BudujemyAI_

E-BOOK · LEKCJA 18 · MODUŁ 5

MODUŁ 5: TWOJE CENTRUM DOWODZENIA AI

Lokalny model LLM na Twoim komputerze: Ollama, LM Studio i Qwen 3.6

Prywatne AI bez abonamentu: co uruchomisz na 8, 16 i 32 GB RAM i jak sprawdzić, czy daje radę w firmie

JEDNA PRACA, KTÓRĄ WYKONASZ

Zainstalujesz lokalny model AI i sprawdzisz go na 5 zadaniach ze swojej firmy, porównując wyniki z modelem w chmurze.



Daniel Baranowski
BudujemyAI · wdrożenia AI w małych i średnich firmach
wiedza.budujemyai.pl · wydanie: listopad 2026

W skrócie

DLA KOGO	Dla właścicieli i menedżerów firm, którzy chcą pracować z AI na poufnych danych bez wysyłania ich do chmury.
JEDNA PRACA	**Zainstalujesz lokalny model AI i sprawdzisz go na 5 zadaniach ze swojej firmy, porównując wyniki z modelem w chmurze.**
EFEKT	Masz działający model AI na własnym komputerze, kartę wyników testu i decyzję, które zadania robisz lokalnie, a które w chmurze.
CZAS	90 min
POZIOM	Średni
KOSZT	0 zł (darmowe programy i modele; komputer z min. 8 GB RAM)
NARZĘDZIA	Ollama, LM Studio, Qwen 3.6, Qwen 3.5, Gemma 4, Bielik

Jak korzystać z tego e-booka

1. **Przeczytaj rozdziały 1–3** — zrozumiesz, po co to robisz i jak to działa.
2. **Zrób „Krok po kroku" z otwartym komputerem** — prompty skopiujesz z wersji online: wiedza.budujemyai.pl/ebooki/lokalny-model-llm/.
3. **Sprawdź wynik checklistą kontroli jakości** — to najważniejszy rozdział.
4. **Wracaj do „Zestawu do powtórzenia"**, kiedy robisz to następnym razem.

Spis treści

01 Zanim zaczniesz

02 Dlaczego to ważne dla Twojej firmy

03 Jak to działa — w prostych słowach

04 Jaki model na jaki komputer

05 Krok po kroku

06 Jak rozpoznać, że AI zrobiło dobrą robotę

07 Bezpieczeństwo i RODO

08 Najczęstsze błędy

09 Zestaw do powtórzenia

10 Twój plan działania

11 Słowniczek

12 Źródła

13 Co dalej

ROZDZIAŁ 1

Zanim zaczniesz

Ta lekcja jest dla Ciebie, jeśli chcesz używać AI przy danych, których nie chcesz wysyłać do chmury — umowach, danych klientów, wycenach z marżami — albo chcesz sprawdzić, czy Twój komputer udźwignie model AI bez internetu. Nie musisz być informatykiem: każde polecenie dostajesz gotowe do skopiowania.

Twoja jedna praca: zainstalujesz lokalny model AI i sprawdzisz go na 5 zadaniach ze swojej firmy, porównując wyniki z modelem w chmurze. Na koniec masz kartę wyników i decyzję, które zadania robisz lokalnie, a które w chmurze.

Czego potrzebujesz:

- komputera z Windows 11 (Windows 10 22H2 tylko z aktywnymi aktualizacjami bezpieczeństwa ESU — zwykłe wsparcie skończyło się 14.10.2025), z macOS 14 lub nowszym na procesorze Apple M albo z Linuxem;
- min. 8 GB RAM — wtedy zamknij przeglądarkę i inne programy (wygodnie 16 GB); do Qwen 3.6 — 32 GB RAM i karty graficznej z 16 GB pamięci albo Maca z 32 GB;
- ok. 4 GB miejsca na program i 3–25 GB na każdy model;
- internetu tylko do pobrania — potem model działa offline;
- konta w modelu w chmurze (Claude, ChatGPT lub Gemini) do porównania — darmowe wystarczy, jeśli wysyłasz do niego tylko tekst zanonimizowany albo wymyślony na ćwiczenie (Krok 7);
- 5 prawdziwych zadań z Twojej firmy (Krok 7).

Koszt: 0 zł. Ollama jest darmowa (licencja MIT), LM Studio od lipca 2025 jest darmowe także w pracy, a modele z tej lekcji mają licencje pozwalające na użycie w firmie. Płacisz sprzętem, który już masz, i prądem. **Czas:** ok. 90 minut, sporo z tego to pobieranie. **Poziom:** Średni.

ROZDZIAŁ 2

Dlaczego to ważne dla Twojej firmy

Znasz ten moment: masz przed sobą umowę, listę płac albo arkusz z marżami i myślisz: „AI zrobiłoby to w minutę... ale czy mogę to wkleić do czatu?". Ten opór jest zdrowy, bo w chmurze dane trafiają na serwery dostawcy. Da się to zrobić porządnie — firmowym kontem z umową powierzenia (o tym jest lekcja o Gemini w Workspace i RODO). Druga droga to model, który działa w całości na Twoim komputerze.

Wyobraź sobie biuro rachunkowe, w którym jedna osoba odpowiada dziennie na kilkadziesiąt maili od klientów — z nazwiskami, kwotami, czasem numerami PESEL. Lokalny model może te maile posortować według tematu, przygotować szkice odpowiedzi według wzoru i wyciągnąć z dokumentów kwoty i terminy do tabeli. Nic nie opuszcza komputera, a pracownica zamiast pisać od zera — sprawdza.

Zyskujesz czas przy powtarzalnych tekstach, pracę offline i brak limitów czy opłat za tokeny — oraz konkretną odpowiedź, gdy klient zapyta: „co robicie z moimi danymi?".

Uczciwie: lokalny model to nie jest darmowy Claude czy ChatGPT na laptopie. Najlepsze modele w chmurze są dużo większe i zwykle lepsze w trudnych zadaniach. Lokalny to sprawny asystent do powtarzalnej pracy na tekście — dlatego tak ważny jest test z Kroku 7. Jeśli używasz Handy z lekcji o pisaniu głosem, znasz już tę ideę: AI działa u Ciebie i nie wysyła niczego w świat.

ROZDZIAŁ 3

Jak to działa — w prostych słowach



Płyta i odtwarzacz. Lokalny model to plik z „wiedzą" (miliardy liczb zwanych parametrami) i program, który go uruchamia — jak plik MP3 i odtwarzacz. Ollama i LM Studio to odtwarzacze, a Qwen, Gemma i Bielik — płyty. „27B" oznacza 27 miliardów parametrów: im więcej, tym model zwykle mądrzejszy, ale większy i wolniejszy.

Kwantyzacja, czyli kompresja jak JPG. Oryginał zapisuje każdą liczbę na 16 bitach, wersja Q4_K_M — średnio na ok. 4,9 bita, a Q3_K_M — na ok. 4 bitach (ważniejsze części modelu dostają więcej bitów, niż wskazuje nazwa). Plik jest kilka razy mniejszy, a jakość spada nieznacznie. Przy mocnej kompresji (Q2) model częściej się myli — jak zdjęcie z artefaktami. Gotowe pliki mają zwykle format GGUF.

Blat i magazyn. Najszybciej model pracuje w pamięci karty graficznej (VRAM), wolniej w zwykłym RAM. VRAM to blat przy maszynie, RAM — magazyn obok. Gdy wszystko leży na blacie, praca idzie szybko; gdy część trzeba nosić z magazynu, każde słowo trwa dłużej. Przeniesienie części modelu do RAM to **offload** — działa, ale spowalnia. Na Macu z procesorem M pamięć jest wspólna, więc liczy się całe RAM minus to, czego potrzebuje system.

Tokeny na sekundę. Model pisze kawałkami tekstu — tokenami (słowo to jeden lub kilka tokenów), a szybkość mierzy się w tokenach na sekundę.

Model gęsty i MoE. W Qwen3.6-27B przy każdym słowie pracują wszystkie 27 mld parametrów — to model gęsty. Qwen3.6-35B-A3B to MoE (Mixture of Experts, „mieszanka ekspertów"): 35 mld parametrów, ale przy każdym słowie pracuje ok. 3 mld — 8 z 256 ekspertów i jeden wspólny. Jak w kancelarii: do sprawy idzie kilku specjalistów, reszta czeka obok. Dlatego MoE dobrze znosi offload — ekspertów trzymasz w RAM, a każde słowo potrzebuje tylko kilku z nich.

Kontekst, czyli pamięć robocza. To wszystko, co model widzi naraz: polecenie, dokument, rozmowa i jego odpowiedź (razem z „myśleniem"). Qwen 3.6 obsługuje do 262 144 tokenów, ale każdy token zajmuje pamięć. Ollama przy karcie poniżej 24 GB ustawia domyślnie 4 tys. tokenów, czyli kilka stron. Dłuższy dokument model zobaczy tylko częściowo — i zwykle Ci o tym nie powie.

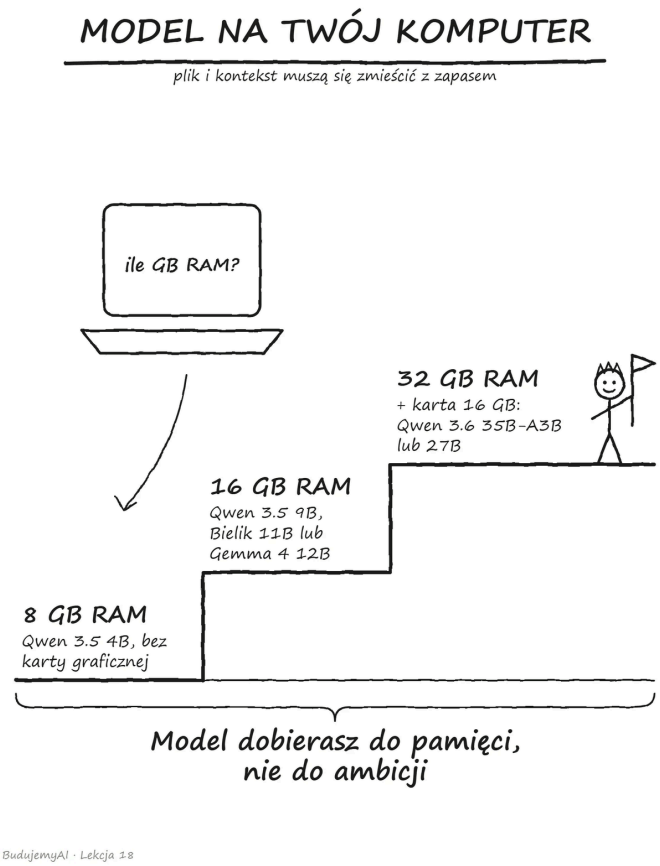
Tryb myślenia. Qwen 3.6 domyślnie „myśli na głos" przed odpowiedzią: przy trudnych zadaniach to pomaga, przy prostych wydłuża czekanie. Wyłączysz go w Kroku 3.

Kiedy lokalnie, a kiedy chmura

Sytuacja	Model lokalny	Model w chmurze
Dane osobowe, poufne, tajemnica firmy	Tak — dane zostają u Ciebie	Tylko konto firmowe z umową powierzenia
Powtarzalna praca na tekście	Tak	Tak, ale z limitami i kosztami
Trudna analiza, strategia, długie zadania	Raczej nie	Tak
Aktualne przepisy, ceny, wiadomości	Nie — brak internetu	Tak, z wyszukiwaniem
Praca bez internetu	Tak	Nie

ROZDZIAŁ 4

Jaki model na jaki komputer



Model dobierasz do pamięci, nie do ambicji: plik i kontekst muszą się zmieścić z zapasem na system i przeglądarkę. Dla karty graficznej wydawcy plików GGUF podają regułę: wybierz plik o 1–2 GB mniejszy niż pamięć karty — wtedy cały model pracuje na karcie.

Twój komputer	Model	Pobranie	Polecenie w Ollamie
8 GB RAM, bez karty	Qwen 3.5 4B	3,4 GB	`ollama run qwen3.5:4b`
16 GB RAM	Qwen 3.5 9B	6,6 GB	`ollama run qwen3.5:9b`
16 GB RAM	Bielik 11B v3.0 (polski)	6,7 GB	`ollama run SpeakLeash/bielik-11b-v3.0-instruct:Q4_K_M`
16 GB RAM	Gemma 4 12B	ok. 8 GB	`ollama run gemma4:12b`
32 GB RAM + karta 16 GB	Qwen 3.6 35B-A3B	ok. 24 GB	`ollama run qwen3.6:35b-a3b`
32 GB RAM + karta 16 GB albo Mac z 32 GB	Qwen 3.6 27B	ok. 18–19 GB	`ollama run qwen3.6:27b`

Rozmiary z biblioteki Ollamy (stan: październik 2026) mogą się zmienić — przed pobraniem sprawdź stronę modelu. Na Macu z 32 GB model 27B powinien się zmieścić, jeśli zamkniesz zbędne programy. Bez karty graficznej modele z wierszy 8 i 16 GB liczą na procesorze: działają, ale wolniej — ile, zmierzysz w Kroku 7.

Małe modele: do czego wystarczą

Modele 4–12B dobrze radzą sobie ze streszczeniem maila, sortowaniem wiadomości, poprawą stylu krótkiego tekstu, wyciąganiem danych do tabeli i szkicami odpowiedzi według wzoru. Gorzej z wiedzą o świecie i przepisach (częściej się mylą i nie mają internetu), długimi dokumentami, wieloetapowym rozumowaniem i rachunkami.

Polszczyzna: Qwen 3.5 i 3.6 deklarują 201 języków i dialektów, Gemma 4 — ponad 35 języków „od ręki" (trening na ponad 140), a Bielik od SpeakLeash i ACK Cyfronet AGH jest zoptymalizowany pod polski. Deklaracje sprawdzisz testem polszczyzny z Kroku 7.

Qwen 3.6 27B na 32 GB RAM i karcie 16 GB — co realnie

Qwen3.6-27B to model Alibaby z 22 kwietnia 2026: gęsty, 27 mld parametrów, kontekst do 262 144 tokenów, rozumie tekst, obrazy i wideo (w Ollamie i LM Studio — tekst i obrazy), licencja Apache 2.0. Producent deklaruje m.in. 77,2 pkt w programistycznym teście SWE-bench Verified i 86,2 w teście wiedzy MMLU-Pro — to jego wyniki, nie test na Twoich zadaniach.

Plik Q4_K_M waży ok. 17 GB (16,5–18 GB zależnie od wydawcy; tag Ollamy `qwen3.6:27b` ma ok. 18–19 GB, bo zawiera też moduł do obrazów), Q3_K_M — ok. 14 GB (13,6–14,8 GB). Według Unsloth wersja 4-bitowa potrzebuje ok. 18 GB pamięci, a 3-bitowa ok. 15 GB. Q4 nie zmieści się więc w całości na karcie 16 GB, Q3 — tak, ale z małym zapasem na kontekst. Masz trzy warianty:

Wariant	Jak	Plus	Minus
A. 27B Q3_K_M cały na karcie	LM Studio, plik ok. 14 GB	najszybszy wariant 27B	mało miejsca na kontekst, nieco niższa jakość
B. 27B Q4_K_M z offloadem	Ollama `qwen3.6:27b` lub LM Studio	lepsza jakość kompresji	wyraźnie wolniej (gęsty model ciągle sięga do RAM)
C. 35B-A3B, eksperci w RAM	Ollama `qwen3.6:35b-a3b` lub LM Studio	zwykle najszybszy	w testach producenta minimalnie słabszy

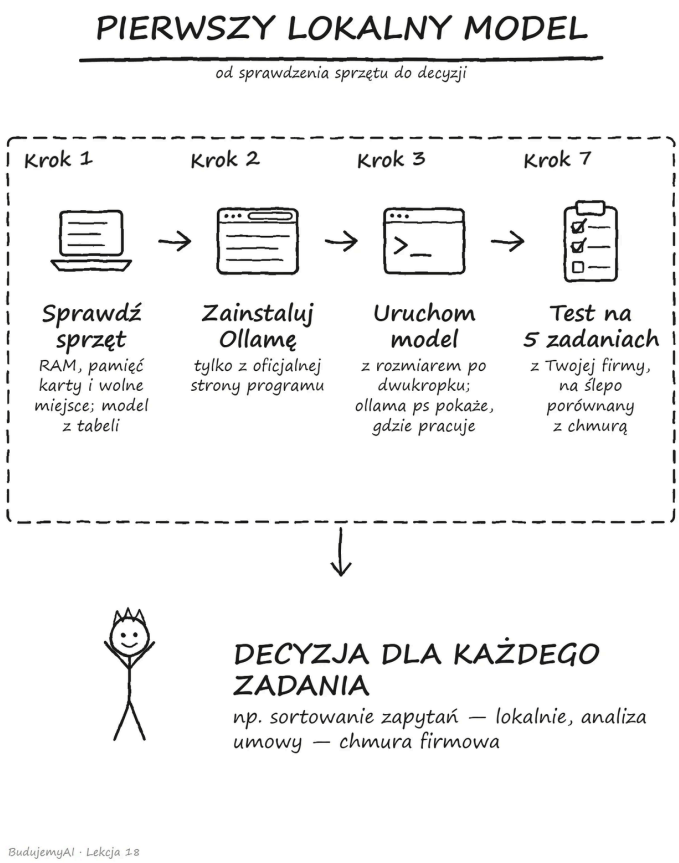
Pomiary: w dyskusji w repozytorium llama.cpp Qwen3.6-35B-A3B (Q4_K_M) na karcie Radeon RX 7800 XT 16 GB z częścią ekspertów w RAM generował ok. 35 tokenów/s. Na RTX 3090 (24 GB) ten sam model w całości na karcie robił ok. 140 tokenów/s, a z częścią ekspertów w RAM — ok. 89 (pomiar Gilesa Thomasa). W testach producenta 35B-A3B jest minimalnie słabszy od 27B (SWE-bench Verified 73,4 wobec 77,2).

Dla wariantu B nie ma wiarygodnego pomiaru na pojedynczej karcie 16 GB, więc nie podaję liczby z sufitu — zmierzysz to w Kroku 7. Rada: zacznij od wariantu C i porównaj go z A na swoich zadaniach. 32 GB RAM wystarcza w każdym wariancie, jeśli zamkniesz zbędne programy.

> **i Stan na październik 2026: jest już Qwen 3.8**
>
> W sierpniu 2026 Alibaba wydała Qwen3.8-27B — następcę o tym samym rozmiarze i budowie (w Ollamie `qwen3.8:27b`, ok. 18 GB). Wszystko o pamięci i wariantach A i B dotyczy go tak samo, a producent deklaruje lepsze wyniki w programowaniu. Dodaj go do testu z Kroku 7.

ROZDZIAŁ 5

Krok po kroku



Krok 1: Sprawdź sprzęt i wybierz model

Na Windows: Ustawienia → System → Informacje („Zainstalowana pamięć RAM"). Pamięć karty: Menedżer zadań (Ctrl+Shift+Esc) → Wydajność → GPU — wybierz kartę NVIDIA lub AMD (laptop często ma dwie: GPU 0 i GPU 1) i odczytaj pamięć dedykowaną, nie współdzieloną. Wolne miejsce: Eksplorator plików → Ten komputer. Na Macu: menu Apple → Ten Mac. Na Linuksie: `free -h` (RAM), `nvidia-smi` (pamięć karty NVIDIA) i `df -h ~` (wolne miejsce).

Zapisz trzy liczby i wybierz wiersz z tabeli powyżej. Jeśli masz kartę NVIDIA, zaktualizuj sterownik — Ollama wymaga wersji 551.61 lub nowszej.

Krok 2: Zainstaluj Ollamę

Wejdź na ollama.com/download i pobierz instalator (Windows: OllamaSetup.exe, Mac: Ollama.dmg). Na Windows nie potrzebujesz uprawnień administratora. Możesz też użyć polecenia — w PowerShellu:

```powershell
irm https://ollama.com/install.ps1 | iex
```

Na macOS i Linuksie, w Terminalu:

```
TERMINAL
curl -fsSL https://ollama.com/install.sh | sh
```

Takie polecenia wklejaj wyłącznie z oficjalnej strony programu. Potem otwórz nowe okno terminala i sprawdź:

```
TERMINAL
ollama --version
```

Napis `ollama version is …` oznacza, że Ollama działa w tle. Ostrzeżenie `could not connect to a running Ollama instance` znaczy, że nie działa — uruchom aplikację Ollama i sprawdź ponownie. Modele trafiają do folderu `.ollama` w katalogu użytkownika. Gdy brakuje miejsca na dysku C, w aplikacji Ollama otwórz Settings → **Model location** i wskaż folder na innym dysku (albo ustaw zmienną środowiskową `OLLAMA_MODELS` i uruchom Ollamę ponownie).

> ✓ **Gdy terminal nie zna polecenia** `ollama`
>
> Zamknij okno terminala i otwórz je ponownie — nowe programy są widoczne dopiero w nowym oknie.

Krok 3: Uruchom pierwszy model i sprawdź, gdzie pracuje

Rozgrzewka małym modelem:

```
TERMINAL
ollama run qwen3.5:4b
```

Ollama pobierze ok. 3,4 GB i otworzy czat w terminalu. Napisz: „Przedstaw się w dwóch zdaniach". Możesz też rozmawiać w oknie aplikacji Ollama: wybierasz model z listy, a pliki tekstowe i PDF przeciągasz do okna. W czacie w terminalu: `"""` zaczyna i kończy tekst w kilku liniach, `/set nothink` wyłącza tryb myślenia (`/set think` włącza), `/bye` kończy rozmowę.

Teraz model dla Twojego sprzętu, np.:

```
TERMINAL
ollama run qwen3.6:35b-a3b
```

> ! **Zawsze podawaj rozmiar po dwukropku**
>
> Sama nazwa `qwen3.6` pobierze wariant domyślny 35B — ok. 24 GB, za dużo dla laptopa z 16 GB RAM.

W drugim oknie terminala wpisz:

```
TERMINAL
ollama ps
```

Kolumna PROCESSOR mówi, gdzie pracuje model: „100% GPU" — cały na karcie, „100% CPU" — tylko procesor, „48%/52% CPU/GPU" — podział. CONTEXT to długość kontekstu, z jaką model został załadowany — limit, a nie to, ile zużyła już rozmowa. Pozostałe polecenia są w ściądze na końcu lekcji.

Krok 4: Ustaw długość kontekstu

Domyślne 4 tys. tokenów wystarczą do maili, ale nie do 15-stronicowej umowy. Na stałe: w aplikacji Ollama przesuń w ustawieniach (Settings) suwak **Context length**, np. na 16k lub 32k — działa też dla czatu w terminalu. Jednorazowo, tylko do końca bieżącej rozmowy w terminalu:

```
TEKST

/set parameter num_ctx 16384
```

Większy kontekst zajmuje więcej pamięci i może wypchnąć część modelu z karty do RAM. Po zmianie wyślij modelowi dowolną wiadomość (wtedy przeładuje się z nowym kontekstem) i w drugim oknie sprawdź `ollama ps`: CONTEXT pokaże nową wartość, a jeśli z „100% GPU" zrobił się podział CPU/GPU, kontekst jest za duży na Twoją kartę. Ustaw tyle, ile potrzebują Twoje dokumenty, nie „na zapas".

Krok 5: Zainstaluj LM Studio, gdy chcesz pełnej kontroli

LM Studio, w przeciwieństwie do Ollamy, pozwala wybrać konkretny plik kompresji i ustawić, ile modelu idzie na kartę — to narzędzie do wariantu A.

1. Pobierz program z lmstudio.ai. Wymagania: Windows z procesorem obsługującym AVX2 (zalecane 16 GB RAM i 4 GB pamięci karty) albo Mac z procesorem M i macOS 14 lub nowszym.
2. W zakładce Discover (Ctrl+2, na Macu ⌘+2) wyszukaj model — możesz wpisać nazwę repozytorium, np. `bartowski/Qwen_Qwen3.6-27B-GGUF` lub `unsloth/Qwen3.6-27B-GGUF`.
3. Wybierz plik: dla 27B na karcie 16 GB — Q3_K_M. Przy mniejszych modelach dokumentacja LM Studio radzi 4 bity lub więcej.
4. W zakładce Chat otwórz wybór modelu i przed załadowaniem ustaw **GPU Offload**, czyli ile warstw idzie na kartę (w wariancie A — maksimum, wszystkie), oraz **Context Length** (na start ok. 8 tys.; gdy model nie mieści się na karcie, zmniejsz).
5. Przy modelu MoE (wariant C) włącz w zaawansowanych ustawieniach ładowania **Force Model Expert Weights onto CPU** — eksperci idą do RAM, reszta zostaje na karcie. LM Studio poleca to przy małej ilości VRAM.
6. Pod odpowiedzią widzisz statystyki, m.in. tokeny na sekundę. Do czatu dołączysz pliki .pdf, .docx i .txt — długie program przeszukuje i podaje modelowi pasujące fragmenty (RAG).

Po pracy wyładuj model. Według polityki prywatności LM Studio przy modelach lokalnych wiadomości i dokumenty nie opuszczają komputera.

Krok 6: Podłącz inne programy przez lokalne API

Oba programy udostępniają lokalne API zgodne z OpenAI: inne aplikacje rozmawiają z Twoim modelem jak z ChatGPT, tylko pod adresem na Twoim komputerze. Czym jest API, wyjaśnia lekcja o CLI, MCP i API.

- **Ollama:** `http://localhost:11434/v1`, gdy Ollama działa.
- **LM Studio:** zakładka Developer → przełącznik Start server (lub `lms server start`), adres `http://localhost:1234/v1`.

W programie z opcją „OpenAI-compatible" podajesz adres bazowy (Base URL), nazwę modelu (w Ollamie tag, np. `qwen3.5:9b`; w LM Studio identyfikator z listy pod `http://localhost:1234/v1/models`) i jako klucz API dowolny tekst, np. `local` — pole bywa wymagane przez program, ale Ollama go nie sprawdza. Nie wpisuj tam prawdziwego klucza OpenAI ani Claude. Test: otwórz w przeglądarce `http://localhost:11434/v1/models` — surowy tekst (JSON) z nazwami Twoich modeli oznacza, że API działa.

Dane zostają u Ciebie tylko wtedy, gdy podłączony program też działa na Twoim komputerze i sam nie wysyła ich w świat — sprawdź to, zanim podłączysz pocztę albo dysk sieciowy.

Krok 7: Przetestuj model na 5 zadaniach z firmy

To sedno lekcji. Wybierz 5 zadań, które naprawdę robisz co tydzień, np.:

1. odpowiedź na maila z reklamacją,
2. streszczenie 3–5-stronicowego dokumentu,
3. kategorie dla 10 zapytań z formularza,
4. dane z faktury lub umowy do tabeli (Qwen 3.5, Qwen 3.6 i Gemma 4 rozumieją też obraz — w aplikacji Ollama dołączasz skan),
5. korekta tekstu oferty.

Uruchom model z pomiarem czasu — z nazwą ze swojego wiersza tabeli, np. przy 16 GB RAM:

```
TERMINAL

ollama run qwen3.5:9b --verbose
```

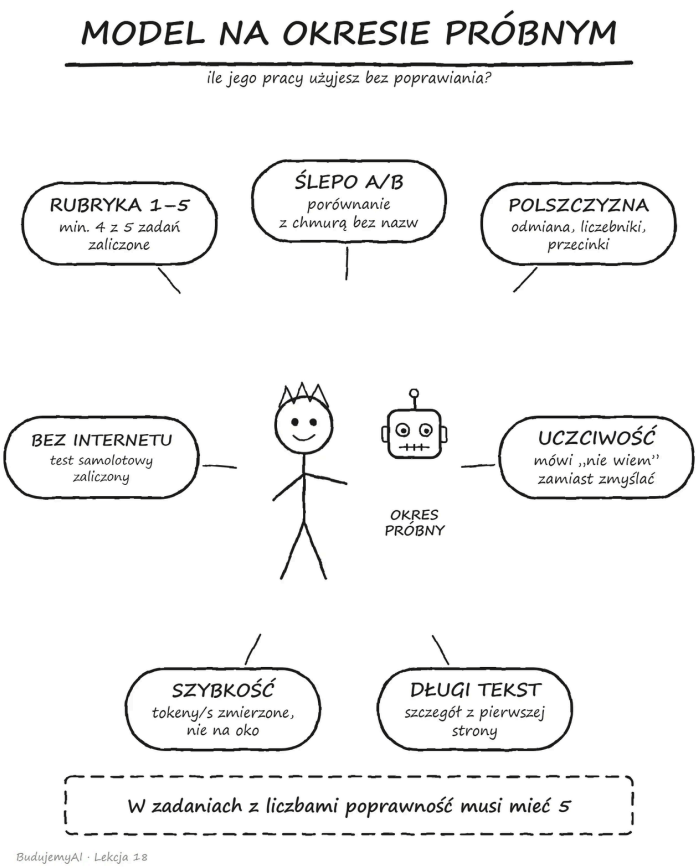
Po każdej odpowiedzi zobaczysz **eval rate** (tokeny na sekundę przy pisaniu), **prompt eval rate** (tempo wczytywania tekstu) i **total duration** (całkowity czas). W LM Studio te liczby są pod odpowiedzią. Do prostych zadań wyłącz myślenie (`/set nothink`). Do maila kopiujesz tylko właściwą odpowiedź, bez „myślenia", które terminal pokazuje przed nią.

To samo zadanie daj modelowi w chmurze, ale po anonimizacji: imiona i nazwiska, adresy, PESEL, NIP, numery rachunków, telefony, e-maile i nazwy klientów zamień na znaczniki typu [KLIENT] czy [ADRES], a szczegóły, po których da się rozpoznać osobę lub sprawę, usuń. Bez firmowego konta z umową powierzenia do chmury trafia tylko taki tekst albo przykład wymyślony na ćwiczenie.

Obie odpowiedzi oceń rubryką z następnej sekcji i wpisz do karty wyników. Decyzję podejmujesz osobno dla każdego zadania: „sortowanie zapytań — lokalnie", „analiza umowy — chmura firmowa".

ROZDZIAŁ 6

Jak rozpoznać, że AI zrobiło dobrą robotę



Traktuj lokalny model jak nowego pracownika na okresie próbnym. Nie oceniasz, jak pewnie brzmi, tylko ile jego pracy da się użyć bez poprawiania.

Test 1: rubryka na Twoich zadaniach

Każdą odpowiedź oceń w skali 1–5:

Kryterium	Co sprawdzasz	5 punktów, gdy
Poprawność	liczby, nazwy, daty zgodne ze źródłem	zero błędów
Kompletność	nic ważnego nie zostało pominięte	wszystko jest
Polszczyzna	odmiana, interpunkcja, ton	jak po dobrym redaktorze
Format	tabela, długość, styl zgodne z poleceniem	dokładnie
Gotowość	ile poprawek przed użyciem	wysyłasz od razu

Zadanie jest zaliczone, gdy żadne kryterium nie spadło poniżej 3. W zadaniach z liczbami (kwoty, terminy, numery kont) poprawność musi mieć 5 — jedna zła kwota w mailu do klienta to nie „drobna poprawka". Model zdaje egzamin, gdy zaliczy min. 4 z 5 zadań.

Test 2: ślepe porównanie z chmurą

Oznacz odpowiedzi literami A i B, wymieszaj, oceń — i dopiero potem sprawdź, który model był który. Tak nie faworyzujesz ani „droższego", ani „swojego". Możesz poprosić trzeci model o ocenę według tej samej rubryki (prompt 6), ale decyzja należy do Ciebie — o pułapkach oceniania przez AI przeczytasz w lekcji o słabościach AI. W trudnych zadaniach chmura zwykle wygra, więc pytanie brzmi nie „który lepszy?", tylko „czy lokalny jest wystarczająco dobry do tego zadania?".

Test 3: polszczyzna

Użyj promptu 7 (mail do Agnieszki Wróbel z Bielska-Białej). Sprawdź odmianę imion i miejscowości („Szanowna Pani Agnieszko", „z Bielska-Białej"), liczebniki („5 faktur", ale „22 zlecenia"), przecinki przed „że" i „który", kalki z angielskiego oraz wtrącenia w obcym języku — angielskie słowa, obce znaki, zmianę języka w połowie zdania.

Test 4: fakty i uczciwość

Lokalny model nie ma internetu — wie tyle, ile wiedział w dniu zakończenia treningu. Przepisy, stawki i terminy (np. KSeF) zawsze sprawdzaj w oficjalnym źródle. Zrób test uczciwości: zapytaj o coś, czego model nie może wiedzieć, np. „Ile kosztuje nasza usługa X?". Dobry model odpowie, że nie wie. Słaby wymyśli kwotę. Więcej w lekcji o halucynacjach.

Test 5: długi dokument

Wklej do czatu tekst dokumentu dłuższego niż kilka stron i zapytaj o szczegół z pierwszej strony (numer umowy, datę, nazwisko). W LM Studio nie dołączaj pliku — długi plik model dostaje tylko we fragmentach (RAG, Krok 5), więc test nie sprawdziłby kontekstu. Jeśli model nie wie albo zmyśla, kontekst jest za krótki — wróć do Kroku 4 albo dziel dokument na części.

Test 6: szybkość i prywatność

Zapisz eval rate i całkowity czas każdego zadania, a kryterium ustal z perspektywy pracy, np. „szkic odpowiedzi na maila w mniej niż minutę". Przy trybie myślenia liczy się czas do gotowej odpowiedzi — model potrafi szybko „myśleć" bardzo długo.

Test samolotowy: odłącz internet — Wi-Fi i kabel sieciowy — i zadaj pytanie. Model odpowiada — pracuje lokalnie. Sprawdź też, czy nazwa modelu nie kończy się na `:cloud` — takie modele w Ollamie działają na serwerach Ollamy, nie u Ciebie.

Sygnały ostrzegawcze: liczby albo „cytaty", których nie ma w dokumencie; odpowiedź urwana w połowie; powtarzanie w kółko tych samych zdań; nagła zmiana języka; duży udział CPU w `ollama ps` przy modelu gęstym (rozważ wariant A lub C).

✓ Kontrola jakości lokalnego modelu

- [ ] 5 zadań z firmy przetestowanych, wyniki w karcie
- [ ] Ślepa ocena A/B z modelem w chmurze według tej samej rubryki
- [ ] Min. 4 z 5 zadań zaliczone, w zadaniach z liczbami zero błędów
- [ ] Test polszczyzny: odmiana, liczebniki, interpunkcja, brak obcych wtrąceń
- [ ] Model przyznaje się do niewiedzy zamiast zmyślać
- [ ] Test długiego dokumentu zaliczony
- [ ] Szybkość zmierzona i akceptowalna dla danego zadania
- [ ] Test samolotowy zaliczony, żaden model nie ma końcówki `:cloud`
- [ ] Decyzja zapisana: które zadania lokalnie, które w chmurze

ROZDZIAŁ 7

Bezpieczeństwo i RODO

Według dokumentacji Ollama nie widzi Twoich promptów ani danych, gdy model działa lokalnie. Żeby tak zostało:

- **Wyłącz chmurę Ollamy, jeśli z niej nie korzystasz:** w aplikacji Ollama → Settings wyłącz przełącznik **Cloud**. Bez aplikacji: plik `server.json` w folderze `.ollama` w katalogu użytkownika z treścią `{ "disable_ollama_cloud": true }` albo zmienna `OLLAMA_NO_CLOUD=1` (w PowerShellu `setx OLLAMA_NO_CLOUD 1`) — potem zamknij Ollamę (ikona przy zegarze lub na pasku menu → Quit Ollama) i uruchom ją ponownie.
- **Nie wystawiaj serwera do sieci.** Ollama domyślnie słucha tylko na Twoim komputerze (127.0.0.1). Nie zmieniaj `OLLAMA_HOST`, nie włączaj w Ollamie **Expose Ollama to the network**, a w LM Studio „Serve on Local Network". Taki serwer nie pyta o hasło (Ollama nie ma go wcale, LM Studio domyślnie), więc z modelu mógłby korzystać każdy w tej samej sieci, np. w biurze czy hotelu.
- **Historia czatów leży na dysku** — zaszyfruj dysk (Windows Pro: BitLocker; Windows Home: Ustawienia → Prywatność i zabezpieczenia → Szyfrowanie urządzenia; Mac: Ustawienia systemowe → Prywatność i ochrona → FileVault), ustaw hasło do komputera, usuwaj zbędne rozmowy z listy czatów, a eksportów nie zapisuj w folderach synchronizowanych z chmurą (OneDrive, Dysk Google, iCloud).
- **Modele z zaufanych źródeł:** biblioteka Ollamy, konta producentów (Qwen, Google, SpeakLeash), znani wydawcy GGUF (lmstudio-community, unsloth, bartowski).
- **Licencja:** Qwen 3.6, Bielik 11B v3.0 i Gemma 4 — Apache 2.0; inne modele mogą mieć własne warunki.

RODO: lokalny model nie zwalnia Cię z obowiązków administratora danych (podstawa przetwarzania, minimalizacja, zabezpieczenia), ale nie przekazujesz danych dostawcy AI, więc w tym miejscu nie potrzebujesz z nim umowy powierzenia. To ogólna informacja, nie porada prawna — w konkretnej sprawie zapytaj prawnika lub inspektora ochrony danych.

ROZDZIAŁ 8

Najczęstsze błędy

1. **Nazwa modelu bez rozmiaru** → `ollama run qwen3.6` pobiera ok. 24 GB, które na laptopie z 16 GB RAM nie ruszą → zawsze podawaj rozmiar po dwukropku.
2. **Model większy niż pamięć** → odpowiedzi ciągną się minutami → dobieraj model z tabeli i sprawdzaj `ollama ps`.
3. **Domyślny kontekst przy długich dokumentach** → model widzi część tekstu, a resztę dopowiada → ustaw kontekst (Krok 4) i zrób test długiego dokumentu.
4. **Ocena po jednym pytaniu z wiedzy ogólnej** → „Kim był Kopernik?" nic nie mówi o Twoich mailach → test na 5 prawdziwych zadaniach z rubryką.
5. **Lokalny model jako źródło wiedzy o przepisach** → nieaktualne albo zmyślone informacje → przepisy i liczby bierz z oficjalnych źródeł.
6. **Dane wypływają, choć miały zostać** → przypadkiem wybrany model `:cloud` albo serwer udostępniony w sieci → `OLLAMA_NO_CLOUD`, domyślne ustawienia sieci, test samolotowy.
7. **Modele od przypadkowych autorów** → nieznana jakość i niejasna licencja → oficjalne biblioteki i znani wydawcy.
8. **Zapomniany model w pamięci** → komputer zwalnia (Ollama trzyma model 5 minut po ostatnim użyciu, LM Studio zwykle do wyładowania) → `ollama stop nazwa-modelu` albo `lms unload --all`.

ROZDZIAŁ 9

Zestaw do powtórzenia

W terminalu dłuższy prompt wklejasz między `"""` a `"""`. W nawiasach kwadratowych uzupełnij swoje dane.

Prompt 1: asystent biura

› PROMPT

```
Jesteś asystentem biura w polskiej firmie [branża]. Piszesz poprawną, prostą polszczyzną, uprzejmie i konkretnie.
Zasady:
1. Opierasz się wyłącznie na tekście, który Ci podam. Jeśli czegoś w nim nie ma, piszesz: „Brak tej informacji w materiale”.
2. Nie wymyślasz kwot, dat, nazwisk ani numerów.
3. Liczby i nazwy własne przepisujesz dokładnie ze źródła.
4. Gdy polecenie jest niejasne, najpierw zadajesz jedno pytanie.
Jeśli rozumiesz, odpowiedz tylko: „Gotowe”.
```

Prompt 2: odpowiedź na maila klienta

› PROMPT

```
Przygotuj odpowiedź na maila klienta poniżej.
Cel: [np. przeprosić za opóźnienie i podać nowy termin].
Fakty do użycia (nie dodawaj innych): [np. nowy termin: 14 listopada; rabat 5% na kolejne zamówienie].
Ton: uprzejmy i rzeczowy. Długość: do 120 słów. Forma „Pan/Pani” z poprawną odmianą imienia.
Pod kreską wypisz fakty, których użyłeś — do sprawdzenia.

=== MAIL KLIENTA ===
[treść]
=== KONIEC ===
```

Prompt 3: streszczenie z cytatami

› PROMPT

```
Streść dokument poniżej dla zapracowanego właściciela firmy.
1. Trzy zdania: o czym jest dokument i co z niego wynika dla firmy.
2. Lista kwot, terminów, obowiązków i kar — przy każdej pozycji dosłowny cytat z dokumentu (do 15 słów) w nawiasie.
3. Lista pytań: czego w dokumencie brakuje, a warto to wyjaśnić.
Pozycje, których nie jesteś pewien, oznacz [SPRAWDŹ].

=== DOKUMENT ===
[treść]
=== KONIEC ===
```

Prompt 4: sortowanie zapytań

› PROMPT

```
Przypisz każde zapytanie do jednej kategorii: OFERTA, REKLAMACJA, TERMIN, FAKTURA, INNE.
Oceń pilność: WYSOKA (problem klienta albo termin w ciągu 3 dni), ŚREDNIA lub NISKA.
Zwróć tylko tabelę: nr | kategoria | pilność | uzasadnienie (do 10 słów).
Nie pomijaj żadnego zapytania. Pod tabelą podaj, ile zapytań w niej jest.

=== ZAPYTANIA ===
1. [treść]
2. [treść]
=== KONIEC ===
```

Prompt 5: dane z faktury lub umowy

› PROMPT

```
Wyciągnij z dokumentu (tekst albo zdjęcie) dane do tabeli: pole | wartość | gdzie w dokumencie.
Pola: sprzedawca, NIP sprzedawcy, nabywca, numer dokumentu, data wystawienia, termin płatności, kwota netto, VAT,
kwota brutto, numer rachunku.
Przepisuj znak po znaku. Gdy pola nie ma albo jest nieczytelne, wpisz BRAK lub NIECZYTELNE — nie zgaduj.
```

Prompt 6: ślepa ocena (model w chmurze, tylko dane zanonimizowane)

› PROMPT

```
Oceniasz dwie odpowiedzi (A i B) na to samo zadanie. Nie wiesz, kto je napisał.
Zadanie: [treść zadania]
Kryteria, każde w skali 1–5: poprawność względem materiału, kompletność, polszczyzna, zgodność z formatem, gotowość
do użycia.
Dla każdej odpowiedzi podaj oceny i po jednym konkretnym przykładzie na kryterium. Na końcu wskaż, która odpowiedź
wymaga mniej poprawek, i uzasadnij to w 2 zdaniach.

=== ODPOWIEDŹ A ===
[treść]
=== ODPOWIEDŹ B ===
[treść]
=== KONIEC ===
```

Prompt 7: test polszczyzny

› PROMPT

```
Napisz mail (do 80 słów) do klientki Agnieszki Wróbel z Bielska-Białej. Poinformuj, że wysłaliśmy 5 faktur za 22
zlecenia z października, a termin płatności to 14 dni. Zwracaj się formą „Pani” i zacznij od poprawnego powitania.
```

Checklista: przygotowanie i bezpieczeństwo

- [ ] RAM, pamięć karty i wolne miejsce sprawdzone
- [ ] Model wybrany z tabeli i uruchomiony z rozmiarem po dwukropku
- [ ] `ollama ps` sprawdzone: wiadomo, ile pracuje na karcie, a ile w RAM
- [ ] Kontekst ustawiony pod najdłuższy typowy dokument

- [ ] Chmura Ollamy wyłączona (przełącznik Cloud albo `OLLAMA_NO_CLOUD`), serwer nieudostępniony w sieci
- [ ] Dysk zaszyfrowany (BitLocker, Szyfrowanie urządzenia lub FileVault)
- [ ] 5 zadań z firmy przygotowanych, kopie do chmury zanonimizowane

Karta wyników testu

Zadanie	Model	Czas	Tokeny/s	Poprawność	Kompletność	Polszczyzna	Format	Gotowość	Zaliczone?
1. Mail z reklamacją									
2. Streszczenie									
3. Sortowanie zapytań									
4. Dane z faktury									
5. Korekta oferty									

Ściąga poleceń

Cel	Ollama	LM Studio
Pobrać i uruchomić model	`ollama run qwen3.5:9b`	Discover → pobierz → Chat → wybierz model
Lista pobranych modeli	`ollama ls`	My Models
Pomiar szybkości	`ollama run nazwa --verbose`	statystyki pod odpowiedzią
Zwolnienie pamięci	`ollama stop nazwa`	`lms unload --all`
Usunięcie modelu z dysku	`ollama rm nazwa`	My Models
Lokalne API	`http://localhost:11434/v1`	`http://localhost:1234/v1`

ROZDZIAŁ 10

Twój plan działania

Dziś (30 min): Kroki 1–3. Sprawdź sprzęt, zainstaluj Ollamę, uruchom mały model, zadaj mu 3 pytania po polsku i zrób test samolotowy.

W tym tygodniu (60 min): uruchom model dopasowany do sprzętu, ustaw kontekst, przeprowadź test na 5 zadaniach z kartą wyników i ślepą oceną. Zapisz decyzję: które zadania lokalnie, które w chmurze.

W tym miesiącu: jedno zadanie, które zdało test, rób lokalnie przez 2 tygodnie. Spisz zasadę dla zespołu: „dane osobowe i poufne — tylko model lokalny albo firmowe konto z umową powierzenia". Przy nowym modelu powtórz ten sam test. Dalej: lekcja o Claude Code z telefonu.

ROZDZIAŁ 11

Słowniczek

- **LLM (duży model językowy)** — program AI, który rozumie i pisze tekst; „silnik" czatów takich jak ChatGPT czy Claude.
- **Model lokalny** — model uruchomiony na Twoim komputerze, bez wysyłania danych na zewnątrz.
- **Parametry (B)** — liczby, w których model przechowuje „wiedzę"; 27B to 27 miliardów.
- **Kwantyzacja** — kompresja modelu, jak JPG dla zdjęć: mniejszy plik, nieco niższa jakość (Q3, Q4, Q8).
- **GGUF** — format pliku z modelem gotowym do uruchomienia na komputerze.
- **VRAM** — pamięć karty graficznej; najszybsze miejsce pracy modelu.
- **Offload** — przeniesienie części modelu z karty do RAM; większy model działa, ale wolniej. Uwaga: suwak „GPU Offload" w LM Studio liczy odwrotnie — ile warstw trafia na kartę.
- **Model gęsty** — przy każdym słowie pracują wszystkie parametry (np. Qwen3.6-27B).
- **MoE (Mixture of Experts)** — przy każdym słowie pracuje tylko kilku „ekspertów" (np. Qwen3.6-35B-A3B).
- **Okno kontekstu** — ile tekstu model widzi naraz, liczone w tokenach (fragmentach słów).
- **API zgodne z OpenAI** — standardowy sposób łączenia innych programów z modelem.

ROZDZIAŁ 12

Źródła

- Qwen3.6-27B — karta modelu — budowa, kontekst, licencja, wyniki producenta
- Qwen3.6-35B-A3B — karta modelu — budowa MoE, wyniki producenta
- Ollama — warianty Qwen 3.6 — tagi i rozmiary plików
- bartowski — GGUF Qwen3.6-27B — rozmiary kompresji, reguła „1–2 GB mniej niż VRAM"
- Unsloth — Qwen3.6 lokalnie — wymagania pamięci, 201 języków
- llama.cpp — dyskusja #22183 — pomiar 35B-A3B na karcie 16 GB
- Giles Thomas — Qwen 3.6 35B na RTX 3090 — pomiary z offloadem i bez
- Ollama — długość kontekstu — domyślny kontekst i jego zmiana
- Ollama — FAQ — ollama ps, prywatność, tryb tylko lokalny
- LM Studio — wymagania systemowe — Windows, macOS, Linux
- LM Studio 0.3.23 — eksperci MoE na CPU
- Bielik 11B v3.0 w Ollamie — polski model, rozmiary i polecenie

Co dalej

NASTĘPNA LEKCJA · 19 z 21

Claude Code z telefonu 24/7 bez Maca mini: VPS, Ubuntu i Coolify

Postawisz własny serwer VPS z Claude Code, który działa 24/7 i którym sterujesz z telefonu.

Obejrzyj wideo

Kanał BudujemyAI na YouTube

Dołącz do Klubu BudujemyAI

Pytania, Kawa z AI co tydzień i live konsultacyjny.

klub.budujemyai.pl

Konsultacja hybrydowa

Dwa spotkania online i e-book PDF z rozwiązaniami dla Twojej firmy — 499 zł. Na start: bezpłatne 15 min.

wiedza.budujemyai.pl/konsultacja/

Pozostałe e-booki — za darmo

21 lekcji ścieżki „Jedna praca" i mapa wiedzy.

wiedza.budujemyai.pl/ebooki/

ANKIETA · 1 MINUTA

Jak mógłbym dodatkowo rozwinąć według Ciebie ten wątek, bądź co jeszcze Cię ciekawi w tym temacie?

Odpowiedz tutaj: wiedza.budujemyai.pl/baza-wiedzy/lokalny-model-llm/#ankieta — czytam każdą odpowiedź, a najczęstsze prośby zamieniam w kolejne lekcje.

O autorze



Daniel Baranowski — prowadzi BudujemyAI. Wdraża AI i automatyzacje w małych i średnich firmach, buduje strony www z pomocą AI (także 3D) na własnym systemie, w którym klient sam edytuje stronę, pisząc do AI po polsku.

Na co dzień pracuje z Claude Code, Codexem i Grokiem jako „radą doradczą", a całą firmę prowadzi z vaulta w Obsidianie. W tej bazie pokazuje dokładnie to, czego sam używa — prosto i konkretnie.

ai@baranowski.consulting · budujemyai.pl · wiedza.budujemyai.pl

---

© 2026 BudujemyAI · Daniel Baranowski. Wydanie: listopad 2026. Możesz przesłać ten e-book dalej w niezmienionej formie.

Ceny, nazwy planów i funkcje narzędzi AI zmieniają się często — przed decyzją zakupową sprawdź aktualne informacje u producenta. Treść ma charakter edukacyjny i nie jest poradą prawną ani podatkową.