

QAマネージャーが考える プロジェクトを成功に導くQAの取り組み

森 俊介、佐久間 浩明、米田 徹

Human Crest

ONE TEAMの精神でチーム一丸となり 心の距離は三密でプロジェクトを成功に導こう

アジャイル開発PJを成功に導く為にQAとしてすべきこととは？

株式会社 Human Crest
第一Quality Solution事業部 森 俊介

■発表概要

◆アジャイルQAとはなにか

- ・アジャイルQAに求められるもの

◆アジャイル開発の現場で直面した課題(実例)及び課題に対する考察

- ・リスク
- ・やるべきこと
- ・必要なスキル
- ・QAチームとしてのアプローチ

◆アジャイル開発プロジェクトを成功に導くために

■自己紹介



・10代

オーストラリアのシドニーにて伸び伸びと育つ

・20代

バックパッカーついで1年間アジアを放浪、その後アフリカや中東にも
足を延ばし、世界の音楽とビールを堪能

・30代

プロのギタリストとして日本全国をライブで訪れ、ご当地料理とビールを堪能

・40代

ITサラリーマンに転身し、ソフトウェアの品質と格闘の日々
息抜きにビールを堪能しすぎ、平成最後に日に痛風発症

◆アジャイルQAとはなにか

短いスパンでイテレーションが繰り返されるアジャイル開発に於けるQA

…… って、そのまんまですねw

ではそんなアジャイルQAに求められるものはなにか



◆アジャイルQAに求められるもの

・チーム内の壁をとっばらいPOや開発とチーム一丸で対応するONE TEAM精神

問題が生じた場合、『QA vs 開発』だったり『開発 vs PO』の構図になるのではなく『問題 vs チーム全体』の構図で対応できるよう PO、開発、QAが一つのチームとして対応する

・変化への対応力

顧客のニーズやビジネス市場の変化は頻繁に起こり得るが、これ等を事前計画を狂わす脅威等ネガティブなものではなく、よりよい成果を生み出す機会とポジティブに捉え、柔軟な姿勢で対応する

・スピード感とバランス感覚

短いスパンでのリリースに対応できるスピード感と限られた期間で最適なテストを実施する(観点やテストパターンを抽出する)バランス感覚

・俯瞰力

対象の製品の全体像をイメージしてテストを実施する

◆現場で直面した課題(実例)及び課題に対しての考察



■ケース① 最低限の仕様も固まっていない

・リスク

要件通りの実装にならない
リリースの遅れ

・やるべきこと

ある程度仕様が固まるまではスプリントバックログには入れない

・必要な能力

状況判断力

・QAチームのアプローチ

QA観点から最低限の仕様が固まっていない要件に対してはテストが出来ない旨 (品質面でリスクがある旨)を提案

■ケース② 仕様が複雑で開発側が正確に理解出来ていない

・リスク

要件通りの実装にならない
リリースの遅れ

・やるべきこと

PO⇔開発で齟齬が無くなるまで話し合う
実運用がイメージ出来ているPOと、出来ていない開発側ではそもそも見えてる景色が違っている
ここの乖離を埋めないと要件との大きなギャップが出来てしまうリスクがある

・必要な能力

コミュニケーション力 / 想像力

・QAチームのアプローチ

- ① QA観点での仕様の紐解き及び影響範囲の分析
- ② ①に基づいたテストパターンの策定

■ケース③ チケットの記載内容が分かり辛い

・リスク

要件通りの実装にならない
リリースの遅れ

・やるべきこと

要件を明確かつシンプルに記載するようにする
開発側は分かり辛い点はうやむやにせず確認する

・必要な能力

国語力(文章力)
積極性(不明点を遠慮せず確認するというマインド)

・QAチームのアプローチ

- ① QA観点での仕様の紐解き及び影響範囲の分析
- ② ①に基づいたテストパターンの策定

■ケース④ 一つのチケットに含まれる開発要件が多い

・リスク

一部の要件の実装漏れ

・やるべきこと

チケットを細分化する

・必要な能力

整理力

・QAチームのアプローチ

リファインメント時に細分化を提案
テストケースは要件毎で分けて作成

■ケース⑤ 木を見て森を見ず

・リスク

要件通りの実装にならない
UI/UXの低下

・やるべきこと

森を見ることが必要
常に森を意識しながら実装 (&テスト)を進める

・必要な能力

俯瞰力

・QAチームのアプローチ

実際の運用を想定したシナリオテストの実施等、全体の流れを確認出来るようにする

■ケース⑥ スプリントのベロシティに無理がある

・リスク

無理なペースでの開発による品質の低下
結果リリースの遅れ

・やるべきこと

最初は手探りな部分があるので仕方が無いがある程度固まってきた際に無理がある場合は見直す

・必要な能力

決断力、調整力

・QAチームのアプローチ

リファインメント等の開発側、PO側共に集まっている会議体で見直しの提案

■ケース⑦ 開発側のストーリーポイントの見積もりが甘い

・リスク

リリースの遅れ

・やるべきこと

最初は手探りな部分があるので仕方が無いがある程度固まってきた際に無理がある場合は見直す

・必要な能力

分析力

・QAチームのアプローチ

疑問がある場合はストーリーポイントの妥当性を確認

■ケース⑧ 手段が目的になってしまっている

・リスク

本来のメリットである、AgilityやFlexibilityやAdjustability が機能しない

・やるべきこと

あくまでも手段であることを念頭に置き、上手くいってない場合はどの部分が上手くいっていないのかを分析し、本来の目的を達成出来るように改善に向けて調整を行う

・必要な能力

決断力、調整力、分析力

・QAチームのアプローチ

本来の目的に対して機能していないセレモニーがあれば改善を提案

■ケース⑨ NOと言えない日本人

・リスク

無理なペースでの開発による品質の低下
結果リリースの遅れ

・やるべきこと

NOと言える日本人になる。
日本人的美德の『我慢』や『忖度』はアジャイル開発の現場では封印する

・必要な能力

愛と勇気

・QAチームのアプローチ

愛と勇気

◆アジャイル開発プロジェクトを成功に導くために

【結論】

～結局は判断力、調整力、決断力 (総じて**コミュニケーション力**) が一番大切である～

WF開発と違い、その時々状況に応じて仕様を変更したり追加したりできるのもアジャイル開発のメリットの一つであり、状況に応じたスケジュールの調整や方針の修正等は必然的に発生する。これらの流動的な変化に対して、最適な判断が出来ることが成功の鍵となる。

その為には**密な情報交換ができるような環境、関係を構築しチーム内の透明性を保つことが大切である。**
(逆にいくらスキルの優秀なメンバーが集まったチームだとしても、お互いに連携や協調がなければアジャイルPJを成功に導くのは難しい。)

愛と勇気をもってお互いをリスペクトし、ONE TEAMでプロジェクトを成功に導きましょう！
ヒューマンクレストのQAチームならありったけの愛と勇気でこちらが実現できます！！

ご清聴ありがとうございました！

JaSST 第2事業部スライド

QAが見るべき品質のスコープとは。

～現代のQAの作業スコープの広がり～

Chapter1

【一部のシステムだけテストしても品質は確保できない。】

- ・フロントエンドから倉庫までの一気通貫テスト開始への道のり

Chapter2

【QAがテストしても、現場の人間が使えなければ品質は保てない。】

- ・店舗連携ECサイトリニューアルの大規模オペレーションテスト
- ・決済アプリリニューアルによる全関係部門でのUAT

Chapter3

【開発が分からないことでもQAが品質を確保しなければならない。】

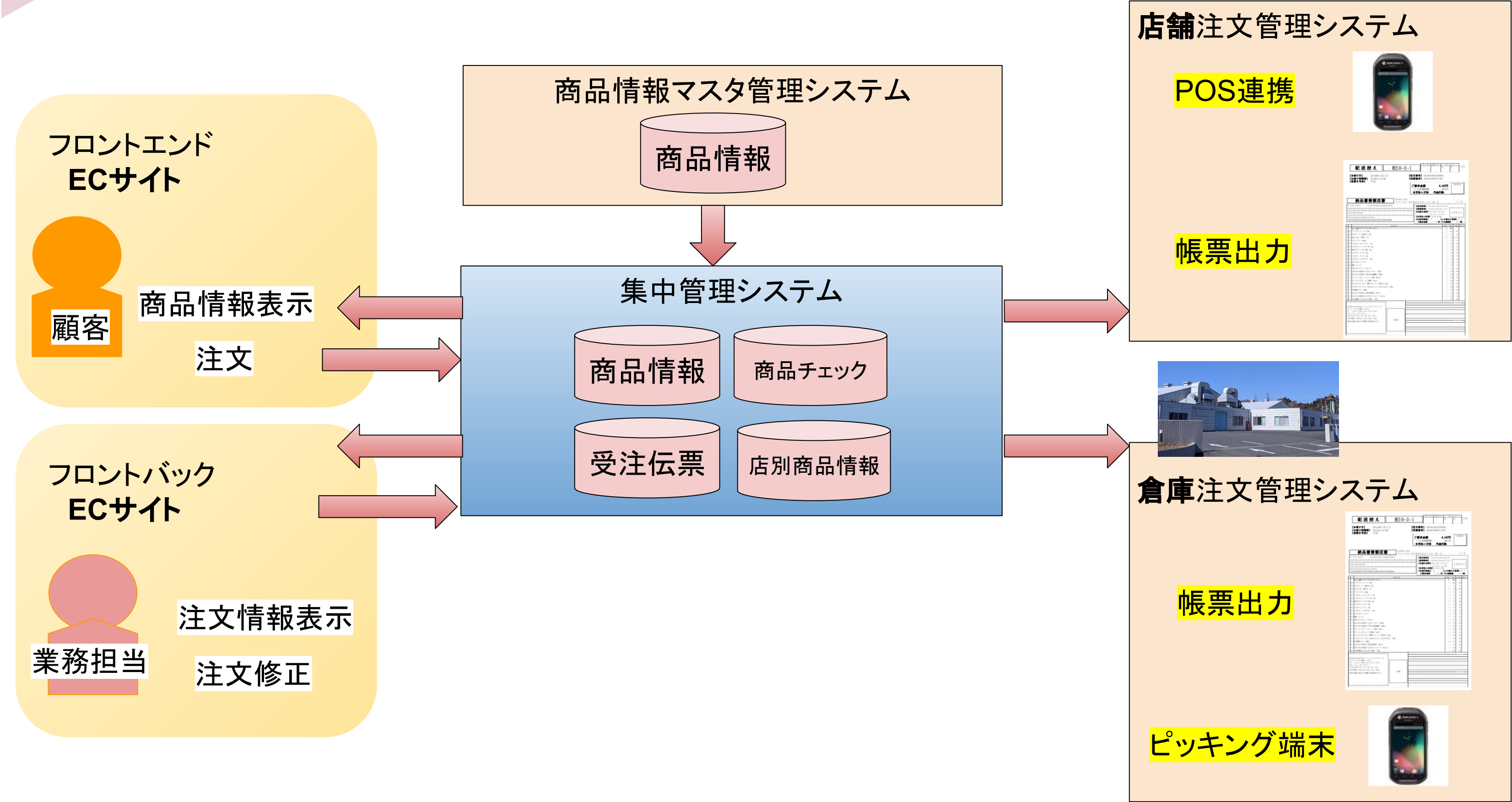
- ・開発と疎通と手法を探りながらのAWS移行

Chapter1

【一部のシステムだけテストしても品質は確保できない。】

・フロントエンドから倉庫までの一気通貫テスト開始への道のり

ネットスーパーシステム全体図



ネットスーパーシステム全体図

開発A社

フロントエンド
ECサイト



顧客

商品情報表示

注文

フロントバック
ECサイト



業務担当

注文情報表示

注文修正

開発B社

商品情報マスタ管理システム

商品情報

集中管理システム

商品情報

商品チェック

受注伝票

店別商品情報

開発B社

店舗注文管理システム

POS連携



帳票出力



開発C社

倉庫注文管理システム

帳票出力



ピッキング端末



ネットスーパーシステム全体図

QA A社

フロントエンド
ECサイト

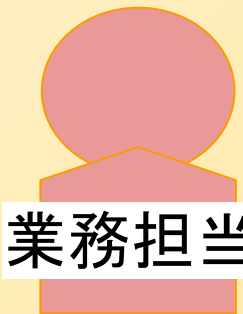


顧客

商品情報表示

注文

フロントバック
ECサイト



業務担当

注文情報表示

注文修正

humancrest社

商品情報マスタ管理システム

商品情報

集中管理システム

商品情報

商品チェック

受注伝票

店別商品情報

humancrest社

店舗注文管理システム

POS連携



帳票出力



QAチーム無し

倉庫注文管理システム

帳票出力



ピッキング端末



開発C社

倉庫側のテストは考えていない。
I/F仕様書を元に実装しているので障害が発生する事は想定していない。

STEP1 倉庫へ入場

①倉庫の場所確認

②倉庫への行き方確認

③倉庫への入り方確認

④倉庫のネットワーク状況確認

⑤倉庫入場許可取得



STEP2 倉庫開発チームとの連携

①倉庫内確認

②倉庫システム構成確認

③倉庫内OTスケジュール確認

④倉庫に影響が出ないテスト手法選択

⑤フロントエンド側とのテスト内容調整

⑥倉庫開発チーム説得



STEP2 倉庫開発チームとの連携

①倉庫内確認

②倉庫システム構成確認

③倉庫内OTスケジュール確認

④倉庫に影響が出ないテスト手法選択

⑤フロントエンド側とのテスト内容調整

⑥倉庫開発チーム説得



倉庫内オペレーションテスト(9:00~17:00) 2交代制

テスト期間=オペレーションテスト期間

OT期間

お届け時間	受注 締め 時間	出荷 締め 時間
10:00~12:00	0:00	9:30
12:00~14:00	0:00	11:30
14:00~16:00	9:00	13:30
16:00~18:00	11:00	15:30
18:00~20:00	13:00	17:30
20:00~22:00	15:00	19:30

倉庫側ピッキング教育にて占有

倉庫内オペレーションテスト(9:00~17:00) 2交代制

テスト期間=オペレーションテスト期間

OT期間

お届け時間	受注 締め 時間	出荷 締め 時間
10:00~12:00	0:00	9:30
12:00~14:00	0:00	11:30
14:00~16:00	9:00	13:30
16:00~18:00	11:00	15:30
18:00~20:00	13:00	17:30
20:00~22:00	15:00	19:30

倉庫側ピッキング教育にて占有

18時以降の発送便であれば
テスト用の注文で占有可能

18時以降発送の5便、6便での発送にてテスト

STEP2 倉庫開発チームとの連携

①倉庫内確認

②倉庫システム構成確認

③倉庫内OTスケジュール確認

④倉庫に影響が出ないテスト手法選択

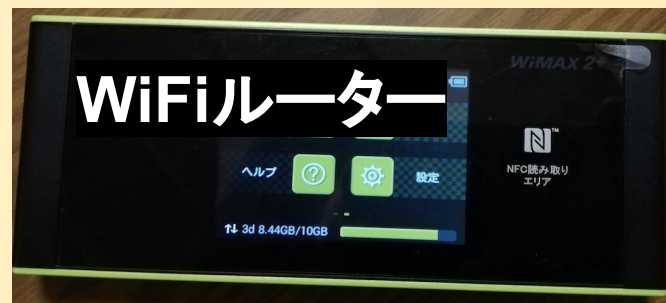
⑤フロントエンド側とのテスト内容調整

⑥倉庫開発チーム説得



STEP3 全体テスト準備

①テスト用Wifiモバイルルーター確保



②倉庫、本部とのリモート接続環境構築



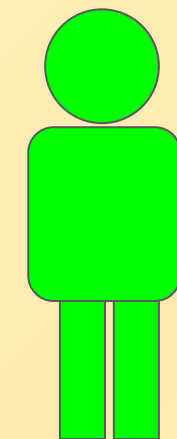
③倉庫連携用フロントエンドテストパターン修正

STEP4 全体テスト実施

①倉庫側QA1名張り付き

②倉庫側障害発生レポート作成

③倉庫側障害修正確認



QA1名

障害発生時に倉庫側での伝票出力確認

バグトラッキングツールでの情報提供

フロントエンド、集中管理システムとの
障害原因切り分け

テスト実施結果

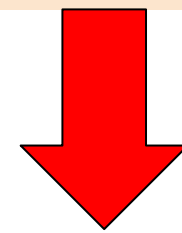
倉庫側システムの障害発生

- ・I/F仕様書記載のパラメーターの使用方法の誤認識
- ・消費税計算方法の算出ミス
- ・フロント側との仕様の相違による発送便の指定間違い。

テスト実施結果

倉庫側システムの障害発生

- ・I/F仕様書記載のパラメーターの使用方法の誤認識
- ・消費税計算方法の算出ミス
- ・フロント側との仕様の相違による発送便の指定間違い。



システム全体の**本番リリース5日前**に全ての障害が修正完了
ギリギリのタイミングでのテスト終了となった。

Chapter2

【QAがテストしても、現場の人間が使えなければ品質は保てない。】

- ・店舗連携ECサイトリニューアルの大規模オペレーションテスト
- ・決済アプリリニューアルによる全関係部門でのUAT

店舗連携ECサイトリニューアルの大規模オペレーションテスト

オペレーションテスト

新たに導入されるシステムを
現場の人間が使える様に
事前にトレーニングを行って
運用開始時に問題が発生しない事を
保証する事

・オペレーションテスト実施STEP

STEP1 各部門の代表選出

STEP2 各部門の基本業務フロー作成

STEP3 各フローで使用するシステムの確認

STEP4 各フローのテストケース作成

STEP5 オペレーションテスト実施、習熟度確認

STEP6 本番リリース

・オペレーションテスト実施STEP

STEP1 各部門の代表選出

STEP2 各部門の基本業務フロー作成

STEP3

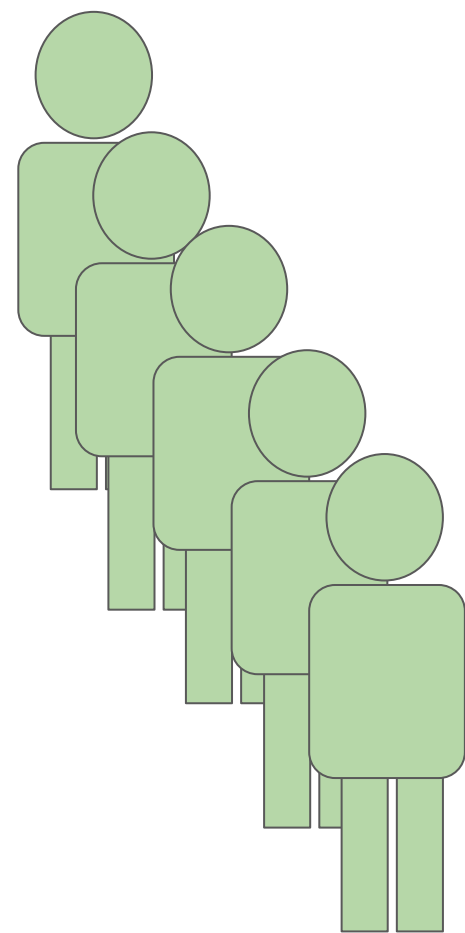
STEP4

**オペレーションテストは、
現場の人間が自分達で
全ての作業を実施する必要がある**

STEP5 オペレーションテスト実施、習熟度確認

STEP6 本番リリース

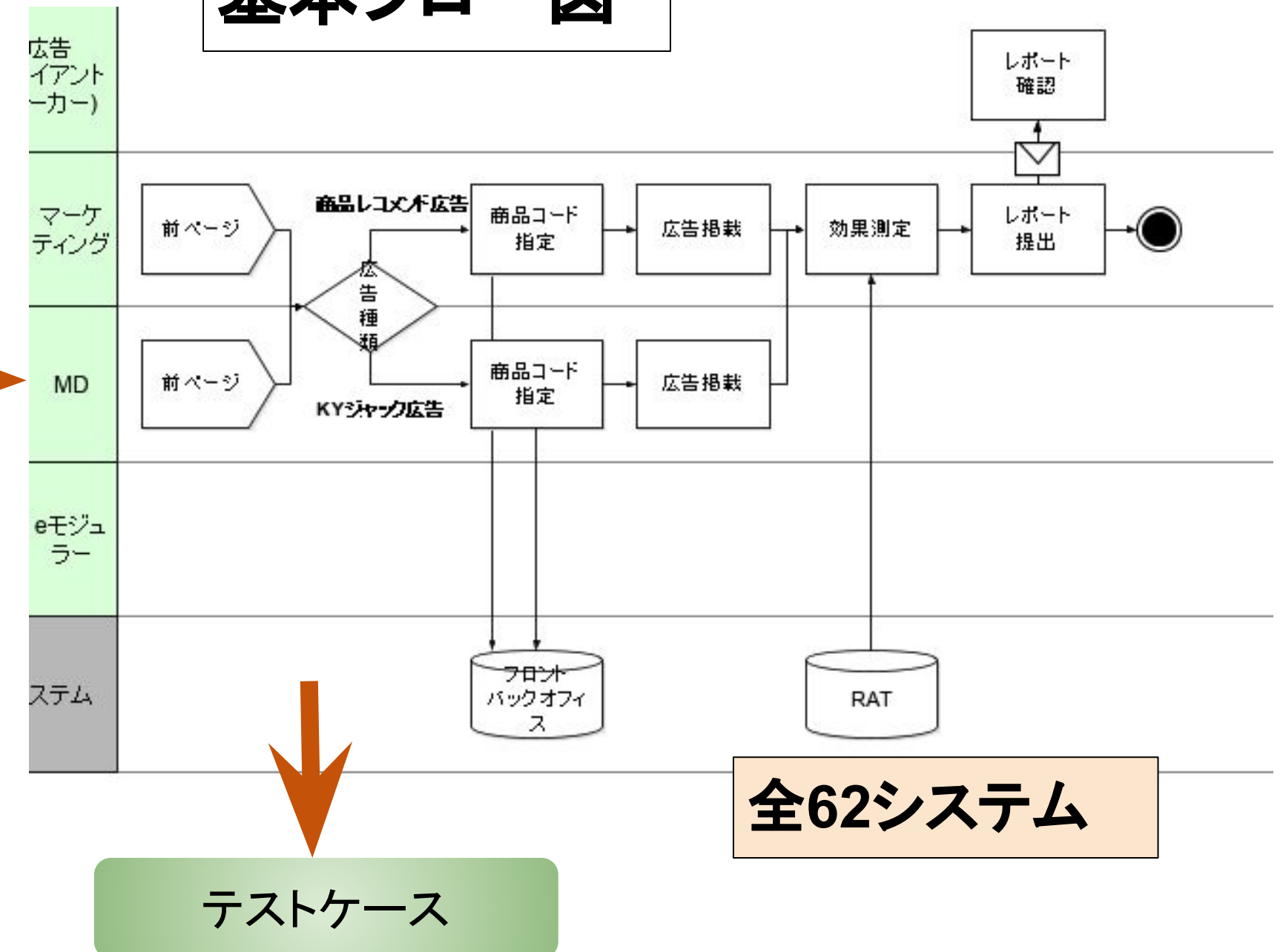
STEP2 各部門の基本業務フロー作成



商品部バイヤー
商品部モジュー
ECサイト事業本部
ECサイト商品戦略部バイヤー
ECサイト商品戦略部
法務部
人財部
コンプライアンス部
プライシングチーム
オペレーションチーム
編成チーム
取引先メーカー
マーケティング/編成部
CS(カスタマーサポート)部
MDシステム部

全24部門

基本フロー図



STEP3 各フローで使用するシステムの確認

STEP4 各フローのテストケース作成

OT対象部門

対象システム

全24部門

全62システム

フローから作成された
オペレーションテストケースオペレーションテスト
120シナリオ丸が入っている部門がOT
実施対象

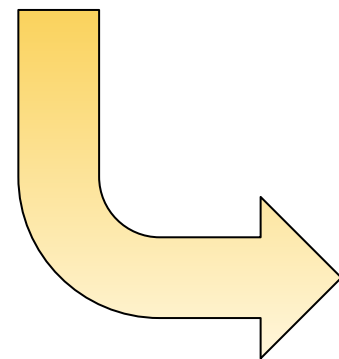
オペレーションテスト実施



本番リリース(ECサイト稼働開始)



ECサイト稼働初日から運用に問題発生せず。



QAチームがテストだけではなく、
ビジネスサイドの運用をOTでサポートする事は
品質のスコープを広げる良い経験になった。

決済アプリリニューアルによる 社内関係者のUAT(受け入れテスト)立案と実施

○UAT実施STEP

STEP1 関連メンバー選出、説明会場確保、実施期間

STEP2 POSレジ設置、レジ使用方法の習得

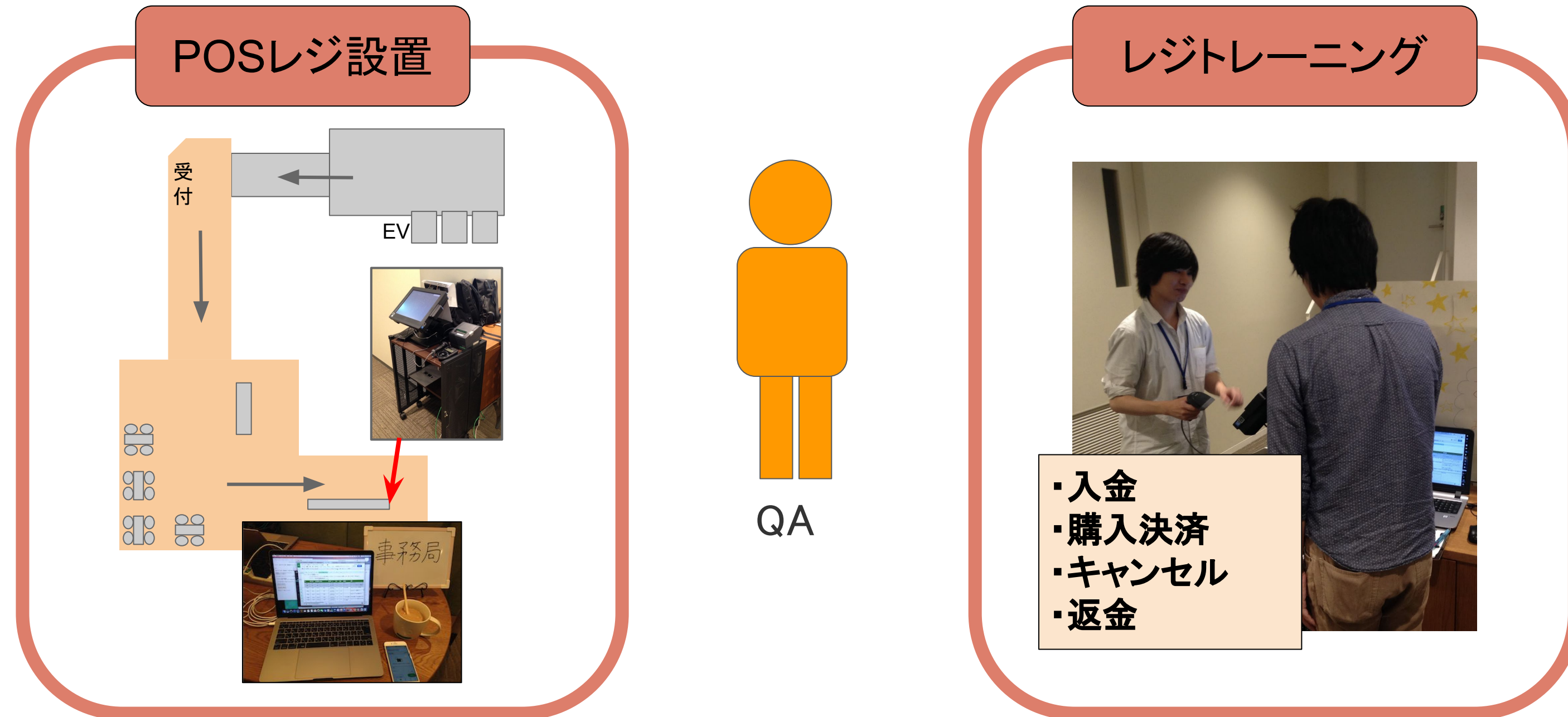
STEP3 説明会資料作成、資料配布、テストアプリの配布

UAT実施

STEP4 フィードバック報告、Q&A収集、Q&Aマニュアル作成



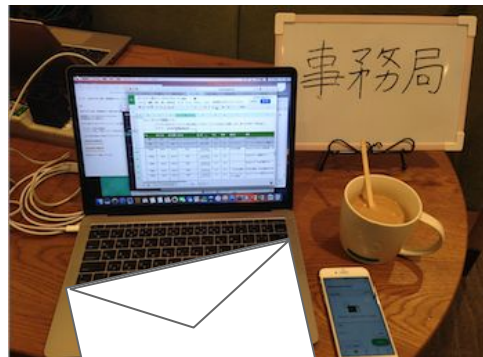
STEP2 POSレジ設置、レジ使用方法の習得



STEP3 説明会資料作成、資料配布、テストアプリの配布

アプリ配布

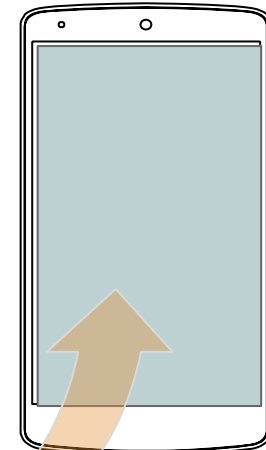
buddybuild



UAT事務局 手動送信メール
(インストールURL)

Android/iOS

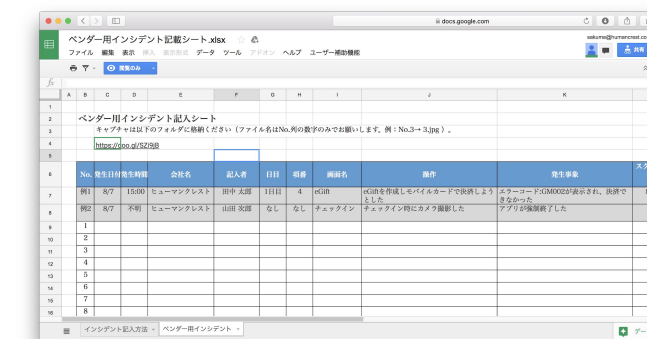
端末ブラウザよりインストール



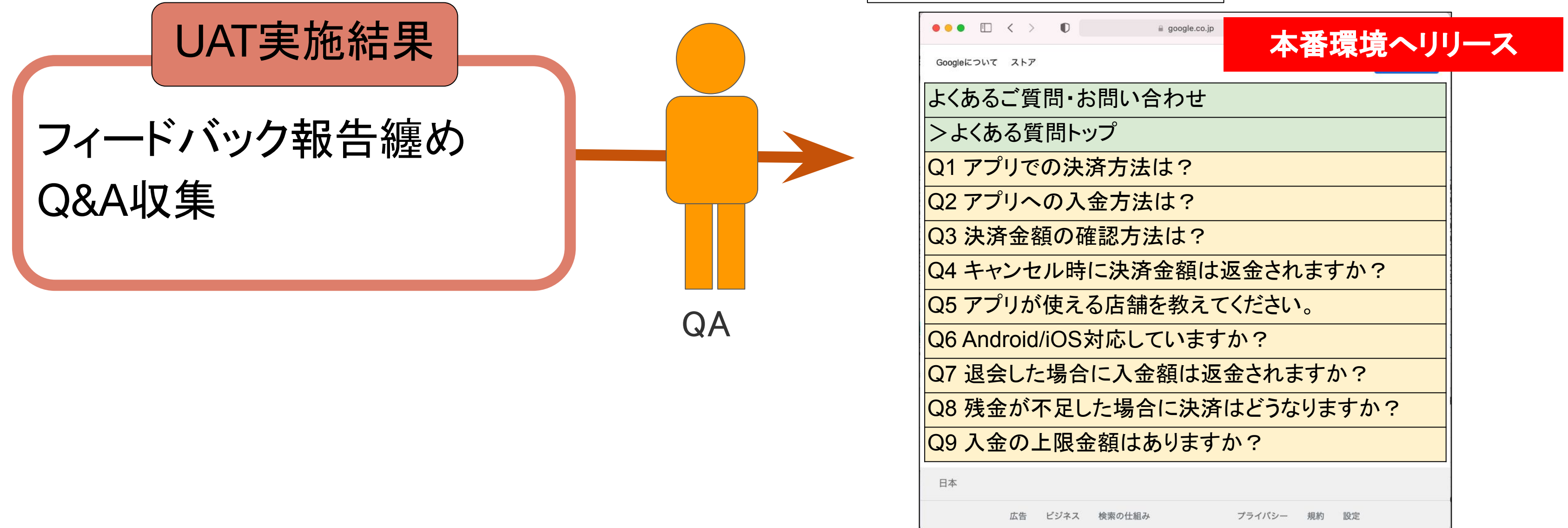
QA

資料作成/配布

- ・説明会スライド
- ・Googleフォームアンケート
- ・フィードバックシート



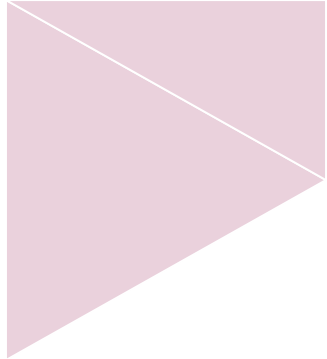
STEP4 フィードバック報告、Q&A収集、Q&Aマニュアル作成



Chapter3

【開発が分からないことでもQAが品質を確保しなければならない。】

- ・開発と疎通と手法を探りながらのAWS移行



オンプレサーバーAWS移行テスト

・オンプレサーバーAWS移行テストとは……

STEP1

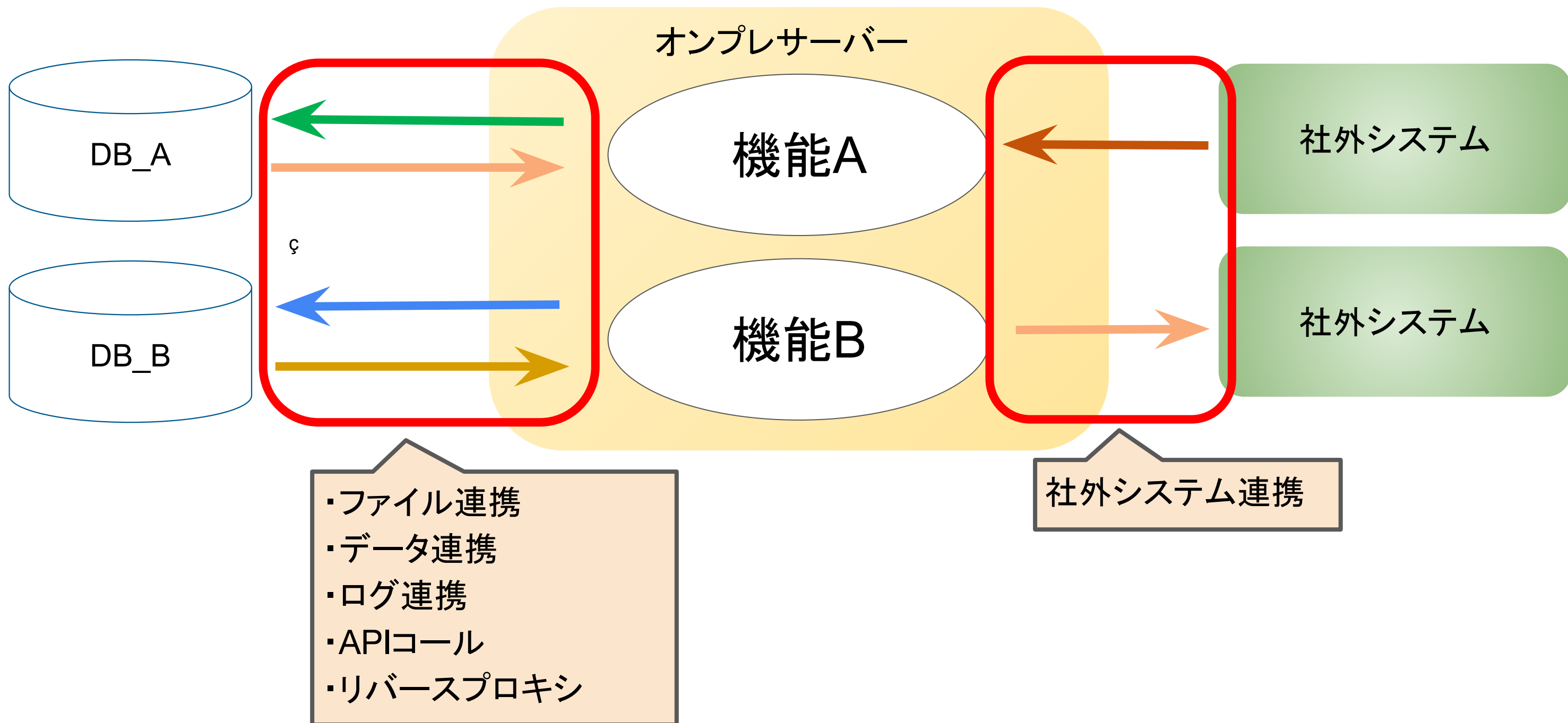
既存のシステムのアーキテクチャ図を元に
疎通確認が必要なシステム連携の疎通リストを作成する。

STEP2

開発がオンプレサーバーをAWSに移行させた後に、
必要な疎通確認をQA側で実施する。

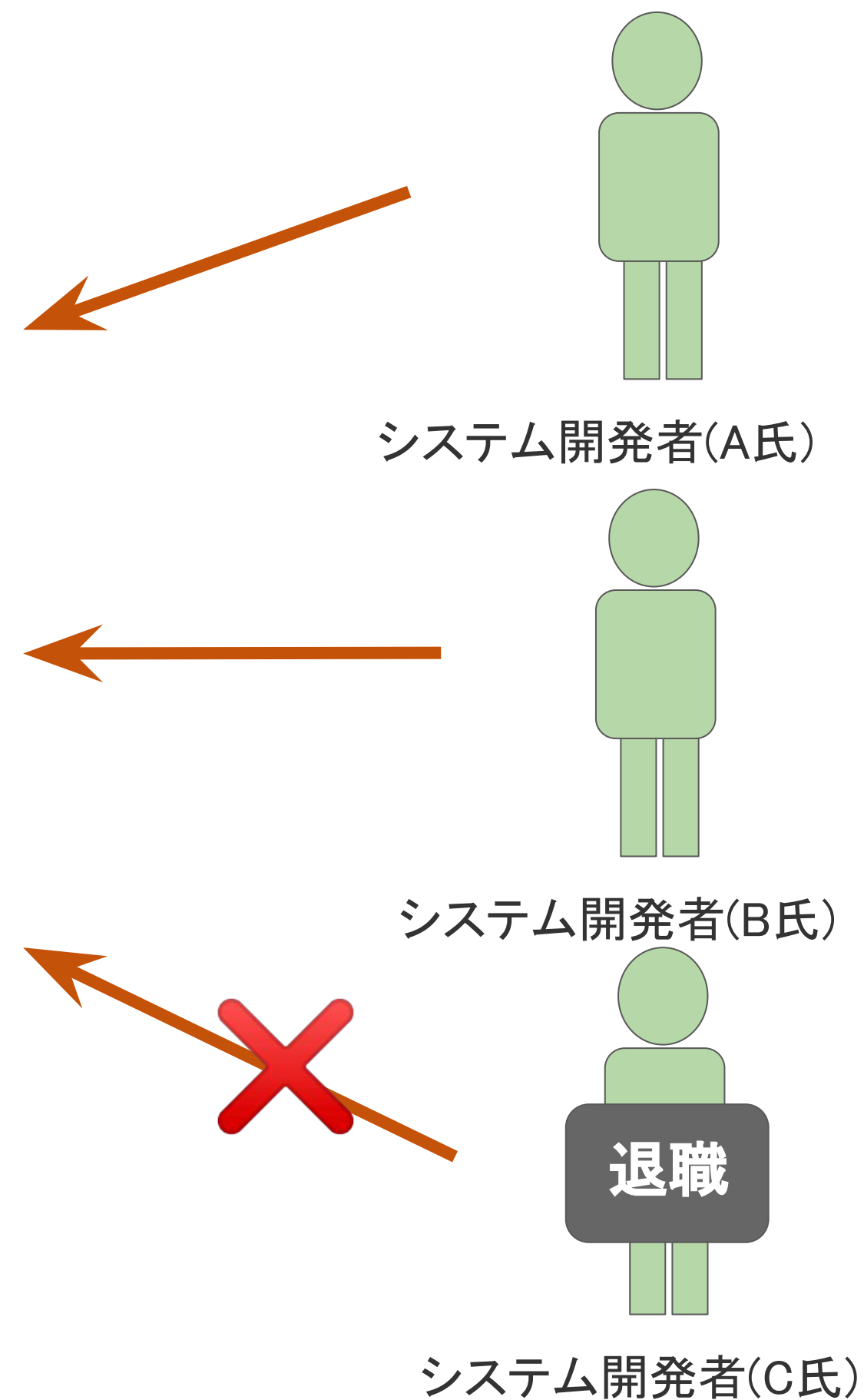
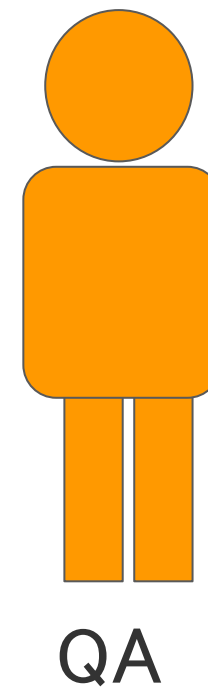
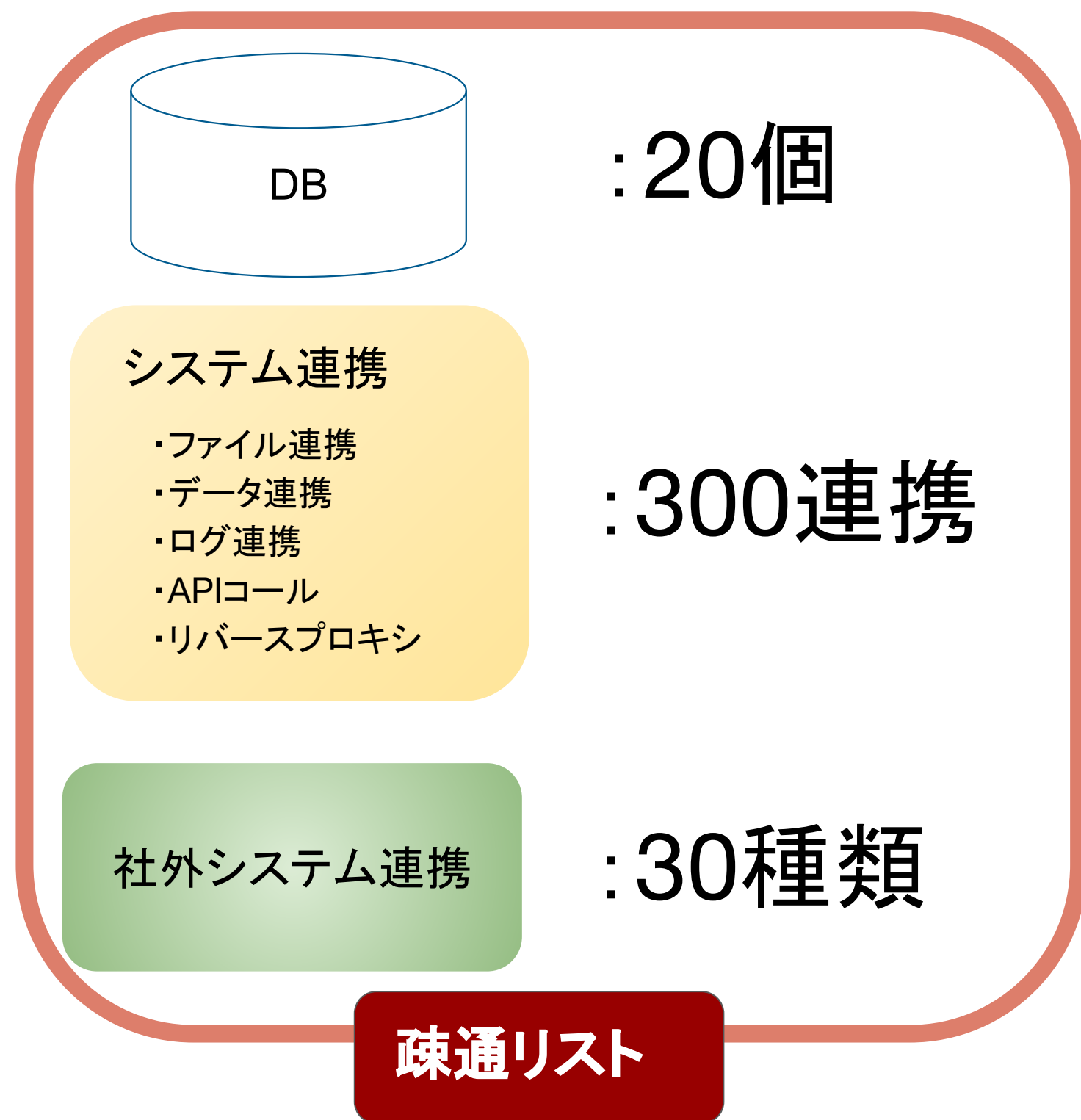
- ・疎通確認が取れることが目的なので連携するデータパターンなどのバリエーションでの確認は不要。
- ・AWSテスト環境で疎通確認を実施した場合でも本番環境で疎通の保証は出来ない。

・システム連携の疎通リスト作成

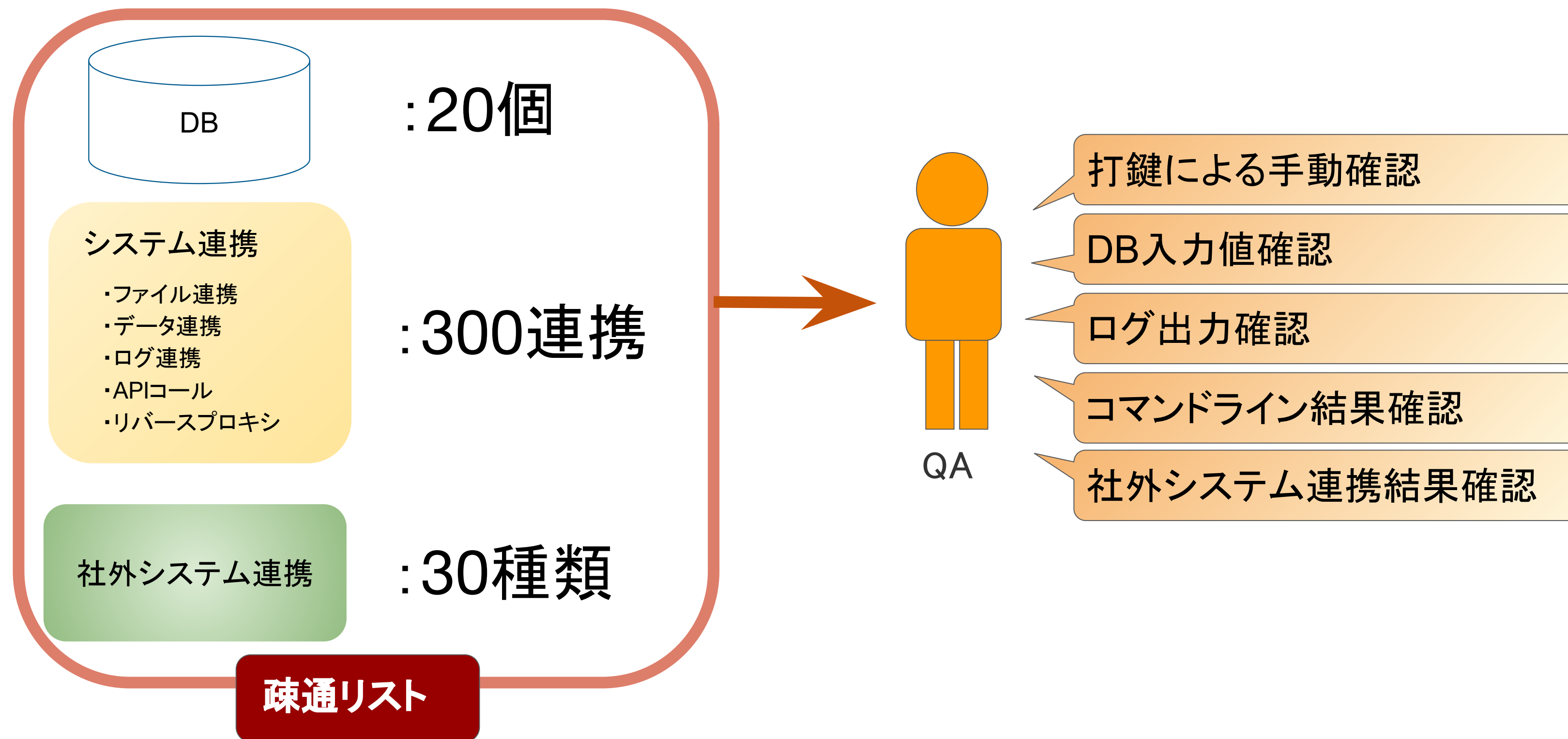


各サーバー内外、各機能の連携や通信の内容を把握しつつリスト化する地道な作業

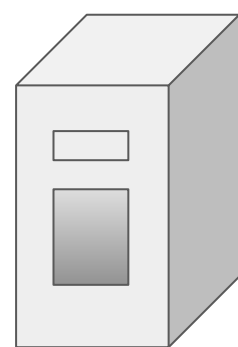
・大規模システムの疎通リスト作成



・疎通リストから各疎通毎のテスト手順を策定

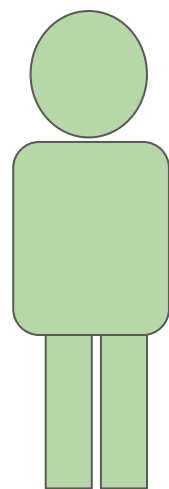


・外部システム連携機能の切替/テストは難易度高

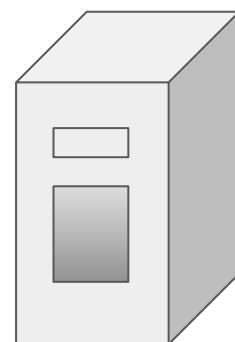


社外システム連携

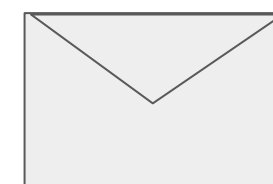
: 30種類



担当者不在

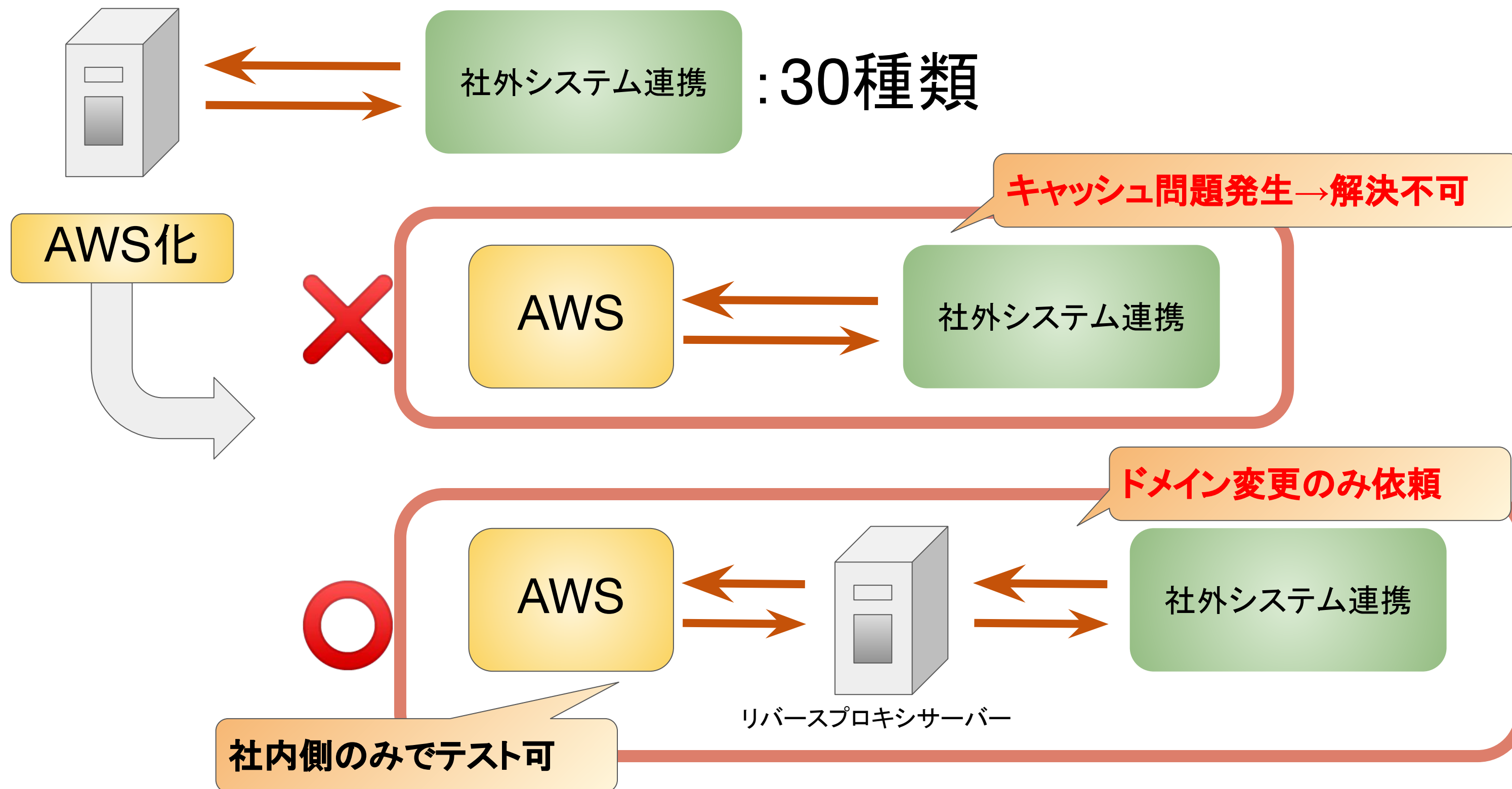


社外システム側
修正不可



メール返答遅延

・テスト/運用が外部要因に依存しないシステム設計



2021.03.15

JaSST'21 Tokyo セッション B4

外部テストベンダーとしての QA組織立ち上げ支援の取り組み

Human Crest Group

株式会社ヒューマンクレスト

第3QuaritySolution事業部 米田 徹

1.はじめに

本日はお話しすること

- ・ 外部テストベンダーとしての最近の引き合い傾向
- ・ QA組織立ち上げ支援について
- ・ やってみてよかった取り組みの共有

2.自己紹介

名前

- ・ 米田 徹（まいた とおる）
- ・ 第3QuaralitySolution事業部

経歴

- ・ 2014年ヒューマンクレスト中途入社
- ・ 2014-2016 ソシャゲアプリのQA
- ・ 2016-2018 保険システムのQA
- ・ 2018-2019 採用サービスのQA,社内の各PJのサポート
- ・ 2019- 各PJの立ち上げ,サポート

3.やっていること

最近の引き合い傾向

- ・ コロナ禍でお客様のサービス需要が拡大
- ・ 開発をスピードアップさせたい
- ・ エンジニアが手動テストまで行っているのが負担
- ・ テストできる人いませんか？

QA組織立ち上げ支援

- ・ 新たにQA組織を立ち上げるときのお手伝い

3.やっていること

ポイント

- ・テストの代行ではなく、QAとしての価値を提供したい
- ・テストがボトルネックになってはダメ
- ・メンバーをアサインして終わりではなく、社内から後方支援する

取り組み方

- ・QAメンバーを素早く立ち上げて、テストを巻き取る
- ・テスト以外でも価値を発揮する
- ・開発スピードと品質の両立に寄与する

3.やっていること

QAメンバーの支援

<導入時>

- ・テストフローの構築
 - ・これまでのテスト戦略の確認、見直し
 - ・現場にフィットしたやり方の模索
- ⇒お客様からヒアリング、パイロットでの進め方を策定

<実施後のフォロー>

- ・実施結果の整理
 - ・振り返りを実施
- ⇒結果のフィードバック、次案件以降の進め方の策定

3.やっていること

絶賛試行錯誤中

- ・ どんな取り組み方が効果的か事例収集中
- ・ 提供するサービスと収益

4.やってみてよかったこと①

不具合分析

4.やってみてよかったこと① 不具合分析

やること

- ・テスト実施後、不具合を分類する
 - ・重要度、混入した工程、原因、機能など
- ・プロジェクト状況を踏まえて分類結果を眺める

4.やってみてよかったこと① 不具合分析

効果

- ・ 品質状況の見える化
- ・ 次の打ち手が分かる
 - ・ 既存のバグ
 - ⇒QA目線で初めて検出。「探索的テストやります」
 - ・ 要件漏れ
 - ⇒テスト設計時に検知。「仕様策定段階から参加します」
 - ・ 仕様認識違いによるバグ
 - ⇒新しいエンジニアが慣れてない。「レビューしてます？」
 - ・ 複数機能同士の干渉
 - ⇒影響確認が必要。「リグレッションテストします」
- ・ 状況に応じた効果的なテスト

4.やってみてよかったこと②

ユーザー目線での気づき

4.やってみてよかったこと②ユーザー目線での気づき

やること

- ・テストと並行して気づきをリスト化
 - ・次に何をすればいいか操作が迷う
 - ・表記ゆれがあって分かりづらい、など

4.やってみてよかったこと②ユーザー目線での気づき

効果

- ・ 作っているとなかなか気づかない
 - ・ 無意識のうちに慣れてしまう
- ⇒ユーザー離脱、操作ミスによる問合せ、の回避

4.やってみてよかったこと③

OneTeamテスト

4.やってみてよかったこと③ OneTeamテスト

OneTeamテストとは

- ・ 職種の垣根を超えて、チームみんなでテストをする
- ・ テスト期間中、みんなで1h程度フリーテストする
- ・ 不具合の検知だけでなく、気づきや感想もOK

4.やってみてよかったこと③ OneTeamテスト

やること

<準備>

- ・テスト計画（目的、テスト対象。簡単な内容でOK）
- ・操作方法（手順メモや動画）
- ・ユーザーデータ、アカウント
- ・気になったところスプレッドシート

<実施>

- ・多忙なメンバーがいる場合は3h程度枠を用意、その間で各自任意の1h実施、という形で実施

<実施後>

- ・挙がった指摘は、リリースまでに対応するもの、リリース後対応するものなどに分類して管理

4.やってみてよかったこと③ OneTeamテスト

効果

- ・ 各職種目線での不具合検知、気づき報告が可能
- ・ それによってテスト観点も増える
- ・ チームが自分の担当箇所以外に触れる機会ができる
- ・ チーム全体のサービス理解や品質へ意識が高まった
(気がする)
- ・ コスパがいい

ご清聴ありがとうございました