnull チェックの責務を負うか決める
    - NullObject パターンを使って、何も起きないように工夫する

NullPointerException

- 対策の方法
  - フレームワークのコードの把握
    - どういう状況でメソッドが null を返すか知っておく
    - Support Library の実態を知っておく
      - 空実装のままリリースされている時がある
    - フレームワークの作法に準じた作り方を覚える

OutOfMemoryError

OutOfMemoryError

- 画像や動画など大きなデータを扱うと必ず目にする例外
  - 文字通り、空きのないヒープにオブジェクトを作ろうとしているときに起きる
- 対策方法はわりと明快
  - 大きなオブジェクトを小さくする
  - 同時にたくさんのオブジェクトをヒープに入れない
  - メモリリークをなくす

StackOverflowError

StackOverflowError

- スタックを使い潰したときに起きる例外
  - e.g. 無限再帰呼び出し
- 見落としがちな箇所
  - 深すぎるレイアウトのネスト: レイアウトも結局は Java のオブジェクトになる
  - 深すぎる依存関係: 過剰なレイヤ化でメソッドのコールスタックに限界がくる

StackOverflowError

- 対策方法
  - レイアウトの階層構造を減らす
    - RelativeLayout や ConstraintLayout などでフラットな構造にする
  - 適切な粒度で設計をする
    - Clean Architecture や DDD などを参考に

Illegal**Exception

Illegal**Exception

- 種類は色々
  - IllegalStateException, IllegalArgumentException, etc...
- 呼び出される側が想定していない状態・引数に直面したときに起きる
  - 呼び出す側が引数チェックや状態チェックをすることを要求している
- 端末依存で公式の Javadoc に書いてないときに飛んでくることがある
  - 怪しいな？と思ったらすぐに状態や引数をログにつめて集められる仕組みを

Illegal**Exception

- 対策方法
  - すぐに直せるものばかりではないことに留意する
  - 必要な情報を集め、対策を練ってから直すこと
    - Crashlytics のログ収集機能で事前条件をチェックするとよい

CursorWindowAllocation Exception

CursorWindowAllocationException

- データベースを扱うと目にする例外
  - Cursor がもつデータが許容範囲を超えたときに起きる
  - リファレンスに Javadoc が無い例外だが、雑に作ると目撃するハメに
  - CursorWindow クラスの中でハンドリングしている
    - CursorWindow.java: http://bit.ly/2eKlL5y

CursorWindowAllocationException

- 対策方法
  - 一気に大量のデータを取得している部分を細切れにする
  - データベースに巨大なデータを保管しない
  - 使い終わった Cursor は close する

TransactionTooLarge Exception

TransactionTooLargeException

- プロセス間通信でメモリを使いすぎたときに起きる例外
  - 1つのプロセスに対して割り当てられるプロセス間通信用の領域は1mB
- 意外なところでプロセス間通信は使われている
  - Context#startActivity
  - PackageManager#queryIntentActivities
  - 他多数

TransactionTooLargeException

- Intent の extra に 1mB を超える String を渡すことでも起きる程度には身近なところに可能性がある
- 対策方法
  - 巨大なデータを直接扱わない
  - プロセス間通信の時間を短くする
  - プロセス間通信の回数そのものを減らす

DeadObjectException

DeadObjectException

- 死んだプロセスに話しかけたときに起きる例外
  - 通信相手のプロセスがいつも生きているとは限らない
- 対策方法
  - ServiceConnection や IBinder.DeathRecipient で死活監視
  - 巨大なデータを通信にのせない
- System Service は死活監視できなさそう……

まとめ

まとめ

- 対策方法そのものに革新的なことはほとんど無い
  - よくある実例をそのまま適用できることも多い
- 考え方の基本
  - 処理の単位やデータを小さく細かくする
  - 設計と表明で責務を明らかにする

まとめ

- いかに早く情報を集めて対策が取れるかが重要
  - 段階的リリース
  - クラッシュログの収集
  - ユーザフィードバックの解析

drivemode

Keishin Yokomaku / Startup Android #1

致命傷を避けるための例外実例集

Registration starts from mid-November!