

ポスター生成システム デザイン作業依頼書

作成日: 2025年11月17日 打ち合わせ日時: 2025年11月18日 (火) 15:00 依頼者: SORANO WORK様 受注者: インフィニティスタイル様

目次

1. システム全体概要
2. 文字デザインの仕組み
3. デザイン作業依頼内容
4. 参考資料

1. システム全体概要

1.1 ポスター生成支援システムとは

パチンコホール向けのポスター（新台入替、リニューアルオープンなど）を **AI とテンプレートを活用して自動生成** するシステムです。

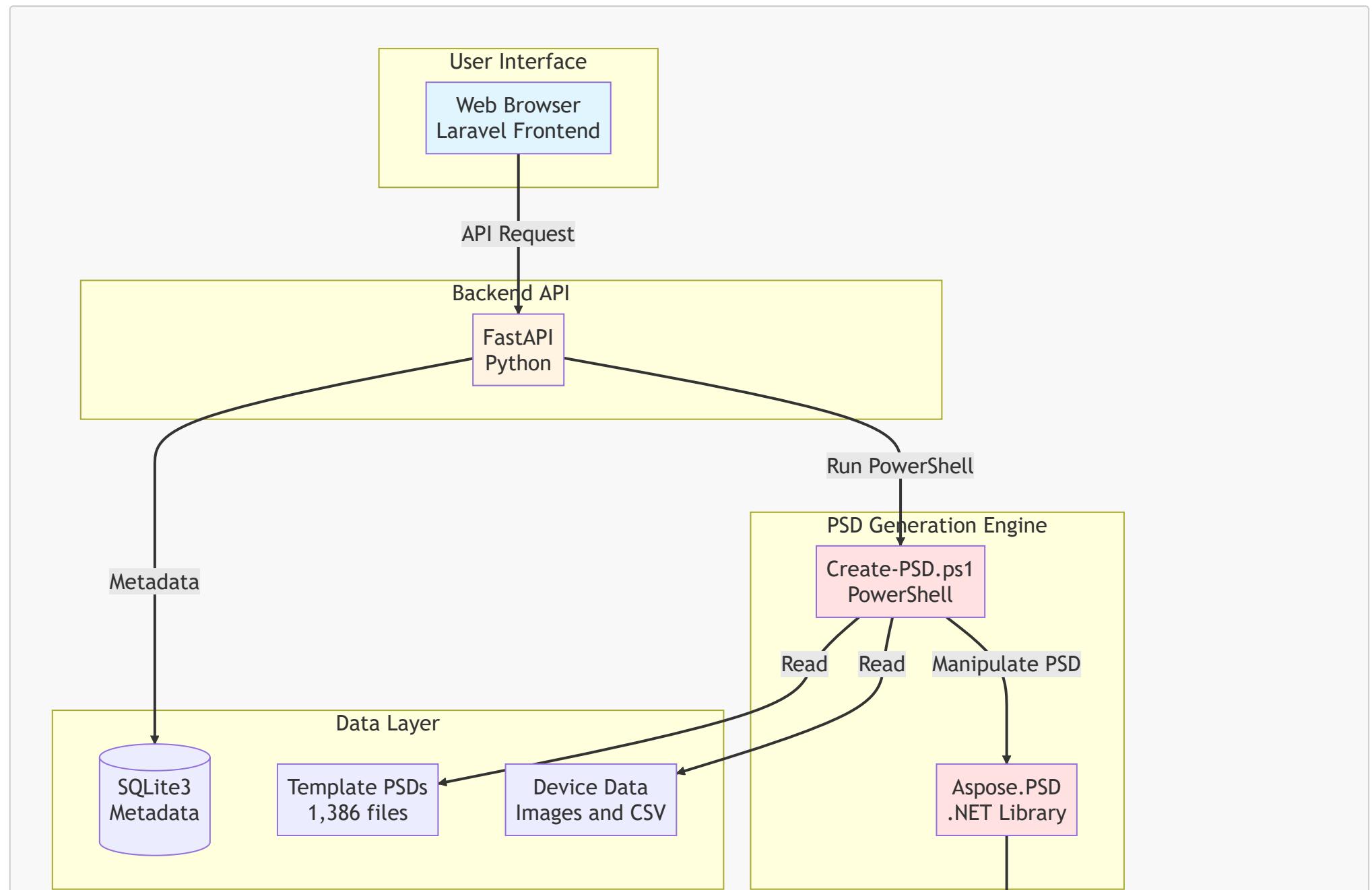
主な機能:

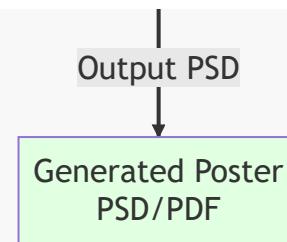
- ユーザーが簡単な条件入力（機種、日付、キャッチコピー等）を行うだけで高品質なポスターを生成
- 生成結果は PSD および PDF 形式で出力
- 過去の生成履歴をアーカイブとして保存・再利用可能

ビジネス価値:

- 専門知識がなくても店舗スタッフが自らポスターを作成可能
- 外部業者への発注コスト削減
- 迅速な販促物作成による集客力向上

1.2 システムアーキテクチャ

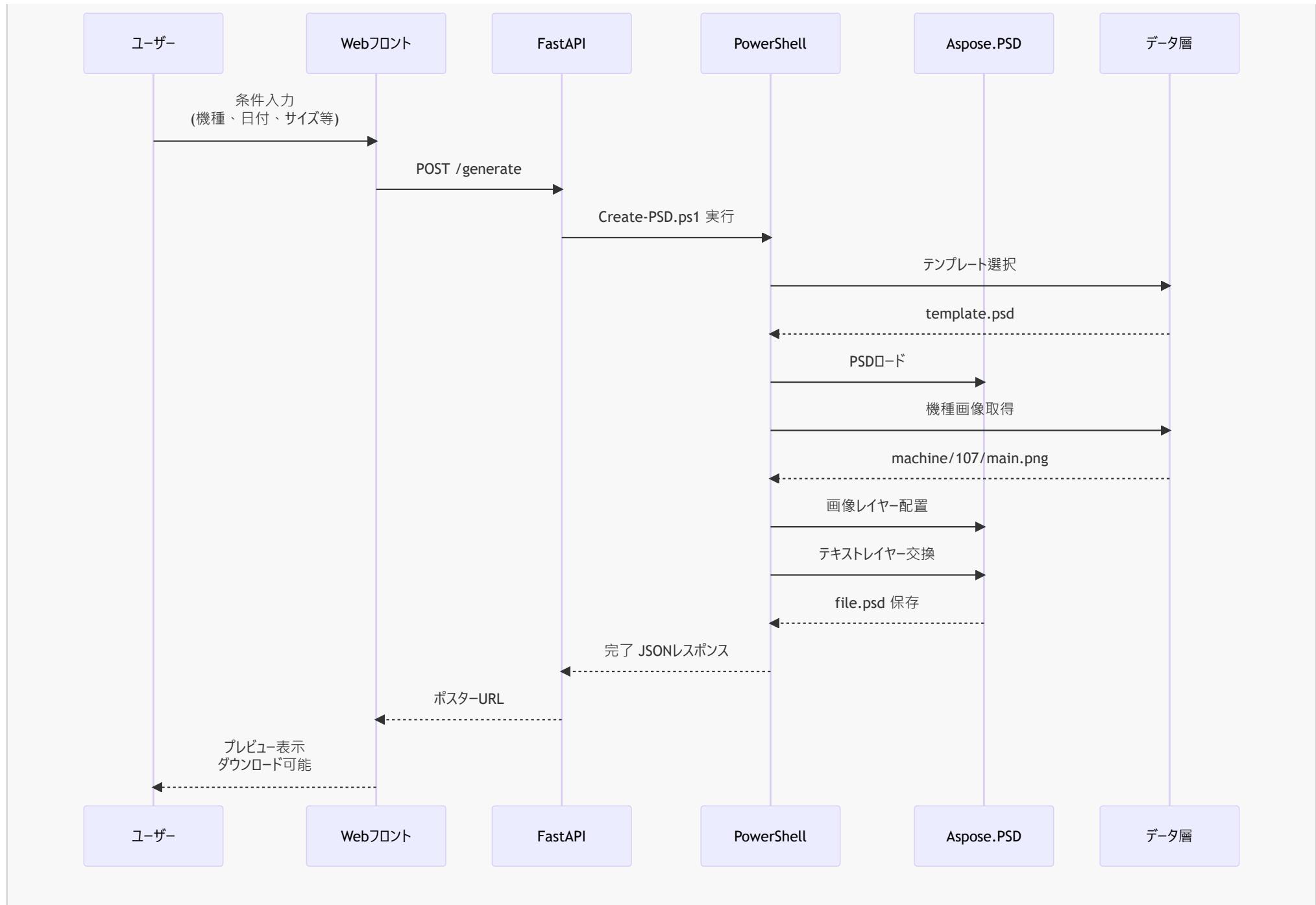




主要コンポーネント:

1. **フロントエンド** (Laravel/PHP): ユーザー操作画面
2. **バックエンドAPI** (FastAPI/Python): リクエスト処理、ビジネスロジック
3. **PSD生成エンジン** (PowerShell + Aspose.PSD): テンプレートPSDの操作・画像配置
4. **データ層**: テンプレートPSD、機種画像、メタデータDB

1.3 ポスター生成フロー



処理時間: 平均 300秒以内 (ベータ版目標)

2. 文字デザインの仕組み

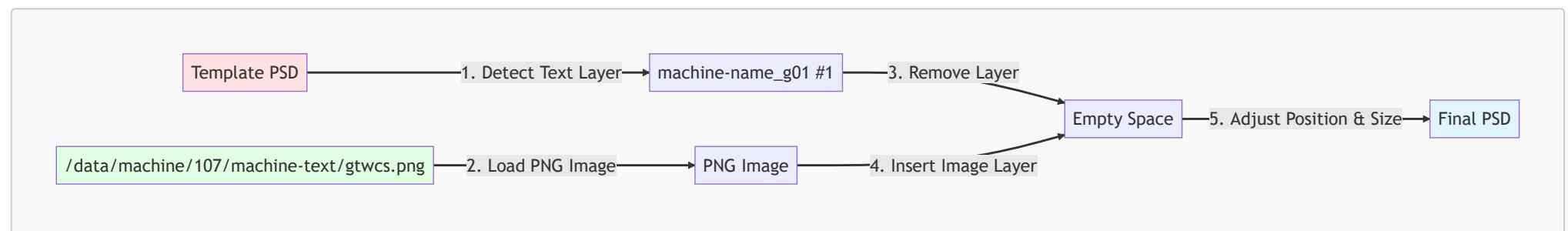
2.1 現在の実装：PNG画像によるテキスト置換

テンプレートPSD内のテキストレイヤー（機種名など）を **事前にレンダリングされたPNG画像で置き換える** 方式を採用しています。

採用理由:

- フォント依存性の排除（クロスプラットフォーム対応）
- デザイン品質の保証（レンダリング結果の完全一致）
- システムの移植性向上

動作原理:



レイヤー命名規則:

- 1番目の機種: `machine-name_g01 #1`
- 2番目の機種: `machine-name_g02 #1`
- 3番目の機種: `machine-name_g03 #1`

画像ファイル名の種類:

- `gtwcs.png`: Grand Text Wide, Color, Small
- `mtwcs.png`: Medium Text Wide, Color, Small
- `gtwcl.png`: Grand Text Wide, Color, Large

2.2 課題：305種類のSVGテンプレート作成が必要

2.2.1 背景

現在のPNG方式では、テキスト内容が **静的（固定）** であるため、動的にテキストを変更できません。

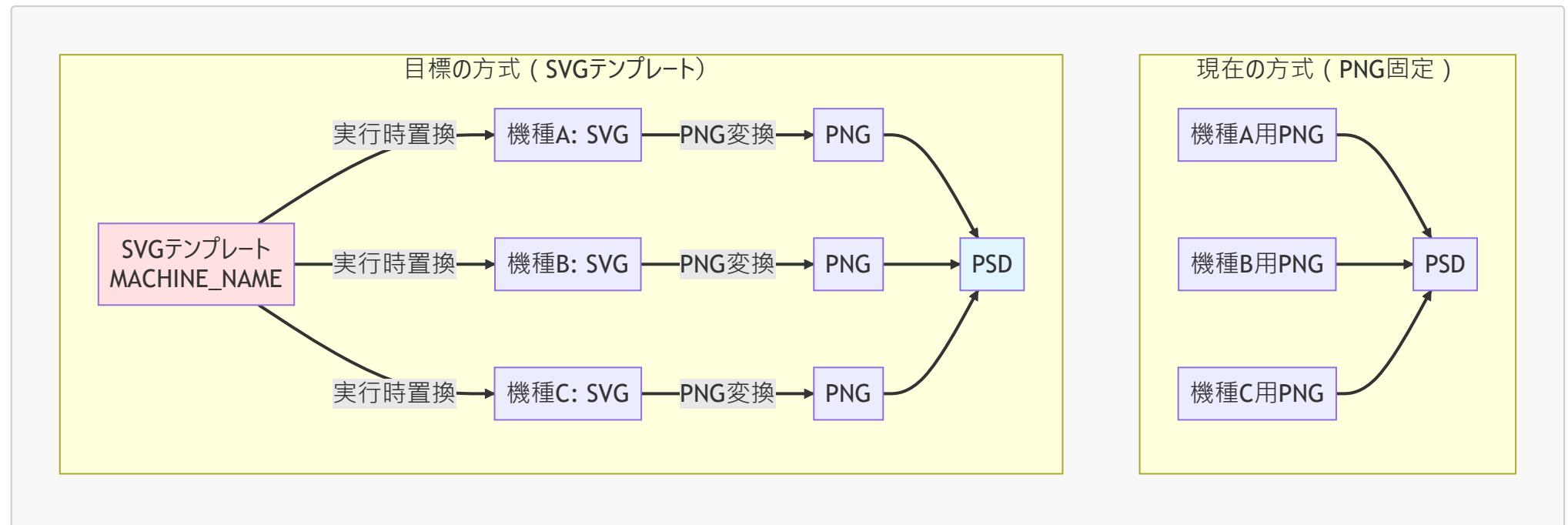
問題点:

- 機種名ごとにPNG画像を個別作成する必要がある
- テキスト内容の変更に柔軟に対応できない
- 運用コストが高い

解決策:

- **SVGテンプレート** を作成し、テキスト部分を変数化 (`{{{MACHINE_NAME}}}`など)
- 実行時にSVGのテキストを置換 → PNG変換 → PSDレイヤーに配置

2.2.2 SVG化の必要性



メリット:

- 1つのSVGテンプレートで複数の機種名に対応
- テキスト変更が容易 (変数置換のみ)
- メンテナンス性向上

2.2.3 対象データ

テンプレートPSDファイル数:

- 横向き: 174個
- 縦向き: 1,212個
- 合計: 1,386個

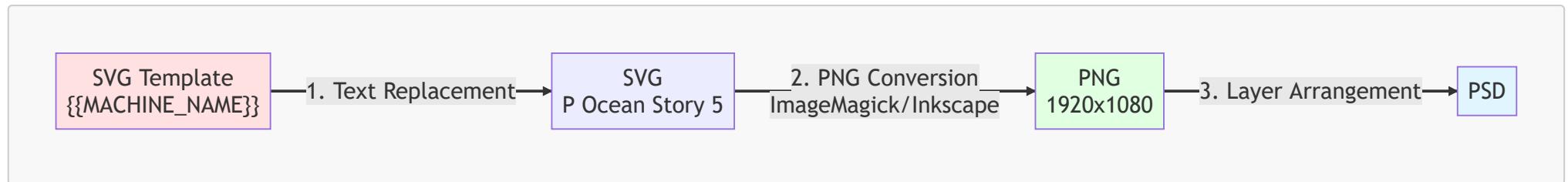
レイアウトパターン抽出結果:

- 同一レイアウトパターンを統合
- 必要なSVG種類: 305個

データソース:

- 圧縮ファイル: [poster-text-all.zip](#) (246MB)
- 含有PNG数: 約300個
- サンプル: [poster-text-all-example/](#) (3個)

2.3 SVG → PNG → PSD 変換フロー



変換ツール候補:

- ImageMagick
- Inkscape (CLI)
- Node.js (svg2png)
- Python (cairosvg)

3. デザイン作業依頼内容

3.1 作業概要

305個のPNG画像 を元に、**同じデザイン品質のSVGファイル** を作成していただきます。

最終目標:

- PNGの視覚的品質を **完全に再現** したSVGテンプレート
 - テキスト部分を **変数化** (`{{{MONTH}}}`, `{{{DAY}}}`, `{{{MACHINE_NAME}}}` など)
 - システムでの自動レンダリングが可能なSVG
-

3.2 詳細要件

3.2.1 ファイル仕様

| 項目 | 要件 |
|----------|--|
| ファイル形式 | SVG (Scalable Vector Graphics) |
| ファイル名 | PNG と同一 (拡張子のみ <code>.svg</code>) 例: <code>f49ccbec-xxxx.png</code> → <code>f49ccbec-xxxx.svg</code> |
| 解像度 | 対応するPNGファイルと同一 (幅・高さピクセル数を合わせる) |
| エンコーディング | UTF-8 |

3.2.2 デザイン品質要件

最重要: ポスターの印刷品質に直結するため、以下の品質基準を厳守

| 要件項目 | 詳細 |
|---------|---|
| テキスト位置 | PNGと完全一致（ピクセル単位） |
| フォント | PNGで使用されたフォントを特定し、SVGで再現 システムインストール済みフォント優先 |
| フォントサイズ | PNGと一致 |
| 文字色 | PNGと一致（カラーコード抽出） |
| エフェクト | PNGで使用されたエフェクト（影、グラデーション、境界線等）をSVGで実装 可能な限り同等以上の品質 |
| レイヤー構成 | 複数テキスト要素がある場合、レイヤー分離を維持 |

3.2.3 変数化仕様

テキスト部分を以下の形式で変数化してください:

変数形式: `{{VARIABLE_NAME}}`

変数例:

- `{{MONTH}}`: 月 (例: `6`)
- `{{DAY}}`: 曜日 (例: `16`)
- `{{WEEKDAY}}`: 曜日 (例: `月`)
- `{{MACHINE_NAME}}`: 機種名 (例: `P大海物語5`)
- `{{CATCH_COPY}}`: キャッチコピー
- その他、テンプレートに応じて柔軟に定義

SVG例:

```
<svg width="1920" height="1080" xmlns="http://www.w3.org/2000/svg">
  <text x="100" y="200" font-family="Noto Sans JP" font-size="72" fill="#FF0000">
    {{MONTH}}月{{DAY}}日 ({{WEEKDAY}})
  </text>
  <text x="100" y="300" font-family="Noto Sans JP Bold" font-size="96" fill="#000000">
    {{MACHINE_NAME}}
  </text>
</svg>
```

3.2.4 使用可能ツール

デザイン作業に使用するツールは問いませんが、以下を推奨します:

推奨ツール:

- Adobe Illustrator
- Inkscape (無料、オープンソース)
- Figma (SVGエクスポート対応)
- Sketch (macOS)

禁止事項:

- ラスター画像 (PNG/JPEG) の埋め込み
- 外部リソースへの依存 (外部CSS、外部フォントファイル)

3.3 納品物

3.3.1 成果物

- **SVGファイル**: 305個
- **ファイル命名規則確認リスト**: CSV形式
 - カラム: `png_filename, svg_filename, width, height, variables`
 - 例:

```
png_filename,svg_filename,width,height,variables
f49ccbec-xxxx.png,f49ccbec-xxxx.svg,1920,1080,"MONTH,DAY,WEEKDAY,MACHINE_NAME"
```

3.3.2 納品形式

- **圧縮ファイル**: `poster-text-all-svg.zip`
- **構成**:

```
poster-text-all-svg/
├── f49ccbec-xxxx.svg
├── f4d51bae-xxxx.svg
├── ...
└── file_list.csv
```

3.3.3 検収基準

- 305個のSVGファイルが全て揃っている
- ファイル名がPNGと一致している（拡張子除く）
- 各SVGがブラウザで正しく表示される
- テキスト変数が `{{VARIABLE_NAME}}` 形式で記述されている
- PNGとのビジュアル比較で品質差異が許容範囲内

- file_list.csv が提供されている

3.4 参考資料・サンプル

3.4.1 提供済みファイル

以下のファイルを本依頼書と同梱しています:

1. **poster-text-all.zip** (246MB)

- 約300個のPNG画像（作業対象）

2. **poster-text-all-example/** (サンプル3個)

- `f49ccbdc-e5a9-4922-a55a-0dfdec45ef9a-text-all.png`
- `f4d51bae-ca0b-4f63-8b5a-2afec474aa19-text-all.png`
- `feba63f9-f9a5-4e78-8976-8ce905a42210-text-all.png`

3. **poster-sample.pdf** (2.9MB)

- 実際に生成されたポスターのサンプル

3.4.2 サンプル画像（本ディレクトリ内）

地域最速最新台導入!

5/15 木 10:00 開店

サンプル画像1: 日付とテキストのレイアウト例

7月7日月 AM 9:00 OPEN

20円
スロット
増台！

サンプル画像2: 複数行テキストとエフェクト例

4.21 月 8:30 OPEN

3機種
30台導入

サンプル画像3: 機種名とキャッチコピー例

3.5 スケジュール

| 項目 | 期日 |
|--------------|---------------------|
| 打ち合わせ | 2025年11月18日（火）15:00 |
| 作業開始 | 打ち合わせ後、別途協議 |
| 中間確認 | 全体の10%完了時点（約30個） |
| 最終納品 | 別途協議 |

3.6 その他の留意事項

3.6.1 フォント権利関係

- システムで使用するフォントは商用利用可能なものに限定
- 推奨: Google Fonts (Noto Sans JP等)
- Adobe Fontsは使用不可 (ライセンス制約)

3.6.2 品質確認プロセス

1. サンプル確認 (5個程度)

- 初期段階で5個のSVGを作成・提出
- 品質基準の共通認識を確立

2. 中間レビュー (60個完了時点)

- 全体の20%完了時にレビュー
- 問題があれば方針調整

3. 最終検収

- 全305個の品質確認
- システム統合テスト

3.6.3 技術サポート

- SVG作成中の技術的質問は1営業日内に対応
- システム統合時の調整は当方で実施
- レンダリング品質の最終調整は協議の上対応

4. 参考資料

4.1 関連ドキュメント

- **システム仕様書:** [specification-appendix-1-20250601.md](#)
- **要求仕様補足:** [specification-non-tech-based-20250531.md](#)
- **既存作業依頼:** [request-20251001.txt](#)

4.2 システム詳細文書

既存の文書化資料（内部リンク）：

- [システムアーキテクチャ](#)
- [核心機能詳細](#)
- [開発者ガイド](#)

5. お問い合わせ

本依頼書に関するご質問・ご相談:

打ち合わせ日時: **2025年11月18日（火）15:00** Zoom URL: <https://us02web.zoom.us/j/6095743704?pwd=WHMxODB4N25RcXBWbmF3bnQ4N2RyQT09> ミーティングID: 6095743704 パス: 929264

以上

本依頼書は SORANO WORK 様とインフィニティスタイル様との間の技術的合意形成を目的としています。作業詳細、納期、費用等は別途契約書にて定めるものとします。