



TOKYO TEST FEST



Quantifying the health of E2E test automation

Nozomi Ito

About me



Nozomi Ito

- CEO of MagicPod
- MagicPod: no-code, cloud-based test automation tool
- The organizer of the Japan Selenium User Community
- Author of a Japanese Selenium book

- 1. Quantify the health of test automation by the health score**
- 2. Breakdown of the health score**
- 3. Improve the health score**

E2Eテスト自動化の成功をどう測定するか?

- コスト削減量?
- 見つけたバグの数?
- Four Keys?
 - デプロイ頻度、変更リードタイム、MTTR、変更失敗率
- ビジネスの成長?
- 利益率?



顧客のテスト自動化の成功
顧客が解約せずに**継続利用**してくれる

継続

料金に見合う成果が出ている
(少なくとも顧客はそう感じている)



解約

料金に見合う成果が出ていない

CANCELLED

「ヘルススコア」による成功度測定



1

100社以上のMagicPodユーザーにヒアリング実施
(解約ユーザー含む)



2

自動化の成果と相関が高そうな指標を元に「ヘルススコア」指標を作成



3

スコアと解約状況を見ながら、半年かけてチューニング



4

ヘルススコアを元にCSアクションを実施

ヘルスコア詳細

- 100点満点
- 80点以上が**グリーン**、80-50点が**オレンジ**、50点未満が**レッド**
- 1週間・1ヶ月などの単位で計算



ざっくり内訳

35点	<p>作ったテストは1日1回以上回しているか</p> <p>(実行したテスト数 x テスト実行日数) / (作成済テスト数 x 平日の日数)</p>
35点	<p>回しているテストは成功しているか 成功したテスト数 / 実行したテスト数</p>
20点	<p>メンテナンスしやすいテストの作りか 共通化の程度、長すぎるテストの割合、など</p>
10点	<p>その他</p>

ヘルススコアはテスト自動化の成功を測定できているか

レッドからオレンジに上がると、解約率が $\frac{1}{2}$ 以下に



レッドからグリーンに上がると、解約率が $\frac{1}{5}$ 以下に



ヘルススコアはテスト自動化の成功を測定できているか

レッドからオレンジに上がると、解約率が $\frac{1}{2}$ 以下に  

レッドからグリーンに上がると、解約率が $\frac{1}{5}$ 以下に  

解約率(≒自動化の成功度)をうまく測定できている

ヘルススコアはユーザーもアクセス可能



ユーザー自身でスコアを改善する取り組みも加速中

- 1. Quantify the health of test automation by the health score**
- 2. Breakdown of the health score**
- 3. Improve the health score**

ヘルススコア計算ロジック概要

35点	作ったテストは1日1回以上回しているか → テストからフィードバックを早いサイクルで得ているか
35点	回しているテストは成功しているか → テストのフィードバックに信頼性があるか
20点	メンテナンスしやすいテストの作りか → テストのフィードバックの信頼性を持続できる作りになっているか
10点	その他

- ヘルススコアは、「テストから信頼性のあるフィードバックを毎日得られるか」かを測定している
- 「コスト削減できたか」とか「バグを見つけたか」とかではない

ヘルススコア 計算ロジック詳細



ヘルスコア計算ロジック詳細

項目	点数
1. 十分な数のテストがあるか	6点
2. 十分な数のメンバーがプロジェクトにいるか	3点
3. 共有ステップを活用しているか	8点
4. 1つのテストが長すぎないか	5点
5. テストが安定するロケータースを使っているか	8点
6. テストを1日1回以上実行しているか	35点
7. テストの失敗率が高すぎないか	35点

1. 十分な数のテストがあるか(6点)

ある程度の数がないと、
自動テストで得られるメリットが限定的

- 20テスト(ユーザーストーリー)以上で満点



2. 十分な数のメンバーがプロジェクトにいるか(3点)

他のメンバーも巻き込むのが継続の秘訣

- 所有者含め4人以上で満点
- 組織の規模によって望ましい人数が違ってくるのが難しい



3. 共有ステップを活用しているか(8点)

共通化をしないとメンテナンスが大変になる

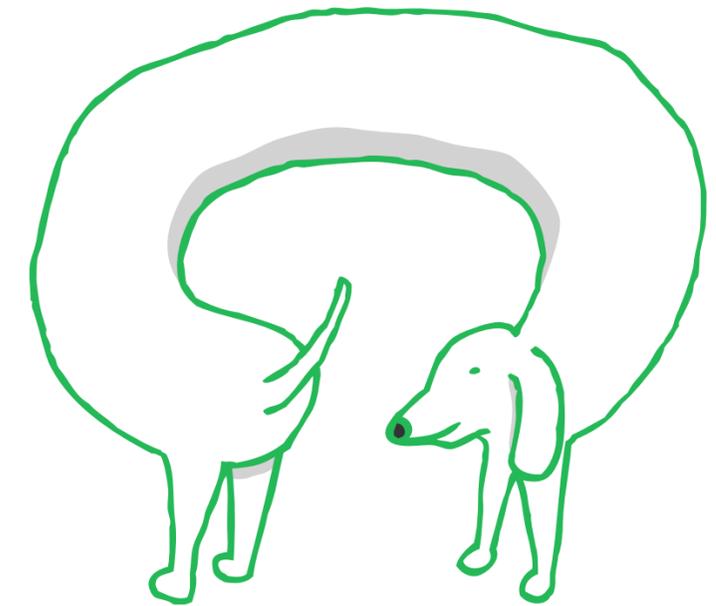
- テスト数の10-20%程度の数の共有ステップを作っていると満点



4. 1つのテストが長すぎないか(5点)

エラー切り分けが大変になるので、
不必要に長いテストは避けるべき

- 200-300ステップ以上で長いテスト扱い
- 長いテストが全体の10%以内なら満点
 - 長くせざるを得ないテストはあるので、満点ラインはそこまで厳しくしてない



5. テストが安定するロケータを使っているか(8点)

複雑なロケータは画面変更に弱い

- テスト内で使っているロケータの80%以上が安定ロケータなら満点



何が安定したロケータか

安定ロケータ

- idとaccessibility id
- 「#id」のCSSセレクター
- `//xxx[yyy='zzz']`形式のxpath

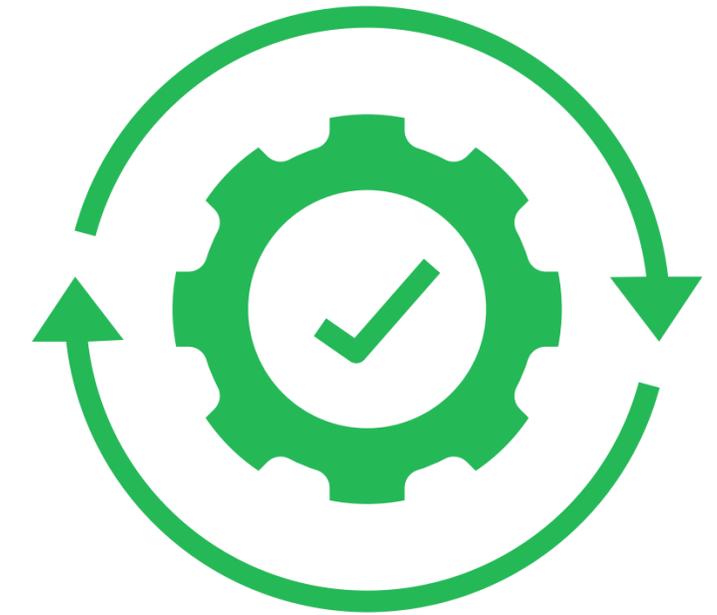
- 属性やテキストで要素が一意に特定されるもの
- containsやstarts-withもOK
- CSSセレクターでも同様のものはOK

不安定ロケータ

他は全て不安定扱い

テストで使う要素には、アプリ側で要素を特定できるid・属性・テキストをつけることを推奨

6. テストを 1日1回以上実行しているか(35点)



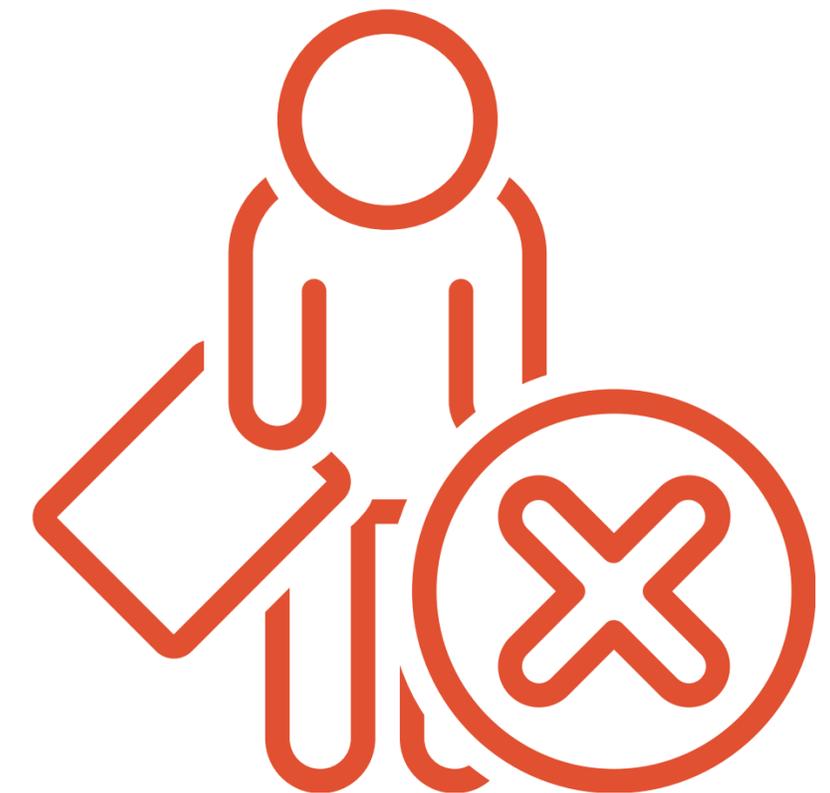
毎日回していないテストは
結局メンテナンスできなくなる

- 全てのテストを1週間のうち4.2日(平均平日日数)以上一括実行していると満点
- $(\text{実行したテスト数} \times \text{テスト実行日数}) / (\text{作成済テスト数} \times \text{平日の日数})$

7. テストの失敗率が高すぎないか(35点)

失敗率が高いのはメンテナンスされていないサイン

- 一括実行の成功の割合が90%以上なら満点



まとめ

項目	点数
1. 十分な数のテストがあるか	6点
2. 十分な数のメンバーがプロジェクトにいるか	3点
3. 共有ステップを活用しているか	8点
4. 1つのテストが長すぎないか	5点
5. テストが安定するロケータースを使っているか	8点
6. テストを1日1回以上実行しているか	35点
7. テストの失敗率が高すぎないか	35点

- 1. Quantify the health of test automation by the health score**
- 2. Breakdown of the health score**
- 3. Improve the health score**

ヘルスコアが上がらないのはなぜ？

MagicPodユーザーヒアリングで 見えてきた要因

1. プロジェクト体制
2. 開発チームの関与の欠如
3. E2Eテスト自動化しすぎ問題



1. プロジェクト体制

開発	テスト	タイプ	デプロイ頻度	ヘルススコア
自社	自社	完全内製	高	😊
外注	外注	完全外注	中	😐
自社	外注	第三者検証	中	😐
外注	自社	ユーザー受入テストの自動化	低	😓

そもそも「完全内製」以外はヘルススコアが上がりにくい

Solution

開発	テスト	タイプ	デプロイ頻度	ヘルススコア	対策
自社	自社	完全内製	高	😊	テストを毎日回して開発生産性向上
外注	外注	完全外注	中	😐	テストを毎日回して開発生産性を上げるよう外注先に働きかける
自社	外注	第三者検証	中	😐	チームの一員としてメンテナンスフェーズまで深く継続関与してもらう
外注	自社	ユーザー受入テストの自動化	低	😓	受入テスト自動化は、自社ではなく開発元に依頼

2.開発チームの関与の欠如

開発チームとの連携がないとE2Eテスト自動化の成功は難しい

- QAだけではCIにつなげない
- 開發生産性向上に活かせないと、自動化のメリットが大幅低下
- テスト側で頑張るよりプロダクトのテストバリエーションを上げる方が経済的なことが多い

Solution

- 開発チームもE2Eテスト自動化に関わる
- CIにつないで毎日実行
- フィードバックを早め、開発生産性を向上
- プロダクトのテストバリエーションを高める

3.E2Eテスト自動化しすぎ問題

1万件の手動テストケースを
全て自動化!!



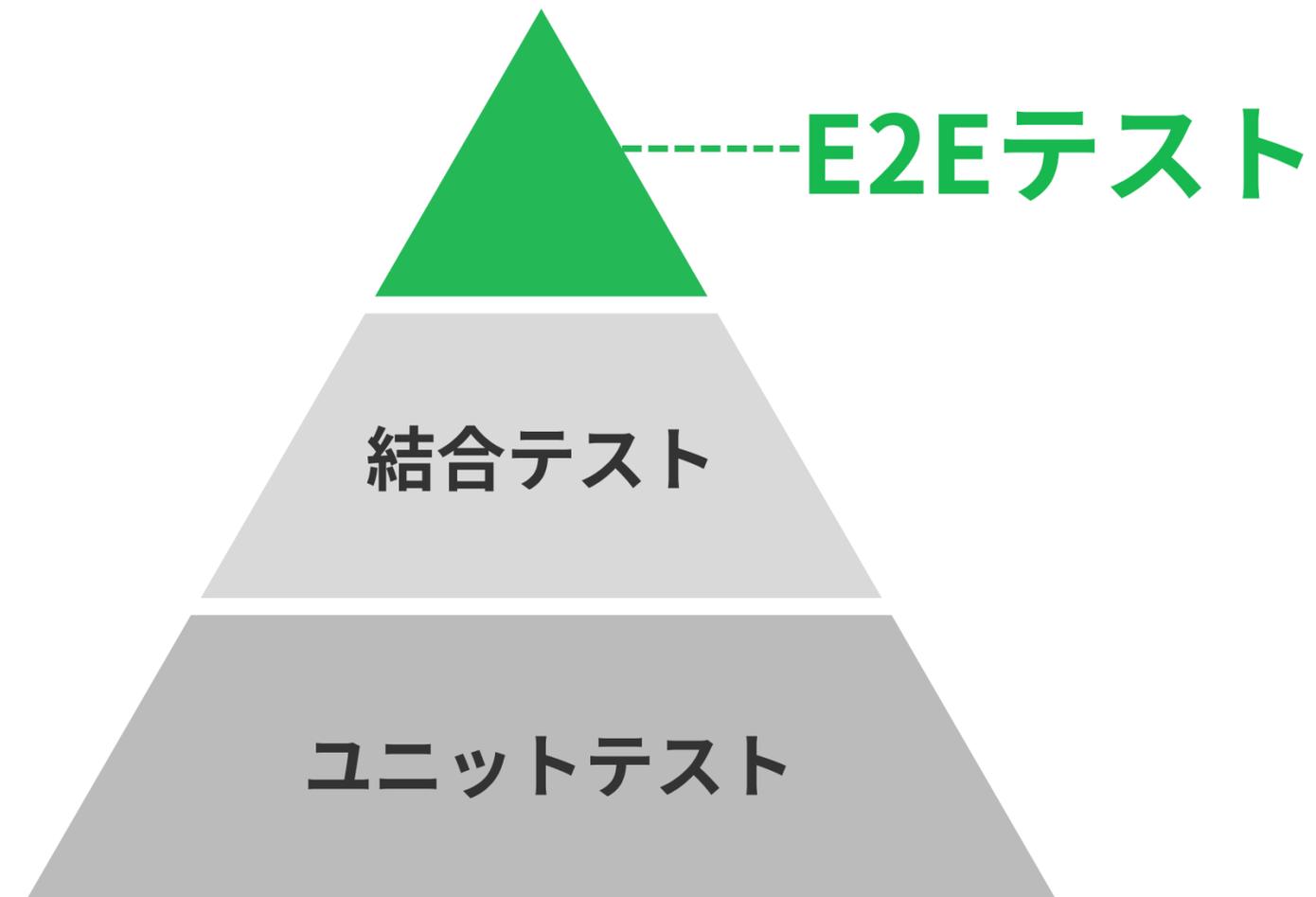
時間がかかる・不安定・共通化不十分により
メンテナンス不能に..



全てをE2Eテスト自動化する前提になってしまっている

自動テストケース設計から開発チームが関わるべき

適切なテストピラミッドのバランスにできる



E2Eテスト自動化成功のためにすべきこと

- テストを継続的に回し、素早くフィードバックを得る
- 安定して信頼性の高いテストの構築
- 開発者がE2Eテスト自動化に関わる

