

X-point

サイボウズ **kintone** 連携サービス 利用ガイド

2026/05/26 版



はじめに

◆本書の目的

本書は、X-point で利用可能な「サイボウズ kintone 連携サービス」の利用手順について説明しています。

◆対象とする読者

本書は「サイボウズ kintone」のアプリ開発者、及び「X-point」のフォーム管理、JavaScript API を利用したプログラミングが可能なユーザーを対象としています。なお、kintone アプリ開発に係わる各種情報はサイボウズ社が提供する情報（「cybozu developer network」など）を御利用願います。

◆対応バージョン（2026/05/26 時点）

| X-point | kintone | 対応ブラウザ（※） |
|---|-----------|------------------------------|
| X-point v3.13 「サイボウズ kintone 連携サービス」及び 「セキュリティ基本サービス」必須 | スタンダードコース | X-point、kintone の双方が対応するブラウザ |

- ※ kintone のドメイン名が「cybozu.com」、「kintone.com」、「cybozu.cn」以外である場合、「サイボウズ kintone 連携サービス」は利用できません。
- ※ 「サイボウズ kintone 連携サービス」の契約が無いお客様は本機能を利用できません。ライセンスが適用されていない場合、様々なエラーが発生します。
 - ① kintone から X-point に送信するとエラーが発生します。
 - ② X-point の書類表示時、フォーム表示時に JavaScript エラーが発生します。
 - ③ 管理者画面に kintone 関連の設定項目が表示されません。
- ※ X-point Cloud の「セキュリティ基本サービス」が必須です。「セキュリティ基本サービス」が無い場合、kintone アプリより連携 API を呼び出す事ができません。
- ※ 対応ブラウザは X-point、kintone の両方で利用可能なブラウザになります。一方のみ対応しているブラウザは正常に動作しない場合があるため利用しないでください。
- ※ X-point のみを利用する場合であっても、kintone 連携が指定されているフォーム画面、マスタ参照画面がある場合、X-point と kintone が対応するブラウザが対応ブラウザになります。双方に対応しないブラウザを利用すると検索画面が表示されない、表示されても操作できない等が発生する可能性があります。今後、対応ブラウザが変更される場合があります。最新情報は kintone、X-point 双方の製品情報をご確認ください。
- ※ 本書は PC ブラウザでの利用を前提に記述されています。タブレット、スマートフォンでの画面レイアウトは PC ブラウザと異なりますのでご注意ください。
- ※ スマートフォンで利用する際は検索画面が画面幅に収まらない場合があります。収まらない場合は、画面を横方向に向けて利用してください。
- ※ kintone REST API でユーザー認証を利用する場合、SAML 認証を利用する環境では注意が必要です。REST API では SAML 認証が指定されている場合であっても利用しません。X-point サーバーに認証用パスワードを登録する必要があります。
- ※ X-point から kintone に送信される各種リクエストでは「User-Agent」ヘッダに「X-point/{バージョン番号}」の製品名とバージョン番号を示す文字列が設定されます。（例「X-point/3.0.00」）。

◆製品名について

本文中、「サイボウズ kintone」は「kintone」、「X-point サーバー」は「X-point」と表記しています。また、各製品の名称は対応バージョンを省略してある箇所もありますのでご了承ください。

◆商標について

本書の一部、または全部を著作権所有者の許諾なしに、商用目的の為に複製、配布することはできません。X-point、エクスポイントの名称およびロゴは株式会社エイトレッドの商標または登録商標です。Microsoft、MS-DOS、Windows は米国 Microsoft Corporation の米国およびその他の国における登録商標です。Macintosh、MacOS は Apple Computer, Inc. の米国およびその他の国における登録商標です。Adobe、Acrobat、Adobe Acrobat は Adobe Systems, Inc. の商標または登録商標です。ORACLE、Java、JavaScript は、Oracle Corporation 及びその子会社、関連会社の米国及びその他の国における登録商標です。サイボウズ、Cybozu、kintone はサイボウズ株式会社の登録商標です。その他、記載された会社名およびロゴ、製品名などは該当する会社の商標または登録商標です。本書では、®、®、(TM) の表示を省略しています。ご了承ください。

◆製作著作

©2026 株式会社エイトレッド

目次／索引

| | |
|--|-----------|
| 第1章. サイボウズ kintone 連携サービスの概要 | 7 |
| 1.1. kintone 連携サービスとは | 7 |
| 1.2. 連携可能な組み合わせ | 8 |
| 1.3. 運用時のネットワーク構成 | 9 |
| 1.3.1. IP アドレス接続制限 | 9 |
| 1.3.2. プロキシサーバーや VPN の影響 | 9 |
| 1.4. 運用イメージ | 10 |
| 第2章. 連携の仕様 | 11 |
| 2.1. ユーザーの考え方 | 11 |
| 2.1.1. kintone → X-point (書類提出) | 11 |
| 2.1.2. X-point → kintone (レコード追加) | 12 |
| 2.1.2.1. 連携設定による連携 | 12 |
| 2.1.2.2. JavaScript による連携 | 14 |
| 2.1.3. kintone アプリから指定書類をブラウザ表示 | 14 |
| 2.2. kintone からの操作 | 15 |
| 2.2.1. X-point に書類を申請・更新 | 15 |
| 2.2.2. X-point の書類を削除 | 17 |
| 2.2.3. 書類の内容を取得 | 18 |
| 2.2.4. 書類の承認状況を取得 | 18 |
| 2.2.5. 書類の承認状況を表示 | 19 |
| 2.2.6. 書類の添付ファイルを取得 | 19 |
| 2.2.7. 承認待ち件数を取得 | 20 |
| 2.2.8. 書類を一括申請 | 20 |
| 2.2.9. kintone アプリのフォームから TSF ファイルを作成 | 21 |
| 2.2.10. kintone アプリから自動申請用の CSV を出力 | 23 |
| 2.2.11. X-point の書類を開く | 23 |
| 2.2.12. kintone から承認する (ワークフロー操作) | 24 |
| 2.3. X-point からの操作 | 26 |
| 2.3.1. ワークフロー処理から kintone アプリにレコードを追加・更新 | 26 |
| 2.3.2. X-point の書類ナビボタンから kintone アプリにレコードを追加・更新 | 30 |
| 2.3.3. クエリ CSV 出力から kintone アプリにレコードを追加 | 31 |
| 2.4. X-point 上で kintone アプリをマスタに利用 | 32 |
| 2.4.1. kintone アプリ・マスタとして利用 | 32 |
| 2.4.2. 固有マスタ定義の項目に利用 | 33 |
| 2.4.3. インポートバッチ登録 | 34 |
| 第3章. 事前準備 | 36 |
| 3.1. kintone 側の準備 | 37 |
| 3.1.1. 連携アプリの作成 | 37 |
| 3.1.2. API トークンの作成 | 39 |
| 3.1.3. 連携ライブラリの登録 | 41 |

| | | |
|---------------|------------------------------------|------------|
| 3.1.4. | kintone アプリの書類 ID フィールドを制御する | 44 |
| 3.2. | X-point 側の準備 | 46 |
| 3.2.1. | フォームの作成 | 46 |
| 3.2.2. | kintone 連携の登録 | 50 |
| 3.2.3. | 自動申請の登録 | 52 |
| 第 4 章. | kintone アプリから X-point への連携 | 57 |
| 4.1. | 連携テンプレートを利用して連携を行う | 57 |
| 4.1.1. | 連携用 JavaScript テンプレートの取得 | 57 |
| 4.1.2. | 連携用 JavaScript テンプレートの修正 | 58 |
| 4.1.3. | 連携用 JavaScript で特殊な扱いをする項目 | 66 |
| 4.1.3.1. | 西暦、月、日フィールド | 66 |
| 4.1.3.2. | チェックボックス | 67 |
| 4.1.3.3. | ラジオボタン | 68 |
| 4.1.4. | 連携用 JavaScript を kintone から実行する | 69 |
| 4.1.4.1. | X-point に書類を申請 | 69 |
| 4.1.4.2. | X-point の書類を削除 | 73 |
| 4.1.4.3. | 申請書類の内容を取得 | 75 |
| 4.1.4.4. | 書類の添付ファイルを取得 | 79 |
| 4.1.4.5. | フォームの定義情報を取得 | 80 |
| 4.1.4.6. | 申請書類の承認状況を取得 | 84 |
| 4.1.4.7. | 承認待ち件数を取得 | 93 |
| 4.1.4.8. | 一括申請 | 95 |
| 4.1.4.9. | 申請書類を表示する | 97 |
| 4.1.4.10. | 承認状況を表示する | 100 |
| 4.2. | X-point の連携 API を使い申請処理を作成 | 102 |
| 4.2.1. | アプリに【申請】ボタンを追加 | 102 |
| 4.2.2. | X-point に書類データを転送 | 104 |
| 4.2.3. | X-point に添付ファイルを転送 | 110 |
| 4.2.4. | X-point にコメントを転送 | 110 |
| 4.2.5. | 申請失敗時の表示 | 111 |
| 4.3. | X-point に登録された書類を kintone 上で開く | 113 |
| 4.3.1. | アプリに【書類表示】ボタンを追加 | 113 |
| 4.3.2. | 書類を表示 | 115 |
| 4.4. | kintone からの操作での注意事項 | 117 |
| 4.4.1. | 承認状況画面のデザインが受ける影響 | 117 |
| 4.4.2. | 書類表示のログイン認証が与える影響 | 118 |
| 第 5 章. | X-point フォームから kintone への連携 | 120 |
| 5.1. | 管理サイトの設定で kintone にレコードを追加・更新する | 120 |
| 5.1.1. | 連携先の kintone アプリを指定 | 120 |
| 5.1.2. | kintone レコード ID の保存フィールドを指定 | 125 |
| 5.1.3. | 添付ファイル登録先を指定 | 126 |
| 5.1.4. | 完了書類の登録先を指定 | 128 |

| | | |
|---------------|--|------------|
| 5.1.5. | kintone レコードを追加・更新するタイミングを指定 | 130 |
| 5.1.6. | フィールド指定 | 131 |
| 5.1.6.1. | チェックボックスの特殊な指定 | 133 |
| 5.1.6.2. | 選択フィールドの特殊な指定 | 134 |
| 5.1.6.3. | 西暦、月、日フィールドの変換動作 | 135 |
| 5.1.7. | コメントの追加 | 136 |
| 5.1.8. | 書類削除時の動作を指定 | 137 |
| 5.1.9. | 承認ステップにより更新する内容を変えたい場合の指定 | 138 |
| 5.2. | JavaScript を利用して kintone にレコードを追加・更新する | 140 |
| 5.2.1. | ナビゲータに「レコード追加」処理を追加 | 140 |
| 5.2.2. | 添付ファイルの送信 | 148 |
| 5.2.3. | コメントの送信 | 149 |
| 5.2.4. | ゲストスペースを利用するアプリへの登録 | 149 |
| 5.2.5. | Administrator 以外で登録する | 150 |
| 5.2.6. | ドメイン名が“cybozu.com”以外の場合 | 150 |
| 5.3. | kintone の登録レコードを削除する | 151 |
| 5.3.1. | ナビゲータに「レコード削除」処理を追加 | 151 |
| 5.4. | kintone アプリの登録レコードをマスタとして利用する | 152 |
| 5.4.1. | kintone 連携マスタの作成 | 152 |
| 5.4.2. | カスタム領域へのインポート | 153 |
| 5.4.2.1. | カスタム領域への TABLE 作成 | 153 |
| 5.4.2.2. | 固有マスタへのレコードインポート | 155 |
| 5.4.2.3. | インポートバッチの利用 | 159 |
| 5.4.3. | マスタ参照コンポーネントの作成 | 161 |
| 5.5. | kintone の登録レコードを検索する | 163 |
| 5.5.1. | アプリの準備 | 164 |
| 5.5.2. | フォームの修正 | 165 |
| 5.6. | 承認管理からの一括更新 | 167 |
| 第 6 章. | X-point 登録フォームから kintone アプリを作成する | 168 |
| 6.1. | kintone アプリ作成機能について | 168 |
| 6.2. | X-point から kintone アプリを作成する | 168 |
| 6.2.1. | kintone アプリ作成準備 | 169 |
| 6.2.2. | kintone アプリ作成実行 | 171 |
| 第 7 章. | kintone 登録アプリから X-point フォームを作成する | 172 |
| 7.1. | kintone フォーム作成機能について | 172 |
| 7.2. | kintone アプリ情報からフォームを作成する | 173 |
| 7.2.1. | kintone フォーム作成準備 | 174 |
| 7.2.2. | kintone フォーム作成実行 | 175 |
| 第 8 章. | CSV ファイルによる連携 | 177 |
| 8.1. | kintone の操作 | 177 |
| 8.1.1. | CSV ファイルのダウンロード | 177 |
| 8.1.2. | CSV ファイルのアップロード | 178 |

| | | |
|--------------|--------------------------------------|------------|
| 8.2. | X-point の操作・設定 | 179 |
| 8.2.1. | 自動申請の設定 | 179 |
| 8.2.2. | 自動申請の実行 | 179 |
| 8.2.3. | クエリ CSV 出力の設定 | 182 |
| 8.2.4. | クエリ CSV 出力 | 184 |
| 8.2.4.1. | クエリ CSV 出力 (ユーザーサイト) | 184 |
| 8.2.4.2. | クエリ CSV 出力 (管理サイト・タスク管理) | 185 |
| 8.2.4.3. | リモート出力 | 185 |
| 8.2.5. | X-point マスタの登録 | 186 |
| 第9章. | 連携 API 仕様 | 189 |
| 9.1. | フィールドの対応 | 189 |
| 9.2. | kintone 連携 「JavaScript 仕様」 | 191 |
| 9.2.1. | kintone 組み込みライブラリ仕様 | 191 |
| 9.2.1.1. | 連携クラス「kintone2xpoint」の仕様 | 192 |
| 9.2.2. | X-point 連携ライブラリ仕様 | 195 |
| 9.2.2.1. | 連携クラス「xpoint2kintone」の仕様 | 195 |
| 9.3. | kintone 連携「X-point REST API 仕様」 | 198 |
| 9.3.1. | kintone → X-point | 198 |
| 9.3.1.1. | 自動申請 (HTTP メソッド: POST) | 198 |
| 9.3.1.2. | 書類表示 (HTTP メソッド: GET) | 200 |
| 9.3.1.3. | フォーム (TSF) 雛形ダウンロード (HTTP メソッド: GET) | 201 |
| 9.3.2. | X-point → kintone | 204 |
| 9.3.2.1. | X-point からの kintone の REST API 実行ルール | 204 |
| 9.3.2.2. | 中継処理で発生するエラー | 205 |
| 9.3.2.3. | ファイルアップロード API 仕様 | 205 |
| 第10章. | その他 | 207 |
| 10.1. | X-point に接続できない場合の動作 | 207 |
| 10.2. | メンテナンスモード | 208 |

第1章. サイボウズ kintone連携サービスの概要

1.1. kintone 連携サービスとは

「サイボウズ kintone 連携サービス」は kintone アプリ、X-point の管理画面で相互登録の指定、もしくはフォームに JavaScript によるプログラムを記述することで、kintone アプリのレコード内容、X-point の書類内容を相互に送信できるようにするオプション・ライセンスです。

【主な機能】

- ① kintone アプリから X-point にレコードの内容を送り、新規書類の申請、既存書類の更新を行う事ができます。
- ② X-point の書類から kintone アプリに書類の内容を送り、レコードを追加、既存レコードの更新を行う事ができます。
管理サイトでの画面設定のみで連携を行うことも可能です。
- ③ kintone アプリから X-point に登録されている書類を表示することができます。(※)
- ④ kintone アプリの JavaScript より X-point の種類内容、承認進捗に関する情報を取得できます。
- ⑤ kintone アプリから出力した CSV ファイルを X-point の自動申請で一括申請できます (※)
- ⑥ X-point から出力した CSV ファイルを kintone アプリのファイル読み込み機能で一括追加できます。(※)

※ kintone は API・JavaScript が利用可能な「スタンダードコース」を利用する必要があります。

※ kintone より X-point に登録された書類を表示する際は、X-point のユーザーID・パスワードの入力、及び表示書類の閲覧権限が必要です。

※ 「サイボウズ kintone 連携サービス」を利用しない場合、自動申請用では書類データ交換用 CSV ファイルを、X-point 自動申請形式に修正する必要があります。

1.2. 連携可能な組み合わせ

kintone は「cybozu.com」上で提供されるサービスです。その為、kintone や X-point を利用する環境により動作に支障が生じる場合があります。kintone や X-point を利用する際にクライアント証明の利用、IP 接続制限などを行う場合は事前に動作環境の確認を行い運用可能な状態に設定の変更を行って下さい。

【利用環境の違いによる組み合わせ可否】

| X-point 利用環境 | | kintone 利用環境 | 利用可否 |
|-----------------------|-----------|---------------------|-------|
| X-point Cloud (https) | 標準利用 | kintone ブラウザ | ◎ (※) |
| | IP 接続制限 | kintone アプリ | ◎ (※) |
| | クライアント証明書 | kintone + IP アドレス制限 | ○ (※) |
| | | kintone + BASIC 認証 | ○ (※) |
| | | kintone + セキュア接続 | ○ (※) |

- ※ kintone で IP アドレス制限を利用する場合、X-point サーバーを IP アドレス制限の対象外に指定してください。
- ※ kintone で BASIC 認証、セキュア接続を利用する場合、サイボウズドットコム ストアにて X-point サーバーを IP アドレス制限の対象外に指定し、X-point サーバーからの通信で BASIC 認証、セキュア接続の対象外にします。
- ※ X-point サーバーを利用する際に HTTP リダイレクトを利用している環境では連携機能が正しく動作しません。リダイレクトを利用している場合、X-point 単独での利用には問題はありませんが、kintone との連携を行う場合にブラウザ仕様による制約が発生し連携機能が動作しません。
- ※ X-point Cloud を https プロトコルで接続するには「セキュリティ基本サービス」の契約が必須です。
- ※ ATLED Cloud の利用においてクライアント証明書を必要とする PC、スマートフォン、タブレットには必ずクライアント証明書を登録して下さい。PC、スマートフォン、タブレット上で X-point の画面を直接表示、利用しない場合であっても、kintone を利用するブラウザ、アプリから X-point をアクセスする際に必要になります。証明書が登録されていない場合、連携機能は動作しません。

1.3. 運用時のネットワーク構成

kintone 連携を行う場合、X-point サーバーは kintone サーバーに対して https プロトコルを利用した直接通信を行います。その為、X-point サーバーが kintone に対し行う通信の妨げとなる要因が運用ネットワークにあると連携できません。

1.3.1. IP アドレス接続制限

kintone のセキュリティ設定で IP アドレス制限を設定する場合、X-point サーバーの IP アドレスを許可する必要があります。IP アドレスは、X-point Cloud のサポート窓口にお問合せください。

なお、X-point サーバーが kintone サーバーに対し直接アクセスを行い IP アドレス接続制限設定の影響を受ける動作は次のような場合です。

- ① X-point 上で kintone のレコード登録・更新を行う REST API を操作する動作が行われるとき。
- ② X-point から添付ファイルを kintone に登録するとき。
- ③ kintone から添付ファイルをダウンロードし X-point の書類に添付する動作が行われるとき。
kintone が X-point に添付ファイルを登録する場合、kintone は X-point に対してキーコードのみを発行します。
その為、X-point が kintone から添付に必要となるファイルの取得を行うための動作です。
- ④ kintone からレコードに指定されたコメント情報を取得し X-point の書類に格納する動作が行われるとき。

※ X-point Cloud の サーバーIP アドレスは弊社サービスの都合により変更される場合があります。

※ kintone サーバーから X-point サーバーに直接アクセスするケースはありません。kintone アプリから X-point への通信は必ずブラウザを介して行われます。

※ kintone のレコード詳細を X-point に送信する動作はブラウザ・kintone アプリを介して行われ、添付ファイル、コメントは、X-point が受信したレコードの詳細に従って X-point が kintone に対して取得動作を行います。

1.3.2. プロキシサーバーや VPN の影響

JavaScript で kintone 連携機能を利用する場合、プロキシサーバーや VPN をご利用の場合は注意が必要です。

kintone 連携機能では X-point サーバーの URL を指定する必要があります。(3.1.3. 連携ライブラリの登録 など)

プロキシや VPN 環境下でも kintone/X-point 双方への接続が可能なおことをご確認ください。

1.4. 運用イメージ

「kintone 連携サービス」は、kintone アプリで登録した情報を元に、X-point に書類申請（提出）を行い、申請した書類を X-point 上でワークフロー処理したのち、結果を再び kintone アプリに戻し、kintone に登録されているレコードを最新情報に更新するという業務を想定し用意されています。本章はこの流れに沿って連携仕様の解説を行います。



cybozu.com

契約書・社内審査

契約書・社内審査 > (すべて) > 1

顧客コード: G017 顧客名: 株式会社紙沢山商事

主担当: 藤 太郎

審査完了希望日: 2017-02-01 管理コード: P018、ECOプロジェクト

審査主旨: ゼロエミッション実現に伴うリサイクル事業者決定に伴う契約

| 契約書番号 | 契約書名 |
|--------|----------|
| S00123 | 資材循環取扱契約 |

審査日: 審査No

添付ファイル

kintone から
提出

転記



契約書 社内審査

申請日: 2017年02月01日
申請者: 藤 太郎
審査No: KS-2017-00001

| 項目 | 上長承認 | 発行 | 承認 | 上長承認 | 発行 | 上長承認 |
|---------|------------------------------|-----------|----|------|----|------|
| 顧客名 | G017 | 株式会社紙沢山商事 | | | | |
| 主担当 | 藤 太郎 | | | | | |
| 審査完了希望日 | 2017年02月01日 | | | | | |
| 管理コード | P018、ECOプロジェクト | | | | | |
| 審査主旨 | ゼロエミッション実現に伴うリサイクル事業者決定に伴う契約 | | | | | |
| 契約書番号 | S00123 | | | | | |
| 契約書名 | 資材循環取扱契約 | | | | | |

書類にkintone の情報を
セットし申請します。

完了時に値を戻せるよ
う、kintone のレコード
ID を保存します。

承認

cybozu.com

契約書・社内審査

契約書・社内審査 > (すべて) > 1

顧客コード: G017 顧客名: 株式会社紙沢山商事

主担当: 藤 太郎

審査完了希望日: 2017-02-01 管理コード: P018、ECOプロジェクト

審査主旨: ゼロエミッション実現に伴うリサイクル事業者決定に伴う契約

| 契約書番号 | 契約書名 |
|--------|----------|
| S00123 | 資材循環取扱契約 |

審査日: 2017-02-10 審査No: KS-2017-00001

添付ファイル

完了時に
kintone を更新

採番の転記と
完了日の反映

契約書 社内審査

申請日: 2017年02月01日
申請者: 藤 太郎
審査No: KS-2017-00001

| 項目 | 上長承認 | 発行 | 承認 | 上長承認 | 発行 | 上長承認 |
|---------|------------------------------|-----------|----|------|----|------|
| 顧客名 | G017 | 株式会社紙沢山商事 | | | | |
| 主担当 | 藤 太郎 | | | | | |
| 審査完了希望日 | 2017年02月01日 | | | | | |
| 管理コード | P018、ECOプロジェクト | | | | | |
| 審査主旨 | ゼロエミッション実現に伴うリサイクル事業者決定に伴う契約 | | | | | |
| 契約書番号 | S00123 | | | | | |
| 契約書名 | 資材循環取扱契約 | | | | | |

保存していたkintone の
レコード ID を使い、
kintone に書類の内容を
送信し、kintone の登録
情報を更新します。

kintone 上で「審査中」
を「完了」にするなど。

第2章. 連携の仕様

2.1. ユーザーの考え方

kintone・X-point の連携では一方のみにユーザーが存在する状態であっても運用が可能です。

但し、連携時の指定方法、各製品の仕様により kintone、X-point の双方にログイン名を持つユーザーであっても、連携設定の方法により、登録・申請時のユーザーが操作を行ったユーザーにならない場合がありますので注意してください。

2.1.1. kintone → X-point (書類提出)

書類提出を行う際の kintone ログイン名と X-point ログイン ID の関係について説明します。

| X-point 側ユーザー指定 | 書類の提出ユーザー |
|----------------------------|---|
| API の実行時に 特に指定をしない | 提出を実行した kintone アプリのログイン名とユーザーCD が一致するユーザーで書類の提出を行います。一致するユーザーが存在しない場合、書類の提出は行いません。 ※ kintone を利用するユーザーが X-point に kintone と同じログイン名・ユーザーCD でアカウント登録されている場合に利用できます。 |
| API の実行時に 代理ユーザーCD 指定あり | 提出を実行した kintone アプリのログイン名とユーザーCD が一致するユーザーが存在する場合、一致したユーザーで書類の提出を行います。 一致するユーザーが存在しない場合、代理ユーザーCD で指定されたユーザー名で提出を行います。 ※ X-point にアカウントを持たないユーザーがいる場合に使用します。 |
| API の実行時に 特定ユーザーCD 指定あり | 常に指定されたユーザーCD のユーザーで書類の提出を行います。 ※ kintone を利用するユーザーが X-point のアカウントを持たない場合に使用します。 X-point に書類の申請を行う場合には、kintone のユーザー名を書類に含めるなどの工夫が必要になります。 |

書類の提出では、kintone のレコードに配置されている添付ファイルフィールドの内容を X-point の添付書類に、レコードのコメントを書類のコメント欄に転記することができます。

添付書類の操作、コメントの転記は連携 API の呼び出し時に実施する・しないの制御ができます。

なお、X-point に申請されると X-point の通知メールポリシーの指定に従い、提出後の承認ステップ、事前通知先に対して通知メールが送信されます。通知メールの送信有無は制御できません。

！注意事項

- ※ 書類作成・編集の際に X-point に添付する書類は 1 バイト以上、X-point の書類添付サイズ上限未満である必要があります。
kintone で許可されている 0 バイトファイルを送るとエラー扱いになり書類の提出・編集は出来ません。

2.1.2. X-point → kintone（レコード追加）

X-point より kintone アプリにレコードを追加・更新する方法は、管理サイトから画面設定で行う方法と、フォームに JavaScript を読み込ませて行う方法の 2 通りがありますが、画面設定による連携を推奨しています。

2.1.2.1. 連携設定による連携

管理サイト「フォーム管理」にて行う kintone 連携により作成される kintone レコードの作成者、更新者の記録は管理者サイトのドメイン管理「kintone 連携」、ユーザーサイトの個人設定「kintone 連携アカウント」の指定内容により記録方法が確定します。

■ ドメイン管理「kintone 連携」（管理サイト）

フォーム管理で kintone アプリとの連携を行うフォームに対して設定した内容に基づき、ワークフロー処理エンジンより実行された場合の認証タイプを指定します。X-point で利用するフォームのナビボタンに組み込む JavaScript API を利用した場合は設定に依存せず API の呼び出し指定に従います。

| | | | | | | | |
|--------------------------|--------------------------------------|--|-----------|--------|-----|--------|--------|
| ドメイン管理 | | →ドメイン | →フォーム関連情報 | →メール設定 | →ログ | →タスク管理 | →定義書出力 |
| kintone連携・設定情報 | | | | | | | |
| kintoneアプリのAPI認証方法を指定します | | | | | | | |
| kintone連携・設定情報 | | | | | | | |
| ユーザー認証 | <input type="radio"/> 必須 | フロント機能【個人設定】で「kintone連携情報」のアカウントが未登録の場合は連携動作せずエラー停止します | | | | | |
| | <input type="radio"/> する | フロント機能【個人設定】で「kintone連携情報」のアカウントが未登録の場合はAPIトークン認証で動作します。 | | | | | |
| | <input checked="" type="radio"/> しない | 連携動作は常にAPIトークン認証になります フロント機能【個人設定】に「kintone連携アカウント」指定項目が表示されなくなります。 | | | | | |

※ 初期値は「しない」になります。「しない」の場合は、API トークン認証のみが実施されます。

【認証の違い】

1. 必須

kintone アプリに接続する際、個人設定に登録されている kintone サーバーのログイン名、パスワードを使用します。指定されているログイン情報で kintone への認証が成功する場合は連携が行われ、認証に失敗する場合は連携フォームの提出・更新操作は中断され書類の提出・更新、kintone へのレコード登録ができません。kintone へのレコード追加・更新を行うユーザー全員が kintone のアカウントを個人設定で登録している必要があります。cybozu.com の監査ログ（システム管理）には認証を行ったユーザーで記録が行われます。

2. する

「必須」、「しない」の両方の動作をあわせ持ったモードになります。個人設定に kintone サーバーのログイン名、パスワードが登録されている場合は登録内容で連携し、未登録の場合は API トークンで認証を行います。その為、cybozu.com の監査ログ（システム管理）への記録は、認証を行ったユーザー名、もしくは“Administrator”になります。kintone アプリのレコード作成者、更新者も同様に ユーザー名もしくは“Administrator”が記録されます。

3. しない

連携動作は常に連携フォームに指定されている kintone アプリの API トークンで行われます。kintone アプリのレコード作成者、更新者の記録は“Administrator”になります。kintone へのレコード追加・更新を行う X-point の書類を利用するユーザーが kintone のログイン名を持たない場合、kintone の利用が一部の部門に限定されるような場合に利用します。cybozu.com の監査ログ（システム管理）には“Administrator”で記録が行われます。

■ 個人設定「kintone 連携アカウント」（ユーザーサイト）

ドメイン管理の「kintone 連携」で認証タイプを“必須”、“する”に指定した場合、X-point の利用ユーザー毎に設定を行います。

| kintone連携アカウント | | |
|----------------|----------|---|
| 1 | サーバーFQDN | devetfnsn.cybozu.com |
| | ログイン名 | <input type="text"/> |
| | パスワード | <input type="password"/> (※)ログイン名を指定する場合は必須項目になります。 |

※連携用アカウントを指定しない場合、kintoneアプリと連携するフォームの提出・保存が出来ない場合があります。

ユーザ認証タイプが「必須」である場合は、ログイン名、パスワードを必ず指定する必要があります。

認証タイプが「する」になっている場合は、ログイン名、パスワードの欄が空欄であっても kintone との連携は動作しますが、kintone アプリのレコード作成者、更新者、及び監査ログの記録は“Administrator”になります。

認証タイプが「しない」に指定されている場合「kintone 連携アカウント」の設定項目は表示されません。

連携先の kintone が複数ある場合（異なる FQDN で API トークンの発行登録がある場合）は、kintoneFQDN 毎に複数のアカウント登録画面が表示されます。

！ 注意事項

- 1) X-point から kintone への連携では kintone の「kintone REST API」が使用され、認証動作は「パスワード認証」、「API トークン認証」で行われます。X-point と kintone (cybozu.com) が SAML 認証で同じ IdP を利用している場合であっても、kintone 連携は SAML とは別の認証が行われることに注意してください。
- 2) ドメイン管理>ドメイン設定>一般ユーザーによる個人設定の許可 にて「プロフィール」を「許可しない」に指定した場合であっても、kintone 連携アカウントの入力修正は禁止できません。編集を禁止する場合は「kintone 認証設定」にて「ユーザ認証」を「しない」に設定して下さい。その場合、連携は常に Administrator で行われます。

2.1.2.2. JavaScript による連携

フォームナビボタンに kintone 連携 JavaScript API を組み込み連携する場合の、kintone レコード作成者、更新者の記録になります。

| 連携時の指定 | 追加レコードの登録・更新ユーザー |
|---------------|---|
| 指定無し | 常に Administrator で登録・更新されます。 ※ kintone の API トークン認証で動作します。 |
| 特定ユーザーCD 指定あり | 指定ユーザーになります。(API のオプション項目で指定) ※ kintone のユーザー認証で動作します。 |

レコードの追加では、X-point の添付書類、コメントも送信することができます。

- ・添付書類： 付け替え
- ・コメント： 追加

添付書類の操作、コメントの転記は連携 API の呼び出し時に実施する・しないの制御ができます。

2.1.3. kintone アプリから指定書類をブラウザ表示

連携 API の呼び出し時に指定した条件で書類が書類 ID 等により一意決定される場合に限り書類の閲覧画面が表示する事が出来ます。実際に書類を表示するためには、X-point に登録されているユーザーアカウント、及びログイン時に使用したユーザーアカウントに表示する書類の閲覧権限が与えられている必要があります。

2.2. kintone からの操作

2.2.1. X-point に書類を申請・更新

kintone アプリから X-point に書類の提出を行う事ができます。提出する X-point のフォームには必ず、kintone アプリのレコード ID を保存するフィールドを持つ必要があります。

■ 利用の前提条件

- 連携するフォームに kintone 連携用の自動申請登録が必要です。
書類に登録される項目値は自動申請機能を介して登録が行われます。その為、登録内容には自動申請と同じ制限事項が発生します。詳細については自動申請機能に関するマニュアル記載をご覧ください。
- kintone アプリから X-point の連携 API を利用するためには X-point 管理サイトの kintone 連携設定にて「追加」権限を持つ API トークンを発行する必要があります。
- X-point から kintone のレコード情報が参照できるよう kintone アプリに「閲覧」が可能な API トークンを発行する必要があります。
- 書類を更新する場合、連携 API を呼び出す際、X-point の書類 ID を指定する必要があります。
書類 ID を指定できない場合は更新することは出来ません。なお、書類 ID は X-point に申請を行う際に取得できます。
(書類の更新可否は書類の状態に依存します)

■ 表定義の利用

- X-point に提出する書類に含めるテーブル定義は 1 個まで、複数のテーブル定義は指定できません。
- X-point に送信する表定義の行数は X-point のフォーム設計で指定されている表行数以下になります。
フォームの設計行を超えて連携データを送信すると書類の申請はできません。kintone から送信する書類データを作る際に書類に格納可能な行数にして下さい。

■ ファイルの添付

X-point への書類申請、書類内容の更新の際に kintone アプリで保持している添付ファイルを書類添付することが出来ます。添付可能なファイルは X-point の仕様に従い、書類ナビボタンでの添付と同等になります。ファイル上限も X-point に依存します。

■ 「send_xpoint」 API を利用した場合の戻り値

書類の提出・更新結果を JSON 形式の応答で返す「send_xpoint」を利用した場合、応答に X-point の書類 ID が含まれます。

【成功例】・・・新規に書類提出を行い書類 ID が採番された場合の応答

```
{
  "docs": [
    {
      "kintone_rec_no": "159",
      "docid": 2792,
      "message_type": 3,
      "message": "書類が提出されました (ID = 2792)"
    }
  ]
}
```

【成功例】・・・既存書類を更新した場合の応答

```
{
  "docs": [
    {
      "kintone_rec_no": "34",
      "docid": 2595,
      "message_type": 3,
      "message": "書類が更新されました (ID = 2595)"
    }
  ]
}
```

【失敗例】・・・更新できない状態の書類を更新した場合の応答

```
{
  "docs": [
    {
      "kintone_rec_no": "159",
      "docid": -1,
      "message_type": 1,
      "message": "指定された書類 ID[2792]はワークフロー処理中のため更新できません。"
    }
  ]
}
```

【失敗例】・・・存在しない書類 ID を指定した場合の応答。

```
{
  "docs": [
    {
      "kintone_rec_no": "159",
      "docid": -1,
      "message_type": 1,
      "message": "指定された書類 ID[2792]は登録されていません。"
    }
  ]
}
```

【失敗例】・・・提出書類の内容に問題があり提出できない場合の応答（申請エラー）

```
{
  "docs": [
    {
      "kintone_rec_no": "160",
      "docid": -1,
      "message_type": 1,
      "message": "書類の提出処理中にエラーが発生しました。No. 1: Error: 明細データが表定義グループの行数を超えています"
    }
  ]
}
```

■ その他留意事項

- X-point に申請できる書類は“ワークフローフォーム”、“通常フォーム”です。
- X-point に対して実行できる操作は書類申請、書類の更新、削除です。
但し、書類の更新、削除は書類の状態が特定の条件に合致している必要があります。
- X-point にナビボタンからの申請・更新を行った際、なんらかの原因でエラーが発生し提出・更新が完了しない場合、利用する API の種類により kintone 側でエラーの発生を検知する事はできません。
 - ※ doEntry を利用する API . . . エラー検知は出来ない
 - ※ doEntryREST を利用する API . . . エラー検知できる
- 書類更新は、完了書類の編集が許可されている場合、提出直後の書類で承認が全く行われていない場合、書類が【下書き】状態になっている場合に限定されます。

2.2.2. X-point の書類を削除

kintone アプリより X-point の書類 ID を指定することで書類の削除を行う事が出来ます。

書類の削除は API を呼び出す際に指定したユーザーが対象となる書類の削除権限を有している場合に限り実行されます。

■ 削除できる書類

1. API を実行する際に指定されたユーザーが対象書類のアクセス権限で削除が許可されていること
2. 対象書類が削除実施可能な状態であること。

■ 削除 API 利用時の戻り値

【正常処理時】 . . . 空の JSON 応答が返されます

```
{}
```

【エラー発生時①】 . . . 削除できない書類に削除 API を実行した場合（承認中など）

```
{
  "kintone_rec_no": "",
  "docid": 2794,
  "message_type": 1,
  "message": "X-point 書類[2794]を削除できません。削除権限が無いか削除できない書類状態です"
}
```

【エラー発生時②】 . . . 削除済みの書類 ID を指定した場合

```
{
  "kintone_rec_no": "",
  "docid": 2793,
  "message_type": 1,
  "message": "X-point 書類[2793]は削除済み、もしくは存在しない書類です"
}
```

【エラー発生時③】 . . . 書類 ID の指定をしなかった場合

```
{
  "kintone_rec_no": "",
  "docid": -1,
  "message_type": 1,
  "message": "削除対象となる書類 ID が指定されていません。"
}
```

2.2.3. 書類の内容を取得

指定された書類の内容を JSON 形式の応答で取得できます。

添付ファイルに関する情報も取得できます。添付ファイルを実際にダウンロードする場合は書類情報に含まれるファイル取得キーを使用しファイルダウンロード API を実行します。

■取得できるデータ

1. 書類の内容
2. 承認状況（現在の状態、現在ステップ番号、版番号）
3. 申請者（所属グループ CD、所属グループ名、申請者名、申請者 CD、申請日）
4. 最終承認者（所属グループ CD、所属グループ名、承認者名、承認者 CD、承認日）
5. 添付ファイル（content-type、ファイル名、ファイルサイズ、取得用キーCD）
6. コメント

2.2.4. 書類の承認状況を取得

指定された書類の承認状況を JSON 形式のデータとして取得できます。

■取得できるデータに含まれる項目

1. 承認状況（現在の状態、現在ステップ）
2. フォームコード、フォーム名、フォームタイプ（ワークフロー、標準）
3. 承認ルートコード、ルート名
4. 件名 1、件名 2
5. 申請者情報
6. 最終処理者情報
7. 承認履歴（版情報を含む全て）

2.2.5. 書類の承認状況を表示

指定された書類の承認状況画面を表示します。表示される画面は X-point の承認状況を表す画面とほぼ同じ内容の画面が表示されます。

■表示画面のイメージ

| 承認進捗状況 (書類No.26180) | | | |
|---------------------|-----------|-------|---------------------|
| 件名1 | 営業部 | | |
| 件名2 | 営業太郎 | | |
| フォーム | A368出張申請書 | 承認ルート | 規定のルート |
| 承認状況 | 承認中 | 申請日 | 2020/10/30 14:14:45 |

| 第1版 | |
|--|--|
| 申請  | 承認  |
| 申請 | STEP 1 |
| 管理者 未所属 20/10/30 14:14:45 | AND (2) 佐伯 宗一 (未処理) 株式会社えいとれっど商事 役員 木々 和明 (未処理) 株式会社えいとれっど商事 社長 |

2.2.6. 書類の添付ファイルを取得

書類内容の取得で得られたキー (fileKey) を指定することで特定書類の添付ファイルをダウンロードする事が出来ます。

2.2.7. 承認待ち件数を取得

ユーザフロント画面の承認ガジェットに表示される内容と同じ各状態の書類件数、及び承認待ち書類（5件）を取得できます。但し、フォームグループ、フォームの絞込が必須になるため、特定の書類に係わる件数に限定され、X-point に接続する際に使用した認証アカウントの承認待ち件数になります。

kintone アプリに関連するフォームの承認中件数などを表す必要がある場合などに利用します。

2.2.8. 書類を一括申請

kintone のレコード一覧画面で利用できる event オブジェクトを利用し複数の書類をまとめて申請する場合に利用します。

レコード一覧表示イベント（app.record.index.show）等で利用でき、複数の書類情報の一つにまとめて X-point に送信します。基本的には 1 件毎の申請情報をまとめて送信するか否かの違いのみで、X-point との通信回数、通信に伴うオーバーヘッド（パラメータなど）が減少しますが、1 件毎の申請を処理する時間は申請数で比例するため、処理件数によってはブラウザタイムアウトが発生することがあります。

| | レコード番号 | 顧客コード | 顧客名 | 主担当 | 審査完了希望日 | 管理コード | 審査主旨 | 審査No | 添付ファイル |
|---|--------|-------|------|---------|------------|-----------|---------|--------|-----------------------------|
| 3 | 87 | G018 | 株式会社 | 1 勝 太郎 | 2017-04-21 | KANRI-000 | X-point | KS-201 | _郵便番号マスタ_20170125T125605+09 |
| | 47 | G018 | 株式会社 | 1 木戸 武史 | 2017-04-21 | KANRI-000 | X-point | KS-201 | KeiyakuShinsa.tsf |
| | 38 | G001 | 株式会社 | 1 木戸 武史 | 1970-01-01 | | 「太陽が | KS-201 | |
| | 35 | G001 | 株式会社 | 1 木戸 武史 | 1970-01-01 | A1 | 「太陽が | KS-201 | |
| | 34 | G002 | 富士山 | 1 勝 太郎 | 1970-01-01 | A1 | 上書き申 | KS-201 | |

！ 注意事項

※ 書類の一括申請は、PC 用 kintone 画面のみで実行できます。スマートフォン用 kintone 画面では利用できません。

これは kintone アプリのスマートフォン用イベント（mobile.app.record.index.show）で取得できるレコード情報の仕様によるものです。

2.2.9. kintone アプリのフォームから TSF ファイルを作成

kintone アプリのフォーム情報を取得し、X-point で対応可能なフィールドを含んだ TSF ファイルを作成します。フィールドのレイアウトは行われず、kintone API で取得した内容に基づいて順にフィールドを配置します。作成方法は 9.3.1.3 「フォーム (TSF) 雛型ダウンロード (HTTP メソッド: GET)」をご参照ください。

【kintone アプリのフォーム例】

The screenshot shows a kintone form with various fields. A red box highlights the '契約書番号' (Contract Number) and '契約書名' (Contract Name) fields. Other fields include '顧客コード', '顧客名', '主担当', '審査完了希望日', '管理コード', '審査主旨', '審査日', '審査No', '書類ID', '添付ファイル', '押印済み契約書', '作成日時', '作成者', '更新者', and '更新日時'.

【出力 TSF ファイル例】

The screenshot shows a grid representing the output of a TSF file. The grid has columns labeled 0, 1, 2, and 3. A red box highlights the '契約書番号' and '契約書名' fields. Other fields include '更新者', '作成者', '更新日時', '作成日時', '顧客コード', '顧客名', '主担当', '審査完了', '管理コード', '審査主旨', '審査No', '審査日', and '書類ID'.

- ※ フィールドを TSF にする際、ラベルとフィールドの 2 つになります。但し、ラベルに限りラベル 1 つに変換されます。
- ※ 赤枠部分はテーブルを表明細定義 (3 行分) に変換している部分。
- ※ 添付ファイルフィールド (添付ファイル、押印済み契約書) は変換されません。
- ※ TSF に作成されるフィールドのサイズは固定です。
- ※ TSF のフィールド並びは、kintone から返されるフォーム情報の並びになります。TSF 変換出力の際に順番を指定することはできません。

【アプリ項目、フィールドの変換】

| kintone アプリ項目／フィールド種類 | | X-point TSF 項目種類 | 備考 |
|-----------------------|-----------|------------------|--|
| アプリ名 | | フォーム名 | |
| アプリコード | | フォームコード | 未指定の場合は「NewForm」 |
| | | TSF ファイル名 | アプリコード+ “.tsf” 未指定の場合は「NewForm.tsf」 |
| アプリの説明 | | 備考 | 「アプリの説明」に記載の内容を転記 |
| フィールド (変換されるもの) | ラベル | ラベルフィールド | 初期値はタグを取り除いて設定 |
| | 文字列 (1行) | 文字フィールド | |
| | 文字列 (複数行) | テキストエリア | |
| | リッチエディター | テキストエリア | 初期値はタグを取り除いて設定 |
| | 数値 | 数値フィールド | 整数フィールドには変換しない |
| | 計算 | | 計算式は変換されません |
| | ラジオボタン | ラジオボタン・グループ | ラジオボタン・グループに変換します |
| | チェックボックス | 複数のチェックボックス | 項目を分解し複数のチェックボックスフィールドに変換 |
| | 日付 | 日時型フィールド | 日付「yyyy/mm/dd」に変換 |
| | 時刻 | 日時型フィールド | 時刻「hh:mm」に変換 |
| | 日時 | 文字フィールド | 文字フィールドで作成 (書式なし) |
| | 複数選択 | リストボックス | 表示行数 4、選択列数 1 で変換します |
| | ドロップダウン | コンボボックス | 選択列数 1 で変換します |
| | リンク | 文字フィールド | |
| | ユーザー選択 | 文字フィールド | |
| | 組織選択 | 文字フィールド | |
| | グループ選択 | 文字フィールド | |
| | ルックアップ | 文字フィールド | |
| | テーブル | 表明細定義 | 3行分の表明細定義に変換 |
| | レコード番号 | 文字フィールド | |
| | 作成者 | 文字フィールド | |
| | 作成日時 | 文字フィールド | |
| | 更新者 | 文字フィールド | |
| 更新日時 | 文字フィールド | | |
| フィールド (変換されないもの) | 添付ファイル | 変換無し | 該当フィールド型が無いため無視します |
| | 関連レコード | 変換無し | |
| | スペース | 変換無し | |
| | 罫線 | 変換無し | |
| | グループ | 変換無し | グループ内に配置したフィールドは変換されません。グループ解除されます。 |

2.2.10. kintone アプリから自動申請用の CSV を出力

kintone からの CSV ファイル出力はブラウザのアプリ画面、もしくはコマンドラインで操作する「コマンドラインツール」から行います。ブラウザ画面からの出力は都度、出力する項目の選択、項目並びの指定を行う必要があるため効率的ではありません。その為、「コマンドラインツール」での作業を基本とします。

コマンドラインツールの利用方法は kintone のサイトをご確認ください。

【コマンドラインツール実行の例】

- ① アプリ ID=12 からユーザーID「administrator」、パスワード「atled@kintone」で CSV ファイルを取得

```
cli-kintone -d ksubdomain.cybozu.com -a 12 -u administrator -p atled@kintone -o csv
```

- ② アプリ ID=12 からアプリのトークンコードを指定して、レコード "\$id", "name", "date" の 3 列を取得

```
cli-kintone -d ksubdomain.cybozu.com -a 12 -t diVUoVQDgnnU7r01v4sFyfUiGnX0ResLGg0fIk1s -c "$id,name,date" -o csv
```

- ③ アプリ ID=12 からアプリのトークンコードを指定して、レコード "\$id", "name" の 2 列に加えテーブル "group1" を取得。

```
cli-kintone -d ksubdomain.cybozu.com -a 12 -t diVUoVQDgnnU7r01v4sFyfUiGnX0ResLGg0fIk1s  
-c "$id,name,date,group1" -o csv
```

※ 取得する CSV ファイルはテーブルの行数分になり、CSV の先頭には 1 行目を示す "*" が追加されます。

2.2.11. X-point の書類を開く

書類を開く API の引数に、指定フォームで登録されている書類を一意に特定する条件が必要です。

書類を表示するウィンドウは必ず「xpform」という名称 (target 属性) が指定されます。そのため、書類を表示したウィンドウを開いたままの状態では他の書類を表示すると、最初に開いた書類と同じウィンドウ上に表示されます。

なお、書類を表示する際に X-point のセッションがない場合はログイン認証画面が表示され、書類を表示する際に既存セッションが存在する場合は、閲覧に指定されたログイン名と既存セッションのログイン名が一致する場合はそのまま表示。不一致の場合は、既存セッションを強制的にログアウトし、閲覧用ブラウザ画面で認証操作を行います。

2.2.12. kintone から承認する（ワークフロー操作）

kintone から連携 API を介して X-point 上のワークフロー書類に対して承認操作を行う事が出来ます。操作を行う場合は書類操作が可能である条件をクリアしている必要があります。

【条件】

- ① ワークフロー操作の連携 API を実行する際に指定されるユーザーが X-point に登録されている事。
- ② 連携 API に指定されているユーザーが、操作の対象となる書類を操作できるステップに存在する事。
- ③ 書類の状態がワークフロー操作を受け付ける状態である事。
- ④ 未定義ステップの候補選択が連携 API の実行時（提出時、承認時）に必要なこと。

■ ワークフロー操作と操作の可否

| 操作の種類 | API メソッド | 説明 |
|---------|------------|--|
| 承認 | doAprv | 現在ステップに指定ユーザーが存在する場合にユーザーの状態を「承認済み」にします。 |
| 差し戻し | doBack | 現在ステップに指定ユーザーが存在する場合に指定されたステップ番号（0番は申請ステップ）に書類を差し戻します。実行時に“差し戻しコメント”、“差し戻し先ステップ番号”を指定する必要があります。 承認ルートの付与権限で差し戻し禁止が指定されたステップでは実行できません。 |
| 保留 | doSuspend | 現在ステップに指定ユーザーが存在する場合に書類を保留状態にします。 |
| 却下 | doReject | 現在ステップに指定ユーザーが存在する場合に書類を却下状態にします。 実行時には“却下コメント”を指定する必要があります。 承認ルートの付与権で却下禁止が指定されたステップでは実行できません。 |
| 承認取戻し | doAprvBack | 現在ステップのユーザーが全て未処理であり、かつ直前のステップに指定ユーザーが存在する場合、直前のステップで書類の状態を承認待ちに戻します。 |
| 申請取戻し | | 提出後の書類に対し、承認、保留などの操作が一切行われていない場合に、書類を下書き状態にします。 ※ 【承認】→【承認取戻し】は操作が行われた事になります。 ※ 申請者への差し戻しがある場合の判定は同一版内での承認などの操作有無になります |
| 完了書類取戻し | | 対象書類の完了取戻しが許可されており、かつ書類取戻しの条件が成立する場合に、書類を承認中に戻します。 |
| 下書き | doDraft | X-point 上に書類を「下書き」状態で新規作成します。 ※ レコード明細に X-point の書類 ID を格納している場合は既存の下書きを更新します。 |
| 提出 | doEntry | X-point 上で「下書き」状態の書類を提出（承認中）にします。 |

※ 全てのワークフロー操作は X-point の書類ビューで行う場合と同じ仕様で動作します。

※ 「承認」、「差し戻し」、「保留」、「却下」の動作は常に、連携フィールドの上書きののちワークフロー操作が行われ、承認ルートの付与権限で指定された「編集できる」の指定は無視されます。

※ 未定義ステップのユーザー指定、閲覧者追加を連携 API から行うことは出来ません。

※ 書類申請メソッド「sendDetailPage」を利用する場合、既存書類が無い場合は「下書き+提出」に相当する動作になります。既存書類が下書き状態で存在する場合は下書き書類の更新になります。

■kintone からの API を実行する際の記述例

アプリのレコード明細に【承認】ボタンを配置する場合のクリックイベント記述例になります。

```
(function () {
  "use strict";

  kintone.events.on(' app.record.detail.show', function (event) {
    var myAprvButton = document.createElement(' button');
    myAprvButton.id = ' btn_wf_aprv';
    myAprvButton.innerHTML = ' 【承認】';
    myAprvButton.onclick = function () {
      var k2x=KeiyakuShinsa("KeiyakuShinsaR5ZF8B!j8srS8ej1EKH!a#zFKEA", "KeiyakuShinsa");
      k2x.doAprv( event );
    };
    kintone.app.record.getHeaderMenuSpaceElement().appendChild(myAprvButton);
  });
})();
```

※ フォームコード「KeiyakuShinsa」、承認ルートコード「KeiyakuShinsa」の場合。

※ 【承認】以外のワークフロー操作で【保留】【取戻し】【提出】【下書き】は記述例の「k2x.doAprv(event)」のメソッド名「doAprv」を対応の操作 (doSuspend, doAprvBack, doEntry, doDraft) に置き換えます。

アプリのレコード明細に【差し戻し】ボタンを配置する場合のクリックイベント記述例になります。

```
(function () {
  "use strict";

  kintone.events.on(' app.record.detail.show', function (event) {
    var myBackButton = document.createElement(' button');
    myBackButton.id = ' my_index_wf_back';
    myBackButton.innerHTML = ' 【差し戻し】';
    myBackButton.onclick = function () {
      var k2x=KeiyakuShinsa("KeiyakuShinsaR5ZF8B!j8srS8ej1EKH!a#zFKEA", "KeiyakuShinsa");
      k2x.wf_comment=' 差し戻し' します';
      k2x.wf_backstep=0; // 差し戻し先のステップ
      k2x.doBack( event );
    };
    kintone.app.record.getHeaderMenuSpaceElement().appendChild(myBackButton);
  });
})();
```

※ フォームコード「KeiyakuShinsa」、承認ルートコード「KeiyakuShinsa」の場合。

※ 差し戻しコメント、差し戻し先は連携インスタンスのプロパティとして指定します。指定は省略できません。

アプリのレコード明細に【却下】ボタンを配置する場合のクリックイベント記述例になります。

```
(function () {
  "use strict";

  kintone.events.on(' app.record.detail.show', function (event) {
    var myRejectButton = document.createElement(' button');
    myRejectButton.id = ' my_index_wf_reject';
    myRejectButton.innerHTML = ' 【却下】';
    myRejectButton.onclick = function () {
      var k2x=KeiyakuShinsa("KeiyakuShinsaR5ZF8B!j8srS8ej1EKH!a#zFKEA", "KeiyakuShinsa");
      k2x.wf_comment=' 却下' します';
      k2x.doReject( event );
    };
    kintone.app.record.getHeaderMenuSpaceElement().appendChild(myRejectButton);
  });
})();
```

※ フォームコード「KeiyakuShinsa」、承認ルートコード「KeiyakuShinsa」の場合。

※ 却下コメントは連携インスタンスのプロパティとして指定します。指定は省略できません。

2.3. X-point からの操作

X-point に登録されている書類を kintone アプリのレコードとして新規追加、あるいはレコード内容の更新を行うことができます。更新の方法は、連携内容の指定を画面から行う方法（プログラムレス）、書類ビューのナビボタンに JavaScript プログラムで組み込む方法、クエリ CSV 出力機能を利用して kintone アプリにレコードを取り込む方法（新規登録のみ）の 3 種類が用意されています。

2.3.1. ワークフロー処理から kintone アプリにレコードを追加・更新

フォーム管理「kintone 連携」設定にて X-point 上で書類のワークフロー処理と連動した動作の設定を行うと、X-point の書類操作と連動して kintone に対するデータ更新をプログラムレスで実現することが可能です。

指定は「kintone 連携」設定画面下側の、X-point から kintone アプリへのレコード追加・更新設定項目で行います。

■ X-point から kintone への連携を指定する画面

【kintone アプリ情報】

書類データを追加・更新する先となる kintone アプリの情報を指定します。

- ドメイン、アプリ ID、API トークン、ゲスト ID
※ API トークンに「レコード閲覧」、「レコード追加」、「レコード編集」のアクセス権が必要です。

【連携基本情報】

- 「kintone レコード ID」書類内保存フィールド
X-point の書類内で kintone アプリのレコード ID を記録するフィールド ID を指定します。フィールドに値が入力されている場合は、kintone に送信するデータにレコード ID として設定されるため kintone のレコードを更新する動作になります。値が未入力の場合、kintone 連携は新規レコード追加で実行されレコード追加の結果で得られる情報をレコード番号として指定フィールド ID の値として書類内に保持します。
保存フィールドを指定しない場合は、常に新規レコード追加の状態での kintone アプリとの連携が動作します。
- 添付ファイルの更新
連携動作の際に、kintone アプリの指定添付ファイルフィールドに書類に添付されているファイルを全てセットします。連携前に添付フィールドに設定されていたファイルは全て置換されます。

- コメントの追加
連携動作の際、書類に記入されているコメント全てが、kintone の対象レコードに追加されます。
kintone アプリのレコードに記入済みのコメントと記入日、内容が一致するコメントは追加されません。
- 書類削除と連動
X-point 上で書類が削除された際に、kintone アプリのレコードに削除を実行します。
※ 書類上に「kintone レコード ID」フィールドが保持され「kintone レコード ID」書類内保存フィールド」に指定されている必要があります。
※ 有効にする場合、連携先 kintone の API トークンに「レコード削除」のアクセス権が必要です。
※ ユーザフロント画面の承認>下書き、検索書類のプロパティにある【削除】、管理サイトの承認状況検索で行う削除の3操作が対象になります。
- 完了書類の PDF を添付
ワークフロー書類の承認完了、承認完了書類の編集保存、通常フォームの提出、通常フォームの修正を実行した際に、書類が PDF 化され、指定されている kintone アプリの添付ファイルフィールドに格納されます。
※ “「kintone レコード ID」書類内保存フィールド” の指定が必要です。

【動作ステップの指定】

指定されたステップの動作が完了する際に kintone へのデータ送信を行うか否か指定します。承認の場合は次のステップに移動、もしくは承認完了のタイミングで動作するため、AND 承認の場合はステップの規定人数が承認をしたのちに行われま

す。

- 申請 (通常フォームの場合は「提出」)
- 承認 1 ~ (ステップ名称の指定がある場合は指定名称で表示、通常フォームの場合は表示無し)
- 完了書類修正 (通常フォームの場合は「修正」)

なお、管理者スキップを行った場合、及び空ステップを指定した場合もステップを完了するタイミングで連携が動作します。
下書き、保留は連携動作しません。

【フィールド指定】

「連携基本情報」の入力後、【確認】ボタンをクリックします。kintone サーバーとの接続が確認されるとボタンがオレンジ色に変わりクリックできるようになり、書類と kintone アプリのフィールドを対応付けする画面が利用可能になります。

| + kintone連携先アプリを追加する | | 備考 | 操作 |
|--|---|----|---|
| 連携先kintone情報 | レコード追加・更新タイミング | | |
| ドメイン devettsn.cybozu.com | 「kintoneレコードID」書類内保存フィールド --- 保存しない --- | | <div style="border: 1px solid gray; padding: 5px;"> <div style="border-bottom: 1px solid gray; padding-bottom: 5px;">登録</div> <div style="padding-bottom: 5px;">更新</div> </div> |
| アプリID 4 | <input type="checkbox"/> 添付ファイルの更新 --- 添付フィールドコード --- | | |
| APIトークン 9fr9VE6umQIoKph4llgClDafsqTSXcvSfNc | <input type="checkbox"/> コメントの追加 <input type="checkbox"/> 書類削除と連動 | | |
| ゲストID 0 (未指定の場合は0) | <input type="checkbox"/> 完了書類のPDFを添付 --- 格納フィールドコード --- | | |
| | <input checked="" type="checkbox"/> 申請 <input type="checkbox"/> 承認1 <input type="checkbox"/> 確認2 <input type="checkbox"/> 法務受付3 <input type="checkbox"/> 法務確認4 <input type="checkbox"/> 法務承認5 <input type="checkbox"/> 結果確認6 <input type="checkbox"/> 部門確認7 <input type="checkbox"/> 完了書類修正 | | |
| | <div style="border: 2px solid red; padding: 5px; display: inline-block;">確認</div> → <div style="border: 2px solid red; padding: 5px; display: inline-block;">フィールド指定</div> | | |
| | | | |

指定に誤りがない場合、
クリック可能になる

| 連携先kintone情報 | レコード追加・更新タイミング | 備考 | 操作 |
|--|---|--|----|
| ドメイン devettfsn.cybozu.com | 「kintoneレコードID」書類内保存フィールド --- 保存しない --- | | |
| アプリID 22 | <input type="checkbox"/> 添付ファイルの更新 --- 添付フィールドコード --- | | |
| APIトークン 9fr9VE6umQloKph4llgClDafsqTSXcvSfNoB5zrQ | <input type="checkbox"/> コメントの追加 | | |
| ゲストID 0 (未指定の場合は0) | <input type="checkbox"/> 書類削除と連動 | | |
| | <input type="checkbox"/> 完了書類のPDFを添付 --- 格納フィールドコード --- | | |
| | <input checked="" type="checkbox"/> 申請 <input type="checkbox"/> 承認1 <input type="checkbox"/> 確認2 <input type="checkbox"/> 法務受付3 <input type="checkbox"/> 法務確認4 <input type="checkbox"/> 法務承認5 <input type="checkbox"/> 結果確認6 <input type="checkbox"/> 部門確認7 <input type="checkbox"/> 完了書類修正 | 登録 管理者 (2020/10/26 09:54:09) 更新 管理者 (2020/10/26 09:57:43) | |
| | フィールド指定 | | |
| <div style="border: 2px solid red; padding: 5px;"> code: GAIA_AP15 (id: jMJMmIlt4gR0bkrygXkRQ) APIトークンとアプリ (id: 22) の組み合わせが正しくありません。 </div> | <div style="border: 2px solid red; padding: 5px; color: red; text-align: center;"> 指定に誤りがある場合、 クリック不可のまま </div> | | |

- ※ 指定誤りがある場合は【確認】ボタンの下に kintone からの応答メッセージが表示されます。表示内容は kintone からの応答をそのまま表示しているため、kintone から戻されるメッセージが非常に長い場合などに画面が崩れる可能性があります。

フィールド指定画面では X-point の書類項目（フィールド ID）を kintone のどの項目（フィールドコード）に対して転送するかの指定を行います。指定では連携時に更新を行うフィールドコードのみ選択します。更新先としてフィールドコードを指定しない場合はフィールドの設定値は更新されません。

X-point フォームの項目

対応する kintone アプリの項目

| X-point (連携元) 契約書社内審査 (C001) | kintone (連携先) AppID:4, devettfsn.cybozu.com | |
|-------------------------------------|--|--------|
| 審査主旨 (txtMemo) [page-1] | 審査主旨 (txtMemo) | 更新される |
| 申請日 (txtShinseibi) [page-1] | 申請日 (txtShinseibi) | |
| 申請者 (txtShinseisya) [page-1] | 申請者 (txtShinseisya) | |
| 審査No (txtNo) [page-1] | --- 連携フィールド --- | 更新されない |
| 顧客コード (txtKokyakucode) [page-1] | 管理コード (txtKanricode) | |

- ※ 連携はフィールドの対応で指定します。フィールドを連結、内容を分割しての対応は指定できません。フィールド内容の加工が必要な場合は JavaScript 等をフォームに組み込み、隠しフィールド等に加工済みのデータが用意されるようにしてください。
- ※ 日時フィールド等、X-point (YYYY/MM/DD HH:MM:SS) と kintone (YYYY-MM-DDTHH:MM-SSZ) で格納書式が異なる一部のフィールドは内容の自動変換が行われます。フィールドの対応がどのように行われるかは、「[フィールドの対応](#)」項を参照してください。

なお、登録は複数の kintone アプリに対して行う事が可能です。同一 kintone アプリへの連携を複数指定する事も可能です。同一の kintone アプリの登録はステップ毎に異なるフィールドを更新するようなケースで利用します。受付ステップで「受付番号」を kintone アプリのレコードに更新、完了時に「審査結果」をレコードに更新というような場合が考えられます。

| + kintone連携先アプリを追加する | | レコード追加・更新タイミング | 備考 | 操作 |
|--|--|--|--|--------------------|
| 連携先kintone情報 ドメイン devettfsn.cybozu.com アプリID 4 APIトークン 9fr9VE6umQloKph4llgClDafsqTSXcvSfNc ゲストID 0 <small>(未指定の場合は0)</small> | レコード追加・更新タイミング 「kintoneレコードID」書類内保存フィールド --- 保存しない --- <input type="checkbox"/> 添付ファイルの更新 --- 添付フィールドコード --- <input type="checkbox"/> コメントの追加 <input type="checkbox"/> 書類削除と連動 <input type="checkbox"/> 完了書類のPDFを添付 --- 格納フィールドコード --- <input checked="" type="checkbox"/> 申請 <input type="checkbox"/> 承認1 <input type="checkbox"/> 確認2 <input type="checkbox"/> 法務受付3 <input type="checkbox"/> 法務確認4 <input type="checkbox"/> 法務承認5 <input type="checkbox"/> 結果確認6 <input type="checkbox"/> 部門確認7 <input type="checkbox"/> 完了書類修正 | | 登録 管理者 (2020/10/26 09:54:09) 更新 管理者 (2020/10/26 09:57:43) | 会 |
| <input type="button" value="確認"/> | | <input type="button" value="フィールド指定"/> | | アプリ①への連携を指定 |
| 連携先kintone情報 ドメイン devettfsn.cybozu.com アプリID 12 APIトークン NcgfF8dsV8jf94lfGskHA8pvSf1q6BZc3M ゲストID 0 <small>(未指定の場合は0)</small> | レコード追加・更新タイミング 「kintoneレコードID」書類内保存フィールド レコードID (kintone_id) [page-1] <input checked="" type="checkbox"/> 添付ファイルの更新 --- 添付フィールドコード --- <input type="checkbox"/> コメントの追加 <input type="checkbox"/> 書類削除と連動 <input type="checkbox"/> 完了書類のPDFを添付 -- 格納フィールドコード -- <input type="checkbox"/> 申請 <input type="checkbox"/> 承認1 <input type="checkbox"/> 確認2 <input type="checkbox"/> 法務受付3 <input checked="" type="checkbox"/> 法務確認4 <input type="checkbox"/> 法務承認5 <input type="checkbox"/> 結果確認6 <input type="checkbox"/> 部門確認7 <input type="checkbox"/> 完了書類修正 | | 登録 更新 | 会 |
| <input type="button" value="確認"/> | | <input type="button" value="フィールド指定"/> | | アプリ②への連携を指定 |

- ※ 複数アプリを登録する場合、異なる kintone サーバーを指定しても動作します。
- ※ kintone への連携はワークフロー処理と同期して行われます。その為、同一ステップで多数の連携を実行すると承認などの動作が連携先の数に応じ徐々に遅くなります。

！ 注意事項

X-point の連携設定では「フォームコード」をキーにして設定情報を取得しています。そのため、異なるフォームに対し連携設定をしている場合でも、同一のフォームコードを使用している場合は、すべての設定が動作対象となります。意図しない複数連携が発生する可能性があるため、フォームコードの一意性を保つようご注意ください。なお、検索用途などで版違いのフォームに対して同一のフォームコードを設定して運用している場合は、過去版の kintone 連携設定を削除することで、1 対 1 の連携状態を維持することが可能です。

2.3.2. X-point の書類ナビボタンから kintone アプリにレコードを追加・更新

X-point の書類閲覧時にナビボタンを操作するタイミングで kintone にレコードの登録を行う事ができます。

kintone にレコード登録を行う際に返されるレコード ID を X-point の書類内で保存するように構成すると、保存したレコード ID を利用することで kintone に登録されたデータの更新を行う事が可能になります。

■ 利用の前提条件

kintone アプリの「アプリの設定」にて「レコード追加」権限を持つ API トークンを発行する必要があります。

■ 表定義の利用

kintone アプリに追加するレコード情報に含める表定義の数、行数に制限はありません。

■ その他留意事項

- X-point から kintone へのレコード登録で添付ファイルを送信する場合、X-point 上で書類 ID が採番された後に行う必要があります。具体的にはフォーム作成時に【申請】ナビで提出をすることが出来ません。
一度下書き状態にしたのち、編集・提出を行う、もしくは承認ステップで行う必要があります。
- X-point から kintone へのレコード登録は、ワークフローフォーム、通常フォームの両方で行う事が可能です。
- X-point の書類内に kintone のレコード ID を保存する場合、書類を編集状態にしてから連携 API を実行する必要があります。これは、レコード登録時に返すレコード ID を書類に反映する処理をフォーム内の JavaScript で実行するため、書類が編集状態になっている場合でなければデータベースに値を登録できないからです。
- kintone のレコードの更新で添付ファイルの送信を行う場合、kintone レコードの指定添付ファイルフィールドに設定されているファイル全てが X-point から新たに送る情報の内容で全て置き換えられます。
レコード更新で送信する情報に添付ファイルに係わる情報が含まれない場合は、添付ファイルフィールドに設定されている添付ファイルの内容は維持されます。
- kintone へのレコード登録・更新でエラーが発生した際に、X-point の申請、承認操作を停止する場合、ナビボタンに組み込む JavaScript 処理で連携 API のエラー判定を行う必要があります。エラー判定処理を行わない場合、kintone への登録でエラーが発生した場合における X-point の申請、承認操作は行われます。
- 書類ナビボタンより kintone 連携を行う場合、書類内のワークフロー取得値を反映する事ができません。
ナビボタンから実行される JavaScript がワークフロー取得値の書類に反映されるタイミングより早く行われるためです。

2.3.3. クエリ CSV 出力から kintone アプリにレコードを追加

kintone に CSV ファイルでレコードを登録する場合の CSV ファイルはクエリ CSV 出力を利用して作成します。

連携を行う場合は出力する CSV ファイルの行頭に開始行マーク「*」が付くよう指定します。クエリ CSV 出力で作成を行うため、タスクコマンドによる「クエリ自動出力」、リモートコマンドによる「クエリリモート出力」機能での利用も可能です。

kintone ではコマンドラインツールを利用して CSV ファイルを登録します。

【コマンドラインツール実行の例】

- ① アプリ ID=12 からユーザーID「administrator」、パスワード「atled@kintone」で CSV ファイルを登録

```
cli-kintone -d ksubdomain.cybozu.com -a 12 -u administrator -p atled@kintone -f csv -e sjis
```

- ② アプリ ID=12 からアプリのトークンコードを指定して、レコード “\$id”, “name”, “date” の 3 列を取得

```
cli-kintone -d ksubdomain.cybozu.com -a 12 -t diVUoVQDgnnU7r01v4sFyfUiGnX0ResLGg0fIk1s  
-c "$id,name,date" -f csv -e sjis
```

- ③ アプリ ID=12 からアプリのトークンコードを指定して、レコード “\$id”, “name” の 2 列に加えテーブル “group1” を取得。

```
cli-kintone -d ksubdomain.cybozu.com -a 12 -t diVUoVQDgnnU7r01v4sFyfUiGnX0ResLGg0fIk1s  
-c "$id,name,date,group1" -f csv -e sjis
```

※ 取得する CSV ファイルはテーブルの行数分になり、CSV の先頭には 1 行目を示す “*” が追加されます。

2. 4. X-point 上で kintone アプリをマスタに利用

kintone 連携サービスが利用可能な X-point では、kintone アプリを直接マスタとして利用、アプリのフォーム定義を固有マスタ作成で利用、アプリに登録されているレコードを固有マスタヘデータインポートができます。

2. 4. 1. kintone アプリ・マスタとして利用

X-point で利用可能なマスタは“簡易マスタ”、“固有マスタ”がありますが、kintone 連携サービスを導入することで kintone アプリを“kintone アプリ・マスタ”として利用することが出来ます。

登録はマスタ管理の「kintone 連携マスタ作成」にて行います。登録後は、簡易マスタ、固有マスタと同様に扱う事が可能になり、「マスタ参照ウィンドウ・コンポーネント」を kintone アプリを直接参照する「マスタ参照ウィンドウ」をフォームに配置できます。

マスタ管理 → マスタ一覧 → 簡易マスタ作成 → **kintoneマスタ作成** → ユーザ固有マスタ → タスクログ

kintone連携マスタ作成

基本情報

| | |
|------------------------|--|
| マスタコード ① 半角英数30文字以内 | |
| マスタ名称 | |
| 備考 | |

kintoneアプリ情報

| | |
|-----------|--|
| ドメイン名 | ① ドメイン名 (必須) cybozu.comは省略可能です |
| アプリID | 0 |
| APIトークン | ① アクセス権「レコード閲覧」、「アプリ管理」が有効なAPIトークンを指定します |
| ゲストスペースID | 0 |

アプリへの接続情報を指定

【利用上の注意】

- ※ kintone アプリ・マスタの利用には、kintone アプリの API トークンが必要です。
使用する API トークンはアクセス権に「レコード閲覧」が許可されている必要があります。
- ※ kintone アプリ・マスタの利用時は、常に X-point と kintone の間で通信が発生します。X-point サーバーと kintone サーバーの間で通信障害などが発生した場合、kintone サーバーがメンテナンスなどで停止している場合は、kintone アプリを参照するマスタを利用した機能が動作しなくなります。
- ※ kintone アプリ・マスタを「マスタ参照コンポーネント」で利用する場合は、“絞込条件”の動作が「簡易マスタ」、「固有マスタ」を利用する際と異なります。これは検索動作が kintone 側で行われるため検索仕様に違いが生じるためです。動作の差異が顕著に発生するのは絞込を文字列の部分一致で行う場合で、kintone は部分一致検索を行う場合もワード単位での一致で動作するためになります。詳細は、kintone developer site 等の情報をご確認ください。
なお、検索動作の差異を回避する場合は、kintone アプリのデータを X-point の固有マスタに取り込み「マスタ参照コンポーネント」を作成するようにします。

2.4.2. 固有マスタ定義の項目に利用

X-point の固有マスタ作成を行う際に、kintone アプリのフォーム定義情報を取得し、固有マスタの初期フィールド項目として利用することが出来ます。通常の固有マスタ「作成」と基本的に同じ操作になり、フィールド一覧の初期取得のみが異なります。

🏠 マスタ管理 → マスター一覧 → 簡易マスタ作成 → kintoneマスタ作成 → ユーザ固有マスタ → タスクログ

ユーザ固有マスタ作成 + 作成 🗑️ 取込 + kintoneから作成 🔄 インポート/バッチ登録

基本情報

テーブル名 kintoneAppMaster

項目を取得するkintoneアプリ情報を指定します

| | |
|-----------|--|
| ドメイン名 | devettfsn.cybozu.com |
| アプリID | 4 |
| APIトークン | 9fr9VE6umQloKph4llgClDafsqTSXcvSfNoB5zrQ |
| ゲストスペースID | 0 (未指定の場合は0) |

取得

フィールドを追加 【取得】でフィールド情報を kintone から得る

| NO | フィールド名 (フィールドID) | データ型 | 桁数 | 主キー | 索引 | 項目削除 |
|----|-------------------|------|----|--------------------------|--------------------------|------|
| 1 | bxtKokyakucode | 文字列型 | 64 | <input type="checkbox"/> | <input type="checkbox"/> | |
| 2 | bxtKanricode | 文字列型 | 64 | <input type="checkbox"/> | <input type="checkbox"/> | 🗑️ |
| 3 | bxtInspectiondate | 文字列型 | 64 | <input type="checkbox"/> | <input type="checkbox"/> | 🗑️ |
| 4 | bxtKokyakumei | 文字列型 | 64 | <input type="checkbox"/> | <input type="checkbox"/> | 🗑️ |
| 5 | bxtTantou | 文字列型 | 64 | <input type="checkbox"/> | <input type="checkbox"/> | 🗑️ |

- ※ データ型、桁数、主キー、索引の項目は設定されません、設定はフィールド名に対してのみ行われます。
- ※ kintone からの情報を取得後、【作成】ボタンをクリックしない限り TABLE は作成されません。
- ※ 指定する kintone アプリの情報が保存されることはありません。

2.4.3. インポートバッチ登録

kintone 連携サービスを導入している環境では、固有マスタの「インポートバッチ登録」を行う事ができます。

通常はインポート対象となる CSV ファイルを指定しますが、kintone 連携サービス導入環境では、データファイル以外に kintone アプリの接続情報を指定できます。

■ 「マスター一覧」で固有マスタにインポートするウィザード（先頭画面）

「kintone 連携サービス」の利用が可能な X-point では赤枠部分が追加されます。

🏠 マスタ管理 → マスター一覧 → 簡易マスタ作成 → kintoneマスタ作成 → ユーザ固有マスタ → タスクログ

ユーザ固有マスタインポート

+ 作成 🗑️ 取込 + kintoneから作成 🔄 インポートバッチ登録

| 基本情報 | |
|----------------------------|---|
| 対象マスタ ※ | kintonemaster |
| インポート定義 | -- インポート定義 -- ※インポート定義を新規作成する場合は指定しません。 |
| 書式 ※ | <input checked="" type="radio"/> カンマ区切 <input type="radio"/> タブ区切 <input checked="" type="checkbox"/> 先頭行をフィールド名とする |
| インポート内容 ※ | <input checked="" type="radio"/> データを完全に置き換える <input type="radio"/> データを追加する |
| インポートできないレコードが検出された場合の動作 ※ | <input type="radio"/> インポートをキャンセル <input checked="" type="radio"/> インポートを継続(インポートできないレコードを除外する) <input type="radio"/> データファイル |
| | 参照 ドラッグ&ドロップするか参照ボタンから選択してください |
| データ ※ | <input checked="" type="radio"/> kintoneアプリ kintone アプリへの接続情報 |
| ドメイン名 ※ | |
| アプリID ※ | 0 |
| APIトークン ※ | |
| ゲストスペースID | 0 (未指定の場合は0) |

※ kintone アプリから取得するデータの形式は「カンマ区切」のみになります。

※ データの先頭行は必ずフィールドコード名が設定されます。

【取得データの例】

"kokyaku_code", "kokyaku_name", "torihiki_busyo", "torihiki_tantou", "docid"

"G001", "株式会社太陽商事", "プロダクト開発", "管理者", "2632"

"G002", "富士山エレクトロニクス株式会社", "勝 太郎", "", "2638"

"G003", "港北都市開発株式会社", "", "", "2640"

"G004", "株式会社田園工業", "", "", "2641"

■ 「ユーザ固有マスタインポート バッチ登録」画面

使用するインポート定義に kintone アプリを指定したものを選択すると、kintone アプリの情報が赤枠部分に表示され、インポートバッチを実行することが出来ます。

🏠 マスタ管理

→ マスター一覧 → 簡易マスタ作成 → kintoneマスタ作成 → ユーザ固有マスタ → タスクログ

ユーザ固有マスタインポート バッチ登録

+ 作成 🗑️ 取込 + kintoneから作成 🔄 インポートバッチ登録

ユーザ固有マスタの自動取り込みの設定を行います。

| 基本情報 | | | | | | | |
|------------------|--|-------|----------------------|-------|---|---------|--|
| インポートのバッチ実行 | <input type="radio"/> 実行する <input checked="" type="radio"/> 実行しない | | | | | | |
| マスタ名 | kintonemaster | | | | | | |
| 使用するインポート定義 | kintonemaster | | | | | | |
| インポート元kintoneアプリ | <table border="1"> <tr> <td>ドメイン名</td> <td>devetffsn.cybozu.com</td> </tr> <tr> <td>アプリID</td> <td>4</td> </tr> <tr> <td>APIトークン</td> <td>9fr9VE6umQIoKph4llgClDafsqTSXcvSfNoB5zrQ</td> </tr> </table> | ドメイン名 | devetffsn.cybozu.com | アプリID | 4 | APIトークン | 9fr9VE6umQIoKph4llgClDafsqTSXcvSfNoB5zrQ |
| ドメイン名 | devetffsn.cybozu.com | | | | | | |
| アプリID | 4 | | | | | | |
| APIトークン | 9fr9VE6umQIoKph4llgClDafsqTSXcvSfNoB5zrQ | | | | | | |
| 備考 | | | | | | | |

↑ kintone アプリへの接続情報

保存

🔔 ユーザ固有マスタインポートバッチ実行について

ユーザ固有マスタインポートバッチを実行するにあたり、別途スケジュール設定を行う必要があります。設定方法については [ユーザ固有マスタインポートバッチ実行マニュアル](#) をご参照下さい。

インポート定義情報

| | |
|----------------------------|-------------------------------|
| 対象マスタ ※ | kintonemaster |
| インポート定義 ※ | kintonemaster |
| 区切り文字 ※ | CSV区切 |
| 先頭行 ※ | 先頭行をフィールド名とする |
| インポート内容 ※ | データを完全に置き換える |
| インポートできないレコードが検出された場合の動作 ※ | インポートを継続 (インポートできないレコードを除外する) |

取込対象-置換定義

| 項目順 | DB列名 | 置換定義 | 初期値 | ゼロパディング | データ不正時の扱い |
|-----|----------------|------|-----|---------|------------------|
| 1 | txtkogyakucode | | | | 項目値を空白としてインポートする |

第3章. 事前準備

連携を行うためには kintone、X-point 双方で連携 API が利用できるよう準備を行う必要があります。

本書では kintone 上で「契約書・社内審査」の基本登録を行ったのち X-point に書類を提出し社内審査のワークフロー処理を行った結果を、X-point から kintone に戻す場合にに基づき説明を行います。

【契約書・社内審査】

～ kintone ～

顧客コード 顧客名

主担当

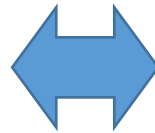
審査完了希望日 管理コード

審査主旨

契約書番号 契約書名

審査日 審査No 添付ファイル

詳細 (最大1 GB)



～ X-point ～

| 契約書 社内審査 | | 申請日: | 2017年02月01日 | | | | |
|----------|------|-------|-------------|------|------|----|------|
| | | 申請者: | 勝 太郎 | | | | |
| | | 審査No: | | | | | |
| 担当 | 上長承認 | 受付 | 添付 | 内容確認 | 上長承認 | 担当 | 上長承認 |
| | | | | | | | |
| 顧客名 | | | | | | | |
| 主担当 | 勝 太郎 | | | | | | |
| 審査完了希望日 | | | | | | | |
| 管理コード | | | | | | | |
| 審査主旨 | | | | | | | |
| 契約書番号 | 契約書名 | | | | | | |
| | | | | | | | |
| | | | | | | | |
| | | | | | | | |

3. 1. kintone 側の準備

3. 1. 1. 連携アプリの作成

X-point の「契約書・社内審査」フォームと連携するアプリを作成します。本項では X-point フォームのフィールド ID とアプリのフィールドコードをあわせていますが、異なるレイアウト、項目名称が異なる場合であっても連携は可能です。但し、X-point が対応しないフィールドタイプ、複数表の送信はできません。

| 社内連絡表の項目名 | タイプ | フィールド名 | フィールドコード | 備考 |
|-----------|----------|---------|-------------------|--------------------|
| 顧客コード | 文字列（1行） | 顧客コード | txtKokyakucode | |
| 顧客名 | 文字列（1行） | 顧客名 | txtKokyakumei | |
| 主担当 | ユーザー選択 | 主担当 | txtTantou | |
| 審査完了希望日 | 日付 | 審査完了希望日 | txtKiboubi | |
| 管理コード | 文字列（1行） | 管理コード | txtKanricode | |
| 審査主旨 | 文字列（複数行） | 審査主旨 | txtMemo | |
| 添付ファイル | 添付ファイル | 添付ファイル | attached_file | X-point ではナビボタンに格納 |
| 審査日 | 日付 | 審査日 | txtInspectiondate | X-point からの戻り値を格納 |
| 審査 No | 文字列（1行） | 審査 No | txtInspectionno | X-point からの戻り値を格納 |
| 書類 ID | 数値 | 書類 ID | docid | X-point の書類 ID を格納 |

| 社内連絡表の項目名 | タイプ | フィールド名 | フィールドコード | 備考 |
|-----------|-----------|----------|------------------|------------------|
| (テーブル部) | (テーブルの設定) | — | group1 | |
| 契約書番号 | 文字列 (1行) | 契約書番号 | txtKeiyakusyono | テーブル内項目 |
| 契約書名 | 文字列 (1行) | 契約書名 | txtKeiyakusyomei | テーブル内項目 |
| (項目なし) | 添付ファイル | 押印済み PDF | pdf_document | 承認完了書類の PDF ファイル |

- ※ kintone では「添付ファイル」フィールドを複数配置でき、かつ夫々のフィールドに複数のファイルを格納できますが、X-point に転送すると全て【添付】ナビボタンに纏めて格納されます。
- ※ kintone の添付ファイルフィールドに同一名のファイルが格納されていた場合、X-point には個々に転送されます。
- ※ X-point から添付ファイルを受け取る場合、添付ファイルは kintone アプリの特定添付フィールドに全て追加されます。
- ※ kintone に格納する書類 ID フィールドはレコード入力・編集の際に値が変更されないよう、別途 JavaScript の組み込みを行い、書類 ID の格納フィールドを編集禁止（無効）、もしくは不可視の状態に指定します。

3.1.2. API トークンの作成

kintone 内で登録レコード操作、X-point から連携を行うアプリのレコード操作できるよう、API トークンを発行します。

■連携を行うアプリを表示し「アプリの設定を変更」メニューを実行します。

The screenshot shows the kintone interface for the '契約書・社内審査' (Contract/In-house Review) app. The top navigation bar includes 'cybozu.com', 'Administrator', and a search bar. Below the navigation bar, there are filters and a table of records. A context menu is open over the table, with a red box highlighting the 'アプリの設定を変更' (Change app settings) option. Other options in the menu include '表示件数' (Number of items to display), '先頭行を固定表示' (Lock first row), and a right arrow.

■「設定」タブで「API トークン」をクリックします。

The screenshot shows the '契約書・社内審査' app settings page. The top navigation bar includes 'cybozu.com', 'Administrator', and a search bar. Below the navigation bar, there are buttons for '変更を中止' (Cancel changes) and 'アプリを更新' (Update app). The main content area is divided into four tabs: 'フォーム' (Form), '一覧' (List), 'グラフ' (Graph), and '設定' (Settings). The '設定' tab is selected, and the 'APIトークン' (API Token) option is highlighted with a red box. Other options in the '設定' tab include '一般設定' (General settings), 'カスタマイズ' (Customize), 'その他の設定' (Other settings), 'アイコンと説明' (Icon and description), 'プラグイン' (Plugins), 'カテゴリ' (Categories), 'プロセス管理' (Process management), 'JavaScript / CSSでカスタマイズ' (Customize with JavaScript / CSS), '言語ごとの名称' (Names by language), '通知' (Notifications), 'レコードのタイトル' (Record title), 'アプリの条件通知' (App condition notifications), '高度な設定' (Advanced settings), 'アクセス権' (Access rights), and 'アクション' (Actions).

■ 「生成する」をクリックします。

The screenshot shows the 'APIトークン' (API Token) page in the cybozu.com administrator interface. The page title is 'APIトークン'. Below the title, there is a text area containing instructions: 'REST APIでこのアプリを操作するためのトークンを生成できます。' and an example command: '例: curl -H "X-Cybozu-API-Token: YOUR_TOKEN" "https://[redacted]cybozu.com/k/v1/record.json?app=22&id=1"'. Below the text, there is a '生成する' (Generate) button highlighted with a red box. Below the button, there is a table with two columns: 'APIトークン' and 'アクセス権'. The table is currently empty, with the text 'APIトークンはありません。' (No API tokens.) displayed below it.

■ 生成されたトークンにアクセス権を設定します。

The screenshot shows the 'APIトークン' page after a token has been generated. The '生成する' button is now disabled. The table below has one row with the following data:

| APIトークン | アクセス権 |
|---|---|
| US8Yz28RN8ZfAslKCqY8viMc4EyaJmX4wWepSmOMA | <input checked="" type="checkbox"/> レコード閲覧 <input checked="" type="checkbox"/> レコード追加 <input checked="" type="checkbox"/> レコード編集 <input type="checkbox"/> レコード削除 <input type="checkbox"/> アプリ管理 |

表示された API トークン（40 文字）を控え、必要なアクセス権をチェックします。

- ※ kintone から X-point への書類申請で利用する場合「レコード閲覧」が必須です。
- ※ X-point から kintone へのレコード追加・更新で利用する場合「レコード追加」、「レコード編集」が必須です。
- ※ X-point 側から kintone に登録されているレコード削除を行う場合、「レコード削除」が必要です。

■ 「保存」をクリックし、発行したトークンとアクセス権を kintone アプリに登録します。

The screenshot shows the 'APIトークン' page with the '保存' (Save) button highlighted by a red box. The table below contains the same data as in the previous screenshot:

| APIトークン | アクセス権 |
|---|---|
| US8Yz28RN8ZfAslKCqY8viMc4EyaJmX4wWepSmOMA | <input checked="" type="checkbox"/> レコード閲覧 <input checked="" type="checkbox"/> レコード追加 <input checked="" type="checkbox"/> レコード編集 <input type="checkbox"/> レコード削除 <input type="checkbox"/> アプリ管理 |

3.1.3. 連携ライブラリの登録

アプリから kintone 連携ライブラリが利用できるよう、アプリの設定からライブラリを登録します。

本項で登録するライブラリは基本ライブラリのみになります。kintone アプリからレコードの内容を X-point に申請する場合は、申請処理を行う JavaScript を記述し kintone アプリに追加する必要があります。

■連携を行うアプリを表示し「アプリの設定を変更」メニューを実行します。



■「設定」タブで「JavaScript/CSS カスタマイズ」をクリックします。



- 「PC用のJavaScriptファイル」で「URL指定で追加」をクリックします。



- 連携を行う X-point の JavaScript ファイルを指定します。

「https://{X-point サーバーFQDN}/xpoint/js/kintone.js」

指定後は必ず【保存】をクリックします。



※ URL 指定 (https で開始) で連携ライブラリを指定した場合、X-point サーバーと通信できない場合 (停止中など) は、JavaScript の読み込みが行われません。

- 「スマートフォン用の JavaScript ファイル」の「URL 指定で追加」も同様に指定します。

{https://{X-point サーバーFQDN}/xpoint/js/kintone.js}

指定後に必ず【保存】をクリックします。



※ URL 指定 (https で開始) で連携ライブラリを指定した場合、X-point サーバーと通信できない場合 (停止中など) は、JavaScript の読み込みが行われません。

- 「保存」を行い kintone アプリに登録します



- 保存後に表示される「アプリを更新」を実行します。



3.1.4. kintone アプリの書類 ID フィールドを制御する

kintone アプリのレコードに保存する書類 ID フィールドの内容をユーザーが編集してしまうとレコードと書類の対応に不整合が生じるため、編集されてしまう事が無いよう JavaScript を組み込み編集時は項目が表示されないようにします。

■アプリに追加する JavaScript をローカルファイルに作成します。

レコード編集画面が表示された際に行われるイベントにフィールドを不可視にする命令を追加します。

```
(function () {
  "use strict";

  // PC用レコード編集イベント
  kintone.events.on('app.record.edit.show', function (event) {
    kintone.app.record.setFieldShown('docid', false); // "docid"フィールドを不可視にする
  });

  // モバイル用レコード編集イベント
  kintone.events.on('mobile.app.record.detail.show', function (event) {
    kintone.mobile.app.record.setFieldShown('docid', false); // "docid"フィールドを不可視にする
  });

})();
```

※ レコード明細で不可視にする場合は「app.record.detail.show」(PC用)、「mobile.app.record.detail.show」(モバイル用)を使用します。

フィールドを編集禁止にする場合は event オブジェクトを使用します。

```
(function () {
  "use strict";

  // PC用レコード編集イベント
  kintone.events.on('app.record.edit.show', function (event) {
    event['record']['docid']['disabled'] = true; // "docid"フィールドを無効にする
    return event;
  });

  // モバイル用レコード編集イベント
  kintone.events.on('mobile.app.record.detail.show', function (event) {
    event['record']['docid']['disabled'] = true; // "docid"フィールドを無効にする
    return event;
  });

})();
```

■ 「アップロードして追加」

作成した JavaScript ファイルを kintone アプリにアップロードします。

JavaScript / CSSでカスタマイズ

JavaScript / CSSファイルを活用して、アプリをカスタマイズできます。
JavaScriptの開発にあたっては、[JavaScriptコーディングガイドライン](#)をご確認ください。

適用範囲

すべてのユーザーに適用 アプリ管理者だけに適用 適用しない

PC用のJavaScriptファイル

スマートフォン用のJavaScriptファイル

アップロードした JavaScript ファイルの内容でエラーが発生しない場合は、レコード編集の画面で書類 ID フィールドが表示されない、あるいは編集禁止の状態になっています。

3.2. X-point 側の準備

3.2.1. フォームの作成

■基本項目の作成

eFormMaker で「契約書・社内審査」を作成します。

作成するフォームの入力部、印影部について ID 名、タグ名、フィールドタイプ等を示します。

【上段項目】

| 項目 | ID 名 | 推奨タグ名 | タイプ | その他 |
|-----------|---------------|-----------|---------|---------|
| 申請日 | txtShinseibi | 申請日 | 文字フィールド | |
| 申請者 | txtShinseisya | 申請者 | 文字フィールド | |
| 審査 No | txtNo | 審査 No | 文字フィールド | 自動採番を設定 |
| 印影・担当 | stpTantou01 | 印影・担当 | 印影 | 印影「申請」 |
| 印影・上長承認 | stpJyoucho01 | 印影・上長承認 | 印影 | 印影「承認」 |
| 印影・法務受付 | stpUketuke | 印影・法務受付 | 印影 | 印影「承認」 |
| 印影・法務内容確認 | stpNaiyou | 印影・法務内容確認 | 印影 | 印影「承認」 |
| 印影・法務上長承認 | stpJyoucho02 | 印影・法務上長承認 | 印影 | 印影「承認」 |
| 印影・担当確認 | stpTantou02 | 印影・担当確認 | 印影 | 印影「確認」 |
| 印影・上長確認 | stpJyoucho03 | 印影・上長確認 | 印影 | 印影「承認」 |

※ 「審査 No」に指定する自動採番コンポーネント。X-point 上で値が決定される項目です。

(コンポーネント指定例)

プレフィックス : 「KS- {YYYY.}」、 番号桁数 : 5、 ゼロパディング : する

※ 印影に指定する印影コンポーネント、承認ステップは作成するワークフローに合わせてください。

本例は、図左側より、申請 00、承認 01、承認 02、承認 03、承認 04、承認 05、承認 06 を承認ステップに指定する全 7 ステップで構成されます。

【中段項目】

| | | |
|---------|-------|-----|
| 顧客名 | 顧客コード | 顧客名 |
| 主担当 | 担当名 | |
| 審査完了希望日 | 完了希望日 | |
| 管理コード | 管理コード | |
| 審査主旨 | メモ | |

| 項目 | ID名 | 推奨タグ名 | タイプ | その他 |
|-------|----------------|-------|---------|-----|
| 顧客コード | txtKokyakucode | 顧客コード | 文字フィールド | |
| 顧客名 | txtKokyakumei | 顧客名 | 文字フィールド | |
| 担当名 | txtTantou | 担当名 | 文字フィールド | |
| 完了希望日 | txtKiboubi | 完了希望日 | 文字フィールド | |
| 管理コード | txtKanricode | 管理コード | 文字フィールド | |
| 審査主旨 | txtMemo | 審査主旨 | テキストエリア | |

【下段項目】

| | |
|-------|------|
| 契約書番号 | 契約書名 |
| 契約書番号 | 契約書名 |
| | |
| | |

| 項目 | ID名 | 推奨タグ名 | タイプ | その他 |
|-----------|------------------|-------|---------|------|
| (表形式グループ) | group1 | — | 表形式グループ | 行数：3 |
| 契約書番号 | txtKeiyakusyono | 契約書番号 | 文字フィールド | |
| 契約書名 | txtKeiyakusyomei | 契約書名 | 文字フィールド | |

【フォーム設定で基本情報を指定】

フォーム名称 : 契約書社内審査
 フォームカナ名 : ケイヤクショシャナイシンサ
 コード : KeiyakuShinsa
 表名 : 指定しません (空欄にします)

『契約書・社内審査』の全体イメージ

| | | | |
|----------|--|--------|------|
| 契約書 社内審査 | | 申請日 : | 申請日 |
| | | 申請者 : | 申請者 |
| | | 審査No : | 審査No |

| | | | | | | |
|----------------------|----------------------|----------------------|----------------------|----------------------|----------------------|----------------------|
| 担当 | 上長承認 | 法務 | | | 担当 | 上長承認 |
| | | 受付 | 内容確認 | 上長承認 | | |
| 名前 00-00-00 申請 | 名前 00-00-00 承認 | 名前 00-00-00 承認 | 名前 00-00-00 承認 | 名前 00-00-00 承認 | 名前 00-00-00 確認 | 名前 00-00-00 承認 |

| | | |
|---------|-------|-----|
| 顧客名 | 顧客コード | 顧客名 |
| 主担当 | 担当名 | |
| 審査完了希望日 | 完了希望日 | |
| 管理コード | 管理コード | |
| 審査主旨 | メモ | |

| | |
|-------|------|
| 契約書番号 | 契約書名 |
| 契約書番号 | 契約書名 |
| | |
| | |

■kintone 連携用項目の追加

kintone から申請（提出）された際に kintone の「レコード ID」（ID : kintone_id）、「ユーザーコード」（ID : usercode）を保存する項目を追加します。X-point 上で書類を申請（提出）する際に kintone の「レコード ID」を保存していない場合、X-point から kintone に値を戻す事が出来なくなります。必ず項目の追加を行って下さい。

The image shows a kintone form interface. At the top, there is a table with two columns: '契約書番号' (Contract Number) and '契約書名' (Contract Name). Below this table, there are three input fields: 'レコードID' (Record ID), 'ユーザーコード' (User Code), and '書類ID' (Document ID). These three fields are enclosed in a red rectangular box, indicating they are the required fields to be added for integration.

【追加項目】

| 項目 | ID 名 | 推奨タグ名 | タイプ | その他 |
|---------|--------------|---------|---------|------------------------|
| レコード ID | kintone_id | レコード ID | 整数フィールド | |
| ユーザーコード | usercode | ユーザーコード | 文字フィールド | |
| 書類 ID | xpoint_docid | 書類 ID | 整数フィールド | ワークフロー取得値で「書類 No」を設定する |

- ※ 「レコード ID」、「ユーザーコード」、「書類 ID」を格納するフィールドはプロパティ項目「不可視」を true にします。但し、連携動作の確認などを行う段階では false（可視状態）にしたほうが好都合である場合もあります。運用・作業の状況に合わせて可視、不可視の指定を行って下さい。
- 値の格納は kintone から連携データを受け取ったとき、あるいは X-point が kintone アプリにレコード追加を行った際に行います。
- ※ kintone のレコード ID は 1 以上の数値になります。整数フィールドが空欄の場合、X-point の JavaScript API では整数フィールドの場合は 0 が返されるため、0 であるか否かで設定状態が確認できます。
- ※ 「ユーザーコード」は kintone アプリから連携情報を受け取る際に、受信データの 1 列目に格納されているユーザーコードを X-point の書類に保持するために準備します。値の格納は kintone から連携データを受け取った際に行います。
- ※ 「書類 ID」は kintone アプリのレコードに X-point の書類 ID を含めるために用意します。kintone アプリ側に書類 ID を保存する事で、kintone アプリ側から X-point の書類内容を更新する必要が生じた場合に、更新対象となる書類の特定が容易になります。

3.2.2. kintone 連携の登録

「契約書社内審査」フォームのアップロード、承認ルートが完了したのち、kintone アプリから連携を行う X-point で書類申請（自動申請の利用）、書類表示が可能となるよう kintone アプリ側のカスタマイズで利用する X-point の API トークンを発行します。

■連携を行うフォームの【⇒他操作】をクリックし、展開されたメニュー内の【kintone 連携】をクリックします。

| 名称 | 枚数 | FCD | テーブル名 | TSFファイル | アクション |
|-----------------|----|------|----------|----------|--|
| 新しいフォームグループ (3) | | | | | |
| A367出張報告書 | 1枚 | A367 | xp788929 | A367.tsf | ⇒ 他操作 |
| A368出張申請書 | 1枚 | A368 | xp12660 | A368.tsf | ⇒ 他操作 |
| 契約書社内審査 | 1枚 | C001 | xp896690 | C001.tsf | ⇒ 他操作 フォームグループ追加 アクセス権限 フォーム編集権限 書類制御 状況確認者 kintone連携 Webhook設定 CSVテンプレート作成 ゲスト |
| 所属なしフォーム (0) | | | | | |

■「X-point 認証トークンを生成する」をクリックします。

フォーム管理 → フォーム → 共通承認ルート → CSVテンプレート → コンポーネント → 提出期限 → 督促メール → 自動申請

kintone連携

kintoneからX-pointの書類情報へのアクセスを制御するAPIトークンを作成します。

| 基本情報 | |
|----------------------|------------------|
| フォーム名称 | 契約書社内審査 (C001) |
| フォーム種別 | ワークフロー |
| + X-point認証トークンを生成する | |

「契約書社内審査」用のAPIトークンは登録されていません。

X-pointからkintoneアプリへのレコード追加・更新タイミングを指定します。

+ kintone連携先アプリを追加する

「契約書社内審査」と連携するkintoneアプリは設定されていません。

■「アクセス権」を指定します。

生成された API トークンにアクセス権、連携元の kintone に関する情報を設定します。

kintoneからX-pointの書類情報へのアクセスを制御するAPIトークンを作成します。

| 基本情報 | | | |
|--|---|----------|----|
| フォーム名称 | 契約書社内審査 (C001) | | |
| フォーム種別 | ワークフロー | | |
| + X-point認証トークンを生成する | | | |
| APIトークン | アクセス権 | 備考 | 操作 |
| NEW C001Lg!Ao5eMn1kCrotaQCbaaXX=sJ5AmG6ogNx | <input checked="" type="checkbox"/> 閲覧 <input type="checkbox"/> 追加 <input type="checkbox"/> 更新 <input type="checkbox"/> 削除 kintone FQDN API Token | 登録 更新 | 削除 |
| kintoneアプリに指定する連携用 JavaScript API のURL 【マニュアル】 [https://xpoint-develop-sunborn.at-ledcloud.net/xpoint/js/kintone.js] [アプリの設定]>「JavaScript / CSSでカスタマイズ」のPC用JavaScriptファイル、スマートフォン用JavaScriptファイルに追加します。 | | | |
| kintone連携APIトークン生成について 【マニュアル】 kintoneのアプリから直接X-pointの自動申請機能を利用する際に指定するAPIトークンを設定します。 X-pointに書類を申請する際に添付ファイルやコメントの転送を行う場合は「kintone FQDN」、kintoneアプリ側の「API Token」の設定も必要になります | | | |
| <input type="button" value="保存"/> | | | |

「API トークン」・・・kintone から X-point の操作を行う場合に指定するトークンを表示します。

「閲覧」・・・ kintone から X-point の書類ビューを開く場合、書類内容の取得 API を利用する場合にチェックします。

「追加」・・・ kintone から申請を行う場合にチェックします。

「更新」・・・ kintone から書類内容の更新を行う場合にチェックします。kintone から X-point に送る連携データ内に更新対象となる X-point の書類 ID を含める必要があります。

「削除」・・・ kintone から書類の削除を行う場合にチェックします。削除する際に、削除を行う書類 ID が必要になります。

「kintone FQDN」・・・kintone アプリに登録された添付ファイル、コメントを X-point に転送する場合に必要です。

「API Token」・・・kintone アプリに登録された添付ファイル、コメントを X-point に転送する場合に必要です。

転送元のアプリで「レコード閲覧」が許可された API トークンを設定してください。

「備考」・・・ API トークンの利用目的などを簡潔に記述します。

※ API トークンは、フォーム毎に最大 10 個登録できます。

※ API トークン欄の **NEW** は保存されていないことを示します。【保存】前に別画面に遷移すると入力は破棄されます。

※ kintone からの利用方法にあわせアクセス権の有無を使い分けます。

※ アクセス権の「kintone FQDN」、「API Token」を指定すると指定した API Token を持つ kintone アプリ専用の設定行になります。

■【保存】をクリックします。

保存後、生成された API トークン（40文字）を控えておきます。API トークンは kintone アプリ内に記述する JavaScript で使用します。

| 基本情報 | | | |
|---|---|--|----|
| フォーム名称 | 契約書社内審査 (C001) | | |
| フォーム種別 | ワークフロー | | |
| + X-point認証トークンを生成する | | | |
| APIトークン | アクセス権 | 備考 | 操作 |
| C001Lg!Ao5eMn1kCrotaQCbaaXX=sJ5AmG6ogNx | <input checked="" type="checkbox"/> 閲覧 <input checked="" type="checkbox"/> 追加 <input checked="" type="checkbox"/> 更新 <input type="checkbox"/> 削除 kintone FQDN API Token | 登録 管理者 (2020/10/21 17:47:24) 更新 管理者 (2020/10/21 17:47:24) | 削除 |
| | kintone FQDN xxxxxx.cybozu.com API Token 9fr9VE6umQloKph4llgClDafsqTSXcv5fNoB5zrQ | | |

3.2.3. 自動申請の登録

kintone から X-point に送られたデータは自動申請機能を介して申請処理が行われます。その為、連携に利用するフォームには必ず自動申請の登録を行います。

【登録において注意が必要なポイント】

- ① 1つの承認ルートに登録できる自動申請設定は1件だけ。

この制約により、連携対象のフォームで「条件付ルート」（自動選択ルート）を利用した自動申請登録が存在する場合、kintone との連携利用を考慮していない自動申請登録（予約項目が無い）を利用しての連携はできません。

連携を可能にするには、既存の自動申請登録の内容を修正し、API 経由での kintone 連携と共通利用ができるよう、CSV フォーマットに kintone の「ユーザーコード」、「レコード ID」などの列を追加する必要があります。

- ② 取り込みを行うファイルの指定で、ファイル名が一致する既存の自動申請設定があってはならない。

kintone との連携を CSV ファイルで行う場合に必要となりますので必ず一意となるように指定してください。

API 経由の連携のみで CSV ファイルによる連携を行わない場合であっても設定が必要になります。

■管理者サイトの自動申請画面で【作成】をクリックします。



※ 登録済みの自動申請を利用する場合、作成する必要はありません。

■ 自動申請の登録を行います。

kintone から受け取るデータ列の定義を行います。kintone からの連携では必須の定義になります。

指定項目の取り込みファイル名の指定は省略できません。kintone からの連携を JavaScript API のみで行う場合、ファイル名は既存の自動申請と重複しない適当な名前を入力してください。(例: kintoneAppid55.csv など)

ファイル名を付ける際に「kintone_{ドメイン CD}_{フォーム CD}-{枝番}.csv」になるようにすると、ファイル名の重複を指定されることは無くなると考えられます。

API 連携、CSV ファイル連携
何れの場合も指定が必須です

CSV ファイル連携で指定が必須です。
目的に合わせて指定します。
API 連携のみの場合、
成功・失敗ともに【何もしない】を指定します。

必ず「項目行として無視する」を選ぶ

必ず「先頭列を書類切換えに利用」を選ぶ

必ず「*」

必ず「」（空欄）

※ 「ファイル名」は省略できません。

- 既存の自動申請登録に「フォーム名」と「申請ルート」が同一のものがあると登録できません。

“{ドメイン CD}_kintone_{kintone AppID}.csv” のような名称を指定すると重複する確率が低くなります。

- API 連携では使用しませんが「取り込みを行うファイル」の指定は必須です。

※ 「申請後の取り込みファイル処理」は CSV ファイルを利用した場合に必須になります。

※ 必ず「取り込みファイル一行目の処理」は「項目行として無視する」、「申請後の切換え判定項目」は「先頭列を書類切換えに利用」（先頭文字は「*」、継続文字は空欄）を指定して下さい。

■連携フィールドの指定を行います。

kintone からのデータとフォーム上に格納するフィールドを対応付けます。API 連携では 1～6 が予約項目になり、JavaScript で追加指定を行う項目は 7 番目以降にセットされます。なお、テーブル形式を含むデータの場合、1 行目を示すマークが行頭に増えますが、対応フィールドの指定を行う画面では行頭文字列の存在は無視して指定を行います。

| CSVフォーマット | | | 一括削除 | 行の追加 |
|--------------------------|----|--------------------------------------|------|------|
| <input type="checkbox"/> | No | フィールド名 (フィールドID) | 並び替え | |
| <input type="checkbox"/> | 1 | ユーザーコード (usercode) [page-1] | 挿入 | 削除 |
| <input type="checkbox"/> | 2 | --- 該当フィールドなし (取り込みを行わない) --- | 挿入 | 削除 |
| <input type="checkbox"/> | 3 | --- 該当フィールドなし (取り込みを行わない) --- | 挿入 | 削除 |
| <input type="checkbox"/> | 4 | --- 該当フィールドなし (取り込みを行わない) --- | 挿入 | 削除 |
| <input type="checkbox"/> | 5 | --- 該当フィールドなし (取り込みを行わない) --- | 挿入 | 削除 |
| <input type="checkbox"/> | 6 | レコードID (kintone_id) [page-1] | 挿入 | 削除 |
| <input type="checkbox"/> | 7 | 申請日 (txtShinseibi) [page-1] | 挿入 | 削除 |
| <input type="checkbox"/> | 8 | 申請者 (txtShinseisy) [page-1] | 挿入 | 削除 |
| <input type="checkbox"/> | 9 | 顧客コード (txtKokyakucode) [page-1] | 挿入 | 削除 |
| <input type="checkbox"/> | 10 | 顧客名 (txtKokyakumei) [page-1] | 挿入 | 削除 |
| <input type="checkbox"/> | 11 | 担当名 (txtTantou) [page-1] | 挿入 | 削除 |
| <input type="checkbox"/> | 12 | 完了希望日 (txtKiboubi) [page-1] | 挿入 | 削除 |
| <input type="checkbox"/> | 13 | 管理コード (txtKanricode) [page-1] | 挿入 | 削除 |
| <input type="checkbox"/> | 14 | 審査主旨 (txtMemo) [page-1] | 挿入 | 削除 |
| <input type="checkbox"/> | No | フィールド名 (フィールドID) | 並び替え | |
| | | 表定義グループ (group1) [page-1] | | |
| <input type="checkbox"/> | 15 | 契約書番号 (txtKeiyakusyono) [page-1] | 挿入 | 削除 |
| <input type="checkbox"/> | 16 | 契約書名 (txtKeiyakusyomei) [page-1] | 挿入 | 削除 |

1 列目に「ユーザーコード」を指定します。
 2～5 列目は必ず「該当フィールド無し」を指定します。
 6 列目に「レコード ID」を指定します。

【予約列の内容】

- ① ユーザーコード . . . kintone のログイン名
- ② レコード番号 . . . kintone の登録レコード番号 (アプリ毎に採番されている番号で ID と同じ)
 アプリコードが指定されている場合は「(アプリコード) - (レコード番号)」になります。
- ③ 更新日時 . . . kintone でのレコード更新日時 (yyyy/mm/dd HH:MM:SS)
- ④ 作成日時 . . . kintone でのレコード作成日時 (yyyy/mm/dd HH:MM:SS)
- ⑤ リビジョン番号 . . . 受信したレコードのリビジョン番号 (1 ~)
- ⑥ ID . . . 受信したレコードの ID 番号 (アプリ毎の採番番号で必ず数値になる。1 ~ 99, 999)

【画面の項目】

- チェック Box . . . チェック後に【一括削除】をクリックするとチェック行が削除できます。
- フィールド名 . . . 対応付けるフォーム上の項目を指定します。
- 並び替え . . . フィールドの順番を入れ替えるダイアログが表示されます。
- 挿入 . . . クリック位置の前に空行が追加されます。
- 削除 . . . クリック位置の行が削除されます。
- 表グループ . . . 連携する表グループを一つ選びます。複数選択できません。

本項では書類への反映を行わない項目 2～5 を取り込み対象外として除外します。運用上必要がある場合は格納先となるフィールドを指定して下さい。項目 7 以降は kintone からの申請情報に含まれる項目の並び (JavaScript で指定する順番) に合わせて順に指定します。

対応付けが完了したら【次へ】をクリックします。

申請ユーザーの指定画面では、「CSV データ指定」を選び、「No.1」(ユーザーコード)の欄で【申請ユーザ】をチェックする指定が基本になります。kintone の利用ユーザーが X-point 上にログイン ID を持たない場合は「ユーザ指定 (一人)」を選び特定の申請者になる設定を行います。

【kintone のログイン名と同じ X-point ユーザーコードを使っている場合】

自動申請設定 一覧 + 作成 タスクログ

申請ユーザ設定・申請フォームで一意となる値の設定を行います。

戻る 次へ

フォーム情報

申請ユーザタイプ ※
 ユーザ指定 (一人) ユーザ 管理者

 CSVデータ指定
※指定したユーザが、一括自動申請書類の申請ユーザとなります。

| No | フィールド名 (フィールドID) | ページ | 申請ユーザ | キー |
|----|------------------------------|-----|----------------------------------|-------------------------------------|
| 1 | ユーザーコード (usercode) | 1 | <input checked="" type="radio"/> | <input type="checkbox"/> |
| 2 | --- 該当フィールドなし (値をセットしない) --- | 1 | | |
| 3 | --- 該当フィールドなし (値をセットしない) --- | 1 | | |
| 4 | --- 該当フィールドなし (値をセットしない) --- | 1 | | |
| 5 | --- 該当フィールドなし (値をセットしない) --- | 1 | | |
| 6 | レコードID (kintone_id) | 1 | <input type="radio"/> | <input checked="" type="checkbox"/> |
| 7 | 申請日 (txtShinseibi) | 1 | <input type="radio"/> | <input type="checkbox"/> |
| 8 | 申請者 (txtShinseisya) | 1 | <input type="radio"/> | <input type="checkbox"/> |

連携 API を使用した場合、項目 1 には kintone にログインしているユーザー名、もしくは連携 API のオプション項目 “usercd”, “proxyusercd” に指定されたコードが設定され処理が行われます。

【kintone のログイン名と X-point のユーザーコードが一致しない、X-point にはログイン ID が無い場合】

自動申請設定 一覧 + 作成 タスクログ

申請ユーザ設定・申請フォームで一意となる値の設定を行います。

戻る 次へ

フォーム情報

申請ユーザタイプ ※
 ユーザ指定 (一人) ユーザ 管理者

 CSVデータ指定
※指定したユーザが、一括自動申請書類の申請ユーザとなります。

| No | フィールド名 (フィールドID) | ページ | 申請ユーザ | キー |
|----|------------------------------|-----|-----------------------|-------------------------------------|
| 1 | ユーザーコード (usercode) | 1 | <input type="radio"/> | <input type="checkbox"/> |
| 2 | --- 該当フィールドなし (値をセットしない) --- | 1 | <input type="radio"/> | |
| 3 | --- 該当フィールドなし (値をセットしない) --- | 1 | <input type="radio"/> | |
| 4 | --- 該当フィールドなし (値をセットしない) --- | 1 | <input type="radio"/> | |
| 5 | --- 該当フィールドなし (値をセットしない) --- | 1 | <input type="radio"/> | |
| 6 | レコードID (kintone_id) | 1 | <input type="radio"/> | <input checked="" type="checkbox"/> |
| 7 | 申請日 (txtShinseibi) | 1 | <input type="radio"/> | <input type="checkbox"/> |
| 8 | 申請者 (txtShinseisya) | 1 | <input type="radio"/> | <input type="checkbox"/> |

- 登録内容を確認し【完了】をクリックします。
- 【完了】をクリックすると設定が保存されます。

自動申請設定

以下の内容で登録します。よろしいですか？

戻る

完了

| フォーム情報 | |
|-----------------|---|
| 自動申請名称 | from kintone |
| フォーム名 (フォームコード) | 契約書社内審査 (C001) |
| 申請ルート (ルートコード) | 自動選択ルート |
| 取り込みを行うファイル | ファイル名: kintoneAppid55.csv |
| 申請後の取り込みファイルの処理 | 成功時: 何もしない 失敗時: 何もしない |
| 一行目の処理 | 項目行として無視する |
| 表定義を含む申請の切換え判定 | 取り込みファイルの先頭列を書類切換えに利用 (先頭行文字: *, 継続行文字:) |
| 備考 | |
| 申請ユーザタイプ | CSVデータ指定 |
| 作成日 | |
| 最終更新日時 | |

CSVフォーマット

| No | フィールド名 (フィールドID) | ページ | 申請ユーザ | キー |
|----|------------------------------|-----|-------|----|
| 1 | ユーザコード (usercode) | 1 | ● | |
| 2 | --- 該当フィールドなし (値をセットしない) --- | 1 | | |
| 3 | --- 該当フィールドなし (値をセットしない) --- | 1 | | |
| 4 | --- 該当フィールドなし (値をセットしない) --- | 1 | | |
| 5 | --- 該当フィールドなし (値をセットしない) --- | 1 | | |
| 6 | レコードID (kintone_id) | 1 | | ● |
| 7 | 申請日 (txtShinseibi) | 1 | | |
| 8 | 申請者 (txtShinseisya) | 1 | | |
| 9 | 顧客コード (txtKokyakucode) | 1 | | |
| 10 | 顧客名 (txtKokyakumei) | 1 | | |
| 11 | 担当者 (txtTantou) | 1 | | |
| 12 | 完了希望日 (txtKiboubi) | 1 | | |
| 13 | 管理コード (txtKanricode) | 1 | | |
| 14 | 審査主旨 (txtMemo) | 1 | | |
| 15 | 契約書番号 (txtKeiyakusyono) | 1 | | |
| 16 | 契約書名 (txtKeiyakusyomei) | 1 | | |

戻る

完了

第4章. kintoneアプリから X-pointへの連携

kintone アプリから書類提出を実行する場合は kintone アプリの JavaScript API を利用し kintone に提出処理を組み込みます。事前準備で kintone アプリに「kintone 連携サービス」用の JavaScript ライブラリを組み込み済みのため、本項は kintone のレコード明細（フォーム画面）にボタンを配置しボタンクリックで X-point への申請を行う場合を例として解説します。

- ※ kintone アプリの開発に関する各種情報については「cybozu developer network」をご確認ください。
- ※ 本章で扱う内容は、kintone アプリで JavaScript API を利用した開発を十分理解していることを前提にしています。
- ※ JavaScript 作成は kintone の「kintone JavaScript コーディングガイドライン」に従ってください。


<https://cybozu.dev/ja/kintone/docs/guideline/coding-guideline/>

4.1. 連携テンプレートを利用して連携を行う

kintone アプリからの書類申請は必ず JavaScript を利用します。その際、kintone アプリに組み込む JavaScript の作成に利用できる JavaScript テンプレートを X-point に生成させることが出来ます。

書類申請の際にフィールドに格納されている情報を JavaScript で加工（連結、分割など）を必要とするケースを除き、テンプレートをそのまま利用できるため、連携動作に必要な設定作業が簡単になります。

4.1.1. 連携用 JavaScript テンプレートの取得

JavaScript テンプレートは、kintone 連携設定でフォームに kintone から X-point の書類情報へのアクセスを制御する API トークンを発行し、かつ当該フォームを利用した自動申請の「申請書類の切換え判定項目」を「先頭列を書類切換えに利用」に指定した場合に入手可能になります。入手可能になるとフォーム管理の「kintone 連携一覧」画面右側にダウンロードアイコンが表示され、このアイコンをクリックすると「{ドメイン名}-{フォームコード}.js」というファイルがダウンロードできます。このファイルを「連携用 JavaScript テンプレート」と呼びます。


☰ フォーム管理

→ フォーム → 共通承認ルート → CSVテンプレート → コンポーネント → 提出期限 → 督促メール → 自動申請

kintone連携・設定フォーム一覧

☰ 一覧 ☑ フォームグループ 📄 フォーム 📄 複数枚フォーム ☰ kintone連携一覧

kintone→X-point 連携データ受信フォーム

| フォーム名称 | フォームCD | APIトークン | 更新日時 | Download |
|-----------|--------|--|---------------------|---|
| A368出張申請書 | A368 | A368K#jlvI1@fOYMXfWpRksiW=KkMLIOP4LkGbhN | 2020/10/21 14:56:33 | 設定無し |
| 契約書社内審査 | C001 | C001LgIAo5eMn!1kCrotaQCbaaXX=sJ5AmG6ogNx | 2020/10/21 17:47:24 |  |

 kintone→X-point 連携データ受信フォーム [マニュアル]
kintoneからのレコード受信で利用可能なAPIトークンの発行と、自動申請設定の登録が行われたフォームのみ表示されます。
連携に利用可能な登録が完了していないフォームの連携JavaScriptの雛型はダウンロードできません。

[X-point→kintone 連携データ送信フォーム]用のAPIトークンは登録されていません。

4.1.2. 連携用 JavaScript テンプレートの修正

連携テンプレートには、kintone から X-point に書類申請を行うための JavaScript クラスが定義され、クラスはフォーム毎に一つの出力となります。その為、同一フォームに複数の kintone 連携用の自動申請が定義されている場合、クラス内に複数のフィールド対応を記述する部分が含まれます。

■クラス名

テンプレートは、X-point のフォームコードをクラス名として作成されます。

```
KeiyakuShinsa=new function(xpoint_api_token, xpoint_routecd) {  
  :  
};
```

フォームコードが使用される

コンストラクタは、X-point→kintone で利用する API トークンと自動申請で指定した承認ルート CD になります。

その為、使用する際は最もシンプルな記述の場合は次のような形式になります。

```
var k2x = KeiyakuShinsa("KeiyakuShinsaM!!RyeHRFYnfBI9Ftrxn01IHLzd", "KeiyakuShinsa");  
k2x.sendDetailPage(event);
```

通常フォーム、条件付きルート(自動選択ルート)を利用するワークフローフォームは、クラス利用時にルート CD の指定を空文字列にします。

■オプション属性

連携テンプレートではインスタンスのプロパティとして設定指定を行います。

連携用の基本 API (kintone2xpoint クラス) を利用する場合、オプション属性はコンストラクタに指定を行いますが、

1) 「固定ユーザーコード」の指定

post_usercd プロパティに固定するユーザーコードを指定します。

```
var k2x = KeiyakuShinsa("KeiyakuShinsaM!!RyeHRFYnfBI9Ftrxn01IHLzd", "KeiyakuShinsa");  
k2x.post_usercd = "u001";
```

指定しない場合は kintone にログイン中のユーザー名が使用されます。プロパティで記述する場合は、インスタンスを作成する際は都度の指定が必要になります。

kintone の利用ユーザーが X-point にアカウントを持たないようなケースでは、常に同じユーザーによる連携が行われるようテンプレートファイル内に記述します。

```
this.post_usercd = 'u001';
```

※ 常に“u001”ユーザーで書類の申請、更新を行う場合の指定です。

2) 「代理ユーザーコード」の指定

post_proxycd プロパティに代理するユーザーコードを指定します。

代理ユーザーコードが使用されるのは、ログイン中のユーザー名と一致するユーザーコードを持つ登録が X-point 内に存在しない場合です。常に代理する動作ではありません。

```
var k2x = KeiyakuShinsa("KeiyakuShinsaM!!RyeHRFYnfBI9Ftrxn01IHLzd", "KeiyakuShinsa");  
k2x.post_proxycd = "u001";
```

指定しない場合は kintone にログイン中のユーザー名が使用されます。

固定ユーザーコード (post_usercd プロパティ) が指定されている場合は、固定ユーザーコードが優先されます。

登録ユーザーが見つからない場合の規定動作として特定のユーザーアカウントで代理申請を行うようにする場合は、テンプレートファイル内に記述します。

```
this.post_proxycd = 'u001';
```

※ 常に“u001”ユーザーで書類の申請、更新の代理を行う場合の指定です。

3) 「コメント登録フラグ」の指定

X-point に書類の申請（更新）を行う際に、kintone アプリのレコードに記入されているコメントを全て X-point の書類に追加します。変更箇所はテンプレートファイル内の「this.post_comment」に true / false で指定します。

```
this.post_comment = 'true'; // コメントを登録する(true), 登録しない(false)
```

テンプレートファイル内に記述しない場合はプロパティ値として代入します。但し、プロパティで記述する場合は、インスタンスを作成する際は都度の指定が必要になります。

```
var k2x = KeiyakuShinsa("KeiyakuShinsaM!!RyeHRFYnfB19Ftrxn01IHLzd", "KeiyakuShinsa");  
k2x.post_comment = true;
```

但し、kintone から X-point の場合はコメントの記入日、内容が一致しているコメントは追加コメントより除外されます。

4) 「ゲストスペース ID」の指定

kintone アプリをゲストスペースで利用する場合、ゲストスペース ID の指定が必要になります。

変更箇所はテンプレートファイル内の「this.guestspace_id」にゲストスペースの ID 番号を記述します。

```
this.guestspace_id = 15;
```

※ ゲストスペース ID が 15 の場合。

テンプレートファイル内に記述しない場合はプロパティ値として代入します。但し、プロパティで記述する場合は、インスタンスを作成する際は都度の指定が必要になります。

```
var k2x = KeiyakuShinsa("KeiyakuShinsaM!!RyeHRFYnfB19Ftrxn01IHLzd", "KeiyakuShinsa");  
k2x.guestspace_id = 15;
```

■kintone アプリ内での書類 ID 格納フィールドコード (xp_docid_fieldcd)

書類を更新する際に対象となる X-point の書類 ID を得るための項目。指定しない場合、常に新規提出で動作します。

指定は連携テンプレート内、つまりファイルを編集して行います。

変更箇所はテンプレートファイル内の「this.xp_docid_fieldcd = ''」に書類 ID を保存するフィールドコードを記入します。

```
this.xp_docid_fieldcd = 'xpoint_docid';
```

テンプレートファイル内に記述しない場合はプロパティ値として代入します。但し、プロパティで記述する場合は、インスタンスを作成する際は都度の指定が必要になります。

```
var k2x = KeiyakuShinsa("KeiyakuShinsaM!!RyeHRFYnfB19Ftrxn01IHLzd", "KeiyakuShinsa");  
k2x.xp_docid_fieldcd = 'xpoint_docid';
```

■添付ファイル格納フィールドコード (xp_attached_fieldcd)

書類に添付するファイルが格納されている添付ファイルフィールドコードを配列形式で列挙します。

指定は連携テンプレート内、つまりファイルを編集して行います。

変更箇所はテンプレートファイル内の「this.xp_attached_fieldcd = []」に添付ファイルを保存するフィールドコードを記入します。

```
this.xp_attached_fieldcd = [ 'attached_file' ];
```

複数の添付ファイルフィールドを書類に添付する場合は“,”で区切り列挙します。

```
this.xp_attached_fieldcd = [ 'attached_file1', 'attached_file2' ];
```

テンプレートファイル内に記述しない場合はプロパティ値として代入します。但し、プロパティで記述する場合は、インスタンスを作成する都度の指定が必要になります。

```
var k2x = KeiyakuShinsa("KeiyakuShinsaM!!RyeHRFYnfBI9Ftrxn011HLzd", "KeiyakuShinsa");  
k2x.xp_attached_fieldcd = [ 'attached_file' ];
```

kintone アプリのテーブル内で添付ファイル格納フィールドの指定も可能です。但し、テーブル内の特定行のみと言う指定はできません。テーブル内の添付フィールドを指定した場合は、指定した添付フィールド列の全てが X-point の書類に添付されます。

■フィールド対応の定義 (lumpy_list)

X-point と kintone で対になるフィールドを定義します。初期状態では X-point の自動申請の定義内容を基にフォーム ID 利用した対応表が配列形式で作成された状態になります。もし、X-point と kintone のフィールド ID とフィールドコードが同じ、グループ ID とテーブルコードが同じ、格納形式の変換が既定動作のまま問題ない場合は、テンプレートに記述されている内容のまま連携が動作します。

【出力テンプレートのフィールド対応付け部分】

フィールドの対応は、X-point のフィールド ID が "xname" プロパティに、kintone のフィールドコードが "field" プロパティに指定されます。表グループは "tableItem" 内に、表グループ ID とテーブルコード ID の対応と合わせて定義されます。

```

this.lumpy_list = [
{
  "routeCd": "KeiyakuShinsa",
  "config": {
    "fieldList": [
      // X-point フィールド ID,      kintone フィールドコード. 表以外の対応
      { "xname": "txtShinseibi",      "field": "txtShinseibi" },
      { "xname": "txtShinseisya",    "field": "txtShinseisya" },
      { "xname": "txtNo",            "field": "txtNo" },
      { "xname": "txtKokyakucode",    "field": "txtKokyakucode" },
      { "xname": "txtKokyakumei",    "field": "txtKokyakumei" },
      { "xname": "txtTantou",        "field": "txtTantou" },
      { "xname": "txtKiboubi",       "field": "txtKiboubi" },
      { "xname": "txtKanricode",     "field": "txtKanricode" },
      { "xname": "txtMemo",          "field": "txtMemo" }
    ],
    "tableItem": {
      // X-point グループ ID,      kintone テーブルコード. 表定義の対応
      "xpoint_groupid": "group1",  "kintone_tablecode": "group1",
      "fieldList": [
        // X-point フィールド ID,      kintone フィールドコード.
        { "xname": "txtKeiyakusyono", "field": "txtKeiyakusyono" },
        { "xname": "txtKeiyakusyomei", "field": "txtKeiyakusyomei" }
      ]
    }
  }
}
]

```

- ※ kintone 側のフィールドコードには X-point のフィールド ID と同じ名前が初期設定されます。
- ※ フィールド対応付け部分は、自動申請の登録情報を基に作成されます。

○自動申請の CSV フォーマットに対応する部分

プロパティの“xname”の並びは自動申請の CSV フォーマットと一致します。kintone 連携の自動申請は先頭の 6 列が予約列になるため、連携テンプレートに出力されるのは 7 列目以降の項目になります。

```

this.lumpy_list = [
{
  "routedc": "KeiyakuShinsa",
  "config": {
    "fieldList": [
      // X-point フィールド ID, kintone フィールドコード.
      { "xname": "txtShinseibi", "field": "txtShinseibi" },
      { "xname": "txtShinseisya", "field": "txtShinseisya" },
      { "xname": "txtNo", "field": "txtNo" },
      { "xname": "txtKokyakucode", "field": "txtKokyakucode" },
      { "xname": "txtKokyakumei", "field": "txtKokyakumei" },
      { "xname": "txtTantou", "field": "txtTantou" },
      { "xname": "txtKiboubi", "field": "txtKiboubi" },
      { "xname": "txtKanricode", "field": "txtKanricode" },
      { "xname": "txtMemo", "field": "txtMemo" }
    ],
    "tableItem": {
      // X-point グループ ID, kintone テーブルコード.
      "xpoint_groupid": "group1", "kintone_tablecode": "group1",
      "fieldList": [
        // X-point フィールド ID, kintone フィールドコード.
        { "xname": "txtKeiyakusyono", "field": "txtKeiyakusyono" },
        { "xname": "txtKeiyakusyomei", "field": "txtKeiyakusyomei" }
      ]
    }
  }
}
]
}

```

表以外のフィールド ID 部分 (基本的に修正しない)

表明細定義の ID 名 (修正してはいけない)

表明細内の項目 ID 名 (基本的に修正しない)

○X-point のフィールド ID と kintone のフィールドコードが異なる場合の修正箇所

フォームのフィールド ID と kintone のフィールドコードが異なる場合は、“field” プロパティの値を修正します。

```

this.lumpy_list = [
{
  "routedc": "KeiyakuShinsa",
  "config": {
    "fieldList": [
      // X-point フィールド ID, kintone フィールドコード.
      { "xname": "txtShinseibi", "field": "txt_postdate" },
      { "xname": "txtShinseisya", "field": "txt_postuser" },
      { "xname": "txtNo", "field": "txt_number" },
      { "xname": "txtKokyakucode", "field": "txt_customercode" },
      { "xname": "txtKokyakumei", "field": "txt_customername" },
      { "xname": "txtTantou", "field": "txt_personname" },
      { "xname": "txtKiboubi", "field": "txt_storecode" },
      { "xname": "txtKanricode", "field": "txt_manegecode" },
      { "xname": "txtMemo", "field": "txt_notice" }
    ],
    "tableItem": {
      // X-point グループ ID, kintone テーブルコード.
      "xpoint_groupid": "group1", "kintone_tablecode": "group1",
      "fieldList": [
        // X-point フィールド ID, kintone フィールドコード.
        { "xname": "txtKeiyakusyono", "field": "txt_agree_no" },
        { "xname": "txtKeiyakusyomei", "field": "txt_agree_title" }
      ]
    }
  }
}
]
}

```

kintone の独自名のフィールドコードに書き換える

テーブルコードに書き換える

kintone の独自名のフィールドコードに書き換える

○X-point のフィールド ID と kintone のフィールドコードが異なる場合の修正箇所

フィールドコードが異なる名前の場合は“field”の後に続く文字列を書き換え、固定値になる部分は“field”を“value”に書き換えます。

値の指定は、申請（更新）API を実行する際に逐次処理を行うコールバック関数を指定して行うことも出来ます。コールバック関数を利用した場合、申請毎に内容を変える、テーブル部分は行毎に内容を変える事が可能になります。

```
    this.lumpy_list = [
    {
      "routeCd": "KeiyakuShinsa",
      "config": {
        "fieldList": [
          // X-point フィールド ID,      kintone フィールドコード
          { "xname": "txtShinseibi",      "field": "txt_postdate"  },
          { "xname": "txtShinseisya",    "field": "txt_postuser"  },
          { "xname": "txtNo",             "field": "txt_number"    },
          { "xname": "txtKokyakucode",    "value": "atled"        },
          { "xname": "txtKokyakumei",     "value": "(株)エイトレッド"},
          { "xname": "txtTantou",         "value": "勝 太郎"     },
          { "xname": "txtKiboubi",        "field": "txt_storecode" },
          { "xname": "txtKanricode",      "field": "txt_manegecode"},
          { "xname": "txtMemo",           "field": "txt_notice"   }
        ],
        "tableItem": {
          // X-point グループ ID,      kintone テーブルコード.
          "xpoint_groupid": "group1",  "kintone_tablecode": "group1",
          "fieldList": [
            // X-point フィールド ID,      kintone フィールドコード.
            { "xname": "txtKeiyakusyono", "field": "txt_agree_no"  },
            { "xname": "txtKeiyakusyomei", "field": "txt_agree_title"}
          ]
        }
      }
    }
  ]
}
```

コード名を書き換える

“value”で固定値を常にセットする

※ コールバック関数を利用した記述方法は、「[X-point に書類を申請](#)」、「[一括申請](#)」の項を参照。

最後に作成した項目数が使用する自動申請の CSV フォーマット列数と一致しているかを確認します。

```

this.lumpy_list = [
{
  "routedc": "KeiyakuShinsa",
  "config": {
    "fieldList": [
      // X-point フィールド ID          kintone フィールドコード
      { "xname": "txtShinseibi",    "field": "txt_postdate"  },
      { "xname": "txtShinseisya",  "field": "txt_postuser"  },
      { "xname": "txtNo",          "field": "txt_number"    },
      { "xname": "txtKokyakucode",  "value": "atled"        },
      { "xname": "txtKokyakumei",  "value": "(株)エイトレッド"},
      { "xname": "txtTantou",      "value": "勝 太郎"      },
      { "xname": "txtKiboubi",     "field": "txt_storecode" },
      { "xname": "txtKanricode",   "field": "txt_manegecode" },
      { "xname": "txtMemo",        "field": "txt_notice"   }
    ],
    "tableItem": {
      // X-point グループ ID,          kintone テーブルコード.
      "xpoint_groupid": "group1",    "kintone_tablecode": "group1",
      "fieldList": [
        // X-point フィールド ID          kintone フィールドコード
        { "xname": "txtKeiyakusyono", "field": "txt_agree_no"  },
        { "xname": "txtKeiyakusyomei", "field": "txt_agree_title" }
      ]
    }
  }
}
]

```

対応する表の ID になること

表明細の列数が一致すること

7 列目からの数が一致すること

☰ フォーム管理 → フォーム → 共通承認ルート → CSVテンプレート → コンポーネント → 提出期限 → 督促

| CSVフォーマット | | | 一括 |
|--------------------------|----|--------------------------------------|----|
| <input type="checkbox"/> | No | フィールド名 (フィールドID) | |
| <input type="checkbox"/> | 1 | ユーザーコード (usercode) [page-1] | ▼ |
| <input type="checkbox"/> | 2 | --- 該当フィールドなし (取り込みを行わない) --- | ▼ |
| <input type="checkbox"/> | 3 | --- 該当フィールドなし (取り込みを行わない) --- | ▼ |
| <input type="checkbox"/> | 4 | --- 該当フィールドなし (取り込みを行わない) --- | ▼ |
| <input type="checkbox"/> | 5 | --- 該当フィールドなし (取り込みを行わない) --- | ▼ |
| <input type="checkbox"/> | 6 | レコードID (kintone_id) [page-1] | ▼ |
| <input type="checkbox"/> | 7 | 申請日 (txtShinseibi) [page-1] | ▼ |
| <input type="checkbox"/> | 8 | 申請者 (txtShinseisya) [page-1] | ▼ |
| <input type="checkbox"/> | 9 | 顧客コード (txtKokyakucode) [page-1] | ▼ |
| <input type="checkbox"/> | 10 | 顧客名 (txtKokyakumei) [page-1] | ▼ |
| <input type="checkbox"/> | 11 | 担当名 (txtTantou) [page-1] | ▼ |
| <input type="checkbox"/> | 12 | 完了希望日 (txtKiboubi) [page-1] | ▼ |
| <input type="checkbox"/> | 13 | 管理コード (txtKanricode) [page-1] | ▼ |
| <input type="checkbox"/> | 14 | 審査主旨 (txtMemo) [page-1] | ▼ |
| <input type="checkbox"/> | | 表定義グループ (group1) [page-1] | ▼ |
| <input type="checkbox"/> | 15 | 契約書番号 (txtKeiyakusyono) [page-1] | ▼ |
| <input type="checkbox"/> | 16 | 契約書名 (txtKeiyakusyomei) [page-1] | ▼ |

○修正を行った連携テンプレートファイルを kintone アプリに登録します。

kintone アプリの設定画面より「JavaScript/CSSのカスタマイズ」を開き、PC用、スマートフォン用の両方に追加します。

保存 キャンセル

JavaScript / CSSでカスタマイズ

JavaScript / CSSファイルを活用して、アプリをカスタマイズできます。
JavaScriptの開発にあたっては、[JavaScriptコーディングガイドライン](#)をご確認ください。

適用範囲
 すべてのユーザーに適用 アプリ管理者だけに適用 適用しない

PC用のJavaScriptファイル

スマートフォン用のJavaScriptファイル

4.1.3. 連携用 JavaScript で特殊な扱いをする項目

連携を行う際、kintone と X-point の間でフィールドが一对一にならない項目、扱いに注意が必要なフィールドについて説明します。

4.1.3.1. 西暦、月、日フィールド

連携用 JavaScript でフィールド対応の定義を行う際、X-point 側が日付セットになる場合に限り指定可能な方法があります。kintone のフィールドが日付、もしくは日時で格納されている情報を、日付セットに格納する場合に利用します。

kintone のフィールド

2017-07-07

2018-08-08 15:30

X-point の日付セット

(西暦、月、日の3フィールド)

2017

06

01

kintone では一つのフィールドにセットされている日付を X-point の書類に設定する際に3フィールドに分割する場合は、JavaScript の対応リスト部分で kintone 側を全て同じフィールドコードで記述します。

```

:
  "fieldList": [
    // X-point フィールド ID,      kintone フィールドコード.
    :
      { "xname": "yearfield",      "field": "yymddfield" }
      { "xname": "monthfield",    "field": "yymddfield" }
      { "xname": "datefield",     "field": "yymddfield" }
    :
  ]
:
  セットする個々のフィールド ID      全て同じフィールドコード

```

X-point は日付、日時を表現する文字列が入力された際にフォームのフィールド・タイプが西暦フィールド、月フィールド、日フィールドの場合に格納先のフィールドに適した情報に変換してセットします。

変換される日付、日時は文字列が次のパターンでに該当する場合には限られます。

【変換が動作するパターン】

日付 : YYYY-MM-DD, YYYY/MM/DD, YY-MM-DD, YY/MM/DD

日時 : YYYY-MM-DD hh:mm, YYYY/MM/DD hh:mm, YY-MM-DD hh:mm, YY/MM/DD hh:mm

4.1.3.2. チェックボックス

kintone ではチェックボックスに複数の項目を持つことが出来ますが、X-point は1つだけになります。その為、複数のチェック項目を持つフィールドを X-point のフィールドと対応させることはできません。

【連携に使用できないタイプ】



項目が2つあるため**連携ができない**。

連携する場合は項目を1つにしなければいけない。

※ チェックボックスを X-point から kintone への連携のみで扱う場合は、kintone 上で複数のチェック項目を扱っても構いません。kintone から X-point に申請・更新する場合に限っての制約になります。

※ 複数項目を持つチェックボックスを X-point のチェックボックスと連携した場合、X-point のチェック項目では指定できない内容が格納される場合があります。指定できない値が格納されると、X-point では未選択になります。

【連携に使用できるタイプ】



項目が1つであるため**連携が可能**

連携が可能なタイプの場合であっても、kintone では非チェック時の値を持つことが出来ないため、eFormMaker では非チェック値を必ず空に指定します。

【チェックボックスのプロパティ】

| | |
|----------|---------|
| 色 | #000000 |
| 初期値 | |
| チェックマーク | ■ |
| 非チェックマーク | □ |
| チェック値 | sample1 |
| 非チェック値 | |

【連携で注意が必要な項目】

初期値 : 必ず空欄にします

チェック値 : kintone の項目値と同じにします

非チェック値 : 必ず空欄にします

※ kintone には初期値を扱う仕組みがないため、kintone 側でチェックされない場合に、X-point の書類に初期値がセットされます。

4.1.3.3. ラジオボタン

kintone のラジオボタンフィールドは、X-point のラジオボタン・グループと対応させて扱います。

【kintone のフォーム】

フィールドの設定

フィールド名 *

radio_group

フィールド名を表示しない

項目と順番

sample1

sample2

並び

横 縦

初期値

sample1

フィールドコード *

radiobutton1

【X-point のフォーム】

項目数と同数のフィールドを配置します



フィールドのチェック値を kintone の項目とあわせませす

| | | | |
|----------|----------------------------------|----------|----------------------------------|
| 初期値 | sample1 | 初期値 | sample1 |
| チェックマーク | <input checked="" type="radio"/> | チェックマーク | <input checked="" type="radio"/> |
| 非チェックマーク | <input type="radio"/> | 非チェックマーク | <input type="radio"/> |
| チェック値 | sample1 | チェック値 | sample2 |

初期値をラジオボタン・グループ設定で kintone と同じにします

ラジオボタン・グループ設定(radio_radiobutton1)

| ID名 | メンバID | タグ名 |
|--------------|--------------|---------|
| radiobutton1 | radiobutton1 | ラジオボタン1 |
| radiobutton2 | radiobutton2 | ラジオボタン2 |

初期値 sample1

kintone で項目に指定された内容と、X-point のチェック値に指定された文字列は必ず一致する必要があります。

4.1.4. 連携用 JavaScript を kintone から実行する

登録したテンプレート JavaScript をアプリから実行できるようにします。

実行はアプリのレコード一覧画面、レコード詳細画面の上部にボタンを追加し、ボタンのクリックで実行できるようにします。

4.1.4.1. X-point に書類を申請

kintone アプリのレコード明細画面上部に【X-point に申請】ボタンを用意し書類申請用の API (sendDetailPage) を使い、ボタンクリックで明細の内容を利用して X-point に書類申請を行います。

■画面のイメージ

【PC 版 kintone】



【スマートフォン版 kintone】



ボタンを追加し、テンプレートの
“sendDetailPage” メソッドで申請します。

アプリに組み込む最もシンプルな JavaScript は次のようなイメージになります。

サンプルの**赤枠**で囲まれた部分が連携テンプレートの JavaScript クラスを実行する部分になります。その他の部分は kintone の開発手法に従います。画面イメージ例以外の組み込みを行う場合は、kintone developer site 等を参考に作業を行って下さい。

なお、テンプレート JavaScript を利用して X-point に書類申請・更新を行った場合、X-point から取得した書類 ID を kintone のレコードに格納を行う過程で必ずブラウザ画面のリフレッシュが実行されます。

```

(function () {
    "use strict";

    // レコード詳細表示イベント (PC)
    kintone.events.on('app.record.detail.show', function (event) {
        // メニュー右側の空白部分にボタンを設置
        var myBtn = document.createElement('button');
        myBtn.id = 'my_post_button';
        myBtn.innerHTML = 'X-point に申請';
        myBtn.onclick = function () {
            var k2x = fieldtest("fieldtestjX&k8fJ0EQkULGbi094Je2R@KS#VMqr", "fieldtest");
            k2x.sendDetailPage(event);
        };
        kintone.app.record.getHeaderMenuSpaceElement().appendChild(myBtn);
    });

    // レコード詳細表示イベント (スマートフォン)
    kintone.events.on('mobile.app.record.detail.show', function (event) {
        var elButton = kintone.mobile.app.getHeaderSpaceElement();
        var bt_div = document.createElement('div');
        bt_div.setAttribute('id', 'addButton');
        bt_div.setAttribute('name', 'addButton');
        var bt = document.createElement('button');
        bt.setAttribute('id', 'bt1');
        bt.onclick = function () {
            var k2x = fieldtest("fieldtestjX&k8fJ0EQkULGbi094Je2R@KS#VMqr", "fieldtest");
            k2x.sendDetailPage(event);
        };
        bt.innerHTML = 'X-point に申請';
        bt_div.appendChild(bt);
        elButton.appendChild(bt_div);
    });

})();

```

PC ブラウザ用

スマートフォン用

■Callback 処理を指定する場合

連携の“sendDetailPage”を呼び出す際、格納値を操作するための JavaScript 関数を指定することができます。

kintone のフィールドに格納されている値をそのままの状態 X-point の書類に格納する場合には必要ありませんが、フィールド値を使って文字列の分割や連結を行う場合には処理用の関数定義が必要になります。

基本的な記述方法は次の 2 種類になります。

【処理関数を別途定義する方法】

```

var rec_func = function(record, fieldList, tablecode, rowno) {
    ~~ 格納値を変更する処理 ~~
};
k2x.sendDetailPage(event, rec_func);

```

【sendDetailPage の引数内に直接記述する方法】

```

k2x.sendDetailPage(event, function(record, fieldList, tablecode, rowno) {
    ~~ 格納値を変更する処理 ~~
});

```

どちらの場合も引数の指定は同じ並びで 4 つになります。

【処理関数の考え方】

処理関数は kintone から X-point に送る書類申請・更新の情報を準備する段階で呼び出されます。処理の呼び出しは

- テーブル定義以外 . . . 1 回だけ呼び出されます
- テーブル定義内 . . . kintone レコードに格納されているテーブル行数分呼び出されます。

X-point の表明細定義の行数とは一致しません。

処理側では引数として受け取る fieldList 情報を加工し、対応するフィールド情報に“value”プロパティを追加し、上書きする値を設定します。“value”値を設定した項目は X-point に申請される書類の該当項目が処理関数で加工した値になります。

【処理関数の引数】

| 引数 | 用途 | 内容 |
|-----------|-------------------|---|
| record | 処理中の kintone レコード | kintone の event.record と同じものです |
| fieldList | 処理中の自動申請対応項目の設定情報 | テーブル以外の部分を処理する場合は項目対比情報の“fieldList”部分が渡されます。 テーブル内の処理を行う場合は“tableItem”内の“fieldList”部分が渡されます。 |
| tablecode | テーブル部のテーブルコード | テーブル以外の処理を行う場合は undefined になります。 テーブル内の行を処理する場合にテーブルコードが指定されます。 |
| rowno | テーブル部の処理行番号 | テーブル以外の処理を行う場合は undefined になります。 テーブル内の行を処理する場合には行番号が指定されます。 |

【処理関数の記述例】

例では、表定義部分で kintone からの入力力で表部分に 1 行目がある場合は 1 行目 2 列目に「あいうえお」、2 行目がある場合は 2 列目に「かきくけこ」を設定し、表以外の部分にある 'txtShinseibi'、'txtKiboubi' に年月日、'txtShinseisya' に kintone のログインユーザー名を設定するよう記述を加えます。その他の連携指定されたフィールドは kintone の入力値のままになります。

```

k2x.sendDetailPage(event, function(record, fieldList, tablecode, rowno) {
  if(tablecode){
    if (rowno == 0){
      fieldList[1].value = "あいうえお";
    }else if (rowno == 1){
      fieldList[1].value = "かきくけこ";
    }
  } else {
    var kb= new Date(event.record['txtKiboubi'].value);
    var mm=kb.getMonth()+1;    mm=(mm<10) ? ('0'+mm) : mm;
    var dd=kb.getDate();      dd=(dd<10) ? ('0'+dd) : dd;
    var post_date = kb.getFullYear() + "年" + mm + "月" + dd + "日";

    for(var n=0;n<fieldList.length;++n){
      if(fieldList[n].xname=='txtShinseibi' || fieldList[n].xname=='txtKiboubi'){
        fieldList[n].value = post_date;
      }
      if(fieldList[n].xname=='txtShinseisya'){
        fieldList[n].value = kintone.getLoginUser()['name'];
      }
    }
  }
});
  
```

→ tablecode 変数が undefined でない場合はテーブル部分を処理していると判定
→ rowno で処理中の行を判断して文字列を設定
→ 処理日で年月日の文字列を作る
→ フィールド ID を判断し日付文字列を設定
→ フィールド ID を判断しログインユーザー名を設定

| テーブル内
| テーブル外

■更新される kintone のフィールド

JavaScript テンプレート内の「this.xp_docid_fieldcd」に書類 ID を格納するフィールドコードを指定している場合、書類の登録が正しく行われた際に、kintone のレコードの書類 ID 欄に登録された書類の ID を設定します。
登録を行う際、ブラウザ画面が再表示される点に注意して下さい。

※ Callback 処理を独自に指定した場合、書類 ID の格納は作成された Callback 処理内で行って下さい。
標準で動作する Callback 処理を利用しない場合、書類 ID の格納は動作しません。

■常に特定のユーザーで提出・更新を行う場合

申請を行う API の呼び出す前にユーザーコードを特定するプロパティを設定します。

1) kintone のログイン ID を X-point のログインに用いる場合の書き方

```
var k2x = fieldtest("fieldtestjX&k8fJ0EQkULGbi094Je2R@KS#VMqr", "fieldtest");  
k2x.sendDetailPage(event);
```

2) 特定のログイン ID を X-point のログインに用いる場合の書き方

```
var k2x = fieldtest("fieldtestjX&k8fJ0EQkULGbi094Je2R@KS#VMqr", "fieldtest");  
k2x.post_proxycd = 'user001';  
k2x.sendDetailPage(event);
```

3) kintone のログイン ID と X-point のユーザーコードが一致しない、X-point にコードがない場合に 特定のユーザーコードを試用する場合の書き方

```
var k2x = fieldtest("fieldtestjX&k8fJ0EQkULGbi094Je2R@KS#VMqr", "fieldtest");  
k2x.post_proxycd = 'user001';  
k2x.sendDetailPage(event);
```

■テーブルの処理に関する制約

kintone から書類を申請する場合、テーブル部について取扱可能な行数に制約があります。

kintone アプリではテーブル部に格納できる行数に制限はありませんが、X-point は表定義を作成する際に行数を指定するため、この行数が制限になります。kintone から申請する書類にテーブルが含まれる場合、X-point で許可されている表定義行数を超えて申請することはできません。

■通常フォームの書類申請・更新を行う場合

sendDetailPage メソッドを使用します。


```
// レコード詳細表示イベント (スマートフォン)
kintone.events.on('mobile.app.record.detail.show', function (event) {
  var elButton = kintone.mobile.app.getHeaderSpaceElement();
  :
  ~~ 「X-point に申請」 ボタンを設置する処理 ~~
  :
  elButton.appendChild(bt_div);
```

```
// ボタンの間にスペースを空ける
var space = document.createElement('span');
space.innerHTML = "&nbsp;";
space.setAttribute('style', 'float:left;');
elButton.appendChild(space);
```

ボタン間の隙間作成

```
// [申請書類の削除]
var bt_div = document.createElement('div');
bt_div.setAttribute('id', 'addButton');
bt_div.setAttribute('name', 'addButton');
var bt = document.createElement('button');
bt.setAttribute('id', 'bt2');
bt.onclick = function() {
  var k2x=fieldtest("fieldtestjX&k8fJOEQkULGbi094Je2R@KS#VMqr", "fieldtest");
  k2x.deleteDoc(event);
};
bt.innerHTML = '申請書類の削除';
bt_div.appendChild(bt);
elButton.appendChild(bt_div);
```

削除ボタンの作成

```
});
```

```
})();
```

■更新される kintone のフィールド

JavaScript テンプレート内の「this.xp_docid_fieldcd」に書類 ID を格納するフィールドコードを指定している場合、書類削除の実行が正しく行われた際に、kintone のレコードに格納されている書類 ID 欄をクリア（空欄）にします。空欄にする際、ブラウザ画面が再表示される点に注意して下さい。

■書類を削除する際のユーザを特定のユーザにする方法

書類の削除を行う場合、指定するユーザーコードは削除対象となる書類を削除する権限を持っている必要があります。その為、kintone を利用するユーザーが X-point にログインアカウントを持っていない場合、もしくは kintone のログイン ID と X-point のログイン名が異なる場合には、ユーザーコードを固定する必要があります。

1) kintone のログイン ID を X-point のログインに用いる場合の書き方

```
var k2x = fieldtest("fieldtestjX&k8fJOEQkULGbi094Je2R@KS#VMqr", "fieldtest");
k2x.deleteDoc(event);
```

2) 特定のログイン ID を X-point のログインに用いる場合の書き方

```
var k2x = fieldtest("fieldtestjX&k8fJOEQkULGbi094Je2R@KS#VMqr", "fieldtest");
k2x.post_proxycd = 'user001';
k2x.deleteDoc(event);
```

4.1.4.3. 申請書類の内容を取得

X-point に提出されている書類の内容を取得します。

取得できる情報には、書類管理情報、書類項目、添付ファイル、コメント の4種類が含まれます。但し、添付ファイル、コメントは書類に添付、記入されている場合のみ含まれる項目になり、添付ファイルを実際にダウンロードする場合は別の API を利用する必要があります。

■使用例

イベント発生時に書類の内容を取得する場合

```
var k2x = fieldtest("fieldtestjX&k8fJOEQkULGbi094Je2R@KS#VMqr", "fieldtest");
var doc_json = k2x.getDocument(event);
```

書類 ID を指定して書類の内容を取得する場合

```
var k2x = fieldtest("fieldtestjX&k8fJOEQkULGbi094Je2R@KS#VMqr", "fieldtest");
var doc_json = k2x.getDocument({X-point 書類 ID});
```

■取得できる情報

| 分類 | プロパティ名 | Type 値 | |
|---------|--------------------|---|-----------|
| 書類管理情報 | 書類 ID | __ID__ | |
| | 起票者 CD | WRITERCD | |
| | 起票者名 | WRITERNAME | |
| | 起票日 | WRITEDATE | |
| | 起票者グループ CD | WRITERGROUPCD | |
| | 起票者グループ名 | WRITERGROUPNAME | |
| | 代理申請者 CD | PROXYWRITERCD | 代理書類の場合のみ |
| | 代理申請者名 | PROXYWRITERNAME | |
| | 最終承認者 CD | LASTAPRVCD | |
| | 最終承認者名 | LASTAPRVNAME | |
| | 最終承認日 | LASTAPRVDATE | |
| | 最終ステップ数 | STPCNT | WF 書類のみ |
| | 現在ステップ番号 | CURRENTSTEP | WF 書類のみ |
| | 現在バージョン番号 | CURRENTVERNO | WF 書類のみ |
| 承認状況 | STAT | WF 書類のみ 0:下書き 1:承認中 2:保留中 3:却下 4:起票者差し戻し 5: 承認者差し戻し 6:承認完了 | |
| フィールド項目 | フィールド ID (文字フィールド) | TEXTFIELD | |
| | フィールド ID (数値フィールド) | NUMBERFIELD | |
| | フィールド ID (整数フィールド) | INTEGERFIELD | |

| | | | |
|--------|---------------------|-------------|-----------------------------------|
| | フィールド ID (テキストエリア) | TEXTAREA | |
| | フィールド ID (パスワード) | PASSWORD | |
| | フィールド ID (ラジオボタン) | RADIOBUTTON | |
| | フィールド ID (チェックボックス) | CHECKBOX | |
| | フィールド ID (コンボボックス) | COMBOBOX | |
| | フィールド ID (リストボックス) | LISTBOX | |
| | フィールド ID (西暦フィールド) | YEARFIELD | |
| | フィールド ID (月フィールド) | MONTHFIELD | |
| | フィールド ID (日フィールド) | DAYFIELD | |
| | フィールド ID (曜日フィールド) | WEEKFIELD | |
| | フィールド ID (日時フィールド) | DATEFIELD | |
| | フィールド ID (隠しフィールド) | HIDDENFIELD | |
| | グループ ID (表明細定義) | SUBTABLE | 行を配列形式で管理し、フィールド項目形式で値を格納 |
| 添付ファイル | 添付ファイル | FILE | 添付ファイルがある場合に限り、配列形式でファイルアクセスキーを返す |
| コメント | コメント | COMMENT | コメントがある場合に限り、配列形式でコメント情報を返す |

■ 応答例 (実際の応答に改行等は含まれていません)

```
{
  "record": {
    "書類 ID": {
      "type": "__ID__",
      "value": "2698"
    },
    "起票者 CD": {
      "type": "WRITERCD",
      "value": "u001"
    },
    "起票者名": {
      "type": "WRITERNAME",
      "value": "勝 太郎"
    },
    "起票日": {
      "type": "WRITEDATE",
      "value": "2017/04/07 14:50:39"
    },
    "起票者グループ CD": {
      "type": "WRITERGROUPCD",
      "value": "1500"
    },
    "起票者グループ名": {
      "type": "WRITERGROUPNAME",
      "value": "システム部"
    },
    "最終承認者 CD": {
      "type": "LASTAPRVCD",
      "value": "u001"
    },
    "最終承認者名": {
      "type": "LASTAPRVNAME",
      "value": "勝 太郎"
    }
  }
}
```

```

"最終承認日": {
  "type": "LASTAPRVDATE",
  "value": "2017/04/07 14:50:41"
},
"最終ステップ数": {
  "type": "STEPCNT",
  "value": "7"
},
"現在ステップ番号": {
  "type": "CURRENTSTEP",
  "value": "2"
},
"現在バージョンNO": {
  "type": "CURRENTVERNO",
  "value": "1"
},
"承認状況": {
  "type": "STAT",
  "value": "1"
},
"txtKanricode": {
  "type": "TEXTFIELD",
  "value": "P018、ECO プロジェクト"
},
"xpoint_docid": {
  "type": "TEXTFIELD",
  "value": ""
},
"txtShinseibi": {
  "type": "TEXTFIELD",
  "value": "2017年02月15日"
},
"txtKokyakumei": {
  "type": "TEXTFIELD",
  "value": "セレブコーヒージャパン株式会社"
},
"contact_id": {
  "type": "TEXTFIELD",
  "value": ""
},
"txtKokyakucode": {
  "type": "TEXTFIELD",
  "value": "G010"
},
"txtTantou": {
  "type": "TEXTFIELD",
  "value": "勝 太郎"
},
"txtShinseisya": {
  "type": "TEXTFIELD",
  "value": "勝 太郎"
},
"kintone_id": {
  "type": "TEXTFIELD",
  "value": "4"
},
"usercode": {
  "type": "TEXTFIELD",
  "value": "u001"
},
"txtKiboubi": {
  "type": "TEXTFIELD",
  "value": "2017年02月15日"
},
"txtNo": {

```

```

    "type": "TEXTFIELD",
    "value": "KS-2017-00340"
  },
  "txtMemo": {
    "type": "TEXTAREA",
    "value": "オフィス環境改善プロジェクト事業者選定契約"
  },
  "group1": {
    "type": "SUBTABLE",
    "value": [
      {
        "rowno": 1,
        "value": {
          "txtKeiyakusyono": {
            "type": "TEXTFIELD",
            "value": "S00230"
          },
          "txtKeiyakusyomei": {
            "type": "TEXTFIELD",
            "value": "備品納入契約"
          }
        }
      },
      {
        "rowno": 2,
        "value": {
          "txtKeiyakusyono": {
            "type": "TEXTFIELD",
            "value": "S00245"
          },
          "txtKeiyakusyomei": {
            "type": "TEXTFIELD",
            "value": "取引口座開設既定"
          }
        }
      }
    ]
  },
  "添付ファイル": {
    "type": "FILE",
    "value": [
      {
        "contentType": "text/csv",
        "fileKey": "MDAwMDAwMDAwMDAwMDI2OTg6MDAwMDA%3D",
        "name": "プロジェクトマスタ_20170117T171228+0900.csv",
        "size": "684"
      },
      {
        "contentType": "text/csv",
        "fileKey": "MDAwMDAwMDAwMDAwMDI2OTg6MDAwMDE%3D",
        "name": "郵便番号マスタ_20170125T125605+0900.csv",
        "size": "4378752"
      }
    ]
  },
  "コメント": {
    "type": "COMMENT",
    "value": [
      {
        "writername": "勝 太郎",
        "up_date": "2017/04/28 16:06:35",
        "backflg": 0,
        "writercode": " u001",
        "writedate": "2017/04/28 16:06:35",
        "up_name": "勝 太郎",

```

```
        "attentionflg": 0,
        "content": "コメント①",
        "up_rcode": " u001"
    }
  ]
}
}
```

4. 1. 4. 4. 書類の添付ファイルを取得

API「getFile」では、書類内容を取得する「getDocument」で得られる添付ファイル情報キーを利用しファイルをダウンロードする URL が作成できます。

getDocument で得られる書類内容には添付ファイルに関する情報が配列形式で含まれており各ファイル情報には“filekey”がセットで返されます。API「getFile」に取得したキー情報を指定することで、添付ファイルをダウンロードするために必要な URL が生成できます。

【getDocument 応答例、添付ファイル部分】

```
{
  "record": {
    :
    "添付ファイル": {
      "type": "FILE",
      "value": [
        {
          "contentType": "text/csv",
          "fileKey": "MDAwMDAwMDAwMDAwMDAwMDI2OTg6MDAwMDA%3D",
          "name": "プロジェクトマスタ_20170117T171228+0900.csv",
          "size": "684"
        }
      ]
    },
    :
  }
}
```

■使用例

イベント発生時に書類の内容を取得する場合

```
var k2x = fieldtest("fieldtestjX&k8fJOEQkULGbi094Je2R@KS#VMqr", "fieldtest");
var json_doc = k2x.getDocument(event);
var doc_json = k2x.getFile();
if (json_doc) {
  var att_files = json_doc.record['添付ファイル'];
  for (var n=0; n < att_files.value.length ; ++n) {
    var download_url = k2x.getFile(att_files.value[n].fileKey);
    :
    取得した URL 情報を使って処理を行います
    :
  }
}
```

4.1.4.5. フォームの定義情報を取得

フォームを構成するフィールドの定義情報を取得します。管理サイトのフォーム管理で各フォームのプロパティに表示されるフィールド情報と基本的に同じになります。(プロパティはデータベースの列名で表示されますが、API はフィールド ID 名になります)

■使用例

特定ステップでボタン表示などの処理を行う場合

```
var k2x=new KeiyakuShinsa(' Contact2MsH4JLsdjEiPJWz1sad!yKWh17QNY!Bw', ' Contact2');  
var formdef_json=k2x.getFormFieldList();
```

■取得できる情報

【格納プロパティ】

| 分類 | プロパティ名 | 取得内容 |
|---------|----------|---|
| フォームコード | formcode | フォームコード |
| フォーム名 | Formname | フォーム名 |
| フィールド | field | 定義されているフィールドの配列。 フィールド ID をプロパティに持つ値が格納されています。 |

【フィールド「field」情報に含まれる各項目】

| 分類 | プロパティ名 | 取得内容 |
|----------|------------|---|
| フォームコード | sortno | フォームのページ番号 (単フォームの場合は常に1) |
| フォーム名 | seq | フォーム内でのフィールド順序。表定義内の場合は列内の順番。 |
| フィールドタイプ | fieldtype | フィールドのタイプを示す文字列 TEXTFIELD 文字フィールド NUMBERFIELD 数値フィールド INTEGERFIELD 整数フィールド TEXTAREA テキストエリア PASSWORD パスワード RADIOBUTTON ラジオボタン CHECKBOX チェックボックス COMBOBOX コンボボックス LISTBOX リストボックス YEARFIELD 西暦フィールド MONTHFIELD 月フィールド DAYFIELD 日フィールド WEEKFIELD 曜日フィールド DATEFIELD 日時フィールド HIDDENFIELD 隠しフィールド SUBTABLE 表定義 |
| 文字列長 | maxlength | 格納文字列長 (表定義の場合は指定されません) |
| 空文字列許可 | notnullflg | 空値の許可 0 : 空値 OK 2 : 空値 NG (表定義の場合は指定されません) |
| 最大行数 | arraysize | 表定義の最大行数 (表定義の場合のみ設定されます) |
| 列項目 | field | 表定義の列項目 (表定義の場合のみ設定されます) |

■応答例 (実際の応答に改行等は含まれていません)

```
{
```

```

"formcode": "KeiyakuShinsa",
"formname": "契約書社内審査",
"field": [
  {
    "txtShinseibi": {
      "sortno": "1",
      "maxlength": 4000,
      "notnullflg": 0,
      "fieldtype": "TEXTFIELD",
      "seq": 1
    }
  },
  {
    "txtShinseisya": {
      "sortno": "1",
      "maxlength": 4000,
      "notnullflg": 0,
      "fieldtype": "TEXTFIELD",
      "seq": 2
    }
  },
  {
    "txtNo": {
      "sortno": "1",
      "maxlength": 4000,
      "notnullflg": 0,
      "fieldtype": "TEXTFIELD",
      "seq": 3
    }
  },
  {
    "txtKokyakucode": {
      "sortno": "1",
      "maxlength": 4000,
      "notnullflg": 0,
      "fieldtype": "TEXTFIELD",
      "seq": 4
    }
  },
  {
    "txtKokyakumei": {
      "sortno": "1",
      "maxlength": 4000,
      "notnullflg": 0,
      "fieldtype": "TEXTFIELD",
      "seq": 5
    }
  },
  {
    "txtTantou": {
      "sortno": "1",
      "maxlength": 4000,
      "notnullflg": 0,
      "fieldtype": "TEXTFIELD",
      "seq": 6
    }
  },
  {
    "txtKiboubi": {
      "sortno": "1",
      "maxlength": 4000,
      "notnullflg": 0,
      "fieldtype": "TEXTFIELD",
      "seq": 7
    }
  }
]

```

```

    },
    {
      "txtKanricode": {
        "sortno": "1",
        "maxlength": 4000,
        "notnullflg": 0,
        "fieldtype": "TEXTFIELD",
        "seq": 8
      }
    },
    {
      "txtMemo": {
        "sortno": "1",
        "maxlength": 4000,
        "notnullflg": 0,
        "fieldtype": "TEXTAREA",
        "seq": 9
      }
    },
    {
      "group1": {
        "sortno": "1",
        "field": [
          {
            "txtKeiyakusyono": {
              "sortno": "1",
              "maxlength": 4000,
              "notnullflg": 0,
              "fieldtype": "TEXTFIELD",
              "seq": 1
            }
          },
          {
            "txtKeiyakusyomei": {
              "sortno": "1",
              "maxlength": 4000,
              "notnullflg": 0,
              "fieldtype": "TEXTFIELD",
              "seq": 2
            }
          }
        ],
        "arraysize": 3,
        "fieldtype": "SUBTABLE",
        "seq": 10
      }
    },
    {
      "kintone_id": {
        "sortno": "1",
        "maxlength": 4000,
        "notnullflg": 0,
        "fieldtype": "TEXTFIELD",
        "seq": 11
      }
    },
    {
      "usercode": {
        "sortno": "1",
        "maxlength": 4000,
        "notnullflg": 0,
        "fieldtype": "TEXTFIELD",
        "seq": 12
      }
    }
  },

```

```
{
  "contact_id": {
    "sortno": "1",
    "maxlength": 4000,
    "notnullflg": 0,
    "fieldtype": "TEXTFIELD",
    "seq": 13
  }
},
{
  "xpoint_docid": {
    "sortno": "1",
    "maxlength": 4000,
    "notnullflg": 0,
    "fieldtype": "TEXTFIELD",
    "seq": 14
  }
}
]
```

4.1.4.6. 申請書類の承認状況を取得

X-point のワークフロー書類について承認状況を取得します。取得できる情報の内容は、管理サイトの承認状況検索で「承認進捗状況」画面に表示される内容と等価です。

kintone 連携によって提出された書類のみではなく、X-point 上のみで提出・承認された書類に関する情報も取得することが出来、kintone 連携で書類閲覧が許可された API トークンが指定されていれば、X-point 上でのユーザアクセス権限とは無関係に内容を取得することが出来ます。

■使用例

特定ステップでボタン表示などの処理を行う場合

```
var k2x = fieldtest("fieldtestjX&k8fJ0EQkULGbi094Je2R@KS#VMqr", "fieldtest");
var doc_json = k2x.getDocumentStatus(event);
if (!doc_json) {
    alert('取得失敗');
} else if ( doc_json.currentstep == 3 ) {
    (特定のステップでボタンを表示するなどの処理を記述します)
}
```

kintone のログイン ID と同じユーザーコードが現在の処理ステップに含まれている（ログインユーザーがワークフロー処理できる）場合にボタン表示などの処理を行う

```
var k2x = fieldtest("fieldtestjX&k8fJ0EQkULGbi094Je2R@KS#VMqr", "fieldtest");
var doc_json = k2x.getDocumentStatus(event);
if (!doc_json) {
    alert('取得失敗');
    return;
}
var cur_usercode=kintone.getLoginUser()['code']; // ログイン中のユーザーID
var flowVersion = doc_json.flowVersions[doc_json.currentverno - 1];
var aprvusers = flowVersion.flowResults[doc_json.currentstep ].aprvusers;
for (var n=0; n < aprvusers.length ; ++n) {
    var aprvuser = aprvusers[n];
    if (cur_usercode == aprvuser.aprv.usercode) {
        (ボタンを表示するなどの処理を記述します)
        break;
    }
}
```

■取得できる情報

| 分類 | プロパティ名 | 取得内容 |
|----------|--------------|--|
| 書類 ID | docid | X-point 上での書類 ID |
| 件名 1 | title1 | 検索一覧で表示される件名 1 の内容 |
| 件名 2 | title2 | 検索一覧で表示される件名 2 の内容 |
| フォーム名 | formname | 書類のフォーム名称 |
| フォーム CD | formcode | 書類のフォームコード |
| 承認ルート名 | routename | 書類の申請で利用した承認ルート名 |
| 承認ルート CD | routecode | 書類の申請で利用した承認ルートコード |
| フォーム種別 | type | workflow (ワークフローフォーム) / standard (一般フォーム) |
| 承認状況コード | statuscode | ワークフロー書類の場合のみ処理状態を数値で設定 0:下書き 1:承認中 2:保留中 3:却下 4:差し戻され 5:差し戻し 6:承認完了 |
| 承認状況名称 | status | 承認状況コードに対応した状態を日本語文字列で設定 承認中、差し戻し、差し戻され、承認完了、下書き、却下、保留中 |
| 最大ステップ数 | stepcnt | 承認ルートの最大ステップ数 ※ 申請ステップ、回覧ステップは数に含まない ※ 空ステップを含むステップ数 |
| 現在ステップ | currentstep | 承認処理中のステップ番号 ※ 申請ステップを 0、承認ステップは 1 から順に設定 |
| 現在版番号 | currentverno | 差し戻し、承認者変更などが行われた際の版履歴数 ※ 初版の場合は 0、差し戻し後に提出、承認、下書き、保留どの操作が行われると加算される |
| 起票者 | writer | 申請者の情報 コード(usercode)、名前(username)、印影名(stampname)、日時(datetime) ※ 承認履歴が複数版の場合は最新版の申請欄に記載されるユーザー名 |
| 最終更新者 | lastaprv | 書類の操作を最後に行ったユーザーの情報 コード(usercode)、名前(username)、印影名(stampname)、日時(datetime) |
| 承認履歴 | flowVersions | 配列形式で版毎の情報が格納される |

【flowVersions の配列に設定される内容】

| 分類 | プロパティ名 | 取得内容 |
|--------|-------------|----------------------|
| ステップ情報 | flowResults | 配列形式で処理ステップの内容、状態を格納 |

※ 配列の 0 番目が第 1 版になります。

※ 現在版番号 (currentverno) を併用すると現在処理中の版情報が取得できます。

【flowResults の配列に設定される内容】

| 分類 | プロパティ名 | 取得内容 |
|----------|------------|---|
| ステップ番号 | stepno | 申請ステップが0、承認ステップ番号を1から順に設定 |
| ステップ名称 | steptitle | 承認ステップに表示される名称 |
| 承認者情報 | aprusers | ステップに指定されている処理者（申請者、承認者、回覧者）の配列 |
| 処理条件 | cond | AND, OR, NUM のタイプを文字列で設定 ※ 申請・回覧ステップではセットしない |
| 多数決処理条件数 | cond_num | condに“NUM”が指定された場合、承認の成立に必要な数 ※ 申請・回覧ステップではセットしない |
| 管理者スキップ | adminskip | 管理者サイトでスキップ処理された事を示すフラグ（0：通常、1：管理者） ※ 申請・回覧ステップではセットしない |
| スキップフラグ | skip | 空ステップを示すフラグ（0：通常、1：スキップ） ※ 申請・回覧ステップではセットしない |
| 差し戻し先 | backstepno | 差し戻し実行時の差し戻し先ステップ番号 ※ 申請・回覧ステップではセットしない ※ 差し戻しが無い場合は常に0 |

- ※ 配列のサイズは回覧ステップが無い場合は「最大ステップ数（stepcnt）+1」になります。
回覧ステップがある場合は「最大ステップ数（stepcnt）+2」です。
- ※ 配列の0番目が申請、1番目以降が承認ステップになります。
- ※ 回覧ステップの情報は配列の最後になります。
- ※ ステップで差し戻しが行われたか否かの判断は“差し戻し先”の設定値ではなく、承認者情報（aprusers）の中に差し戻しを行ったユーザーが存在するか調べてください。

【aprusers の配列に設定される内容】

| 分類 | プロパティ名 | 取得内容 |
|---------|------------|--|
| 処理者情報 | aprv | 承認者、回覧者の情報 コード(usercode)、名前(username)、部署コード（groupcd）、 部署名（groupname）、役職コード（partcd）、役職名（partname）、 印影名（stampname）、日時（datetime） |
| ステップ状態 | statuscode | 処理者の状態 0:未処理/未読, 1:承認済/既読, 2:却下済, 3:差し戻し, 4:保留中 ※申請ステップの場合は常に0が設定されます |
| ステップ状態名 | status | ステップ状態を文字列で設定 |

■ 応答例（実際の応答に改行等は含まれていません）

```
{
  "docid": 2698,
  "title1": "KS-2017-00340/G010/セレブコーヒージャパン株式会社",
  "title2": "勝 太郎/4",
  "formname": "契約書社内審査",
  "formcode": "KeiyakuShinsa",
  "routename": "契約書社内審査",
  "routecode": "KeiyakuShinsa",
  "type": "workflow",
  "statuscode": 1,
  "status": "承認中",
  "stepcnt": 7,
  "currentstep": 2,
  "currentverno": 1,
  "writer": {
    "usercode": "u001",
    "username": "勝 太郎",
    "stampname": "勝",
    "datetime": "2017-04-07T05:50:00.000Z"
  },
  "lastaprv": {
    "usercode": "u001",
    "username": "勝 太郎",
    "stampname": "",
    "datetime": "2017-04-07T05:50:00.000Z"
  },
  "flowVersions": [
    {
      "flowResults": [
        {
          "stepno": 0,
          "steptitle": "申請",
          "aprvusers": [
            {
              "aprv": {
                "usercode": "u001",
                "username": "勝 太郎",
                "stampname": "勝",
                "datetime": "2017-04-07T05:50:00.000Z",
                "groupcd": "1500",
                "groupname": "システム部",
                "partcd": "060",
                "partname": "一般"
              },
              "statuscode": 0,
              "status": "申請"
            }
          ]
        },
        {
          "stepno": 1,
          "steptitle": "承認",
          "aprvusers": [],
          "cond": "AND",
          "cond_num": 0,
          "adminskip": 0,
          "skip": 1,
          "backstepno": 0
        },
        {
          "stepno": 2,
          "steptitle": "確認",

```

```

    "aprvusers": [
      {
        "aprv": {
          "usercode": "u012",
          "username": "鈴木 正二",
          "stampname": "鈴木",
          "datetime": "",
          "groupcd": "1500",
          "groupname": "システム部",
          "partcd": "030",
          "partname": "部長"
        },
        "statuscode": 0,
        "status": "未処理"
      },
      {
        "aprv": {
          "usercode": "u009",
          "username": "宮沢 謙一",
          "stampname": "宮沢",
          "datetime": "",
          "groupcd": "1500",
          "groupname": "システム部",
          "partcd": "040",
          "partname": "課長"
        },
        "statuscode": 0,
        "status": "未処理"
      }
    ],
    "cond": "NUM",
    "cond_num": 1,
    "adminskip": 0,
    "skip": 0,
    "backstepno": 0
  },
  {
    "stepno": 3,
    "steptitle": "法務受付",
    "aprvusers": [
      {
        "aprv": {
          "usercode": "u005",
          "username": "坂本 一馬",
          "stampname": "坂本",
          "datetime": "",
          "groupcd": "1420",
          "groupname": "総務課",
          "partcd": "060",
          "partname": "一般"
        },
        "statuscode": 0,
        "status": "未処理"
      }
    ],
    "cond": "OR",
    "cond_num": 0,
    "adminskip": 0,
    "skip": 0,
    "backstepno": 0
  },
  {
    "stepno": 4,
    "steptitle": "法務確認",
    "aprvusers": [

```

```

    {
      "aprv": {
        "usercode": "u006",
        "username": "夏目 一刻",
        "stampname": "夏目(一)",
        "datetime": "",
        "groupcd": "1420",
        "groupname": "総務課",
        "partcd": "040",
        "partname": "課長"
      },
      "statuscode": 0,
      "status": "未処理"
    }
  ],
  "cond": "OR",
  "cond_num": 0,
  "adminskip": 0,
  "skip": 0,
  "backstepno": 0
},
{
  "stepno": 5,
  "steptitle": "法務承認",
  "aprvusers": [
    {
      "aprv": {
        "usercode": "u008",
        "username": "太田 治",
        "stampname": "太田",
        "datetime": "",
        "groupcd": "1400",
        "groupname": "管理部",
        "partcd": "030",
        "partname": "部長"
      },
      "statuscode": 0,
      "status": "未処理"
    }
  ],
  "cond": "OR",
  "cond_num": 0,
  "adminskip": 0,
  "skip": 0,
  "backstepno": 0
},
{
  "stepno": 6,
  "steptitle": "結果確認",
  "aprvusers": [
    {
      "aprv": {
        "usercode": "u001",
        "username": "勝 太郎",
        "stampname": "勝",
        "datetime": "",
        "groupcd": "1500",
        "groupname": "システム部",
        "partcd": "060",
        "partname": "一般"
      },
      "statuscode": 0,
      "status": "未処理"
    }
  ],

```

```

"cond": "AND",
"cond_num": 0,
"adminskip": 0,
"skip": 0,
"backstepno": 0
},
{
"stepno": 7,
"steptitle": "部門確認",
"aprusers": [
{
"apr": {
"usercode": "u012",
"username": "鈴木 正二",
"stampname": "鈴木",
"datetime": "",
"groupcd": "1500",
"groupname": "システム部",
"partcd": "030",
"partname": "部長"
},
"statuscode": 0,
"status": "未処理"
},
{
"apr": {
"usercode": "u009",
"username": "宮沢 謙一",
"stampname": "宮沢",
"datetime": "",
"groupcd": "1500",
"groupname": "システム部",
"partcd": "040",
"partname": "課長"
},
"statuscode": 0,
"status": "未処理"
}
],
"cond": "OR",
"cond_num": 0,
"adminskip": 0,
"skip": 0,
"backstepno": 0
},
{
"stepno": 8,
"steptitle": "回覧",
"aprusers": [
{
"apr": {
"usercode": "u003",
"username": "織田 信介",
"stampname": "織田",
"datetime": "",
"groupcd": "1300",
"groupname": "営業部",
"partcd": "030",
"partname": "部長"
},
"statuscode": 0,
"status": "未読"
},
{
"apr": {

```

```

        "usercode": "u002",
        "username": "木戸 武史",
        "stampname": "木戸",
        "datetime": "",
        "groupcd": "1310",
        "groupname": "営業 1 課",
        "partcd": "040",
        "partname": "課長"
    },
    "statuscode": 0,
    "status": "未読"
},
{
    "apr": {
        "usercode": "u004",
        "username": "西本 武則",
        "stampname": "西本",
        "datetime": "",
        "groupcd": "1310",
        "groupname": "営業 1 課",
        "partcd": "050",
        "partname": "主任"
    },
    "statuscode": 0,
    "status": "未読"
},
{
    "apr": {
        "usercode": "u001",
        "username": "勝 太郎",
        "stampname": "勝",
        "datetime": "",
        "groupcd": "1310",
        "groupname": "営業 1 課",
        "partcd": "060",
        "partname": "一般"
    },
    "statuscode": 0,
    "status": "未読"
},
{
    "apr": {
        "usercode": "u015",
        "username": "村井 透",
        "stampname": "村井",
        "datetime": "",
        "groupcd": "1320",
        "groupname": "営業 2 課",
        "partcd": "040",
        "partname": "課長"
    },
    "statuscode": 0,
    "status": "未読"
},
{
    "apr": {
        "usercode": "u013",
        "username": "濱崎 伝朗",
        "stampname": "浜崎",
        "datetime": "",
        "groupcd": "1320",
        "groupname": "営業 2 課",
        "partcd": "050",
        "partname": "主任"
    },

```

```
    "statuscode": 0,
    "status": "未読"
  },
  {
    "aprv": {
      "usercode": "u017",
      "username": "佐伯 千秋",
      "stampname": "佐伯(千)",
      "datetime": "",
      "groupcd": "1320",
      "groupname": "営業 2 課",
      "partcd": "060",
      "partname": "一般"
    },
    "statuscode": 0,
    "status": "未読"
  },
  {
    "aprv": {
      "usercode": "u014",
      "username": "澤田 由美江",
      "stampname": "澤田",
      "datetime": "",
      "groupcd": "1330",
      "groupname": "営業 3 課",
      "partcd": "040",
      "partname": "課長"
    },
    "statuscode": 0,
    "status": "未読"
  },
  {
    "aprv": {
      "usercode": "u018",
      "username": "濱崎 敏子",
      "stampname": "濱崎",
      "datetime": "",
      "groupcd": "1330",
      "groupname": "営業 3 課",
      "partcd": "060",
      "partname": "一般"
    },
    "statuscode": 0,
    "status": "未読"
  }
]
]
]
]
]
```

4.1.4.7. 承認待ち件数を取得

X-point の承認件数を取得します。ユーザー画面 TOP の承認ガジェットでフォームグループ、フォーム名を指定した際に表示される件数、一覧表示と同じものが取得可能です。取得できる内容は API の実行を行った kintone 利用中のユーザー、あるいは指定されたユーザーが X-point 上でログインした際の承認件数になります。

■使用例

```
var k2x = fieldtest("fieldtestjX&k8fJOEQkULGbi094Je2R@KS#VMqr", "fieldtest");
var doc_json = k2x.getWorkflowCount(フォームグループ);
if (doc_json.wait == null) {
    alert('取得失敗');
    return;
}
if ( doc_json.wait > 0 ) {
    alert('承認待ちの書類が'+ doc_json.wait + '件あります');
} else {
    alert('承認待ちの書類はありません');
}
```

■取得できる情報

連携 API 「getWorkflowCount」を実行した際に kintone にログインしていたユーザーのログイン名、もしくは連携インスタンスのプロパティ (post_usercd) に指定されたユーザーに係わる書類の情報が所得できます。

| 分類 | プロパティ名 | 取得内容 |
|-----------|------------|---|
| 承認待ち | wait | 承認待ち件数 |
| 下書き | draft | 下書き件数 |
| 差し戻し | doback | 承認書類差し戻し件数 |
| 差し戻され | beback | 申請書類差し戻し件数 |
| 保留 | suspend | 保留件数 |
| 承認中 (申請) | apply | 申請した書類の承認中件数 |
| 承認中 (承認) | aprvt | 承認した書類の承認中件数 |
| 承認完了 (申請) | comp_apply | 申請した書類の承認完了件数 |
| 承認完了 (承認) | comp_aprv | 承認した書類の承認完了件数 |
| 却下 | reject | 申請し却下された書類の件数 |
| 回覧 | Circ | 回覧既読処理されていない書類の件数 |
| 承認待ち書類 | wait_list | 承認待ち書類を最大 5 件取得できます。 書類情報には、書類 ID(docid)、件名 1 (title)、フォーム名(name)、起票者名(writername)、起票日時(writedate)の 5 項目が設定される |

■ 応答例（実際の応答に改行等は含まれていません）

```
{
  "suspend": 0,
  "wait": 11,
  "beback": 0,
  "comp_aprv": 0,
  "apply": 0,
  "reject": 0,
  "draft": 0,
  "circ": 4,
  "comp_aply": 0,
  "doback": 0,
  "aprv": 0,
  "wait_list": [
    {
      "writername": "勝 太郎",
      "docid": 2552,
      "name": "契約書社内審査",
      "writedate": "2017-02-24T07:46:00.000Z",
      "title": "KS-2017-00254/G001/株式会社太陽商事"
    },
    {
      "writername": "勝 太郎",
      "docid": 2546,
      "name": "契約書社内審査",
      "writedate": "2017-02-24T07:44:00.000Z",
      "title": "KS-2017-00340/G010/セレブコーヒージャパン株式会社"
    },
    {
      "writername": "勝 太郎",
      "docid": 2545,
      "name": "契約書社内審査",
      "writedate": "2017-02-24T07:44:00.000Z",
      "title": "KS-2017-00341/G017/株式会社紙沢山商事"
    },
    {
      "writername": "勝 太郎",
      "docid": 2544,
      "name": "契約書社内審査",
      "writedate": "2017-02-24T07:44:00.000Z",
      "title": "KS-2017-00333/G002/富士山エレクトロニクス株式会社"
    },
    {
      "writername": "勝 太郎",
      "docid": 2543,
      "name": "契約書社内審査",
      "writedate": "2017-02-24T07:44:00.000Z",
      "title": "KS-2017-00327/G018/株式会社金鯨通運 "
    }
  ]
}
```

4.1.4.8. 一括申請

kintone アプリのレコード一覧画面上部に【X-point に一括申請】ボタンを用意し書類申請用の API (sendListPage) を使い、ボタンクリックで一覧に表示されているレコードを X-point に一括書類申請します。

■画面のイメージ

【PC 版 kintone】



【スマートフォン版 kintone】



ボタンを追加し、テンプレートの
“sendListPage” メソッドで申請します。

アプリに組み込む最もシンプルな JavaScript は次のようなイメージになります。

サンプルの **赤字** で囲まれた部分が連携テンプレートの JavaScript クラスを実行する部分になります。その他の部分は kintone の開発手法に従います。画面イメージ例以外の組み込みを行う場合は、kintone developer site 等を参考に作業を行って下さい。

なお、テンプレート JavaScript を利用して X-point に書類申請・更新を行った場合、X-point から取得した書類 ID を kintone のレコードに格納を行う過程で必ずブラウザ画面のリフレッシュが実行されます。

```
(function () {
  "use strict";

  // レコード一覧表示イベント (PC)
  kintone.events.on('app.record.index.show', function (event) {
    if (document.getElementById('my_post_button') == null) {
      var myIndexButton = document.createElement('button');
      myIndexButton.id = 'my_post_button';
      myIndexButton.innerHTML = 'X-point に一括申請';
      myIndexButton.onclick = function () {
        var k2x=fieldtest("fieldtestjX&k8fJ0EQkULGbi094Je2R@KS#VMqr","fieldtest");
        k2x.sendListPage(event);
      }
      // メニューの右側の空白部分にボタンを設置
      kintone.app.getHeaderMenuSpaceElement().appendChild(myIndexButton);
    }
  });
})();
```

PC ブラウザ用

```

(function () {
    "use strict";

    // レコード一覧表示イベント (スマートフォン)
    kintone.events.on('mobile.app.record.index.show', function (event) {
        if (document.getElementById('addButton') == null) {
            var elButton = kintone.mobile.app.getHeaderSpaceElement();
            var bt_div = document.createElement('div');
            bt_div.setAttribute('id', 'addButton');
            bt_div.setAttribute('name', 'addButton');
            var bt = document.createElement('button');           スマートフォン用
            bt.setAttribute('id', 'bt1');
            bt.onclick = function() {
                var k2x=fieldtest("fieldtestjX&k8fJOEQkULGbi094Je2R@KS#VMqr", "fieldtest");
                k2x.sendListPage(event);
            };
            bt.innerHTML = 'X-pointに一括申請';
            bt_div.appendChild(bt);
            elButton.appendChild(bt_div);
        }
    });
})();

```

■ Callback 処理を指定した際に得られる応答例 (実際の応答に改行等は含まれていません)

sendDetailPage メソッドを event のみの指定で呼び出した場合に得られる戻り値は true / false のみになりますが、Callback 処理用の function を指定した場合に X-point より得られる応答を説明します。

応答は処理対象となった書類毎に状態が返されます。応答が複数件になる点を除き 1 件毎の申請である sendDetailPage の場合と同じ内容になります。

【応答例】

提出、更新、更新不可、指定書類がない場合が混在した場合の例。

```

{
  "docs": [
    {"kintone_rec_no": "12", "docid": -1, "message_type": 1,
     "message": "指定された書類 ID[2878]は登録されていません。"},
    {"kintone_rec_no": "11", "docid": -1, "message_type": 1,
     "message": "指定された書類 ID[2858]はワークフロー処理中のため更新できません。"},
    {"kintone_rec_no": "9", "docid": -1, "message_type": 1,
     "message": "指定された書類 ID[2850]はワークフロー処理中のため更新できません。"},
    {"kintone_rec_no": "8", "docid": -1, "message_type": 1,
     "message": "指定された書類 ID[2849]はワークフロー処理中のため更新できません。"},
    {"kintone_rec_no": "7", "docid": -1, "message_type": 1,
     "message": "指定された書類 ID[2848]はワークフロー処理中のため更新できません。"},
    {"kintone_rec_no": "6", "docid": 2879, "message_type": 3, "message": "書類が提出されました (ID = 2879)"},
    {"kintone_rec_no": "5", "docid": 2844, "message_type": 3, "message": "書類が更新されました (ID = 2844)"},
    {"kintone_rec_no": "4", "docid": 2880, "message_type": 3, "message": "書類が提出されました (ID = 2880)"},
    {"kintone_rec_no": "3", "docid": 2881, "message_type": 3, "message": "書類が提出されました (ID = 2881)"},
    {"kintone_rec_no": "2", "docid": 2882, "message_type": 3, "message": "書類が提出されました (ID = 2882)"},
    {"kintone_rec_no": "1", "docid": 2883, "message_type": 3, "message": "書類が提出されました (ID = 2883)"}
  ]
}

```

■取得できる情報

| 分類 | プロパティ名 | 取得内容 |
|-----------------|----------------|--|
| 処理結果の配列 | docs | kintone レコード ID、書類 ID、応答種別、応答内容をセットにした配列。 sendDetailPage の場合は処理件数に応じた数を格納 |
| kintone レコード ID | kintone_rec_no | エラー原因に係わる kintone のレコード ID kintone レコードと関係ない場合は -1 |
| X-point 書類 ID | docid | X-point 上での書類 ID |
| 応答種別 | message_type | -1, 0 : 正常応答 1: エラーメッセージ 2: 警告メッセージ 3: インフォメーション |
| 応答内容 | message | メッセージテキスト |

4. 1. 4. 9. 申請書類を表示する

JavaScript テンプレートの「xp_docid_fieldcd」プロパティに指定したフィールドコードを参照し kintone アプリのレコードに格納されている対応する書類 ID を取得して X-point の書類閲覧ビューを表示します。

書類を表示する際、API「openDoc」に指定した X-point ユーザー（未指定の場合は、kintone ログイン名）での認証を求められ、認証に成功すると書類が表示されます。直前に同じユーザー ID を利用した書類閲覧を行っている場合は同一ユーザーであるとみなされ認証することなく書類が表示されます。

■画面のイメージ

【PC 版 kintone】



【スマートフォン版 kintone】



ボタンを追加し、テンプレートの
"openDoc"メソッドで表示します。

※ 書類が格納されている X-point に対し同一ブラウザでのセッションにおいてログイン済みの場合はログイン認証が省略されますが、ログインしていたユーザー ID と kintone から表示を行う際の指定ログイン ID が異なる場合、既存のログインセッションが切断されます。切断されたログインセッションは書類閲覧後に復元されません。

アプリに組み込む最もシンプルな JavaScript は次のようなイメージになります。

サンプルは【書類申請】、【書類表示】の2ボタンを用意するパターンになり、**赤枠**で囲まれた部分が削除を実行する部分です。

```
(function () {
  "use strict";

  // レコード詳細表示イベント (PC)
  kintone.events.on('app.record.detail.show', function (event) {
    // メニュー右側の空白部分にボタンを設置
    var myBtn = document.createElement('button');
    myBtn.id = 'my_post_button';
    myBtn.innerHTML = 'X-point に申請';
    myBtn.onclick = function () {
      var k2x=fieldtest("fieldtestjX&k8fJ0EQkULGbi094Je2R@KS#VMqr", "fieldtest");
      k2x.sendDetailPage(event);
    }
    kintone.app.record.getHeaderMenuSpaceElement().appendChild(myBtn);

    var space = document.createElement('span');
    space.innerHTML = "&nbsp;&nbsp;&nbsp;";
    kintone.app.record.getHeaderMenuSpaceElement().appendChild(space);

    var myBtn = document.createElement('button');
    myBtn.id = 'my_opendoc_button';
    myBtn.innerHTML = '書類の表示';
    myBtn.onclick = function () {
      var k2x=fieldtest("fieldtestjX&k8fJ0EQkULGbi094Je2R@KS#VMqr", "fieldtest");
      k2x.openDoc(event);
    }
    kintone.app.record.getHeaderMenuSpaceElement().appendChild(myBtn);

  });
})();
```

ボタン間の隙間作成

表示ボタンの作成

PC ブラウザ用

var k2x=fieldtest("fieldtestjX&k8fJ0EQkULGbi094Je2R@KS#VMqr", "fieldtest");
k2x.openDoc(event);

```

(function () {
    "use strict";

    // レコード詳細表示イベント (スマートフォン)
    kintone.events.on('mobile.app.record.detail.show', function (event) {
        var elButton = kintone.mobile.app.getHeaderSpaceElement();
        var bt_div = document.createElement('div');
        bt_div.setAttribute('id', 'addButton');
        bt_div.setAttribute('name', 'addButton');
        bt_div.setAttribute('style', 'float:left;');
        var bt = document.createElement('button');
        bt.setAttribute('id', 'bt1');
        bt.onclick = function() {
            var k2x=fieldtest("fieldtestjX&k8fJ0EQkULGbi094Je2R@KS#VMqr", "fieldtest");
            k2x.sendDetailPage(event);
        };
        bt.innerHTML = '書類の表示';
        bt_div.appendChild(bt);
        elButton.appendChild(bt_div);

        // ボタンの間にスペースを空ける
        var space = document.createElement('span');
        space.innerHTML = "&nbsp;";
        space.setAttribute('style', 'float:left;');
        elButton.appendChild(space);

        // [書類の表示]
        var bt_div = document.createElement('div');
        bt_div.setAttribute('id', 'addButton');
        bt_div.setAttribute('name', 'addButton');
        var bt = document.createElement('button');
        bt.setAttribute('id', 'bt3');
        bt.onclick = function() {
            var k2x=fieldtest("fieldtestjX&k8fJ0EQkULGbi094Je2R@KS#VMqr", "fieldtest");
            k2x.openDoc(event);
        };
        bt.innerHTML = '申請書類の削除';
        bt_div.appendChild(bt);
        elButton.appendChild(bt_div);
    });
})();

```

ボタン間の隙間作成

表示ボタンの作成

スマートフォン用

4.1.4.10. 承認状況を表示する

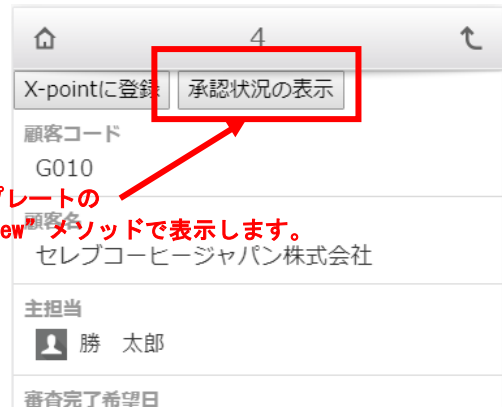
JavaScript テンプレートの「xp_docid_fieldcd」プロパティに指定したフィールドコードを参照し kintone アプリのレコードに格納されている対応する書類 ID を取得して X-point の承認状況の確認画面を表示します。

■画面のイメージ

【PC 版 kintone】



【スマートフォン版 kintone】



ボタンを追加し、テンプレートの
“getDocumentStatusView”メソッドで表示します。

アプリに組み込む最もシンプルな JavaScript は次のようなイメージになります。

サンプルは【書類申請】、【状況画面】の2ボタンを用意するパターンになり、**赤枠**で囲まれた部分が削除を実行する部分です。

```
(function () {
  "use strict";

  // レコード詳細表示イベント (PC)
  kintone.events.on('app.record.detail.show', function (event) {
    // メニュー右側の空白部分にボタンを設置
    var myBtn = document.createElement('button');
    myBtn.id = 'my_post_button';
    myBtn.innerHTML = 'X-point に申請';
    myBtn.onclick = function () {
      var k2x=fieldtest("fieldtestjX&k8fJOEQkULGbi094Je2R@KS#VMqr", "fieldtest");
      k2x.sendDetailPage(event);
    }
    kintone.app.record.getHeaderMenuSpaceElement().appendChild(myBtn);

    var space = document.createElement('span');
    space.innerHTML = "&nbsp;&nbsp;&nbsp;";
    kintone.app.record.getHeaderMenuSpaceElement().appendChild(space);

    var myBtn = document.createElement('button');
    myBtn.id = 'my_opendoc_button';
    myBtn.innerHTML = '状況画面';
    myBtn.onclick = function () {
      var k2x=fieldtest("fieldtestjX&k8fJOEQkULGbi094Je2R@KS#VMqr", "fieldtest");
      k2x.getDocumentStatusView(event);
    }
    kintone.app.record.getHeaderMenuSpaceElement().appendChild(myBtn);
  });
})();
```

ボタン間の隙間作成

表示ボタンの作成

PC ブラウザ用

```

(function () {
    "use strict";

    // レコード詳細表示イベント (スマートフォン)
    kintone.events.on('mobile.app.record.detail.show', function (event) {
        var elButton = kintone.mobile.app.getHeaderSpaceElement();
        var bt_div = document.createElement('div');
        bt_div.setAttribute('id', 'addButton');
        bt_div.setAttribute('name', 'addButton');
        bt_div.setAttribute('style', 'float:left;');
        var bt = document.createElement('button');
        bt.setAttribute('id', 'bt1');
        bt.onclick = function () {
            var k2x=fieldtest("fieldtestjX&k8fJOEQkULGbi094Je2R@KS#VMqr", "fieldtest");
            k2x.sendDetailPage(event);
        };
        bt.innerHTML = 'X-point に申請';
        bt_div.appendChild(bt);
        elButton.appendChild(bt_div);

        // ボタンの間にスペースを空ける
        var space = document.createElement('span');
        space.innerHTML = "&nbsp;";
        space.setAttribute('style', 'float:left;');
        elButton.appendChild(space);

        // [書類の承認状況表示]
        var bt_div = document.createElement('div');
        bt_div.setAttribute('id', 'addButton');
        bt_div.setAttribute('name', 'addButton');
        var bt = document.createElement('button');
        bt.setAttribute('id', 'bt3');
        bt.onclick = function () {
            var k2x=fieldtest("fieldtestjX&k8fJOEQkULGbi094Je2R@KS#VMqr", "fieldtest");
            k2x.getDocumentStatusView(event);
        };
        bt.innerHTML = '状況画面';
        bt_div.appendChild(bt);
        elButton.appendChild(bt_div);
    });
})();

```

ボタン間の隙間作成

表示ボタンの作成

スマートフォン用

■実行例 (画面イメージ)

| 承認進捗状況 (書類No.26180) | | | |
|---|---|--|---------------------|
| 件名1 | 営業部 | | |
| 件名2 | 営業太郎 | | |
| フォーム | A368出張申請書 | 承認ルート | 規定のルート |
| 承認状況 | 承認中 | 申請日 | 2020/10/30 14:14:45 |
| 第1版 | | | |
| 申請  申請 | 承認  STEP 1 | | |
| 管理者 未所属 20/10/30 14:14:45 | AND (2) | 佐伯 宗一 (未処理) 株式会社えいとれつ商事 ※ 役員 木々 和明 (未処理) 株式会社えいとれつ商事 ※ 社長 | |

！注意事項

承認状況の確認画面の API では、kintone から送信されたユーザ情報(ログイン ID)が X-point に存在するかチェックが行われます。“指定ユーザ[[ログイン ID]]は X-point に登録されていません” というエラーが表示される場合は、API 実行時のユーザーをご確認ください。

4. 2. X-point の連携 API を使い申請処理を作成

X-point が生成する連携テンプレート JavaScript を利用せず、連携用 API を直接利用して X-point への書類申請を行います。書類の提出を行う際、入力されている各種情報を加工（項目の連結、分割など）する場合で、連携テンプレートで用意されている Callback 関数では処理が出来ないような場合に利用する方法です。

※ X-point v2.4.02 で導入された連携方法になります。v2.5 以降は連携テンプレートの利用を推奨します。

4. 2. 1. アプリに【申請】ボタンを追加

アプリのレコード詳細画面に【X-point に申請】ボタンを配置しボタンクリックで申請処理が実行されるようにします。ボタン追加は JavaScript で行います。JavaScript はローカルファイルで作成します。

【kintone-sample.js】（ボタン作成まで、ボタンクリックで動作する kintone_to_xpoint は本項では作成しません）

```
(function () {
  "use strict";

  kintone.events.on('app.record.detail.show', function (event) { // レコード詳細表示イベント(PC)
    var btn = document.createElement('button');
    btn.id = 'btn_xp_doEntry';
    btn.innerHTML = 'X-point に申請';
    btn.onclick = function () {
      kintone_to_xpoint(event);
    }
    kintone.app.record.getHeaderMenuSpaceElement().appendChild(btn);
  });

  kintone.events.on('mobile.app.record.detail.show', function (event) { // レコード一覧表示イベント(モバイル)
    var bt_div = document.createElement('div');
    bt_div.setAttribute('id', 'addButton');
    bt_div.setAttribute('name', 'addButton');
    var bt = document.createElement('button');
    bt.setAttribute('id', 'btn_xp_doEntry');
    bt.onclick = function () {
      kintone_to_xpoint(event);
    };
    bt.innerHTML = 'X-point に申請';
    bt_div.appendChild(bt);
    kintone.mobile.app.getHeaderSpaceElement().appendChild(bt_div);
  });
})();
```

【参考】

第 2 回 レコード一覧画面にボタンを置いてみよう！

<https://cybozudev.zendesk.com/hc/ja/articles/201767270>

JavaScript を記述したファイルができあがったら、kintone の「アプリの設定を変更」→「JavaScript / CSS でカスタマイズ」から「PC 用の JavaScript ファイル」、「スマートフォン用の JavaScript ファイル」それぞれにアップロードします。

■ 「アップロードして追加」



- ※ PC 用、スマートフォン用で異なる JavaScript を作成する場合は、個別のファイルをアップロードしてください。共通で動作する場合は同じファイルをアップロードします。（共通で利用できない場合など）
- ※ アップロードする JavaScript ファイルは必ず「UTF-8（BOM 無し）」、拡張子「js」で作成します。

アップロードが完了し JavaScript の実行エラーが発生しない場合、レコード詳細画面に【X-point に申請】ボタンが表示されます。

PC ブラウザ（IE11 の場合）



モバイルブラウザ（iPhone 6 の場合）



- ※ レコード詳細画面へのボタン追加は比較的単純なプログラムで行います。正しく表示されない、JavaScript エラーが発生するなどの場合は、作成したプログラムを見直してください。
- ※ JavaScript の動作確認では各ブラウザの持つ開発ツールを利用するなどしてください。

4.2.2. X-point に書類データを転送

アプリに【申請】ボタンを追加した際に未作成の「kintone_to_xpoint」を追加し X-point への申請を行います。
処理を追加した後は、JavaScript ファイルをアップロードし直します。

【kintone-smample.js の後半】

```
kintone_to_xpoint=function(event) {
  var domCd      = "trial";           // 申請を行う X-point のドメインコード
  var fqdn       = "xpoint.atledcloud.jp"; // X-point が動作するサーバーの FQDN
  var formCd     = "KeiyakuShinsa";    // 申請を行うフォームのフォームコード
  var routecd   = "KeiyakuShinsa";    // 申請を行う承認ルートのコード
  var api_token  = "KeiyakuShinsaM!!RyeHRFYnfBI9Ftrxn01HLzd";
  var options    = { "proxyusercd": "u001" }; // X-point にログイン名と一致するユーザーCD が無い場合に使う CD

  var now = new Date();                // 申請日を「年月日」形式で作成する
  var mm=now.getMonth()+1;
  mm=(mm<10) ? ('0'+mm) : mm;
  var dd=now.getDate();
  dd=(dd<10) ? ('0'+dd) : dd;
  var post_date = now.getFullYear() + "年" + mm + "月" + dd + "日";
  var loginname=kintone.getLoginUser()['name']; // 申請者名を kintone から得る

  var record = event.record;
  var appId  = event.appId;
  var k2x = new kintone2xpoint(domCd, fqdn, formCd, routecd, api_token,options); // 送信インスタンス作成
  var body_json = k2x.create_recordset(record,appId);

  // 表以外の部分
  k2x.add_elements(body_json, record, [
    // X-point 側      , kintone 側
    { "xname": "txtShinseibi", "value": post_date },
    { "xname": "txtShinseisya", "value": loginname },
    { "xname": "txtKokyakucode", "field": "txtKokyakucode" },
    { "xname": "txtKokyakumei", "field": "txtKokyakumei" },
    { "xname": "txtTantou", "field": "txtTantou" },
    { "xname": "txtKiboubi", "field": "txtKiboubi" },
    { "xname": "txtKanricode", "field": "txtKanricode" },
    { "xname": "txtMemo", "field": "txtMemo" }
  ]);

  // 表部分を指定
  var tbl_group1 = k2x.create_table_porperty(body_json, "group1");
  k2x.add_table_elements(tbl_group1, record['group1'], [
    { "xname": "txtKeiyakusyono", "field": "txtKeiyakusyono" },
    { "xname": "txtKeiyakusyomei", "field": "txtKeiyakusyomei" } ]]);

  k2x.send(body_json); // 書類を申請する
};
```

send_kintone メソッドを利用する方法もあります

申請処理の主要部分は送信データの作成と送信から成り立っています。

送信データの作成部分から送信の内容を順に説明します。

■初期設定

```
var domCd      = "trial";           // 申請を行う X-point のドメインコード
var fqdn      = "xpoint.atledcloud.jp"; // X-point が動作するサーバーの FQDN
var formcd    = "KeiyakuShinsa";    // 申請を行うフォームのフォームコード
var routecd   = "KeiyakuShinsa";    // 申請を行う承認ルートのコード
var api_token = "KeiyakuShinsaM!!RyeHRFYnfB19Ftrxn01HLzd";
var options   = { "proxyusercd": "u001" }; // X-point にログイン名と一致するユーザーCD が無い場合に使う CD
```

連携パラメータの設定

```
var now = new Date(); // 申請日を「年月日」形式で作成する
var mm=now.getMonth()+1;
    mm=(mm<10) ? ('0'+mm) : mm;
var dd=now.getDate();
    dd=(dd<10) ? ('0'+dd) : dd;
var post_date = now.getFullYear() + "年" + mm + "月" + dd + "日";
var loginname=kintone.getLoginUser()['name']; // 申請者名を kintone から得る
```

申請日、申請者に指定する
内容を JavaScript で作成

```
var record = event.record;
var appId  = event.appId;
var k2x = new kintone2xpoint(domCd, fqdn, formcd, routecd, api_token, options); // 送信インスタンス作成
var body_json = k2x.create_recordset(record, appId);
```

連携インスタンスの準備

連携先を指定するパラメータを変数にセットし準備します。予め変数で接続先に関する情報を用意することでテスト環境から運用環境への変更などの作業が容易になります。

自動申請はフォームのフィールドに指定した初期設定値プロパティが動作しない仕様のため、送信する項目の中に“申請日”、“申請者”の情報を設定するための準備を行い、その後 kintone2xpoint で送信用のインスタンス（例では k2x）を作成します。

引数には接続先となる X-point サーバーに係わる情報と基本的な動作を指定します。【申請】に使用される「自動申請設定」は「フォームコード」と「承認ルートコード」の一致で関連付けられます。

なお、指定する API トークンはアクセス権に【追加】が指定されている必要があります。

ゲストスペースで利用するアプリの場合は初期化設定のオプション項目でゲストスペースの ID を指定してください。

【ゲストスペース ID の指定】

```
var domCd      = "trial";           // 申請を行う X-point のドメインコード
var fqdn      = "xpoint.atledcloud.jp"; // X-point が動作するサーバーの FQDN
var formcd    = "KeiyakuShinsa";    // 申請を行うフォームのフォームコード
var routecd   = "KeiyakuShinsa";    // 申請を行う承認ルートのコード
var api_token = "KeiyakuShinsaM!!RyeHRFYnfB19Ftrxn01HLzd";
var options   = { "proxyusercd": "u001", // X-point にログイン名と一致するユーザーCD が無い場合に使う CD
                  "guest": "10"        // kintone アプリのゲストスペースの ID
                };
```

■項目設定（表以外）

```
k2x.add_elements(body_json, record, [  
    // X-point 側, kintone 側  
    { "xname": "txtShinseibi", "value": post_date },  
    { "xname": "txtShinseisya", "value": loginname },  
    { "xname": "txtKokyakucode", "field": "txtKokyakucode" },  
    { "xname": "txtKokyakumei", "field": "txtKokyakumei" },  
    { "xname": "txtTantou", "field": "txtTantou" },  
    { "xname": "txtKiboubi", "field": "txtKiboubi" },  
    { "xname": "txtKanricode", "field": "txtKanricode" },  
    { "xname": "txtMemo", "field": "txtMemo" }  
]);
```

表に含まれないフィールドの転送を行います。自動申請設定ウィザードの「CSV フォーマット」設定で行った7列目以降の項目が該当します。

第3引数が X-point と kintone の項目対応付けを行う部分になり、“xname” が X-point 側のフィールド ID、“field” が kintone 側のフィールドコードに対応します。自動申請設定と対応付けられるため、“xname” の並びは自動申請設定の7列名以降と一致する必要があります。個々の項目を記述する際、

```
{ "xname": "txtKokyakucode", "field": "txtKokyakucode" }
```

と記述すると、kintone のレコードに登録されている内容が申請の内容となります。記述の際に

```
{ "xname": "txtKokyakucode", "field": "txtKokyakucode", "value": "加工した文字列" }
```

と “value” プロパティの追記を行うと、“field” が示す項目の内容ではなく “value” に指定した値が X-point に送信されます。“value” の利用では

```
{ "xname": "txtKokyakucode", "value": "加工した文字列" }
```

で指定することができます。“field”項目が無い場合は“value”の値は kintone のフィールド型 “文字列（1行）”として X-point に送信されます。“field”を併用した場合は field に指定したフィールドコードが示す型で “value” 値が送信されます。

■項目設定（表）

表部分の内容を X-point に送信します。表定義の内容をそのまま転送する方法と、登録内容を加工しながら送信する方法の2パターンを選択することができます。

1) 内容をそのまま送信する

```
var tbl_group1 = k2x.create_table_porperty(body_json, "group1");
k2x.add_table_elements(tbl_group1, record['group1'], [
    { "xname": "txtKeiyakusyono", "field": "txtKeiyakusyono" },
    { "xname": "txtKeiyakusyomei", "field": "txtKeiyakusyomei" } ] );
```

2) 内容を加工しながら送信する

```
var tbl_group1 = k2x.create_table_porperty(body_json, "group1");
var tableData = record['group1']; // kintone の表部分を取り出す
for(var rowno=0;rowno < tableData.value.length; ++rowno){ // kintone の表レコード分繰り返す
    var str = "加工した内容";
    var tbl_row = k2x.add_table_row(tbl_group1); // 送信用行配列を作る
    k2x.add_elements(tbl_row, tableData.value[rowno]['value'], [
        { "xname": "txtKeiyakusyono", "field": "txtKeiyakusyono" },
        { "xname": "txtKeiyakusyomei", "field": "txtKeiyakusyomei", "value": str } ] );
}
```

■送信（申請）

作成した送信用データを送信します。

【タイプ1】

kintone に登録するレコード情報を送信します。登録結果を JavaScript 内で取得することは出来ません。

```
k2x.send(body_json);
```

【タイプ2】

kintone への登録について成否を知ることが出来ます。第2引数に指定した書類内のフィールド ID が示すフィールドに、kintone のレコード ID を設定することが出来ます。

■標準の登録処理を利用する場合の記述方法

```
var status = k2x.send_kintone(body_json, 'kintone_rec_id');
if (status) {
    alert("kintone に登録しました");
} else {
    alert("kintone に登録できませんでした");
}
```

※ 複数枚フォームの場合、フィールド ID の次（第3引数）にページ番号を追記します。

第2引数に kintone からの JSON 応答を処理する Callback 関数を指定することができます。Callback 関数を指定する場合、標準動作の処理が実行されなくなるため、書類内のフィールドにレコード ID を保存する処理を作成して頂く必要があります。

■ Callback 処理を自作する場合の記述方法

```
// Callback 処理
var send_kintone_callback=function(status, responseText, f_kintone_id, pgno) {
  var ret_status = false;
  if((200 <= status && status < 300) || (status == 304)) {
    if (responseText && responseText.length > 0) {
      var kr = JSON.parse(responseText);
      if (kr.id && f_kintone_id) {
        var f=XWEB.getXPObject(f_kintone_id, pgno);
        if(f) {
          f.setValue(kr.id); // 書類上のフィールドに kintone のデータ ID を設定する
        }
      }
      ret_status = true;
    } else {
      alert("その他の応答:" + status);
    }
  } else { // 応答エラーの場合
    if (responseText && responseText.length > 0) {
      var kr = JSON.parse(responseText);
      alert("code : " + kr.code + " , id : " + kr.id + "¥n" +
          kr.message + (kr.errors ? this.errorDetail(kr.errors): ""));
    } else {
      alert("その他の応答:" + status);
    }
  }
  return ret_status; // send_kintone の呼び出し元に処理結果を返す
};
// 送信処理
var status = k2x.send_kintone(body_json, send_kintone_callback, 'kintone_rec_id');
if (status) {
  alert("kintone に登録しました");
} else {
  alert("kintone に登録できませんでした");
}
```

Callback 処理を作成した場合、X-point からの応答が“responseText”変数にセットされます。この内容を JSON.parse で解析することで、応答の内容を利用できるようになります。JSON.parse 解析できない場合は、ブラウザが X-point と通信できないといった通信経路の問題が生じている場合があります。

【申請成功のケース】（実際の応答に改行等は含まれていません）

```
{
  "docs": [
    {
      "kintone_rec_no": "12",
      "docid": 2889,
      "message_type": 3,
      "message": "書類が提出されました (ID = 2889)"
    }
  ]
}
```

【申請失敗のケース】（実際の応答に改行等は含まれていません）

```
{
  "docs": [
    {
      "kintone_rec_no": "12",
      "docid": -1,
      "message_type": 1,
      "message": "指定された書類 ID[2878]は登録されていません。"
    }
  ]
}
```

【取得できる情報】

| 分類 | プロパティ名 | 取得内容 |
|-----------------|----------------|---|
| 処理結果の配列 | docs | kintone レコード ID、書類 ID、応答種別、応答内容をセットにした配列。 sendDetailPage の場合は 1 件のみ格納される |
| kintone レコード ID | kintone_rec_no | エラー原因に係わる kintone のレコード ID。 kintone レコードと関係ない場合は -1 |
| X-point 書類 ID | docid | X-point 上での書類 ID |
| 応答種別 | message_type | 1: エラーメッセージ、2: 警告メッセージ、3: インフォメーション ※ エラー(1)、その他(2,3)で扱って構いません |
| 応答内容 | message | メッセージテキスト |

4.2.3. X-point に添付ファイルを転送

X-point の申請に添付ファイルを含める場合、“send” を実行する直前に、kintone 内に格納された添付ファイルに関する情報を送信データに追加します。

```
var tbl_group1 = k2x.create_table_property(body_json, "group1");
k2x.add_table_elements(tbl_group1, record['group1'], [
    { "xname": "txtKeiyakusyono", "field": "txtKeiyakusyono" },
    { "xname": "txtKeiyakusyomei", "field": "txtKeiyakusyomei" } ]);

// 添付ファイル情報（実ファイルは X-point 側から DL される）
k2x.add_attach_files(body_json, record['attached_file']);

k2x.send(body_json);
```

追加部分

“add_attach_file”メソッドに添付ファイルのフィールドコードを指定し、ファイル取得に必要なキー情報を追加設定します。情報を追加すると X-point 側で申請を行う際に、kintone から添付ファイル本体をダウンロードし申請書類に添付を行います。

- ※ X-point 側の「kintone 連携設定」の項目に kintone が動作するサーバーの FQDN、ファイル取得のための API トークン登録が必須です。登録は、kintone2xpoint クラスを初期化する際に指定した API トークンが対応する項目になります。（必要なアクセス権は「レコード閲覧」）

4.2.4. X-point にコメントを転送

連携用のクラス kintone2xpoint を初期化する際に申請書類にコメントを転記する指定を行います。

```
var domCd      = "trial";           // 申請を行う X-point のドメインコード
var fqdn       = "xpoint.atledcloud.jp"; // X-point が動作するサーバーの FQDN
var formcd     = "KeiyakuShinsa";    // 申請を行うフォームのフォームコード
var routecd    = "KeiyakuShinsa";    // 申請を行う承認ルートコード
var api_token  = "KeiyakuShinsaM!!RyeHRFYnfBI9Ftrxn01lHLzd";
var options    = { "proxyusercd": "u001", // X-point にログイン名と一致するユーザーCD が無い場合に使う CD
                  "comment": true // 申請時に kintone のコメントを転記する
                };
```

..... (途中省略)

```
var k2x = new kintone2xpoint(domCd, fqdn, formcd, routecd, api_token, options);
```

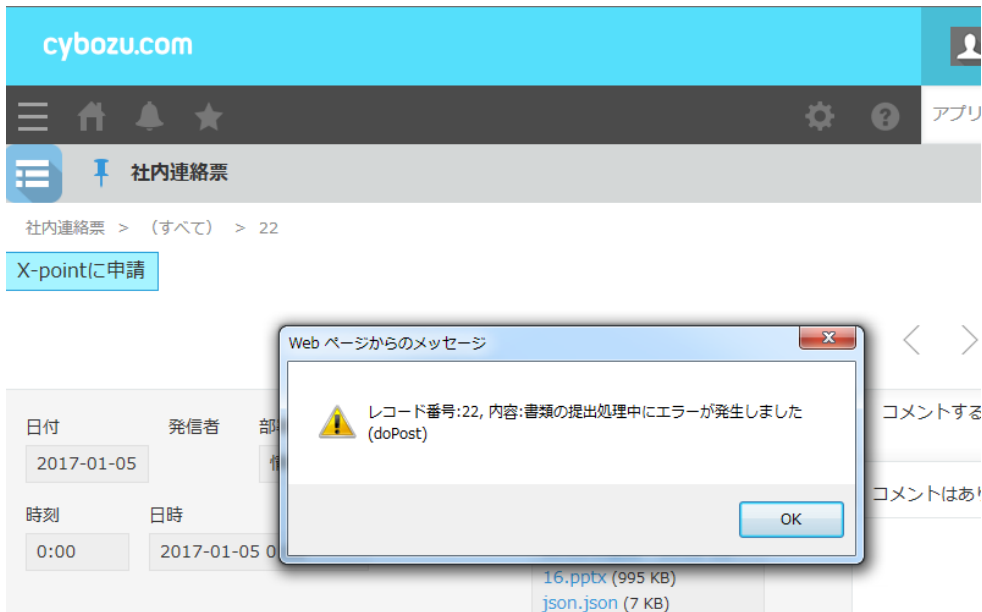
kintone のレコードに設定されたコメント内容は kintone の JavaScript API では取得できないため、X-point サーバー側から取得がおこなわれます。登録されるコメントは、kintone のログイン名と一致するユーザーコードを持つユーザーは、X-point 上での登録名で、X-point に一致するユーザーが居ない場合は kintone の登録名でコメント登録されます。X-point 上にユーザー登録がある場合のみ登録時にセットされたユーザーに限りコメントの修正ができます。

- ※ X-point 側の「kintone 連携設定」の項目に kintone が動作するサーバーの FQDN、ファイル取得のための API トークン登録が必須です。登録は、kintone2xpoint クラスを初期化する際に指定した API トークンが対応する項目になります。（必要なアクセス権は「レコード閲覧」）

4.2.5. 申請失敗時の表示

X-point に申請できない形式のデータを送信した場合はエラーが発生します。kintone の画面上にはエラーの発生を示すダイアログが表示されます。

【申請エラー発生時の表示】



発生したエラーの詳細は「管理サイト」の「ドメイン」>「タスク管理」で確認します。



kintone 連携での申請はタスク種別が「kintone 連携」で記録されます。

ドメイン管理 → ドメイン → フォーム関連情報 → メール設定 → ログ → タスク管理 → 定義書出力

タスク管理 > kintone連携 > タスクログ

タスク

- ▶ 自動申請
- ▶ ユーザインポート
- ▶ マスタインポート
- ▶ クエリCSV出力
- ▶ 督促メール
- ▶ 回覧既読
- ▶ クリーンアップ
- ▶ タスクログ削除
- ▶ kintone連携実行ログ

kintone連携タスクログ

検索条件

実行日 2020 年 10 月 30 日 ~ 2020 年 10 月 30 日

実行結果 成功 失敗 警告

30件 ずつ 検索

5件中 1-5表示

| 処理日 | 実行結果 | メッセージ |
|---------------------|------|-----------------------------|
| 2020/10/30 13:44:52 | 失敗 | エラー: 1件 自動申請名: from kintone |
| 2020/10/30 13:43:28 | 失敗 | エラー: 1件 自動申請名: from kintone |
| 2020/10/30 13:42:23 | 失敗 | エラー: 1件 自動申請名: from kintone |
| 2020/10/30 13:41:21 | 失敗 | エラー: 1件 自動申請名: from kintone |
| 2020/10/30 13:33:44 | 失敗 | エラー: 1件 自動申請名: from kintone |

5件中 1-5表示

※ 指定直後は操作した日に記録された「失敗」、「警告」が表示されます。「成功」をチェックし【検索】を行うと X-point に申請が正しく行われた場合も一覧に表示されるようになります。

メッセージ部分をクリックすると詳細が表示されます。

実行ログ

| 処理日 | 実行結果 | メッセージ |
|---------------------|------|-----------------------------|
| 2020/10/30 13:49:53 | 成功 | 成功: 1件 自動申請名: from kintone |
| 2020/10/30 13:49:43 | 成功 | 成功: 1件 自動申請名: from kintone |
| 2020/10/30 13:49:40 | 成功 | 成功: 1件 自動申請名: from kintone |
| 2020/10/30 13:48:02 | 成功 | 成功: 1件 自動申請名: from kintone |
| 2020/10/30 13:47:32 | 成功 | 成功: 1件 自動申請名: from kintone |
| 2020/10/30 13:44:52 | 失敗 | エラー: 1件 自動申請名: from kintone |
| 2020/10/30 13:43:28 | 失敗 | エラー: 1件 自動申請名: from kintone |

【成功をクリック】

2020/10/30 13:49:53 (kintone連携) [検索画面へ戻る](#)

| 明細番号 | 実行結果 | 詳細メッセージ |
|------|------|---------------------------------|
| 1 | 成功 | [ID:26179] 書類が提出されました。件名1: C001 |

【失敗をクリック】

2020/10/30 13:44:52 (kintone連携) [検索画面へ戻る](#)

| 明細番号 | 実行結果 | 詳細メッセージ |
|------|------|--|
| 1 | エラー | No.1: Error: 承認ルートで設定された承認者に対して該当者が一人も見つからないため提出できません。承認ルートの設定が適切かをシステム管理者にお問い合わせください。 |

記録されるメッセージの内容は「自動申請」機能と同じ形式での出力になります。

例は申請した kintone のデータ項目数と「自動申請」で定義したフィールド数が異なる場合のエラーです。

4.3. X-point に登録された書類を kintone 上で開く

kintone アプリから X-point の書類を表示することができます。但し、X-point の書類を開くことができるユーザーは、X-point にログイン ID を持つユーザーだけになります。

- ※ X-point にログイン ID が必要です。
- ※ X-point のログイン ID は kintone のログイン名と同一、もしくは連携 JavaScript クラス (kintone2xpoint) のオプションパラメータ (usercd) で指定されたログイン ID と一致する必要があります。
- ※ kintone が持つ情報を使って、X-point 上の書類を一意に特定する必要があります。
- ※ X-point v2.4.02 で導入された連携方法になります。v2.5 以降は連携テンプレートの利用を推奨します。

4.3.1. アプリに【書類表示】ボタンを追加

kintone アプリのレコード詳細を表示する画面の【X-point に申請】ボタン横に【申請書類を表示】ボタンを追加します。



【kintone-sample.js】(ボタン作成まで、ボタンクリックで動作する xpoint_openDoc は次項で作成します)

```
(function () {
  "use strict";

  // レコード詳細表示イベント (PC)
  kintone.events.on('app.record.detail.show', function (event) {
    // メニュー右側の空白部分にボタンを設置
    var btn = document.createElement('button');
    btn.id = 'btn_xp_doEntry';
    btn.innerHTML = 'X-point に申請';
    btn.onclick = function () {
      kintone_to_xpoint(event);
    }
    kintone.app.record.getHeaderMenuSpaceElement().appendChild(btn);

    var space = document.createElement('span');
    space.innerHTML = "&nbsp;&nbsp;&nbsp;";
    kintone.app.record.getHeaderMenuSpaceElement().appendChild(space);

    var btn = document.createElement('button');
    btn.id = 'btn_xp_openDoc';
    btn.innerHTML = '申請書類を表示';
    btn.onclick = function () {
      xpoint_openDoc();
    }
    kintone.app.record.getHeaderMenuSpaceElement().appendChild(btn);
  });

  // レコード一覧表示イベント (モバイル)
  kintone.events.on('mobile.app.record.detail.show', function (event) {
    var bt_div = document.createElement('div');
    bt_div.setAttribute('id', 'addButton');
    bt_div.setAttribute('name', 'addButton');
    var bt = document.createElement('button');
    bt.setAttribute('id', 'btn_xp_doEntry');
    bt.onclick = function () {
      kintone_to_xpoint(event);
    };
    bt.innerHTML = 'X-point に申請';
    bt_div.appendChild(bt);

    var space = document.createElement('span');
    space.innerHTML = "&nbsp;&nbsp;&nbsp;";
    bt_div.appendChild(space);

    var bt = document.createElement('button');
    bt.setAttribute('id', 'btn_xp_openDoc');
    bt.onclick = function () {
      xpoint_openDoc(event);
    };
    bt.innerHTML = '申請書類を表示';
    bt_div.appendChild(bt);
    kintone.mobile.app.getHeaderSpaceElement().appendChild(bt_div);
  });
})();
```

【X-point に申請】 ボタンの右側にスペースを作る

【申請書類を表示】 ボタンを追加する

【X-point に申請】 ボタンの右側にスペースを作る

【申請書類を表示】 ボタンを追加する

4.3.2. 書類を表示

アプリに【申請書類を表示】ボタンを追加した際に未作成の「xpoint_openDoc」を追加します。

【kintone-smample.js 追加する xpoint_openDoc 関数】

```
xpoint_openDoc=function() {  
  var domCd      = "trial";           // 申請を行う X-point のドメインコード  
  var fqdn       = "xpoint.atledcloud.jp"; // X-point が動作するサーバーの FQDN  
  var formcd     = "KeiyakuShinsa";    // 申請を行うフォームのフォームコード  
  var routecd   = "KeiyakuShinsa";    // 申請を行う承認ルートのコード  
  var api_token  = "KeiyakuShinsaM!!RyeHRFYnfB19Ftrxn01HLzd";  
  
  var k2x = new kintone2xpoint(domCd, fqdn, formcd, routecd, api_token);  
  // 表示中のレコード ID を渡す  
  k2x.openDoc( { "xname":"kintone_id:9", "value": kintone.app.record.getId() } );  
};
```

※ 指定する API トークンは X-point の管理サイトで取得したものです。

※ openDoc 関数の"xname":"については、上記例のように「フィールド ID+":"+フィールドタイプ」で指定してください。

書類表示の処理は初期設定、表示条件の指定から成り立っています。

以降で各処理を順に説明します。

■初期設定部分

```
var domCd      = "trial";           // 申請を行う X-point のドメインコード  
var fqdn       = "xpoint.atledcloud.jp"; // X-point が動作するサーバーの FQDN  
var formcd     = "KeiyakuShinsa";    // 申請を行うフォームのフォームコード  
var routecd   = "KeiyakuShinsa";    // 申請を行う承認ルートのコード  
var api_token  = "KeiyakuShinsaM!!RyeHRFYnfB19Ftrxn01HLzd";  
  
var k2x = new kintone2xpoint(domCd, fqdn, formcd, routecd, api_token);
```

基本的に申請を行う場合と同じ手順で初期設定を行い表示用のインスタンス（例では k2x）を作成します。

引数には接続先となる X-point サーバーに係わる情報と基本的な動作を指定します。書類表示では指定する API トークンのアクセス権に【追加】が指定されている必要があります。

■書類表示部分

書類表示は初期化を行った連携用のインスタンス（例では k2x）で“openDoc”を実行します。但し、実行する際に必ず登録書類を一意（1件）に絞り込むための条件指定を指定します。表示は登録されている書類のなかから、指定した条件とフォームコードで決定されます。

【書類の絞り込み条件】

- 1) 初期化時に指定されたフォームコードが対応するフォームで登録された書類
- 2) “openDoc” の引数で指定された条件

【レコード ID による表示】

```
k2x.openDoc( { "xname": "kintone_id", "value": kintone.app.record.getId() } );
```

上記の例は、X-point の書類に Hidden フィールドで設定した kintone アプリのレコード ID による絞りこみを行います。

“xname”がフォーム上のフィールド名、“value”が比較する値になります。

比較値に「%」等を含めることはできません。(前方一致、後方一致、及び正規表現の利用はできません)

【複数条件の指定例】

```
k2x.openDoc([ { "xname": "kintone_id", "value": kintone.app.record.getId() },  
              { "xname": "app_id", "value": kintone.app.getId() }  
            ] );
```

絞り込み条件は複数指定することが可能で、対応項目と値の組み合わせを配列で指定します。複数指定した場合は全て AND 条件で動作します。上記の例は、kintone アプリのレコード ID、及びアプリ ID での絞り込みになります。(書類にアプリ ID も登録する必要があります)

【複数枚フォームの書類における指定例】

```
k2x.openDoc( { "xname": "kintone_id", "value": kintone.app.record.getId(), "page": 2 } );
```

複数枚フォームを利用する書類を表示する場合は “xname”, “value” に加えて “page” 項目を追加し、“xname” で指定するフィールドが何ページ目にあるかを指定します。

【kintone 上で一意になる番号を付加している場合の例】

```
var record = kintone.app.record.get();  
var denpyo_no = record['record']['DENPYO_NO']['value'];  
  
k2x.openDoc( { "xname": "denpyo_no", "value": denpyo_no } );
```

上記は “denpyo_no” フィールドに格納された内容と一致する書類を表示する場合があります。

【閲覧用の X-point ログイン ID を使う場合】

```
var options = { "usercd": "viewuser" };  
  
var k2x = new kintone2xpoint(domCd, fqdn, formcd, routecd, api_token, options);  
k2x.openDoc( { "xname": "kintone_id", "value": kintone.app.record.getId() } );
```

kintone を利用するユーザーが X-point のログイン ID を持たないような場合に書類閲覧をさせるには、閲覧専用のログイン ID を用意します。例は “viewuser” を閲覧専用ログイン ID とした場合です。

4. 4. kintone からの操作での注意事項

kintone から X-point に格納されている書類の操作（書類表示、ワークフロー操作）を行う場合、同一ブラウザで X-point を利用していると相互に影響する場合があります。

利用中の X-point に影響を受ける機能は承認状況のダイアログ表示、影響を相互に受ける機能は書類ビューアによる書類表示です。その他の機能は kintone と X-point を利用するブラウザが同一であっても影響を受けることはありません。

4. 4. 1. 承認状況画面のデザインが受ける影響

kintone より承認状況画面の表示（getDocumentStatusView）を行った場合、承認状況の照会を行う X-point に同一セッションでログインしているユーザーの画面デザイン設定に影響を受けます。影響を受けるのはデザインテーマのみです。表示される承認状況は指定された書類の情報が正しく表示されます。

■ 同一ブラウザで X-point の利用が無い場合、利用ユーザーが初期設定から変更していない場合

| 承認進捗状況 (書類No.26180) | | | |
|---------------------|-----------|-------|---------------------|
| 件名1 | 営業部 | | |
| 件名2 | 営業太郎 | | |
| フォーム | A368出張申請書 | 承認ルート | 規定のルート |
| 承認状況 | 承認中 | 申請日 | 2020/10/30 14:14:45 |

| 第1版 | |
|--|--|
| 申請  申請 | 承認  STEP 1 |
| 管理者 未所属 20/10/30 14:14:45 | AND (2) 佐伯 宗一 (未処理) 株式会社えいとれつ商事 役員 木々 和明 (未処理) 株式会社えいとれつ商事 社長 |

■ 同一ブラウザで X-point の利用があり、利用ユーザーがデザインを変更している場合（例は「グレー」テーマ）

| 承認進捗状況 (書類No.26180) | | | |
|---------------------|-----------|-------|---------------------|
| 件名1 | 営業部 | | |
| 件名2 | 営業太郎 | | |
| フォーム | A368出張申請書 | 承認ルート | 規定のルート |
| 承認状況 | 承認中 | 申請日 | 2020/10/30 14:14:45 |

| 第1版 | |
|--|--|
| 申請  申請 | 承認  STEP 1 |
| 管理者 未所属 20/10/30 14:14:45 | AND (2) 佐伯 宗一 (未処理) 株式会社えいとれつ商事 役員 木々 和明 (未処理) 株式会社えいとれつ商事 社長 |

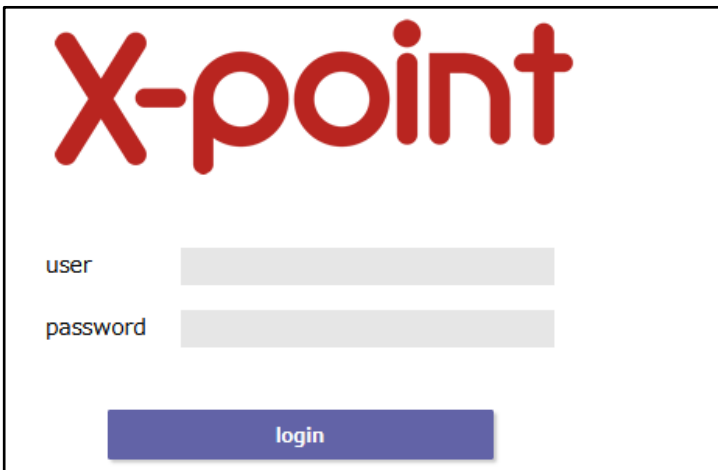
4.4.2. 書類表示のログイン認証が与える影響

kintone より書類表示 (openDoc) を行った場合、書類が格納されている X-point に同一セッションでログインしているユーザーが API 実行時に指定されているユーザーと同一であるか否かによる動作が異なります。

※ API 実行時の指定ユーザーとは、連携インスタンスの “post_usercd” (あるいは “post_proxycd”) に指定されたユーザー、指定されていない場合は、kintone のログイン名を指定ユーザーに使用します。

1) 書類を格納している X-point サーバーにログインしているユーザーがない場合

表示する書類を格納する X-point サーバーに対するログインセッションが無い場合、X-point にログインをする必要があります。ログインは、連携 API (openDoc) を実行する際に指定されたユーザーに対応するログイン ID、パスワードの入力が必要です。



The image shows a login form for X-point. At the top, the text "X-point" is written in a large, red, stylized font. Below this, there are two input fields: one labeled "user" and one labeled "password". Both fields are currently empty. At the bottom of the form, there is a blue button with the text "login" in white.

ログイン認証が成功すると書類の内容が表示され、以後はブラウザを全て終了するか、ログインセッションが破棄されない限り、同じユーザーで書類を表示する間はログイン認証画面の表示は行われず直接書類内容の表示になります。



The image shows a screenshot of a document review interface. The main header is a dark green bar with the text "契約書 社内審査" (Contract Document Internal Review). To the right of this header, there are three fields: "申請日:" (Application Date), "申請者:" (Applicant), and "審査No:" (Review No.) with the value "KS-2017-00469". Below the header, there is a flowchart showing the review process. It starts with a box labeled "担当" (Responsible) containing a red circular stamp that says "木戸 17-06-27 申請" (Kido 17-06-27 Application). This is followed by an arrow pointing to a box labeled "上長承認" (Supervisor Approval). Another arrow points to a box labeled "法務" (Legal), which is further divided into three sub-sections: "受付" (Receipt), "内容確認" (Content Confirmation), and "上長承認" (Supervisor Approval). This is followed by another arrow pointing to a box labeled "担当" (Responsible), and finally an arrow pointing to a box labeled "上長承認" (Supervisor Approval). Below the flowchart, there is a table with the following data:

| | | |
|---------|------------|----------|
| 顧客名 | G012 | 株式会社がつてん |
| 主担当 | 木戸 武史 | |
| 審査完了希望日 | 2017/06/20 | |

On the right side of the screen, there is a vertical column of buttons: "提出" (Submit), "編集" (Edit), "下書き" (Draft), "コピー" (Copy), "承認" (Approve), "却下" (Reject), "保留" (Hold), "差し戻し" (Return), "承認取戻" (Return Approval), "PDF", and "添付" (Attachment).

2) API 実行時に指定されたユーザーコードと X-point サーバーにログイン中のユーザーが一致する場合

書類が格納されている X-point にログイン済みの場合、API 実行で指定されたユーザーと同一であれば認証画面を介することなく書類の内容が表示されます。

3) API 実行時に指定されたユーザーコードと X-point サーバーにログイン中のユーザーが一致しない場合

書類が格納されている X-point にログイン中のユーザーが API 実行時に指定されたユーザーと異なる場合、既存のログインが破棄され、API 実行時に指定されたユーザーでの認証が行われます。

つまり、認証画面が表示されたタイミングで既存の X-point 利用ユーザーは強制的にログアウトが行われます。その際、既存のブラウザ画面は切断前の画面表示のままになっている点に注意して下さい。

第5章. X-pointフォームから kintoneへの連携

5.1. 管理サイトの設定で kintone にレコードを追加・更新する

フォーム管理画面の「kintone 連携」に用意されている X-point から kintone へのレコード追加・更新機能を利用する場合の手順を解説します。設定画面を利用しての連携になるため、フォーム、アプリ上に用意された項目の対応を指定する形式になり書類上の項目を加工が必要なケースでの利用は出来ません。

- ※ kintone にレコードを追加・更新するためには、kintone アプリの設定で API トークンの“アクセス権”に“レコード追加”、“レコード編集”が許可されている必要があります。
- ※ JavaScript を利用した連携ではフォームナビボタンの操作が必須になるため、ナビボタンを操作できないケースでは連携動作に制約が発生する場合があります。具体的な制約事項は「[JavaScript を利用して kintone にレコードを追加・更新する](#)」を参照してください。

5.1.1. 連携先の kintone アプリを指定

連携先となる kintone アプリは管理サイトの「フォーム管理」で一覧のアクション項目から「kintone 連携」指定して行います。

| 名称 | 枚数 | FCD | テーブル名 | TSFファイル | アクション |
|-----------------|----|------|----------|----------|--|
| 新しいフォームグループ (3) | | | | | |
| A367出張報告書 | 1枚 | A367 | xp788929 | A367.tsf | 他操作 |
| A368出張申請書 | 1枚 | A368 | xp12660 | A368.tsf | 他操作 |
| 契約書社内審査 | 1枚 | C001 | xp896690 | C001.tsf | 他操作 フォーム追加 アクセス権限 フォーム編集権限 書類制御 印刷権限 kintone連携 Webhook設定 CSVファイル作成 ゲスト |

「kintone 連携」画面には上側に「kintone→X-point」、下側に「X-point→kintone」の設定項目が用意されているため、下側にある【連携アプリを追加する】をクリックして設定画面を準備します。

クリックすると設定用の欄が追加されます。

X-pointからkintoneアプリへのレコード追加・更新タイミングを指定します。

| 連携先kintone情報 | レコード追加・更新タイミング | 備考 | 操作 |
|---|--|----|------------------------------|
| <div style="border: 2px solid red; padding: 5px;"><p>+ kintone連携先アプリを追加する</p><p>連携先kintone情報</p><p>ドメイン</p><p>アプリID 0</p><p>APIトークン</p><p>ゲストID 0 (未指定の場合は0)</p><p><input type="button" value="確認"/></p><p>連携先のkintone情報を指定してください</p></div> | <p>「kintoneレコードID」書類内保存フィールド --- 保存しない ---</p> <p><input type="checkbox"/> 添付ファイルの更新 --- 添付フィールドコード ---</p> <p><input type="checkbox"/> コメントの追加</p> <p><input type="checkbox"/> 書類削除と連動</p> <p><input type="checkbox"/> 完了書類のPDFを添付 -- 格納フィールドコード --</p> <p><input type="checkbox"/> 申請 <input type="checkbox"/> 承認1 <input type="checkbox"/> 完了書類修正</p> <p><input type="button" value="フィールド指定"/></p> | | <p>登録</p> <p>更新</p> <p>會</p> |

フォーム（本例の場合は「契約書社内調査」）が連携する kintone アプリの情報を指定します。

- kintone サーバーの FQDN
- アプリ ID
- アプリの API トークン
- ゲスト ID （指定無しの場合は 0 を設定）

API トークン以外はアプリを利用する際にブラウザのアドレスバーに表示される内容から確認することが出来ます。

【通常のアプリ（ゲスト ID を含まない）】



【ゲストアプリ（ゲスト ID を含む）】



API トークンは各アプリの設定画面で発行します。設定画面はアプリ画面右上の【・・・】から表示します。

| レコード | 顧客コード | 顧客名 | 主担当 | 審査完了希望日 | 管理コード | 審査主旨 | 審査No | 添 |
|------|-------|------|-------|------------|-----------|---------|--------|---|
| 87 | G018 | 株式会社 | 勝 太郎 | 2017-04-21 | KANRI-00C | X-point | KS-201 | |
| 47 | G018 | 株式会社 | 木戸 武史 | 2017-04-21 | KANRI-00C | X-point | KS-201 | |
| 38 | G001 | 株式会社 | 木戸 武史 | 1970-01-01 | | 「太陽が | KS-201 | |

アプリの【設定】画面で【API トークン】項目をクリックします。



【API トークン】画面が表示されたら【生成する】をクリックします。クリックすると、新しいAPI トークンが発行され、API トークンに対してアクセス権を指定する行が追加されます。



指定が完了したら、【API トークン】画面で【保存】、アプリ設定画面で【アプリを更新】を行い発行した API トークンを有効にします。

収集した連携先の kintone アプリ情報を設定します。項目の指定ができれば【確認】ボタンをクリックします。指定内容に誤りがない場合（kintone アプリからの応答が得られた）、【フィールド指定】のボタンがグレーからオレンジになります。

X-pointからkintoneアプリへのレコード追加・更新タイミングを指定します。

| 連携先kintone情報 | レコード追加・更新タイミング | 備考 | 操作 |
|---|---|----|--|
| ドメイン devettsfn.cybozu.com アプリID 4 APIトークン VE6umQloKph4llgClDafsqTSXcvSfNoB5zrQ ゲストID 0 <small>(未指定の場合は0)</small> <input type="button" value="確認"/> 連携先のkintone情報を指定してください | 「kintoneレコードID」書類内保存フィールド --- 保存しない --- <input type="checkbox"/> 添付ファイルの更新 --- 添付フィールドコード --- <input type="checkbox"/> コメントの追加 <input type="checkbox"/> 書類削除と連動 <input type="checkbox"/> 完了書類のPDFを添付 --- 格納フィールドコード --- <input type="checkbox"/> 申請 <input type="checkbox"/> 承認1 <input type="checkbox"/> 完了書類修正 <input type="button" value="フィールド指定"/> | | <input type="button" value="登録"/> <input type="button" value="更新"/> |

正しく指定すると色が変わる

【確認】を押下

X-pointからkintoneアプリへのレコード追加・更新タイミングを指定します。

| 連携先kintone情報 | レコード追加・更新タイミング | 備考 | 操作 |
|---|---|----|---|
| ドメイン devettsfn.cybozu.com アプリID 4 APIトークン VE6umQloKph4llgClDafsqTSXcvSfNoB5zrQ ゲストID 0 <small>(未指定の場合は0)</small> <input type="button" value="確認"/> | 「kintoneレコードID」書類内保存フィールド --- 保存しない --- <input type="checkbox"/> 添付ファイルの更新 --- 添付フィールドコード --- <input type="checkbox"/> コメントの追加 <input type="checkbox"/> 書類削除と連動 <input type="checkbox"/> 完了書類のPDFを添付 --- 格納フィールドコード --- <input type="checkbox"/> 申請 <input type="checkbox"/> 承認1 <input type="checkbox"/> 完了書類修正 <input type="button" value="フィールド指定"/> | | <input type="button" value="登録"/> <input type="button" value="更新"/> <input type="button" value="保存"/> |

接続が確認できると【フィールド設定】がクリック可能になります

| X-point (連携元) 契約書社内審査 (C001) | kintone (連携先) AppID:4, devettsfn.cybozu.com |
|-------------------------------------|--|
| 審査主旨 (txtMemo) [page-1] | 審査主旨 (txtMemo) |
| 申請日 (txtShinseibi) [page-1] | 申請日 (txtShinseibi) |
| 申請者 (txtShinseisya) [page-1] | 申請者 (txtShinseisya) |
| 審査No (txtNo) [page-1] | --- 連携フィールド --- |
| 顧客コード (txtKokyakucode) [page-1] | 管理コード (txtKanricode) |
| 担当名 (txtTantou) [page-1] | --- 連携フィールド --- |
| 完了希望日 (txtKiboubi) [page-1] | --- 連携フィールド --- |
| 管理コード (txtKanricode) [page-1] | --- 連携フィールド --- |

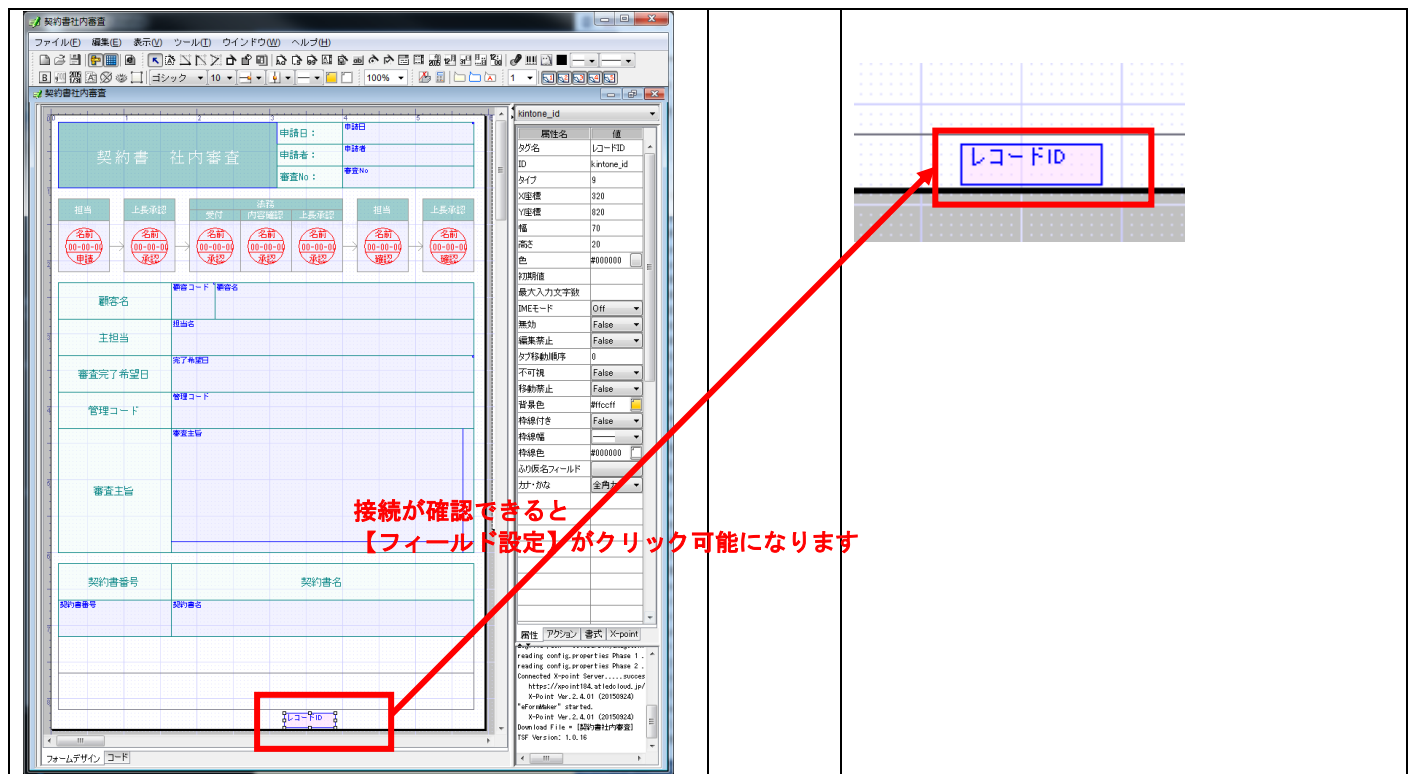
【フィールド設定】を押下し、連携元と連携先の対応項目を設定したのち、【閉じる】を押下します。

その後、指定ができたことを確認ののち【保存】をクリックします。

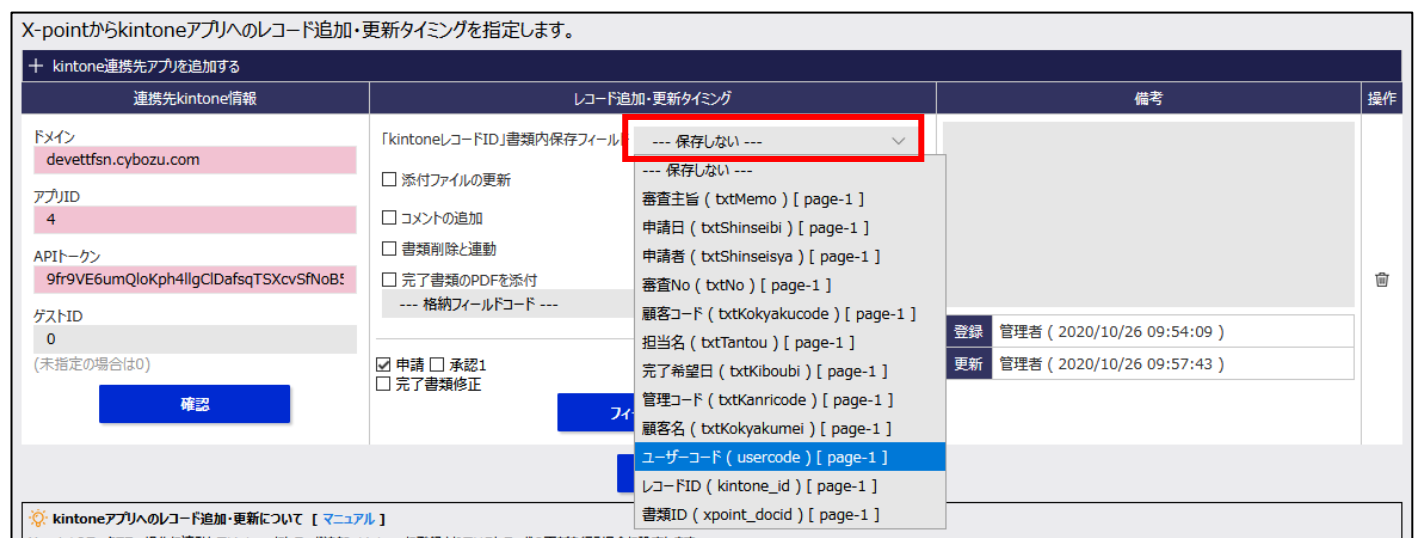
5.1.2. kintone レコード ID の保存フィールドを指定

kintone に追加したレコードを更新する必要がある場合、kintone に登録されているレコードを識別するためのレコード ID を更新する情報と送信する必要があります。そのためには、X-point の書類上に用意したレコード ID 保存フィールドの ID を指定します。本例では使用するフォーム「契約書社内審査」の下部に専用のフィールドを用意して利用します。

なお、kintone に登録したレコードを更新しない場合は、本項で行うレコード ID 保存用のフィールドを指定する必要はありません。



X-point が kintone へのレコード追加に成功すると kintone より追加されたレコード ID が応答として返されます。戻されたレコード ID を書類上の指定フィールドに保存します。



【「kintone レコード ID」書類内保存フィールド】にフィールド ID を指定したのち【保存】を行います。

X-pointからkintoneアプリへのレコード追加・更新タイミングを指定します。

| + kintone連携先アプリを追加する | | レコード追加・更新タイミング | 備考 | 操作 |
|---|--|----------------|--|----|
| 連携先kintone情報 ドメイン devettfsn.cybozu.com アプリID 4 APIトークン 9fr9VE6umQloKph4llgClDafsqTSXcvSfNoB! ゲストID 0 (未指定の場合は0) 確認 | レコード追加・更新タイミング 「kintoneレコードID」書類内保存フィールド ユーザーコード (usercode) [page-1] <input type="checkbox"/> 添付ファイルの更新 --- 添付フィールドコード --- <input type="checkbox"/> コメントの追加 <input type="checkbox"/> 書類削除と連動 <input type="checkbox"/> 完了書類のPDFを添付 --- 格納フィールドコード --- <input checked="" type="checkbox"/> 申請 <input type="checkbox"/> 承認1 <input type="checkbox"/> 完了書類修正 フィールド指定 | | 登録 管理者 (2020/10/26 09:54:09) 更新 管理者 (2020/10/26 09:57:43) | 會 |
| 保存 | | | | |

5. 1. 3. 添付ファイル登録先を指定

書類に添付されているファイルを kintone アプリの添付ファイルフィールドに格納します。

添付ファイルの扱いは X-point では書類の【添付】ナビボタンに一括して、kintone では複数の添付フィールドに夫々複数添付することが可能です。その為、書類の添付ファイルを kintone のレコードに格納する場合は対象となる添付ファイルフィールドを1つ指定し、指定された添付ファイルフィールドに全ての添付ファイルを格納します。

【X-point の書類】

この書類に添付されているファイル

| No | ファイル | サイズ | 備考 |
|----|--|----------------|----|
| 1 | 郵便番号マスタ _20170125T125605+0900.csv | 451,000bytes | |
| 2 | プロジェクトマスタ _20170125T171228+0900.csv | 9,993,398bytes | |

【kintone の添付ファイルフィールド】

添付ファイル

| |
|---|
| 郵便番号マスタ _20170125T125605+0900.csv (4 MB) プロジェクトマスタ _20170117T171228+0900.csv (684 bytes) |
|---|

X-pointからkintoneアプリへのレコード追加・更新タイミングを指定します。

| 連携先kintone情報 | レコード追加・更新タイミング | 備考 | 操作 |
|--|--|---|--|
| ドメイン devettfsn.cybozu.com アプリID 4 APIトークン 9fr9VE6umQloKph4llgClDafsqTSXcvSfNoB! ゲストID 0 (未指定の場合は0) | 「kintoneレコードID」書類内保存フィールド --- 保存しない --- <input checked="" type="checkbox"/> 添付ファイルの更新 <input type="checkbox"/> コメントの追加 <input type="checkbox"/> 書類削除と連動 <input type="checkbox"/> 完了書類のPDFを添付 --- 格納フィールドコード --- <input checked="" type="checkbox"/> 申請 <input type="checkbox"/> 承認1 <input type="checkbox"/> 完了書類修正 | --- 保存しない --- --- 添付フィールドコード --- 添付ファイル (attached_file) 押印済みPDF (pdf_document) | 登録 管理者 (2020/10/26 09:54:09) 更新 管理者 (2020/10/26 09:57:43) |

確認 フィールド指定 保存

ファイル保存先のフィールドコードを指定し【添付ファイルの更新】をチェック

X-pointからkintoneアプリへのレコード追加・更新タイミングを指定します。

| 連携先kintone情報 | レコード追加・更新タイミング | 備考 | 操作 |
|--|--|---|--|
| ドメイン devettfsn.cybozu.com アプリID 4 APIトークン 9fr9VE6umQloKph4llgClDafsqTSXcvSfNoB! ゲストID 0 (未指定の場合は0) | 「kintoneレコードID」書類内保存フィールド --- 保存しない --- <input checked="" type="checkbox"/> 添付ファイルの更新 <input type="checkbox"/> コメントの追加 <input type="checkbox"/> 書類削除と連動 <input type="checkbox"/> 完了書類のPDFを添付 --- 格納フィールドコード --- <input checked="" type="checkbox"/> 申請 <input type="checkbox"/> 承認1 <input type="checkbox"/> 完了書類修正 | --- 保存しない --- --- 添付フィールドコード --- 添付ファイル (attached_file) | 登録 管理者 (2020/10/26 09:54:09) 更新 管理者 (2020/10/26 09:57:43) |

確認 フィールド指定 保存

※ 連携先の kintone 情報に誤りがある場合、フィールドコード選択を行うコンボボックスに項目が表示されません。

5.1.4. 完了書類の登録先を指定

X-point 上で書類のワークフロー処理が完了した際に書類を PDF にしたものを指定された添付ファイルフィールドに登録します。
登録される PDF ファイルの名称は「[フォーム名].pdf」になります。

| 契約書 社内審査 | | 申請日: | 1970年01月01日 | | | | | | | | | | |
|----------|--------------------------------|-------|------------------|----|------------------|------------|----|------|------------------|----|------------------|------|------------------|
| | | 申請者: | 山口 直人 | | | | | | | | | | |
| | | 審査No: | KS-2017-00320 | | | | | | | | | | |
| 担当 | 山口 (17-04-18) 申請 | 上長承認 | 宮沢 (17-03-15) 承認 | 受付 | 坂本 (17-03-15) 承認 | 法務 内容確認 | 夏目 | 上長承認 | 太田 (17-03-15) 承認 | 担当 | 山口 (17-03-24) 確認 | 上長承認 | 宮沢 (17-03-24) 確認 |
| 顧客名 | G001 株式会社太陽商事 | | | | | | | | | | | | |
| 主担当 | 木戸 武史 | | | | | | | | | | | | |
| 審査完了希望日 | 1970年01月01日 | | | | | | | | | | | | |
| 管理コード | | | | | | | | | | | | | |
| 審査主旨 | 「太陽がいっぱいお天気大作戦」機材設置事業者との契約内容確認 | | | | | | | | | | | | |
| 契約書番号 | 契約書名 | | | | | | | | | | | | |
| S12000 | 設備納入契約 | | | | | | | | | | | | |
| K34000 | 取引契約 | | | | | | | | | | | | |

| 契約書 社内審査 | | 申請日: | 1970年01月01日 | | | | | | | | | | |
|----------|--------------------------------|-------|------------------|----|------------------|------------|----|------|------------------|----|------------------|------|------------------|
| | | 申請者: | 山口 直人 | | | | | | | | | | |
| | | 審査No: | KS-2017-00320 | | | | | | | | | | |
| 担当 | 山口 (17-03-17) 申請 | 上長承認 | 宮沢 (17-03-15) 承認 | 受付 | 坂本 (17-03-15) 承認 | 法務 内容確認 | 夏目 | 上長承認 | 太田 (17-03-15) 承認 | 担当 | 山口 (17-03-24) 確認 | 上長承認 | 宮沢 (17-03-24) 確認 |
| 顧客名 | G001 株式会社太陽商事 | | | | | | | | | | | | |
| 主担当 | 木戸 武史 | | | | | | | | | | | | |
| 審査完了希望日 | 1970年01月01日 | | | | | | | | | | | | |
| 管理コード | | | | | | | | | | | | | |
| 審査主旨 | 「太陽がいっぱいお天気大作戦」機材設置事業者との契約内容確認 | | | | | | | | | | | | |
| 契約書番号 | 契約書名 | | | | | | | | | | | | |
| S12000 | 設備納入契約 | | | | | | | | | | | | |
| K34000 | 取引契約 | | | | | | | | | | | | |

承認完了時に
格納

kintone アプリから PDF を
ダウンロード表示

押印済み契約書

契約書社内審査.pdf (11 KB)

フォーム名称の PDF を格納

X-pointからkintoneアプリへのレコード追加・更新タイミングを指定します。

| 接続先kintone情報 | レコード追加・更新タイミング | 備考 | 操作 |
|--|--|--|-----------------------------------|
| ドメイン devettfsn.cybozu.com | 「kintoneレコードID」書類内保存フィールド --- 保存しない --- <input checked="" type="checkbox"/> 添付ファイルの更新 添付ファイル (attached_file) <input type="checkbox"/> コメントの追加 <input type="checkbox"/> 書類削除と連動 | | 會 |
| アプリID 4 | <input checked="" type="checkbox"/> 完了書類のPDFを添付 --- 格納フィールドコード --- --- 格納フィールドコード --- <input checked="" type="checkbox"/> 添付ファイル (attached_file) <input type="checkbox"/> 押印済みPDF (pdf_document) | | |
| APIトークン 9fr9VE6umQloKph4llgClDafsqTSXcvSfNoB! | | 登録 管理者 (2020/10/26 09:54:09) 更新 管理者 (2020/10/26 09:57:43) | |
| ゲストID 0 (未指定の場合は0) | | | |
| <input type="button" value="確認"/> | <input type="button" value="フィールド指定"/> | | <input type="button" value="保存"/> |

X-pointからkintoneアプリへのレコード追加・更新タイミングを指定します。

| 接続先kintone情報 | レコード追加・更新タイミング | 備考 | 操作 |
|--|---|--|-----------------------------------|
| ドメイン devettfsn.cybozu.com | 「kintoneレコードID」書類内保存フィールド --- 保存しない --- <input checked="" type="checkbox"/> 添付ファイルの更新 添付ファイル (attached_file) <input type="checkbox"/> コメントの追加 <input type="checkbox"/> 書類削除と連動 | | 會 |
| アプリID 4 | <input checked="" type="checkbox"/> 完了書類のPDFを添付 --- 格納フィールドコード --- | | |
| APIトークン 9fr9VE6umQloKph4llgClDafsqTSXcvSfNoB! | <input checked="" type="checkbox"/> 申請 <input type="checkbox"/> 承認1 <input type="checkbox"/> 完了書類修正 | 登録 管理者 (2020/10/26 09:54:09) 更新 管理者 (2020/10/26 09:57:43) | |
| ゲストID 0 (未指定の場合は0) | | | |
| <input type="button" value="確認"/> | <input type="button" value="フィールド指定"/> | | <input type="button" value="保存"/> |

PDF ファイル保存先のフィールドコードを指定し
【完了書類の PDF を添付】をチェック

5.1.5. kintone レコードを追加・更新するタイミングを指定

書類を kintone に追加・更新するタイミングを指定します。kintone への連携動作は承認ステップ毎に実行の有無を指定でき、連携の動作は書類入力が行われた後の一連の処理が完了した後に行われるため、自動採番コンポーネントの動作結果が反映された状態で連携が行われます。

※ JavaScript による連携の場合、【申請】ステップで自動採番が行われる場合であっても申請ステップの連携で採番された番号を kintone のレコードに含めることは出来ません。ナビボタンに記述した JavaScript が採番前に実行されるためです。「kintone 連携」画面で指定する連携は、自動採番等の実行された後のデータを利用して連携するため、採番番号など各種のワークフローに係わる値が含まれた状態で動作します。

※

【ワークフローフォーム】

【通常フォーム】

申請 承認1 承認2
 完了書類修正

提出 修正

フィールド指定

フィールド指定

※ 下書き、保留、差し戻し、取戻し操作での連携は実行されません。

ワークフローフォームの場合に限り、設定されている承認ルート指定、管理サイトの承認状況画面での操作に影響を受けます。

【影響の種類】

① 「空ステップ」の場合

空ステップを連携動作に指定した場合、空ステップを通過するタイミングで連携が実行されます。具体的には、承認1が空ステップである場合に、申請、承認1、承認2で連携を有効にすると、申請された書類は、申請、承認1の連携が動作し、承認2で承認待ち（停止）状態になります。

② 「連続承認」が動作した場合

同一ユーザーが連続するステップが存在する場合などに「連続承認」が動作します。たとえば、申請、承認1、承認2に同じユーザーが指定されている場合、申請された書類は承認3で待ちになります。このようなケースでの連携は申請、承認1、承認2の連携を実行します。

③ 「完了書類の再申請」を行った場合

ドメイン管理で「再申請ポリシー」を「利用する」に指定している場合、完了状態のワークフロー書類を更新すると、書類の再申請処理が開始されます。その際、「完了書類修正」、「申請」の連携が動作します。（申請後、ステップ1で停止しない場合は、停止するステップの前ステップ迄の連携も動作します）

④ 「管理者スキップ」を行った場合

管理サイトの承認状況画面で承認者のスキップを行い結果次のステップに移動する場合、スキップされる承認ステップで連携動作が指定されている場合はスキップするステップの連携を実行してから承認待ちになります。

5.1.6. フィールド指定

連携を実行する際にフィールド値を設定する先となるフィールドコードを指定します。連携先となる kintone のフィールドコードを指定しない場合は、kintone 側の値は更新されず連携動作前の値が維持されます。

| X-point (連携元) 契約書社内審査 (C001) | kintone (連携先) AppID:4, devettfsn.cybozu.com |
|-------------------------------------|--|
| 審査主旨 (txtMemo) [page-1] | 審査主旨 (txtMemo) |
| 申請日 (txtShinseibi) [page-1] | 申請日 (txtShinseibi) |
| 申請者 (txtShinseisya) [page-1] | 申請者 (txtShinseisya) |
| 審査No (txtNo) [page-1] | --- 連携フィールド --- |
| 顧客コード (txtKokyakucode) [page-1] | 管理コード (txtKanricode) |
| 担当名 (txtTantou) [page-1] | --- 連携フィールド --- |
| 完了希望日 (txtKiboubi) [page-1] | --- 連携フィールド --- |
| 管理コード (txtKanricode) [page-1] | --- 連携フィールド --- |

連携時のデータには kintone (右側) のリストで選択されたフィールドのみが含まれます。

転送を行う際に X-point のフィールドにセットされている文字列は一定の条件に従って変換が行われます。

【連携フィールドのリストに表示されるフィールド種類】

| X-point フィールド種類・元 | kintone フィールド・先 | 変換処理の対象パターン | 変換後 kintone 用パターン |
|----------------------------|--|---|--|
| 文字フィールド テキストエリア | SINGLE_LINE_TEXT (文字列・1行) MULTI_LINE_TEXT (文字列・複数行) | 変換動作はありません | kintone で処理不可能な場合、 kintone 側でエラーになります |
| 文字フィールド・書式有り 日時フィールド・日付 | DATE (日付) | YYYY 年 MM 月 DD 日 yyyy/mm/dd yyyy/mm/dd hh:mm yyyy-mm-dd hh:mm yyyy-mm-dd hh:mm:ss yy-mm-dd yy/mm/dd yyyymmdd | yyyy-MM-dd ※“-”区切りの日付に変換します ※西暦2桁は先頭に“20”が追加されます |
| | 他 | | kintone で処理不可能な場合、 kintone 側でエラーになります |

| X-point フィールド種類・元 | kintone フィールド・先 | 変換処理の対象パターン | 変換後 kintone 用パターン |
|----------------------------|--|--|---|
| 文字フィールド・書式有り | DATETIME (日時) | YYYY 年 MM 月 DD 日 HH 時 MM 分 | yyyy-MM-dd' T' HH:mm:ss. SSS' Z |
| | | YYYY 年 MM 月 DD 日 | |
| | | yyyy/mm/dd hh:mm:ss | ※全て ISO8601 形式に変換します |
| | | yyyy-mm-dd hh:mm:ss | ※西暦 2 桁は先頭に“20”が追加されます |
| | | yyyy/mm/dd hh:mm | ※時刻がない場合は“00:00:00”になります |
| | | yyyy-mm-dd hh:mm | |
| | | yyyy/mm/dd | |
| | | yyyy-mm-dd | ※日付がない場合は処理日で追加されます |
| | | yy/mm/dd | |
| | | yy-mm-dd | |
| | | yyyymmdd | |
| | | hh:mm | |
| | 他 | kintone で処理不可能な場合、kintone 側でエラーになります | |
| 文字フィールド・書式有り 日時フィールド・時刻 | TIME (時刻) | hh:mm AM | HH:mm (24 時間形に変換) |
| | | hh:mm:ss AM | |
| | | hh:mm、 | |
| | | hh:mm:ss | |
| | | 他 | kintone で処理不可能な場合、kintone 側でエラーになります |
| テキストエリア | RICH_TEXT (リッチエディター) | 改行コードを に置換 1. ¥r¥n → 2. ¥n → | |
| 整数フィールド 数値フィールド | NUMBER (数値) | 変換動作はありません | 計算式はコピーされません |
| コンボボックス リストボックス | DROP_DOWN/MULTI_SELECT | 変換動作はありません | |
| ラジオボタン | RADIO_BUTTON (ラジオボタン) | 変換動作はありません | ※チェック値を相互で合わせる必要があります |
| チェックボックス | CHECK_BOX (チェックボックス) | kintone の同一チェックボックスに追加動作を行います。 | ※「チェックボックスの特殊な指定」を参照 ※非チェック値は利用できません |
| 文字フィールド | LINK (リンク) | 変換動作はありません | kintone で処理不可能な場合、kintone 側でエラーになります |
| 文字フィールド | USER_SELECT (ユーザー選択) ORGANIZATION_SELECT (組織選択) GROUP_SELECT (ロール選択) | 変換動作はありません | ※「選択フィールドの特殊な指定」を参照 |
| 西暦フィールド | SINGLE_LINE_TEXT (文字列・1行) | 文字列 1 行、日付、日時を指定した場合に限り、YYYY-MM-DD、又は yyyy-MM-dd' T' HH:mm:ss. SSS' Z ぶ変換されます。 | ※「西暦、月、日フィールドの変換動作」を参照 |
| 月フィールド | DATE (日付) | | |
| 日フィールド | DATETIME (日時) その他 | | |

5.1.6.1. チェックボックスの特殊な指定

X-point ではチェックボックスが単独項目で扱われますが、kintone では複数のチェックをまとめて扱う事が出来ます。その為、次のようなケースではX-point では2項目、kintone では1項目になります。

【X-point のフォーム】

The screenshot shows two separate checkbox fields in a form. The first field is labeled 'checkbox1' and contains the text 'チェックボックス1' with an unchecked checkbox. The second field is labeled 'checkbox2' and contains the text 'チェックボックス2' with an unchecked checkbox.

checkbox1、checkbox2 の2フィールド。
それぞれのフィールドのチェック値に
sample1、sample2 が設定される。

【kintone のフォーム】

The screenshot shows a single 'checkbox_group' field in a kintone form. It contains two items: 'sample1' and 'sample2', each with an unchecked checkbox.

checkbox_group の1フィールド。フィールドの項目と
して、sample1、sample2 の項目がある

数が異なるチェックボックス項目の対応は【フィールド指定】において、kintone 側で同じフィールドコードを指定します。

| | |
|------------------------------------|-------------------------|
| チェックボックス1 (checkbox1) [page-1] | checkbox (チェックボックス_0) |
| チェックボックス2 (checkbox2) [page-1] | checkbox (チェックボックス_0) |

チェックボックス以外の項目で同じフィールドコードを指定すると一方の値のみがkintoneにセットされます。

チェックボックスに限り、同じ連携先のフィールドコードが存在する場合、選択肢が複数あるものとして処理を行います。

なお、X-point では「非チェック値」を各項目で扱う事ができますが、kintone では扱う事が出来ないため、X-point のフォームを修正し「**非チェック値**」は空欄にする必要があります。空欄にしない場合、kintone 側で「選択肢がありません」という内容の入力チェックエラーが発生します。

【エラー例】

- kintone連携
code:CB_VA01, id: 1505999166-1455776981
message: 入力内容が正しくありません。
record.checkbox_group.values[0].value:"unchecked 1"は選択肢がありません。

5.1.6.2. 選択フィールドの特殊な指定

ユーザー選択、組織選択、グループ選択の3フィールドは、cybozu.com 共通管理機能で登録されるユーザーの「ログイン名」、組織情報の「組織コード」、グループ（ロール）情報の「グループコード」に対応しています。従って、X-point の書類上では、kintone のログイン名、組織コード、グループコードをフィールドに入力し kintone の各フィールドに対応付けることで動作します。また、3タイプでは【フィールド指定】において、kintone 側で同じフィールドコードを指定することができます。

【X-point の書類】

| | | |
|--------------|---------------------------------------|---|
| user_select | <input type="text" value="u002"/> | <input type="text" value="u001"/> |
| org_select | <input type="text" value="1310"/> | <input type="text" value="1320"/> |
| group_select | <input type="text" value="everyone"/> | <input type="text" value="Administrato"/> |

【kintone のレコード】

| |
|--|
| user_select |
|  木戸 武史 |
|  勝 太郎 |
| org_select |
|  営業 1 課 |
|  営業 2 課 |
| group_select |
|  Everyone |
|  Administrators |

- ※ 各フィールドに対し複数の書類フィールドを対応させた場合。
- ※ X-point から kintone に送る値は「ログイン名」、「組織コード」、「グループコード」です。表示名を指定しても kintone 側で無視されます。（エラーにはなりません）

5.1.6.3. 西暦、月、日フィールドの変換動作

X-point では日付を西暦、月、日を別のフィールドに分けて登録する事が出来、“日付セット”として登録する事ができます。本項では日付セットを kintone アプリに登録する際、各フィールドがどのように変換されるかを説明します。

■変換ルール

| フィールド種類 | | |
|---------|---------|---|
| X-point | kintone | 変換結果 |
| 西暦フィールド | 文字列（1行） | YYYY-MM-DD 形式で変換され、月・日はアプリに登録した日付になります。 |
| | 日付 | YYYY-MM-DD 形式で変換され、月・日はアプリに登録した日付になります。 |
| | 日時 | YYYY-MM-DDTHH:mm:ss.SSS.Z 形式で変換され、月・日はアプリに登録した日付に、時刻部は 09:00.000（日本時間の午前0時）になります。 |
| | その他 | 西暦フィールドに指定された内容のまま登録されます。kintone のフィールドタイプが登録できない形式の場合はエラーが発生します。 |
| 月フィールド | 文字列（1行） | YYYY-MM-DD 形式で変換され、年・日はアプリに登録した日付になります。 |
| | 日付 | YYYY-MM-DD 形式で変換され、年・日はアプリに登録した日付になります。 |
| | 日時 | YYYY-MM-DDTHH:mm:ss.SSS.Z 形式で変換され、年・日はアプリに登録した日付に、時刻部は 09:00.000（日本時間の午前0時）になります。 |
| | その他 | 月フィールドに指定された内容のまま登録されます。kintone のフィールドタイプが登録できない形式の場合はエラーが発生します。 |
| 日フィールド | 文字列（1行） | YYYY-MM-DD 形式で変換され、年・月はアプリに登録した日付になります。 |
| | 日付 | YYYY-MM-DD 形式で変換され、年・月はアプリに登録した日付になります。 |
| | 日時 | YYYY-MM-DDTHH:mm:ss.SSS.Z 形式で変換され、年・月はアプリに登録した日付に、時刻部は 09:00.000（日本時間の午前0時）になります。 |
| | その他 | 日フィールドに指定された内容のまま登録されます。kintone のフィールドタイプが登録できない形式の場合はエラーが発生します。 |

連携を指定する際に kintone 側のフィールドに同じ項目（文字列（1行）、日付、日時の何れか）を指定した場合、値が連結されて登録が行われます。

| | |
|------------------------------------|------------------------|
| 西暦フィールド1 (yearfield1) [page-1] | 日付セット(文字列) (dateset) |
| 月フィールド1 (monthfield1) [page-1] | 日付セット(文字列) (dateset) |
| 日フィールド1 (dayfield1) [page-1] | 日付セット(文字列) (dateset) |
| 曜日フィールド1 (weekfield1) [page-1] | --- 連携フィールド --- |

kintone アプリの
同じフィールドコードを指定

※ 曜日フィールドは指定できません。指定は無視されます。

■文字列（1行）フィールドに登録する

日付セット (文字) 2018 06 20 水

日付セット(文字列)

2018-06-20

■日付フィールドに登録する

日付セット (日付) 2016 11 11 金

日付セット (日付)

2016-11-11

■日時フィールドに登録する

日付セット (日時) 2020 07 20 月

日付セット (日時)

2020-07-20 9:00

5.1.7. コメントの追加

kintone に書類の登録、更新を行う際、X-point の書類に記入されているコメントを kintone のレコードに追加します。

注意が必要なポイントは、書類に記入されたコメント全てが常に全件追加されることです。その為、レコードの追加以外に更新も行うケースでは同じコメントが複数追加される事に注意してください。

注意が必要なパターンは次のような流れで動作するようなケースです。

- ① コメント A、B が記入された書類を kintone にレコード追加。
- ② X-point 上で書類にコメント C を記入。
- ③ 承認完了時に書類を kintone にレコード更新。

承認完了となる際に、コメント A、B、C が kintone レコードに追加されますが、A、B は kintone にレコードが追加された①の時点でコメント追加されているため、③のタイミングで kintone 上のコメントは、A、B、A、B、C の順で計 5 件の登録になります。

指定は「kintone 連携」画面の「コメントの追加」をチェックして行います。

X-pointからkintoneアプリへのレコード追加・更新タイミングを指定します。

| 連携先kintone情報 | レコード追加・更新タイミング | 備考 | 操作 |
|--|---|--|----|
| ドメイン devettfsn.cybozu.com | 「kintoneレコードID」書類内保存フィールド レコードID (kintone_id) [page-1] ▼ <input type="checkbox"/> 添付ファイルの更新 添付ファイル (attached_file) ▼ | | |
| アプリID 4 | <input checked="" type="checkbox"/> コメントの追加 | | |
| APIトークン 9fr9VE6umQloKph4llgCIDafsqT5XcvSfNoB! | <input type="checkbox"/> 書類削除と連動 <input type="checkbox"/> 完了書類のPDFを添付 押印済みPDF (pdf_document) ▼ | | |
| ゲストID 0 (未指定の場合は0) | <input checked="" type="checkbox"/> 申請 <input type="checkbox"/> 承認1 <input type="checkbox"/> 完了書類修正 | 登録 管理者 (2020/10/26 09:54:09) 更新 管理者 (2020/10/26 09:57:43) | |

確認 フィールド指定 保存

！ 注意事項

kintone に追加されるコメントの投稿ユーザーは「Administrator」になります。この動作は連携で使用する kintone REST API の仕様に基づきます。その為、今後 kintone REST API の仕様変更等が発生した際に動作が変わる可能性がありますので御注意願います。詳細は「cybozu developer network」(<https://developer.cybozu.io/>)を確認して下さい。

5.1.8. 書類削除時の動作を指定

X-point に登録されている書類を削除した際に、kintone アプリの登録レコードも削除するか否かの指定になります。

【削除が可能なレコードの条件】

- ① 連携先 kintone アプリの API トークンのアクセス権に「削除」が有効にされていること。
- ② X-point の書類上に kintone アプリのレコード ID を保存するフィールドがあり、対応するレコード ID が実際に設定されていること。

【削除実行が行われるタイミング】

kintone アプリのレコード ID が保持されている書類を次の条件下で削除した場合にアプリのレコードが削除されます。

- ① 下書き状態になっている書類を承認一覧の「下書き」、あるいは検索一覧の書類プロパティより削除実行したとき。
- ② 承認操作が一度も実行されていない申請書類を検索一覧のプロパティから削除を実行したとき。
- ③ 管理サイトの承認状況画面より書類の kintone アプリのレコード ID が保持されている書類を削除したとき。
- ④ フォームのアクセス権で書類の削除が許可されたユーザーが検索一覧より書類の削除を実行したとき。

指定は「kintone 連携」画面の「書類削除と連動」をチェックして行います。

X-pointからkintoneアプリへのレコード追加・更新タイミングを指定します。

| + kintone連携先アプリを追加する | | レコード追加・更新タイミング | 備考 | 操作 |
|--|---|--|--|----|
| 連携先kintone情報 | | | | |
| ドメイン devettfsn.cybozu.com | | 「kintoneレコードID」書類内保存フィールド レコードID (kintone_id) [page-1] | | |
| アプリID 4 | | <input type="checkbox"/> 添付ファイルの更新 添付ファイル (attached_file) | | |
| APIトークン 9fr9VE6umQloKph4llgClDafsqTSXcvSfNoB! | <input checked="" type="checkbox"/> 書類削除と連動 | <input type="checkbox"/> コメントの追加 押印済みPDF (pdf_document) | | |
| ゲストID 0 (未指定の場合は0) | <input type="checkbox"/> 完了書類のPDFを添付 | <input checked="" type="checkbox"/> 申請 <input type="checkbox"/> 承認1 <input type="checkbox"/> 完了書類修正 | 登録 管理者 (2020/10/26 09:54:09) 更新 管理者 (2020/10/26 09:57:43) | |

確認 フィールド指定

アクセス権で「削除」が指定された
API トークンを指定します

5.1.9. 承認ステップにより更新する内容を変えたい場合の指定

承認の進捗状況に応じて kintone アプリの対象レコードの異なるフィールドを更新する場合は、同一の kintone アプリに対する指定を複数行うことで対処します。

本例で利用する「契約書社内審査」の例では申請以後6回の承認が行われ、次のようなワークフロー処理が想定されます。



- ① kintone から「契約書社内審査」を申請する。
- ② 法務部門が受付承認を行ったタイミングで「審査 No」が採番される。
- ③ 法務部門で申請部門が必要とする書類を添付する。
- ④ 法務部門の上長承認後に「管理コード」設定される審査結果を kintone に反映する。
- ⑤ 申請部門での確認が完了すると必要な書類、審査結果の PDF を kintone 上で確認が可能にする。

このパターンでは、②で“審査番号”を kintone 上で確認可能とすることで審査中を表し、④で“管理コード”を、⑤で必要書類の添付と結果 PDF を kintone に戻すという3段階の異なる更新を行います。

| 連携先kintone情報 | レコード追加・更新タイミング | 備考 | 操作 |
|---|---|---|--|
| ドメイン devettsn.cybozu.com アプリID 4 APIトークン 9fr9VE6umQloKph4llgClDafsqTSXcvSfN ゲストID 0 (未指定の場合は0) <input type="button" value="確認"/> | <input checked="" type="checkbox"/> kintoneレコードID「書類内保存フィールド」 <input type="checkbox"/> 添付ファイルの更新 <input type="checkbox"/> コメントの追加 <input type="checkbox"/> 書類削除と連動 <input type="checkbox"/> 完了書類のPDFを添付 <input checked="" type="checkbox"/> 申請 <input type="checkbox"/> 承認1 <input type="checkbox"/> 確認2 <input checked="" type="checkbox"/> 法務受付3 <input type="checkbox"/> 法務確認4 <input type="checkbox"/> 法務承認5 <input type="checkbox"/> 結果確認6 <input type="checkbox"/> 部門確認7 <input type="checkbox"/> 完了書類修正 <input type="button" value="フィールド指定"/> | レコードID (kintone_id) [page-1] 添付ファイル (attached_file) --- 格納フィールドコード --- 法務部門の受付時に「審査 No」を採番し kintone アプリに採番結果を更新する | <input type="button" value="登録"/> <input type="button" value="更新"/> 管理者 (2020/10/26 09:54:09) 管理者 (2020/10/26 09:57:43) |
| ドメイン devettsn.cybozu.com アプリID 4 APIトークン umQloKph4llgClDafsqTSXcvSfNoB5zrQ ゲストID 0 (未指定の場合は0) <input type="button" value="確認"/> | <input type="checkbox"/> kintoneレコードID「書類内保存フィールド」 <input type="checkbox"/> 添付ファイルの更新 <input type="checkbox"/> コメントの追加 <input type="checkbox"/> 書類削除と連動 <input type="checkbox"/> 完了書類のPDFを添付 <input type="checkbox"/> 申請 <input type="checkbox"/> 承認1 <input type="checkbox"/> 確認2 <input type="checkbox"/> 法務受付3 <input type="checkbox"/> 法務確認4 <input checked="" type="checkbox"/> 法務承認5 <input type="checkbox"/> 結果確認6 <input type="checkbox"/> 部門確認7 <input type="checkbox"/> 完了書類修正 <input type="button" value="フィールド指定"/> | レコードID (kintone_id) [page-1] --- 添付フィールドコード --- --- 格納フィールドコード --- 法務部門の上長承認時に審査結果を表す「管理コード」を kintone に反映する | <input type="button" value="登録"/> <input type="button" value="更新"/> |
| ドメイン devettsn.cybozu.com アプリID 4 APIトークン umQloKph4llgClDafsqTSXcvSfNoB5zrQ ゲストID 0 (未指定の場合は0) <input type="button" value="確認"/> | <input type="checkbox"/> kintoneレコードID「書類内保存フィールド」 <input type="checkbox"/> 添付ファイルの更新 <input type="checkbox"/> コメントの追加 <input type="checkbox"/> 書類削除と連動 <input checked="" type="checkbox"/> 完了書類のPDFを添付 <input type="checkbox"/> 申請 <input type="checkbox"/> 承認1 <input type="checkbox"/> 確認2 <input type="checkbox"/> 法務受付3 <input type="checkbox"/> 法務確認4 <input type="checkbox"/> 法務承認5 <input type="checkbox"/> 結果確認6 <input checked="" type="checkbox"/> 部門確認7 <input type="checkbox"/> 完了書類修正 <input type="button" value="フィールド指定"/> | レコードID (kintone_id) [page-1] 押印済みPDF (pdf_document) 押印済みPDF (pdf_document) 申請部門の確認完了後、添付ファイル、審査結果 PDF を kintone に反映する | <input type="button" value="登録"/> <input type="button" value="更新"/> |

1) 法務部門・受付時の「審査 No」を更新する指定

| | |
|-------------------------------------|--------------------------|
| 申請者 (txtShinseisya) [page-1] | --- 連携フィールド --- |
| 審査No (txtNo) [page-1] | 審査No (txtInspectionno) |
| 顧客コード (txtKokyakucode) [page-1] | --- 連携フィールド --- |

2) 法務部門・上長承認時の「管理コード」を更新する指定

| | |
|-----------------------------------|------------------------|
| 完了希望日 (txtKiboubi) [page-1] | --- 連携フィールド --- |
| 管理コード (txtKanricode) [page-1] | 管理コード (txtKanricode) |
| 顧客名 (txtKokyakumei) [page-1] | --- 連携フィールド --- |

3) 申請部門・上長確認完了（ワークフロー終了）時に「契約書番号」、「契約書名」を更新する指定

| | | | |
|--------|--------------------------------------|--------|---------------------------|
| group1 | 契約書番号 (txtKeiyakusyono) [page-1] | group1 | 契約書番号 (txtKeiyakusyono) |
| | 契約書名 (txtKeiyakusyomei) [page-1] | | 契約書名 (txtKeiyakusyomei) |

5.2. JavaScript を利用して kintone にレコードを追加・更新する

書類入力フォームに JavaScript を記述して X-point から kintone にレコードを追加・更新する場合の手順を解説します。連携動作の確認作業は X-point 上にフォームをアップロードしてから行うため、連携用 JavaScript の組み込みは eFormMaker で、動作確認は X-point で行います。

なお、JavaScript を利用した連携ではフォームナビボタンの操作が必須になるため、ナビボタンを操作できないケースでは連携動作しません。具体的には、採番コンポーネントの採番結果を kintone に送る（ナビボタンの動作後に採番されるため）、管理者スクリプトなど管理者による操作では連携しないなどが制約事項にあたります。

- ※ 管理サイトの画面設定による連携を推奨しています。
- ※ kintone アプリの設定で API トークンに「レコード追加」、「レコード編集」が許可される必要があります。
- ※ kintone の登録レコードを更新する場合の手順も一部の指定を除き追加と同様の手順になります。

5.2.1. ナビゲータに「レコード追加」処理を追加

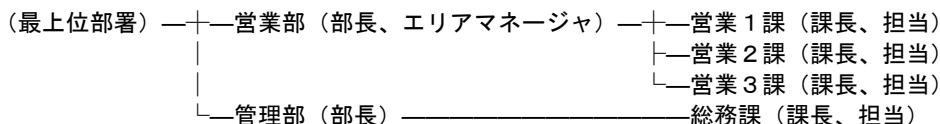
連携を行うフォームを eFormMaker で編集します。追加（更新）処理は「フォーム設定」の「xpPrepare 関数 JavaScript」に記述します。本項では特定の承認ステップで【承認】操作を行った際にレコード追加を行う処理を記述します。

■ 作成内容

「契約書・社内審査」フォームの「法務・上長承認」が処理されるタイミングで kintone アプリへのレコード追加・更新を行います。承認ルートとして図に示す流れを想定します。

| ■ 契約書社内審査 (KeiyakuShinsa) 更新: 2020/10/30 14:34:28 | | | | | | | | | | |
|---|-----------------------------------|------------------------|------------------------|------------------------|------------------------|------------------------|------------------------|------------------------|------------------------|---------------|
| 参照 共通承認ルート参照なし | | | | | | | | | | |
| 条件 | | | | | | | | | | |
| 名称 | 申請STEP | STEP 1 | STEP 2 | STEP 3 | STEP 4 | STEP 5 | STEP 6 | STEP 7 | 回復 | |
| 担当者 | 全ユーザ利用可能 部長 エリアマネージャー 課長 | 部長 エリアマネージャー 課長 | 総務課 (部長) | 総務課 (部長) | 総務課 (部長) | 管理部 (部長) | 申請者 | 部長 エリアマネージャー 課長 | 社長 | 該当者は設定されていません |
| 権限 | 回復 添付 | 回復 編集 却下 差し戻し 添付 | 回復 編集 却下 差し戻し 添付 | 回復 編集 却下 差し戻し 添付 | 回復 編集 却下 差し戻し 添付 | 回復 編集 却下 差し戻し 添付 | 回復 編集 却下 差し戻し 添付 | 回復 編集 却下 差し戻し 添付 | 回復 編集 却下 差し戻し 添付 | |
| 自動採番 | - | - | - | - | - | - | - | - | - | |
| 強管理 | - | - | - | - | - | - | - | - | - | |
| 自動結果 | - | - | - | - | - | - | - | - | - | |
| ステップ承認条件 | AND | AND | AND | AND | AND | AND | AND | AND | | |
| 該当者なしの場合 | スキップする | スキップする | スキップする | スキップする | スキップする | スキップする | スキップする | スキップする | | |

【部署の構成】

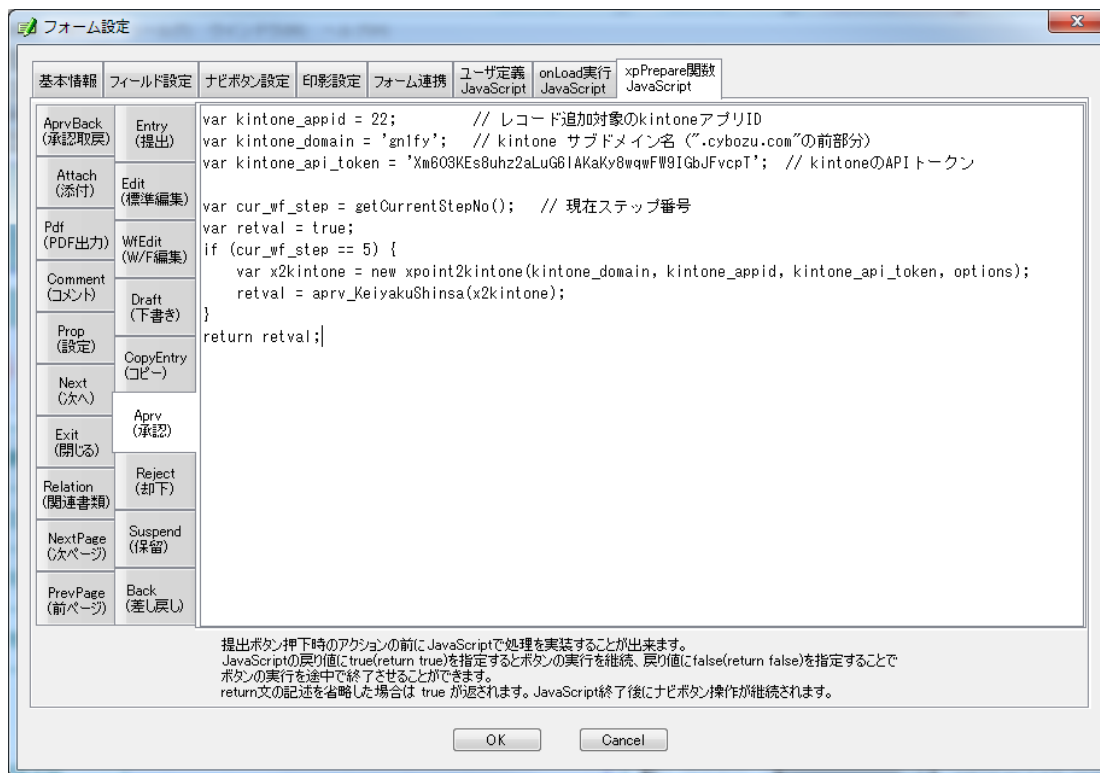


【連携内容】

- 1) 【承認】 ボタンに連携動作を設定し第4ステップで kintone 連携を動作させる処理を記述。
 - ※ 第4ステップは「契約書・社内審査」で法務・上長承認が実行されるステップです。
- 2) 連携動作の主要処理はフォームの「ユーザー定義 JavaScript」に記述。
 - ※ 本項は送信処理を関数 (function) として定義するため「ユーザー定義 JavaScript」を利用します。function に記述しない場合は全て「xpPrepare 関数 JavaScript」に記述しても問題ありません。

■【承認】 ボタンの処理

JavaScript は、eFormMaker の「フォーム設定」 > 「xpPrepare 関数 JavaScript」 > 「Aprv（承認）」に記述します。



【承認】 ナビボタンで行う処理は次の 2 点になります。

- 1) 連携先に関する情報の初期設定を行う。
- 2) 承認ステップ 4 で【承認】 を実行したタイミングで kintone へのレコード更新処理を実行する。

記述されている JavaScript 部分を拡大すると次のようになります。

```

var kintone_appid = 22; // レコード追加対象の kintone アプリ ID
var kintone_domain = 'cbkintone'; // kintone サブドメイン名 (".cybozu.com"の前部分)
var kintone_api_token = 'Xm603KEs8uhz2aLuG6IAKaKy8wqfFW91GbJFvcpT'; // kintone の API トークン
var retval = true;
var cur_wf_step = getCurrentStepNo(); // 現在ステップ番号
if (cur_wf_step == 4) {
    var x2kintone = new xpoint2kintone(kintone_domain, kintone_appid, kintone_api_token);
    retval = aprv_KeiyakuShinsa(x2kintone);
}
return retval;

```

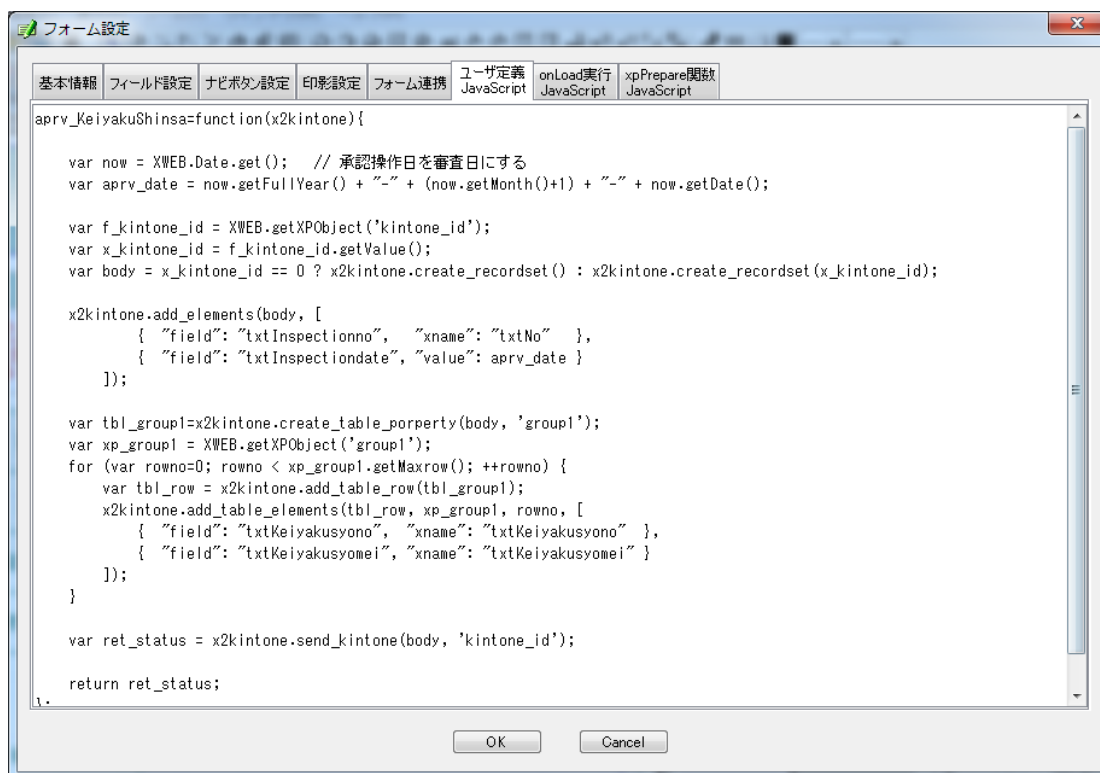
赤枠で囲まれていない部分は、承認ステップ 4 でのみレコード追加が行われるようにするための IF 文になります。

レコード内容の送信を行う “aprv_KeiyakuShinsa” の戻り値を return 文で返すことにより、送信エラーが発生した際に【承認】の動作を中断するようにしています。

※ 連携パラメータ設定、連携用インスタンス“x2kintone”生成は「ユーザ定義 JavaScript」で行っても問題ありません。

■レコード追加の本体処理

レコード追加の本体処理 “arprv_KeiyakuShinsa” は eFormMaker の「フォーム設定」>「ユーザ定義 JavaScript」に記述します。



プログラムで行うことは次の2点になります。

- 1) X-point で採番された「審査 No」、【承認】を実行した日付を kintone のレコードに反映する。
- 2) 「契約書・社内審査」書類の表形式部分「契約書番号」、「契約書名」の3行分を kintone に戻し更新する。

レコード追加の本体処理は、書類内容の値収集、書類内容送信 & 送信後処理の2段階に分けることができます。

全体を処理するプログラム例の後、内容を部分毎に解説します。

■作成する JavaScript の全体

```
aprv_KeiyakuShinsa=function(x2kintone) {  
    var now = XWEB.Date.get(); // 承認操作日を審査日にする  
    var aprv_date = now.getFullYear() + "-" + (now.getMonth()+1) + "-" + now.getDate();  
  
    var f_kintone_id = XWEB.getXPObject('kintone_id');  
    var x_kintone_id = f_kintone_id.getValue();  
    var body = x_kintone_id == 0 ? x2kintone.create_recordset() : x2kintone.create_recordset(x_kintone_id);  
  
    x2kintone.add_elements(body, [  
        { "field": "txtInspectionno", "xname": "txtNo" },  
        { "field": "txtInspectiondate", "value": aprv_date }  
    ]);  
  
    var tbl_group1=x2kintone.create_table_porperty(body, 'group1');  
    var xp_group1 = XWEB.getXPObject('group1');  
    for (var rowno=0; rowno < xp_group1.getMaxrow(); ++rowno) {  
        var tbl_row = x2kintone.add_table_row(tbl_group1);  
        x2kintone.add_table_elements(tbl_row, xp_group1, rowno, [  
            { "field": "txtKeiyakusyono", "xname": "txtKeiyakusyono" },  
            { "field": "txtKeiyakusyomei", "xname": "txtKeiyakusyomei" }  
        ]);  
    }  
  
    var ret_status = x2kintone.send_kintone(body, 'kintone_id');  
  
    return ret_status;  
};
```

■書類内容の値収集

```
var now = XWEB.Date.get(); // 承認操作日を審査日にする  
var aprv_date = now.getFullYear() + "-" + (now.getMonth()+1) + "-" + now.getDate();
```

承認ステップ4の操作日を kintone の「審査日」（日付型）に設定するため、JavaScript で“YYYY-MM-DD”形式の文字列を作成します。現在の日時を取得する際「new Date()」ではなく「XWEB.Date.get()」という関数を利用しています。この関数を利用して X-point サーバーに設定されている日時を取得します。サーバーの日時を取得するのはブラウザが動作している PC 等で日時が正しくない状態においても正しい日付が得られるようにするものです。

なお、「審査日」を日時型（DATETIME）で作成している場合は“YYYY-MM-DD”ではなく ISO8601 形式“YYYY-MM-DDTHH:MM:SS.NNNZ”で指定します。kintone 連携では Date 型を ISO8601 に変換する場合の関数が連携インスタンスに組み込まれていますので必要に応じて御利用ください。

【ISO8601 形式変換例】

```
var now = XWEB.Date.get();  
var aprv_datetime = x2kintone.ISODateString(now);
```

※ JavaScript では通常 Date 型を ISO8601 形式に変換する場合、“toISOString()”を利用しますが、X-point はブラウザ動作モードの都合上“toISOString()”を利用できません。

```
var f_kintone_id = XWEB.getXObject('kintone_id');
```

書類に登録されている kintone のレコード ID を格納しているフィールドを取得します。

『kintone アプリから X-point への連携』で作成したフォーム、連携 JavaScript では kintone から書類申請された際に kintone のレコード ID を格納しています。格納したレコード ID は kintone 登録データの更新、削除などの場合に利用することになります。

```
var x_kintone_id = f_kintone_id.getValue(); 新規追加用の雛形 更新用の雛形  
var body = x_kintone_id == 0 ? x2kintone.create_recordset() : x2kintone.create_recordset(x_kintone_id);
```

kintone に送信するレコード情報の雛形を作成します。作成した変数は JSON 型で格納が行われます。

雛形を作成する際に kintone のレコード ID が 0 以外の場合は kintone のレコード ID を引数に指定し、0 の場合は指定しません。

これは kintone に送信するレコード情報に kintone のレコード ID が含まれている場合は“更新”、含まれない場合は“新規追加”になるためです。

- ※ 「フォームの作成」で X-point のフォームに追加した kintone 連携用項目では“kintone_id”を数値フィールドで作成しています。数値フィールドは空欄の場合、getValue() で 0 が返されるため判定に利用しています。
- ※ 本書で利用しているフォーム「契約書・社内審査」は kintone から X-point に申請した書類の情報を kintone に戻すことを前提としているため、新規追加に関する実装は必須ではありません。

```
x2kintone.add_elements(body, [  
  { "field": "txtInspectionno", "xname": "txtNo" },  
  { "field": "txtInspectiondate", "value": aprv_date }  
]);  
X-point フィールド ID  
kintone フィールドコード JavaScript 変数の値をセットする
```

表定義以外の部分のフィールドを送信します。“field”プロパティに指定する値は kintone アプリのフィールドコードになります。“xname”プロパティに X-point フォームのフィールド ID を指定すると該当するフィールド ID の項目に登録されている内容が kintone に登録されます。“value”プロパティに指定した値は kintone に登録される値が value に指定した値になります。本例の場合は審査日の日付文字列が変数 aprv_date に設定されているため、aprv_date の内容が kintone の“txtInspectiondate”に登録されます。

なお、複数枚フォームを扱う場合は“xname”とともに“page”属性を指定します。

【2 ページ目に txtNo がある場合】

```
{ "field": "txtInspectionno", "xname": "txtNo", "page": 2 },
```

この場合、複数枚フォームの 2 ページ目にある“txtNo”フィールドの値が送信されます。ページ番号を変数で扱う場合、変数の内容が undefined になる場合は単一フォームとして処理されますので変数処理を行う場合は注意してください。

ここで「審査 No」、「審査日」以外の項目も含めて kintone に戻す場合の JavaScript も示します。

フォーム上の日付フィールド (txtKiboubi) を得る方法、kintone 側のフィールドがユーザー選択 (txtTantou) と特殊になるため指定方法に工夫されています。

```
var dd = x2kintone.date_xp2kintone( XWEB.getXPObject('txtKiboubi',pgno).getValue() );
if (isNaN(dd)){
  alert('扱う事ができない日付書式です');
  return;
}
```

日付を Date 型に変換

```
var strKiboubi = dd.getFullYear() + "-" + (dd.getMonth()+1) + "-" + dd.getDate();
```

```
x2kintone.add_elements(body, [
  { "field": "txtInspectionno", "xname": "txtNo" },
  { "field": "txtInspectiondate", "value": aprv_date },
  { "field": "txtKokyakucode", "xname": "txtKokyakucode" },
  { "field": "txtKokyakumei", "xname": "txtKokyakumei" },
  { "field": "txtTantou", "value": [ { "code": getLoginUserCode() } ] },
  { "field": "txtKiboubi", "value": strKiboubi },
  { "field": "txtKanricode", "xname": "txtKanricode" },
  { "field": "txtMemo", "xname": "txtMemo" }
]);
```

X-point が扱う日付書式は「****年**月**日**時**分」や「yyyy/mm/dd hh:mm」、「yyyy-mm-dd」のように様々な形式が指定可能です。

そこで“date_xp2kintone”を利用し、これらの書式を可能な限り JavaScript の Date 型に変換します。

変換できない日付の例としては前ゼロ無しの場合に「112」が得られた場合、1/12 であるのか 11/2 であるのかといった場合などの区切り記号が含まれないケースが該当します。

転送する項目部分では、ユーザー選択 (txtTantou) の指定方法が特殊な形式になります。

```
{ "field": "txtTantou", "value": [ { "code": getLoginUserCode() } ] },
```

例ではユーザー選択の値として X-point にログイン中のユーザーコードを指定します。value への指定は kintone API のフィールド指定形式に倣った形式で指定します。ユーザー選択以外では“組織選択”、“グループ選択”、“カテゴリ”、“チェックボックス”、“複数選択”が同様に指定方法になります。指定の方法は本項後半にて説明します。

```

var tbl_group1=x2kintone.create_table_porperty(body, 'group1');
var xp_group1 = XWEB.getXPObject('group1');
for (var rowno=0; rowno < xp_group1.getMaxrow(); ++rowno) {
  var tbl_row = x2kintone.add_table_row(tbl_group1);
  x2kintone.add_table_elements(tbl_row, xp_group1, rowno, [
    { "field": "txtKeiyakusyono", "xname": "txtKeiyakusyono" },
    { "field": "txtKeiyakusyomei", "xname": "txtKeiyakusyomei" }
  ]);
}

```

kintone のテーブルに設定されたフィールドコード

1 行分の格納エリアを確保する

確保した行エリアに設定内容を登録

表部分のフィールド処理方法になります。表グループに指定された行数分繰り返します。

設定するフィールドの考え方は表以外の場合と同様に“field”は kintone のフィールドコード、“xname”は X-point のフィールド ID、“value”は設定値を kintone に送る場合になります。

※ X-point から kintone にレコード追加する場合、送信するテーブルの数に制限はありません。
制限は kintone から X-point に申請する場合のみです。

なお、複数枚フォームで表形式を扱う場合は add_table_elements で“page”属性の指定は行わず、表グループの情報を取得する際にページ番号を指定します。

```
var xp_group1 = XWEB.getXPObject('group1', pgno);
```

これは、表グループがページ情報を持っているためです。

kintone に登録する書類情報が出来上がったら“send_kintone”メソッドで情報を送信します。

```

var ret_status = x2kintone.send_kintone(body, 'kintone_id');
return ret_status;
};

```

送信が成功すると登録された kintone のレコード ID が“kintone_id”に設定されます。レコード ID が不要の場合は、

```
var ret_status = x2kintone.send_kintone(body, body);
```

のみで構いません。

レコード登録の際に特別な処理を行いたい場合は、レコード ID の戻し先ではなく Callback 用の function を指定します。

Callback 用の function を指定した場合、レコード登録後に kintone からの応答が戻った段階で第 2 引数の callback_func が実行されます。“send_kintone” 以降の処理は callback_func の実行後に行われます。

```
var ret_status = false; // send_kintone 実行後のエラー判定値が入る。エラーなし = true、エラーあり = false

var callback_func = function(status, responseText) {
  if((200 <= status && status < 300) || (status == 304)) {
    if (responseText && responseText.length > 0) {
      var kr = JSON.parse(responseText);
      if (kr.id) {
        x_kintone_id = kr.id;
        f_kintone_id.setValue(kr.id); // X-point のフィールドに kintone のデータ ID を設定する
      }
      ret_status = true;
    } else { // kintone 送信後の結果を処理する Callback 処理
      alert("その他の応答:" + status);
    }
  } else {
    if (responseText && responseText.length > 0) {
      var kr = JSON.parse(responseText);
      alert("code : " + kr.code + ", id : " + kr.id + "\n" + kr.message + x2kintone.errorDetail(kr.errors));
    } else {
      alert("その他の応答:" + status);
    }
  }
};

x2kintone.send_kintone(body, callback_func); // 書類内容の送信

return ret_status;
```

なお、responseText に設定される内容は kintone REST API の共通仕様に基づく応答内容になり JSON 形式で返されます。

※ REST API の共通仕様 (<https://cybozudev.zendesk.com/hc/ja/articles/201941754>)

■JavaScript での値送信に「value」プロパティを使う場合の注意

フィールドの内容を value プロパティで指定する場合、kintone 側のフィールドタイプが、チェックボックス、複数選択、ユーザー選択、組織選択、グループ選択 の場合は記述方法に注意が必要です。

【通常の場合】

value に文字列を指定します。

```
{ "field": "(フィールドコード)", "value": "(設定する値・文字列)" }
```

【チェックボックス、複数選択】

[]で囲み設定する選択肢を指定します。選択肢が複数ある場合は“,”で区切ります。

```
{ "field": "(フィールドコード)", "value": [ "選択肢 1", "選択肢 2" ] }
```

【ユーザー選択、組織選択、グループ選択】

[]で囲み設定するコードを「 { "code": "(設定コード)" } 」に形式で指定します。複数ある場合は“,”で区切ります。

```
{ "field": "(フィールドコード)", "value": [ { "code": "(設定コード)" },
                                           { "code": "(設定コード)" } ] }
```

※ cybozu developer network「フィールド形式」

(<https://cybozudev.zendesk.com/hc/ja/articles/202166330>)

5.2.2. 添付ファイルの送信

添付ファイルの送信は書類内容を送信する前段階で行います。これはファイルを kintone に登録した際に発行される keycode を登録するレコードに含める必要があるためです。添付ファイルを送信する “send_attach_files” を実行すると kintone に送信するデータ内に登録された際に発行されたキー自動的に設定されます。

なお、ファイルを kintone にアップロードしたのちにレコード追加を中断すると、ファイルのみが kintone にのこることになりますが、kintone ではどのレコードからも参照されない添付ファイルは3日ほど経過すると自動的に削除される仕様であるため、書類本体の登録を中断しても問題となる可能性はありません。

```
// 添付ファイルをアップロードし kintone の格納キーを取得する。  
var responseText = x2kintone.send_attach_files(body, "attached_file");  
if (responseText) {  
    var kr = JSON.parse(responseText);  
    alert("code : " + kr.code + " , id : " + kr.id + "¥n" + kr.message + x2kintone.errorDetail(kr.errors));  
    return false;  
}  
  
var ret_status = x2kintone.send_kintone(body, 'kintone_id');
```

“send_attach_files” の第2引数 (“attached_file”) は、kintone の添付ファイルフィールドのフィールドコードになります。登録後に戻される応答は kintone の REST API からのレスポンスと同一形式で kintone から返されます。正常時の応答は undefined になる点も kintone と同様です。

- ※ REST API の共通仕様 (<https://cybozudev.zendesk.com/hc/ja/articles/201941754>)
- ※ 添付ファイルの指定を行うと kintone のレコードに登録されている添付ファイルの内容は全て、X-point から送信した内容に置き換えられます。X-point からの送信情報に添付フィールドに係わる項目が指定されていない場合は、kintone 側の添付状態に影響を与えません。
- ※ 第2引数は “attached_file” とし、kintone アプリ側のフィールドコードもこれに合わせる必要があります。

5.2.3. コメントの送信

kintone のコメントはレコード ID を指定して登録する必要があるためレコードの登録後に行います。

送信用のメソッド “send_comments” にレコード ID を指定して呼び出すことで、書類に記載されているコメント全件が kintone に送信されます。

```
var ret_status = x2kintone.send_kintone(body, 'kintone_id');

var x_kintone_id = f_kintone_id.getValue();      ← kintone からレコード ID を得たことを判断する値を得る
if (ret_status && x_kintone_id != 0) {
  var responseText = x2kintone.send_comments(x_kintone_id);
  if (responseText) {
    var kr = JSON.parse(responseText);
    alert("code : " + kr.code + ", id : " + kr.id + "¥n" + kr.message + x2kintone.errorDetail(kr.errors));
  } else {
    ret_status = true;
  }
}
```

登録後に戻される応答は kintone の REST API からのレスポンスと同一形式で kintone から返されます。

正常時の応答は undefined になる点も kintone と同様です。

※ REST API の共通仕様 (<https://cybozudev.zendesk.com/hc/ja/articles/201941754>)

5.2.4. ゲストスペースを利用するアプリへの登録

kintone アプリがゲストスペースに配置されている場合は連携クラスの初期化パラメータにオプション項目を追加します。

```
var kintone_appid    = 12;           // レコード追加対象の kintone アプリ ID
var kintone_subdomain = 'cbkintone'; // kintone サブドメイン名 (".cybozu.com" の前部分)
var kintone_api_token = 'diVUoVQDgnnU7r01v4sFyfUiGnX0ResLGgOfIk1s'; // kintone の API トークン
var options = { "guest" : "15" }; // オプション項目でゲストスペースの ID を指定する

var cur_wf_step = getCurrentStepNo(); // 現在ステップ番号

var retval = true;
if (cur_wf_step == 4) {
  var x2kintone = new xpoint2kintone(kintone_subdomain, kintone_appid, kintone_api_token, options);
  retval = aprv_KeiyakuShinsa(x2kintone);
}
return retval;
```

オプション項目の “guest” プロパティにゲストスペースの ID を指定します。

5.2.5. Administrator 以外で登録する

kintone へのレコード追加を API トークン による認証で行うとレコードの追加・更新は全て kintone の Administrator ユーザーにより行われます。Administrator 以外のユーザーで行う場合はオプション項目の `userid`、プロパティに kintone ユーザー名、`passwd` プロパティに平文で指定ユーザーのパスワードを設定し連携クラスのインスタンスを作成します。

```
var kintone_appid      = 12;           // レコード追加対象の kintone アプリ ID
var kintone_subdomain = 'cbkintone'; // kintone サブドメイン名 (".cybozu.com"の前部分)
var kintone_api_token = 'diVUoVQDgnnU7r01v4sFyfUiGnX0ResLGgOfIk1s'; // kintone の API トークン
var options = { "userid" : "u001", "passwd" : "pass001" };

var cur_wf_step = getCurrentStepNo(); // 現在ステップ番号

var retval = true;
if (cur_wf_step == 4) {
    var x2kintone = new xpoint2kintone( kintone_subdomain, kintone_appid, kintone_api_token, options );
    retval = aprv_KeiyakuShinsa(x2kintone);
}
return retval;
```

5.2.6. ドメイン名が“cybozu.com”以外の場合

kintone が使用する FQDN は通常「(サブドメイン).cybozu.com」になりますが、「(サブドメイン).kintone.com」などドメイン名が異なる場合、オプション項目を利用してドメイン名を変更します。

```
var kintone_appid      = 12;           // レコード追加対象の kintone アプリ ID
var kintone_subdomain = 'cbkintone'; // kintone サブドメイン名 (".cybozu.com"の前部分)
var kintone_api_token = 'diVUoVQDgnnU7r01v4sFyfUiGnX0ResLGgOfIk1s'; // kintone の API トークン
var options = { "domain" : "kintone.com" }; // ドメイン名を kintone.com にする

var cur_wf_step = getCurrentStepNo(); // 現在ステップ番号

var retval = true;
if (cur_wf_step == 4) {
    var x2kintone = new xpoint2kintone( kintone_subdomain, kintone_appid, kintone_api_token, options );
    retval = aprv_KeiyakuShinsa(x2kintone);
}
return retval;
```

上記例の場合、オプション項目の指定により「cbkintone.kintone.com」になります。オプション項目無しでは「cbkintone.cybozu.com」です。

5.3. kintone の登録レコードを削除する

X-point のフォームナビゲータから kintone に登録されているレコードを削除する場合の手順を解説します。

- ※ X-point から kintone のレコードを削除するためには“アクセス権”で“レコード削除”が許可された kintone アプリの API トークンが必要になります。
- ※ X-point に登録されている書類を検索、承認管理などの画面操作と連動して kintone の登録レコードを削除する場合は、管理サイトでの設定を使用して下さい。

5.3.1. ナビゲータに「レコード削除」処理を追加

削除を行うナビゲータボタンに処理を追加します。

```
var kintone_appid = 12; // 対象の kintone アプリ ID 初期設定パラメータ
var kintone_subdomain = 'cbkintone'; // kintone サブドメイン名 (".cybozu.com" の前部分)
var kintone_api_token = 'diVUoVQDgnnU7r01v4sFyfUiGnX0ResLGg0fIk1s'; // kintone API トークン (レコード削除)

var x2kintone = new xpoint2kintone( kintone_subdomain, kintone_appid, kintone_api_token );
var recid = XWEB.getXPObject('kintone_id').getValue();
if (recid !=0) { // レコード ID が登録されている場合に削除を実行する
  var callback_func = function(status, responseText) {
    if((200 <= status && status < 300) || (status == 304)) {
      if (responseText && responseText.length > 0) {
        var kr = JSON.parse(responseText);
        XWEB.getXPObject('kintone_id').setValue(0);
        alert("kintone の登録を削除しました"); kintone 送信後の結果を処理する Callback 処理
      }
    } else { // 通信エラー他 (kintone のエラー応答を含む)
      if (responseText && responseText.length > 0) {
        var kr = JSON.parse(responseText);
        alert("code : " + kr.code + ", id : " + kr.id + "¥n" + kr.message + x2kintone.errorDetail(kr.errors));
      }
    }
  };

  x2kintone.delete_kintone(recid, callback_func); ←削除の実行
}
```

登録後に戻される応答は kintone の REST API からのレスポンスと同一形式で kintone から返されます。

正常時の応答は undefined になる点も kintone と同様です。

- ※ REST API の共通仕様 (<https://cybozudev.zendesk.com/hc/ja/articles/201941754>)

5. 4. kintone アプリの登録レコードをマスタとして利用する

kintone アプリに登録されているレコード情報をマスタ参照コンポーネントで利用する手順を説明します。マスタとして利用できるように設定することで、マスタ参照コンポーネント、マスタ検索での利用が可能になります。登録はアプリに登録された内容を直接利用する方法と、X-point の固有マスタに取り込みを行ったのちに利用する方法の2パターンがあります。

5. 4. 1. kintone 連携マスタの作成

連携サービスを導入している場合、マスタ管理メニューに【kintone マスタ作成】が表示されます。

画面からは、X-point 内にデータを持たず kintone アプリのレコードを X-point から直接利用するための参照登録を行います。

直接参照の場合、レコードの検索等で kintone の機能を利用することになるため、kintone の動作仕様による検索仕様の違いが発生します。詳細は、kintone developer site 等を参考にして下さい。

| kintone連携マスタ作成 | |
|------------------------|---|
| 基本情報 | |
| マスタコード ① 半角英数30文字以内 | KTMASTER |
| マスタ名称 | kintone顧客マスタ |
| 備考 | |
| kintoneアプリ情報 | |
| ドメイン名 | kintone.cybozu.com ① ドメイン名 (必須) cybozu.comは省略可能です |
| アプリID | 21 |
| APIトークン | OguVy7CjamClnbq5Mb5HkvLbg9ClHmvjDIaewF08 ① アクセス権「レコード閲覧」、「アプリ管理」が有効なAPIトークンを指定します |
| ゲストスペースID | 0 |

マスタ名称を登録します

kintone アプリの参照情報を登録します

| 項目名 | 用途 |
|------------|---|
| マスタコード | 登録コード |
| マスタ名称 | 登録名称 |
| 備考 | マスタの説明などを入力します |
| ドメイン名 | 参照するアプリが登録されている kintone の FQDN を指定します |
| アプリ ID | 参照するアプリの ID を数値で指定します |
| API トークン | アプリのレコード内容が参照 (閲覧権限あり) できる API トークンを指定します |
| ゲストスペース ID | アプリがゲストスペースに配置される場合、ゲストスペースの ID を指定します |

5.4.2. カスタム領域へのインポート

kintone のアプリのレコードを X-point の固有マスタに全て取り込み利用する方法です。

アプリを直接利用する場合に検索実行などで X-point と kintone の検索仕様の違いによる影響を避ける場合に利用します。

5.4.2.1. カスタム領域への TABLE 作成

連携サービスを導入している場合、マスタ管理の上部メニューに表示される【ユーザ固有マスタ】をクリックすると、サブメニューに【kintone から作成】が表示されます。

画面からは、X-point 内にデータを持たず kintone アプリのレコードを X-point から直接利用するための参照登録を行います。

■ マスタ名とマスタ利用する kintone アプリ情報の指定

必要な項目を指定したのち【取得】をクリックします。

マスタ管理 → マスター一覧 → 簡易マスタ作成 → kintoneマスタ作成 → ユーザ固有マスタ → タスクログ

ユーザ固有マスタ作成 + 作成 取込 + kintoneから作成 インポートバッチ登録

基本情報

テーブル名 kintoneCustomer

項目を取得するkintoneアプリ情報を指定します

| | |
|-----------|--|
| ドメイン名 | devettfsn.cybozu.com |
| アプリID | 4 |
| APIトークン | 9fr9VE6umQloKph4llgClDafsqTSXcvSfNoB5zrQ |
| ゲストスペースID | 0 (未指定の場合は0) |

取得

| 項目名 | 用途 |
|------------|---|
| テーブル名 | カスタム領域に作成する TABLE 名 |
| ドメイン名 | 参照するアプリが登録されている kintone の FQDN を指定します |
| アプリ ID | 参照するアプリの ID を数値で指定します |
| API トークン | アプリのレコード内容が参照（閲覧権限あり）できる API トークンを指定します |
| ゲストスペース ID | アプリがゲストスペースに配置される場合、ゲストスペースの ID を指定します |

■ データ型、桁数、主キー、索引の指定

アプリの指定内容に誤りが無ければ【取得】後、アプリのフィールドコードがフィールドとして追加されます。

通常のユーザー固有マスタ作成と同様にデータ型、桁数、主キー、索引の有無を指定し、最後に【作成】をクリックします。

マスタ管理 → マスター一覧 → 簡易マスタ作成 → kintoneマスタ作成 → ユーザ固有マスタ → タスクログ

ユーザー固有マスタ作成 + 作成 取込 + kintoneから作成 インポートバッチ登録

基本情報

テーブル名: kintoneCustomer

項目を取得するkintoneアプリ情報を指定します

ドメイン名: devettfsn.cybozu.com

アプリID: 4

APIトークン: 9fr9VE6umQloKph4llgCIDafsqTSXcvSfNoB5zrQ

ゲストスペースID: 0 (未指定の場合は0)

取得

フィールドを追加

| NO | フィールド名 (フィールドID) | データ型 | 桁数 | 主キー | 索引 | 項目削除 |
|----|------------------|------|----|-------------------------------------|--------------------------|------|
| 1 | txtKokyakucode | 文字列型 | 64 | <input checked="" type="checkbox"/> | <input type="checkbox"/> | |
| 2 | txtKiboubi | 文字列型 | 64 | <input type="checkbox"/> | <input type="checkbox"/> | 🗑 |
| 3 | txtMemo | 文字列型 | 64 | <input type="checkbox"/> | <input type="checkbox"/> | 🗑 |
| 4 | txtInspectionno | 文字列型 | 64 | <input type="checkbox"/> | <input type="checkbox"/> | 🗑 |
| 5 | txtShinseibi | 文字列型 | 64 | <input type="checkbox"/> | <input type="checkbox"/> | 🗑 |
| 6 | docid | 数値型 | 12 | <input type="checkbox"/> | <input type="checkbox"/> | 🗑 |

フィールド名に半角英数字以外を利用している場合は、フィールド名を半角英数字・記号（“_”）に書き換えてください。

■ ユーザ固有マスタの取り込み

作成した固有マスタを X-point から参照できるよう取り込みを行います。

マスタ管理 → マスター一覧 → 簡易マスタ作成 → kintoneマスタ作成 → ユーザ固有マスタ → タスクログ

ユーザー固有マスタ取込 + 作成 取込 + kintoneから作成 インポートバッチ登録

検索条件

スキーマ: 標準のスキーマ (XPCUSTOM) 閲覧権限のある全てのスキーマ

テーブル名: _____ を含む

30件 ずつ 検索

28件中 1-28表示

| <input type="checkbox"/> | DB上のマスタ名 | X-point上の識別名 | 備考 | 取込済 | |
|--------------------------|--------------------------------|--------------------------------|----|-----|---|
| <input type="checkbox"/> | abcdefghijklmnopqr stuvwxyz | abcdefghijklmnopqrstu vwxyz | | | ↓ |
| <input type="checkbox"/> | kintonecustomer | kintonecustomer | | | 🗑 |

5.4.2.2. 固有マスタへのレコードインポート

固有マスタに kintone アプリのレコードを取り込みます。取得件数の制限は出来ません。

- kintone アプリのレコードを取り込む固有マスタテーブルの【インポート】アイコンをクリックします

The screenshot shows the 'Master Management' page. At the top, there are navigation links: 'マスター一覧', '簡易マスタ作成', 'kintoneマスタ作成', 'ユーザ固有マスタ', and 'タスクログ'. Below this is a table titled 'マスター一覧' with columns: 'No', 'マスタ名', '項目数', '備考', and '操作'. The first row shows '1', 'kintonecustomer', '-', and an '操作' column containing an 'Import' icon (represented by a square with an arrow pointing right). A red box highlights this icon, with a red arrow pointing to it from the text 'インポートアイコン'.

- kintone アプリからレコードを取り込む指定を行います

The screenshot shows the 'User-specific Master Import' configuration page. It has a breadcrumb trail: 'マスター一覧' → '簡易マスタ作成' → 'kintoneマスタ作成' → 'ユーザ固有マスタ' → 'タスクログ'. Below the breadcrumb are buttons: '+ 作成', '取込', '+ kintoneから作成', and 'インポートバッチ登録'. The page is divided into sections: '基本情報', 'インポート定義', 'インポート内容', and 'データ'. In the 'インポート定義' section, 'カンマ区切' and '先頭行をフィールド名とする' are selected and highlighted with red boxes. In the 'データ' section, the 'kintoneアプリ' option is selected, and its details (Domain name, App ID, API Token, Guest Space ID) are shown. A red box highlights this entire section, with a red arrow pointing to it from the text 'kintone アプリへの接続情報'. At the bottom, there is a blue button labeled '次へ'.

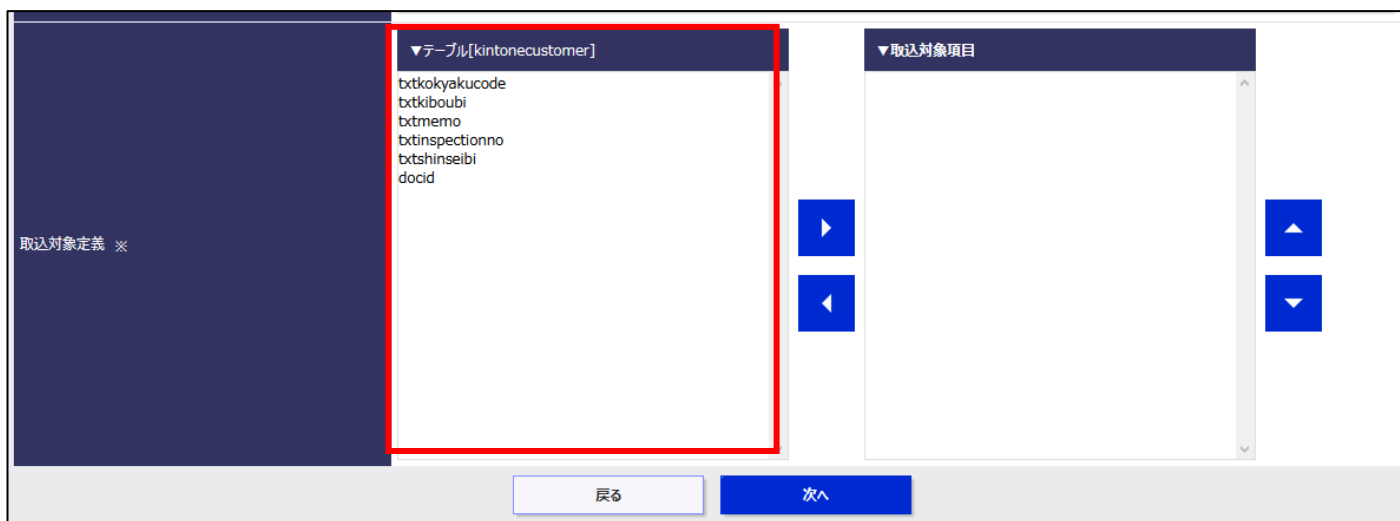
| 項目名 | 内容 |
|--------------|---|
| 対象マスタ | 取り込みを行う固有マスタ名を指定 |
| インポート定義 | 登録済みの定義を利用する場合に選択します |
| 書式 | 【カンマ区切】を選択、【先頭行をフィールド名とする】をチェック |
| インポート内容 | 運用に合わせて指定して下さい |
| インポートできない・・・ | 運用に合わせて指定して下さい |
| ドメイン名 | 参照するアプリが登録されている kintone の FQDN を指定します |
| アプリ ID | 参照するアプリの ID を数値で指定します |
| API トークン | アプリのレコード内容が参照（閲覧権限あり）できる API トークンを指定します |
| ゲストスペース ID | アプリがゲストスペースに配置される場合、ゲストスペースの ID を指定します |

■ kintone アプリからレコードを取り込む指定を行います

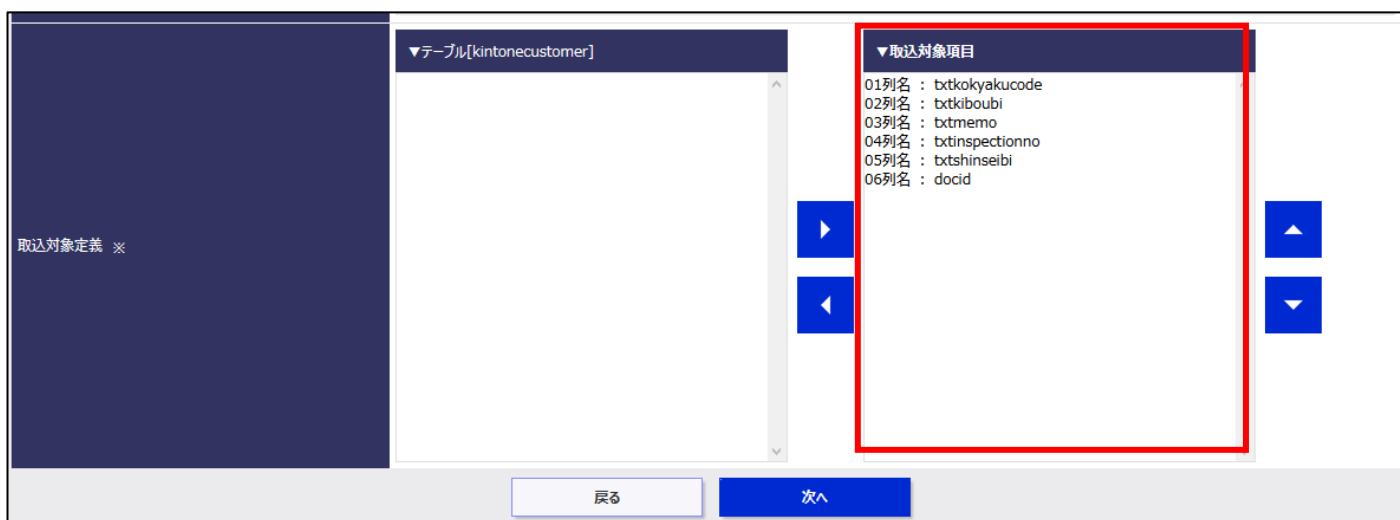
「データビュー」項目に登録レコードの先頭部分、及びアプリのフィールドコードが一覧に表示されることを確認します。

エラーが表示されている場合は、【戻る】ボタンで前画面に戻り kintone アプリの情報を確認、修正して下さい。

書式は「カンマ区切り」、「先頭行をフィールド名とする」から変更しません。



固有マスタに取り込む項目を選び【取り込み対象項目】に設定します。



※ 指定方法の詳細は「X-point 管理者マニュアル」を参照願います。

- 必要に応じ項目の置換定義を行います。

- ※ 指定方法の詳細は「X-point 管理者マニュアル」を参照願います。

- 設定内容を確認し設定の保存（定義内容を保存）、インポートの実行を行います。

| 基本情報 | | | | | |
|----------------------------|--|------|-----|---------|-----------|
| 対象マスタ ※ | kintonecustomer | | | | |
| インポート定義 ※ | | | | | |
| 区切り文字 ※ | CSV区切 | | | | |
| 先頭行 ※ | 先頭行をフィールド名とする | | | | |
| インポート内容 ※ | データを完全に置き換える | | | | |
| インポートできないレコードが検出された場合の動作 ※ | インポートを継続 (インポートできないレコードを除外する) | | | | |
| ドメイン名 ※ | devetfsn.cybozu.com | | | | |
| アプリID ※ | 4 | | | | |
| APIトークン ※ | 9fr9VE6umQloKph4llgClDafsqTSXcvSfNoB5zrQ | | | | |
| ゲストスペースID ※ | 0 | | | | |
| 取込対象・置換定義 | | | | | |
| 項目順 | DB列名 | 置換定義 | 初期値 | ゼロパディング | データ不正時の扱い |
| 1 | btkkoyakucode | | | | エラーとする |
| 2 | btkiboubi | | | | エラーとする |
| 3 | btxmemo | | | | エラーとする |
| 4 | btxinspectionno | | | | エラーとする |
| 5 | btxshinseibi | | | | エラーとする |
| 6 | docid | | | | エラーとする |

インポートバッチ機能を利用する場合は定義の保存が必須になります。

■ マスタ登録中の画面

【登録中の画面】

🏠 マスタ管理 → マスター一覧 → 簡易マスタ作成 → kintoneマスタ作成 → ユーザ固有マスタ → タスクログ

マスター一覧 + 作成 📄 取込 + kintoneから作成 🔄 インポートバッチ登録
 処理済み 0 件 正常 0 件 / エラー 0 件 経過時間：秒

固有マスタに登録中です。登録処理が完了するまで暫くお待ちください。

中止

⚠️ 固有マスタインポート
 別の画面に移動してもマスタインポート処理は継続され全てのレコードをインポートします。
 【中止】ボタンを押すとマスタインポートが中止されます。処理済のレコードはインポート開始前に指定する「インポートできないレコードが検出された場合の動作」に従います。

【全件の登録が正常に行われた場合】

マスター一覧 + 作成 📄 取込 + kintoneから作成 🔄 インポートバッチ登録
 全 10 件 正常 10 件 / エラー 0 件 所要時間：0.202 秒

固有マスタへの登録が完了しました。

【登録データにエラーが発生した場合】

マスター一覧 + 作成 📄 取込 + kintoneから作成 🔄 インポートバッチ登録
 全 21 件 正常 0 件 / エラー 21 件 所要時間：0.21 秒

固有マスタへの登録が完了しました。

| 行数 | データ内容 | 結果 | メッセージ |
|----|--|----|--------------------------|
| 1 | "txtKokyakucode","txtKanricode","txtInspectiondate","txtKokyakumei","txtTantou","txtKiboubi","txtMemo","txtInspecti onno","txtShinseibi","txtShinseisya","checkbox_0" | 失敗 | 取込対象列数(2)とデータ数(1)が一致しません |
| 2 | "C001","K001","2020-10-01","顧客太郎","戸田 翔馬","2020-10-01","審査します | 失敗 | 取込対象列数(2)とデータ数(1)が一致しません |

5.4.2.3. インポートバッチの利用

固有マスタへのインポートを行う際、インポートの指定内容を“インポート定義”として保存した場合、インポートバッチを利用可能になります。kintone アプリからのレコード取り込みにインポートバッチを活用することで、固有マスタの内容を定期的に最新状態に更新することが可能になります。

■ インポートバッチ登録一覧

バッチ一覧の表示にインポート元が kintone アプリであるか、CSV ファイルであるかの区別はありません。

マスタ名をクリックして登録内容で確認します。

| No | マスタ名 (X-point上の識別名) | インポート定義 | バッチ実行 | 備考 |
|----|---------------------|-------------------------|-------|----|
| 1 | kintoneappmaster2 | kintoneappmaster2Import | 実行しない | |
| 2 | kintonemaster | | 実行しない | |

■ インポートバッチ登録内容

登録画面では最初に「使用するインポート定義」を選択します。

固有マスタインポートの定義を選ぶ

「インポート定義」を選択すると、インポート元になる kintone アプリ、項目の置換定義などが表示されます。
 その後、マスタインポートタスクが動作する際にインポートを実行するか否かを指定して保存します。

| 項目名 | 内容 |
|--------------------|---|
| インポートバッチの実行 | マスタインポートタスクを実行する際に、マスタのインポートを動作させるか否かを指定します |
| マスタ名 | インポートが行われるマスタ名が表示されます |
| 使用するインポート定義 | 保存されている「ユーザ固有マスタインポート」定義を選択します |
| インポート元 kintone アプリ | インポート元になるアプリの情報が表示されます |
| 備考 | メモ欄 |
| インポート定義情報 | 取り込み時のデータ処理方法 |
| 取り込み対象・置換定義 | 処理対象となるフィールドに関する指定 |

5.4.3. マスタ参照コンポーネントの作成

固有マスタに取り込みを行った場合は当然ですが、kintone 連携マスタ (kintone アプリを直接利用する形式) も、通常のマスタ参照コンポーネントと同様にマスタ参照コンポーネントで利用する事が出来ます。

作成したコンポーネントは eFormMaker を利用してフォーム上で指定を行い、マスタ参照ウィンドウ、マスタ検索の動作が可能になります。

フォーム>コンポーネント>マスタ参照ウィンドウの「+」をクリックします。

The screenshot shows the 'atled 管理者' dashboard. The navigation menu on the left includes 'フォーム管理', 'ユーザ', '承認', and 'マスタ'. The main content area is titled 'コンポーネント' and contains a table with the following data:

| コンポーネント | 件数 | 一覧 | 作成 | 備考 |
|------------|----|----|----|---------------------------|
| マスタ参照ウィンドウ | 1 | 🗑️ | + | マスタ参照で利用するウィンドウコンポーネントです。 |
| 自動採番 | 1 | 🗑️ | + | 自動採番コンポーネントです。 |
| 印影 | 3 | 🗑️ | + | 印影コンポーネントです。 |
| 残管理 | 1 | 🗑️ | + | 残管理コンポーネントです。 |

The screenshot shows the configuration page for the 'マスタ参照ウィンドウ' component. The configuration is as follows:

| 基本情報 | | | |
|---------------|---|--|--|
| 名称 | kintone顧客マスタ | カナ | kintoneマスタ |
| マスタ選択 | kintonemaster | 主キー | --- 主キー --- |
| 表示形式 | <input type="radio"/> スクロール式(全件取得) サンプル <input checked="" type="radio"/> ページ切替式(一部取得) サンプル | 絞り込み条件欄 | <input checked="" type="radio"/> 使用しない サンプル <input type="radio"/> 使用する サンプル |
| 表示行数 | 10 行 | クリアボタン | <input type="radio"/> 使用しない <input checked="" type="radio"/> 使用する |
| 見出し背景色 | #557799 | 見出しフォント色 | #FFFFFF |
| 奇数行背景色 | #FFFFFF | 偶数行背景色 | #FFFFFF |
| 罫線色 | #335577 | | |
| 説明 | | | |
| WHERE句追加記述 | WHERE | ※意図しない動作を防ぐため、WHERE句の記述全体を()で括弧することを推奨します。 | |
| ORDER BY句追加記述 | ORDER BY | | |

但し、一部の機能において kintone の検索機能に依存する制約が発生します。

【画面項目の指定における注意点】

| 項目 | 事象 |
|------------|---|
| 表示形式 | 全レコードの一括取得を前提とした「スクロール式」は指定できません。 kintone のレコード検索機能が 1 回あたり最大 500 行に制約があるためです。 |
| 表示行数 | 500 行を超える指定を行っても 500 行以上の表示は行われません。 kintone のレコード検索機能が 1 回あたり最大 500 行に制約があるためです。 |
| WHERE 句 | kintone REST API「レコードの取得 (GET)」の検索で指定する query パラメータの記述に従います。 |
| ORDER BY 句 | kintone REST API「レコードの取得 (GET)」の検索で指定する ORDER BY 句の記述に従います。 |

【検索動作の違い】

検索時に指定する条件は kintone のレコード取得 API に従って動作します。

特に文字列の部分一致 (LIKE) による動作では、X-point の固有マスタを利用する場合と異なる動きをする場合がありますので注意して下さい。なお、kintone の検索仕様については kintone developer site をご確認ください。

※ レコードの取得 (GET)

<https://cybozudev.zendesk.com/hc/ja/articles/202331474>

5. 5. kintone の登録レコードを検索する

X-point で書類を入力する際、JavaScript API を利用して kintone に登録されているデータを使いマスタ検索を実現する場合の手順を説明します。

※ X-point から kintone のデータを検索するためには“アクセス権”で“レコード閲覧”が許可された kintone アプリの API トークンが必要になります。

本項では、フォーム上の“顧客コード” (txtKokyakucode) でコードの入力後、フィールドからフォーカスが外れるタイミングで検索動作を実施し、“顧客名” (txtKokyakumei) に対応する内容を kintone の登録に基づいて設定を行う場合を説明します。

【フォーム】

| | | |
|-----|-------|-----|
| 顧客名 | 顧客コード | 顧客名 |
| | | |
| | 担当名 | |

【入力画面】

1) 入力前

| | | |
|-----|--|--|
| 顧客名 | | |
|-----|--|--|

2) コードの入力

| | | |
|-----|------|--|
| 顧客名 | G010 | |
|-----|------|--|

3) 検索動作後の状態 (フォーカス移動後)

| | | |
|-----|------|-----------------|
| 顧客名 | G010 | セレブコーヒージャパン株式会社 |
|-----|------|-----------------|

5.5.1. アプリの準備

kintone にマスタ検索に使用するデータを格納するアプリを用意しデータの登録を行います。

【アプリの一覧画面】

| レコード番号↑ | 顧客コード | 顧客名 | 部署 | 担当 | |
|---------|-------|---------------|----|----|--|
| 1 | G001 | 株式会社太陽商事 | | | |
| 2 | G002 | 富士山エレクトロニクス株式 | | | |
| 3 | G003 | 港北都市開発株式会社 | | | |
| 4 | G004 | 株式会社田園工業 | | | |

【配置するフィールド】

| フィールド名 | 種類 | フィールドコード | 必須項目 | 値の重複を禁止 |
|--------|----------|-----------------|------|---------|
| 顧客コード | 文字列 (1行) | kokyaku_code | チェック | チェックする |
| 顧客名 | 文字列 (1行) | kokyaku_name | — | — |
| 部署 | 文字列 (1行) | torihiki_busyo | — | — |
| 担当 | 文字列 (1行) | torihiki_tantou | — | — |

※ 検索に利用する項目は「顧客コード」、「顧客名」のみです。

※ 「レコード番号」は kintone が自動的に定義、決定する項目です。改めて指定の必要はありません。

本項では次のデータがアプリに登録されていることを前提に説明を進めます。

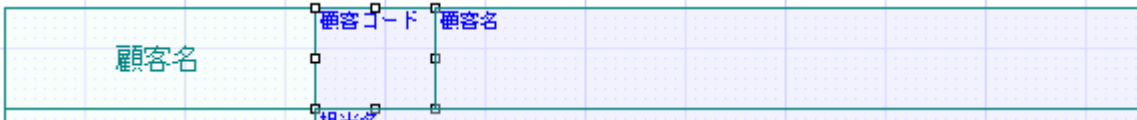
| 顧客コード | 顧客名 |
|-------|-----------------|
| G010 | セレブコーヒージャパン株式会社 |
| G017 | 株式会社紙沢山商事 |

5.5.2. フォームの修正

マスタ検索動作はフォームのコード欄に JavaScript として実装します。

eFormMaker でマスタ検索を組み込むフォームを開き“コード”入力欄を選択します。

■ 「顧客コード」(txtKokyakucode) を選択



■ プロパティ項目の「アクション」から「onBlur」イベントにアクションを追加します。

| イベント | アクション |
|--------------|-------|
| onClick | |
| onDbClick | |
| onChange | |
| onKeyDown | |
| onKeyPress | |
| onKeyUp | |
| onFocus | |
| onBlur | |
| onMouseOver | |
| onMouseOut | |
| R/O Calendar | True |
| R/O Window | True |

```
var kt_appid = 21;
var kt_domain = 'gnify';
var kt_api_token = 'OrUaiVM82dGPmEcuLbqDFgsduN9iCj20n08Z2xCN';

// 連携Objectを生成
var x2kintone = new xpoint2kintone(kt_domain, kt_appid, kt_api_token)

// 戻り値フィールドのObject
var kokyaku_code = XWEB.getXPObj('txtKokyakucode');
var kokyaku_name = XWEB.getXPObj('txtKokyakumei');

var query_text = 'kokyaku_code=' + kokyaku_code.getValue().toUpperCase();
var fields = [ '$id', 'kokyaku_code', 'kokyaku_name' ];

var call_back = function(status,responseText){
  if (status === 200) {
    if (JSON.parse(responseText))
      if (kr.records.length>0){
        kokyaku_code.setValue( kr.records[0].kokyaku_code.value );
        kokyaku_name.setValue( kr.records[0].kokyaku_name.value );
      } else {
        kokyaku_code.setValue( "" );
      }
  }
}
```

【アクションの記述に必要な情報】

- 参照データを格納している kintone アプリの ID 番号
- kintone アプリのフィールドコード名（コード格納項目、名称格納項目）
- kintone アプリの登録内容を閲覧できる API トークン（「レコード閲覧」権限付きのもの）
- X-point 入力フォームのフィールドコード名（コード格納項目、名称格納項目）

※ マスタ検索の利用では X-point 側の API トークンを必要としません。

■アクションに記述するプログラムの内容

```

var kt_appid = 21;           // 検索データが登録されている kintone アプリ ID
var kt_domain = '*****'; // kintone サブドメイン名 ("cybozu.com"の前部分)
var kt_api_token = '0rUAivM82dGpMEcuLbqDFgsduN9iCj20n08Z2xCN'; // kintone API トークン (レコード閲覧)
                                                                    初期設定パラメータ

// kintone 連携用のインスタンスを生成する
var x2kintone = new xpoint2kintone(kt_domain, kt_appid, kt_api_token);
                                                                    連携インスタンス生成

var kokyaku_code = XWEB.getXPObject('txtKokyakucode'); // 顧客コード
var kokyaku_name = XWEB.getXPObject('txtKokyakumei'); // 顧客名
                                                                    検索の対象となるフォーム上のフィールドを取得

var query_text = 'kokyaku_code="' + kokyaku_code.getValue().toUpperCase() + '"'; // 検索条件

var fields = [ '$id', 'kokyaku_code', 'kokyaku_name' ]; // アプリから取得する項目の定義

var call_back = function(status, responseText) {
  if (status == 200) {
    kr=JSON.parse(responseText)
    if (kr.records.length>0) {
      kokyaku_code.setValue( kr.records[0].kokyaku_code.value );
      kokyaku_name.setValue( kr.records[0].kokyaku_name.value );
    } else {
      kokyaku_code.setValue( "" );
      kokyaku_name.setValue( "" );
    }
  } else {
    kr=JSON.parse(responseText);
    alert( "[kintone Error] code: " + kr.code + ", id: " + kr.id + ", message: " + kr.message );
  }
};
                                                                    検索結果をフォームに反映する Call Back 処理
                                                                    ←検索結果があった場合
                                                                    ←検索結果が無かった場合
                                                                    ←連携動作の際、通信でエラーが発生した場合

x2kintone.search_kintone(query_text, fields, call_back);
                                                                    ←検索の実行

```

マスタ検索の絞りこみ条件、戻り値項目の指定、結果の反映は“search_kintone”に指定する引数で行います。内部の処理は kintone REST API の「レコードの取得」機能で実現されています。その為、絞りこみ条件（クエリ）、戻り値（フィールドコード）、Callback 処理の記述方法は cybozu developer network の kintone REST API 「レコードの取得（GET）」（レコードの一括取得（クエリで条件を指定））の内容に従って記述を行って下さい。

本例のクエリ条件は、顧客コード（txtKokyakucode）でフィールドコード「kokyaku_code」の欄と一致するものを検索します。

```
var query_text = 'kokyaku_code="' + kokyaku_code.getValue().toUpperCase() + '"';
```

戻り値の指定は、kintone のレコード ID、kokyaku_code、kokyaku_name の 3 項目を取得します。

```
var fields = [ '$id', 'kokyaku_code', 'kokyaku_name' ];
```

戻り値を設定する際、複数レコード該当した場合には戻り値の先頭レコードを利用する動作になっています。

```
kokyaku_code.setValue( "" );
kokyaku_name.setValue( "" );
```

動作の確認は X-point にフォームをアップロードしてから行います。eFormMaker のプレビュー機能では確認できません。

5.6. 承認管理からの一括更新

通常の kintone 連携動作は書類の提出、承認などの操作と連動して行われますが、例外的に連携動作を承認状況検索画面から行う事が出来ます。

実行時の注意事項として現在の書類の状況に従って連携が再実行される点になります。つまり、承認中であれば連携設定の動作ステップに現在ステップを指定していること、完了書類であれば「完了書類修正」を指定している必要があります。

■ 承認状況検索で書類のリストを表示

承認管理 →承認状況検索 →代理設定 →一括承認者設定 →「承認」設定 →承認完了書類の閲覧管理 →未承認者変更

承認状況検索 > A368出張申請書 ↶ 一覧へ戻る

| フォーム(名称、カナ、備考) | |
|----------------|--|
| 書類(件名1、件名2) | |
| 申請日 | |
| 承認完了日 | |
| 最終更新日 | |
| 書類No | |
| 滞留状況 | |

チェック項目を 1件中 1 - 1表示

| <input type="checkbox"/> | ▼No | ▼件名 1 | ▼件名 2 | ▼ルート名 | ▼状況 | ▼申請日 | ▼申請者 | 現在承認等 | ▼最終更新 |
|--------------------------|-------|-------|-------|--------|-----|------------------------|------|----------------|------------------------|
| <input type="checkbox"/> | 26180 | 営業部 | 営業太郎 | 規定のルート | 承認中 | 2020/10/30 14:14:45 | 管理者 | 木々 和明 佐伯 崇一 | 2020/10/30 14:14:46 |

■ 連携を再実行する書類をチェック

チェック項目を 1件中 1 - 1表示

| <input type="checkbox"/> | ▼No | ▼件名 1 | ▼件名 2 | ▼ルート名 | ▼状況 | ▼申請日 | ▼申請者 | 現在承認等 | ▼最終更新 |
|-------------------------------------|-------|-------|-------|--------|-----|------------------------|------|----------------|------------------------|
| <input checked="" type="checkbox"/> | 26180 | 営業部 | 営業太郎 | 規定のルート | 承認中 | 2020/10/30 14:14:45 | 管理者 | 木々 和明 佐伯 崇一 | 2020/10/30 14:14:46 |

■ コンボボックスから「kintone 連携の実行」を選択するとチェックされた書類の連携が再実行

チェック項目を 1件中 1 - 1表示

チェック項目を
削除する

| <input type="checkbox"/> | ▼件名 1 | ▼件名 2 | ▼ルート名 | ▼状況 | ▼申請日 | ▼申請者 | 現在承認等 | ▼最終更新 |
|-------------------------------------|-------|-------|--------|-----|------------------------|------|----------------|------------------------|
| <input checked="" type="checkbox"/> | 営業部 | 営業太郎 | 規定のルート | 承認中 | 2020/10/30 14:14:45 | 管理者 | 木々 和明 佐伯 崇一 | 2020/10/30 14:14:46 |

第6章. X-pointフォームからkintoneアプリを作成する

6.1. kintone アプリ作成機能について

X-point のフォームの情報をもとに kintone へ API リクエストを送信し、アプリを作成することが可能です。

連携に必要な自動申請設定や認証トークンの発行などは、必要に応じ自動的に行われます。

本機能は、X-point と kintone との連携における初期設定を簡略化することを目的としています。

既存アプリを修正する機能は含まれておりません。そのため、自動申請の項目を追加する場合や承認ルートコード等の変更を行う等、連携設定を修正する必要がある場合は、別途、連携用 JavaScript を修正する必要がある点にご留意ください。

【本機能で作成されるもの】

- ① kintone アプリ（未公開の状態で作成されます）
- ② 対象フォームの自動申請設定
- ③ 対象フォームの X-point 認証トークン
- ④ 連携用 JavaScript
- ⑤ 連携用 API 実行ボタン（申請、承認、書類表示、承認待ち件数取得）

6.2. X-point から kintone アプリを作成する

連携アプリを作成するフォームの【⇒他操作】をクリックし、展開されたメニュー内の【kintone アプリ作成】をクリックします。

The screenshot shows the X-point interface with a table of forms. The table has columns for Name, Count, FCD, Table Name, and TSF File. A dropdown menu is open for the 'A316 稟議書' form, showing various actions including 'kintone アプリ作成' (kintone app creation).

| 名称 | 枚数 | FCD | テーブル名 | TSFファイル | アクション |
|----------|----|------|---------|----------|--|
| A316 稟議書 | 1枚 | A316 | xp70507 | A316.tsf | ⇒ 他操作 フォームグループ追加 アクセス権限 フィールド編集権限 書類制御 状況確認者 kintone連携 kintoneアプリ作成 Webhook設定 電帳法保存設定 CSVテンプレート作成 ゲスト |

！注意事項

- ・ アプリ作成を行うフォームはフォームグループに所属させてください。
- ・ 登録したフォーム種別がワークフローフォームの場合は承認ルートの設定も前提条件となります。

6.2.1. kintone アプリ作成準備

■基本情報

kintone アプリを作成するため kintone 認証情報、接続先等を入力します。

kintone連携アプリ作成 一覧 フォームグループ フォーム 複数枚フォーム kintone連携一覧

【A316稟議書】のフォーム登録情報をもとにkintone連携アプリを作成します。色付きは必須項目です。

| 基本情報 | |
|----------------|--|
| フォーム名称 | A316稟議書 (A316) |
| フォーム種別 | ワークフロー |
| 連携データ送信の方向 | <input checked="" type="radio"/> 双方向 <input type="radio"/> X-point から kintone <input type="radio"/> kintone から X-point |
| kintone認証情報 | ログイン名 <input type="text"/> パスワード <input type="text"/> |
| 接続先kintoneドメイン | <input type="text"/> |
| kintoneアプリ名 | <input type="text"/> |
| kintoneアプリ作成場所 | <input type="checkbox"/> ゲストスペースを利用する スペースID <input type="text" value="0"/> (未指定の場合は0) スレッドID <input type="text" value="0"/> (未指定の場合は0) |
| 書類ID格納フィールド | -- フィールド名 -- |

■基本情報の入力項目

| 項目 | 入力 | 説明 |
|------------------|----|--|
| 連携データの送信方向 | 選択 | アプリの使用目的に応じ変更してください。 |
| 認証情報：ログイン名 | 必須 | アプリ作成を行う kintone ユーザのログイン名を入力してください。 |
| 認証情報：パスワード | 必須 | アプリ作成を行う kintone ユーザのパスワードを入力してください。 |
| 接続先 kintone ドメイン | 必須 | 接続先の kintone サーバの FQDN を入力してください。 |
| kintone アプリ名 | 必須 | 作成したいアプリの名前を任意に入力してください。 |
| ゲストスペースの利用 | 任意 | ゲストスペースにアプリを作成する場合はチェックを入れてください。 チェックを入れた場合、スペース ID とスレッド ID の入力が必要となります。 |
| スペース ID | 任意 | アプリをスペースに作成する場合はスペース ID を入力します。 スペース ID を入力した場合、スレッド ID の入力が必要となります。 |
| スレッド ID | 任意 | スペースにアプリを作成する場合はスレッド ID を入力してください。 スレッド ID を入力した場合、スペース ID の入力が必要となります。 |
| 書類 ID 格納フィールド | 必須 | X-point フォーム上の書類 ID を格納するフィールドを選択してください。 |

cybozu.com 共通管理者以外のユーザで認証し、アプリ作成を行う場合は

kintone システム管理>アクセス権 よりシステム管理権限を付与してください。

アクセス権 [?ヘルプ](#)

kintone全体に関わるアクセス権の設定ができます。
kintoneシステム管理者の指定（システムの管理権限の設定）も、この画面から行います。

アクセス権の一覧

行を追加して、アクセス権を設定します。上の行の設定が優先されます。

cybozu.com共通管理者には、すべての権限が付与されます。

追加する

| 優先順位 | ユーザー/組織/グループ | アクセス権の継承 | システムの管理 | アプリグループの管理 (閲覧のみ) | アプリグループの管理 | アプリの作成 | アプリの管理 | スペースの作成 | ゲストスペースの作成 |
|------|--------------|------------------------------------|-----------------------------|-----------------------------|-----------------------------|--|--|--|-----------------------------|
| | Everyone | <input type="checkbox"/> 下位組織に継承する | <input type="checkbox"/> 許可 | <input type="checkbox"/> 許可 | <input type="checkbox"/> 許可 | <input checked="" type="checkbox"/> 許可 | <input checked="" type="checkbox"/> 許可 | <input checked="" type="checkbox"/> 許可 | <input type="checkbox"/> 許可 |

■自動申請設定情報

kintone から X-point に書類の提出を行う場合は対象フォームに kintone 連携用の自動申請登録が必要です。
設定されていない場合や、基本情報で連携データの送信方向を「X-point から kintone」以外に設定した場合は、
自動申請設定を新たに作成します。

フォーム (A316稟議書 (明細入)) には自動申請設定が登録されていないため新たに作成します。

| 自動申請設定情報 (kintone連携用) | | |
|-----------------------|--------------|--|
| 申請ルート (ルートコード) | -- ルート -- | ユーザ選択ルートで、ルートコードが設定されていないものはリストに表示されません。 |
| ユーザコード対応フィールド | -- フィールド名 -- | |
| レコードID対応フィールド | -- フィールド名 -- | |
| 表定義 | -- 表定義名 -- | |

■自動設定情報の入力項目

| 項目 | 入力 | 説明 |
|---------------|----|--|
| 申請ルート | ※ | 自動申請で利用する承認ルートを選択します。 対象のフォームがワークフローフォームの場合は入力が必要となります。 |
| ユーザコードフィールド | 必須 | kintone のユーザコードを保存するフォーム上の文字フィールド |
| レコード ID フィールド | 必須 | kintone のレコード ID を保存するフォーム上の整数フィールド |
| 表定義 | 任意 | 自動申請項目で表定義を利用する場合は選択してください。利用できる表定義はいずれか1つのみとなります。 |

本機能を用いて作成される自動申請設定は、kintone 連携を行う際の基本的な内容に基づいて作成されます。

また、kintone のログイン名と同じ X-point ユーザコードを使用していることを前提としており、申請ユーザータイプは「CSV データ指定」に固定されます。詳細については、「3.2.3 自動申請の登録」を参照してください。

！注意事項

作成元となる X-point フォームのフィールドに対応していない kintone アプリのフィールドは作成されず、自動申請の連携フィールドも設定されません。対象のフィールドは以下の通りです。

- ・ラベルフィールド
- ・選択肢が設定されていないコンボボックス、リストボックス
- ・kintone アプリに存在しないタイプのフィールド (イメージ、イメージアップロード、印影、バーコード、パスワード、不可視)

また、複数枚フォームから kintone アプリを作成した場合、ページ違いでフィールド ID が重複しているフィールドは1つしか作成されません。作成されるフィールドはページ番号の大きいほうのフィールドを参照します。

▶特記事項

・コンボボックス・リストボックスにおいて選択肢として登録済みの簡易マスタを設定する場合、kintone アプリ上ではテキストフィールドとして登録されます。

6.2.2. kintone アプリ作成実行

設定した内容で kintone アプリ作成を行います。実行の際は必須項目を入力の上、【作成】ボタンを押下してください。

■アプリ作成実行結果

kintone連携アプリ作成 > 完了

一覧 フォームグループ フォーム 複数枚フォーム kintone連携一覧

kintone連携アプリ作成関連処理の実行結果です。

| ステータス | 項目 | 詳細 |
|-------|---------------------------|------------|
| 成功 | アプリ作成 | アプリID: 365 |
| 成功 | フィールド登録 | |
| 成功 | 自動申請登録 | |
| 成功 | X-point認証トークン生成 | |
| 成功 | kintone連携テンプレート発行 | |
| 成功 | kintone連携テンプレートファイルアップロード | |
| 成功 | 連携用パーツ作成 | |
| 成功 | 連携用パーツアップロード | |
| 成功 | 連携用JavaScriptファイル登録 | |

X-pointからkintoneアプリへのレコード追加・更新を行うため、kintoneにてアプリを公開したのち以下よりkintone連携アプリの追加を行ってください

[kintone連携設定](#)

アプリ作成を行うと関連処理の実行結果が一覧表示されます。失敗した項目は原因が詳細欄に表示されます。

また、連携データの送信方向を「双方向」、「X-point から kintone」に設定した場合は kintone 連携設定を追加で行う必要があるため、実行結果一覧表の下に kintone 連携設定へのリンクが表示されます。設定は「5.1.1. 連携先の kintone アプリを指定」と同じ流れで行います。

■作成された kintone アプリ

実行結果の項目名「アプリ作成」の詳細欄には作成したアプリの ID が返却されています。

kintone アプリ管理画面より対応する ID のアプリが作成されていることをご確認ください。

アプリの一覧 [CSV形式でダウンロードする](#)

表示するアプリ

| ID | アプリ名 | 設定 | 所属スペース | アプリグループ | ステータス | レコードの最終更新日 | レコード数 | フィールド数 | 1日のAPIリクエスト数 | 添付ファイル | カスタイズ |
|-----|------|----|--------|---------|-------|------------|-------|--------|--------------|---------|-------|
| 277 | 案議 | | (なし) | Public | 運用開始前 | 2025-06-09 | 0 | 0 | 0 | 0.00 GB | あり |

本機能により自動で作成されたアプリはフィールドのレイアウト設定がされておりません。任意に変更してください。

また、アプリは未公開の状態で作成されますのでアプリの設定画面より公開設定を行ってください。

X-point から kintone ヘデータ送信を行う場合は kintone アプリの API トークンを発行の上、アプリを公開し、X-point 上で kintone 連携設定を行ってください。

第7章. kintoneアプリからX-pointフォームを作成する

本章では kintone アプリ上で設定される「フォーム」の定義情報をもとに、X-point 上で書類申請用の「フォーム」を作成・登録する機能について説明します。なお、本章において「kintone フォーム作成」とは kintone フォームの定義を参照して X-point フォームを作成することを指します。

7.1. kintone フォーム作成機能について

本機能を利用すると、kintone アプリのフォーム定義（フィールド構成や項目情報）もとに、X-point フォームを作成・登録することが可能です。本機能により作成された X-point フォームは、X-point 内で書類申請を行うことで申請内容が kintone アプリへ連携される状態となります。

【本機能で作成されるもの】

- ① X-point フォーム
- ② 作成されたフォームに関する各種設定

| 項目 | 説明 |
|---------------------------------|--|
| フォームグループ設定 | フォームを使用用途によってグループ分けして公開するためのものです。 本機能を利用すると、新たに「kintone 連携フォームグループ（自動作成）」が作成され、登録されるフォームはこのグループに所属します。全ユーザが提出可能です。 |
| 承認ルート設定 | ワークフローを自動で回すための承認の経路です。 本機能で登録されるフォームの承認ルートは条件付ルートの規定のルートで、ステップ数4、全て空ステップの承認者不在ルートとなります。登録後に承認者を設定し利用してください。 |
| 自動申請設定 | kintone アプリから X-point 書類を提出する際に必要な設定で、フォーム作成と同時に kintone 連携用の自動申請設定が作成可能です。 kintone のログイン名と同じ X-point ユーザコードを使っている想定で設定が作成されます。 変更が必要な場合、本マニュアルの「3.2.3 自動申請の登録」を参照してください。 |
| kintone 連携設定/ X-point 認証トークン | kintone アプリから X-point 書類へデータを送信するための認証情報が作成されます。 生成される API トークンのアクセス権は全てにチェックが付いた状態となります。 |
| kintone 連携設定/ kintone 連携先アプリ | X-point 書類から kintone アプリへデータを送信するための連携設定が作成されます。 初期設定では、申請・承認・承認完了後の修正保存のタイミングでデータが送信されます。 また、「フィールド指定」にて X-point フォームと kintone アプリのフィールドを対応付けることで、対応するフィールド間でデータ連携行われます。 (※一部、対応付けできないフィールドがあります。) |

※ レコード番号、作成者、作成日時、更新者、更新日時、計算フィールドは X-point フォームに作成されますが、kintone 連携設定ではフィールド対応付けの対象外となります。

7.2. kintone アプリ情報からフォームを作成する

■事前準備

本機能を利用するには事前に以下の準備が必要です。

- kintone アプリが作成、公開されていること
- 該当アプリの API トークンが発行されていること

本機能では kintone アプリに配置されているフィールドをもとに X-point フォームを作成します。

また X-point と kintone の連携では X-point フォーム上で以下の情報を管理するためのフィールドを使用します。

- レコード ID
- ユーザコード
- 書類 ID

X-point から kintone へのレコード登録・更新を行わない場合、これらのフィールド定義は必須ではありませんが、

将来的な X-point から kintone への連携を考慮し、あらかじめ kintone アプリにこれらのフィールドを定義しておくことを推奨します。

詳細な手順については第 3 章「事前準備」の以下の項目をご確認ください。

- 3.1.1. 連携アプリの作成
- 3.1.2. API トークンの作成

■kintone フォーム作成

上記の事前準備が完了した後、本機能を実行し kintone アプリをもとに X-point フォームを作成します。

本機能を実行するにはフォーム管理画面詳細ヘッダ中の【kintone フォーム】をクリックします。

| | 名称 | 枚数 | FCD | テーブル名 | TSFファイル | アクション |
|---|----------------|----|---------------|----------|-------------------|-------------------|
| — | 申請書 (5) | | | | | |
| 🔍 | 物品購入申請書 | 1枚 | F002_1 | F002_1 | F002_1.tsf | ✏️ 🗑️ 📄 👤 Σ → 他操作 |
| 🔍 | 社用車ドライバート利用報告書 | 1枚 | SK120 | xp16150 | SK120.tsf | ✏️ 🗑️ 📄 👤 Σ → 他操作 |
| 🔍 | 社員名簿 | 1枚 | a547 | xp891056 | a547.tsf | ✏️ 🗑️ 📄 👤 Σ → 他操作 |
| 🔍 | 出張精算書 | 1枚 | teikokusyoin1 | xp232201 | teikokusyoin1.tsf | ✏️ 🗑️ 📄 👤 Σ → 他操作 |

7.2.1. kintone フォーム作成準備

kintoneフォーム作成

[一覧](#)
[フォームグループ](#)
[フォーム](#)
[複数枚フォーム](#)
[kintone連携一覧](#)
[kintoneフォーム](#)

kintoneのアプリ情報をもとにフォームを作成します。

フォーム作成関連項目

| 連携先kintone情報 | 作成するフォームの基本情報 |
|--------------------------|--|
| kintoneドメイン [入力欄] | フォーム名称 [入力欄] |
| アプリID [入力欄] | フォームコード [入力欄] |
| APIトークン [入力欄] | 件名 -- フィールド名 -- [選択欄] |
| ゲストID 0 (未指定の場合は0) | フォーム種別 <input checked="" type="radio"/> ワークフロー <input type="radio"/> 通常 |
| [確認] | 備考 [入力欄] |

作成するフォームの自動申請設定を登録します。

自動申請設定項目

| | |
|------------------------|---|
| 自動申請 | <input checked="" type="radio"/> 作成する <input type="radio"/> 作成しない |
| 「kintoneレコードID」対応フィールド | -- フィールド名 -- [選択欄] |
| 「ユーザコード」対応フィールド | -- フィールド名 -- [選択欄] |
| 表定義 | -- 表定義名 -- [選択欄] |

[作成]

■ 連携先 kintone 情報

作成したいフォームの元になる kintone アプリの情報を指定します。

連携先 kintone 情報欄の入力後、確認ボタンを押下するとアプリ情報の取得が行われます。

| 項目 | 入力 | 説明 |
|--------------|----|---------------------------|
| kintone ドメイン | 必須 | 接続先の kintone サーバの FQDN |
| アプリ ID | 必須 | フォーム作成元になりたいアプリの ID |
| API トークン | 必須 | フォーム作成元になりたいアプリの API トークン |
| ゲスト ID | 任意 | ゲストスペースのアプリを利用する場合 |

■ 作成するフォームの基本情報

作成するフォームの基本情報を入力します。

| 項目 | 入力 | 説明 |
|---------|----|---|
| フォーム名称 | 必須 | フォーム名称を指定します。 初期取得値：アプリ名称 |
| フォームコード | 必須 | フォームを識別する一意のコードを指定します。 初期取得値：アプリコード |
| 件名 | 必須 | ユーザサイト【検索】【承認】両画面で、提出書類を一覧表示する際に使用する「件名 1」のフィールドを指定します。 |
| フォーム種別 | 選択 | 作成するフォームをワークフローで使用するか否かを選択します。 初期値：ワークフロー |
| 備考 | 任意 | 作成するフォームについて、補足情報などがある場合に入力します。 |

■ 自動申請設定項目

kintone アプリから X-point フォームへの連携を行う際、自動申請設定が必要となります。

本項目を設定することでフォーム作成と同時に kintone 連携用の自動申請設定を作成できます。

連携のための自動申請設定に関して、詳細は本マニュアル 3.2.3.自動申請の登録をご確認ください。

| 項目 | 入力 | 説明 |
|------------------------------|----|---|
| 自動申請 | 選択 | 自動申請を作成するか作成しないかを選択します |
| 「kintone レコード ID」 対応フィールド | 必須 | kintone レコード ID を格納するフィールドを指定します |
| 「ユーザコード」 対応フィールド | 必須 | kintone のログイン名にあたる情報を格納するフィールドを指定します |
| 表定義 | 任意 | kintone アプリにおける「テーブル」は X-point フォームでは表定義に変換されず 連携のための自動申請では表定義を 1 つ選びます。(複数は選択できません) |

！ 注意事項

「kintone レコード ID」対応フィールドには整数値を保持するフィールドをご設定いただくようお願いいたします。

整数以外の値を保持するフィールドを設定した場合、kintone へのフィールド値送信時にエラーとなり、書類の申請等が行えない可能性があります。

7.2.2. kintone フォーム作成実行

必要項目を入力し【作成】ボタンを押下すると tsf ファイルが作成されます。

kintoneフォーム作成>確認

kintoneのアプリ情報をもとにフォームを作成します。

① 必要に応じTSFのレイアウトを修正しアップロードしてください。不要な場合はそのまま登録ボタンを押下することでフォームが登録されます。

tsfファイル確認

生成されたTSFファイル [【案議書 \(kringi.tsf\)】をダウンロード](#)

参照

TSFファイル-修正

ドラッグ&ドロップするか参照ボタンから選択してください

登録

作成された tsf ファイルを確認したい場合は「生成された TSF ファイル」欄のリンクからファイルをダウンロードしてください。内容に問題がなければそのまま登録ボタンを押して登録を完了します。

修正が必要な場合は、eFormMaker 等のツールで編集してください。編集後は保存したファイルを修正欄にアップロードし登録を行ってください。

！ 注意事項

・作成された tsf ファイル内のフィールド ID を変更した場合、連携設定が正常に行われません。フォーム登録前の修正はレイアウトを変更する程度にとどめることを推奨いたします。

・本機能で作成された tsf におけるフィールド変換は、本マニュアル「9.3.1.3 フォーム (TSF) 雛型ダウンロード」の「■変換されるフィールドの種類」に記載の内容に基づく仕様です。

以下の画面が表示されたら登録完了です。

kintoneフォーム作成 > 完了

[一覧](#)
[フォームグループ](#)
[フォーム](#)
[複数枚フォーム](#)
[kintone連携一覧](#)
[kintoneフォーム](#)

kintoneフォーム作成-関連処理実行結果一覧

| ステータス | 項目 | 詳細 |
|-------|-----------------|--------------------------------|
| 成功 | フォーム登録 | フォーム名称: 稟議書 |
| 成功 | kintone連携先アプリ登録 | アプリID: 365 |
| 成功 | フォームグループ設定 | |
| 成功 | 承認ルート作成 | 承認ルート名称: kintone連携承認ルート (自動作成) |
| 成功 | X-point認証トークン生成 | |
| 成功 | 自動申請登録 | 自動申請名称: 稟議書_kintone連携 |

① フォーム作成が完了いたしました。以下から次のアクションを選択して下さい。

オプション設定

- [フォームグループへ追加](#)
- [承認ルート設定](#)
- [フィールド編集権限設定](#)
- [状況確認ユーザ設定](#)
- [マイクエリ権限設定](#)

その他の遷移先

- [フォーム一覧へ](#)
- [続けて新規フォーム作成](#)
- [プロパティ表示](#)

特記事項

kintone から X-point ヘデータを送信する場合は、kintone アプリ側で追加の設定が必要です。
 詳細は第 4 章「kintone アプリから X-point への連携を行う」を参照してください。

連携設定に失敗した場合、設定の整合性を保つため、フォーム作成以外の処理はすべてキャンセルされます。このため、「フォームグループ」「承認ルート」「kintone 連携設定」「X-point 認証トークン」「自動申請」が部分的に登録されることはありません。
 なお、フォーム作成と連携設定は異なる処理単位で実行されるため、連携設定に失敗した場合でもフォームのみが作成された状態で残ります。

連携設定に失敗した場合は以下の点をご確認ください。

| 項目 | 内容 |
|------------------|------------------------------|
| kintone API トークン | 必要な操作権限を持つ API トークンを使用しているか |
| kintone アプリ | フィールド定義が変更・削除されていないか |
| ネットワーク | kintone および X-point への通信が可能か |

X-point と kintone の双方向でデータ連携を行う場合は、第 4 章および第 5 章の手順を実施してください。

第8章. CSVファイルによる連携

8.1. kintone の操作

kintone のアプリに登録されているレコードの内容を CSV ファイルに出力、CSV ファイルの内容をレコードに追加する場合は「kintone コマンドラインツール」を利用します。本項ではツール利用方法の概要を説明します。

なお、ツールの入手方法、利用手順の詳細は「cybozu developer network」をご確認ください。

- ※ kintone コマンドラインツール(β) <https://cybozudev.zendesk.com/hc/ja/articles/202708214>
- ※ kintone コマンドラインツールの使い方 <https://cybozudev.zendesk.com/hc/ja/articles/202957070>
- ※ 本項の内容は 2016/12/27 時点の「kintone コマンドラインツール」仕様に基づいています。

8.1.1. CSV ファイルのダウンロード

X-point の自動申請機能を利用して登録できるようアプリに登録されているレコードをコマンドラインツールで CSV ファイルに変換します。X-point の自動申請機能は添付ファイルを扱う事ができないため、ダウンロードは添付ファイル以外のフィールドについて行います。

■ダウンロード元になるアプリ例

| 契約書番号 | 契約書名 |
|--------|----------|
| S00123 | 資材循環取扱契約 |

■アプリ番号

22

■kintone サブドメイン

ksubdomain (FQDN: ksubdomain.cybozuy.com)

■フィールド名 : フィールドコード

顧客コード : txtKokyakucode

顧客名 : txtKokyakumei

主担当 : txtTantou

審査完了希望日 : txtKiboubi

管理コード : txtKanricode

審査主旨 : txtMemo

テーブル : group1

■API トークン (レコード閲覧可)

diVUoVQDgnnU7r01v4sFyfUiGnX0ResLGg0fIk1s

■コマンド実行例

```
cli-kintone.exe -a 22 -o csv -d ksubdomain -t "Xm603KEs8uhz2aLuG6lAKaKy8wqwFW9IGbJFvcpT"  
-c "$id,txtKokyakucode,txtKokyakumei,txtTantou,txtKiboubi,txtKanricode,txtMemo,group1"  
-e sjis > kintone_a22.csv
```

■出力 CSV 「kintone_a22.csv」例

```
*,"$id","txtKokyakucode","txtKokyakumei","txtTantou","txtKiboubi","txtKanricode","txtMemo","group1","txtKeiyakusyono","txtKeiyakusyomei"
*, "2", "G010", "セレクトコーヒージャパン株式会社", "u001", "2017-02-15", "", "X-point での起票", "103011", "記入してください", "納入に係わる契約事項の締結署名を記入"
, "2", "G010", "セレクトコーヒージャパン株式会社", "u001", "2017-02-15", "", "X-point での起票", "103012", "", ""
, "2", "G010", "セレクトコーヒージャパン株式会社", "u001", "2017-02-15", "", "X-point での起票", "103013", "", ""
*, "1", "G017", "株式会社紙沢山商事", "u001", "2017-02-01", "P018、ECO プロジェクト", "ゼロエミッション実現に伴うリサイクル事業者選定契約", "102998", "S00123", "資材循環取扱契約"
, "1", "G017", "株式会社紙沢山商事", "u001", "2017-02-01", "P018、ECO プロジェクト", "ゼロエミッション実現に伴うリサイクル事業者選定契約", "103021", "", ""
, "1", "G017", "株式会社紙沢山商事", "u001", "2017-02-01", "P018、ECO プロジェクト", "ゼロエミッション実現に伴うリサイクル事業者選定契約", "103022", "", ""
```

8.1.2. CSV ファイルのアップロード

kintone アプリへのレコード追加では CSV ファイルの列「\$id」を空欄にするか、アプリのレコード ID を指定するかにより動作が変わります。空欄の場合は追加、既存レコード ID を指定すると追加（存在しない場合はエラー）になります。

■入力 CSV 「kintone_a12.csv」例（レコード追加の場合）

```
*,"title","name","textarea","group","CAA","CBB"
*, "kintone 連携", "栄都 太郎", "連携動作サンプル", "情報システム", "1 行目-A", "1 行目-B"
, "kintone 連携", "栄都 太郎", "連携動作サンプル", "情報システム", "2 行目-A", "2 行目-B"
```

■アップロードを行うコマンド実行例

```
cli-kintone.exe -a 12 -o csv -d ksubdomain
-t "diVUoVQDgnnU7r01v4sFyfUiGnX0ResLGg0fIk1s" -e sjis -f to_kintone.csv
```

8. 2. X-point の操作・設定

8. 2. 1. 自動申請の設定

kintone より出力した CSV ファイルを X-point の書類として申請する場合は自動申請機能を利用します。

「事前準備」>「X-point 側の準備」>「自動申請の登録」の記載内容、及び「管理者機能マニュアル ～自動申請～」に従って CSV ファイルの内容を登録するフォームの設定を行って下さい。

8. 2. 2. 自動申請の実行

管理サイトの「ドメイン」>「タスク管理」から自動申請用の CSV ファイルをアップロードします。

■タスク管理で自動申請設定を選択します。

The screenshot shows the 'Task Management' page with the breadcrumb 'タスク管理 > 自動申請 > データファイル'. A dropdown menu is open, showing '自動申請設定' selected and highlighted with a red box. The dropdown also shows '--- 自動申請設定 ---'. A red message above the dropdown says '自動申請設定を選択して下さい。' (Please select automatic application settings.)

■自動申請用 CSV ファイルをアップロードします。

The screenshot shows the 'Automatic Application Settings' page with the breadcrumb 'タスク管理 > 自動申請 > データファイル'. The 'データファイル管理' section is highlighted with a red box. It contains a table with columns 'No', 'ファイル名', '最終更新日時', and 'ファイルサイズ'. Below the table, there is a red message: 'データファイルのアップロード [ファイルを選択] ファイルが選択されていません [アップロード]' (Upload CSV file [Select file] No files are selected [Upload]).

※ アップロードされたファイルは指定した自動申請で登録されている CSV ファイル名で内部管理されます。

■アップロードを完了するとファイルが登録されます。

ドメイン管理 → ドメイン → フォーム関連情報 → メール設定 → ログ → タスク管理 → 定義書出力

タスク管理 > 自動申請 > データファイル

アップロードが完了しました。
・申請用csvデータを指定してアップロードする際、ファイルは自動申請設定で決められたファイル名に変換されてアップロードされます。
・アップロード時に同名ファイルが存在する場合は上書きされます。

自動申請設定 from kintone【契約書社内審査 (契約書社内審査)】

| No | ファイル名 | 最終更新日時 | ファイルサイズ |
|----|--------------------|---------------------|---------|
| 1 | kintoneAppid55.csv | 2020/11/02 10:52:46 | 961バイト |

データファイルのアップロード [ファイルを選択](#) ファイルが選択されていません [アップロード](#)

■「実行」メニューの利用でアップロードされている CSV ファイル全てを直ちに申請できます。

ドメイン管理 → ドメイン → フォーム関連情報 → メール設定 → ログ → タスク管理 → 定義書出力

タスク管理 > 自動申請 > 実行

自動申請を実行します。
・各ドメインの → 自動申請設定 に基づいてすべての自動申請を実行します。
・「開始」ボタンを押下すると全てのドメインの自動申請タスクが実行されます。
・実行データについては、→ データファイルページ にて確認してください。

[開始](#)

「実行」メニューから自動申請を行った場合は直ちに結果が表示されます。

ドメイン管理 → ドメイン → フォーム関連情報 → メール設定 → ログ → タスク管理 → 定義書出力

タスク管理 > 自動申請 > 実行

自動申請の実行が完了しました。
→ タスクログ から実行履歴を確認できます。

▼ 実行結果ログ [実行結果ログをダウンロード](#)

```
*****  
X-point Command Tools '自動申請'  
  
Copyright (C) 2012-2020 ATLED CORP. All rights reserved.  
*****  
  
Checking domain code: [ trial ]  
Checking form code: [ DEMO_S01 ]  
対象ファイルが存在しません  
1件のエラーが発生しました。エラーが発生した書類は申請されていません。  
Executed document count: [ 1 ( error 1 ) ]
```

※ タスクログには実行エラーの内容が記録されています。

- 「スケジュール」メニューを利用すると CSV ファイルのアップロード時刻を指定できます。
時刻の指定は CSV ファイルがアップロードされていない状態でも指定できます。

ドメイン管理 → ドメイン → フォーム関連情報 → メール設定 → ログ → タスク管理 → 定義書出力

タスク管理 > 自動申請 > スケジュール

自動申請スケジュールを設定します。

スケジュール設定

開始時間 ※

なし
 毎時 -- 分
 毎日 3 時 20 分
 毎週 -- 曜 -- 時 -- 分
 毎月 -- 日 -- 時 -- 分

保存

- ※ 図の場合、毎日午前 3 時 20 分に自動申請が開始されます。
- ※ 指定された時刻になるとアップロードされている CSV ファイル全てで自動申請が実行されます。

スケジュール指定による自動申請実行の結果はタスクログに内容が記録されます。

ドメイン管理 → ドメイン → フォーム関連情報 → メール設定 → ログ → タスク管理 → 定義書出力

タスク管理 > 自動申請 > タスクログ

自動申請タスクログ

検索条件

実行日: 2020 年 10 月 2 日 ~ 2020 年 12 月 2 日

実行結果: 成功 失敗 警告

30件 ずつ 検索

4件中 1-4表示

| 処理日 | 実行結果 | メッセージ |
|---------------------|------|------------------------------|
| 2020/11/02 10:55:26 | 失敗 | エラー: 12件 自動申請名: from kintone |
| 2020/10/22 17:34:42 | 失敗 | エラー: 1件 自動申請名: from kintone |
| 2020/10/22 17:34:30 | 失敗 | エラー: 1件 自動申請名: from kintone |
| 2020/10/22 17:34:04 | 失敗 | エラー: 1件 自動申請名: from kintone |

4件中 1-4表示

- ※ 自動申請から登録した場合のログは、タスクメニューの「kintone 連携」では確認できません。
「kintone 連携」の実行ログで確認できるのは、kintone 連携 API を利用した申請書類のみです。

8.2.3. クエリ CSV 出力の設定

X-point より CSV ファイルを出力し kintone に一括登録する場合は「クエリ」機能を利用します。クエリ機能の設定により書類内容の CSV ファイル出力が可能になります。

- 1) クエリ実行画面（ブラウザ）からの CSV ファイルダウンロード
- 2) タスク実行（CSV ファイル自動出力）による AP サーバー上への CSV ファイル出力
- 3) リモート出力機能を利用したネットワーク経由での CSV ファイル出力

クエリ管理 →クエリ設定 →マイクエリ権限設定 →タスクログ

クエリ作成 一覧 Σ クエリグループ Σ クエリ

[次へ](#)

| | | | |
|--|---|-------------|---------------------|
| コピー元 | クエリグループ 未所属 | クエリ --クエリ-- | コピー |
| 基本情報 | | | |
| クエリコード ① 省略すると自動生成されます。 | | | |
| クエリ名 ① | 契約書社内審査 for kintone | | |
| フォーム ① | 契約書社内審査 | | |
| クエリ種別 ① | <input checked="" type="radio"/> 一覧 <input type="radio"/> サマリ <input type="radio"/> クロス集計 | | |
| 表定義 ① | <input checked="" type="radio"/> 使用する 表定義グループ (group1) <input type="radio"/> 使用しない | | |
| 開始行マークの指定 ① | <input checked="" type="radio"/> 使用する 先頭行文字 * 継続行文字 <input type="radio"/> 使用しない | | |
| CSVファイル名 ① CSVファイル名を指定しない場合は「query.csv」となります。 | .csv | | |
| CSVファイルのヘッダー表示 | <input checked="" type="radio"/> 表示する <input type="radio"/> 表示しない | | |
| CSVファイル出力済み管理 | <input checked="" type="radio"/> CSVファイル自動出力された書類だけを出力済みにする <input type="radio"/> クエリ機能で出力された書類を全て出力済みにする ※管理者サイトで出力された書類は、出力済みにはなりません。 | | |
| 備考 | | | |

kintone で利用する場合は「使用する」、
先頭行文字「*」、継続行文字「」（空欄）にする

| | | |
|---------------|---|---|
| CSVファイル自動出力設定 | <input checked="" type="radio"/> する <input type="radio"/> しない | タスク機能を利用する場合は 「CSV ファイル自動出設定」を「する」にします |
| CSVファイル出力時の処理 | <input checked="" type="radio"/> 上書きする <input type="radio"/> 上書きしない（出力を行わない） <input type="radio"/> ファイル名を変更する ※ファイル名の末尾に実行日時を付け加えます。 | |
| リモート出力設定 | <input type="radio"/> する <input checked="" type="radio"/> しない | リモート出力を利用する場合は、 「リモート出力設定」を「する」にします |

★ CSVファイル自動出力設定について
CSVファイル自動出力設定を行うには、別途スケジュール設定を行う必要があります。
設定方法については [CSVファイル自動出力設定マニュアル](#) をご参照下さい。

出力するフィールドは、kintone に取り込む項目を表示列に指定し、クエリ表示名を kintone のフィールドコードにします。
レコード ID は「\$id」に設定します。

クエリ管理 →クエリ設定 →マイクエリ権限設定 →タスクログ

クエリ作成 一覧 Σ クエリグループ Σ クエリ

一覧に表示する列を選択し、その列の書式を設定して下さい。

| 表示列設定 カスタム列を追加する 表示列の並び順設定 | | | | |
|---|----------|----------|---------|-------------|
| <input checked="" type="checkbox"/> | 列名 | クエリ表示名 | 型 | |
| <input type="checkbox"/> | 件名1 | 件名1 | | フォーマット設定を表示 |
| <input type="checkbox"/> | 件名2 | 件名2 | | フォーマット設定を表示 |
| <input type="checkbox"/> | 申請者 | 申請者 | | フォーマット設定を表示 |
| <input type="checkbox"/> | 申請日 | 申請日 | | フォーマット設定を表示 |
| <input type="checkbox"/> | 最終更新者 | 最終更新者 | | フォーマット設定を表示 |
| <input type="checkbox"/> | 最終更新日 | 最終更新日 | | フォーマット設定を表示 |
| <input type="checkbox"/> | 承認完了日 | 承認完了日 | | フォーマット設定を表示 |
| <input type="checkbox"/> | 承認状況 | 承認状況 | | フォーマット設定を表示 |
| <input type="checkbox"/> | 添付ファイル有無 | 添付ファイル有無 | | フォーマット設定を表示 |
| <input type="checkbox"/> | 書類No | 書類No | | フォーマット設定を表示 |
| <input checked="" type="checkbox"/> | 審査主旨 | 審査主旨 | テキストエリア | フォーマット設定を表示 |
| <input checked="" type="checkbox"/> | 契約書番号 | 契約書番号 | 文字フィールド | フォーマット設定を表示 |
| <input checked="" type="checkbox"/> | 契約書名 | 契約書名 | 文字フィールド | フォーマット設定を表示 |
| <input checked="" type="checkbox"/> | 申請日 | 申請日 | 文字フィールド | フォーマット設定を表示 |
| <input checked="" type="checkbox"/> | 申請者 | 申請者 | 文字フィールド | フォーマット設定を表示 |
| <input checked="" type="checkbox"/> | 審査No | 審査No | 文字フィールド | フォーマット設定を表示 |

- ※ kintone に追加を行う場合は、レコード ID を出力項目に含めない、
もしくはフィールド初期値が空欄になるようにフォームを作成してください。

8.2.4. クエリ CSV 出力

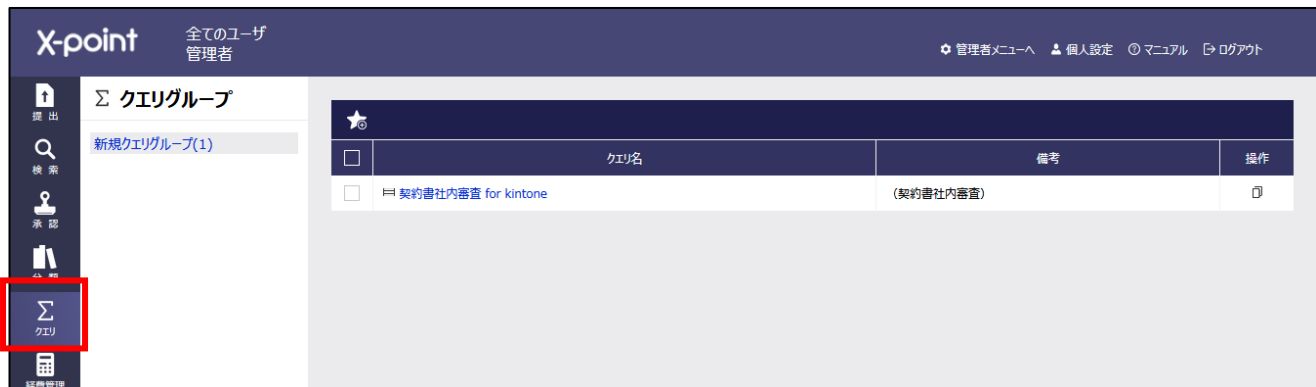
クエリによるCSV出力は、クエリ実行画面、APサーバー内で実行する「クエリCSV自動出力」タスク、クライアントPCなどから利用する「リモート出力」により、定義されたクエリを実行し取り出します。

本項では実行例を示します。各機能の詳細は管理者マニュアル、リモート出力マニュアルを参照してください。

8.2.4.1. クエリ CSV 出力（ユーザーサイト）

クエリ実行画面（ブラウザ）からのダウンロードを行います。

■クエリ画面を表示します。



■クエリ画面を表示します。

クエリ画面からCSVファイル出力を行うクエリを実行し、出力予定のデータ一覧を表示したのちに

【CSV出力】ボタンをクリックします。



8.2.4.2. クエリ CSV 出力（管理サイト・タスク管理）

「管理サイト」>「タスク管理」>「クエリ CSV 出力」を利用し、クエリ CSV ファイルの自動出力を実行することができます。
「クエリ CSV 出力」を利用することで、X-point に登録されているクエリ CSV 出力を決まった時間に実行することができます。
作成された CSV ファイルは「クエリ CSV 出力」>「データファイル」より、任意のタイミングでダウンロードできます。
詳細は「管理者機能マニュアル ～クエリ CSV 自動出力～」をご覧ください。

8.2.4.3. リモート出力

クライアント PC 等に導入したリモート出力コマンドを実行し CSV ファイルを取得します。

※ 参考マニュアル「リモート出力機能 導入・設定ガイド」

【Windows の場合】

リモート出力用のコマンドを“c:¥xpoint-client”に導入した場合。

```
C:¥> cd c:¥xpoint-client  
C:¥xpoint-client>run_query.bat -target KeiyakuShinsa4kintone -dir C:¥xpoint-client¥csvfile
```

【Linux の場合】

リモート出力用のコマンドを“/usr/local/xpoint/xpoint-client”に導入した場合。

```
[root@xpoint xpoint-client]# cd /usr/local/xpoint/xpoint-client  
[root@xpoint xpoint-client]# ./run_query.sh -target KeiyakuShinsa4kintone -dir /usr/local/xpoint/xpoint-client/csv
```

8.2.5. X-point マスタの登録

kintone アプリに登録されているデータを X-point の簡易マスタ、固有マスタに登録します。

■ 登録の手順

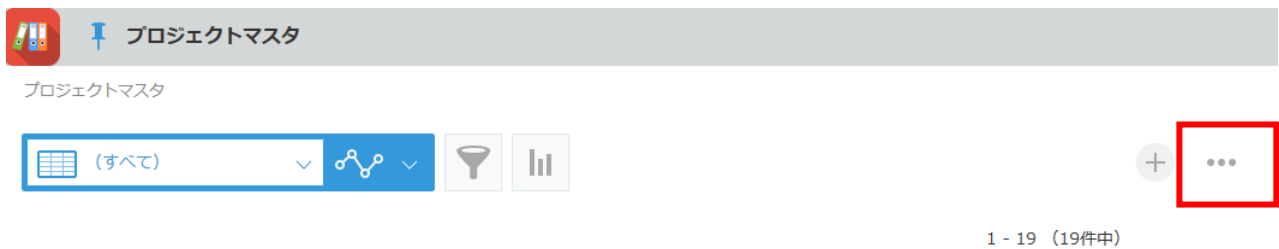
X-point マスタへの登録は kintone アプリのファイル書き出し機能、もしくはコマンドラインツールを利用して行います。本項ではアプリ画面のファイル書き出し機能を利用してファイルを出力するまでの説明を行います。X-point への取り込みは、X-point の管理者マニュアルに従って作業を行ってください。

- ① kintone でアプリのデータを「ファイル書き出し」で CSV ファイル化する
- ② X-point のマスタにデータを登録する
 - 固有マスタの場合は予め X-point 上で TABLE を定義し「取り込み済み」を行い利用可能な状態にしておきます。
 - 固有マスタは「ユーザ固有マスタインポート」より ① で作成したファイルを取り込みます。
 - 簡易マスタは「簡易マスタインポート」より ① で作成した CSV ファイルを取り込みます。

コマンドラインツールの利用は「kintone アプリから自動申請用の CSV を出力」の章を参照してください。

■ アプリ登録データの CSV ファイル出力

アプリのレコード一覧画面を表示し「オプション」をクリックします。



プロジェクトマスタ

プロジェクトマスタ

(すべて) [Filter] [Chart]

1 - 19 (19件中)

| レコード番号 | コード | 名称 | |
|--------|------|-----|-----------------|
| 19 | P999 | その他 | [Edit] [Delete] |

表示されたメニューより「ファイルに書き出す」をクリックします。



プロジェクトマスタ

プロジェクトマスタ

(すべて) [Filter] [Chart]

1 - 19 (19件中)

| レコード番号 | コード | 名称 | |
|--------|------|-----|-----------------|
| 19 | P999 | その他 | [Edit] [Delete] |

- 表示件数 >
- 先頭行を固定表示
- ファイルに書き出す**
- ファイルから読み込む
- アプリの設定を変更

出力画面にて CSV ファイルの書き出す列名（フィールド名）を指定します。

フィールド名の指定が完了したら【書き出す】をクリックします。

| レコード番号 | コード | 名称 |
|--------|-----|-----|
| 文字列 | 文字列 | 文字列 |
| 文字列 | 文字列 | 文字列 |
| 文字列 | 文字列 | 文字列 |
| 文字列 | 文字列 | 文字列 |
| 文字列 | 文字列 | 文字列 |

※ 出力するフィールドは固有マスタに取り込む項目を指定します。

※ 簡易マスタはコード、名称のみの扱いです。2項目のみ指定します。

| レコード番号 | 名称 |
|--------|-----|
| 文字列 | 文字列 |
| 文字列 | 文字列 |
| 文字列 | 文字列 |
| 文字列 | 文字列 |
| 文字列 | 文字列 |

出力用のファイルが出来上がると一覧が表示されます。

| ファイル名 | アプリ名 | 作成開始日時 | 削除される日時 | |
|--|-----------|------------------|------------------|---|
| プロジェクトマスタ_20170117T174914+0900.csv (684 bytes) | プロジェクトマスタ | 2017-01-17 17:49 | 2017-01-20 17:49 | ✕ |
| プロジェクトマスタ_20170117T171448+0900.csv (583 bytes) | プロジェクトマスタ | 2017-01-17 17:14 | 2017-01-20 17:14 | ✕ |

| アプリ名 | 作成開始日時 | 状況 | 詳細 |
|------------|--------|----|----|
| データがありません。 | | | |

※ ダウンロード可能なファイルが2点ある場合の例です。

目的のファイル名をクリックしダウンロードを行います。

プロジェクトマスタ 出力されたファイル

再読み込み プロジェクトマスタ

ダウンロード可能なファイル

| ファイル名 | アプリ名 | 作成開始日時 | 削除される日時 | |
|--|-----------|------------------|------------------|--------------------------|
| <input checked="" type="checkbox"/> プロジェクトマスタ_20170117T174914+0900.csv (684 bytes) | プロジェクトマスタ | 2017-01-17 17:49 | 2017-01-20 17:49 | <input type="checkbox"/> |
| <input type="checkbox"/> プロジェクトマスタ_20170117T171418+0900.csv (589 bytes) | プロジェクトマスタ | 2017-01-17 17:14 | 2017-01-20 17:14 | <input type="checkbox"/> |

作成中のファイル

| アプリ名 | 作成開始日時 | 状況 | 詳細 |
|------------|--------|----|----|
| データがありません。 | | | |

gnify.cybozu.com から プロジェクトマスタ_20170117T174914+0900.csv を開くか、または保存しますか?

Copyright (C) 201... ファイルを開く(O) 保存(S) キャンセル(C) 100%

ダウンロードされたファイルの内容 (例)

"レコード番号","コード","名称"
"19","P999","その他"
"18","P018","ECO プロジェクト"
"17","P017","社内経費削減プロジェクト"
"16","P016","スマイリープロジェクト"
"15","P015","エベレストプロジェクト"
"14","P014","業務改善プロジェクト"
"13","P013","景気回復プロジェクト"
"12","P012","労務改善プロジェクト"
"11","P011","新規拡販プロジェクト"
"10","P010","社員教育プロジェクト"
"9","P009","内部統制プロジェクト"
"8","P008","自社製品開発プロジェクト"
"7","P007","新規事業推進プロジェクト"
"6","P006","リーサルウェポン"
"5","P005","プロジェクトX"
"4","P004","ボランティアプロジェクト"
"3","P003","人材向上プロジェクト"
"2","P002","事務所移転プロジェクト"
"1","P001","モバイル推進プロジェクト"

第9章. 連携API仕様

9.1. フィールドの対応

X-point と kintone のフィールド種類について比較します。

データ転送時のエラー判定は受信側で行います。つまり、X-point から kintone に文字列を送信する際、64 文字を越えた文字列を送信できますが、kintone が登録を行う際に文字長の溢れをエラーとして表示します。逆の場合も同様に、kintone から X-point への送信では X-point 側で登録エラーを発生させます。

【対応表】・・・「連携」列はデータ送信可能方向を示します (⇔双方向、←X-point のみ送信可、→kintone のみ送信可)

| X-point | 連携 | kintone | 備考 |
|-------------------|----|--------------------------------|--|
| 文字フィールド | ⇔ | 文字列 (1行) * SINGLE_LINE_TEXT | 最大文字列 ・ X-point=4000Byte (指定がある場合は制限値まで) ・ kintone=重複禁止の場合 64 文字 (全半角問わず) 重複可の場合は制限無し |
| テキストエリア | ← | リッチエディター * RICH_TEXT | X-point に渡すデータはタグ部分が除去されます。“<”, “>” で囲まれた部分は全て削除されるため、必要以上に削除される場合があります。不都合ある場合は連携ライブラリでの変換を利用せず、JavaScript で変換後 “value” で値を渡す ・ X-point=4000Byte ・ kintone=制限無し |
| テキストエリア | ← | 文字列 (複数行) * MULTI_LINE_TEXT | 最大文字列 ・ X-point=4000Byte ・ kintone=制限無し |
| パスワード | → | 文字列 (1行) * SINGLE_LINE_TEXT | 対応項目が無いため kintone では可視状態になる ・ kintone で文字フィールド等に格納するとパスワードが可視状態になる |
| 数値フィールド | ⇔ | 数値 * NUMBER | 最大文字列 ・ kintone=全体 30 桁、小数部 10 桁 (初期整数部 12 桁) |
| 整数フィールド | ⇔ | 数値 * NUMBER | 最大文字列 ・ X-point=96 文字 |
| 数値、整数、 文字フィールド | ← | 計算 * CALC | kintone は計算結果がセットされているため “N/A” である場合以外は X-point に値を格納する X-point から kintone への転送はできません |
| 西暦 | → | 文字列 (1行) * SINGLE_LINE_TEXT | kintone では 文字列フィールドに格納 |
| 月 | → | 文字列 (1行) * SINGLE_LINE_TEXT | kintone では 文字列フィールドに格納 |
| 日 | → | 文字列 (1行) * SINGLE_LINE_TEXT | kintone では 文字列フィールドに格納 |
| 曜日 | → | 文字列 (1行) * SINGLE_LINE_TEXT | kintone では 文字列フィールドに格納 |
| ラジオボタン | ⇔ | ラジオボタン * RADIO_BUTTON | kintone では表示文字列と値が同じ ・ X-point=4000byte ・ kintone=格納値 128 文字 |
| チェックボックス | ⇔ | チェックボックス * CHECK_BOX | kintone では表示文字列と値が同じ ・ X-point=4000byte ・ kintone=格納値 128 文字 ※kintone はグループ扱いですが、X-point は1個毎の扱い 対応付けは kintone のチェックボックスをグループ毎に1個の状態にする |

| X-point | 連携 | kintone | 備考 |
|---|----|------------------------------|---|
| リストボックス | ⇔ | 複数選択 *MULTI_SELECT | kintone では表示文字列と値が同じ ・ X-point=4000byte ・ kintone=格納値 128 文字 ※ X-point に複数選択を転送することはできません ※ kintone で複数選択している場合は先頭の 1 個だけ転送 |
| コンボボックス | ⇔ | ドロップダウン *DROP_DOWN | kintone では表示文字列と値が同じ ・ X-point=4000byte ・ kintone=格納値 128 文字 |
| 日時 | ⇔ | 日付 *DATE | kintone から： yyyy/mm/dd で日時フィールドに変換される X-point から： 日時フィールドは “yyyy-mm-dd” に変換される “yyyy/mm/dd”, “yy/mm/dd” が対応。 ※ kintone への設定を ISO UTC 形式で行うとエラーが発生 |
| | ⇔ | 時刻 *TIME | kintone から： hh:mi で日時フィールドに設定される X-point から： 日時フィールドは “hh:m” に変換される “hh:mm”, “hh:mm:ss”, “hhh:mm”, “hhh:mm:ss” が対応 ※ kintone への設定を ISO UTC 形式で行うとエラーが発生 ※ kintone へ “24:00” 以上の値を連携するとエラーが発生 (kintone の「時刻」は “24:00” 以上の値を取り扱うことができない) |
| 文字 (yyyy) ” yyyy/mm/dd HH:mm” ” yyyy-mm-dd HH:mm” | ← | 日時 *DATETIME | kintone から： ISO UTC 形式を転送します。 書式指定が必要な場合予め kintone 上で行う X-point から： “yyyy/mm/dd HH:mm”, “yyyy-mm-dd HH:mm” の場合に転送できる |
| 添付ファイル | ⇔ | 添付フィールド | X-point はナビボタンになります。 X-point から kintone への送信は単一の添付フィールドのみ |
| 文字フィールド | ⇔ | リンク *LINK | X-point には リンク先が文字列で転送される |
| 文字フィールド | ← | ユーザー選択 *USER_SELECT | kintone→X-point のみ 表示名 (コード) で送信される 複数選択されている場合は “,” で区切られる |
| 文字フィールド | ← | カテゴリ *CATEGORY | |
| 文字フィールド | ← | 組織選択 *ORGANIZATION_SELECT | |
| 文字フィールド | ← | グループ選択 *GROUP_SELECT | |
| － (該当無し) － | × | 関連レコード一覧 | 変換されない |
| 文字フィールド 数値、整数 | ← | ルックアップ | Kintone→X-point のみ。 kintone の SINGLE_LINE_TEXT または、NUMBER と同様になる |
| 文字フィールド | ← | レコード番号 *RECORD_NUMBER | X-point は代替として文字フィールドになる |
| 文字フィールド | ← | 作成者 *CREATOR | kintone→X-point のみ。(X-point は代替) 表示名 (コード) |
| 文字フィールド | ← | 作成日時 *CREATED_TIME | kintone 側は ISO8601 (RFC3339) 表記、 X-point 側では JST-9 “YYYY/MM/DD hh:mm:ss” で表記する |
| 文字フィールド | ← | 更新者 *MODIFIER | kintone→X-point のみ。 表示名 (コード) |
| 文字フィールド | ← | 更新日時 *UPDATED_TIME | kintone 側は ISO8601 (RFC3339) 表記、 X-point 側では JST-9 “YYYY/MM/DD hh:mm:ss” で表記 |
| － (該当無し) － | × | ステータス *STATUS | 変換されない 取得時のレコード状態 |

※ フィールドの入力制限 (https://help.cybozu.com/ja/k/user/field_restriction.html)

※ フィールド名 : kintone = 128 文字 (全半角とわず)、X-point = 128 文字

※ フィールドコード: kintone = 128 文字 (全半角とわず)、X-point = 30 文字 (英数字)

※ リッチテキストをテキストエリアに移す際の HTML タグ除去ルール (正規表現で実施)

- ・
,
, </div>, の後 … 改行に置き換えられます。
- ・ “<”, “>” で囲まれた文字列 … 全て除去されます。

同様の形式で記述されている部分は HTML 用のタグでなくても除去されます。

9.2. kintone 連携 「JavaScript 仕様」

9.2.1. kintone 組み込みライブラリ仕様

kintone から X-point への連携は「サイボウズ kintone 連携サービス」ライブラリ内に用意されている「kintone2xpoint」クラスを利用して行います。kintone からの連携は2種類の動作が行われます。

① X-point への書類申請

kintone にログインしているユーザーのログイン名と X-point の登録ユーザーコードが一致するユーザーを探し、一致したユーザーで書類申請を行います。一致するユーザーが存在しない場合は、オプション項目の指定に従って動作します。

② X-point に登録されている書類を開く

実行時に指定された条件で X-point の登録書類を絞りこみ、一意に特定できた場合に書類ビューを開きます。

但し、書類内容の閲覧には X-point に登録されているユーザーアカウントでの認証、及び認証に使用したユーザーに開こうとする書類の閲覧権限がある場合に限り書類の内容を確認することができます。

【X-point に書類を申請する例】

```
var domCd      = " (書類を送信する X-point のドメインコード) ";
var fqdn       = " (X-point のサーバーFQDN) ";
var formcd     = " (申請書類のフォームコード) ";
var routecd   = " (申請する承認ルートコード) ";
var api_token  = " (申請に利用する kintone 連携の API トークン) ";
var options    = { "proxyusercd": " (kintone ログイン名に相当するユーザーがない場合のユーザーコード) ",
                  };

var k2x = new kintone2xpoint(domCd, fqdn, formcd, routecd, api_token, options);
      :
      (書類の内容を作成する)
      :
k2x.add_attach_files(body_json, record['attached_file']); // 添付ファイルの filekey 情報を追加

k2x.send(body_json); // X-point に書類データを送信
```

【X-point の書類を専用ビューで開く例】

```
var domCd      = " (書類を送信する X-point のドメインコード) ";
var fqdn       = " (X-point のサーバーFQDN) ";
var formcd     = " (申請書類のフォームコード) ";
var routecd   = " (申請する承認ルートコード) ";
var api_token  = " (申請に利用する kintone 連携の API トークン) ";

var k2x = new kintone2xpoint(domCd, fqdn, formcd, routecd, api_token);
// 表示中のレコード ID を渡し、同じ ID を持つ書類を特定して表示
k2x.openDoc( { "kintone_id" : kintone.app.record.getId() } );
```

9.2.1.1. 連携クラス「kintone2xpoint」の仕様

■コンストラクタ

```
kintone2xpoint( domCd, fqdn, formcd, routecd, api_token, options )
```

【引数の説明】

| 項目名 | 内容 | 必須 | 説明 | |
|-----------|---------------------|------------|--------------------------------|--|
| domCd | ドメインコード | 必須 | 連携 X-point のドメインコード | |
| fqdn | X-point AP サーバーFQDN | 必須 | 連携 X-point の AP サーバーが動作する FQDN | |
| formcd | フォームコード | 必須 | 連携対象のフォームコード | |
| routecd | 承認ルートコード | 必須 | 連携対象のフォームが利用する承認ルートコード | |
| api_token | X-point 連携 API トークン | 必須 | 連携対象へのアクセス権を示す API トークン | |
| options | usercd | ユーザーコード | 任意 | kintone のログインユーザー名に関係なく、指定された X-point ユーザーコードに該当するユーザーで書類申請を行う。(既定値：指定なし) |
| | proxyusercd | 代理ユーザー | 任意 | kintone のログインユーザーに該当する X-point ユーザーコードが無い場合に代理ユーザーコードを利用して申請を行う。(既定値：指定なし) |
| | comment | コメント転送 | 任意 | 書類申請を行う際に kintone のコメントを転送。コメント転送する = true 転送しない = false (既定値：false) |
| | guest | ゲストスペース ID | 任意 | 連携元の kintone アプリがゲストスペースに配置される場合にゲスト ID (数値) を指定。(既定値：指定なし) |
| | path | パス | 任意 | X-point の URI パスが「/xpoint」以外である場合に指定。(既定値：/xpoint) |
| | protocol | プロトコル | 任意 | X-point との通信が https 以外である場合に指定。(既定値：https) |

• オプション項目の指定例

```
var options = { "proxyuser": "u001", "comment": true, "guest": 15 };
var k2x = new kintone2xpoint( domCd, fqdn, formcd, routecd, token, options );
```

- オプション項目でユーザーコード (usercd) が指定されている場合、代理ユーザ (proxyusercd) の指定は無視されます。
- オプション項目のプロトコルを利用することは通常ありません。kintone が https で動作する状態で、kintone 内に http を利用した動作を組み込むことはブラウザセキュリティ上禁止されているためです。

■メソッド概要

| 戻り値 | メソッド名 | 説明 |
|----------------|---|---|
| JSON Object | create_recordset(record [, appId]) 【パラメータ】 <ul style="list-style-type: none"> record kintone event に含まれる record プロパティ値 appId kintone event に含まれる appId プロパティ値 | X-point 送信用データの初期設定 ※ 引数「appId」は PC 環境のみで利用する場合は省略可 ※ モバイル環境での利用は必須 |
| 無し | add_elements(json, record, elements) 【パラメータ】 <ul style="list-style-type: none"> json create_recordset が生成した JSON Object record kintone event に含まれる record プロパティ elements 転送を行うフィールド名対比情報 | フィールド(表明細以外)を追加 *elements <pre>[{ "field": "CA", "xname": "CA" }, { "field": "CC", "value": cc_value }, { "field": "CC", "value": cc_value, "page": 2 }]</pre> |
| JSON Object | create_table_porperty(json, table_name) 【パラメータ】 <ul style="list-style-type: none"> json create_recordset が生成した JSON Object table_name テーブル ID 名 | 表明細を格納するプロパティを指定名で生成します。生成した Object 内に add_table_row で作成する行データを格納 <pre>{ "type": "SUBTABLE", "value": [] }</pre> |
| JSON Object | add_table_row(table_json) 【パラメータ】 <ul style="list-style-type: none"> table_json 指定 TABLE の value に行 Object({})を追加する | 表明細のカラムを格納するための配列を返す Object を作る ※ 作成した行 Object の中に、X-point と kintone の項目対比情報を作成 |
| 無し | add_table_elements(tbl_json, tableData, elements) 【パラメータ】 <ul style="list-style-type: none"> tbl_json add_table_row 作成のテーブル行 Object tableData create_table_porperty で作成した追加を行う kintone の TABLE データ elements 転送を行うフィールド名対比情報 | フィールド(表明細部)を追加 *elements <pre>[{ "field": "CA", "xname": "CA" }, { "field": "CC", "value": cc_value }]</pre> ※ ページ番号の指定は無し |
| 無し | add_attach_files(json, fieldVals) 【パラメータ】 <ul style="list-style-type: none"> json create_recordset が生成した JSON Object fieldVals 添付ファイル情報の配列 添付ファイル情報 att_file[]. contentType att_file[]. fileKey att_file[]. name att_file[]. size | 添付ファイルは X-point に送る際、一つの項目にまとめて情報を送信 * 複数の添付ファイルフィールドがある場合は複数回 CALL し追加を行う <pre>k2x.add_attach_files(json, record[' att1']); k2x.add_attach_files(json, record[' att2']);</pre> |
| 無し | clear_attach_files(json) 【パラメータ】 <ul style="list-style-type: none"> json create_recordset が生成した JSON Object | 添付ファイルの格納情報をクリア (破棄) |
| 無し | add_record_property(json, col_name, fieldVals) 【パラメータ】 <ul style="list-style-type: none"> json col_name fieldVals fieldVals.type fieldVals.value | 送信用の情報セットを作成 (ID 名 : プロパティ項目、設定内容 : 型&値) |

| 戻り値 | メソッド名 | 説明 |
|----------------|--|--|
| 無し | send(json) 【パラメータ】 • json 申請書類データ | X-point に書類の登録・更新データを送信 |
| 無し | openDoc(書類 ID) 【パラメータ】 • 書類 ID | 指定 ID の書類をブラウザで表示 |
| 無し | openDoc(query_values) 【パラメータ】 • query_values 絞込み条件の指定 { "フィールド ID1": "値 1", "フィールド ID2": "値 2", : "フィールド IDn": "値 n" } | 指定書類をブラウザで表示 |
| JSON Object | getDocumentStatus(書類 ID) 【パラメータ】 • 書類 ID 【戻り値】 ※ 戻り値サンプルを参考にしてください。 | 指定された書類の承認状況を JSON 形式で返す |
| JSON Object | getWorkflowCount(フォームグループ名) 【パラメータ】 • フォームグループ名 | ユーザフロント画面の承認がジェットでフォームグループ、フォームを指定した際に表示される内容と同じものを JSON 形式で返す |

9.2.2. X-point 連携ライブラリ仕様

X-point から kintone への連携は「サイボウズ kintone 連携サービス」ライブラリとして用意されている「xpoint2kintone」クラスを利用して行います。

9.2.2.1. 連携クラス「xpoint2kintone」の仕様

■コンストラクタ

```
xpoint2kintone( subdomain, app_id, api_token, options )
```

【引数の説明】

| 項目名 | 内容 | 必須 | 説明 |
|-----------|------------------|----|--|
| subdomain | kintone FQDN | 必須 | ドメインが". cybozu. com"である場合の、連携する kintone のサブドメイン名 |
| app_id | kintone アプリ ID | 必須 | 連携する kintone のアプリ ID |
| api_token | kintone API トークン | 必須 | 連携する kintone アプリの API トークン |
| options | userid | 任意 | ユーザー認証で連携する場合に kintone のログイン名を設定 (既定値: ログイン中ユーザーのログイン名) |
| | passwd | 任意 | ユーザー認証で連携する場合に kintone のパスワードを平文で設定 (既定値: 指定なし) |
| | fulldomain | 任意 | ドメインが". cybozu. com"以外である場合、サブドメインを含めフルドメイン形式で指定 (例: renkei. kintone. com) 指定する場合は subdomain に空文字列[""]を記述します。 |
| | guest | 任意 | 連携元の kintone アプリがゲストスペースに配置される場合にゲスト ID (数値) を指定 (既定値: 指定なし) |

- オプション項目の指定例

```
var options = { "userid": "u001", "passwd": "tarou", "guest": 15 };
var k2x = new xpoint2kintone( subdomain, app_id, api_token, options );
```
- options.userid は指定の有無と無関係に、X-point に現在ログインしているユーザーID が指定されます。
- ユーザー認証で動作させる場合、userid, passwd の両項目を同時に指定する必要があります。

■メソッド概要

| 戻り値 | メソッド名 | 説明 |
|----------------|--|--|
| JSON Object | create_recordset() 【パラメータ】 | kintone にレコード追加を行う際の、 送信用データの初期設定を行う |
| JSON Object | create_recordset(recid) 【パラメータ】 • recid 更新を行うレコード ID | kintone に登録されているレコードの更新を行 う際の、送信用データの初期設定を行う |
| JSON Object | create_recordset(recid, revision) 【パラメータ】 • recid レコード ID • revision レコードレビジョン | kintone に登録されているレコードの更新で対 照となるレビジョンも指定する際の、送信用デ ータの初期設定を行う |
| 無し | add_elements (json, elements) 【パラメータ】 • json create_recordset で作成した JSON Object • elements 転送先フィールド ID、転送元フィールド ID もしくは 設定値の情報 | 表定義以外の部分に設定するフィールドの値を 設定する。 *elements { "field": "date", "value": x_date }, { "field": "group", "xname": "name", "page": pgno }] |
| JSON Object | create_table_porperty (json, table_name) 【パラメータ】 • json create_recordset で作成した JSON Object • table_name テーブルのフィールドコード | 表部分に転送するためのプロパティを作成する |
| JSON Object | add_table_row (table) 【パラメータ】 • table create_table_porperty で作成した JSON Object | テーブル Object に新規行データを追加します |
| 無し | add_table_elements (tbl_row, tableData, rowno, elements) 【パラメータ】 • tbl_row add_table_row で作成した JSON Object • tableData X-point の表グループを示す XPObject • rowno kintone への送信情報を作る X-point の表処理行 • elements 転送先フィールド ID、転送元フィールド ID もしくは 設定値の情報 | 指定フィールド、固定値の転送情報を作成しま す。add_elements と異なり、ページ番号は指定 しません。ページは tableData 内で既に指定さ れているためです。 *elements [{ "field": "CAA", "value": v_caa }, { "field": "CBB", "xname": "CBB"}] |
| JSON Object | send_attach_files=function(json, attached_file_id) 【パラメータ】 • json create_recordset で作成した JSON Object • attached_file_id 添付ファイルを設定する kintone 側フィールドコー ド | 添付ファイルを kintone に転送し、kintone 上 のファイルキー文字列を転送用 JSON オブジェ クト内に設定します。 ※ 戻り値はエラー発生時のみ。 ※ 正常処理時の戻り値は undefined |

| 戻り値 | メソッド名 | 説明 |
|---------|--|--|
| Boolean | send_kintone(json, f_kintone_id) 【パラメータ】 <ul style="list-style-type: none"> • json create_recordset で作成した JSON Object • f_kintone_id レコード ID を設定するフィールド ID 名 | kintone のレコード追加・更新データを送信する 登録後に応答として戻される kintone のレコード ID を“f_kintone_id”が示すフィールドにセット |
| Boolean | send_kintone(json) 【パラメータ】 <ul style="list-style-type: none"> • json create_recordset で作成した JSON Object | kintone のレコード追加・更新データを送信 |
| Boolean | send_kintone(json, callback_func) 【パラメータ】 <ul style="list-style-type: none"> • json create_recordset で作成した JSON Object • callback_func 処理完了時の後処理プログラム | kintone のレコード追加・更新データを送信する kintone の record.json API 仕様に基づく応答が call_back 関数に返されます。処理コードを記述する |
| 無し | delete_kintone(recid, callback_func) 【パラメータ】 <ul style="list-style-type: none"> • json create_recordset で作成した JSON Object • callback_func 処理完了時の後処理プログラム（省略可） | kintone の登録レコードを削除する ※ call_backfunc を省略した場合、 連携処理共通エラー画面が表示される |
| 無し | send_comments(kintone_id) 【パラメータ】 <ul style="list-style-type: none"> • json create_recordset で作成した JSON Object | 書類に記入されているコメントの内容全てを kintone の指定レコードに送信する |
| 無し | search_kintone(query_text, fields, callback_func) 【パラメータ】 <ul style="list-style-type: none"> • query_text アプリ内の登録レコードを検索するための条件式を記述する • fields X-point に戻す値 • callback_func 処理完了時の後処理プログラム（省略可） | 指定 kintone アプリの登録レコードを検索し、 条件に一致するレコードの内容を書類内に転記 (マスタ検索相当の動作を行います) ※ call_backfunc を省略した場合、 連携処理共通エラー画面が表示する |
| String | ISODateString | Date 型 Object を UTC 形式の文字列で返す ※ kintone に送信可能な形式に変換する。 |
| Date | ISODateParse | UTC 形式の日付書式を Date 型に変換する ※ kintone 内の日付、日時は UTC 型 |
| Date | date_xp2kintone | yyyy 年 MM 月 dd 日 HH 時 mm 分、 yyyy 年 MM 月 dd 日 等の X-point 独自書式の文字列を解析し、Date 型に変換 変換できない場合は元の文字列が返される |

9.3. kintone 連携「X-point REST API 仕様」

X-point 内に用意された REST API の仕様を記述します。kintone に組み込みを行う JavaScript プログラムより、X-point の自動申請などを実行する API と、X-point の書類をブラウザ表示する際に、kintone の REST API がフォーム JavaScript から利用できるようにした中継 I/F から構成されます。

※ 本項に記載の REST API 形式データは、kintone 用、X-point 用に用意された JavaScript ライブラリを利用することで自動的に生成されます。本項記載の REST API を直接呼び出すことは基本的な動作を実現する上で利用の必要はありません。

9.3.1. kintone → X-point

kintone より X-point に対して書類の登録、閲覧を行う API です。

9.3.1.1. 自動申請 (HTTP メソッド : POST)

kintone から X-point に書類の追加権限のあるユーザーが 1 件の書類を追加登録します。

■URI

https://{FQDN}/xpoint/xrh/{ドメイン CD}/doEntry

■アクセス権

kintone 連携設定のアクセス権で「追加」の設定が必要です。

■リクエストプロパティ

| パラメータ名 | 指定する値 | 説明 |
|--------|------------------|------------------------------------|
| json | 登録内容を示す JSON 文字列 | 登録を行うレコードの内容を JSON 形式で表した内容を指定します。 |

■パラメータ「json」の項目について

| 項番 | パラメータ | 論理名 | 型 | 階層 | 必須 | 備考 |
|----|-------------|---------------------|--------|----|----|--|
| 1 | usercode | kintone ログイン名 | String | 1 | ○ | |
| 2 | fcd | 自動申請先フォーム CD | String | 1 | ○ | |
| 3 | routecode | 自動申請先承認ルート CD | String | 1 | ○ | |
| 4 | token | X-point API トークン CD | String | 1 | ○ | |
| 5 | app_id | kintone アプリ ID | String | 1 | ○ | |
| 6 | proxyusercd | 代用ユーザーコード | String | 1 | ○ | usercode の指定と同一ユーザー CD が X-point に無い場合に使うユーザーコード |
| 7 | comments | kintone コメント転記有無 | String | 1 | | 転記する : true |

| | | | | | | |
|----|------------------|--------------------|--------|---|---|----------------------|
| | | | | | | 転記なし：省略または false |
| 8 | guest_space_id | kintone ゲストスペース ID | String | 1 | | ゲストスペースの場合のみ指定 |
| 9 | record | | Object | 1 | | |
| 10 | record_number | kintone レコード番号 | String | 2 | ○ | |
| 11 | updated_time | kintone 更新日時 | String | 2 | | YYYY/MM/DD HH:MI:SS |
| 12 | created_time | kintone 作成日時 | String | 2 | | YYYY/MM/DD HH:MI:SS |
| 13 | revision | レビジョン番号 | String | 2 | ○ | |
| 14 | id | kintone ID 番号 | String | 2 | ○ | |
| 15 | X-point フィールド ID | | Object | 2 | | kintone フィールド CD に相当 |
| 16 | type | kintone での TYPE 名 | String | 3 | | ※ |
| 17 | value | 設定値 | 可変 | 3 | | |

※cybozu developer network 【フィールド形式】参照

<https://developer.cybozu.io/hc/ja/articles/202166330>

パラメータ “json” の内容

```

{
  "usercode"      : (kintone ログイン名) ,
  "fcd"          : (自動申請先フォーム CD) ,
  "routecode"    : (自動申請先承認ルート CD) ,
  "token"        : (X-point API トークン CD) ,
  "app_id"       : (kintone アプリ ID) ,
  "proxyusercd" : (usercode の指定と同一ユーザーCD が X-point に無い場合に使うユーザーコード) ,
  "comments"     : (kintone のコメントを転記する場合は true、転記なしは省略または false) ,
  "guest_space_id" : (kintone ゲストスペース ID、ゲストスペースの場合のみ指定する) ,
  "record": {
    "$record_number" : (kintone レコード番号、必須) ,
    "$updated_time"  : (kintone 更新日時、YYYY/MM/DD HH:MI:SS 形式) ,
    "$created_time"  : (kintone 作成日時、YYYY/MM/DD HH:MI:SS 形式) ,
    "$revision"      : (kintone レビジョン番号、必須) ,
    "$id"            : (kintone ID 番号、必須) ,

    "X-point フィールド ID" : { "type" : (kintone での TYPE 名), "value": (設定値) }
    "X-point フィールド ID" : { "type" : (kintone での TYPE 名), "value": (設定値) }
    "X-point フィールド ID" : { "type" : (kintone での TYPE 名), "value": (設定値) }
    "X-point フィールド ID" : { "type" : "SUBTABLE", "value": [
      {
        "value": [
          ←表 1 行目
          "列 1 フィールド ID" : { "type" : (kintone での TYPE 名), "value": (設定値) }
          "列 2 フィールド ID" : { "type" : (kintone での TYPE 名), "value": (設定値) }
          :
        ],
        "value": [
          ←表 2 行目
          "列 1 フィールド ID" : { "type" : (kintone での TYPE 名), "value": (設定値) }
          "列 2 フィールド ID" : { "type" : (kintone での TYPE 名), "value": (設定値) }
          :
        ]
      }
    ]
  }
}

```

9.3.1.2. 書類表示 (HTTP メソッド : GET)

kintone から X-point に登録されている書類を指定しブラウザに表示します。

但し、表示の際に書類を一意に決定できる条件の指定と閲覧可能なユーザーによるログインが必要になります。

■URI

https://{FQDN}/xpoint/xrh/{ドメイン CD}/openDoc

■アクセス権

kintone 連携設定のアクセス権で「閲覧」の設定が必要です。

■リクエストプロパティ

| パラメータ名 | 指定する値 | 説明 |
|--------|-----------------------|---|
| key | 書類表示ユーザーCD と Token | 書類表示を行うユーザーCD と閲覧を許可されたトークンを合成した文字列 |
| query | 書類指定条件を base64 に変換した値 | 閲覧対象書類を一意に決定するための条件 指定文字列は全体を base64 で encode した文字列を指定します。 |

key の作成方法

key = BASE64((Unix Epoch タイム) + " : " + ユーザーCD + API トークン)

パラメータ “query” 省略時に設定される内容

```
{
  "formcd" : (フォームコード)
}
```

推奨 “query” 設定される内容

```
{
  "kintone_id" : (レコード ID)
}
```

【例】

https://xpoint.atledcloud.jp/xpoint/openDoc.jsp?key=18e7927702f1d80219dec013a70c526502dcb9dea80ffa38

9.3.1.3. フォーム (TSF) 雛形ダウンロード (HTTP メソッド : GET)

kintone アプリ情報を指定し所定の URI にアクセスを行うとフィールド定義のみが行われた TSF ファイルの雛形がダウンロードできます。定義されているフォームに配置されたフィールドの ID には kintone のフィールドコードが設定されています。

■URI

https://{FQDN}/xpoint/xrh/{ドメイン CD}/getTsf

■リクエストプロパティ

| パラメータ名 | 指定する値 | 説明 |
|----------------|----------------------|---|
| app_id | kintone アプリ ID | TSF ファイルの雛形を作成するアプリの ID |
| api_token | kintone アプリ API トークン | アプリ権限が指定された API トークン |
| kintone_fqdn | kintone FQDN | kintone の FQDN ※ FQDN 後半が “. cybozu. com” の場合はサブドメインのみの指定が可能 |
| guest_space_id | ゲストスペース ID | ゲストスペースに配置されたアプリの場合に限り指定が必要 ※ 省略可 |

【例】

https://{X-point FQDN}/xpoint/xrh/trial/getTsf?

app_id=6&kintone_fqdn=ksubdomain&api_token=EgQX5LbKDLzd00EFcZqrD1dMp60rb6D7TBU9Va4C

■変換されるフィールドの種類

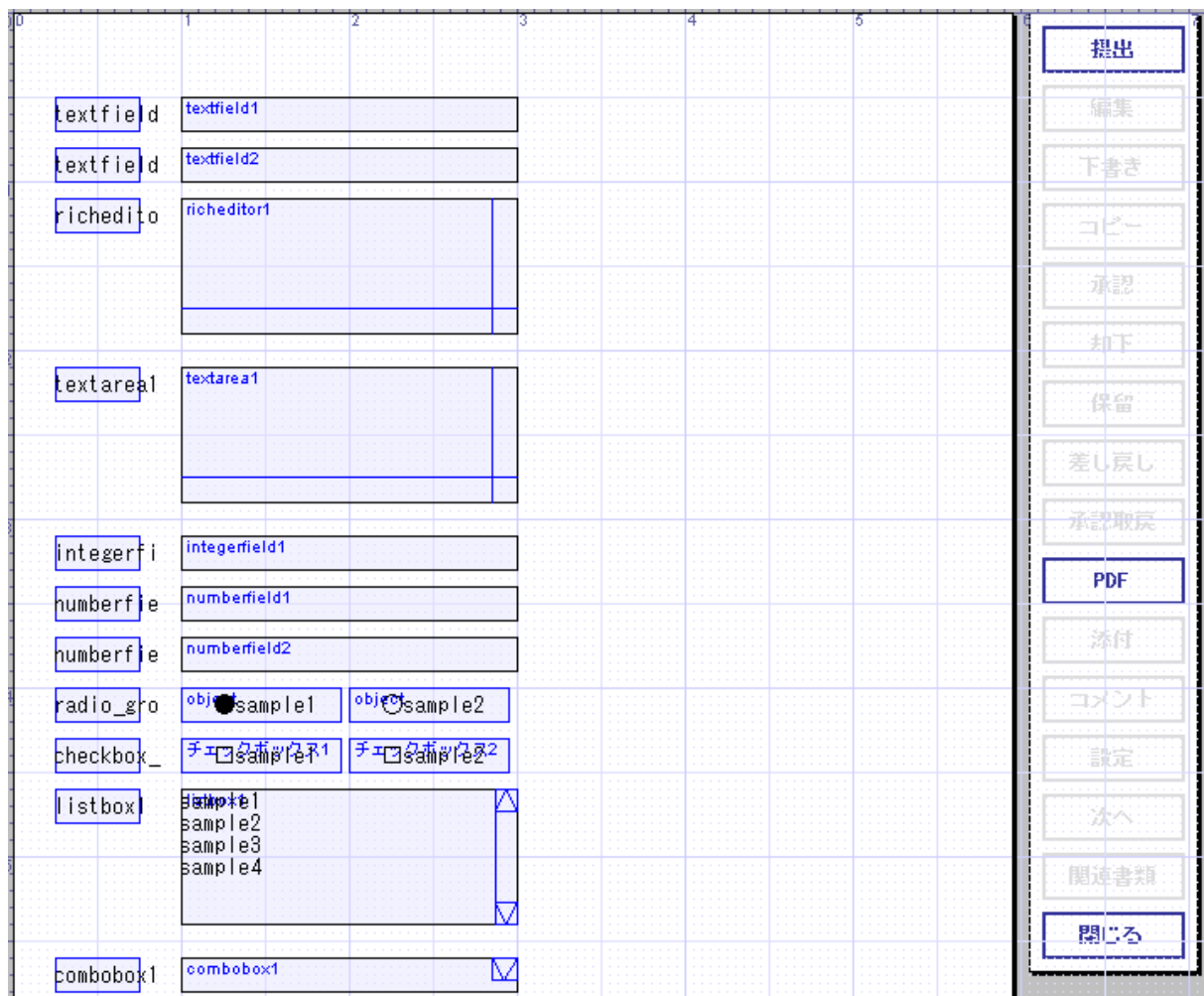
表に記載されたフィールドのみ TSF ファイルに変換出力されます。項目のサイズ、レイアウトなどは再現されません。
eFormMaker で配置などの調整を行って下さい。

| フィールド種類 | 変換後のタイプ | 説明 |
|----------|-------------|--------------------------------|
| ラベル | 文字ラベル | |
| 文字列（1行） | 文字フィールド | |
| リッチエディター | テキストエリア | |
| 文字列（複数行） | テキストエリア | |
| 数値 | 数値フィールド | |
| 計算 | 数値フィールド | 計算式はコピーされません |
| ラジオボタン | ラジオボタン・グループ | 個々のラジオボタンがグループ化された状態でフィールドを作成 |
| チェックボックス | 複数のチェックボックス | チェックボックス1個ずつに分解された状態でフィールドを作成 |
| 複数選択 | リストボックス | オプション値がコピーされ1列の定義でフィールドを作成 |
| ドロップダウン | コンボボックス | オプション値がコピーされ1列の定義でフィールドを作成 |
| 日付 | 日時フィールド | 書式「YYYY/MM/DD」の日時フィールドフィールドを作成 |
| 時刻 | 日時フィールド | 書式「hh:mm」の日時フィールド作成 |
| 日時 | 文字フィールド | 文字フィールドで作成（書式無し） |
| ユーザー選択 | 文字フィールド | 文字フィールドで作成 |
| 組織選択 | 文字フィールド | 文字フィールドで作成 |
| グループ選択 | 文字フィールド | 文字フィールドで作成 |
| テーブル | 表定義グループ | 最大3行の表定義明細として表グループを定義 |
| ルックアップ | 数値フィールド | |
| リンク | 文字フィールド | |
| レコード番号 | 文字フィールド | |
| 作成者 | 文字フィールド | 文字フィールドで作成 |
| 作成日時 | 文字フィールド | 文字フィールドで作成（書式無し） |
| 更新者 | 文字フィールド | 文字フィールドで作成 |
| 更新日時 | 文字フィールド | 文字フィールドで作成（書式無し） |

！注意事項

・TSF 変換時、kintone アプリの複数選択フィールドおよび、ドロップダウンフィールドの要素として扱う値に半角カンマ「,」が含まれている場合、カンマの前後で別々の選択肢として要素が分割されます。分割したくない場合は別の記号に置き換えてください。

■生成された TSF ファイルの表示イメージ



※ 生成されたフォームは必ず eFormMaker でレイアウトを修正した後に X-point へのアップロードを行って下さい。

！注意事項

- ・ kintone アプリを TSF ファイルに変換する際、フィールドに設定していた必須入力設定は引き継がれません。eFormMaker 上で再度必須設定する必要があります。
- ・ 以下の文字を kintone アプリのフィールド名、初期値に使用している状態で tsf 作成を行うと eFormMaker で開いた際に読み込みエラーが発生します。使用している場合は別の文字に置き換えてください。

「&」: アンパサンド (半角)

「<」: 大なり (半角)

「>」: 小なり (半角)

「”」: ダブルクォーテーション (半角)

9.3.2. X-point → kintone

X-point のフォームに記述する JavaScript から kintone の REST API を利用するための中継処理になります。

通常、kintone が動作するドメイン (cybozu.com) と X-point が動作するドメインは異なるドメインになるため、X-point のフォーム内に記述した JavaScript より XMLHttpRequest 等を利用して kintone の REST API を利用するとクロスドメイン制約により実行することができません。X-point の kintone 連携では X-point サーバーが kintone との通信を中継することにより、クロスドメイン制約を受けない動作を実現しています。

※ 唯一の例外として添付ファイルのアップロードのみ kintone と異なる仕様で動作します。

9.3.2.1. X-point からの kintone の REST API 実行ルール

X-point のフォームに記述する JavaScript から kintone の REST API を実行する場合は次のルールに従って記述します。

■kintone REST API 仕様に記述されている URI 記述例 (レコード取得)

【通常アプリ】

`https://(サブドメイン名).cybozu.com/k/v1/record.json`

【ゲストスペース内アプリ】

`https://(サブドメイン名).cybozu.com/k/guest/(スペースのID)/v1/record.json`

■X-point のフォーム内から呼び出す場合の URI 記述例 (レコード取得)

【通常アプリ】

`https://(X-point FQDN 名)/xpoint/krh/(kintone サブドメイン名).cybozu.com/k/v1/record.json`

【ゲストスペース内アプリ】

`https://(X-point FQDN 名)/xpoint/krh/(kintone サブドメイン名).cybozu.com/k/guest/(スペースのID)/v1/record.json`

URI の記述以外、クエリ文字列の指定方法は kintone と同一になります。HTTP リクエストヘッダの指定方法は以下のルールに従います。Cookie については転送を行いません。そのため、セッション認証で利用されているケースには対応しません。

【HTTP ヘッダ指定値を転送する】・・・指定されている場合のみ転送します

| HTTP リクエストヘッダ名 | 用途 | 備考 |
|------------------------|---------------|----|
| X-Cybozu-Authorization | パスワード認証 | |
| X-Cybozu-API-Token | API トークン認証 | |
| Authorization | Basic 認証 | |
| X-Requested-With | XHR 利用を示すヘッダ値 | |

【HTTP ヘッダに固定値を設定する】

| HTTP リクエストヘッダ名 | 用途 | 備考 |
|------------------------|-----------------|--|
| X-HTTP-Method-Override | PUT/DELETE の上書き | Method 値 (POST/PUT/GET/DELETE) の場合と同じ値を設定 PUT/DELETE を許可しない Firewall などへの対応 |

9.3.2.2. 中継処理で発生するエラー

中継処理内でエラーが発生した場合、エラー応答は kintone のエラー応答形式で返されます。

【応答の例】

```
{
  "message": "オプション・ライセンス「サイボウズ kintone 連携アダプタ」が指定されていません。",
  "id": "1505999166897850006",
  "code": "XP_101"
}
```

【X-point 独自エラー】・・・http response code は全て 510 になります。

| message | id | code |
|---|---------|--------|
| オプション・ライセンス「サイボウズ kintone 連携アダプタ」が指定されていません | 内部シリアル値 | XP_101 |

- ※ 内部シリアル値は固定値ではありません。X-point の内部インスタンスを元にした値が返されます。
- ※ ライセンスに係わるエラー以外は全て添付ファイルを X-point から kintone に転送する際に発生する可能性のあるエラーです。

9.3.2.3. ファイルアップロード API 仕様

URI の末尾が「/file.json」、かつ HTTP メソッドが「POST」である場合に限り、kintone のファイルアップロード REST API では無く、X-point 固有のファイルアップロード動作が行われます。

kintone のファイルアップロード動作はファイルそのものを multipart 形式で POST し、kintone に登録された filekey を応答として得る動作をしますが、X-point では指定書類の指定添付ファイルを kintone にアップロードし登録を行い filekey を返す動作になります。返される filekey 情報は kintone のファイルアップロード API と同じ形式の戻り値になります。

- ※ 同一 URL を指定し GET で実行した場合は kintone のファイルダウンロード API 動作になります。

■HTTP メソッド

POST

■URI

https://(X-point FQDN 名)/xpoint/krh/(サブドメイン名).cybozu.com/k/v1/file.json

■リクエスト形式

“application/x-www-form-urlencoded” 形式で送信します。

【リクエストの例】

docid=1798&seqno=0

■ リクエストパラメータ

| パラメータ名 | 指定する値 | 必須 | 説明 |
|--------|-------|----|--|
| docid | 数値 | 必須 | X-point 書類番号 |
| seqno | 数値 | 必須 | 添付ファイルの格納連番 書類ビューの添付ファイル表示で左側に表示される番号から 1を減算した値と一致します。 |

■ 応答の例

```
{ "fileKey": "93a999f4-7ae8-4c3e-9fed-573456da17ba" }
```

■ アップロードしたファイルとレコードの関連付け

cybozu developer network (<https://cybozudev.zendesk.com/hc/ja>) の「ファイルアップロード」を参照してください。

【X-point 独自エラー】・・・http response code は全て 510 になります。

| message | id | code |
|--|---------|--------|
| 指定された書類番号が不正です (*****) | 内部シリアル値 | XP_110 |
| 指定された添付ファイルの登録情報が見つかりません (書類 ID: *****, 添付連番:*****) | ↑ | XP_111 |
| 指定された添付ファイルが保存されていません (書類 ID: *****, 添付連番: *****, ファイル名: *****) | ↑ | XP_112 |

※ 内部シリアル値は固定値ではありません。X-point の内部インスタンスを元にした値が返されます。

第10章. その他

10.1. X-point に接続できない場合の動作

kintone が連携する X-point サーバーがメンテナンスなどで停止している場合、連携設定を行っている kintone アプリを起動するとアラート表示が行われます。



この表示は管理サイトの「kintone 連携・設定フォーム一覧」で取得する連携用 JavaScript テンプレートファイルが行っています。kintone のアプリの設定で「JavaScript / CSS でカスタマイズ」の項で指定する JavaScript ファイルは次の順で読み込み指定が行われます。

1. <https://{X-pointのFQDN}/xpoint/js/kintone.js> ← 連携用 JavaScript API ファイル
2. `kintone-{フォームコード}.js` ← JavaScript テンプレートファイルを修正したもの
3. kintone アプリにボタンなどを配置する JavaScript ← お客様が作成した JavaScript

X-point サーバーが停止、あるいはブラウザから接続できない場合は最初に読み込まれる “kintone.js” が取得できないため、2番目に読み込まれる kintone 内に登録している JavaScript テンプレートファイルで連携に必要な情報が得られていないかの判断を行いアラート表示が行われます。

10.2.

メンテナンスモード

X-point の管理サイトで動作モードを「メンテナンスモード」にした場合、kintone アプリからの連携操作はできなくなります。
例外として、承認状況をダイアログ形式で表示する機能のみ利用することが出来ます。

■改訂履歴

| 改版 | 改版内容 |
|--------------|---|
| 2021年04月01版 | 初版 |
| 2021年05月21版 | 「2.3.1.ワークフロー処理から kintone アプリにレコードを追加・更新」の「完了書類の PDF を添付」に前提条件を追記 |
| 2021年05月21版 | 「5.6.承認管理からの一括更新」の一括更新で連携を動作させるための条件を適切な内容に修正 |
| 2021年05月21版 | 「第4章.kintone アプリから X-point への連携」注意事項の誤記を修正 |
| 2021年05月21版 | 「7.3.1.3.フォーム(TSF)雛形ダウンロード (HTTP メソッド:GET)」のパラメータ名の誤りを修正 “appid”→“app_id” |
| 2021年05月21版 | 「5.2.2.添付ファイルの送信」に send_attach_files 関数の第2引数の注意事項を追記 |
| 2021年09月16版 | X-point に0バイトのファイルは添付できない事を注意事項として記載 |
| 2022年03月14版 | マニュアルタイトル等の各部の名称を変更 |
| 2022年07月20日版 | 「Internet Explorer」に関わる記載を削除 |
| 2023年03月23日版 | 「1.3.2. プロキシサーバーや VPN の影響」章タイトルを変更し、プロキシや VPN 環境下の注意点の説明を修正。 ※旧タイトル:「リバースプロキシ、VPN 接続など」 |
| 2023年11月06日版 | 「5.1.5. kintone レコードを追加・更新するタイミングを指定」の②「連続承認」が動作した場合 の説明を修正。 |
| 2023年12月13日版 | 「2.2.12 kintone から承認する(ワークフロー操作)」API メソッド名の誤記を修正 |
| 2024年03月18日版 | 「4.1.4.10.承認状況を表示する」に、kintone から送信されたユーザ情報(ログイン ID)が X-point に存在する必要がある仕様を明記 |
| 2024年03月27日版 | 「7.1.フィールドの対応」のルックアップの連携矢印を修正 |
| 2024年06月21日版 | 「4.1.4.1.X-point に書類を申請」に通常フォームで使用するメソッドを明記 |
| 2024年06月25日版 | 「4.1.4.7 承認待ち件数を取得」使用例の内容を修正 |
| 2024年07月23日版 | 「はじめに」に「User-Agent」ヘッダの指定について追記 |
| 2024年07月25日版 | 「5.4.3.マスタ参照コンポーネントの作成」に、kintone 連携マスタの場合は「表示形式」に”スクロール式”が指定できない制約を追記 |
| 2025年07月18日版 | 「第6章 X-point 登録フォームから kintone アプリを作成する」を追加、また以降の章番号の修正 |
| 2025年07月23日版 | 「2.2.9. kintone アプリのフォームから TSF ファイルを作成」作成方法の補足説明を追記 「2.3.1. ワークフロー処理から kintone アプリにレコードを追加・更新」フォームコードが重複した場合の動作を注意事項に追記 「第4章.kintone アプリから X-point への連携」記載のリンクを最新のものに修正 |
| 2025年10月22日版 | 「2.1.2.1. 連携設定による連携」認証の違い「2. する」に説明を追記 「第7章.kintone 登録アプリから X-point フォームを作成する」を追加、また以降の章番号の修正 |
| 2025年12月04日版 | 「9.3.1.3. フォーム(TSF)雛型ダウンロード(HTTP メソッド:GET)」変換されるフィールドの種類を追記 |
| 2026年01月08日版 | 「2.2.9. kintone アプリのフォームから TSF ファイルを作成」 【アプリ項目、フィールドの変換】表の誤りを修正 日時 日時型フィールド → 日時 文字フィールド 「9.3.1.3. フォーム(TSF)雛型ダウンロード(HTTP メソッド:GET)」 必須設定、使用可能文字に関する注意事項を追記 |
| 2026年04月09日版 | 「3.2.2. kintone 連携の登録」誤字の修正 「第7章.kintone 登録アプリから X-point フォームを作成する」機能強化に際して、説明の追加とキャプチャを修正 |
| 2026年05月26日版 | 「7.1.kintone フォーム作成機能について」説明の修正 「7.2.1 kintone フォーム作成準備」注意事項を追記 「7.2.2. kintone フォーム作成実行」注意事項を追記 |