

Vault構造 / 方法論

PARA Projects/Areas/Resources/Archiveの4分類。主題でなく行動可能性で振り分け、終わった案件はArchiveへ移すだけで現役が軽くなる。

1_Projects/ 2_Areas/ 3_Resources/ 4_Archive/

Zettelkasten 1ノート1アイデアの原子ノートを相互リンクで連結。フォルダ階層に頼らず、後から思わぬ結節点が生まれる前提で書く。

202606201430 単体テストの境界値.md

Johnny.Decimal 最大10エリア×10カテゴリの番号体系で物理的に置き場を固定。誰が見ても同じ場所に戻せるのでチーム共有に向く。

10-19_経営/ 11_契約/ 11.03_NDA雛形.md

MOC(Maps of Content) テーマの目次ノートを1枚作り関連ノートを[[]]で束ねる。タグ検索より見通しが良く、入口ノートとして機能する。

[[案件MOC]]に各案件ノートを列挙

デイリーノート 日付ノートに着想/ログ/タスクを即時記録し、後で恒久ノートへ昇格させる。発生源と整理先を分けて取りこぼしを防ぐ。

2026-06-20.md にInbox記録

会社ナレッジ

★生成AIガイドライン 社内のAI利用可否・入力禁止データ・モデル選定方針をノート化。AIに毎回読ませる前提で末尾に更新日と承認者を明記する。

Knowledge/生成AIガイドライン.md\nupdated: 2026-06-20

★コーディング規約 命名/ディレクトリ/Lint設定/レビュー基準を1ノートに集約。AIエージェントに参照させ、生成コードの揺れを規約側で吸収する。

[[コーディング規約]]をシステムプロンプトで参照

★セキュリティ要件 扱う情報の機密区分とAI入力可否を表で固定。社外秘・個人情報はVault外という線引きを冒頭に置く。

区分: 公開/社内/機密、機密はVault登録禁止

用語集 / 略語表 社内固有の略語・サービス名・登場人物を定義。AIが文脈を取り違えないよう、読み仮名と一文の説明を添える。

Glossary/用語集.md にKey:: 値

業務フロー / 承認経路 見積→契約→納品などの流れと承認者を手順ノートに。AIに段取りを聞く際の判断材料になり、属人化を薄める。

[[承認フロー]]from 申請 to 部長承認

パーソナライズ(人間×AI)

★AIペルソナ 口調/性格/禁止表現/出力フォーマットをノート化しシステムプロンプトとして読ませる。毎回同じ振る舞いを再現でき指示が短くなる。

AI/persona.md に「敬体・絵文字禁止」等

★人間の基礎情報 職種/得意領域/利き手ツール/苦手分野を1ノートに。AIが前提を補い、初歩説明や不要な選択肢提示を省ける。

AI/about_me.md にrole/stack/preferences

振る舞いルール 推測は明示せよ・典拠必須・断定回避など行動規範を箇条書き。AIの暴走や思い込み回答を運用ルールで抑える。

AI/rules.md 「不確実な点は質問する」

口調サンプル 実際の文章を数本貼り、語彙と語尾の癖を学習素材にする。抽象的な指示より具体例の方が文体が安定する。

AI/voice_samples.md に過去メール3本

過去の実績

★過去案件の生データ 提案/議事録/成果物の原文をそのまま保管しMOCから辿れるようにする。要約だけだと細部の根拠が消えAIの再利用精度が落ちる。

Archive/2025_A社案件/提案v3.md

★案件サマリ 1案件1ノートで目的・規模・成否・学びを定型項目で要約。frontmatterに分類を持たせ横断検索の入口にする。

client:: A社\nresult:: 受注\nlessons:: 価格交渉

再利用テンプレ 過去成果から流用可能な雛形を抽出して別フォルダに。次案件でAIに「[[見積テンプレ]]を踏襲」と指示しやすくなる。

Resources/templates/見積.md

失敗ログ トラブルと対処を時系列で残し原因を一文で固定。同じ轍を避ける判断材料になり、AIにも反面教師として渡せる。

Areas/失敗ログ.md に日付+原因+対処

他資料への接続

内部リンク[[]] 関連ノートを[[ノート名]]で直結し双方向リンクで往復可能にする。フォルダ移動に強く、知識のネットワークが育つ。[[コーディング規約#命名]]で見出し参照

埋め込み[[]] 他ノートの該当ブロックを![[]]で本文に展開し転記を避ける。一次情報を一箇所に保ち更新漏れを防ぐ。

![[用語集#略語]]

Webクリッパー 公式Web Clipperで記事をMarkdown化しsource:: にURLを残す。AIが出典付きで引用でき、後追い検証ができる。

source:: https://...\nclipped:: 2026-06-20

外部ファイル参照 PDFや図面はVault内に置きリンクし、巨大バイナリはパスだけ記す。本文から所在を辿れ、Vaultの肥大化も避ける。

[[設計図.pdf]] / path:: //server/spec

Canvas / Bases Canvasで関係を図にし、Bパースペクティブでノートを表ビュー化。一瞥と俯瞰の両方から既存資料へ接続する。

案件.canvas / clients.base

メタデータ/Properties設計

frontmatter基本 ノート冒頭の---で囲んだYAMLにkey: 値で属性を持たせる。検索/Dataview/Basesの土台になり、後から問い合わせできる。

---\ntitle: 規約\ntags: [rule]\n---

型を意識する text/list/number/checkbox/dateの型に沿って値を入れる。日付はYYYY-MM-DD固定でソートやフィルタが効く。

due: 2026-06-30\nndone: false

定番プロパティ tags/aliases/cssclassesは組み込み。別名aliasesでリンク表記の揺れを吸収し、検索ヒット率を上げる。

aliases: [規約, coding-rule]

鮮度メタ updated/owner/statusを全ノート共通で持たせAIに鮮度を判断させる。古い社内ルールを誤って引かせない歯止めになる。

updated: 2026-06-20\nstatus: active

Dataview/Bases連携 frontmatterやKey:: 値のインラインフィールドを表に集計。手作業の一覧更新を不要にし、Vaultをデータベース化する。

TABLE status FROM #案件

書き方・命名・粒度

原子ノート 1ノート1概念に絞り再利用と再結合を効かせる。盛り込みすぎたノートはリンク先として呼びづらく死蔵しやすい。

「楽観ロック」を独立ノート化

命名規則 内容が分かる名詞句で統一し、時系列が要る物だけ日付接頭辞を付ける。ファイル名がそのままリンク語になる前提で付ける。

20260620_議事_A社.md / 楽観ロック.md

見出し構造 ##見出しでブロックを区切りAIと人の両方が拾いやすくする。見出し単位の埋め込み![[#見出し]]も効くようになる。

結論 / ## 根拠 / ## 次アクション

理由を書く 事実だけでなく「なぜ残したか・どう使うか・誰向けか」を一行添える。AIが文脈を補完でき再利用の精度が上がる。

why:: 価格交渉の前例として参照

恒久ノートへ昇格 デイリーの走り書きを後で整え恒久ノートに移す。発生源と整理済み分け、Inboxを溜めっぱなしにしない。

Inbox/ → Resources/へ移動

AI連携(プラグイン/MCP)

Smart Connections ローカル埋め込みで関連ノートを意味検索し執筆中に提示。APIキー不要でノートを外に出さず、リンク候補を見つけれれる。

サイドバーに関連ノート自動表示

Copilot for Obsidian Vault内でチャットしノート横断のRAG回答を得る。Web/YouTube取り込みやエージェント操作にも対応する。

「#案件 から要点を要約」と指示

obsidian-claude-code-mcp MCPサーバ経由でClaude Code等がVaultを直接読み書き。CLIのエージェントにノート検索・更新を任せられる。

claude mcp add でvault接続

Text Generator テンプレートにプロンプトを埋め込みノート内で生成を実行。定型の下書きやメタ付与を半自動化できる。

{{title}}を基に章立て生成

Templater 新規ノートにfrontmatterや日付を自動挿入し記法を統一。AIが前提とする項目を取りこぼさず揃えられる。

<% tp.date.now() %> で日付自動

人間×AI 共同運用ルール

★機密の線引き 個人情報・社外秘・認証情報はVaultに入れずパス参照だけにする。AIに丸ごと読ませる前提だからこそ入口で遮断する。

.obsidianignore に secrets/ を指定

AIに書かせたら人がレビュー AI生成ノートはreviewed: falseで隔離し人が確認後にtrueへ。誤りが他ノートへ伝播する前に止める。

author:: ai\nreviewed:: false

AI作業ログ AIが何を更新したかを専用ノートに残し追跡可能にする。意図せぬ改変を後から辿って戻せる。

AI/changelog.md に日付+変更点

鮮度管理 updatedが古いノートを定期的に棚卸しし旧版を明示する。AIに古い規約やバージョンを引かせない。

status:: deprecated を付与

Git/バックアップ VaultをGit管理し変更履歴と復元点を確保する。AIの一括編集が暴れても直前へ戻せる安全網になる。

git commit -m 'AI整理' / 自動同期

フォルダ vs タグ vs リンク

手段	得意	使いどころ
フォルダ	排他的な一次置き場・物理的な所在	Inbox/Projects/Archive等のライフサイクル区分、誰でも同じ場所に戻せる構造
タグ	多面的な属性付与・横断抽出	#案件 #要レビュー のような状態や種別、1ノートに複数の切り口
リンク[[]]	意味的なつながり・知識の網	関連ノート連結とMOCの入口、後から結節点が育つ前提の文脈接続
Properties	型付き検索・自動集計	updated/status/clientでDataviewやBasesから表ビュー化・フィルタ

Tips・補足

機密(個人情報・社外秘・認証情報)はVaultに入れずパス参照だけにする

VaultはGit管理し、AIの一括編集も直前コミットへ戻せるようにする

全ノートにupdated/statusを持たせ、古い情報をAIに引かせない

AIに書かせたノートはreviewed: falseで隔離し、人が確認後に公開する

一次情報は1箇所に置き、他は[[]]と![[]]で参照して更新漏れを防ぐ