

次世代型ソフトウェア開発モデルレビュー支援ツール「ModelLogue」詳細レポート

1. ビジョン：成果物から「意思決定プロセス」の資産化へ

従来のシステム開発では、レビューの結果（修正された図面）のみが記録され、そこに至るまでの「なぜこの設計になったのか」「どのような懸念が議論されたのか」というプロセスは、散逸する傾向にありました。ModelLogueは、図面を「一時的な思考のキャンバス（Volatile Canvas）」と定義し、人間とAIの「対話ログ」を設計の本質的な資産として扱う、新しいモデルレビュー・プラットフォームを目指します。

2. コア・アーキテクチャ：疎結合型ハブ・モデル

本ツールは機能を抱え込まない「持たないアーキテクチャ」を徹底し、外部の専門サービスをオーケストレーションすることで機能します。

- **フロントエンド（Review Hub）**：React Flowを搭載し、モデル図を「指差し」しながらAIと対話できるインターフェース。
- **オーケストレーター（n8n）**：
 - SlackやTeamsからの非同期フィードバックの収集。
 - Gemini APIによるソースコード修正のトリガー。
 - GitHubへの「ソース+会話ログ」の自動パブリッシュ。
- **推論エンジン（Gemini 2.5 Flash）**：単なるコード修正ではなく、モデルの背後にある議論の文脈（Context）を読み取り、設計意図を汲み取った改善案を提案。
- **描画エンジン（PlantUML Server）**：ソースコードを即座にSVG化し、セッション中のみ有効なプレビューを提供。

3. 非同期レビューによるパラダイムシフト

「リアルタイムの会議」から「非同期のコラボレーション」へシフトすることで、以下の価値を創出します。

- **思考の深化と効率化**：レビュー参加者は、SlackやTeamsを通じて、自身の隙間時間や最も集中できるタイミングでモデルに対するフィードバックを送ることが可能。
- **AIによる議論の自律集約**：AIがチャット上の分散した意見をリアルタイムで解析。矛盾点や未解決事項を抽出し、ModelLogue上のモデルへ反映するドラフトを常に最新に保ちます。
- **場所と時間の制約からの解放**：物理的な集合を前提としないため、グローバルチームやリモートワーク環境における品質担保のボトルネックを解消します。

4. 究極のトレーサビリティ（証跡管理）

「Canvasは一時的ですぐに消える」という特性を、逆に「外部ストレージ（GitHub）への即時同期」のトリガーとして活用します。

- **議論のパッケージ化:** 最終承認が下りた際、修正されたモデル図のソースコードだけでなく、そこに至るまでの全チャット履歴を「レビュー証跡」としてGitHubへセットでコミットします。
- **ナレッジの継承:** 後任の担当者は、GitHubのリポジトリを見るだけで、過去のレビューでのやり取りを振り返り、設計の背景を完璧に把握することができます。

5. 実装ロードマップ：段階的進化

1. **Phase 1 (MVP):** PlantUMLソースの受領・表示と、AIによる対話型修正の基本フローの確立。
2. **Phase 2 (Collaborative):** React Flowによる図の要素単位でのポインティング・レビューと、Slack連携による非同期フィードバックのプロトタイプ。
3. **Phase 3 (Enterprise):** GitHub APIとの完全統合による自動コミットと、n8nによる業務フローへの完全な溶け込み。

6. 結論

ModelLogueは、単に図を描くためのツールではありません。それは、「開発者の対話」を最大化し、ソフトウェアモデルを確かな資産として残すためのインターフェースです。AIと外部ツールの力を借りることで、レビューを「重たい会議」から「クリエイティブな対話」へと変革します。

作成日: 2026年2月7日 作成者: プロジェクト推進チーム