



mabl-cosme

生成AI系SaaSを模したE2Eテストデモアプリ



アプリケーション概要

mabl-cosmeとは

- **目的:** mabl + LLM による「UIコード生成」「自動テスト生成」のデモ題材
- **特徴:**
 - 生成AI系コスメ/ビジュアルSaaSを模したSPA
 - **JWT認証**付きAPI (mablのAPIテストデモに最適)
 - **data-testid**による安定したセレクタ設計
 - 多言語対応 (日本語/英語/中国語)



ワークフロー

典型的なSaaSワークフローを再現

1. ログイン (JWT認証)
2. 画像アップロード (製品写真などを想定)
3. AI背景生成 (DALL-E 3 API呼び出し)
4. 背景と元画像の合成 (Canvas API)
5. 色調補正 (色温度/彩度)
6. 保存・ダウンロード
7. ギャラリー表示



アーキテクチャ

技術スタック

レイヤー	技術
フロントエンド	React 18 + TypeScript + Vite
スタイルリング	Tailwind CSS (CDN)
バックエンド	Node.js + Express
認証	JWT (jsonwebtoken)
AI統合	OpenAI DALL-E 3
デプロイ	Docker + Google Cloud Run



JWT認証フロー

サーバーサイド認証の実装

1. POST /api/login (username, password)
↓
2. サーバーがJWTトークンを返却 { token: "eyJhbG..." }
↓
3. POST /api/openai (Authorization: Bearer <token>)
↓
4. OpenAI API呼び出し → 結果返却

- **トークン有効期限:** 24時間
- **認証エラー:** 401 Unauthorized



APIエンドポイント

mablのAPIテストで検証可能

エンドポイント	説明
POST /api/login	JWT認証（トークン取得）
POST /api/openai	AI背景生成（JWT必須）

レスポンス例:

- 認証成功: `{ "token": "eyJhbG..."}`
- 認証失敗: `{ "error": "Invalid credentials" }`



data-testid設計

テスト自動化のためのセレクタ

- **目的:** mablやPlaywrightがUI要素を一意に検出
- **命名規則:**
 - ボタン: `btn-{action}` (例: `btn-login`, `btn-save`)
 - 入力: `{name}` (例: `username`, `password`)
 - セレクト: `{name}-select` (例: `lang-select`)

ポイント: CSSクラスやDOM構造が変わっても、`data-testid`は安定



主要なdata-testid一覧

mablテストで使用するセレクタ

カテゴリ	data-testid	説明
認証	username	ユーザー名入力
認証	password	パスワード入力
認証	btn-login	ログインボタン
認証	login-error	エラー表示
画像	btn-upload	アップロード
画像	btn-ai-generate	AI生成
保存	btn-save	保存ボタン



テストシナリオ A

基本E2Eフロー

1. アプリを開く
2. ユーザー名・パスワードを入力 → ログイン
3. 画像をアップロード
4. 「背景をAIで生成」ボタン押下
5. 色温度・彩度スライダーで調整
6. 「保存」ボタン押下 → ギャラリーに追加
7. 「ダウンロード」でPNGを保存



テストシナリオ A'

認証エラー

1. アプリを開く
2. 不正なユーザー名・パスワードを入力
3. `login-error` に "Invalid credentials" 表示
4. 正しい認証情報を入力 → ログイン成功

検証ポイント: エラーメッセージの表示確認



その他のテストシナリオ

シナリオ B~E

シナリオ	内容
B: 多言語確認	言語切替でUI変化を確認
C: 入力エラー	10MB超/非画像ファイルのエラー
D: 画像なし操作	画像未選択でのアラート確認
E: APIサーバー切替	同一/別ホストの切替



環境変数

設定項目

変数名	説明	デフォルト
OPENAI_API_KEY	OpenAI APIキー	(必須)
AUTH_USERNAME	ログイン用ユーザー名	demo
AUTH_PASSWORD	ログイン用パスワード	demo123
JWT_SECRET	JWT署名用シークレット	(自動生成)



デプロイ

GitHub Actions + Cloud Run

```
main ブランチへのpush  
↓  
GitHub Actions トリガー  
↓  
Docker イメージをビルド  
↓  
Artifact Registry にプッシュ  
↓  
Cloud Run にデプロイ
```



開発コマンド

ローカル開発

```
# 開発サーバー起動（フロント+バック同時）
npm run dev:all
```

```
# Docker で起動
docker compose up -d --build
```

```
# ビルド
npm run build
```



まとめ

mabl-cosmeの特徴

- **JWT認証**: サーバーサイドJWT認証によるセキュアなAPI
- **実際のAI連携**: DALL-E 3 APIによる背景生成
- **テスタブル設計**: data-testidによる安定したセレクタ
- **多言語対応**: 日本語/英語/中国語
- **クラウドデプロイ**: Docker + Cloud Run + GitHub Actions

ご清聴ありがとうございました！