

可能な限り「モダンUI」での対応は、従来の「クラシックUI」と比較し、SharePoint標準で提供される「Webパーツ」が利用できないため、想定されるユーザー要望は殆どはSharePointだけでは実現できません。

① モダンUIのリストやライブラリへは「Webパーツ」を追加不可

- ・ビューの表示編集は、基本「JSON」設定のみ
→ビューへのアクセス権限設定ができない。 TreeView表示機能の追加などではない。
- ・フォームの表示編集や機能追加は「PowerApps」でのみ
→ヘッダー、本文、フッターの表示編集は、基本「JSON」設定のみだが、列ごとのClass指定などによる表示制御は、期待する動作設定が難しい。
→フォームへの表示制御・機能追加は、「PowerApps」での編集作業が必要。
クラシックUIは、「NewForm」「DispForm」「EditForm」ごとに編集が可能だが、モダンUI用の「PowerApps」画面は一つの画面設定で対応は必要であり、専用コードでの記述が必要なため、専任者での対応を推奨。

② モダンUIのサイトのページに限り「Webパーツ」の追加は許可

SharePoint OnlineのJavaScript Object Model(JSOM)は利用できないため、サイトに限り、大別して2つの対策案が考えられます。

【参考URL(内田洋行 太田さんのブログ):<https://idea.tostring.jp/?p=3986>】

◆ SharePoint Framework の利用【本命】

敷居が高いと思われる「SharePoint Framework」(公開されているソースコード有り)の「スクリプト エディタWebパーツ」をパッケージ化して活用

※参考資料：SharePointを学ぼう_17.【モダンUI】サイトの編集「スクリプトエディタを利用」

◆ サイトページに標準機能の組み合わせで行う方法

クラシックUIの標準機能をサイトに「埋め込み Web パーツで表示」する内容であり、サイトに固着しなくても良い場合は、SharePoint標準の「WikiPage」の活用をお勧めします。

① JavaScript を埋め込んだページを作成する。 - 「WikiPage」(.aspx)でも代用可能？

② SharePoint Online のデータを利用する場合には REST API を利用する

>SharePoint Online のドキュメントライブラリにアップロードした「.aspx」ファイルからJSOMを利用 ⇒ <https://idea.tostring.jp/?p=5462>

>ページ上への表示に iframe が利用されるため、CSS も自分で作成しておく必要がある。

>例えば、Office UI Fabric JS ⇒ <https://developer.microsoft.com/en-us/fabric-js>

この Office UI Fabric を基に自分で「CSS」のカスタマイズを行い、ページを拡張子「.aspx」として保存して任意のドキュメントライブラリにアップロード。

③ 埋め込み Web パーツで表示

これを SharePoint Online モダン ページの埋め込み Web パーツで呼出しは、SharePoint Online のドキュメントライブラリに保存されているページを直接参照できる URL を指定する。

これで、ページ上に先ほど作成したページが埋め込み Web パーツによって iframe のコンテンツとして表示されます。

表示される高さなどを指定したい場合には、埋め込み Web パーツの設定に iframe タグを記述することもできます。