

Quality Control Policy



品質管理ポリシー Ver 2.2

～クロス・コミュニケーションにおける品質向上に向けた取り組みについて～

Introduction



クロス・コミュニケーションにおける品質の考えについて

クロス・コミュニケーションにおいては、
次ページ以降記載の通り品質向上に向けた各種取組を推進しております。
これまで多くのシステム開発にて得たノウハウをもとに、更に品質を安定化させ、
より多くの方に安心して利用いただけるモノづくりを目指します。

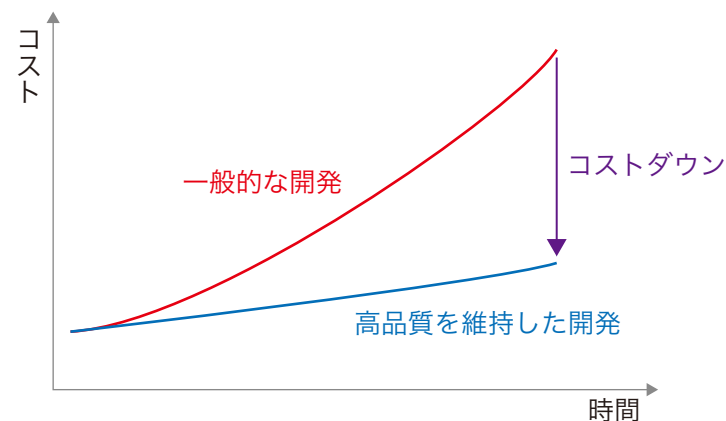
1 品質を高めることのメリット

システム構築後の保守・運用期間が長くなるにつれ、追加・改修コストやリスクが増える傾向が多くみられます。計画段階から、品質計画を立て、結果として品質が向上すると中長期でコストを抑えられ、使いやすいシステムになります。

中長期視点によるコスト削減の実現

【主な理由】

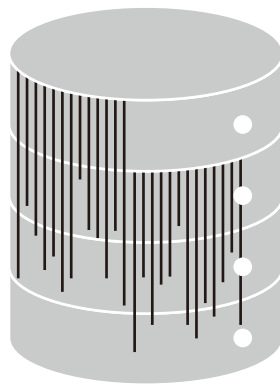
- ・ シンプルな設計により追加開発や改修が容易になる
- ・ テスト自動化によりコスト削減とヌケ・モレ・人的ミスを防止
- ・ レビューを通して双方の認識齟齬を防ぐ
- ・ 属人化を排除
- ・ ビルド・デプロイ・リリースを自動化し人為的ミスを減らし、余計なコストを削減



ビジネススピードの向上やリスク削減を実現

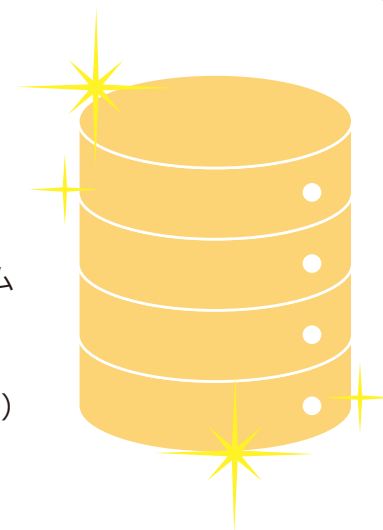
品質が悪いシステム

- ・ 追加・改修にコストがかかる
- ・ 陳腐化によるリスクの増加
- ・ リリース時のトラブル
- ・ 使いにくい、ユーザ評価が低い



品質が良いシステム

- ・ 追加、改修がしやすく
ビジネススピードを加速
- ・ リスクが少ない安定したシステム
- ・ スムーズにリリース
- ・ 魅力的品質が高い
(使いやすく、ユーザ評価が高い)



以下記載のような各種課題事項に対して、クロス・コミュニケーションでは適切な改善施策を行い、品質向上の実現を致します。

課題事項

- ① 開発を委託したが、不具合が多く品質に不満足な結果になった…
- ② 要求に対して意図していないシステムが出来上がってきて困っている…何が原因なのかわからない
- ③ 機能は要求どおりだが、システムに合った性能になっていないため、結局使い勝手が悪い…
- ④ コミュニケーションがうまくいかず、開発が進んでから認識違いが発覚し、大幅なロスが発生してしまった…
- ⑤ 開発がベンダー任せになっており、開発プロセスや納品物の品質レベルが見えず不安だ…
- ⑥ 事業として新たな領域のアプリ開発を依頼したいが、どうすれば望む品質になる要求を出せるのかわからない…
- ⑦ 開発側とのレビュー会を開き、事前に品質をチェックしていきたいが、どうレビューするかの観点・ノウハウがない…
- ⑧ QCD のバランスがわからない…開発側でコントロールしてほしい
- ⑨ 自社開発のシステムを検証・試験しているが、効果的な方法がわからず、余計なコストが増え困っている…

改善施策

PART. 1

《プロジェクト計画》

① ② ④ ⑤ ⑧

- ・プロジェクト計画
- ・マネジメントプロセス
- ・円滑なコミュニケーション
- ・出荷判定基準とレビュー

PART. 2

《品質監査》

⑥ ⑦ ⑨

- ・テスト計画
- ・お客様視点で早期発見
- ・開発資源の一元管理
- ・ソースコード管理
- ・自動化

PART. 3

《非機能要件》

③

- ・非機能要件の明確化
- ・インフラ環境
- ・データの安全性と保全性
- ・セキュリティ対策について

当たり前品質及び魅力的品質の実現

3 品質とは？ 品質特性と非機能要件の比較

機能要件だけでなく、非機能要件（セキュリティや保守・運用）を考慮します。
非機能要件と ISO で定義されている品質特性は下記のような親和性があります。

品質特性（ISO/IEC25010）と非機能要件

品質特性	概要	対応する非機能要件
使用性	利用者の使いやすさ UI/UX	—
機能適合性	実装された機能がニーズを満たす度合い	—
セキュリティ	不正にアクセスがされることなく、 情報やデータが保護される度合い	セキュリティ
性能効率性	システムや実行時の性能や資源効率の度合い	性能・拡張性
互換性	他製品やシステムと機能や資源効率の度合い	性能・拡張性
信頼性	必要時に実行することができる度合い	可用性
保守性	効果的、効率的に保守や修正ができる度合い	運用・保守性
移植性	効果的、効率的に他のハードウェアや 実行環境に移植できる度合い	移行性

PART. 1 Project Plan



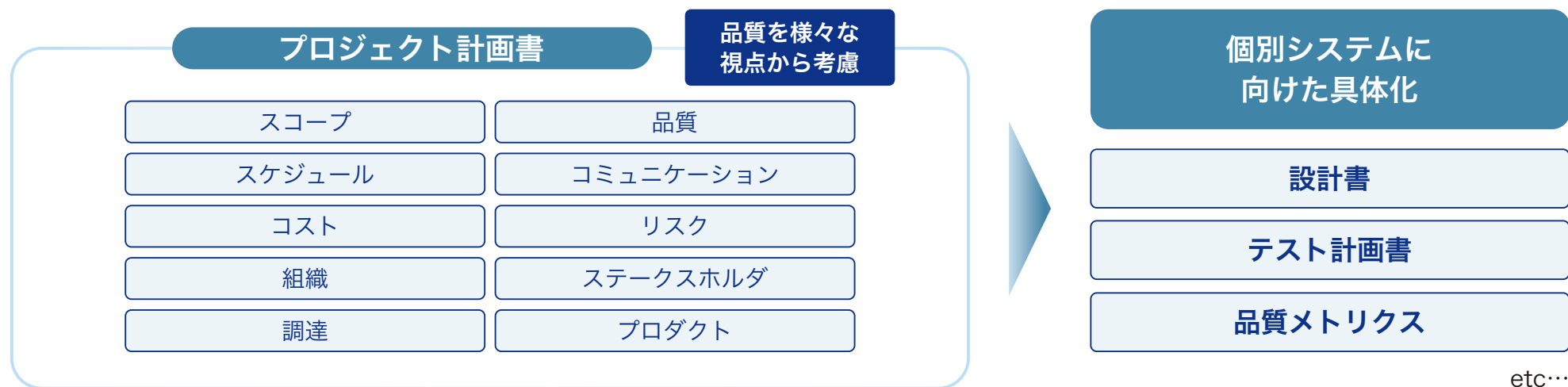
《プロジェクト計画》

～品質向上の実現に向けたプロジェクト計画策定～

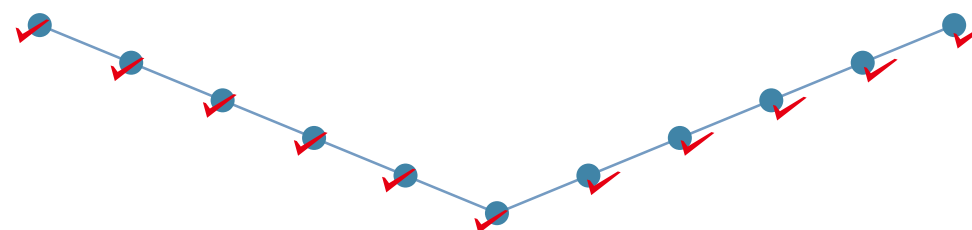
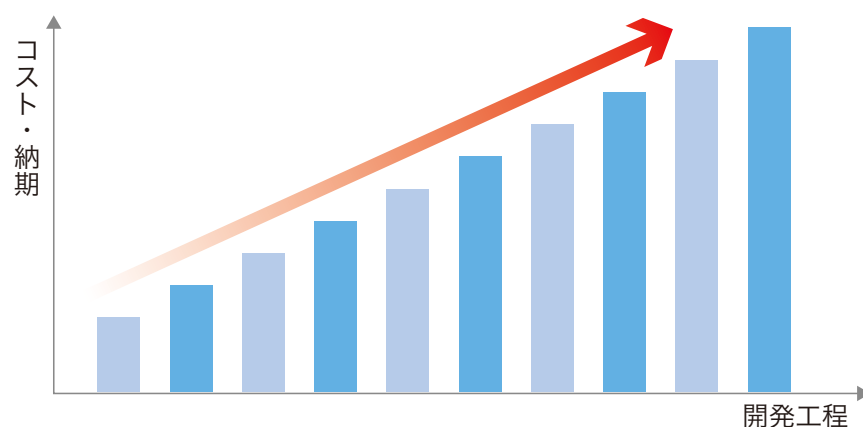
お客様にご満足いただける品質向上を実現すべく、
プロジェクト計画を立案します。

1 プロジェクト計画

「品質向上視点でプロジェクト計画」を立てることで、品質を担保するための活動を行います。



各工程ごとにレビューを実施し、計画で定めた品質基準が達成されているかを確認します。
リスクを早期に発見し、対策することで品質を担保しています。



工程毎に品質をチェックし、
コスト・納期の増大を防ぎます

※規模・内容によってはプロジェクト計画書を作成せずに進行させることもございます。

2 プロジェクト計画を実現するためのマネジメントプロセス

マネジメントプロセスを明確化することで品質向上を目指します。さらに各プロジェクト特性に合わせ継続的に改善します。

マネジメントプロセス	お見積り	標準化された項目に基づき、漏れや齟齬の無い精度の高く、妥当性のある見積もりをします。
	プロジェクト計画	プロジェクト計画を立て、コミュニケーションルールやステークスホルダを確認し、スムーズにプロジェクト開始ができる準備を進めます。
	要件定義・基本設計	標準ドキュメントを用い、評価基準による各種ドキュメントやプロセスのチェック・レビューを行い、漏れや齟齬の無い設計をします。
	製造・テスト	開発標準に基づいた実装・ソースレビューを行います。テスト計画に基づきテストを実施をします。
	出荷判定	プロジェクト計画時に出荷判定基準を定め、出荷判定を行います。
	リリース	リリース計画を立て、あらかじめ定めた手順によりリリース作業を行います。属人的なミスを防ぐよう自動化による対応も実施します。

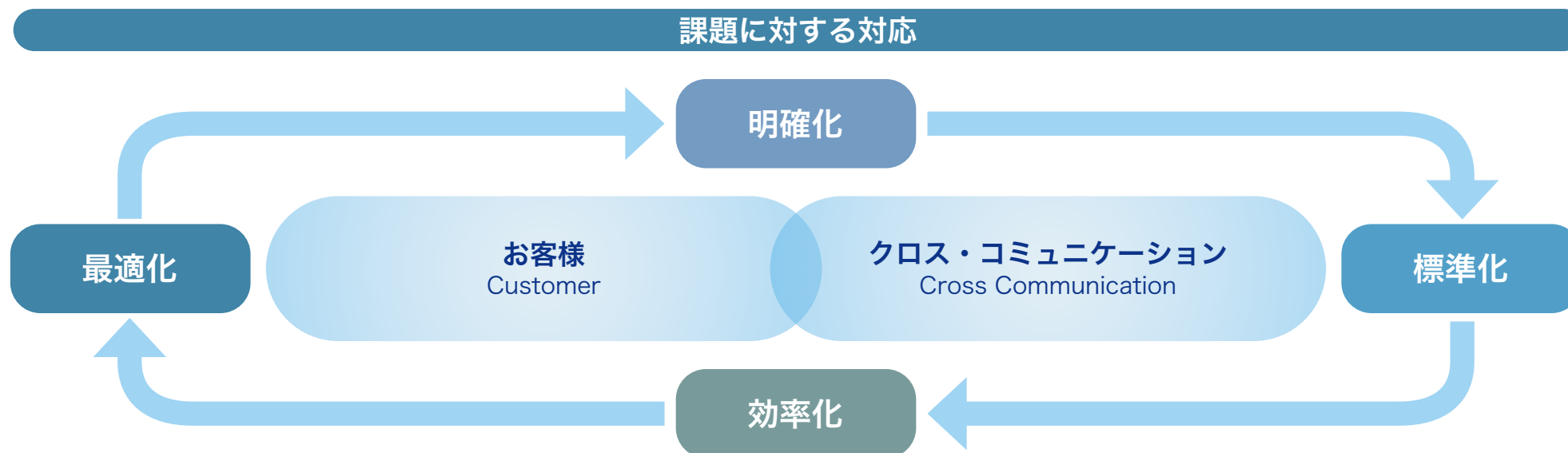
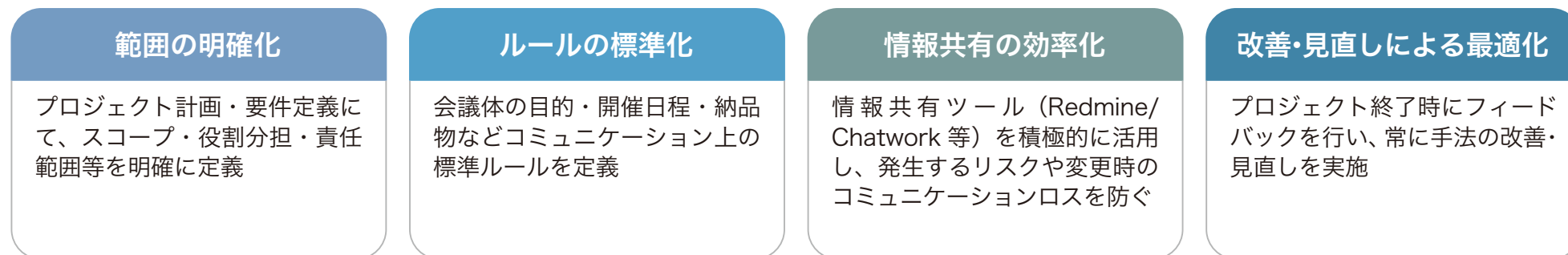
CI/CD（継続的インテグレーション / 継続的デリバリー）ツールを導入しビルド・デプロイ・リリースを自動化し、開発・保守プロセスの負荷軽減と属人化を排除します。



お客様・製品・取り巻く環境を考慮し、改善を継続的に実施します

3 円滑なコミュニケーションの徹底

コミュニケーション不足により認識齟齬が生まれます。円滑にコミュニケーションを図るため、以下4つの手段を定めたうえでプロジェクトを開始します。

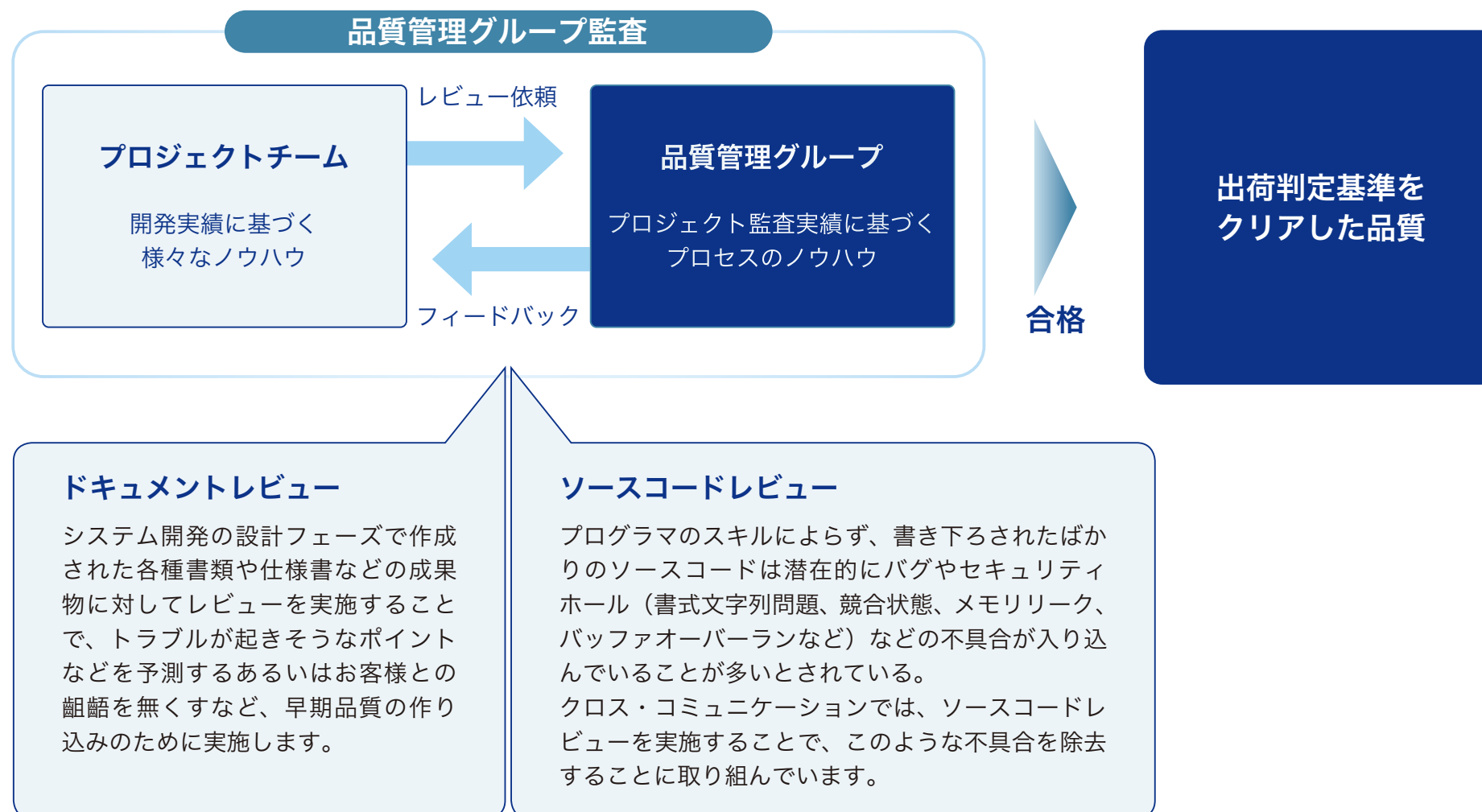


システム開発において、システム取得者側のステークホルダー間でシステム仕様やシステム化の範囲などを一旦合意したとしても、実現されたもの（α版やβ版）を見て初めて齟齬があったことが判明することが多々あります。

クロス・コミュニケーションでは、この課題に対応するため、要件定義プロセスにおいて「ワイヤーフレーム」「モック」「プロトタイプ」などの手法で、ステークホルダー間で発生する認識齟齬を早期に無くすことに取り組んでいます。

4 出荷判定基準を定め・レビューを実施

品質管理グループが顧客視点でレビューを実施し、
フェーズごとに品質に影響を及ぼすリスクを含んでいないかの確認を徹底しています。

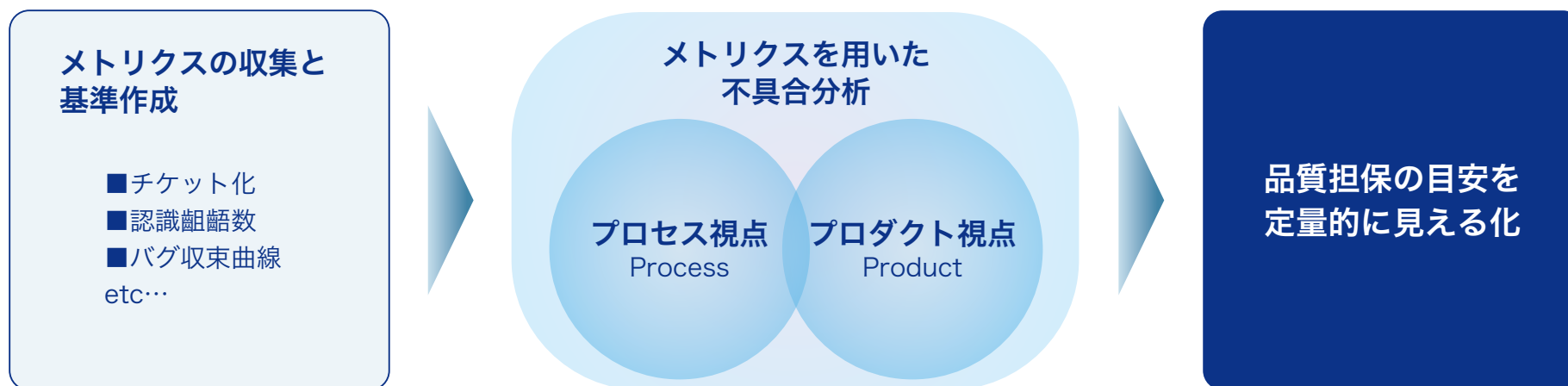


プロジェクト管理プロセスの周知・教育

レビューなどを通しプロセスの教育を実施し、組織として本プロセスの周知・徹底を図っております。
これにより、開発チーム全員が同じ認識のもとでプロジェクトに参画できるため、品質を安定させることができます。

品質メトリクスを用いて不具合状況を可視化・分析・対策

品質向上を効率的に実現するためには、メトリクスを用いた「品質の見える化」をします。
メトリクスを用いた不具合分析を行い、不具合対策や不具合発生抑制をします。



※メトリクス：グラフなどで可視化したもの。例えばバグ収束曲線を用いてテストケースの質・量の妥当性を検証したり、不具合発生が多い箇所を特定し、テストリソースを配分したりし、品質向上・改善に利用するものである。

リスクマネジメントの取り組み

リスクマネジメントは、トラブルを未然に予見しプロジェクトに与える損失を最小限に止めることを目指したもので、プロジェクト管理には欠かせない項目の一つです。
クロス・コミュニケーションでは、見積り段階からリスクの洗い出し・分析を行い、プロジェクト計画時に詳細化し、プロジェクトの完了まで定期的に継続することでリスクの管理を実現しています。

当社は、当社の顧客となる取引先企業様に対してソリューションサービス・コンテンツサービスを高い品質で提供することによって、顧客の事業活動に貢献することを目指しております。これらの業務を遂行するにあたり、一般の消費者、取引先、従業員の皆様から個人情報をご提供いただくことは非常に重要な事業活動と認識しており、それらの個人情報を保護する観点から、以下の通り「個人情報保護方針」を定め、これを実施し、維持し、かつ改善することを宣言いたします。

1. 当社は、当社の事業活動において取得・利用する個人情報および従業員等の個人情報の取り扱いに関し、個人情報の取り扱いに関する法令、国が定める指針その他の規範を遵守いたします。また、さらに個人情報の日本産業規格「個人情報保護マネジメントシステム—要求事項」（JIS Q 15001）に準拠した当社の個人情報保護マネジメントシステム策定・遵守し、個人情報を保護いたします。
2. 当社は、個人情報の取得・利用にあたっては、その利用目的を特定することとし、その利用目的、問い合わせ窓口をお知らせしたうえで、必要な範囲での取得および利用を行います。また、必要な範囲を超えた個人情報の取り扱い（目的外利用）を行わないために、適切な管理措置を講じます。
3. 当社は、取得した個人情報を適切な方法で管理し、ご本人の同意を得ている場合や法令に基づく場合等を除き、第三者に開示・提供することはありません。アンケート、キャンペーンの応募等を受けつける際に取得する個人情報は、個人情報を含まない形で集計されて当社の事業活動に利用させていただくほか、皆様への回答、連絡やプレゼントの送付に利用させていただきます。
4. 当社は、取得した個人情報に関し、不正アクセス、個人情報の漏えい、滅失または毀損の防止および是正に努め、厳正な管理のもとで安全に蓄積・保管します。
5. 当社は、委託元よりお預かりした個人情報を、厳正なる管理のもと、契約の範囲内で利用します。
6. 当社は、個人情報の取り扱いに関する苦情および相談を受けた場合は、その内容について事実関係等を調査し、合理的な期間内を持って対応いたします。また、ご本人様より、個人情報の確認、訂正等を希望される場合には、合理的な範囲ですみやかに対応させていただきます。
7. 当社は、社会情勢・環境の変化を踏まえて、個人情報保護マネジメントシステムの継続的改善に努めます。

2011年8月1日制定 2022年7月1日改訂
株式会社クロス・コミュニケーション 代表取締役社長 信太 孝一

PART. 2 Quality Audit



《品質監査》

～開発実績に基づくノウハウを用いた品質監査～

品質保証グループが開発標準を策定し
お客様目線で品質監査を実施しています。

1 品質監査の視点

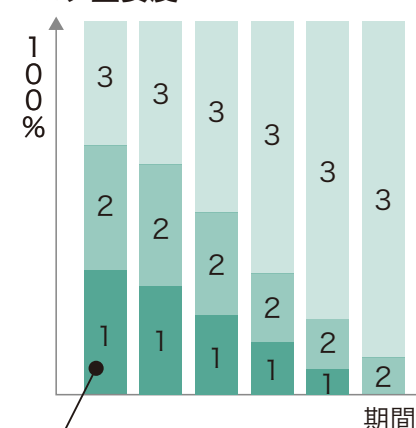
品質要求はプロジェクトごとに異なります。プロジェクト計画で品質管理基準を策定し品質メトリクスなどを活用し課題やリスクが可視化されているか監査します。

プロジェクト計画時に管理基準を策定

項目	内容	目標
品質 機能性	機能における認識距離を短くする	総合テストにおいて認識距離0.0件以下 開発時改修RT0.0日以下
品質 操作性	UI/UXにおける認識距離を短くする	総合テストにおいて認識距離0.0件以下 開発時改修RT0.0日以下
品質 効率性	画面表示・ファイル出力・パッチ処理速度	画面表示2秒以内 ファイル出力10秒以内 パッチ処理30分以内
品質 保守性	機能追加・改修	保守時機能追加・改修リードタイム0日以内
品質 信頼性	24/365監視 メンテナンス時間 稼働率	メンテナンス時間 2時間/月 稼働率99% 毎日スモークテスト (Selenium)
品質 移植性	既定移行時間	既定移行時間2日以内 (休日対応)
非機能要件	IPAフォーマットを利用し要件定義で定める	
バグ管理	累積バグ数 重要度バグ比率 画面再現バグ数 バグフィックスリードタイム推移	重要度が「高」のバグが残存していないこと テストケースの90%をパスしていること [現在から] 48時間以内に重要度「高」のバグが発見されていないこと 修正リードタイム 2営業日以内
テストアプローチ	ホワイトボックステスト・ユニットテストで実施 ※テストの書きすぎに注意 同時分組・境界値分析 (場合により) ディシジョンテーブル・状態遷移図など	実行範囲 (C1) カバレッジ 60-90% 目安 テストコード作成比率 30%以下 重要バグ数 0 テストケース実行数 99% テスト手法 (同時分組、境界値分析など) を明記する フォーマットを要件定義で合意する 顧客同意が得られているか
ドキュメント	顧客と弊社で共通認識を得るドキュメント 必要なドキュメントを必要に応じて作成する	コードレビュー・テストコードレビュー WARNING, ERROR, FATAL ゼロ 毎日マージリクエスト受付しdevelopブランチにマージ
ソース管理	linkを導入 フォーマットは全員共通 テストコードは必須	前日の予定管理 (予定と実績の乖離率)
タスク管理	Redmine 1画面表示1チケット バーンダウンチャート	プラグイン管理はcomposer,jsonをGitで共有し、composer installのみ利用 保証期間は要件定義で定義する
開発環境	PHP7.2 Nginx 1.15.1 Apache 2.4.34 MySQL 8.0.11 Laravel 5.6	開発環境、ステージング、本番環境全てDocker利用
インフラ環境	AWS コンテナ (Docker利用)	全て最新版を対象
サポート環境/バージョン	Win7, 10/Chrome, IE, Firefox, Mac/Safari, Chrome iOS/Safari Android/Chrome	ファイル配属、DBマイグレーション、データ投入などが自動化されていること
ビルド・デプロイ方法	Jenkinsで自動ビルド・デプロイ (CI/CD)	
その他	リリース後改修が少ないので保守チームに移管	

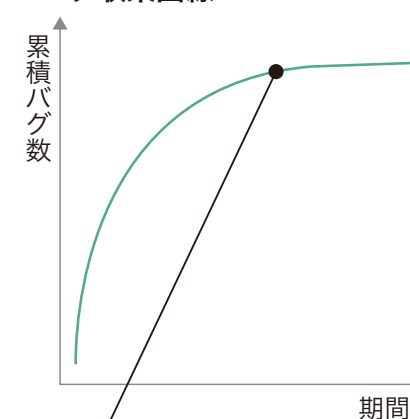
品質メトリクスの取得

■バグ重要度



重要度が高いバグを優先的に対応

■バグ収束曲線



寝てきたら安定稼働？

品質監査の視点

テスト計画

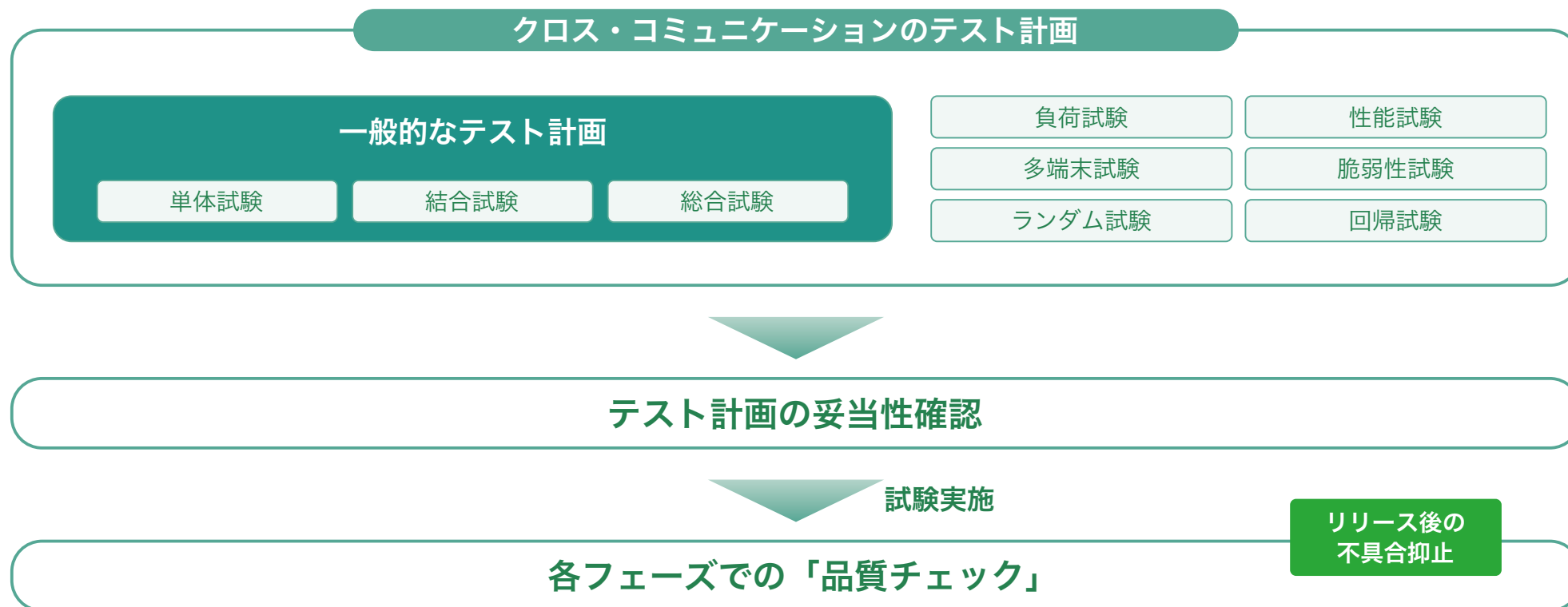
お客様視点の早期確認

開発資源の一元管理

ソースコード管理

自動化

テスト計画時に一般的な単体試験・結合試験・総合試験に加え、必要に応じて下記の試験を実施し、品質の担保に努めています。



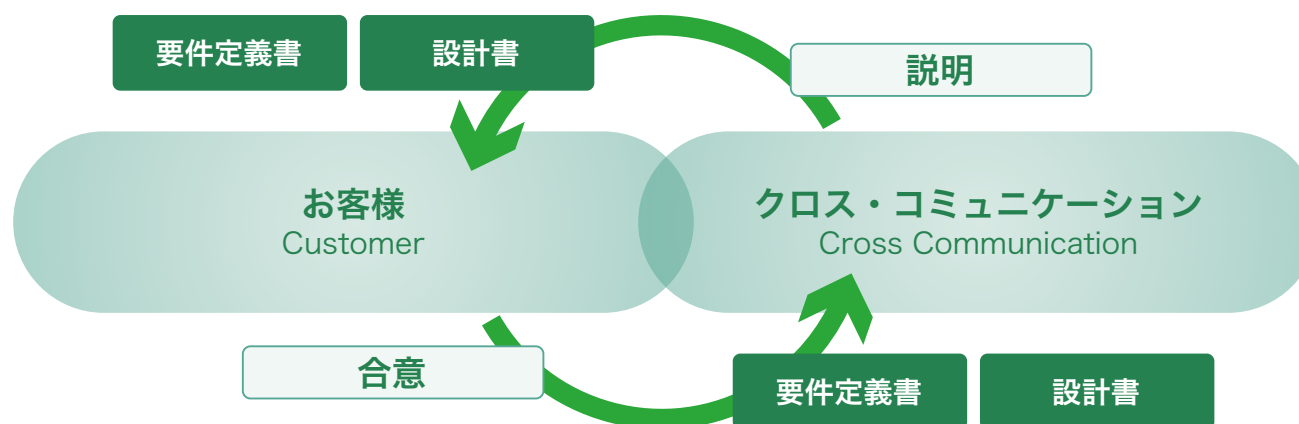
試験実施前に試験項目レビューを実施し、テスト計画（前提条件、スケジュール、体制、視点、実施に必要な情報等）の妥当性を確認しています。さらに試験実施事後の出荷判定において、試験状況、残障害、性能評価、セキュリティ評価等の「品質チェック」を行い、システムが十分な品質を担保してるかを確認し、リリース後の市場不具合を未然に防ぐ取り組みに努めています。

3 お客様視点の早期確認

品質向上をするうえでステークホルダー間の認識齟齬をなくすることが重要です。
UI/UX 専門チームの参画やプロトタイプ・α版での早期確認をします。

ドキュメント方針

ドキュメントの整合性や仕様の認識齟齬がないかをお客様と早期に確認を取りながら共通認識として開発を進めていきます。



・誰が見ても理解できる内容になっているか

・必要なものを必要なだけ作成しているか

開発時の取り組み

UI/UX 専門チームの参画

豊富な開発ノウハウを持つ社内のUI/UX 専門チームがプロジェクトの初期段階から参画し、PC/ モバイルアプリケーション開発のノウハウを各プロジェクトで共有・フォローします。

ワイヤーフレーム・モック制作

プロジェクトの初期段階、あるいは要件定義の段階で画面、機能といった完成形のイメージを実際に確認出来るワイヤーフレーム・モックを作成し、お客様と共有します。

α版・β版・プロトタイプ

システム完成前に、α版でデザインを確認したり、β版（プロトタイプ）で動作を確認して頂き、認識齟齬が無いかをご確認いただきます。

柔軟性のある開発手法

お客様のご要望に合わせた柔軟な開発手法で対応します。

システム開発におけるソースのバージョン管理を、柔軟かつ信頼性の高いツールを用いて行っており、安定して品質を保てる体制を整えています。

信頼性の高いツールによる一元管理



誰が何を管理しているのかが不明な状況



一元管理しているため共有も簡易

ソースコード管理

ソースコードは git で一元管理をしています。これにより、複数の開発メンバーが参加するプロジェクトにおいても、レビューなどが容易に行えソースコードの品質を保てます。

開発資源バックアップ

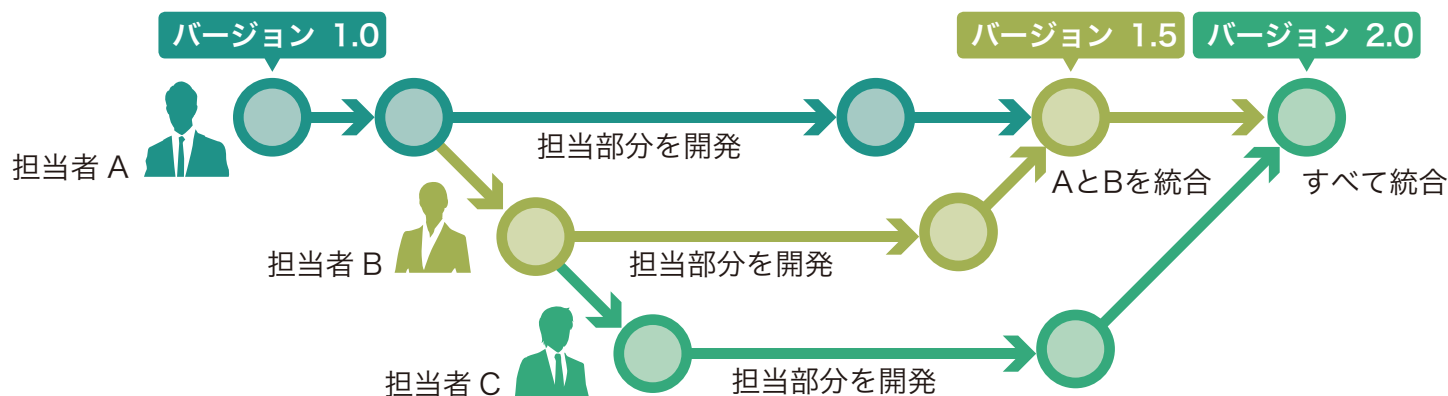
クロス・コミュニケーションでは、万が一障害が発生した場合も、開発資源のバックアップを取っているため、迅速にシステムを復旧させることが可能です。

ドキュメント・課題管理

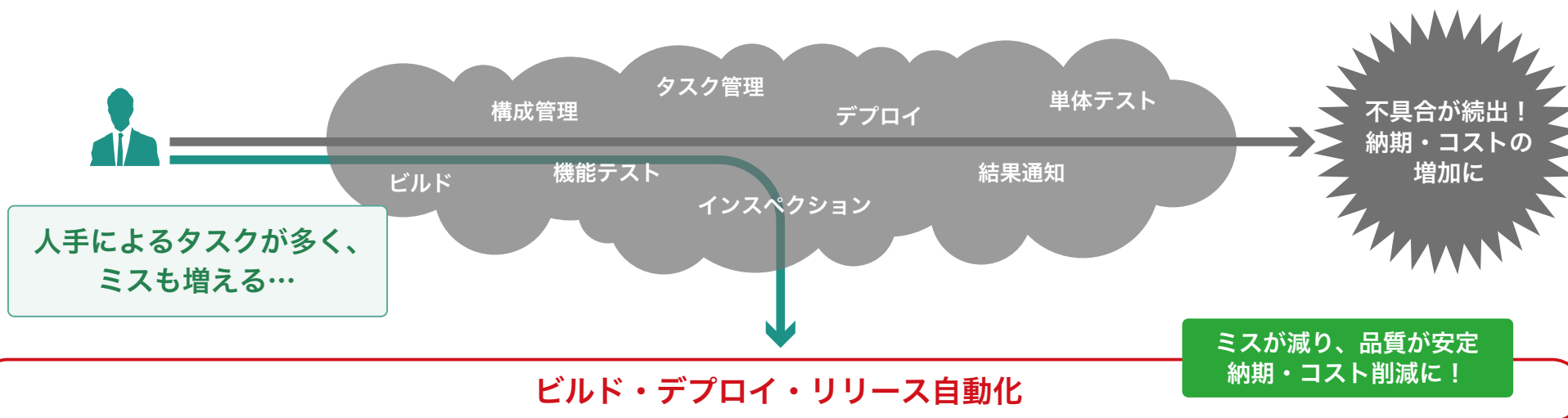
Redmine などでチケット管理を行い、必要に応じてお客様とも直接共有させていただき、課題の共有を実現しています。

Git-flow とは開発担当者の役割を明確にし、同時並行で担当する開発を進め、最終的に開発したソースコードを統合する運用ルールです。また、ソースコードのビルド・デプロイ・リリース時に人為的なミスにより不具合や遅延が発生します。CI/CD ツールでこれらを自動化することで人為的ミスを減らします。

Git-flow」によるソースコード運用フロー



ビルド・デプロイ等の自動化について



モダンな Web アプリケーションとしてのあるべき姿を 12 のベストプラクティスにまとめた「The Twelve Factor App」をベースにシステム開発に取り組んでいます。

The Twelve Factor App

1	コードベース	バージョン管理されている 1 つのコードベースと複数のデプロイ
2	依存関係	依存関係を明示的に宣言し分離する
3	設定	設定を環境変数に格納する
4	バックエンドサービス	バックエンドサービスをアタッチされたリソースとして扱う
5	ビルド・リリース・実行	ビルド、リリース、実行の 3 つのステージを厳密に分離する
6	プロセス	アプリケーションを 1 つもしくは複数のステートレスなプロセスとして実行する
7	ポートバインディング	ポートバインディングを通してサービスを公開する
8	並行性	プロセスモデルによってスケールアウトする
9	廃棄容易性	高速な起動とグレースフルシャットダウンで堅牢性を最大化する
10	開発・本番一致	開発、ステージング、本番環境をできるだけ一致させた状態を保つ
11	ログ	ログをイベントストリームとして扱う
12	管理プロセス	管理タスクを 1 回限りのプロセスとして実行する

PART. 3 Non-functional Requirements



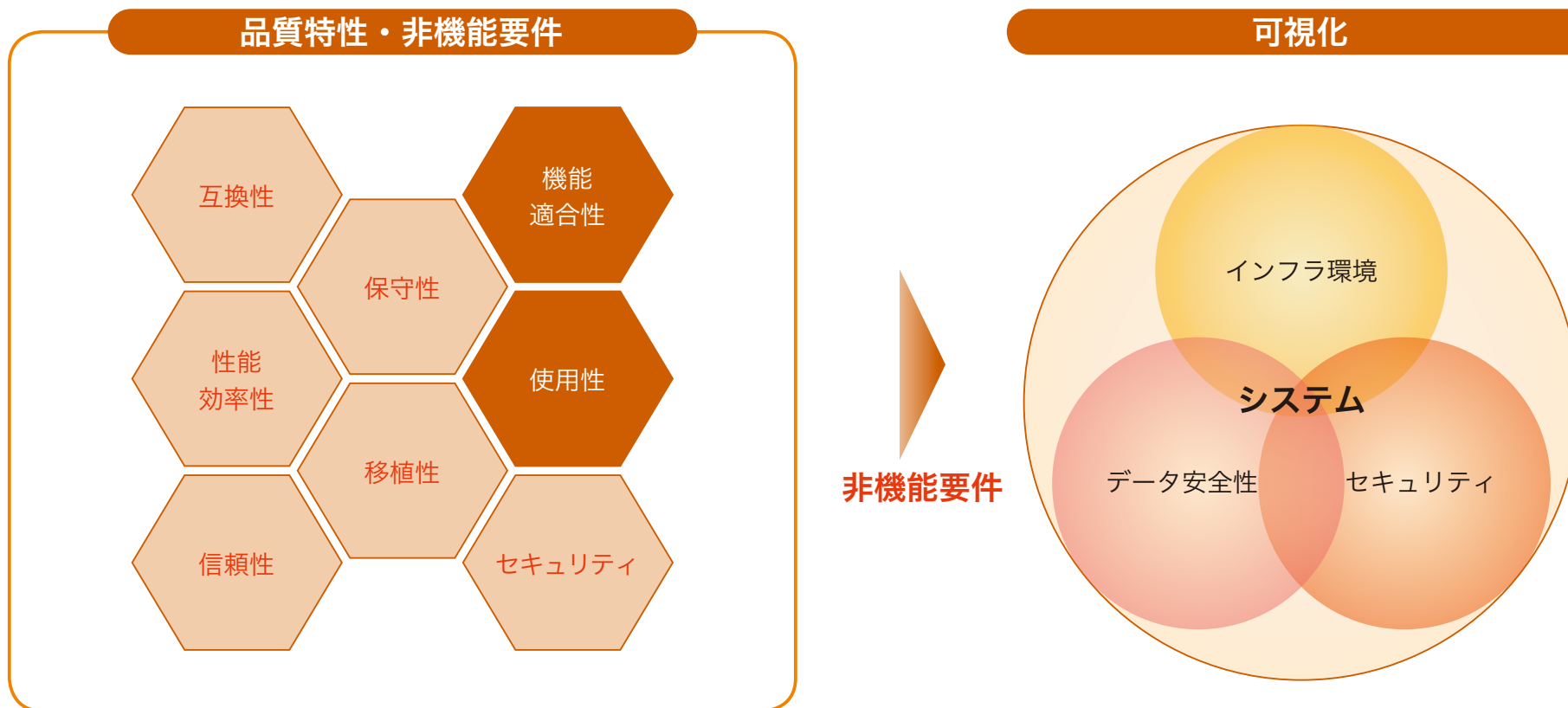
《非機能要件》

～お客様のビジネス環境に應えるための非機能要件の可視化～

目に見える機能に留まらず、
目に見えない非機能要件を可視化します。

1 品質特性と非機能要件

非機能要件を考慮することで、セキュリティ向上・信頼性向上・保守や移行を容易にします。
これらを実現するためにインフラの状態を可視化します。



インフラについて

システムの基盤となるインフラは安定したサービスを実現するための重要な要素です。
システムのセキュリティ確保、処理の分散化や多重化による信頼性の高いインフラの構築に努めています。

インフラ環境について以下の事項に取り組み、お客様が安心してシステムを利用するための環境構築に努めています。



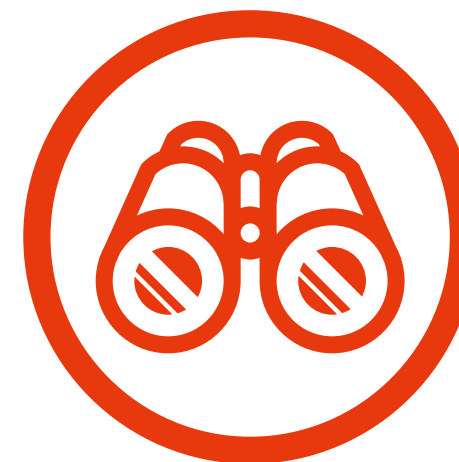
インフラ環境の信頼性

分散化による高い処理能力と、多重化による高い信頼性の確保を開発実績に基づいて実現し、さらにお客様の変化するビジネス環境にも柔軟に対応できる環境を構築しています。



システムダウン対策

自家発電を備えたインターネット・データセンターにサーバを設置しており、自然災害などが発生した場合も安定した運用が可能となっています。



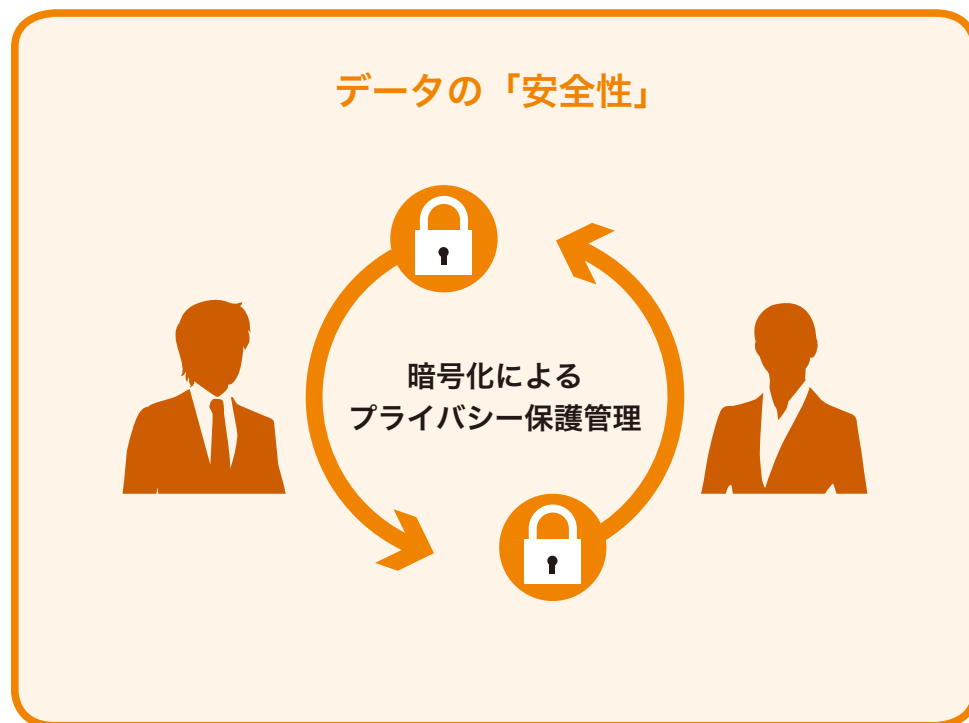
システム監視

24 時間 365 日の稼働を前提とした有人監視を行い、障害発生時も迅速に対応いたします。さらに、監視システムによりサーバをリアルタイムで状況監視する体制も整えています。

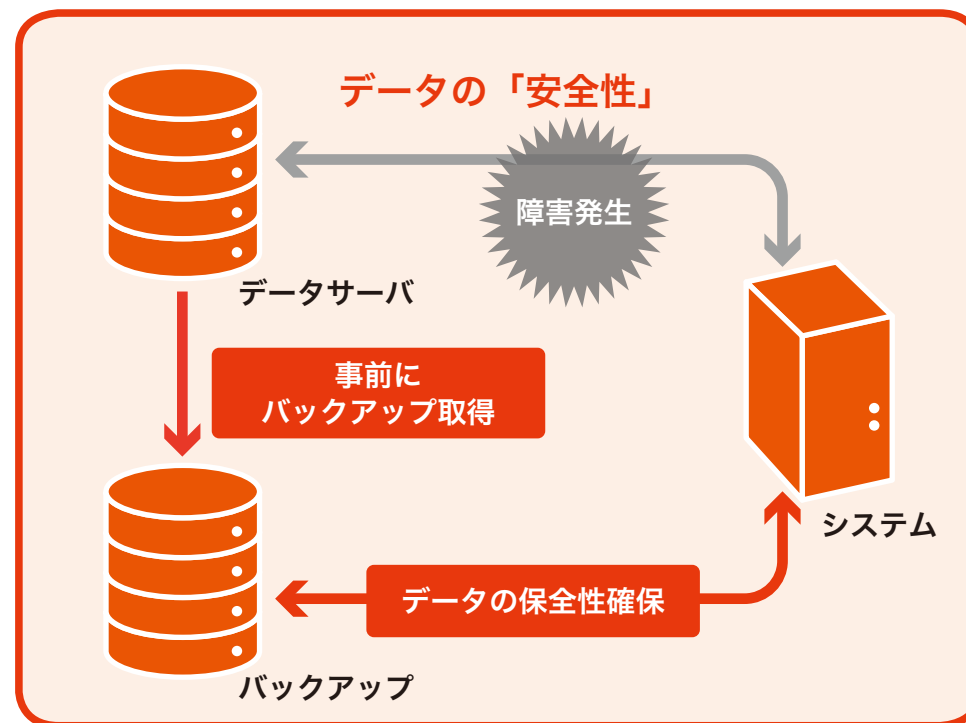
3 データの安全性・保全性

システムにおいて暗号化によるデータ保護や、障害発生時のデータの保全性確保など、普段の運用から障害時の対応まで考慮した利用時品質の実現に努めています。

普段の運用

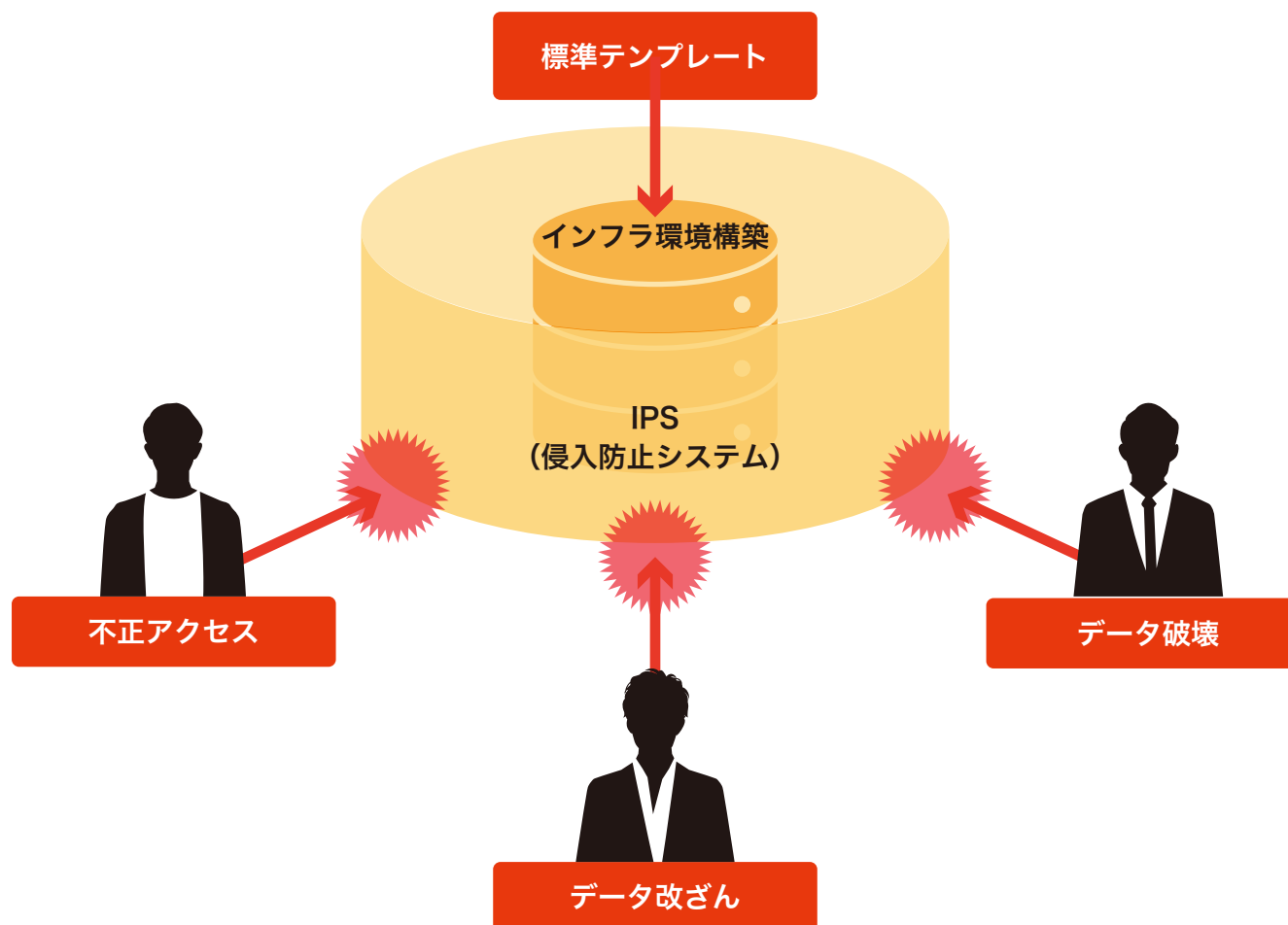


障害発生時の対応



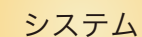
4 セキュリティ対策について

ファイアウォールや IPS(侵入防止システム) によって、外部の不正アクセスから保護し、データの改ざん・破壊を防止しています。
また、構築についてはテンプレートを用い、属人的な作業・運用ミスを徹底的に排除し、
更にリリースのセキュリティチェックを徹底することによりセキュリティの確保に努めております。



常時システム監視

アクセスログ監視ツール



人

個人情報

Reference

參考資料

こんなことで困っていませんか？

コーポレートサイト内で「こんな課題をお持ちの方に」一覧

アーキテクチャ設計／システム基盤構築

- ・ サービスや製品の UI/UX デザインが、使いやすくなっているか可視化をしたい。
- ・ アクティブユーザー数を増やしたい。
- ・ ユーザーの利用率を高めたい。
- ・ ユーザー視点でのモノづくりを実施したい。
- ・ ユーザーインサイトを元に効果を最大化させたい。

データドリブン／マーケティング

- ・ 自社が保有するデータを活用しきれていない。
- ・ 3rdParty データと組みあわせて顧客課題を抽出したい。
- ・ 分析ができるツールを探している。
- ・ マーケティングプラットフォームの導入を検討している。

各種リサーチ

- ・ サービスや製品の現状を把握したい。
- ・ 競合他社の状況が知りたい。
- ・ ターゲットとするユーザーがどのような思いで、どのような行動をとるか知りたい。
- ・ 海外のユーザーに対しても調査を実施したい。

スマホアプリ開発

- ・ スマホアプリを作りたいが、どのように進めたら良いかわからない。
- ・ iOS アプリ、Android アプリごとに開発するコストを減らしたい。
- ・ アプリ開発の最新技術を活用し、ユーザービリティが高いアプリを制作したい。
- ・ スマホアプリ開発をして販促活動を強化したい。
- ・ アプリ開発の実績が豊富な会社に制作を依頼したい。

Webシステム開発

- ・ Web サービスの立ち上げに伴いシステム開発をしたい。
- ・ 既存ベンダーの Web システムの品質が低く、対応しきれない。
- ・ デバイスごとに最適な Web システムを開発したい。
- ・ Web システムを利用して業務効率化を図りたい。
- ・ Web システム開発の実績が豊富な会社に構築を依頼したい。

Web制作

- ・ ユーザーインサイトを軸にサイトリニューアルを行いたい。
- ・ 集客効果やコンバージョン率を向上させたい。
- ・ デザインは感性のみでは無く、データによる根拠を裏付けたデザインにしたい。
- ・ 対象ユーザーのニーズを加味した UI を実現したい。
- ・ 運用しやすいサイトを設計してほしい。

アーキテクチャ設計／システム基盤構築

- ・ モバイルアプリケーション開発における、インフラ構築のノウハウが無い。
- ・ アーキテクチャ設計から構築まで経験を持つエンジニアに任せたい。
- ・ AWS クラウド上にシステム構築・移行を検討している。
- ・ 24 時間 365 日の監視体制を構築したい。

Webサイト運用

- ・ Web サイト運用に手間がかかり、マーケティング戦略の立案に手が回らない。
- ・ Web サイトからの見込み客獲得／売上アップを目指したい。
- ・ 複数の Web 制作会社や Web サイト運用会社への依頼を一本化したい。
- ・ Web サイトに適切な KGI / KPI を設定して PDCA を回したい。
- ・ Web サイト運用の実績が豊富な会社に依頼したい。

検証・試験

- ・ アプリや Web サイトの検証に時間・コストがかかっている。
- ・ 機種依存の不具合をチェックしたいが検証端末が足りない。
- ・ 自社でデバッグすることで起こる「慣れ」による見落しを防ぎたい。
- ・ 自社でのデバッグリソースでかかるコストを見直したい。
- ・ 検証・試験の実績が豊富な会社に依頼したい。

カスタマーサポート

- ・ カスタマーサポート業務を BPO したいが、高い品質も担保したい。
- ・ 業務が拡大する中で、コア業務へ集中できる環境を求めている。
- ・ 属人化した業務の安定稼働を求めている。
- ・ 繁忙期のスタッフ教育や、処理作業の負担を改善したい。
- ・ カスタマーサポートの実績が多い会社に BPO したい。

コーポレートサイト内で「こんな課題をお持ちの方に」クロス・コミュニケーションの答え一覧

1 開発を委託したが、不具合が多く品質に不満足な結果になった…

「クロス・コミュニケーションの品質マネジメントの理念」

クロス・コミュニケーションでは、お客様にご満足いただける「魅力的品質」をご提供するための最適な QCD バランスを実現します。

2 要求に対して意図していないシステムが出来上がってきて困っている…何が原因なのかわからない

「品質計画」

お客様の要求を漏れなくシステムに反映するには、初期段階での計画が重要です。クロス・コミュニケーションでは、魅力的品質の実現に向けた取り組みを行っています。

3 機能は要求どおりだが、システムに合った性能になっていないため、結局使い勝手が悪い…

「品質管理」

システムの機能のみに留まらず、実際にシステムを利用する際の環境（システムの性能）まで考慮し、目に見えない品質の「見える化」に努めています。

4 コミュニケーションがうまくいかず、開発が進んでから認識違いが発覚し、大幅なロスが発生してしまった…

「要求仕様・ユーザ視点の早期確認の徹底」
ドキュメント方針

クロス・コミュニケーションでは、「ステークホルダー間の共通認識を生む」ための取り組みを徹底しており、常にお客様とコミュニケーションを取りつつ開発を進めています。

5 開発がベンダー任せになっており、開発プロセスや納品物の品質レベルが見えず不安だ…

「品質管理チームによるレビューの徹底」

品質管理チームが過去の実績に基づく様々なノウハウをベースに、開発の重要なフェースで「品質移行判定」を行うため、「作りこまれた品質」を保証します。

6 事業として新たな領域のアプリ開発を依頼したいが、どうすれば望む品質になる要求を出せるのかわからない…

「品質監査」

豊富な開発実績を基に「お客様目線のレビュー」を行うことができるので、お客様の望む品質をクロス・コミュニケーションが見極め、実現します。

7 開発側とのレビュー会を開き、事前に品質をチェックしていきたいが、どうレビューするか観点・ノウハウがない…

「要求仕様・ユーザ視点の早期確認の徹底」
開発時の取り組み

クロス・コミュニケーションでは、開発時にプロトタイプングやモック制作を行い、お客様が「実際に見て、良し悪しを判断できる」ための取り組みを行っています。

8 QCDのバランスがわからない…開発側でコントロールしてほしい

「クロス・コミュニケーションの品質マネジメントの理念」

クロス・コミュニケーションでは、お客様にご満足いただける「魅力的品質」をご提供するための最適な QCD バランスを実現します。

9 自社開発のシステムを検証・試験しているが、効果的な方法がわからず、余計なコストが増え困っている…

「CI ツールを用いた自動化の徹底」

クロス・コミュニケーションではテストなどの自動化を行い、効率的、また人為的な不具合を減らし、コストと納期の増加を防ぐ取り組みをしています。



株式会社クロス・コミュニケーション

〒102-0076 東京都千代田区五番町6-1 AKビルディング 3F

phone : 03-6859-2262 fax : 03-3288-6352

【URL】 <http://www.cross-c.co.jp> 【E-mail】 cc-sales@cross-c.co.jp

【過去事例】

弊社品質にご評価頂いた実績の最新情報を弊社のホームページに公開しています。

<http://www.cross-c.co.jp/case/>