

アプリ運用ルール策定ガイド 拡張機能編 〈第1版〉

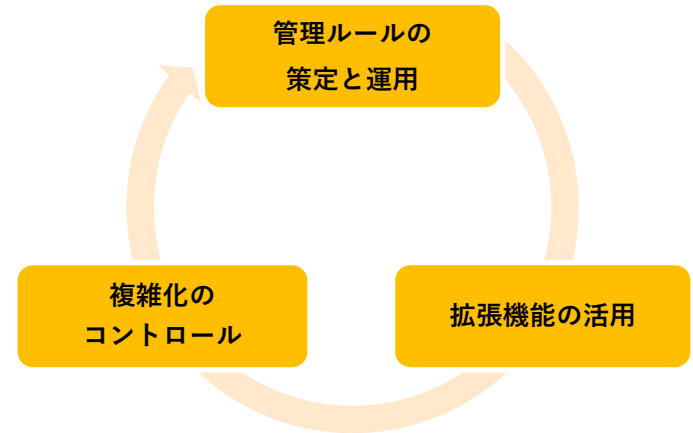
サイボウズ株式会社



※この資料は2021年11月時点のkintoneを基に作成しております

当資料の作成背景と目的

- kintoneの拡張機能を活用すると、より便利な業務システムを構築できます。小規模なチームがkintoneを上手く活用できた後、**そのkintone上で社内の他部門も利用を始めていく**というケースもよく見られます。
- このようにkintoneを多くのユーザーで利用する際、運用保守を円滑に進めていくためのポイントがあります。そういったポイントの1つが「**拡張機能の管理ルール策定**」です。
- 当資料は、この「**拡張機能の管理ルールを策定する**」際の参考資料としてご利用ください。
- 策定した管理ルールを元に運用保守を円滑に進めながら、**より強力な業務改善のためのプラットフォームとして利用いただける**のが理想です。
- 当資料は、主に**IT部門**の方を対象読者として想定しています。



目次

●	kintoneでの機能追加	3
●	サイボウズが提供する拡張機能	4
●	機能追加するためには？	5
●	kintoneでの機能追加	6
●	運用保守を円滑にするための観点	7
●	運用保守しにくくなる原因	8
●	管理ルール策定にあたっての参考情報	9
●	考慮すべき5つのポイント	10
●	0. 管理体制の構築	11
●	1. 機能追加のコントロール	12
●	2. 管理権限のコントロール	15
●	3. 要求と要件のコントロール	17
●	4. 棚卸し	18

kintoneでの機能追加

サイボウズが提供する拡張機能

kintoneをスタンダードコースで契約することで、以下4種類の拡張機能を利用することができます。

REST API

他システムとデータを連携できるようになる。kintoneのデータを他システムに渡したり、他システムのデータをkintoneに取り込むことができる。



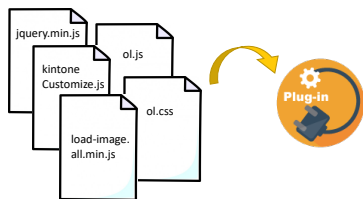
画面カスタマイズ

画面をカスタマイズすることで、標準搭載されていない機能を追加することができる。



プラグイン

複数の JavaScript や CSS をひとまとめにして適用できる追加プログラム。



Webhook

他システムにデータを送信する機能。kintoneにデータが登録されたり、更新されたりしたタイミングで、そのデータを他システムに送信することができる。



Webhooks logo
The MIT License
(<https://github.com/logo/webhooks>)
Copyright © 2014 <friends@segment.io>

これら「サイボウズが提供する拡張機能」を利用することで様々な機能を追加できます。

機能追加するためには？

大きく以下2つの方法で機能が追加できます。両方の方法を組み合わせることも可能です。



個別にシステム開発する

- 前頁で紹介した「サイボウズが提供する拡張機能」を使って、個別にシステム開発する方法です。
- 自由度が高い反面、プログラム開発が多く必要となり、**一般的なシステム開発に近い方法**となります。

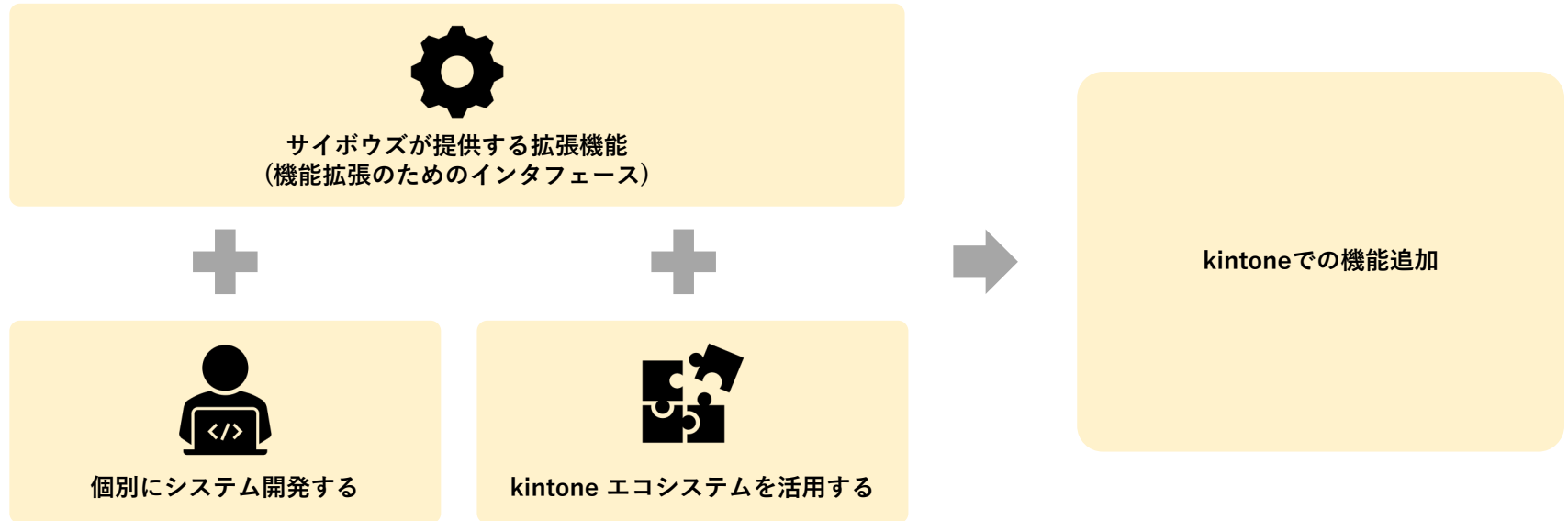


kintone エコシステムを活用する

- 前頁で紹介した「サイボウズが提供する拡張機能」に「**kintone エコシステムが提供する拡張機能**」を組み合わせ、機能追加する方法です。
- 自由度が抑えられる反面、プログラム開発もなるべく抑えられ、**いわゆるローコード開発に近い方法**です。

kintoneでの機能追加

つまり「サイボウズが提供する拡張機能」に前述の2つの方法を組み合わせることで、kintoneに機能追加できます。



運用保守を円滑にするための観点

以下の観点を押さえることで、kintoneの拡張機能が運用保守し易くなります。

1 機能追加のコントロール

kintoneとは別の
連携サービス



追加された
プラグイン



kintone外にある
データ連携用
プログラム



追加された
カスタマイズ



kintone

管理対象を抑えることで、
自分たちがコントロールし易い範囲に
留めることができる

2 管理権限のコントロール

管理も
担ってくれる人



機能追加
したいだけの人



管理を担ってくれる人に
管理権限を付与することで、
秩序立てて機能追加することができる

3 要求と要件のコントロール

プラグイン追加



背景を残す



カスタマイズ追加



kintone

機能追加の背景となった
要求事項や要件を記録することで、
運用保守時に判断し易くできる

運用保守しにくくなる原因

kintoneの拡張機能が運用保守しにくくなる原因としては、主に以下の3つが考えられます。

1 度重なる機能追加で管理対象が増えがち

kintoneとは別の
連携サービス



追加された
プラグイン



kintone

kintone外にある
データ連携用
プログラム



追加された
カスタマイズ
プログラム



管理対象が増えるので単純に複雑になる

2 機能追加したい人に管理権限を付与しがち

プラグインを
インストール
したいです。



どうぞ。
管理権限を
付与しますね！



アプリをカス
タマイズした
いです。



権限を付与しすぎると機能も増え過ぎる

3 機能だけ追加しがち

プラグイン追加



kintone

カスタマイズ追加




導入理由や経緯が分からなくなり、
運用保守時の判断が難しくなる

管理ルール策定にあたっての参考情報

考慮すべき5つのポイント

管理体制をまず構築したうえで、先の「運用保守を円滑にするための観点」で挙げた3点を押さえます。

さらに運用をすすめていくなかで棚卸しを行うことで、顕在化した問題点に対処していきつつ、kintoneを継続的な業務改善ができるプラットフォームとして保つことができます。



0. 管理体制の構築

1. 機能追加のコントロール

2. 管理権限のコントロール



3. 要求と要件のコントロール

4. 棚卸し

0. 管理体制の構築

まずは**管理体制を構築**しましょう。最低限、**以下2つの役割を誰が担当するのか**を決めておくことをお勧めします。

※カスタマイズしない場合は**kintone全体の拡張機能管理者**の1つのみ。

役割	担当範囲	付与しておく権限	期待する行動	補足
kintone全体の 拡張機能管理者 	自社のkintoneで利用する 拡張機能を管理する	cybozu.com共通管理者	<ul style="list-style-type: none">● 機能追加を行うかどうかを判断する● kintoneシステム管理者以上の権限が必要な機能追加の設定作業を行う<ul style="list-style-type: none">・プラグインをインストールする・カスタマイズを適用する● 追加した機能の棚卸しをする	以下の機能差異があるため、kintoneシステム管理者では権限的に不足です <ul style="list-style-type: none">● 定期メンテナンスや緊急メンテナンスなどの情報はメール配信されない● 自分がアプリ管理者に設定されていないアプリを操作できない
カスタマイズの 保守担当者 	自社のkintoneで利用する カスタマイズを開発/保守する	(不要)	<ul style="list-style-type: none">● カスタマイズを開発する(外注する)● カスタマイズを保守する● カスタマイズの<u>適用はしない</u>	カスタマイズを外注する際も、将来的な保守性を考慮し、自社でもある程度保守できる状態にしておくのが望ましいです

1. 拡張機能のコントロール

利用する拡張機能を限定することで、自分たちがコントロールしやすい範囲に留めることができます。

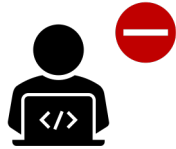
kintoneの管理権限では利用する拡張機能を選択することができないため、**運用でのコントロール**(※)となります。

※管理権限を持つユーザー同士で「当社では JavaScript / CSS カスタマイズの利用は止めておきましょう」など取り決める。

コントロールするにあたっての考え方の一例は以下の通りです。

個別システム開発を禁止する

- ・ JavaScript / CSS カスタマイズ禁止
- ・ 自作プラグインのインストール禁止



②個別開発する必要が出てきたら

①機能追加が必要になったときはまず

ノーコードでkintoneアプリをカスタマイズできる連携製品（例：gusuku Customine）を許可する

- ・ プログラムを書くのではなく、仕様を自然言語で書いて機能実装



段階的な拡張機能の追加

- ・ kintoneエコシステムで提供される拡張機能を一括ではなく段階的に追加



または

導入時に一括で拡張機能を追加

- ・ kintoneエコシステムで提供される拡張機能のうち必要なものをあらかじめ一括で追加



※開発元が違うプラグインの競合などに要注意

参考：IT部門の積極的関与が難しい場合

IT部門が業務上、kintoneの運用に積極的な関与が難しいケースもあると考えられます。
そのような場合は「個別にシステム開発する」方法は採用しないというポリシーにすることで、
IT部門でのコントロールが必要となるポイントを減らすという選択肢があります。



参考：個別システム開発にIT部門がなぜ必要か？

以下のような技術要素を用いてのシステム開発となるため、画面開発やデータ連携を行う場合はIT部門も交えた保守体制を組みましょう。

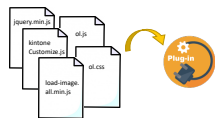
画面カスタマイズ(プラグイン含む)

● JavaScriptカスタマイズ

- kintone JavaScript API : kintone 公式のAPI
- kintone.api : kintone REST API を実行できる
- kintone.proxy : 外部通信できる
- kintone.plugin.app.proxy : プラグイン用に情報秘匿できる
- kintone に限らない一般的な JavaScript の技術
 - 他システムとのデータ連携 (fetchメソッドなど)
 - DOM操作 ※非推奨



● CSSカスタマイズ



データ連携

● kintone REST API : kintone 公式の API

- 認 証
 - OAuth
 - APIトークンでの認証
 - ユーザー認証



Webhooks logo
The MIT License
(<https://github.com/segment-io/webhooks>)
Copyright © 2014 <friends@segment.io>

● Webhook



2. 管理権限のコントロール

「0. 管理体制の構築」を把握できているユーザーに限定して管理権限を付与することで、秩序だった機能追加が可能です。

kintoneで強い管理権限は「cybozu.com共通管理者」と「kintoneシステム管理者」の2つです。

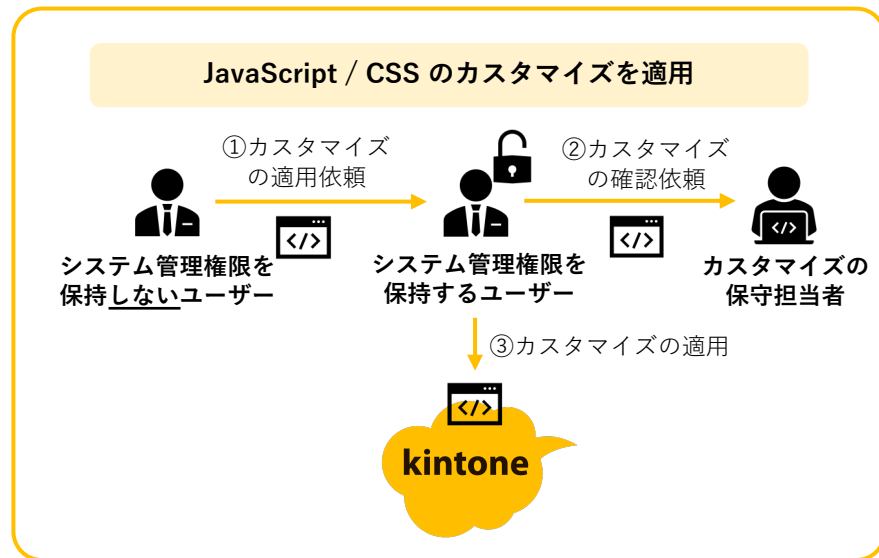
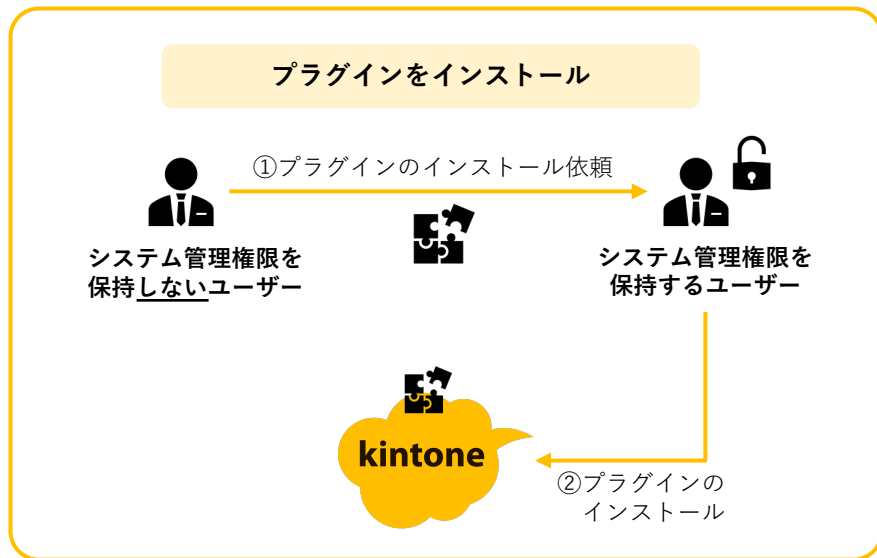
cybozu.com共通管理者は操作できる機能が多い反面、その名の通り、cybozu.com全体の管理機能も操作可能となります。

機能拡張のための インターフェース	cybozu.com共通管理者	kintoneシステム管理者	(どちらの権限もない場合)
APIトークン (REST API)	○：管理できる	△： <u>アプリのアクセス権(アプリ管理)</u> があれば管理できる	△： <u>アプリのアクセス権(アプリ管理)</u> があれば管理できる
Webhook	○：設定できる	△： <u>アプリのアクセス権(アプリ管理)</u> があれば管理できる	△： <u>アプリのアクセス権(アプリ管理)</u> があれば管理できる
JavaScript / CSSでカスタマイズ (kintone全体)	○：適用できる	○：適用できる	×：適用できない
JavaScript / CSSでカスタマイズ (アプリ)	○：適用できる	△： <u>アプリのアクセス権(アプリ管理)</u> があれば適用できる	×：適用できない
プラグイン	○：インストールできる	○：インストールできる	×：インストールできない

▼ kintone上での管理者についての詳細はこちら
kintone よくあるご質問 (FAQ) - 「kintone」の管理者について
<https://faq.cybozu.info/alphascope/cybozu/web/kintone/Detail.aspx?id=1863>

参考：システム管理権限の必要な操作を他ユーザーが行いたい場合

システム管理権限の必要な操作を他ユーザーが行いたい場合も、基本的には権限を保持するユーザーが行いましょう。



3. 要求と要件のコントロール

機能追加の背景となった要求事項や要件を記録することで、運用保守時に判断し易くなります。

アプリ管理者用メモに記録する等の形でご対応ください。



The image shows a mobile application interface for '採用チーム予算管理アプリ' (Recruitment Team Budget Management App). The top navigation bar includes a home icon, a notification bell, and a star icon. Below the navigation bar, the page title is '採用チーム予算管理アプリ > アプリの設定' (Recruitment Team Budget Management App > App Settings). The main content area features a red circular icon with a person silhouette, followed by the app name '採用チーム予算管理アプリ'. A red rectangular box highlights the link 'アプリ管理者用メモ' (App Administrator Memo). Below this are two buttons: 'フォーム' (Form) and '一覧' (List). A red arrow points from the highlighted link to a detailed view of the 'アプリ管理者用メモ' (App Administrator Memo) dialog box.

アプリ管理者用メモ

【アプリの作成経緯】

- ・ 経理から依頼を受け、採用チーム用にも予算管理アプリを作成しました。
- ・ 入力項目と一覧の設定は、「予算管理（人事部用）」を参考にしています。

【注意事項】

- ・ 「評価」フィールドは、社内の人事評価システムと連携するカスタマイズをしています。設定を変更する場合はシステムに互換性が必要なので、人事の田中までご連絡ください。

ⓘ アプリテンプレートやアプリの再利用時に、このメモの内容は含まれません。

アプリ管理者用メモは、アプリ作成の目的や設計のポイントなど、メモしておきたい情報を自由に書くことができる機能です。

※ アプリ管理権限のあるユーザーのみ閲覧できます。

編集する 閉じる

https://jp.cybozu.help/k/ja/user/app_settings/notes_for_app_administrators.html

4. 棚卸し

kintone全体の拡張機能管理者を中心に、追加した機能の棚卸しを行きましょう。半年に一回の開催がお勧めです。
棚卸しするポイントとしては以下を参考にしてください。

確認対象	機能	観点
プラグイン	kintoneシステム管理 > プラグイン	<ul style="list-style-type: none">・用途不明なプラグインがインストールされていないか・利用されていないプラグインがないか
アプリのカスタマイズ	kintoneシステム管理 > アプリ管理	<ul style="list-style-type: none">・カスタマイズを使用中であるアプリの数・用途不明なカスタマイズがないか
kintone全体のカスタマイズ	kintoneシステム管理 > JavaScript / CSSでカスタマイズ	<ul style="list-style-type: none">・kintone全体のカスタマイズをしていないか・用途不明なカスタマイズではないか
REST API	kintoneシステム管理 > アプリ管理	<ul style="list-style-type: none">・1日のAPIリクエスト数が最も多いアプリは何か・そのAPIリクエスト数はいくらか・何アプリでAPIリクエスト数が1以上か・用途不明なAPIリクエストが発生しているアプリはないか
kintoneとは別の連携サービス	(連携サービス側の設定画面など)	<ul style="list-style-type: none">・kintoneとの接続設定で用途不明なものがないか・利用されていない設定が残っていないか・連携サービス側の利用ユーザーの棚卸しはできているか

策定した管理ルールを元に、運用保守を円滑に進めながら
継続的な業務改善を進めてください！