

# \_jarvis

Centralny mózg systemu — metodologia, wiedza, pamięć, narzędzia. Mini-Jarvis, bo od razu wiadomo o co chodzi.

## Status

Element	Wartość
Faza	1 - LITE (MVP)
Rola	Centrum operacyjne dla wszystkich projektów
Utworzono	16.01.2026
Ostatnia aktualizacja	17.01.2026

## Co to jest \_jarvis?

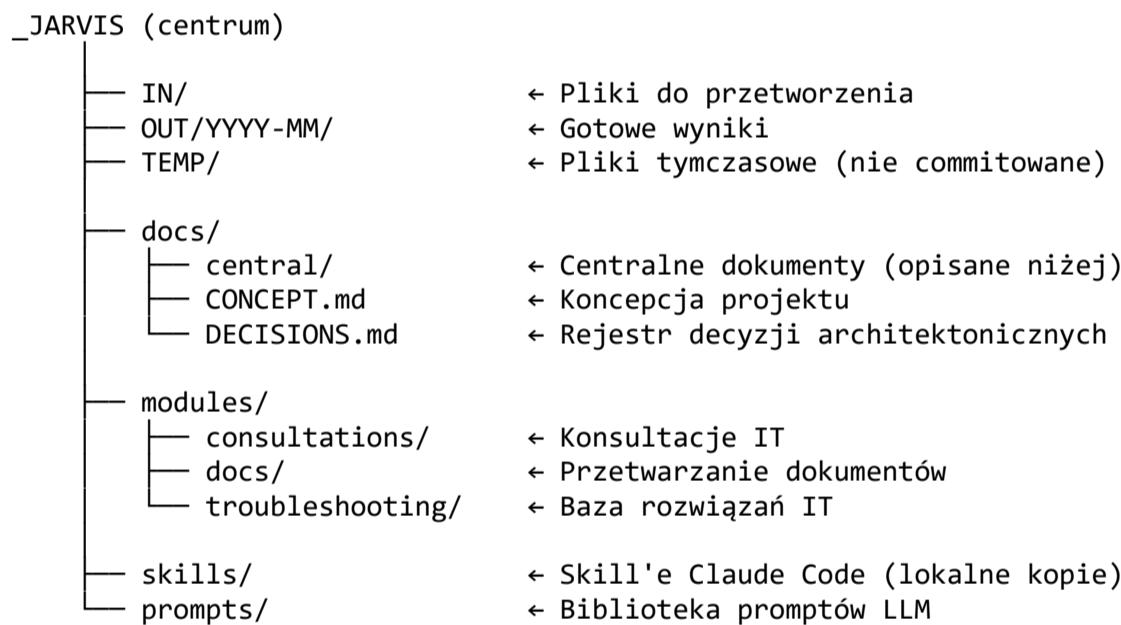
\_jarvis to **centralny asystent** integrujący: - Zarządzanie notatkami i wiedzą - Listy TODO i zadania - Konsultacje IT i troubleshooting - Przetwarzanie dokumentów (OCR, tłumaczenia, analizy) - Metodologię i dobre praktyki - Pamięć między sesjami Claude

### Problem który rozwiązuje

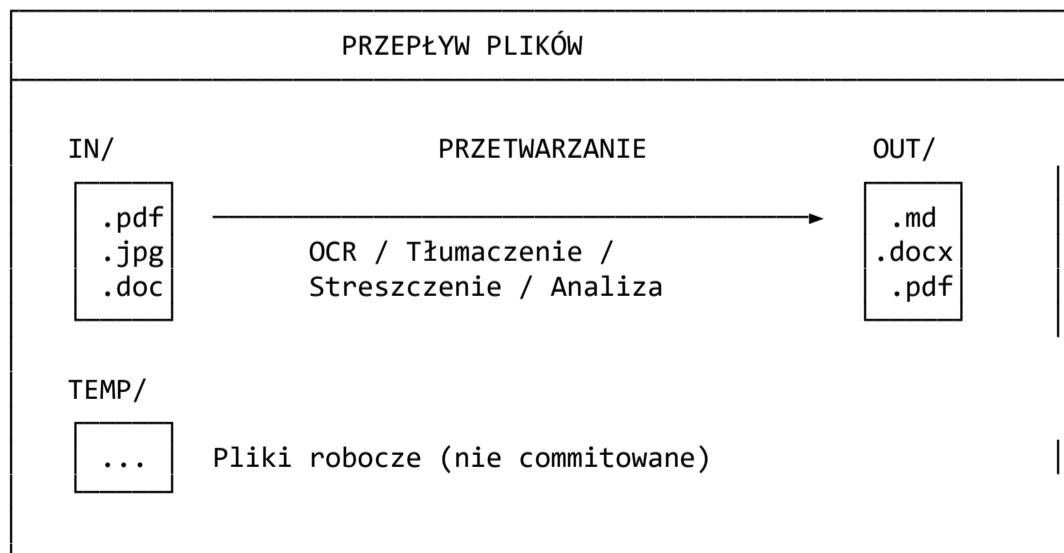
- Claude wpisuje rzeczy w złe miejsca
- Wiedza rozproszona między projektami
- Brak pamięci kontekstu między sesjami
- Powtarzające się problemy IT bez dokumentacji

## Architektura

### Diagram struktury



## Metodologia: Przepływ pracy (IN → OUT)



## Metodologia: Centralizacja wiedzy

Wszystkie projekty używają wspólnych dokumentów z tagowaniem @projekt:

```

docs/central/
  ├── CLAUDE_RULES.md      ← Zasady dla Claude (datownik, hierarchia)
  ├── GIT_STRATEGY.md      ← Format commitów, strategia push
  └── STYLE_GUIDE.md        ← Formatowanie dokumentów

  ├── TODO.md               ← Zadania (tagowane @projekt)
  ├── NOTES.md              ← Notatki (tagowane @projekt)
  ├── JOURNAL.md            ← Dziennik (tagowane @projekt)
  └── KNOWLEDGE.md          ← Baza wiedzy

  ├── MEMORY.md             ← Pamięć między sesjami Claude
  └── WEEKLY REVIEW.md      ← Przeglądy tygodniowe

  ├── WORKFLOW_LIFECYCLE.md  ← Metodologia Stage-Gate
  ├── PROJECT_REGISTRY.md    ← Status wszystkich pomysłów/projektów
  ├── PORTFOLIO.md           ← Szczegóły projektów operacyjnych
  └── RESOURCES.md          ← Zasoby i infrastruktura

  ├── METHODOLOGY.md         ← Techniki kreatywnego myślenia
  ├── SKILLS.md              ← Nabyte umiejętności
  ├── SYSTEM_MAP.md          ← Mapa powiązań systemu
  └── BEST_PRACTICES.md       ← Dobre praktyki (12 domen)

  PITFALLS/
    └── n8n.md                ← Pułapki narzędzi (n8n, API...)
    ...

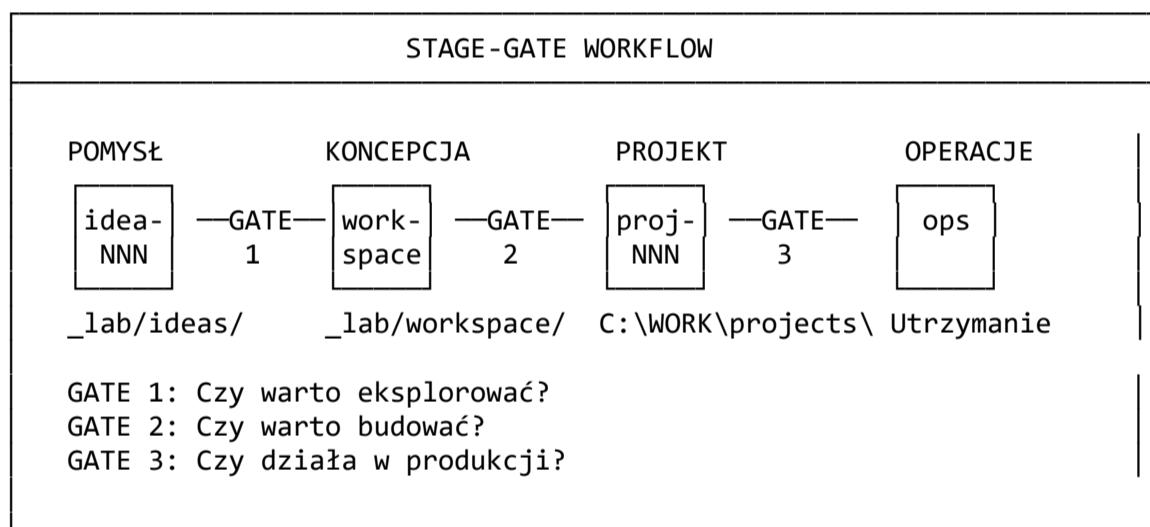
```

#### Zasada tagowania:

## 17.01.2026 – Tytuł wpisu @\_jarvis @\_meta

Treść wpisu...

#### Metodologia: Stage-Gate (WORKFLOW\_LIFECYCLE)



#### Metodologia: Moduły funkcjonalne

```

modules/
  ├── consultations/           ← KONSULTACJE IT
    ├── CONSULTATION_REGISTRY.md   ← Rejestr wszystkich konsultacji
    ├── knowledge-base/
      ├── subiekty-nexo/
      ├── wordpress/
      └── [produkt]/
    └── YYYY-MM/                 ← Konsultacje pogrupowane miesięcznie
      └── YYYY-MM-DD_[klient]-[temat]/
        ├── NOTATKI.md           ← Główne notatki
        ├── files/                ← Pliki od klienta
        ├── screenshots/          ← Zrzuty ekranu
        └── workspace/            ← Pliki robocze, raporty

  ├── docs/                     ← PRZETWARZANIE DOKUMENTÓW
    ├── templates/
      ├── PROMPT_OCR.md
      ├── PROMPT_TLUMACZENIE.md
      ├── PROMPT_STRESZCZENIE.md
      └── PROMPT_ANALIZA.md
    └── archive/                 ← Archiwum historycznych wyników
      └── ocr/YYYY-MM/

  └── troubleshooting/          ← BAZA ROZWIĄZAŃ IT
    └── templates/              ← Szablon problemu

```

windows/	← BSOD, sterowniki, uprawnienia
wordpress/	← Wtyczki, motywy, bazy danych
n8n/	← Workflow, credentials, node'y
hosting/	← cPanel, DNS, SSL, FTP, email
linux/	← Uprawnienia, procesy, cron
networking/	← DNS, firewall, VPN, proxy
databases/	← MySQL, PostgreSQL, backup
other/	← Wszystko inne

---

## Metodologia: Skills (Claude Code)

Skills to instrukcje dla Claude, które aktywują się automatycznie:

~/.claude/skills/	← Globalne (działają wszędzie)
└── jarvis-docs/	└── skill.md
└── jarvis-troubleshoot/	└── skill.md
└── jarvis/skills/	← Instrukcje przetwarzania dokumentów
└── jarvis-docs/	← Instrukcje troubleshooting IT
└── jarvis-troubleshoot/	← Lokalne kopie (source of truth)

**Aktywacja automatyczna:** | Fraza użytkownika | Aktywowany skill | |—————| |—————| | “OCR”, “rozpoznaj tekst” | jarvis-docs || “przetłumacz”, “tłumaczenie” | jarvis-docs || “streszczenie”, “podsumuj” | jarvis-docs || “problem z”, “nie działa” | jarvis-troubleshoot || “konsultacja”, “klient zgłosił” | jarvis-troubleshoot |

---

## Metodologia: Zasady YAGNI i DRY

**YAGNI (You Ain’t Gonna Need It):** - Nie tworzymy struktury “na zapas” - Folder powstaje gdy jest potrzebny - Dokument powstaje gdy ma >1 użycie

**DRY (Don’t Repeat Yourself):** - Jeden plik = jedno źródło prawdy - Wiedza z konsultacji → knowledge-base - Powtarzające się problemy → troubleshooting

---

## Metodologia: Rejestracja i śledzenie

Rejestr	Plik	Zawartość
Projekty	PROJECT_REGISTRY.md	Wszystkie pomysły i projekty
Konsultacje	CONSULTATION_REGISTRY.md	Wszystkie konsultacje IT
Decyzje	DECISIONS.md	Decyzje architektoniczne

**Statusy:** | Symbol | Znaczenie | ——— | ——— | | ● | W trakcie | | ● | Zakończone/Rozwiążane | | ● | Zablokowane/Nierozwiążane | | ○ | Przekazane/Odłożone |

---

## Skill'e (Claude Code)

\_jarvis dostarcza globalne skill'e dla Claude Code:

Skill	Aktywacja	Funkcja
<b>jarvis-docs</b>	“OCR”, “przetłumacz”, “streszczenie”	Przetwarzanie dokumentów
<b>jarvis-troubleshoot</b>	“problem z”, “nie działa”, “konsultacja”	IT troubleshooting

Skill'e są zainstalowane globalnie w ~/.claude/skills/ i działają we wszystkich projektach.

---

## Roadmapa

### Faza 1: LITE (MVP) — TERAZ

- ☒ Struktura projektu
- ☒ Centralne dokumenty (TODO, NOTES, JOURNAL, KNOWLEDGE)
- ☒ Moduł konsultacji IT (z \_tools-and-lab)
- ☒ Moduł dokumentów (z \_doc-assistant)
- ☒ Skill'e globalne
- ☒ Foldery IN/OUT/TEMP
- MCP jarvis-notes
- MCP jarvis-todo

## Faza 2: CORE

- Telegram bot jako interfejs mobilny
- Google Calendar integration
- Gmail search
- PostgreSQL dla pamięci długoterminowej

## Faza 3: FULL

- RAG na dokumentach (GDrive, Notion)
  - Vector DB (Chroma → Pinecone)
  - Voice interface (Whisper + TTS)
  - Proaktywne powiadomienia
- 

## Dokumentacja

Dokument	Opis
CLAUDE.md	Instrukcje dla Claude
docs/CONCEPT.md	Pełna koncepcja projektu
docs/central/	Centralne dokumenty

---

## Powiązania

PROJEKTY NARZĘDZIOWE (\_)  
└── \_jarvis      ← MÓZG (ten projekt)  
  └── \_meta        ← portfolio, szablony  
  └── \_lab          ← inkubator pomysłów

\_jarvis wchłonął funkcjonalności z: - \_doc-assistant → modules/docs/ - \_tools-and-lab → modules/consultations/ +  
modules/troubleshooting/

---

## Quick Start

# Skill'e są już zainstalowane globalnie  
# Claude automatycznie je aktywuje gdy:

# Dokumenty:  
"Zrób OCR tego zdjęcia"  
"Przetłumacz ten dokument"  
"Streśc ten artykuł"

# IT Troubleshooting:  
"Mam problem z WordPress"  
"Klient zgłosił błąd"  
"Otwórz nową konsultację"

---

GitHub: [SlawomirPost/\\_jarvis](#)