

AgeAgePlugin 利用方法

nishikawa-r

第1版

目次

1. はじめに	1
1.1. このドキュメントについて	1
1.2. OS	1
1.3. セットアップ実行環境	1
2. 各種パッケージインストール	2
2.1. npm	2
2.2. create-plugin(プラグインのひな形作成に使用)	2
2.3. plugin-packer(プラグインのパッケージ化に使用)	2
2.4. customize-uploader(カスタマイズファイルのアップロードに使用)	2
3. アプリケーションダウンロード~実行 (AgeAgePlugin)	3
3.1. ダウンロード	3
3.2. 実行	3
4. 画面仕様書	4
4.1. メイン画面	4
4.2. プラグイン名の作成画面	6
4.3. manifest.json変更画面	7
4.4. カスタマイズ名の作成画面	9
4.5. customize-manifest.json変更画面	10
4.6. URL追加画面	12
4.7. URL変更画面	13
5. タブ「プラグイン用」操作説明	14
5.1. プラグイン環境名を追加する	14
5.2. プラグイン環境名を削除する	14
5.3. プラグイン環境名を選択する	14
5.4. プラグイン環境名に詳細情報を保存する	15
5.5. ディレクトリ/ppkファイルの選択	15
5.6. プラグインのアプリ名/アプリバージョン/説明を変更する	15
5.7. プラグインに使用するJSファイルのディレクトリを追加する	16
5.8. プラグインに使用するJSのURLを追加する	16
5.9. プラグインに使用するJSファイルのディレクトリを変更する	16
5.10. プラグインに使用するJSのURLを変更する	17
5.11. プラグインに使用するJSファイル/JSのディレクトリ/URLを削除する	17
5.12. プラグインに使用するCSSファイルのディレクトリを追加する	17
5.13. プラグインに使用するCSSのURLを追加する	17
5.14. プラグインに使用するCSSファイルのディレクトリを変更する	18
5.15. プラグインに使用するCSSのURLを変更する	18

5.16. プラグインに使用するCSSファイル/CSSのディレクトリ/URLを削除する	18
5.17. プラグインアップロードを実行する	19
5.18. プラグインアップロード実行中に処理を終了させる	19
5.19. プラグインアップロード実行ログ内容を保存する	19
6. タブ「カスタマイズファイル用」操作説明	20
6.1. カスタマイズの環境名を追加する	20
6.2. カスタマイズ環境名を削除する	20
6.3. カスタマイズ環境名を選択する	21
6.4. カスタマイズ環境名に詳細情報を保存する	21
6.5. ディレクトリの選択	21
6.6. カスタマイズとしてkintone環境に追加するJSファイルのディレクトリを追加する	21
6.7. カスタマイズとしてkintone環境に追加するJSのURLを追加する	22
6.8. カスタマイズとしてkintone環境に変更を加えるJSファイルのディレクトリを変更する	22
6.9. カスタマイズとしてkintone環境に変更を加えるJSのURLを変更する	22
6.10. カスタマイズとして使用するJSファイル/JSのディレクトリ/URLを削除する	23
6.11. カスタマイズとしてkintone環境に追加するCSSファイルのディレクトリを追加する	23
6.12. カスタマイズとしてkintone環境に追加するCSSのURLを追加する	23
6.13. カスタマイズとしてkintone環境に変更を加えるCSSファイルのディレクトリを変更する	24
6.14. カスタマイズとしてkintone環境に変更を加えるCSSのURLを変更する	24
6.15. カスタマイズとして使用するCSSファイル/CSSのディレクトリ/URLを削除する	24
6.16. カスタマイズファイルアップロードを実行する	25
6.17. カスタマイズファイルアップロード実行中に処理を終了させる	25
6.18. カスタマイズファイルアップロード実行ログ内容を保存する	25
7. 更新履歴	26

Chapter 1. はじめに

1.1. このドキュメントについて

このドキュメントでは AgeAgePlugin のセットアップ方法と操作説明について記述します。

1.2. OS

AgeAgePlugin では下記の OS を動作対象とします。

- Windows10

1.3. セットアップ実行環境

- npm コマンドが実行できる環境
- インターネットに接続されている環境

Chapter 2. 各種パッケージインストール

2.1. npm

- パッケージを管理するために用いるnpmをインストールします。
- [公式サイト](#)からインストーラーをダウンロードし、インストーラー起動後に指示に従いインストールを行ってください。

2.2. create-plugin(プラグインのひな形作成に使用)

- プラグインのひな形を作成するのに用いるパッケージをグローバルインストールします。
- コマンドプロンプトで下記のコマンドを実行します。

```
npm install -g @kintone/create-plugin
```

2.3. plugin-packer(プラグインのパッケージ化に使用)

- プラグインのパッケージ化を行うパッケージをグローバルインストールします。
- コマンドプロンプトで下記のコマンドを実行します。

```
npm install -g @kintone/plugin-packer
```

2.4. customize-uploader(カスタマイズファイルのアップロードに使用)

- カスタマイズファイルのアップロードを行うパッケージをグローバルインストールします。
- コマンドプロンプトで下記のコマンドを実行します。

```
npm install -g @kintone/customize-uploader
```



グローバルインストールとは？

PC上のどのディレクトリからもコマンドが実行できる場所（PATHが通った場所）にインストールされます。

Chapter 3. アプリケーションダウンロード～実行 (AgeAgePlugin)

3.1. ダウンロード

1. [リリースページ](#) にて最新版のzipファイルをダウンロードする。
2. zipファイルを展開する。

3.2. 実行

1. 展開したフォルダの下記のexeファイルを実行

AgeAgePlugin-v[x.x.x.]/bin/Release/AgeAgePlugin.exe

```
AgeAgePlugin-v[x.x.x.]
├── bin
│   └── Release
│       └── AgeAgePlugin.exe
```

下記のダイアログが出現した場合は **1. 各種パッケージインストール** を行ってください。

インストールしてください

×



- ・npmがインストールされていません
- ・create-pluginがインストールされていません
- ・kintone-plugin-uploaderがインストールされていません
- ・kintone-plugin-packerがインストールされていません
- ・kintone-customize-uploaderがインストールされていません

OK

Chapter 4. 画面仕様書

4.1. メイン画面

4.1.1. タブ「プラグイン用」

AgeAgePlugin

プラグイン用

カスタマイズファイル用

プラグイン名

1

2

3

4

詳細情報を非表示にする

詳細情報

URL

5

ユーザー名

6

パスワード

7

表示

ディレクトリ

9

10

マニフェスト変更

ver12

ppk

13

14

保存

ログ保存

実行

13

実行時にクリアにする

19

Text is not SVG - cannot display

表 1. タブ「プラグイン用」の項目

No.	画面オブジェクト	説明
1	プラグインアップロード環境名 選択コンボボックス	保存されているプラグインをアップロードする環境名の選択を行う。
2	プラグインアップロード環境名 追加ボタン	プラグインをアップロードする環境名の追加を行う。押下時、プラグイン名の作成画面に遷移する。
3	プラグインアップロード環境削 除ボタン	プラグインをアップロードする環境情報の削除を行う。
4	詳細情報非表示チェックボッ クス	チェック時に詳細情報を非表示にする
5	URL	プラグインのアップロードを行うkintone環境のURLを入力するテキストボックス
6	ユーザー名	プラグインのアップロードを行うkintone環境にて管理者権限を持つユーザーのユーザー名を入力する テキストボックス
7	パスワード	プラグインのアップロードを行うkintone環境にて管理者権限を持つユーザーのパスワードを入力する テキストボックス
8	パスワード伏字切り替えボタン	パスワードの伏字を表示/非表示に切り替えるボタン
9	プラグインファイルディレクト リ	アップロードを行うプラグインのディレクトリが入るテキストボックス

No.	画面オブジェクト	説明
10	プラグインファイルディレクトリ参照ボタン	アップロードを行うプラグインのディレクトリ参照ボタン
11	マニフェスト変更ボタン	プラグインのmanifest.jsonの値の変更を行うボタン。押下時、manifest.jsonの変更画面に遷移する。
12	プラグインのバージョン数	プラグインのmanifest.jsonに定義されているバージョン数が表示される。 ※自動バージョンアップ有効時は [現時点のバージョン] → [アップロード実行時のバージョン] となる。
13	ppkファイルディレクトリ	ppkファイルのディレクトリが入るテキストボックス
14	ppkファイル参照ボタン	ppkファイルの参照ボタン
15	プラグイン環境詳細情報保存ボタン	詳細情報を保存するボタン。
16	実行ログ保存ボタン	押下時、ログ内容を任意の場所に保存するボタン。
17	実行ボタン	詳細情報の値を元にplugin-packer/plugin-uploaderを並列に実行するボタン。
18	実行ログクリアチェックボックス	チェック時、実行するたびにログ内容をクリアにする
19	ログ内容	標準出力される文字列をログとして表示する。

4.1.2. タブ「カスタマイズファイル用」

AgeAgePlugin

プラグイン用

カスタマイズファイル用

カスタマイズ名

1

2

3

4 詳細情報を非表示にする

詳細情報

URL

5

ユーザー名

6

パスワード

7

表示

ディレクトリ

9

10

マニフェスト変更

アプリID

12

保存

ログ保存

実行

18 実行時にクリアにする

17

Text is not SVG - cannot display

表 2. タブ「カスタマイズファイル用」の項目

No.	画面オブジェクト	説明
1	カスタマイズファイルアップロード環境選択コンボボックス	保存されているカスタマイズファイルをアップロードする環境の選択を行う。

No.	画面オブジェクト	説明
2	カスタマイズファイルアップロード環境名追加ボタン	カスタマイズファイルをアップロードする環境名情報の追加を行う。押下時、カスタマイズ名の作成画面に遷移する。
3	カスタマイズファイルアップロード環境削除ボタン	カスタマイズファイルをアップロードする環境情報の削除を行う。
4	詳細情報非表示チェックボックス	チェック時に詳細情報を非表示にする
5	URL	カスタマイズファイルのアップロードを行うkintone環境のURLを入力するテキストボックス
6	ユーザー名	カスタマイズファイルのアップロードを行うkintone環境にて管理者権限を持つユーザーのユーザー名を入力するテキストボックス
7	パスワード	カスタマイズファイルのアップロードを行うkintone環境にて管理者権限を持つユーザーのパスワードを入力するテキストボックス
8	パスワード伏字切り替えボタン	パスワードの伏字を表示/非表示に切り替えるボタン
9	カスタマイズファイルディレクトリ	アップロードを行うカスタマイズファイルのディレクトリが入るテキストボックス
10	カスタマイズファイルディレクトリ参照ボタン	アップロードを行うカスタマイズファイルのディレクトリ参照ボタン
11	マニフェスト変更ボタン	カスタマイズファイルのcustomize-manifest.jsonの値の変更を行うボタン。押下時、customize-manifest.jsonの変更画面に遷移する。
12	アプリID	カスタマイズファイルをアップロードするアプリのIDが入るテキストボックス
13	カスタマイズファイル環境詳細情報保存ボタン	詳細情報を保存するボタン。
14	実行ログ保存ボタン	押下時、ログ内容を任意の場所に保存するボタン。
15	実行ボタン	詳細情報の値を元にcustomize-uploaderを実行するボタン。
16	実行ログクリアチェックボックス	チェック時、実行するたびにログ内容をクリアにする
17	ログ内容	標準出力される文字列をログとして表示する。

4.2. プラグイン名の作成画面

表 3. プラグイン名の作成画面の項目

No.	画面オブジェクト	説明
1	プラグインの環境名入力用のテキストボックス	プラグインの環境名選択項目に使用する名前を入力する。
2	create-plugin実行選択チェックボックス	チェック時、OKボタン押下後、create-pluginを実行しプラグインのひな型を任意のディレクトリに作成する。
3	キャンセルボタン	プラグイン環境名の作成をキャンセルする。
4	OKボタン	プラグインの環境名を環境名選択項目として追加する。

4.3. manifest.json変更画面

4.3.1. タブ「基本情報」

manifest.json変更

基本情報

JS

CSS

名前

subtablePlugin!!

説明

関連レコード一覧のように関連する別アプリのサブテーブル行を一覧表示するプラグインです。

バージョン

43

プラスするバージョン数

04

5

自動バージョンアップを有効にする

キャンセル

変更を保存

Text is not SVG - cannot display

表 4. タブ「基本情報」の項目

No.	画面オブジェクト	説明
1	プラグイン名	プラグインの名前を入力する。
2	プラグイン説明	プラグインの説明を入力する。
3	プラグインバージョン数	プラグインのバージョン数を入力する。
4	プラグインアップロード実行時 プラスするバージョン数コンボ ボックス	プラグインアップロード実行時にプラスするバージョン数を選択 or 入力する

No.	画面オブジェクト	説明
5	プラグイン自動バージョンアップチェックボックス	チェック時に実行した際、プラグインアップロード実行時プラスするバージョン数コンボボックスの値分をバージョン数にプラスした形で定義ファイルを上書きする。
6	キャンセルボタン	manifest.jsonの変更をキャンセルする。
7	変更保存ボタン	押下時、manifest.jsonに変更を加える。

4.3.2. タブ「JS」

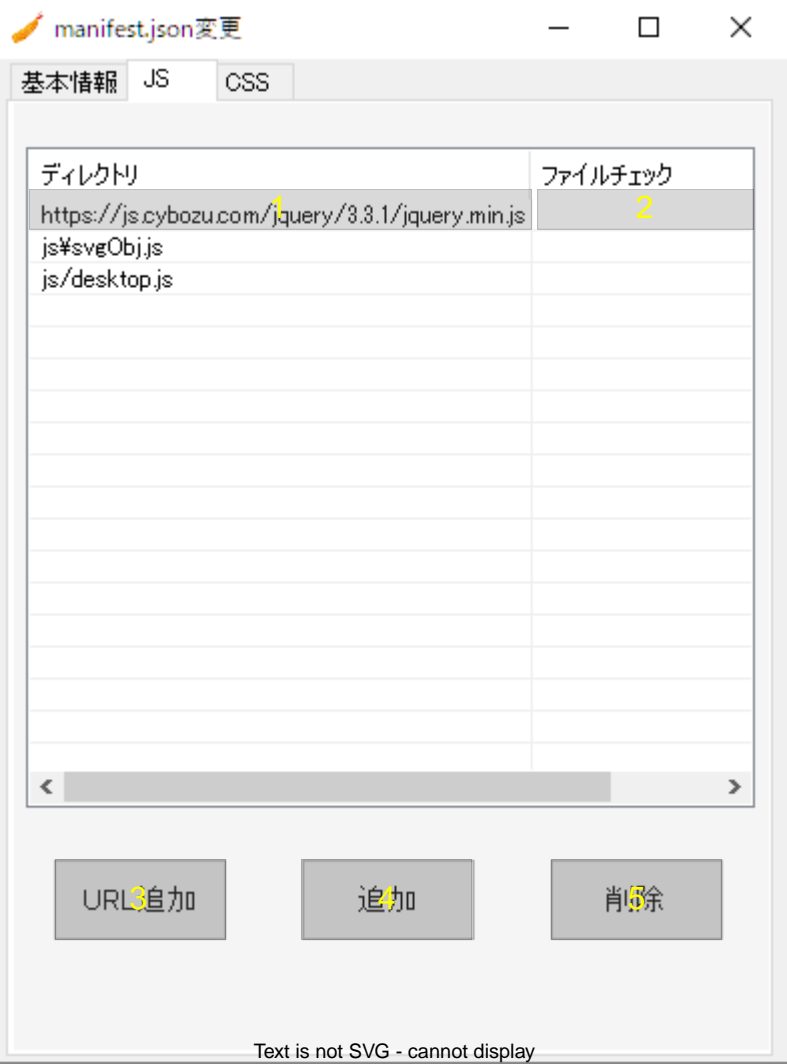


表 5. タブ「JS」の項目

No.	画面オブジェクト	説明
1	ディレクトリ	manifest.jsonに定義されているjsファイルのディレクトリ/URLが表示される。 ダブルクリック:JSファイル⇒ディレクトリ変更 URL⇒URL変更画面に遷移する。
2	ファイルチェック	ファイル/URLの存在確認の結果が表示される。 JSファイル⇒有:空,無:FILE_NOT_FOUND URL⇒有:空,無:URL_NOT_SUCCESSFULL
3	URL追加ボタン	URLを追加します。 押下時、URL追加画面に遷移する。
4	追加ボタン	JSファイルのディレクトリを追加します。 押下時、ディレクトリ選択を行い選択したディレクトリの値がディレクトリに追加されます。
5	削除ボタン	選択したディレクトリをリストから削除します。

4.3.3. タブ「CSS」

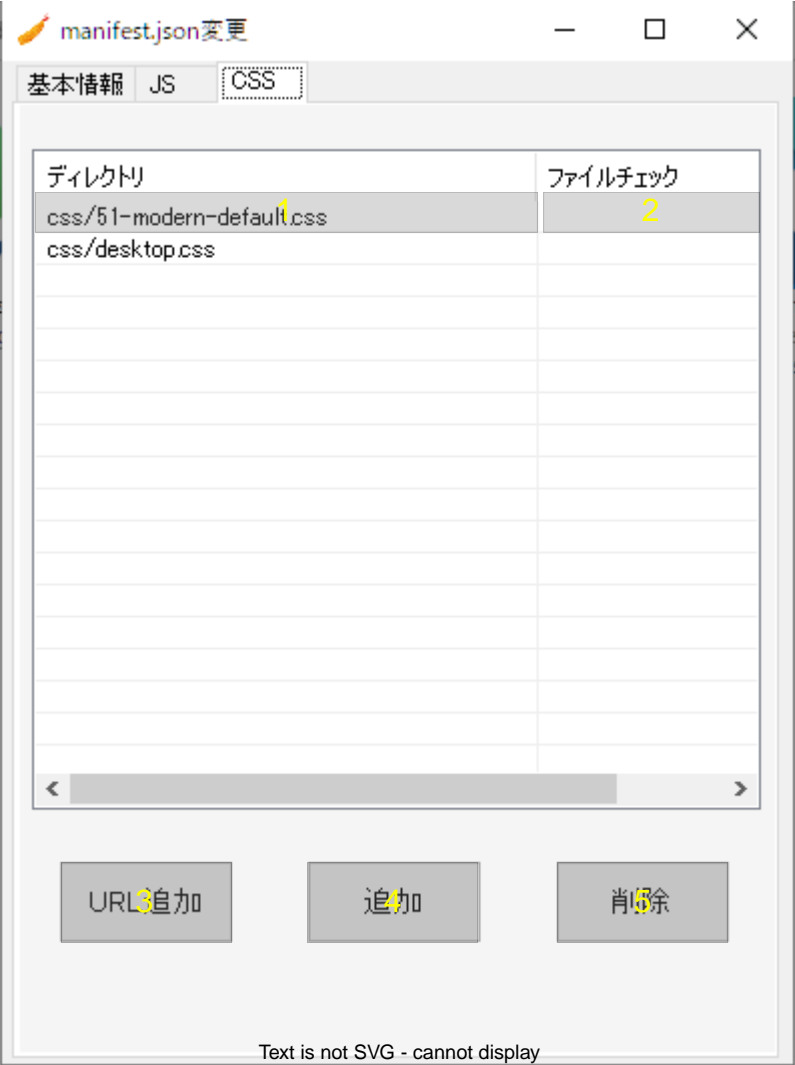


表 6. タブ「CSS」の項目

No.	画面オブジェクト	説明
1	ディレクトリ	manifest.jsonに定義されているcssファイルのディレクトリ/URLが表示される。 ダブルクリック:cssファイル⇒ディレクトリ変更 URL⇒URL変更画面に遷移する。
2	ファイルチェック	ファイル/URLの存在確認の結果が表示される。 cssファイル⇒有:空,無:FILE_NOT_FOUND URL⇒有:空,無:URL_NOT_SUCCESSFULL
3	URL追加ボタン	URLを追加します。押下時、URL追加画面に遷移する。
4	追加ボタン	CSSファイルのディレクトリを追加します。押下時、ディレクトリ選択を行い選択したディレクトリの値がディレクトリに追加されます。
5	削除ボタン	選択したディレクトリをリストから削除します。

4.4. カスタマイズ名の作成画面

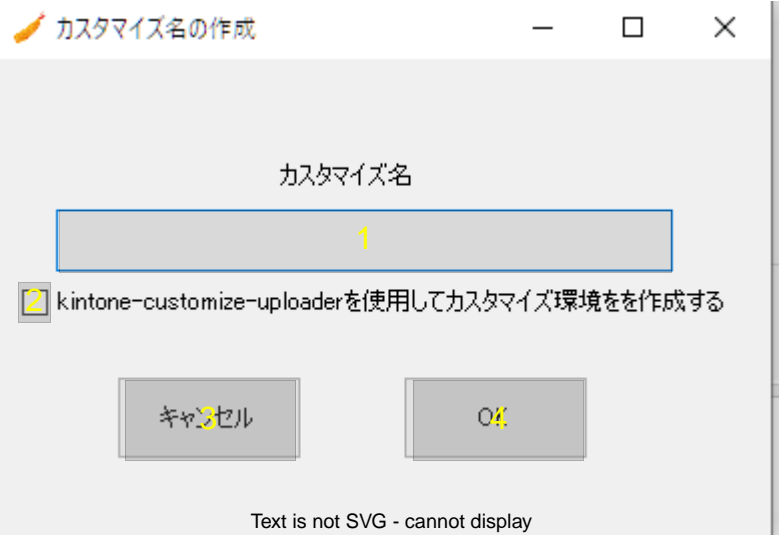


表 7. カスタマイズ名の作成画面の項目

No.	画面オブジェクト	説明
1	カスタマイズの環境名入力用のテキストボックス	カスタマイズの環境名選択項目に使用する名前を入力する。
2	kintone-customize-uploader実行選択チェックボックス	チェック時、OKボタン押下後、cuatomize-uploaderを実行しカスタマイズファイルアップロードに必要なひな型を任意のディレクトリに作成する。
3	キャンセルボタン	カスタマイズ環境名の作成をキャンセルする。
4	OKボタン	カスタマイズの環境名を環境名選択項目として追加する。

4.5. customize-manifest.json変更画面

4.5.1. タブ「JS」

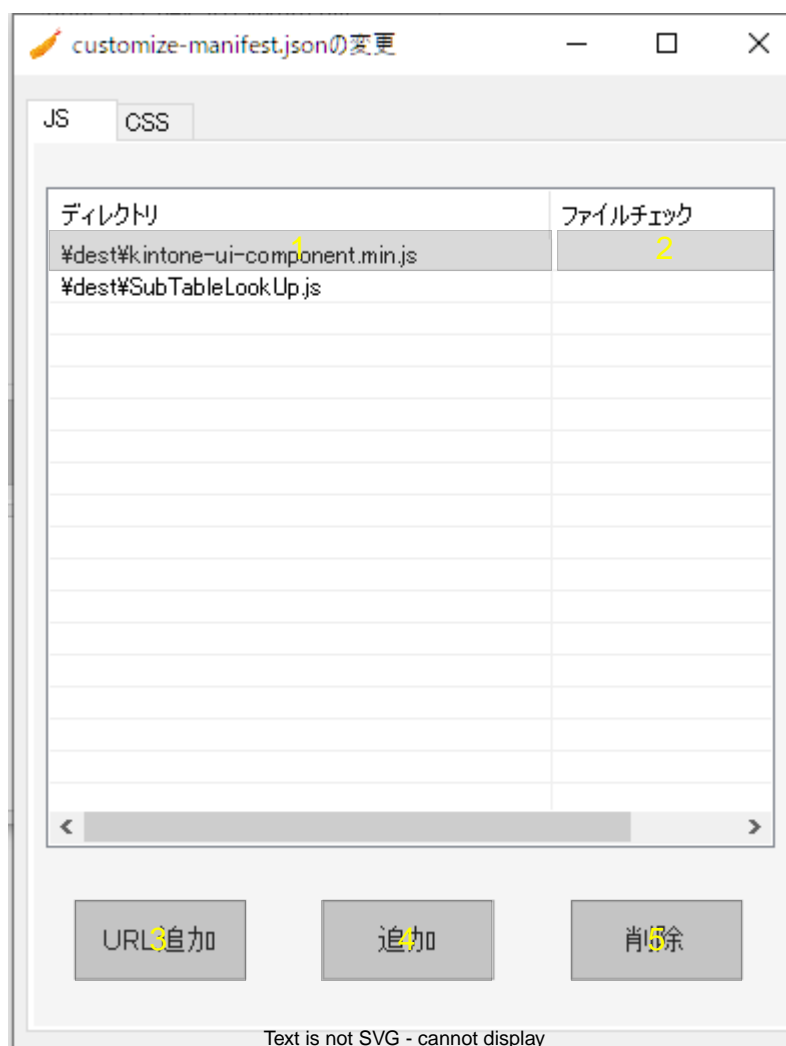


表 8. タブ「JS」の項目

No.	画面オブジェクト	説明
1	ディレクトリ	custommize-manifest.jsonに定義されているjsファイルのディレクトリ/URLが表示される。 ダブルクリック:JSファイル⇒ディレクトリ変更 URL⇒URL変更画面に遷移する。
2	ファイルチェック	ファイル/URLの存在確認の結果が表示される。 JSファイル⇒有:空,無:FILE_NOT_FOUND URL⇒有:空,無:URL_NOT_SUCCESSFULL
3	URL追加ボタン	URLを追加します。 押下時、URL追加画面に遷移する。
4	追加ボタン	JSファイルのディレクトリを追加します。 押下時、ディレクトリ選択を行い選択したディレクトリの値がディレクトリに追加されます。
5	削除ボタン	選択したディレクトリをリストから削除します。

4.5.2. タブ「CSS」

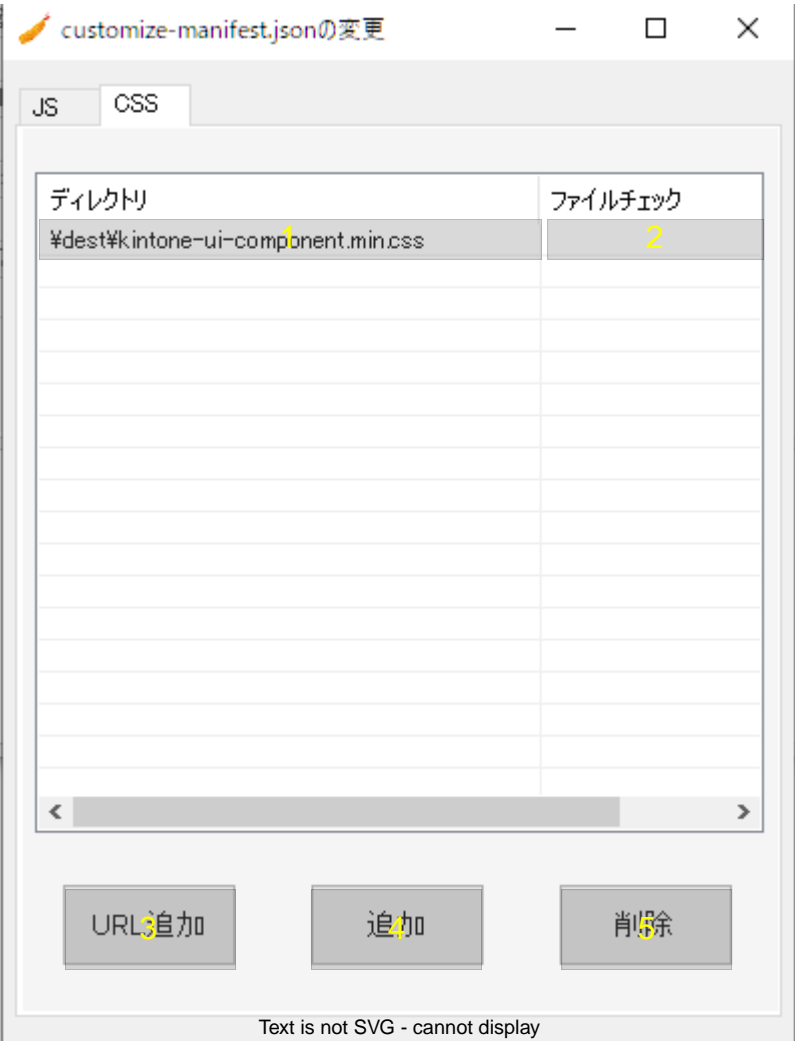


表 9. タブ「CSS」の項目

No.	画面オブジェクト	説明
1	ディレクトリ	customize-manifest.jsonに定義されているcssファイルのディレクトリ/URLが表示される。 ダブルクリック:cssファイル⇒ディレクトリ変更 URL⇒URL変更画面に遷移する。
2	ファイルチェック	ファイル/URLの存在確認の結果が表示される。 cssファイル⇒有:空,無:FILE_NOT_FOUND URL⇒有:空,無:URL_NOT_SUCCESSFULL
3	URL追加ボタン	URLを追加します。 押下時、URL追加画面に遷移する。
4	追加ボタン	CSSファイルのディレクトリを追加します。 押下時、ディレクトリ選択を行い選択したディレクトリの値がディレクトリに追加されます。
5	削除ボタン	選択したディレクトリをリストから削除します。

4.6. URL追加画面



表 10. URL追加画面の項目

No.	画面オブジェクト	説明
1	追加URL	ディレクトリのリストに追加したいURLを入力します。
2	キャンセルボタン	URLの追加をキャンセルする。
3	OKボタン	入力したURLをディレクトリのリストに追加します。

4.7. URL変更画面

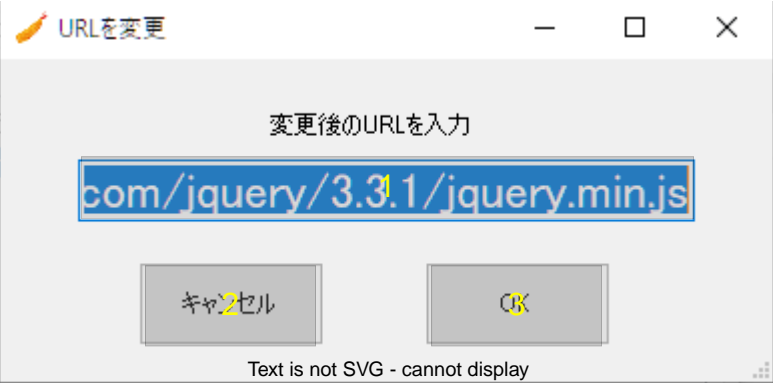


表 11. URL追加画面の項目

No.	画面オブジェクト	説明
1	変更後URL	ディレクトリのリストにて選択したURLを変更します。
2	キャンセルボタン	URLの変更をキャンセルする。
3	OKボタン	ディレクトリのリストにて選択したURLを変更します。

Chapter 5. タブ「プラグイン用」操作説明

AgeAgePlugin

プラグイン用

カスタマイズファイル用

カスタマイズ名

+

-

☐ 詳細情報を非表示にする

詳細情報

URL

ユーザー名

パスワード

表示

ディレクトリ

...

マニフェスト変更

アプリID

保存

ログ保存

実行

☐ 実行時にクリアにする

5.1. プラグイン環境名を追加する

表 12. 操作方法

No.	操作	補足
1	プラグインアップロード環境名追加ボタンをクリック	クリック後、プラグイン名の作成画面に遷移する。
2	プラグイン名の作成画面にてプラグイン名を入力	プラグインのひな型から作成する場合は create-plugin実行選択チェックボックス にチェックを入れること。
3	「追加が完了しました」とダイアログが出現するのでOKボタンを押下する	

5.2. プラグイン環境名を削除する

表 13. 操作方法

No.	操作	補足
1	プラグインアップロード環境削除ボタンをクリック	
2	「削除してもよろしいですか」とダイアログが出現するのでOKボタンを押下する	※この際、保存した詳細情報も削除されます。

5.3. プラグイン環境名を選択する

表 14. 操作方法

No.	操作	補足
1	プラグインアップロード環境名 選択コンボボックスをクリック	
2	任意の選択項目を選択する	選択後、詳細情報の値が環境名選択時に保存した詳細情報の値に変更される。

5.4. プラグイン環境名に詳細情報を保存する

表 15. 操作方法

No.	操作	補足
1	詳細情報のURL/ユーザー名/パスワード/ディレクトリ/ppkファイルを入力 or 選択されていることを確認。	
2	プラグイン環境詳細情報保存ボタンをクリックする	

5.5. ディレクトリ/ppkファイルの選択

表 16. 操作方法

No.	操作	補足
1	プラグインファイルディレクトリ参照ボタンをクリックする	
2	任意のディレクトリを選択する	
3	ppkファイル参照ボタンをクリックする	
4	任意のppkファイルを選択する	

ディレクトリは下記の通りに指定すること

指定するディレクトリ

- src
- manifest.json



ppkファイルは下記の通りに指定すること

指定したディレクトリ

- 指定するppkファイル



ppkファイルとは
秘密鍵ファイルのことであり、2回目以降のプラグインのパッケージングの際に用いる。

5.6. プラグインのアプリ名/アプリバージョン/説明を変更する

表 17. 操作方法

No.	操作	補足
1	マニフェスト変更ボタンをクリック	その後、manifest.json変更画面に遷移する。
2	manifest.json変更画面のタブ「基本情報」にてアプリ名/アプリバージョン/説明を変更する。	

No.	操作	補足
3	manifest.json変更画面にて変更保存ボタンをクリック	

5.7. プラグインに使用するJSファイルのディレクトリを追加する

表 18. 操作方法

No.	操作	補足
1	マニフェスト変更ボタンをクリック	その後、manifest.json変更画面に遷移する。
2	manifest.json変更画面にてタブ「JS」を選択する	
3	追加ボタンをクリックし任意のファイルを選択する	※複数選択可

JSファイルのディレクトリは下記の通りに指定すること



メイン画面 タブ「プラグイン用」ディレクトリ

└─src

└─指定するJSファイル

5.8. プラグインに使用するJSのURLを追加する

表 19. 操作方法

No.	操作	補足
1	マニフェスト変更ボタンをクリック	その後、manifest.json変更画面に遷移する。
2	manifest.json変更画面にてタブ「JS」を選択する	
3	URL追加ボタンをクリックする	その後、URL追加画面に遷移する
4	URL追加画面の追加URLに任意のURLを入力する	
5	URL追加画面のOKボタンをクリックする	

5.9. プラグインに使用するJSファイルのディレクトリを変更する

表 20. 操作方法

No.	操作	補足
1	マニフェスト変更ボタンをクリック	その後、manifest.json変更画面に遷移する。
2	manifest.json変更画面にてタブ「JS」を選択する	
3	リストのJSのディレクトリの値をダブルクリックする	
4	JSファイルのディレクトリ選択を行う	



JSファイルのディレクトリは下記の通りに指定すること

メイン画面 タブ「プラグイン用」ディレクトリ

└─src
└─指定するJSファイル

5.10. プラグインに使用するJSのURLを変更する

表 21. 操作方法

No.	操作	補足
1	マニフェスト変更ボタンをクリック	その後、manifest.json変更画面に遷移する。
2	manifest.json変更画面にてタブ「JS」を選択する	
3	リストのJSのURLの値をダブルクリックする	その後、URL変更画面に遷移する
4	URL変更画面の変更後URLに任意のURLを入力する	
5	URL変更画面のOKボタンをクリックする	

5.11. プラグインに使用するJSファイル/JSのディレクトリ/URLを削除する

表 22. 操作方法

No.	操作	補足
1	マニフェスト変更ボタンをクリック	その後、manifest.json変更画面に遷移する。
2	manifest.json変更画面にてタブ「JS」を選択する	
3	リストのJS/JSファイルのディレクトリ/URLを選択する	※複数選択可
4	削除ボタンをクリックする	

5.12. プラグインに使用するCSSファイルのディレクトリを追加する

表 23. 操作方法

No.	操作	補足
1	マニフェスト変更ボタンをクリック	その後、manifest.json変更画面に遷移する。
2	manifest.json変更画面にてタブ「CSS」を選択する	
3	追加ボタンをクリックし任意のファイルを選択する	※複数選択可

CSSファイルのディレクトリは下記の通りに指定すること



メイン画面 タブ「プラグイン用」ディレクトリ
└─src
└─指定するCSSファイル

5.13. プラグインに使用するCSSのURLを追加する

表 24. 操作方法

No.	操作	補足
1	マニフェスト変更ボタンをクリック	その後、manifest.json変更画面に遷移する。
2	manifest.json変更画面にてタブ「CSS」を選択する	
3	URL追加ボタンをクリックする	その後、URL追加画面に遷移する
4	URL追加画面の追加URLに任意のURLを入力する	
5	URL追加画面のOKボタンをクリックする	

5.14. プラグインに使用するCSSファイルのディレクトリを変更する

表 25. 操作方法

No.	操作	補足
1	マニフェスト変更ボタンをクリック	その後、manifest.json変更画面に遷移する。
2	manifest.json変更画面にてタブ「CSS」を選択する	
3	リストのCSSのディレクトリの値をダブルクリックする	
4	JSファイルのディレクトリ選択を行う	

CSSファイルのディレクトリは下記の通りに指定すること



メイン画面 タブ「プラグイン用」ディレクトリ

└─src

└─指定するCSSファイル

5.15. プラグインに使用するCSSのURLを変更する

表 26. 操作方法

No.	操作	補足
1	マニフェスト変更ボタンをクリック	その後、manifest.json変更画面に遷移する。
2	manifest.json変更画面にてタブ「CSS」を選択する	
3	リストのCSSのURLの値をダブルクリックする	その後、URL変更画面に遷移する
4	URL変更画面の変更後URLに任意のURLを入力する	
5	URL変更画面のOKボタンをクリックする	

5.16. プラグインに使用するCSSファイル/CSSのディレクトリ/URLを削除する

表 27. 操作方法

No.	操作	補足
1	マニフェスト変更ボタンをクリック	その後、manifest.json変更画面に遷移する。
2	manifest.json変更画面にてタブ「CSS」を選択する	
3	リストのCSS/CSSファイルのディレクトリ/URLを選択する	※複数選択可
4	削除ボタンをクリックする	

5.17. プラグインアップロードを実行する

表 28. 操作方法

No.	操作	補足
1	詳細情報が正しく入力されていることを確認	
2	実行ボタンをクリックする	クリック後、「実行を終了」にテキストが変更する。 エラーがあった場合、即時に中断され、エラーダイアログが結果として帰ってきます。 標準出力はログ内容に出力されます。 ※監視モードでjs/cssファイルを監視しているので変更を加えれば自動的にパッケージング化されkintone環境にアップロードされます。



二回目以降やプラグインに変更を加える場合はppkファイルを指定してください。(指定しないとkintone環境にて新規にプラグインが追加されます)

5.18. プラグインアップロード実行中に処理を終了させる

表 29. 操作方法

No.	操作	補足
1	実行を終了ボタンをクリックする	押下後、実行が終了される

5.19. プラグインアップロード実行ログ内容を保存する

表 30. 操作方法

No.	操作	補足
1	実行ログ保存ボタンをクリックする	
2	ディレクトリ選択後、ログ内容が任意の名前でテキストファイルとして保存される。	名前はデフォルトでは <code>Log_[YYmmdd].txt</code> となっています。

Chapter 6. タブ「カスタマイズファイル用」操作説明

AgeAgePlugin

プラグイン用

カスタマイズファイル用

カスタマイズ名

+

-

☐ 詳細情報を非表示にする

詳細情報

URL

ユーザー名

パスワード

表示

ディレクトリ

...

マニフェスト変更

アプリID

保存

ログ保存

実行

☐ 実行時にクリアにする



アップロードを行うkintone環境の指定のアプリにすでにカスタマイズファイルがある場合、全てのファイルはcustomize-manifest.jsonで定義したjs/cssに書き換えられてしまうので、必ずカスタマイズがされていないアプリを使用するか、アプリのカスタマイズファイルが消えてもいい環境でのみ実行する様にしてください。

6.1. カスタマイズの環境名を追加する

表 31. 操作方法

No.	操作	補足
1	カスタマイズファイルアップロード環境名追加ボタンをクリック	クリック後、カスタマイズ名の作成画面に遷移する。
2	カスタマイズ名の作成画面にてカスタマイズ名を入力	カスタマイズアップロードのひな型から作成する場合は kintone-customize-uploader実行選択チェックボックス にチェックを入れること。
3	「追加が完了しました」とダイアログが出現するのでOKボタンを押下する	

6.2. カスタマイズ環境名を削除する

表 32. 操作方法

No.	操作	補足
1	カスタマイズファイルアップロード環境削除ボタンをクリック	

No.	操作	補足
2	「削除してもよろしいですか」とダイアログが出現するのでOKボタンを押下する	※この際、保存した詳細情報も削除されます。

6.3. カスタマイズ環境名を選択する

表 33. 操作方法

No.	操作	補足
1	カスタマイズファイルアップロード環境選択コンボボックスをクリック	
2	任意の選択項目を選択する	選択後、詳細情報の値が環境名選択時に保存した詳細情報の値に変更される。

6.4. カスタマイズ環境名に詳細情報を保存する

表 34. 操作方法

No.	操作	補足
1	詳細情報のURL/ユーザー名/パスワード/ディレクトリ/アプリIDを入力 or 選択されていることを確認。	
2	カスタマイズファイル環境詳細情報保存ボタンをクリックする	

6.5. ディレクトリの選択

表 35. 操作方法

No.	操作	補足
1	カスタマイズファイルディレクトリ参照ボタンをクリックする	
2	任意のディレクトリを選択する	

ディレクトリは下記の通りに指定すること



指定するディレクトリ
└─dest
 └─customize-manifest.json

6.6. カスタマイズとしてkintone環境に追加するJSファイルのディレクトリを追加する

表 36. 操作方法

No.	操作	補足
1	マニフェスト変更ボタンをクリック	その後、customize-manifest.json変更画面に遷移する。
2	customize-manifest.json変更画面にてタブ「JS」を選択する	
3	追加ボタンをクリックし任意のファイルを選択する	※複数選択可

JSファイルのディレクトリは下記の通りに指定すること



メイン画面 タブ「カスタマイズファイル用」ディレクトリ
 └─dest
 └─指定するJSファイル

6.7. カスタマイズとしてkintone環境に追加するJSのURLを追加する

表 37. 操作方法

No.	操作	補足
1	マニフェスト変更ボタンをクリック	その後、customize-manifest.json変更画面に遷移する。
2	customize-manifest.json変更画面にてタブ「JS」を選択する	
3	URL追加ボタンをクリックする	その後、URL追加画面に遷移する
4	URL追加画面の追加URLに任意のURLを入力する	
5	URL追加画面のOKボタンをクリックする	

6.8. カスタマイズとしてkintone環境に変更を加えるJSファイルのディレクトリを変更する

表 38. 操作方法

No.	操作	補足
1	マニフェスト変更ボタンをクリック	その後、customize-manifest.json変更画面に遷移する。
2	customize-manifest.json変更画面にてタブ「JS」を選択する	
3	リストのJSのディレクトリの値をダブルクリックする	
4	JSファイルのディレクトリ選択を行う	

JSファイルのディレクトリは下記の通りに指定すること



メイン画面 タブ「カスタマイズファイル用」ディレクトリ
 └─dest
 └─指定するJSファイル

6.9. カスタマイズとしてkintone環境に変更を加えるJSのURLを変更する

表 39. 操作方法

No.	操作	補足
1	マニフェスト変更ボタンをクリック	その後、customize-manifest.json変更画面に遷移する。
2	customize-manifest.json変更画面にてタブ「JS」を選択する	

No.	操作	補足
3	リストのJSのURLの値をダブルクリックする	その後、URL変更画面に遷移する
4	URL変更画面の変更後URLに任意のURLを入力する	
5	URL変更画面のOKボタンをクリックする	

6.10. カスタマイズとして使用するJSファイル/JSのディレクトリ/URLを削除する

表 40. 操作方法

No.	操作	補足
1	マニフェスト変更ボタンをクリック	その後、customize-manifest.json変更画面に遷移する。
2	customize-manifest.json変更画面にてタブ「JS」を選択する	
3	リストのJS/JSファイルのディレクトリ/URLを選択する	※複数選択可
4	削除ボタンをクリックする	

6.11. カスタマイズとしてkintone環境に追加するCSSファイルのディレクトリを追加する

表 41. 操作方法

No.	操作	補足
1	マニフェスト変更ボタンをクリック	その後、customize-manifest.json変更画面に遷移する。
2	customize-manifest.json変更画面にてタブ「CSS」を選択する	
3	追加ボタンをクリックし任意のファイルを選択する	※複数選択可

CSSファイルのディレクトリは下記の通りに指定すること



メイン画面 タブ「カスタマイズファイル用」ディレクトリ
 └─dest
 └─指定するCSSファイル

6.12. カスタマイズとしてkintone環境に追加するCSSのURLを追加する

表 42. 操作方法

No.	操作	補足
1	マニフェスト変更ボタンをクリック	その後、customize-manifest.json変更画面に遷移する。
2	customize-manifest.json変更画面にてタブ「CSS」を選択する	

No.	操作	補足
3	URL追加ボタンをクリックする	その後、URL追加画面に遷移する
4	URL追加画面の追加URLに任意のURLを入力する	
5	URL追加画面のOKボタンをクリックする	

6.13. カスタマイズとしてkintone環境に変更を加えるCSSファイルのディレクトリを変更する

表 43. 操作方法

No.	操作	補足
1	マニフェスト変更ボタンをクリック	その後、customize-manifest.json変更画面に遷移する。
2	customize-manifest.json変更画面にてタブ「CSS」を選択する	
3	リストのCSSのディレクトリの値をダブルクリックする	
4	CSSファイルのディレクトリ選択を行う	

CSSファイルのディレクトリは下記の通りに指定すること



メイン画面 タブ「カスタマイズファイル用」ディレクトリ

- └─dest
 - └─指定するCSSファイル

6.14. カスタマイズとしてkintone環境に変更を加えるCSSのURLを変更する

表 44. 操作方法

No.	操作	補足
1	マニフェスト変更ボタンをクリック	その後、customize-manifest.json変更画面に遷移する。
2	customize-manifest.json変更画面にてタブ「CSS」を選択する	
3	リストのCSSのURLの値をダブルクリックする	その後、URL変更画面に遷移する
4	URL変更画面の変更後URLに任意のURLを入力する	
5	URL変更画面のOKボタンをクリックする	

6.15. カスタマイズとして使用するCSSファイル/CSSのディレクトリ/URLを削除する

表 45. 操作方法

No.	操作	補足
1	マニフェスト変更ボタンをクリック	その後、customize-manifest.json変更画面に遷移する。
2	customize-manifest.json変更画面にてタブ「CSS」を選択する	
3	リストのCSS/CSSファイルのディレクトリ/URLを選択する	※複数選択可
4	削除ボタンをクリックする	

6.16. カスタマイズファイルアップロードを実行する

表 46. 操作方法

No.	操作	補足
1	詳細情報が正しく入力されていることを確認	
2	実行ボタンをクリックする	クリック後、「実行を終了」にテキストが変更する。 エラーがあった場合、即時に中断され、エラーダイアログが結果として帰ってきます。 標準出力はログ内容に出力されます。 ※監視モードでjs/cssファイルを監視しているので変更を加えれば自動的にメイン画面で指定したアプリIDを持つアプリのkintone環境にアップロードされます。

6.17. カスタマイズファイルアップロード実行中に処理を終了させる

表 47. 操作方法

No.	操作	補足
1	実行を終了ボタンをクリックする	押下後、実行が終了される

6.18. カスタマイズファイルアップロード実行ログ内容を保存する

表 48. 操作方法

No.	操作	補足
1	実行ログ保存ボタンをクリックする	
2	ディレクトリ選択後、ログ内容が任意の名前でテキストファイルとして保存される。	名前はデフォルトでは <code>Log_[YYmmdd].txt</code> となっています。

Chapter 7. 更新履歴

版数	作成者	更新日	更新内容
第1版	nishikawa-r	2022/11/12	新規作成