Covenant Personal Edition — ユーザーマニュアル

本書は Covenant Personal Edition の**使い方**を整理したユーザーマニュアルです。アプリのインストールは **インストールガイド**、Ollama 導入から初回利用までの最短手順は **クイックスタートガイド** を参照してください。

---

0. 本書の位置付け

0.1 対象読者

- インストール・初回セットアップ完了後、Covenant Personal Edition の操作を理解したいユーザー
- 機密レベル設定やポリシー編集、LAN別ノード / クラウド AI などの応用的な使い方を知りたいユーザー

0.2 関連文書との役割分離

文書	用途
インストールガイド	アプリ本体のインストール手順
クイックスタートガイド	Ollama 導入から初回利用までの最短手順
本書 (ユーザーマニュアル)	Gate の仕組み・機密レベル設定・日常操作・高度な設定 (LAN別ノード / クラウド AI)
オンラインヘルプ (アプリ内蔵)	包括的ヘルプ (Part 1 ユーザー向け + Part 2 業務の相手方向け)

0.3 表現規律

本書は claim 文書ではなく **操作手順開示** です。

- ✅ 採用: 「評価が実行されます」「結果が表示されます」
- ❌ 禁則: 「100% 正確な評価」「絶対に漏洩しません」

AI 判定は **評価** (確率的、保証なし)、ポリシーは **判定** (決定的、設定値ベース)、最終的な **判断** はユーザー (人間) が行う三段構造です。

---

1. 全体像（処理の流れ）

Covenant はメッセージを送信する前に、以下の Gate パイプラインで評価します。インストール直後のセットアップ手順は **クイックスタートガイド** を参照してください。

```
ユーザー入力
    ↓
Gate 0（Input、入力前処理）
    ↓
Gate 1（Prefilter、明白なルール違反の高速検知）
    ↓
Gate 2（Field Policy、フィールド・規範ポリシー判定）
    ↓
Gate 2.5（Context、文脈評価 = ローカル LLM）
    ↓
Gate 2.8（Harmless、無害化前段スクリーニング = ローカル LLM）
    ↓
Gate 3（Gradation、機密度グラデーション評価 = ローカル LLM）
    ↓
[PASS] → 送信先選択（ローカル / クラウド）
[YELLOW / GRAY] → ユーザー承認待ち
[RED] → 高リスク評価（既定では内容を確認のうえ承認すれば続行可能）
```

上記の PASS / YELLOW / GRAY / RED は **AI による評価結果**です。一方、Gate 1 / 2 で**ご自身が設定した明示的なポリシールール**に該当した場合は、そのルールに指定した動作（ブロック / 承認待ち / 警告）が適用されます。**AI の評価は警告までで、確実に止められるのは明示ルール**という役割分担です（詳細は **オンラインヘルプ**）。

Gate 2.5 / 2.8 / 3 で使われるモデルの詳細は、アプリのメニューから「📖 ヘルプ」でも確認できます。

---

2. 主要 Gate（0-3）の動作

2.1 Gate 0 — Input（入力前処理）

ユーザー入力の前処理（改行正規化、文字エンコーディング統一等）。ユーザー操作なし。

2.2 Gate 1 — Prefilter（明白な違反検知）

YAML ポリシー（`org.yaml` / `project.yaml`）に明示された明白なルール違反（例: 機密キーワードの直接記述、特定パターン）を高速検知。LLM を使わず、決定的に判定。

2.3 Gate 2 — Field Policy（フィールド・規範ポリシー判定）

`project.yaml` および個人層ポリシーに基づき、入力内容のフィールド（個人情報 / 業務情報 / 技術情報 等）と規範を照合。決定的判定。

> 用途別 YAML の使い分け: `project.yaml` ＋個人層ポリシーは Gate 2 (Field Policy)、`rubric.yaml` は Gate 2.5 / 2.8 / 3 (LLM による評価) で使用されます。

2.4 Gate 2.5 — Context（文脈評価、ローカル LLM）

決定的判定で曖昧な箇所について、ローカル LLM（デフォルト `gemma3n:latest`、Settings で切替可能）が文脈評価を実施。`pending_rules` が解決される。

2.5 Gate 2.8 — Harmless（無害化前段スクリーニング、ローカル LLM）

機密度評価の前段として、無害化判定（低機密度のクイックパス）を実施。

2.6 Gate 3 — Gradation（機密度グラデーション評価、ローカル LLM）

最終的なリスク評価。`rubric.yaml` に基づき、**倫理 (ethics)** と **安全 (safety)** の 2 軸で評価します。

各軸について、**発生可能性 (probability) / 影響度 (impact) / 緊急度 (urgency)** の 3 項目を **1〜5 の 5 段階**で評価し、その結果から各軸 **0〜10 のスコア**を算出します。しきい値（既定: 警告 5.1 / 高リスク 8.0）で評価結果が決まります。

評価結果には `[P:5 I:1 U:1]` のように 3 項目の評価値が表示されるため、**どの観点でリスクありと評価されたか**を確認できます（例: 影響度・緊急度は低いのに発生可能性だけが高い、など）。

機密レベル (L1〜L4) の分類とあわせて、送信先 LLM の振り分けに使用します。

詳細は **オンラインヘルプ** Part 2 を参照してください。

---

3. 機密レベル設定

3.1 機密レベル L1〜L4 の定義

機密レベル	内容	送信先（デフォルト）
機密レベル L1（公開）	公開情報、機密度なし	ローカル LLM / クラウド LLM（ユーザー設定）
機密レベル L2（社外秘）	業務情報、低機密度	ローカル LLM / クラウド LLM（ユーザー設定）
機密レベル L3（秘）	業務情報、中機密度	**ローカル LLM のみ**（クラウド送信なし）
機密レベル L4（極秘）	極秘情報、削除可能性あり	**ローカル LLM のみ** + 即時完全削除可能

3.2 ポリシー編集による調整

機密レベル分類は、ユーザーが `rubric.yaml` / `org.yaml` / `project.yaml` を編集することで調整可能です（「**ポリシーを育てる**」設計）。

編集方法

- **直接編集**: VSCode 等のエディタで YAML ファイルを編集
- **Policy Translator**: 自然言語からの YAML 変換（NL → YAML、Translator がローカル LLM で変換）
- **ポリシーホットリロード**: 編集後の保存で即時反映、アプリ再起動不要

詳細は **オンラインヘルプ** Part 1 §3 を参照してください。

---

4. ローカル LLM の選び方（チューニング）

初回セットアップ時のモデル選定（PC スペック別の推奨モデル早見表）は **クイックスタートガイド §2** を参照してください。本節は、運用を始めた後の**調整**を扱います。

4.1 モデルの切り替え手順

Settings 画面の「LLM プリセット」、または `models.yaml`（Windows: `%APPDATA%\Covenant\models.yaml` / macOS: `~/Library/Application Support/Covenant/models.yaml`）を編集します。モデルは事前に Ollama でダウンロードしておく必要があります（例: `ollama pull llama3.1:8b`）。

4.2 困ったときの対処

症状	対処
応答が遅い / PC が熱くなる	1 段階軽量なモデルに切り替え、または §7.1 (LAN別ノード構成) を検討
応答品質に物足りなさを感じる	同じ構成の範囲で別系統のモデルを試す (Llama / Qwen / Gemma)
モデルが起動しない / アプリが不安定	最も軽量なモデル (Llama 3.2 1B / Phi-3 mini) に切り替え

> Gate 評価用モデル (Gate 2.5 / 2.8 / 3) は、チャット応答用とは別に Settings で設定します。本節はチャット応答用モデルの調整を対象としています。

---

5. よくある操作

5.1 メッセージ送信と評価結果の確認

1. メイン画面の入力欄にテキストを入力
2. 送信ボタンをクリック
3. Gate パイプラインが走り、評価結果が表示される (PASS / YELLOW / GRAY / RED)
4. PASS の場合、応答が表示される (送信先 LLM はメッセージ下部に表示)

5.2 監査ログ (audit.jsonl) の確認

Covenant は全 Gate 評価結果を `audit.jsonl` にローカル保存します (追記型のプレーンテキストログ)。**改ざん検知のためのハッシュチェーン機構は現状実装していません** (将来のバージョンで検討予定)。ログはご自身で内容を確認・検証できます。

```
# Windows
Get-Content "$env:APPDATA\Covenant\logs\audit.jsonl" | Select-Object -Last 30
```

```
# macOS
tail -n 30 ~/Library/Application\ Support/Covenant/logs/audit.jsonl
```

5.3 スレッド (会話履歴) 管理

- **新規スレッド作成**: メイン画面左サイドバー「+」ボタン
- **スレッド検索**: 左サイドバー上部の検索欄

- **スレッド削除**: 右クリック → 「削除」(機密レベル L4 相当のデータは即時完全削除可能)
- **保存期間**: **アーカイブ済み**スレッドを既定 90 日で自動削除します。設定画面で 0〜3650 日に変更でき、**0 を設定すると自動削除は行われません。アーカイブしていないスレッドは自動削除の対象外**で、期限なく保持されます

5.4 ライセンス情報の確認

Settings 画面 → 「ライセンス情報」で、現在のライセンスキー (マスク表示) + 有効期限 + 使用可能台数を確認できます。

5.5 Bug Report 機能

問題を発見した場合は、メイン画面メニュー → 「Bug Report」から自発的に送信可能です。送信される情報は version / platform / license 状態 / audit_tail のみ (API キー等の機密情報は含まれません)。

---

6. プライバシーと透明性

6.1 ローカルファースト原則

Covenant は **ローカルファースト** で設計されており、ユーザー入力・評価結果・監査ログはすべてユーザー PC 上に保存されます。クラウド送信が発生するのは以下の場合のみです:

- **クラウド LLM 利用時** (ユーザー設定で明示的に有効化、機密レベル L1 / L2 のみ送信)
- **ユーザー自発の Bug Report 機能** (ユーザー操作後の送信)

6.2 実装しない機能・行わない運用

Covenant は、以下の実装・動作は行いません:

1. ❌ 利用ログを Exapolicy LLC のサーバーに送信しません (テレメトリ非実装)
2. ❌ ユーザーデータを AI モデルの学習に使いません
3. ❌ 広告を表示しません
4. ❌ ユーザー情報を第三者に販売しません
5. ❌ ユーザーの同意なく、データの取り扱いに関わる機能を変更しません
6. ❌ Personal Edition のクラウド SaaS への強制移行を行いません
7. ❌ 課金モデルの一方的な切替を行いません

---

7. 高度な設定

初回セットアップ (構成 A、自分の PC でローカル LLM を動かす) は **クイックスタートガイド §1** で完結します。本節は、それ以外の構成を選ぶ場合の実装詳細です。

7.1 LAN 別ノード構成

こんな方に: ノート PC のスペックに不安がある / 高性能な別の PC (デスクトップ、Mac mini 等) が LAN 内にある。

ローカル LLM を別の 1 台 (「ノード」) に任せることで、ノート PC 自体は軽いまま、ガバナンス機能を制限なく使えます。

ノード側 (LLM を動かす PC) の設定

1. ノードにする PC に Ollama をインストール
2. モデルを取得: `ollama pull llama3.1:8b`
3. **他の PC から接続できるように Ollama を設定**: - **Windows**: システム環境変数に `OLLAMA_HOST` = `0.0.0.0:11434` を設定 → Ollama を再起動 - **macOS**: ターミナルで `launchctl setenv OLLAMA_HOST "0.0.0.0:11434"` → Ollama を再起動
4. ノードの **LAN 内 IP アドレス**を確認: - Windows: コマンドプロンプトで `ipconfig` → IPv4 アドレス (例: `192.168.1.100`) - macOS: システム設定 > ネットワーク、またはターミナルで `ipconfig getifaddr en0`

Covenant 側 (ノート PC) の設定 — Settings 画面で設定 (推奨)

機密度評価を行う Gate 3 を、ノードの IP に向けます。**Settings 画面から操作するのが簡単です** (テキストエディタでの編集ミスを防げます)。

1. Covenant の **Settings 画面**を開く
2. 「**LLM プリセット**」で「新規プリセット追加」をクリックし、以下を入力: - プリセット名: `LAN ノード Llama 3.1 8B` (任意の分かりやすい名前) - プラットフォーム: `ollama` - モデル: `llama3.1:8b` - ホスト: `http://192.168.1.100:11434` (← 上記で確認したノードの IP)
3. 「**Gate 3 — 倫理・安全スコア**」セクションで、プリセット欄から今作成したプリセットを選択
4. 設定は自動保存されます

> **画像解析 (VLM) も別ノードに任せる場合**: Settings の「VLM — 画像解析」セクションでホストを同じノードの IP に設定します。ノード側で `ollama pull minicpm-v` が必要です。

補足: テキストエディタで直接編集する場合（上級者向け）

`models.yaml`（Windows: `%APPDATA%\Covenant\models.yaml` / macOS: `~/Library/Application Support/Covenant/models.yaml`）を直接編集することもできます:

```
gate3_platform: "ollama"
gate3_model: "llama3.1:8b"
gate3_host: "http://192.168.1.100:11434"   # ← ノードの IP アドレス
```

(YAML のインデントや既存の設定との重複に注意してください。不安な場合は上の Settings 画面操作を推奨します)

接続確認

Covenant を再起動 → ブラウザで `http://<ノードの IP アドレス>:11434` を開き「**Ollama is running**」と表示されるか確認、または簡単なメッセージを送って応答が返れば成功です。

> **注意**: ノードと Covenant が**同じ LAN（ネットワーク）内**にある必要があります。ファイアウォールでポート `11434` がブロックされていないか確認してください。インターネット越しの接続は本書では扱いません（セキュリティ上、LAN 内利用を推奨）。

7.2 クラウド AI の設定

こんな方に: 最新・高性能なクラウド AI (Claude / Gemini / GPT) を、機密でない処理に使いたい。

> **重要**: クラウド AI に送信された内容はプロバイダーのサーバーで処理されます。Covenant は機密度を評価して送信先を仕分けますが、**何を機密とするかはお客様のポリシーで定義します**（§3）。機密情報の送信可否はお客様の判断によります。

API キーの取得

各プロバイダーのコンソールで API キーを発行してください（発行手順は各社のサイトを参照）:

プロバイダー	API キー取得先
Anthropic (Claude)	console.anthropic.com
Google (Gemini)	aistudio.google.com
OpenAI (GPT)	platform.openai.com

API キーの設定 — Settings 画面で設定（推奨）

1. Covenant の **Settings 画面** → 「**API キー**」セクションを開く
2. 使うプロバイダーの欄にキーを貼り付け: Claude API Key (Anthropic) / Gemini API Key (Google) / OpenAI API Key
3. 入力したキーは自動的にマスキング表示されます（ `••••••••` ）。確認したいときは「👁 表示」をクリック
4. 設定は自動保存されます（内部的に機密ファイル `secrets.yaml` に保存され、第三者には共有されません）

> 入力したキーはお使いの PC 内にのみ保存されます。クラウドや当社サーバーには送信されません。
>
> 設定画面から入力したキーは OS のキーチェーン（Windows: 資格情報マネージャー / macOS: キーチェーン）に安全に保存されます。 `secrets.yaml` への直接記述は、キーチェーンが使えない環境向けの下位互換手段です（記述したキーは次回アクセス時にキーチェーンへ移行されます）。

プリセットの追加 — Settings 画面で設定（推奨）

1. Covenant の **Settings 画面** → 「**LLM プリセット**」→「**新規プリセット追加**」
2. 以下を入力: - プリセット名: `Claude Sonnet 4.6`（任意） - プラットフォーム: `claude`（または `gemini` / `openai` ） - モデル: `claude-sonnet-4-6`（使うモデル名） - ホスト: 空欄（クラウド AI はホスト不要）
3. チャット画面の LLM 選択メニューから、作成したプリセットを選べます

Web 検索を有効にする（任意）

クラウド AI に最新情報をインターネット検索させたい場合は、プリセット作成時に「Web 検索」を有効にします（既定はオフ）。Settings 画面のプリセット編集で Web 検索のオン/オフを切り替えられます。

> **補足**: Web 検索を ON にしても、AI が「自分で答えられる」と判断したクエリでは検索が走らないことがあります（これは AI 提供者側の判断による正常な動作です）。為替レート・最新ニュースなど具体的な最新情報を求めると検索が起動します。

7.3 画像ファイル（VLM）設定

画像ファイルを添付して評価したい場合は、画像内容を読み取る VLM (Vision Language Model) として `minicpm-v` を設定します。

1. モデルを取得 (未取得の場合): `ollama pull minicpm-v` (約 5.5 GB)
2. Covenant の **Settings 画面** → 「**VLM — 画像解析**」セクションで設定: - モデル: `minicpm-v` - ホスト: `http://localhost:11434` (§7.1 の別ノードを使う場合はそのノードの IP)
3. 画像を添付すると、minicpm-v が画像内容を日本語で記述し、その記述文が Gate 評価の対象になります。

> **注意**: VLM のモデルとホストの**両方**が設定されていないと、画像添付時に「VLM が設定されていません」エラーになります。テキストのみ利用する場合は本設定は不要です。
>
> **添付ファイルのサイズ上限**: 添付できるファイルは **約 25 MB** までです。これを超えるファイルや、Word / Excel など圧縮形式で**展開後に非常に大きくなる**ファイルは、「サイズ上限」エラーで読み込みを拒否します。大きい場合は分割・縮小してください。

---

8. サポート

不具合を見つけたときは、メイン画面メニューの「**Bug Report**」から報告いただけます。バージョン情報など (API キー等の機密情報は含みません) が自動で添付されるため、スムーズに対応できます (§5.5)。

それ以外のご質問・ご要望は、**cov-support@exapolicy.co.jp** までご連絡ください。内容により、確認・返信にお時間をいただく場合があります。

---

9. 次に読むもの

- **インストールガイド** / **クイックスタートガイド** — セットアップの手順
- アプリのメニューから「📖 ヘルプ」でいつでも詳しい操作方法を確認できます

利用規約・プライバシーポリシーは、初回起動時の同意画面、またはアプリの「About」からいつでも確認できます。
