

知識ゼロから学ぶ ソフトウェアテスト【改訂版】

60216114 作島大智

テスト作成

- テストは構造的の弱くバグの多い2割から8割のバグを見つけることが重要.
- ソフトウェア開発者とは逆にソフトウェアテスト担当者は数学的知識の有無がその優秀度合いに大きく作用している.

テストの種類

ホワイトボックステスト

→プログラムの論理構造が正しいか

ブラックボックステスト

→プログラムの入力処理に対する
出力処理が適切か

探索的テスト

→テストケースベースのテスト

テストの種類

パフォーマンステスト

→非機能要求のテスト

セキュリティテスト

→解析・変更or削除・監視の機能を3つすべて満たすか

フ ァ ズ テ ス ト

セキュリティ分野でのランダムテストを指す.

アプリケーションを基本コンポーネントに
分解し徹底的に調べ上げる.

テスト責任者の必要なスキル

- 例外が発生した場合にそれが本当に致命的問題であるのか軽微なのかを判断する能力
- 予算策定とコストに対する意識
→品質とコストは相関関係がある
- スケジュールをコントロールできるプランを立てることができる能力

実際にテストする

- テストする必要のない機能を正確に把握する.
- 重要な機能→テストが簡単な機能→残りの機能の順番で行う.

自動化について

- 主たる目的は「コスト削減」
- 自動化するよりも手動のほうがコストがかからない場合があることを認識すべき.
→テスト自動化はコード保守コストが大きい

自動化の対象

自動化に適するテスト

- ステークテスト
- パフォーマンステスト
- **API**のテスト

自動化に適さないテスト

- グラフィックやサウンドなどのメディア関係
- 回帰テスト

疑問点

- 構造的に弱いとは？
数値化などで定義し議論できるのか.
- テストケース実行で見つかるバグは**50%**以下というのは事実なのか.(本に書いてあった)
- 探索的テストを行える技量はどの程度なのか.
- テスト自動化におけるコード保守のコスト削減に関する研究はどんな感じか.

疑問点

- 信頼度曲線よくわからない
- 「コード10-15行でテストケース1つ By 日立」
は正確な基準なのか.

今後の方針

- 今週はこの本しか読めていないので論文に手を付ける.
- テスト自動化は難しいのではないかと考えているのでその自動化を成功させるために必要であるといわれるテストコードの保守はどうか.

参考資料

- 日本従事者のためのテストカンファレンス
(<http://www.jasst.jp/>)

- テストツール(OSS)

Selenium(<http://www.seleniumhq.org/>)はいい感じ

- Googleアルゴリズム

Scoreが高いファイルに70%の確率でバグが存在

次回までに

- テストツールのメンテナンスを理解するために継続的インテグレーション(CI)勉強する.
- CIツール(Jenkins,travis)

以下の論文読む.

- DeFlaker: Automatically Detecting Flaky Tests)
- Time to Clean Your Test Objectives
- 特にアプローチに着目して.

知識の理解度向上

- セキュリティテストはほかのテストと何が違うのか。
- 従来のソフトウェアテストは開発者がうまくプログラムできなかったソフトウェアの障害を未然に防ぐ方法論。
- セキュリティは悪意のある攻撃が引き起こす問題を見つけることはソフトウェアテストの範疇にはいままで含まれていなかった。