

テストイングライブ 試験計画書



チーム北国(きたぐに)
～北都システム(株)～

目次



- チーム紹介
- テストの仕方
- テスティングライブでは
- テスト設計
- テスト実施



チーム紹介



チーム北国(北都システム)

長谷川 聡 (業務歴:11年)

組込み系製品の評価が得意、第三者検証の観点で長年の経験からバグを嗅ぎつける。

高橋 勲充 (業務歴:9年)

組込み系製品の開発が得意、開発者視点の斬新な発想で深いところに埋め込まれたバグを引きずり出す。

岡本 陽介 (業務歴:3年)

組込み系製品の開発が得意、現在習得中のテスト技術を用いて論理的にバグを追い込む。

ミッション

他のチームより、早く多くのバグを発見してテストイ
ングライブで**優勝**する。



テストの仕方(通常時)

普段はこんな風にテストしています。

正常性を確認するテスト

【概要】

製品に問題のないことを保障するテスト。
OKであることを確認して、安全性の根拠を示す。

【項目抽出のポイント】

できるだけ多くの機能に触る
機能やイベント一覧やマトリクス化して試験を設計

【試験実施のポイント】

基本機能から確認を行い複雑なものへ、製品の成熟度に合わせて実施を計画
要員数/環境数等も含めてテストを計画

機能試験

シナリオ試験

安定試験

リリース確認試験

連続試験

問題を見つけるテスト

【概要】

製品に作りこまれてしまっている問題を見つけるテスト。
NGを発見して、製品に欠陥があることを示し対処を促すことで品質を上げる。

【項目抽出のポイント】

問題の多く残っていそうな機能、問題の多く残っていそうな観点を重点的に設計
- 複雑な機能、開発規模、前工程の状況、開発チーム構成、類似製品の傾向等

【試験実施のポイント】

リリース時期に合わせた、重要度、影響度を考慮した試験計画

複合試験

競合試験

負荷試験

ランダム試験

機能強化試験



テストングライブでは(1)

• 通常のテストとこんなに違う

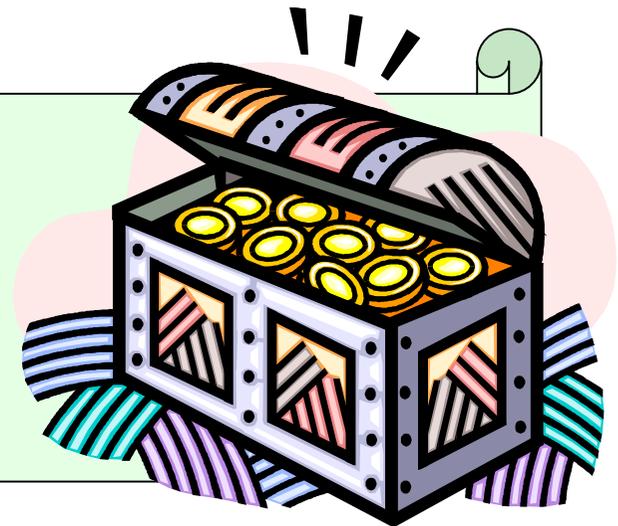
- ・正常性は確認しなくてよい。
- ・バグが残ってしまっているのではなく、意図的に作りこんでいる。
- ・極端に短い時間制限がある。
- ・他のチームより早く見つける必要がある。
- ・すべてのバグを見つける必要がない。
- ・バグにレベル(重要度)の差がない。

などなど

ミッション遂行のためには、
普段通りの考え方だけでは
駄目。

これは普通のテストでない。
テスト技術を地図にした

宝探した！！

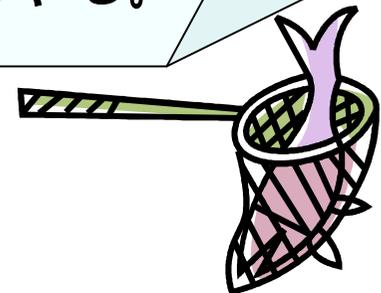


テストングライブでは(2)



【テスト方針】

- ・ バグを見つける試験のみ行う。
- ・ バグレベル(重要度)は無視し、簡単なバグを早く見つける。
- ・ バグ「残ってしまっているもの」でなく、「仕様」として考える。
- ・ バグレベルと個数を予測して、計画を立てる。
- ・ 1環境、3人チームの条件で効率的な実施を行う。
- ・ 製品品質レベルで、残バグ数で計画を見直す。
- ・ 目的達成のために、事前にできることはすべてやる。



テスト設計



Predict

埋め込まれたバグを見つける試験を行う

- ・幅広い機能にバグを入れてくるはず
- ・基本機能に問題が含まれているはず
- ・バグを埋め込む方針を考慮 **よくありがちなバグ**も入れてくるはず
- ・バグの埋め込み方を**開発視点**で考慮

初期化処理削除、エスケープシーケンス処理削除、変数不定値、配列範囲超過、Switch文Break削除、キャスト誤り、条件分の条件変更、無限ループ、文字列処理削除(¥0削除等)、数値丸め込み処理変更、イベントハンドラ削除、メモリ開放漏れ 等

Target

バグのレベルを分類し、分類ごとの項目を作成

- ・レベルA : **見つけやすいバグ**
- ・レベルB : **見つけづらいバグ**
- ・レベルC : **まず見つからないバグ** レベルCについては項目抽出しない。

Speed

実施スピードの追求

- ・短期決戦だから**敢えて手順を作成**し、同じ手順は2回やらない。
項目毎の手順ではなく、手順の中に項目を埋め込む手順設計。
- ・準備期間が長く、実施時間が少ない対策として事前に実施**リハーサル**を繰り返す。



テスト実施

Control

テストスケジュールについて

- ・バグのレベルを分類して、前半はレベルAから実施し次にレベルBへ。
- ・パターンの多い試験は最後に実施。

Flexible

ハーフタイムで計画見直し

- ・バグレベルごとの個数を予測し、ハーフタイム時のレベルAバグ数を基準に後半の作戦変更
 - バグ個数予測値(レベルA:16個、レベルB:20個、レベルC:4個)
- ・アプリケーション機能をカテゴリに分類し、ハーフタイム時の状況に合わせて後半は、バグの出していない機能を重点的に攻める。

Assign

チームの役割分担

- ・オペレーター: 試験の実施をする役割
- ・ナビゲーター: オペレータの試験手順をナビゲートする役割
- ・マネージャー: 問題の報告ならびに、バグ発見状況を管理する役割



ありがとうございました。
優勝目指してがんばります。