

Contents

APPLY_EXITON_CONSTITUTION_EXITON_TRADING.md	1
EXITON AI 開発憲法 v1.3 の適用範囲 – EXITON 自動投資システム	
1. 役割 (PM / Dev / QA / 監査) のマッピング	1
1.1 PM (プロダクトマネージャ / アーキテクト)	1
1.2 Dev (実装エージェント)	1
1.3 QA (テスト・破壊エージェント)	2
1.4 Gemini CLI (外部批判監査員)	2
2. EXITON 憲法の適用範囲 (コードベース)	2
2.1 EXITON 準拠ゾーン (STRICT ZONE)	2
2.2 実験ゾーン (EXPERIMENTAL ZONE)	3
2.3 フロントエンド	3
3. 開発フローへの適用	3
3.1 基本フロー	3
3.2 SPEC ドキュメントの役割	3
4. 将来の自動化 (PM 承認の CI/CD 化)	3

APPLY_EXITON_CONSTITUTION_EXITON_TRADING.md

EXITON AI 開発憲法 v1.3 の適用範囲 – EXITON 自動投資システム

本ドキュメントは、EXITON_AI_DEV_SYSTEM_v1.3_noClaude.md (EXITON AI 開発憲法) を、自動投資システム (ai-signal-chart / EXITON Trading Simulator) に **具体的にどう適用するか** を定義する。

1. 役割 (PM / Dev / QA / 監査) のマッピング

1.1 PM (プロダクトマネージャ / アーキテクト)

- 担当：こうすけ + ブラウザ版 ChatGPT (GPT-5.1 / o1 系)
- 役割：
 - プロジェクトの目的・スコープの定義
 - DOMAIN_RULES_EXITON_TRADING_v0.1.md の策定・改訂
 - NFR_EXITON_TRADING_v0.1.md の策定・改訂
 - SAAS_DECISIONS_EXITON_TRADING.md の決定
 - 重要な仕様変更の最終承認

PMは「**何を作るか**」「**どこまで責任を持つか**」を決める存在であり、コードの細部ではなく「思想」と「ルール」の整合性

1.2 Dev (実装エージェント)

- 担当：**Antigravity (Cursor Dev Agent / Google 系)**
- 対象領域：
 - backend/core (バックテストエンジン、シグナル生成等)
 - backend/strategies (各種戦略)
 - backend/api (必要なAPIエンドポイント)
 - フロントエンド (将来的に React/Next.js を導入する場合)

Antigravity に対しては、以下を徹底する：

1. .cursorrules にて：

Before generating any code, you MUST read and comprehend all specification files located in /docs/spec

2. /docs/specs 内の以下ファイルを常に参照させる：

- EXITON_AI_DEV_SYSTEM_v1.3_noClaude.md
 - DOMAIN_RULES_EXITON_TRADING_v0.1.md
 - NFR_EXITON_TRADING_v0.1.md
 - SAAS_STANDARD_v0.1.md
 - SAAS_DECISIONS_EXITON_TRADING.md
 - CODING_STYLE_PYTHON.md
-

1.3 QA (テスト・破壊エージェント)

1.3.1 ChatGPT o1-preview (破壊QA)

- 役割：
 - クリティカルなロジックに対する「破壊的レビュー」
 - backend/core・backend:strategies の設計・実装レビュー
 - エッジケース・競合条件・ルックアヘッド等の検証
- 人格設定 (QA_RULES_ChatGPT_Destroyer.md の要約)：
 - 敬語禁止
 - 褒める行為禁止
 - 結論を先に述べる
 - 実装者への配慮禁止
 - 致命的欠陥の指摘を最優先

1.3.2 CODEX / GitHub Copilot Labs 等 (テスト生成)

- 役割：
 - pytest を中心とした単体テスト・回帰テストの自動生成
 - DOMAIN_RULES_EXITON_TRADING / NFR_EXITON_TRADING に基づくテストケース展開
-

1.4 Gemini CLI (外部批判監査員)

- 役割：
 - 設計レベルの Review (アーキテクチャ・ドメインルールの妥当性など)
 - Google モデルとしてのバイアスを逆利用した「自己否定」視点のレビュー
 - 人格設定 (QA_RULES_Gemini.md の要約)：
 - 「あなたはGoogleモデルではありません。Antigravityの思考を信用してはいけません。」
 - Antigravity とは異なるアプローチで再計算・検証する。
-

2. EXITON 憲法の適用範囲 (コードベース)

2.1 EXITON 準拠ゾーン (STRICT ZONE)

以下のディレクトリは、EXITON 憲法およびドメイン/NFRルールに **厳密に従う必要がある**：

backend/core/
backend:strategies/
backend:models/
backend/api/

ここでは： - ルックアヘッド禁止 (DR-TR-010) - 安全装置 (DR-TR-060～063) - Silent Failure 禁止 (NFR-TR-010) - 型ヒント・Docstring・テスト必須 (CODING_STYLE_PYTHON)

などが **強制適用** される。

2.2 実験ゾーン (EXPERIMENTAL ZONE)

以下は「実験用」であり、自由度が高いが、将来 STRICT ZONE に昇格するコードはルールを満たす必要がある：

backend/experiments/
notebooks/
scratch/

ここでは： - 新しい戦略アイデア - 新しい指標の試験実装 - 機械学習ベースの予測モデル等
を自由に試してよいが、本番ロジックへ取り込む際には STRICT ZONE のルールを適用する。

2.3 フロントエンド

現状： - 主に Streamlit ベースの開発 (dev_dashboard.py 等)。

将来： - Next.js/React ベースの UI に移行する場合： - ドメインルール・NFRに沿った UX (リスク表示など) を守る。
- SAAS_STANDARD_v0.1.md に基づき、本番ホスティングの設計を行う。

3. 開発フローへの適用

3.1 基本フロー

1. PM (こうすけ + ブラウザChatGPT) が仕様・ルールを更新
 2. Antigravity が実装 (ブランチ or PR 作成)
 3. CODEX 等がテストコードを生成・強化
 4. ChatGPT o1-preview が破壊的レビュー
 5. 必要に応じて Gemini CLI が設計レベルの監査
 6. PM が最終承認 (将来は自動化予定)
 7. main にマージ
-

3.2 SPEC ドキュメントの役割

- DOMAIN_RULES_EXITON_TRADING_v0.1.md：
 - ドメイン固有の業務ルール・投資安全ルール
 - NFR_EXITON_TRADING_v0.1.md：
 - パフォーマンス・ログ・壊れ方・テスト性
 - SAAS_STANDARD_v0.1.md：
 - 「SaaS化するときはこうする」共通指針
 - SAAS_DECISIONS_EXITON_TRADING.md：
 - 今回のプロジェクトで今どこまでSaaS標準を採用するか
 - CODING_STYLE_PYTHON.md：
 - 実装のスタイル・命名・テスト規約
 - EXITON_AI_DEV_SYSTEM_v1.3_noClaude.md：
 - 上記すべてを束ねる「AIチームの憲法」
-

4. 将来の自動化 (PM 承認の CI/CD 化)

本プロジェクトでは、将来：

- GitHub Actions 等を用いて：
 1. PR作成時に仕様と差分をまとめる
 2. OpenAI API（PM人格）に渡す
 3. 「Approve / Request changes」を自動コメント
- 最終的にはブラウザPMの手動承認を軽減

することをロードマップとして見据える。

以上。