

デザイナーとエンジニアの 役割とワークフロー

2023.10 MxEサービスドキュメント

目次

- [文書の目的](#)
- [工程と主な作業項目](#)
 - [工程概要](#)
 - [工程 1 : GUIの構造設計](#)
 - [工程 2 : デザインの作成](#)
 - [工程 3 : スクリプトの実装](#)
- [典型的なワークフロー](#)

文書の目的 (1/2)

- 例えば、新しいダッシュボードのデザイン画が一枚あるとします。この静止画をベースに動きを持ったGUIアプリケーションを完成させるには、デザイナー/エンジニアはそれぞれどのような視点で何をしたら良いでしょうか？
- MxEは、デザイナーとエンジニアがいかに適切に協働できるかがGUIコンテンツ制作の重要なポイントと考え、PSDインポーターとGUI Object連携を活用したGUI制作を推奨しています。
- MatrixEngineを使ったGUI開発の流れは、大枠では次図のようになります。
(「MatrixEngine概要」より抜粋)

MatrixEngineを使ったGUI開発の流れ

Step1

■ 起案・企画

アプリの機能 & 画面仕様、
デザインの検討

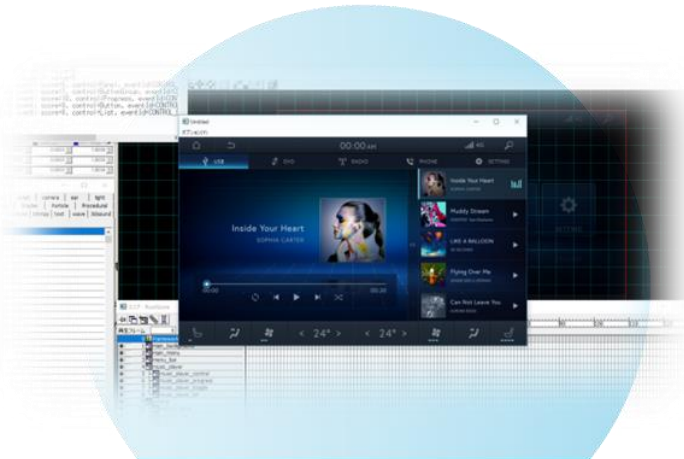


Photoshop等のツール

- ・ 機能 & 遷移の検討
- ・ 各画面設計
- ・ GUI部品の抽出
- ・ デザインデータ作成

Step2

■ PC上でGUIコンテンツを制作



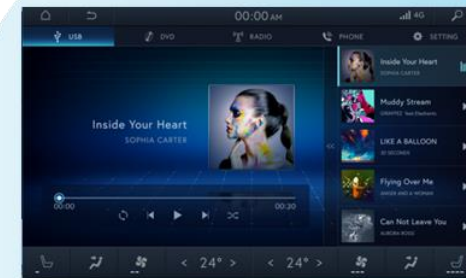
MatrixEngine SDK

1. Photoshopデザインデータをインポート
2. GUIオブジェクトに動作を設定・実装
3. ビルド・GUIコンテンツファイルを出力

(※受託開発も承ります)

Step3

■ ターゲット環境へ配置



MatrixEngine Player

1. GUIコンテンツファイルと購入したMxE Playerを環境へコピー

⇒ 完了！

文書の目的 (2/2)

- 本書では、ベースとなるデザインアイデアが既に存在する場合、MxEでGUIアプリケーションを制作するにはデザイナーとエンジニアがどのように協働すると良いか、Step1～Step3の具体的な作業内容とフローを説明します。

工程と主な作業項目

デザイナーとエンジニアの協働工程

工程概要

- MxEが推奨する「PSDインポーターとGUI Object連携を活用したGUI制作」では、デザイナーとエンジニアの協働が必要となる場面は大きく次の3工程に分けられます。

1. GUIの構造設計

MxEでのGUI制作上必要なルールをデザイナー/エンジニア間で共有する工程です。

デザイナー/エンジニアが一緒に検討

2. デザインの作成

工程1で取り決めたルールに沿って画面部品のデザイン（PSDファイル）を制作します。

デザイナーがリード
エンジニアがサポート

3. スクリプトの実装

PSDファイルを開発環境に取り込み、動きを実装します。

エンジニアがリード
デザイナーがサポート

* 工程1～2が「[MatrixEngineを使ったGUI開発の流れ](#)」のStep1の内容、工程3がStep2～3の開発環境/ターゲット環境で実施される内容です。

工程 1 : GUIの構造設計 (1/2)

目的と概要

- MxEでのGUI制作上必要なルールをデザイナー/エンジニア間で共有する工程です。
- 個々のデザインに入る前に、ターゲットの仕様に合わせて画面上のGUIパーツの構造設計をします。
- 利用するGUIObjectとPSDファイルの制作ルールを取り決め、デザイナーとエンジニアで共有することがこの工程のゴールです。

必要な知識

- 基本的なMatrixEngineの知識
- GUI Object(HMIContentsFramework)の知識
- PSDインポーターの使い方の知識
- 基本的なPhotoshopの知識

工程 1：GUIの構造設計 (2/2)

主な作業項目

□ レイヤー／ファイル構成決め

- 画面毎、機能毎にGUIパーツをグループ化したり、複数画面で共通のGUIを抽出したりすることで、PSDファイルのレイヤー構成（=MatrixEngineにおけるスコア構成）やファイル構成を決めます。

□ ルール決め

- PSDファイル内部で使用するレイヤー名（=MatrixEngineにおけるスコア名、キャスト名）のルールなどを取り決めます。

□ GUIステータスの洗い出し

- 複雑な動作をするGUIについては、必要となるGUIのステータスを洗い出しておきます。

工程 2：デザインの作成 (1/2)

目的と概要

- 工程1で取り決めたルールに沿って画面部品のデザイン（PSD）ファイルを制作します。
- MxE SDKで動作確認済みのPSDファイルを完成させることが、この工程のゴールです。

必要な知識

- 基本的なPhotoshopの知識
- MatrixEngine SDKの知識
 - （※初歩的な操作方法の知識があれば作業可能です。また、一部の定型文を除きスクリプト記述は不要です。）
- PSDインポーターの使い方の知識
- Photoshop用プラグイン（GUIObject Constructor）の使い方知識

工程 2：デザインの作成 (2/2)

主な作業項目

■ PSDファイル・素材データの作成

- デザイン作成後、PSDインポーター用にPSDファイルを作成します。
- トランジションアニメーションのパラメーターを設定、確認します。
- 9パッチ用のプロシージャルキャスト素材等、PSDファイル以外に提出が必要な一部素材を用意します。
- ソフト側からのデザイン関係のフィードバックがあればPSDファイルを修正します。

■ 動作確認

- 作成したPSDファイルをSDKにインポートしGUIの動作確認をします。
- 必要に応じて定型文のスクリプトを記述し、画面遷移等のアニメーションを確認します。

工程 3 ： スクリプトの実装 (1/2)

目的と概要

- 工程2で制作したPSDファイルを開発環境（MxE SDK）に取り込み、動きを実装します。
- ターゲット環境で動作を確認し、GUIコンテンツを仕上げるのがこの工程のゴールです。

必要な知識

- MatrixEngine SDKの知識
 - ※基本的には Photoshop の知識は必要ありません。

工程 3：スクリプトの実装 (2/2)

主な作業項目

□ 動作確認

- デザイナーから提出されたPSDファイルをインポートし、想定された構造になっているか確認します。
- デザイナーからPSDデータの更新があれば、再インポートして動作確認します。

□ 実装

- 設計に基づいてGUIコンテンツとイベント処理を実装します。

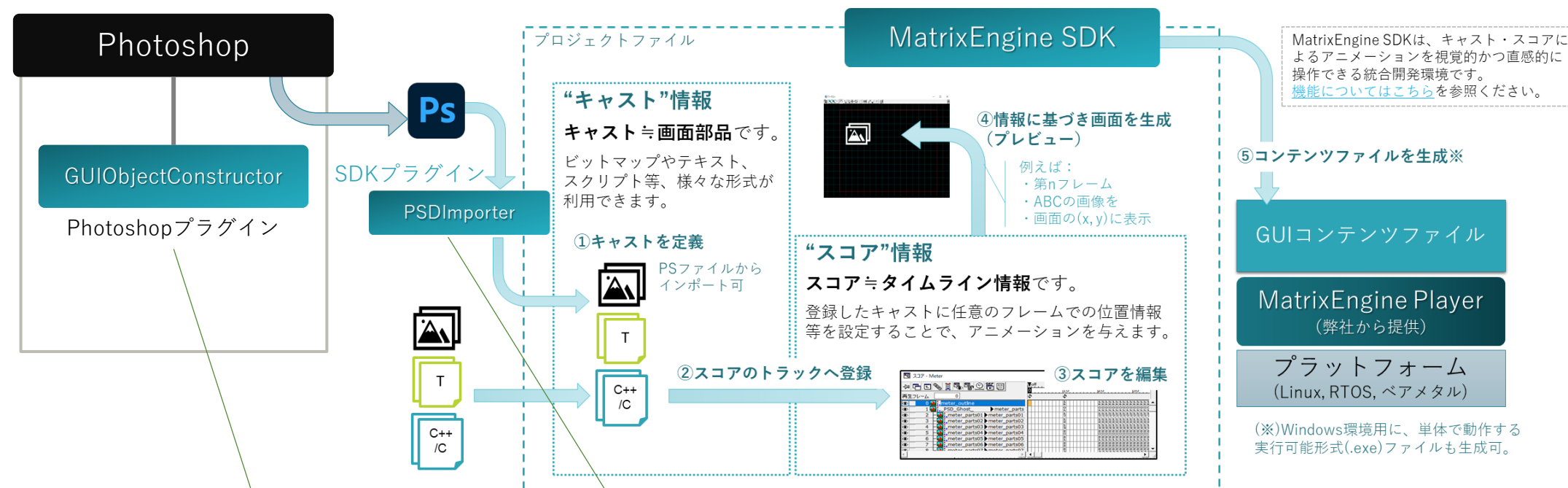
□ フィードバック

- スコア構成やレイアウトなどに問題があれば、デザイナーにフィードバックします。

MatrixEngineの基本的な仕組みと必要な知識

デザイン制作

GUIコンテンツ開発

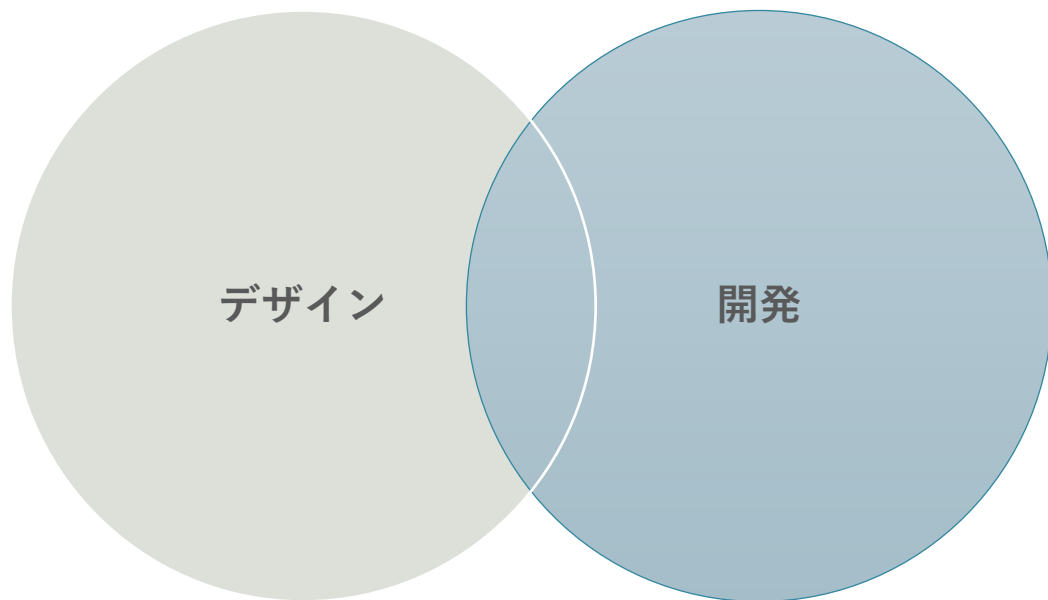


(※)Windows環境用に、単体で動作する実行可能形式(.exe)ファイルも生成可。

MatrixEngineでは、デザインと開発をスムーズにつなぐ様々な仕組みを提供しています

MatrixEngineの基本的な仕組みと必要な知識

- MxEでは、**デザイナーとエンジニアがいかに適切に協働できるかがGUIコンテンツ制作の重要なポイント**と考え、PSDインポーターとGUI Object連携を活用したGUI制作を推奨しています。



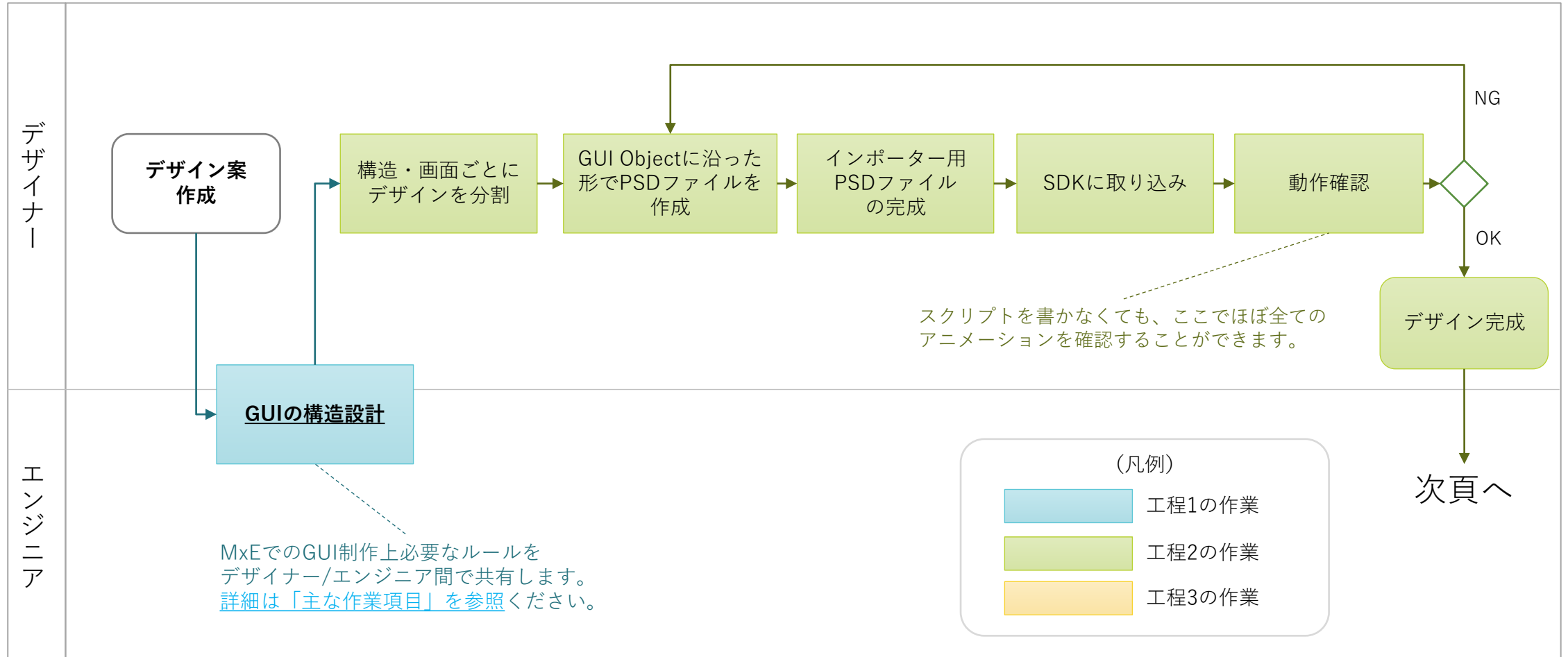
- 一般に、デザインやデザインツール（Photoshop）の知識はデザイナーが、開発や開発に必要なツールの知識はエンジニアが担います。MxEでも基本的に変わりはありません。
- ただし、デザイン領域とエンジニアリング領域両方をまたぐ「GUIの構造設計」のような工程では、**デザイナー/エンジニア両者がお互いの領域の知識を適切なレベルで理解すること**が大切だと考えます。

MatrixEngineは「ローコード」（≡習得容易性）を製品コンセプトの1つとしています

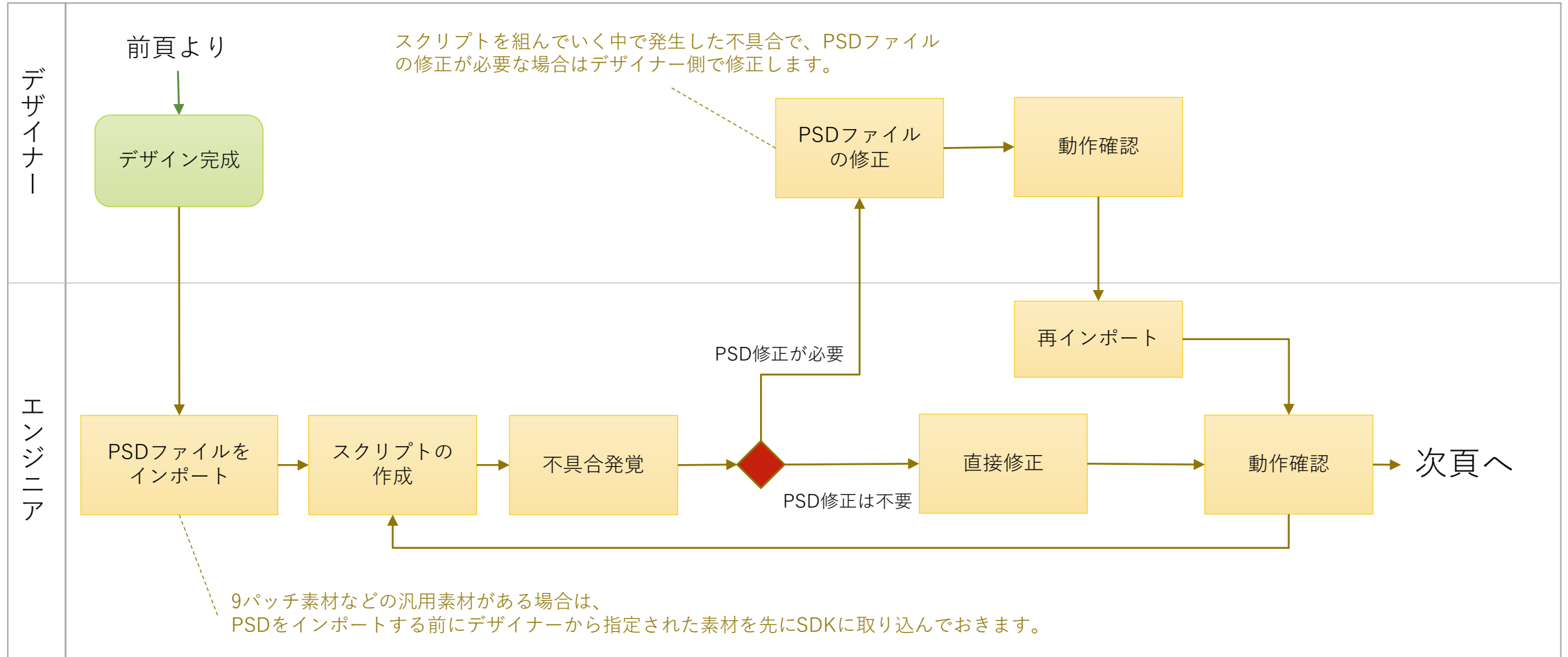
典型的なワークフロー

PSDインポーターとGUI Object連携を活用したGUI制作フロー

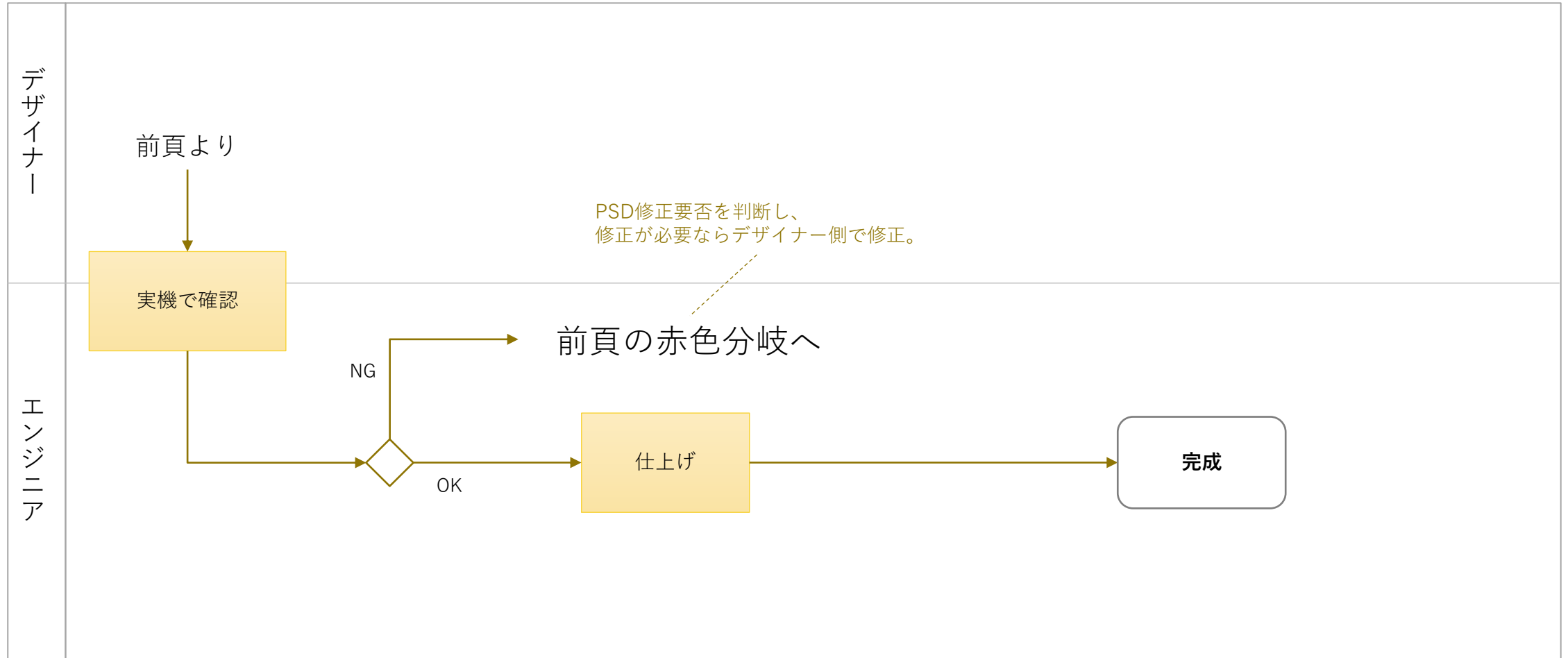
ワークフロー (1/3)



ワークフロー (2/3)



ワークフロー (3/3)



お問い合わせ

- 担当営業、もしくは以下よりお問い合わせください。
 - <https://www.matrixengine.jp/contact>
 - info@matrixengine.jp

『IoTデザインの未来を創る』



<https://www.matrixengine.jp>