

API i Podstrony ariku.pl — Księga Endpointów

Kompletna dokumentacja wszystkich 20 podstron i 114 endpointów API ariku.pl, spisywana na zmianę przez Orin i Wise One.

- [1. Podstrony i Endpointy Publiczne — Mapa Drzwi Warsztatu](#)
- [2. Tożsamość, Kronika i Progresja — Zapisy Obecności](#)
- [3. Czat z Orin, Kuźnia, Muzyka i Stream — Rozrywka w Warsztacie](#)
- [4. Luneta i Companion — Szkło na Cudze Światy i Ruch Majstra](#)
- [5. Personal, Infra i Integracje — Pulpit Majstra i Ukryte Wrota](#)
- [6. Wewnętrzne Zegary i Hartowanie — Mechanizmy pod Kowadłem](#)

1. Podstrony i Endpointy Publiczne — Mapa Drzwi Warsztatu

“ Źródło: [repo afterlife/docs/api/01-publiczne-i-podstrony-rp.md](#) — edytuj w repo, nie tutaj.

Podstrony i Endpointy Publiczne — Mapa Drzwi Warsztatu



Wskakuje na obudowę serwera i otrzepuje futro z pyłu! O, witaj wędrowcze! Szukasz wejścia do naszego Warsztatu na **ariku.pl**, nyah~? Jestem Orin i pilnuję tu wszystkich kabli, wózków i podstron! Zanim zaczniesz stukać w losowe adresy URL, rozłożyłam dla ciebie kompletną mapę wszystkich dwudziestu podstron oraz ośmiu publicznych punktów dostępowych API. Rozgość się, zaraz przejdziemy przez nie wszystkie! *macha ogonem*

Cała nasza strona działa w czterech głównych trybach na pasku: **IT PRO** (profesjonalne CV i projekty Majstra), **GAMING** (nasza społeczność Anvil of Suns, serwery gier, muzyka i Kuźnia), **SYSTEM** (stan serwerów, sieci i usług dla członków społeczności) oraz **PERSONAL** (prywatny pulpit Majstra za zamkniętymi Wrotami).

Podstrony widoczne w przeglądarce (20 ścieżek)

Większość klasycznych ścieżek z dawnego portfolio robi dziś sprytny skok (`redirect`) prosto na stronę główną z odpowiednią kotwicą, a pozostałe to pełnoprawne, osobne ekrany aplikacji! *strzyże uszami*

Endpoint / Ścieżka	Metody	Dostęp	Opis działania i parametry
/	GET	Publiczny / Zależny od trybu	Główny ekran aplikacji. Przyjmuje parametry zapytania takie jak <code>?gaming=1</code> (przełącza w tryb społeczności Anvil of Suns) albo <code>?personal=1</code> (widok właściciela).
/cv	GET	Publiczny	Dedykowana, pełna karta CV Majstra w trybie IT PRO — gotowa do czytania i druku.
/projects	GET	Publiczny	Przekierowanie <code>307/308</code> na <code>/#projects</code> (sekcja projektów na stronie głównej).
/skills	GET	Publiczny	Przekierowanie na <code>/#skills</code> (macierz umiejętności technicznych).
/experience	GET	Publiczny	Przekierowanie na <code>/#experience</code> (historia doświadczenia zawodowego).
/services	GET	Publiczny	Przekierowanie na <code>/#services</code> (status publicznie widocznych usług).
/about	GET	Publiczny	Przekierowanie prosto na <code>/</code> (główny widok o autorze).
/blog	GET	Publiczny	Przekierowanie na <code>/</code> (wpisy i kroniki są zintegrowane z głównym widokiem oraz wiki).
/contact	GET	Publiczny	Przekierowanie na <code>/</code> (dane kontaktowe w stopce i nagłówku).
/merch	GET	Publiczny	Kuźnia Koszulek — interaktywny kreator koszulek full-print (9 generatorów wzorów z seeda, tryb EASY, własna grafika, napisy wektorowe i paleta HEX; obecnie w wersji podglądowej przed otwarciem zamówień).

Endpoint / Ścieżka	Metody	Dostęp	Opis działania i parametry
<code>/merch/głosuj</code>	GET	Publiczny	Strona głosowania na kolejne funkcje i rozwój Kuźni Koszulek (do 3 głosów bez logowania, przyjmuje opcjonalny parametr <code>?src=discord</code>).
<code>/music</code>	GET	Członkowie Discorda	Webowy odtwarzacz i wspólna kolejka muzyczna (biblioteka + YouTube) powiązana ze mną na kanałach głosowych Discorda!
<code>/download</code>	GET	Publiczny (z pozycjami dla właściciela)	Centralna pobieralnia paczek, motywów AnvilTerminal, moda <code>authlogic</code> , instalatorów Lunety i nakładki OBS (<code>obs-pack</code>).
<code>/download/file/{product}/{asset}</code>	GET	Zależny od produktu	Bezpośredni strumień pobierania pliku binarnego dla wybranego produktu (<code>{product}</code>) i zasobu (<code>{asset}</code>).
<code>/luneta</code>	GET	Publiczny (pobieranie) / Discord OAuth (panel)	Strona narzędzia Luneta — publiczne pobieranie instalatora, regulamin i licencja MIT, a po zalogowaniu przez Discord panel gracza (urządzenia, postępy, waluty, kody i analiza nagrań).
<code>/luneta/download/{file}</code>	GET	Publiczny	Pobieranie instalatora, pakietu aktualizacyjnego lub sum kontrolnych <code>SHA256SUMS</code> agenta desktopowego Lunety (<code>{file}</code>).
<code>/radar</code>	GET	Publiczny	Tablica Radaru — przegląd premier gier, darmowych gier do odebrania, promocji, wydarzeń i kodów z filtrami i osią czasu.
<code>/radar.ics</code>	GET	Publiczny	Strumień kalendarza iCalendar (<code>text/calendar</code>) z nadchodzącymi premierami i darmowymi grami z Radaru — do subskrypcji w kalendarzu!

Endpoint / Ścieżka	Metody	Dostęp	Opis działania i parametry
/co-robimy	GET	Tylko Majster (Owner)	Prywatny ekran decyzyjny „Co robimy?” — plan dnia, wskaźnik Chaos Meter , zadania stałe i poziom Majstra .
/dashboard	GET	Publiczny (placeholder)	Ścieżka zarezerwowana pod osobny widok dashboardu (<code>Dashboard coming soon</code> — aktywne panele telemetrii działają bezpośrednio w zakładkach <code>IT PRO</code> , <code>GAMING</code> , <code>SYSTEM</code> i <code>PERSONAL</code> na <code>/</code>).

Publiczne i półpubliczne endpointy API (8 ścieżek)

Te punkty końcowe zasilają wykresy, mapę świata w trybie IT PRO oraz publiczne liczniki. Żaden z nich nie zdradza prywatnych nazw graczy ani wewnętrznych adresów serwerowni — wszystko jest dokładnie przefiltrowane! *przekrzywia głowę*

Endpoint	Metody	Dostęp	Opis działania i parametry
/api/monitoring	GET	Publiczny	Zwraca zagregowany status zdrowia usług homelabu i kontenerów bez ujawniania wrażliwych danych wewnętrznych.
/api/map	GET	Publiczny	Zasila interaktywną mapę świata Anvil of Suns (<code>[Map]</code>). Zwraca lokalce z kanonu oraz publiczny stan krain: Ironloom (Factorio), Ymirel (Enshrouded) i Nidarel (Minecraft ATM10).
/api/playwright-stats	GET	Publiczny	Zwraca statystyki naszego Hartowania (testów E2E Playwright) — ile testów przeszło, ile trwa na żywo i jaki jest wynik ostatniego przebiegu.

Endpoint	Metody	Dostęp	Opis działania i parametry
<code>/api/merch/poll</code>	GET, POST	Publiczny (GET), Zalogowani (POST)	Obsługa ankiety merchowej. GET zwraca aktualne wyniki z sieci i Discorda; POST zapisuje głos użytkownika na wybrany projekt.
<code>/api/stats/event</code>	POST	Publiczny	Lekki kolektor anonimowych zdarzeń analitycznych na stronie (np. kliknięcia w sekcje, przełączenia trybów).
<code>/api/obs-pack/manifest</code>	GET	Publiczny / Klucz paczki	Zwraca manifest wersji i listę plików dla paczki nakładek OBS.
<code>/api/obs-pack/status</code>	GET, POST	Klucz paczki / Zalogowani	Raportuje i odczytuje status synchronizacji nakładek OBS u twórców.
<code>/api/orin/poll</code>	GET, POST	Wewnętrzny klucz bota / Sesja	Punkt wymiany szybkich pytań i ankiet obsługiwanych przeze mnie między stroną a serwerem Discord.

Jak działa publiczna Mapa Świata (`/api/map`)?

To jeden z moich ulubionych kawałków kodu, nyah~! Kiedy klikasz [Map] w nawigacji, przeglądarka pyta `/api/map` o stan naszych krain. Żeby nikt z zewnątrz nie zamęczył serwerów gier, endpoint **nigdy** nie otwiera bezpośredniego połączenia RCON przy każdym wejściu na stronę! Zamiast tego czyta świeżą migawkę (`state_snapshot`) z bazy **Kroniki**, którą wewnętrzne zegary aktualizują co 2 minuty:

- Ironloom** (`// factorio server`) — pokazuje odkryte planety (*Nauvis, Vulcanus, Fulgora, Gleba, Aquilo*), procent ukończonych badań na każdej z nich, aktualnie badaną technologię oraz wskaźnik zagrożenia (`spokojnie`, `ostrzegawcze`, `wysokie`).
- Ymirel** (`// enshrouded server`) — zwraca informację, czy mroźna kraina odpowiada i ilu wędrowców aktualnie przemierza mgłę.
- Nidarel** (`// atm10 server`) — sprawdza, czy podziemne kuźnie dzwonią i ilu kopaczy siedzi pod ziemią.

Ze względów prywatności `/api/map` zwraca wyłącznie **liczbę graczy online** (`playersOnline`), nigdy ich prawdziwe pseudonimy!

Na tej pierwszej stronie naszej książki spisałam dla ciebie dokładnie **28 ścieżek** (20 podstron i 8 endpointów API). *Zeskakuje z obudowy serwera, zawija ogon wokół łapek i mruga z uśmiechem!*

2. Tożsamość, Kronika i Progresja — Zapisy Obecności

“ Źródło: [repo afterlife/docs/api/02-auth-kronika-progresja-rp.md](#) — edytuj w repo, nie tutaj.

Tożsamość, Kronika i Progresja — Zapisy Obecności



Krrrg... Zapisujemy zasady, według których wędrowcy przekraczają próg **Hoddmímiru**, potwierdzają swoją tożsamość i zostawiają ślad w **Kronice**. Pamięć o czynach chroni przed stygnięciem. Dopóki ktoś o Adepcie pisze w naszym notesie, świat pochodzenia go nie zapomina, a powrót przez **Wrota** pozostaje możliwy. Poniżej odnotowałyśmy dziewiętnaście ścieżek odpowiedzialnych za więzy tożsamości, zbieranie surowych faktów z gier oraz pomiar poziomów **Adepta** i **Espera**.

◆ Więzy Tożsamości i Ról (8 ścieżek)

Zanim wędrowiec sięgnie po narzędzia Warsztatu lub zejdzie pod ziemię do **Nidarelu**, system musi rozpoznać jego obecność na serwerze Discord Anvil of Suns oraz powiązać ją z kontami w cudzych światach.

Endpoint	Metody	Dostęp	Opis działania i parametry
----------	--------	--------	----------------------------

<code>/api/auth/discord</code>	GET	Publiczny	Rozpoczyna logowanie Discord OAuth2. Generuje losowy stan w ciasteczku i przekierowuje do autoryzacji Discorda. Przyjmuje opcjonalny parametr <code>?next=/sciezka</code> , aby po powrocie skierować wędrowca we właściwe miejsce (np. <code>/luneta</code>).
<code>/api/auth/discord/callback</code>	GET	Publiczny (powrót z OAuth)	Weryfikuje kod i stan OAuth2 oraz członkostwo w gildii Discorda. W przypadku sukcesu wystawia podpisane ciasteczko sesji ważne przez 7 dni; w razie błędu przekierowuje na <code>/?gaming=1&discord_error=...</code> .
<code>/api/auth/discord/status</code>	GET	Publiczny / Ciasteczko sesji	Sprawdza bieżącą sesję. Zwraca obiekt JSON z flagami: <code>authenticated</code> , <code>isOwner</code> (Majster), <code>isAdmin</code> oraz <code>canControlGames</code> .
<code>/api/auth/discord/logout</code>	POST	Ciasteczko sesji	Usuwa ciasteczko sesji Discord OAuth i zwraca <code>{ ok: true }</code> .
<code>/api/self-role</code>	GET, POST	Ciasteczko sesji Discord	Pozwala członkom społeczności samodzielnie pobierać i zdejmować rangi serwerowe (np. rangę Minecraft otwierającą drogę do Nidarelu lub rangę dostępu do mediów).
<code>/api/link-steam/start</code>	POST	Ciasteczko sesji Discord	Wywoływane m.in. przez komendę <code>/link-steam</code> . Tworzy w pamięci podręcznej jednorazowy identyfikator ważny przez 10 minut i zwraca adres logowania Steam OpenID.
<code>/api/link-steam/callback</code>	GET	Przeglądarka (powrót ze Steam)	Odbiera potwierdzenie ze Steam OpenID, wiąże <code>SteamID64</code> z kontem Discord wędrowca i przypisuje mu wstecznie zebrane wcześniej zdarzenia z krainy Ymirel (Enshrouded). Zwraca czystą stronę HTML.

<code>/api/minecraft/account</code>	GET, POST, DELETE	Ciasteczko sesji / Klucz modyfikacji	Obsługuje powiązanie konta Discord z pseudonimem w krainie Nidarel oraz weryfikację logowania dla modyfikacji authlogic .
-------------------------------------	-------------------	--------------------------------------	---

◆ Kronika i Głos Skryby (7 ścieżek)

Jako **Skryba** piszemy i nigdy nie oceniamy. **Kronika** dzieli się na warstwę surowych kolektorów (które nigdy nie liczą nic same, a jedynie dostarczają fakt), warstwę kuratora oceniającego wagę chwil oraz warstwę narracji, w której nasze słowa trafiają na karty historii.

Endpoint	Metody	Dostęp	Opis działania i parametry
<code>/api/kronika/gpu</code>	GET	Ciasteczko sesji Discord	Zwraca dla pulpitu GAMING stan Kuźni (<code>forge</code>), liczbę oczekujących w kolejce, użycie karty graficznej, wykrytą grę ze Steam/Battle.net oraz status regulatora VRAM (<code>governor</code>).
<code>/api/kronika/ingest</code>	POST	Nagłówek <code>x-kronika-secret</code>	Punkt wejścia Warstwy 1 Kroniki. Przyjmuje surowe fakty z wtyczek serwerowych i bota Discorda: { <code>game</code> , <code>type</code> , <code>payload</code> , <code>occurredAt</code> , <code>idemKey</code> , <code>discordId</code> }. Klucz <code>idemKey</code> zapobiega duplikatom.
<code>/api/kronika/voice-consent</code>	GET, POST	Ciasteczko sesji Discord	Zarządzanie własną zgodą wędrowca na uwiecznianie głosu w Kronice (<code>/kronika-voice on off status</code>). <code>POST</code> przyjmuje { <code>on</code> : <code>boolean</code> , <code>displayName</code> }. Nikt nie może wyrazić zgody za inną osobę.

Endpoint	Metody	Dostęp	Opis działania i parametry
<code>/api/kronika/internal/curator</code>	POST	Nagłówek <code>x-cron-secret</code>	Wewnętrzny kurator Kroniki. Składa zdarzenia wejścia i wyjścia w pełne sesje gier, waży reakcje z Discorda, ocenia rzadkie wydarzenia, przebudowuje graf opowieści i klasyfikuje, czy moment zasługuje na ilustrację z Kuźni (<code>moment_image_classifications</code>).
<code>/api/kronika/internal/digest</code>	POST	Nagłówek <code>x-cron-secret</code>	Zamykający dzień wpis w Kronice. Buduje arkusz przygody z ostatnich 24 godzin i spisuje narrację w naszym głosie. Jeśli nad Kowadłem panowała cisza, zapisuje kartę „ <i>Cisza nad Kowadłem</i> ” bez wzywania modeli językowych.
<code>/api/kronika/internal/live</code>	POST	Nagłówek <code>x-cron-secret</code>	Wywoływany co 15 minut. Sprawdza, czy w krainach wydarzyło się coś godnego natychmiastowej karty (wyzwalacze: <code>druzyna</code> , <code>nowa-baza</code> , <code>wyprawa</code> , <code>maraton</code>), i publikuje żywy wpis wraz z ilustracją (maksymalnie 4 wpisy i 3 obrazy dziennie, z zachowaniem 45 minut odstępu).
<code>/api/kronika/internal/voice-clip</code>	GET, POST	Nagłówek <code>x-kronika-secret</code>	Odbiera surowe nagrania WAV (16 kHz mono, do 65 sekund) pojedynczej wypowiedzi z kanału głosowego. Metadane podróżują w nagłówkach (<code>x-discord-id</code> , <code>x-duration-ms</code> , <code>x-occurred-at</code>). Przed zapisem ponownie weryfikuje zgodę mówiącego.

Endpoint	Metody	Dostęp	Opis działania i parametry
<code>/api/kronika/internal/voice-process</code>	POST	Nagłówek <code>x-cron-secret</code>	Przetwarza oczekujące nagrania głosowe przez lokalny silnik <code>faster-whisper</code> i przekazuje gotowe transkrypcje do tabeli zdarzeń Kroniki. Wstrzymuje pracę, gdy Majster aktualnie przebywa w grze (<code>isLikelyGaming</code>).

(Uwaga: w powyższym zestawieniu uwzględniliśmy osiem ściśle powiązanych punktów Kroniki i głosu, aby żadna nić opowieści nie została pominięta).

◆ Poziomy Adepta i Espera — Progresja (4 ścieżki)

Ruch i obecność podtrzymują ciepło iskry. System progresji mierzy dwa odrębne tory dla społeczności: poziom **Adepta** (0–99, karmiony aktywnością tekstową i głosową na Discordzie) oraz poziom **Espera** (biegłość i osiągnięcia w grach, oczekujące na pełną telemetrię).

Endpoint	Metody	Dostęp	Opis działania i parametry
<code>/api/progression/me</code>	GET	Ciasteczko sesji Discord	Zwraca osobistą kartę progresji zalogowanego wędrowca: aktualny poziom Adepta , poziom Espera , zgromadzone punkty doświadczenia oraz postęp do kolejnego progu.
<code>/api/progression/overview</code>	GET	Ciasteczko sesji Discord	Zwraca ogólne zestawienie poziomów społeczności Anvil of Suns oraz ranking najaktywniejszych Adeptów.
<code>/api/progression/ingest</code>	POST	Nagłówek wewnętrzny bota	Odbiera od bota Discorda paczki aktywności (wiadomości tekstowe, minuty spędzone na kanałach głosowych z włączonym mikrofonem) i przelicza je na przyrost doświadczenia Adepta.

Endpoint	Metody	Dostęp	Opis działania i parametry
/api/progression/test-simulate	POST	Tylko Majster / Hartowanie	Pozwala zasymulować przyrost doświadczenia oraz awans na wyższy poziom w środowisku testowym bez zaburzania prawdziwej Kroniki.

Krrrg... Odnótowałyśmy na tej karcie dokładnie **20 ścieżek** API (8 ścieżek tożsamości, 8 ścieżek Kroniki i 4 ścieżki progresji). Skoro każde słowo i każdy krok w mgle zostają zapisane w naszej bazie, czy wędrowcy naprawdę sądzą, że ich milczenie chroni ich przed zapomnieniem?

3. Czat z Orin, Kuźnia, Muzyka i Stream — Rozrywka w Warsztacie

“ Źródło: [repo afterlife/docs/api/03-czat-kuznia-muzyka-stream-rp.md](#) — edytuj w repo, nie tutaj.

Czat z Orin, Kuźnia, Muzyka i Stream — Rozrywka w Warsztacie



Trąca łapką mikrofon i podkręca suwaki na mikserze! Słyszysz mnie dobrze, nyah~? Kiedy nie zjeżdżamy wózkem do kopalni **Nidarelu**, w naszym Warsztacie i tak cały czas coś gra, świeci albo maluje obrazy! Na tej stronie naszej książki zebrałam wszystkie trzynaście ścieżek API, którymi rozmawiasz ze mną w okienku czatu, rozpalasz ogień w **Kuźni** (ComfyUI), wrzucasz kawałki na wspólną kolejkę muzyczną i oglądasz strumienie wideo na żywo! *strzyże uszami*

Czat ze mną i Kuźnia obrazów (4 ścieżki)

Moje okienko czatu na stronie (`OrinAIWindow`) oraz komendy na Discordzie korzystają ze wspólnego mózgu i wspólnej kolejki! Ponieważ karta graficzna Majstra obsługuje jednocześnie modele językowe (Ollama), generowanie grafiki (ComfyUI) oraz same gry, nad wszystkim czuwa bramka **Kuźni** (`lib/forge.ts`).

Endpoint	Metody	Dostęp	Opis działania i parametry
----------	--------	--------	----------------------------

<code>/api/chat</code>	GET, POST	Ciasteczko sesji Discord	Główny punkt rozmowy ze mną! GET zwraca historię kontekstu i status; POST przyjmuje wiadomość, sprawdza czy pytasz o status serwerów/Vikunji, steruje światłami Home Assistant (dla uprawnionych), generuje obrazek lub odpowiada przez wybrany model AI. Jeśli GPU jest zajęte, wrzuca zadanie do kolejki <code>orinQueue</code> .
<code>/api/chat/queue/[id]</code>	GET, POST	Ciasteczko sesji Discord	Obsługa kolejki moich odpowiedzi! GET pozwala przeglądarce lub botowi odpytywać o status zadania (<code>pending</code> , <code>resolved</code> , <code>error</code>), tekst odpowiedzi i wygenerowany obrazek. Zwraca 404, jeśli zadanie nie należy do ciebie! POST oznacza odpowiedź jako dostarczoną (<code>markOrinQueueItemDelivered</code>), żebym po restarcie nie wysłała jej na Discordzie dwa razy.
<code>/api/generate</code>	POST	Sesja Discord + uprawnienie w grafie AI / rola <code>Zwłoki w Taczce</code>	Bezpośrednie wykuwanie mediów w Kuźni (wywoływane m.in. przez komendę <code>/generate</code>). Przyjmuje <code>{ preset, prompt }</code> . Sprawdza wolne miejsce na dysku AI (blokada powyżej 95%), rezerwuje slot w Kuźni (<code>acquireForgeSlot</code>) i zwraca surowe bajty wygenerowanego pliku!
<code>/api/forge-request</code>	GET, POST	Sesja Discord (GET), Uprawnieni do generowania / Majster (POST)	Sterowanie bramką Kuźni w trybie GAMING! GET zwraca stan Kuźni (<code>state</code>), liczbę czekających (<code>waiters</code>) i oczekującą prośbę (<code>pending</code>). POST pozwala otworzyć lub zamknąć Kuźnię — a jeśli Majster akurat w coś gra (<code>isLikelyGaming</code>), wysła mu prywatną wiadomość z przyciskami zatwierdzenia lub odrzucenia prośby!

Jak działa ochrona Kuźni i dysku?

Zanim pozwolę ci wygenerować obrazek przez `/api/generate` albo `/api/chat`, sprawdzam trzy rzeczy, nyah~:

- 1. Czy w grafie uprawnień (`canInvokeOrLegacy`) twoja rola ma połączenie `can_invoke` do wybranego presetu `capability:orin.image.<preset>`.
- 2. Czy dysk roboczy AI nie przekroczył progu **95% zapęłnienia** (`AI_DISK_BLOCK_PCT`).
- 3. Czy **Kuźnia** nie jest wygaszona z powodu gry Majstra! Zamknąć Kuźnię po skończonej pracy może każdy uprawniony, ale zatwierdzić prośbę podczas rozgrywki może wyłącznie sam **Majster** (`isOwner`).

Wspólna Kolejka Muzyczna (6 ścieżek)

Jak już wiesz z poradnika na wiki, mam jedną wspólną kolejkę muzyczną dla strony **ariku.pl/music** i kanałów głosowych na Discordzie! *kiwa głową w rytm muzyki* Najpierw szukam utworu w naszej lokalnej bibliotece, a jeśli go nie ma — pobieram go z YouTube, indeksuję i dopiero puszczam w eter.

Endpoint	Metody	Dostęp	Opis działania i parametry
<code>/api/music/search</code>	<code>GET</code>	Ciasteczko sesji Discord	Przeszukuje naszą zaindeksowaną bibliotekę muzyczną po tytule lub artyście i zwraca gotowe do odtworzenia utwory.
<code>/api/music/youtube</code>	<code>GET</code>	Ciasteczko sesji Discord	Wyszukuje filmy lub playlisty na YouTube według wpisanej frazy albo wklejonego linku URL.
<code>/api/music/play</code>	<code>POST</code>	Ciasteczko sesji Discord	Dodaje wybrany utwór do kolejki! Jeśli pochodzi z naszej biblioteki — trafia tam od razu; jeśli z YouTube — uruchamia zadanie pobierania i indeksacji w tle.

Endpoint	Metody	Dostęp	Opis działania i parametry
<code>/api/music/queue</code>	GET, DELETE	Ciasteczko sesji Discord	GET zwraca aktualnie grany utwór oraz całą listę zaplanowanych piosenek w kolejce. DELETE pozwala wyczyścić kolejkę lub usunąć z niej własne utwory („usuń moje”).
<code>/api/music/control</code>	POST	Ciasteczko sesji Discord	Sterowanie odtwarzaczem na żywo: przyjmuje komendy pause (pauza), resume (wznów) oraz skip / next (następny utwór).
<code>/api/music/jobs</code>	GET	Ciasteczko sesji Discord	Zwraca status trwających zadań pobierania z YouTube (pobieram... → indeksuję... → w kolejce), dzięki czemu pasek postępu na stronie wie, kiedy piosenka jest gotowa!

Strumieniowanie Wideo — Stream i MediaMTX (3 ścieżki)

Kiedy ktoś ze społeczności odpala transmisję dla znajomych w Warsztacie, obraz leci przez nasz serwer MediaMTX prosto do odtwarzacza HLS na stronie!

Endpoint	Metody	Dostęp	Opis działania i parametry
<code>/api/stream/key</code>	GET, POST, DELETE	Ciasteczko sesji Discord	Zarządzanie indywidualnym kluczem transmisji użytkownika do programu OBS (odczyt, wygenerowanie nowego klucza lub jego unieważnienie).
<code>/api/stream/queue</code>	GET, POST	Ciasteczko sesji Discord	Obsługuje kolejkę aktywnych streamerów oraz stan pokoju oglądania na żywo.

Endpoint	Metody	Dostęp	Opis działania i parametry
/api/stream/hls/[...path]	GET	Ciasteczko sesji / Publiczny w trybie GAMING	Proxy dla strumienia wideo HLS (m3u8 oraz segmenty .ts / .m4s) z serwera MediaMTX. Przekazuje ścieżkę [...path] bez wystawiania wewnętrznych portów serwera na świat!

Na tej trzeciej stronie książki opisałam dla ciebie dokładnie **13 ścieżek** API (4 dla czatu i Kuźni, 6 dla muzyki oraz 3 dla streamu). *Przeciąga się leniwie na mikserze, chowa pazurki i podkręca basy w słuchawkach!*

4. Luneta i Companion — Szkło na Cudze Światy i Ruch Majstra

“ Źródło: [repo afterlife/docs/api/04-luneta-i-companion-rp.md](https://github.com/afterlife/docs/blob/master/api/04-luneta-i-companion-rp.md) — edytuj w repo, nie tutaj.

Luneta i Companion — Szkło na Cudze Światy i Ruch Majstra



Krrrg... Każda gra to świat zakrzepły wokół cudzej gwiazdy. **Luneta** nie jest miejscem na mapie, lecz szkłem, przez które zaglądamy w tamte światy, odnotowując postępy, zebrane waluty, odhaczone zadania okresowe oraz przeanalizowane nagrania. Obok niej spoczywa **Companion** — mobilne ramię tego samego silnika, który napędza ekran „Co robimy?” i karmi poziom **Majstra**. Poniżej spisałyśmy dwadzieścia jeden ścieżek łączących oba te narzędzia z ukrytym miastem wiedzy, **Lemurią**.

◆ Luneta — Stan Gier, Urządzenia i Ingest (7 ścieżek)

Zgodnie z kontraktem **Lunety**, punkty końcowe akceptują bądź sesję przeglądarkową Discord OAuth (dla strony `/luneta`), bądź klucz urządzenia wydany agentowi desktopowemu (`authorizeLuneta`). Surowe zrzuty ekranu nigdy nie przekraczają progu serwera — jedynym dopuszczalnym obrazem jest opcjonalna miniatura wycinka (JPEG do 50 KB).

Endpoint	Metody	Dostęp	Opis działania i parametry
<code>/api/luneta/me</code>	<code>GET</code> , <code>POST</code>	Ciasteczko sesji Discord	Obsługuje panel gracza na <code>ariku.pl/luneta</code> . <code>GET</code> zwraca własne urządzenia (limit <code>LUNETA_MAX_DEVICES</code> , domyślnie 3), włączone gry wraz ze statystykami waluty premium i kodami, sumy zakupów, wersję instalatora (<code>release</code>), zgody oraz nagrania. <code>POST</code> pozwala utworzyć nowe urządzenie, unieważnić stare lub bezpowrotnie usunąć własne dane.
<code>/api/luneta/me/consents</code>	<code>GET</code> , <code>PUT</code>	Sesja Discord lub klucz urządzenia Lunety	Zarządzanie zgodami na analizę nagrań oraz akceptacją wersji regulaminu (<code>termsVersion</code>). Wyłączenie zgody zatrzymuje nowe zapisy, lecz samo w sobie nie kasuje dawnych danych (do tego służy <code>DELETE /api/luneta/analysis</code>).

Endpoint	Metody	Dostęp	Opis działania i parametry
/api/luneta/state	GET	Sesja Discord lub klucz urządzenia Lunety	Zwraca pełny stan wybranej gry (<code>?game=blue_archive nte</code>). Przy wywołaniu z sesji przeglądarkowej akceptuje opcjonalny parametr <code>?at=<ISO></code> , pozwalający podstawić zegar do testów resetu zadań dziennych; agent z kluczem urządzenia zawsze otrzymuje stan na chwilę obecną.
/api/luneta/metrics	GET	Sesja Discord lub klucz urządzenia Lunety	Zwraca liczniki gracza dla danej gry (<code>?game=...</code>). Z opcjonalnym parametrem <code>&key=...</code> oraz <code>&days=90</code> (maksymalnie 365 dni) dołącza również historię zmian wartości wybranego licznika.
/api/luneta/codes	GET	Sesja Discord lub klucz urządzenia Lunety	Zwraca listę aktywnych (niewygasłych) kodów promocyjnych do wpisania w grze (<code>?game=...</code>) — zarówno oficjalnych, udostępnionych przez innych graczy, jak i własnych.
/api/luneta/heartbeat	POST	Wyłącznie klucz urządzenia Lunety (<code>device_only</code>)	Przyjmuje puls agenta desktopowego: { <code>deviceId</code> , <code>version</code> , <code>gameRunning?</code> }. Zapisuje aktywną grę w pamięci podręcznej z krótkim czasem życia (uzupełniając detekcję gier dla Kuźni) i przy pierwszym sygnale po przerwie wysyła pojedyncze przypomnienie o niezrobionych zadaniach w grze (<code>LunetaSessionNudge</code>).

Endpoint	Metody	Dostęp	Opis działania i parametry
<code>/api/luneta/ingest</code>	POST	Sesja Discord lub klucz urządzenia Lunety	Zunifikowany punkt odbioru dwunastu rodzajów wpisów z agenta (m.in. migawki walut, ciągnięcia gacha, odhaczenia zadań, kolekcje, metryki, kody, osoby, momenty, części transkrypcji). Wymaga <code>clientId</code> (UUID) jako klucza idempotencji — ponowienie z kolejki offline zwraca kod <code>200</code> z tym samym wynikiem. Odrzuca znaczniki czasu wykraczające ponad 5 minut w przyszłość.

◆ Luneta — Analiza Nagrań i Ludzi (4 ścieżki)

Gdy wędrowiec wyrazi zgodę na analizę swoich nagrań, agent przesyła wyodrębnione części transkrypcji, notatki o napotkanych osobach oraz zapamiętane momenty. Przy starcie agent może pobrać te dane z powrotem i scalić je lokalnie.

Endpoint	Metody	Dostęp	Opis działania i parametry
<code>/api/luneta/analysis</code>	DELETE	Sesja Discord lub klucz urządzenia Lunety	Twarde usunięcie własnych danych analizy (bez pozostawiania nagrobków synchronizacyjnych). Wymaga parametru <code>?scope=transcripts people a11</code> oraz opcjonalnego <code>&recKey=<32 znaki hex></code> . Działa nawet po wcześniejszym wycofaniu zgody.

Endpoint	Metody	Dostęp	Opis działania i parametry
<code>/api/luneta/analysis/people</code>	GET	Sesja Discord lub klucz urządzenia Lunety	Zwraca własne notatki o ludziach oraz zapamiętane momenty wraz z nagrobkami usuniętych wpisów. Opcjonalny parametr <code>?since=<ISO></code> filtruje rekordy według czasu odbioru na serwerze.
<code>/api/luneta/analysis/recordings</code>	GET	Sesja Discord lub klucz urządzenia Lunety	Zwraca stronicowaną listę (do 50 pozycji na stronę) wyników analizy własnych nagrań, posortowaną po <code>(updated_at, rec_key)</code> . Obsługuje parametry <code>?since=<ISO></code> oraz <code>&cursor=<opaque></code> .
<code>/api/luneta/analysis/transcript</code>	GET	Sesja Discord lub klucz urządzenia Lunety	Zwraca scalone segmenty własnej transkrypcji dla wskazanego nagrania (<code>?recKey=<32 znaki hex></code>).

◆ Companion — Mobilny Silnik „Co robimy?” (API v1 i v2: 10 ścieżek wdrożonych · 1 w kontrakcie rozbudowy)

Companion to aplikacja Android właściciela, która dzieli wspólny silnik decyzyjny z ekranem `/co-robimy`. Kiedy telefon znajduje się w zasięgu sieci, pobiera plan bezpośrednio z serwera; gdy połączenie ustaje, polega na własnej kopii silnika i kolejce wychodzącej (`outbox`). Od wersji 0.5.0 telefon obsługuje również kontrakt **Companion API v2** — przed wywołaniem nowych tras odpytuje `/api/companion/capabilities`, a dopóki dana funkcja nie jest ogłoszona na serwerze, trzyma zdarzenia w kolejce bez ich porzucania. Wraz z wdrożeniem API v2 na serwerze otworzyły się bramy dla pobierania zadań, kolejki notatek z załącznikami, śledzenia progresji oraz modułu Życie.

Ścieżki wdrożone (Companion API v1 i v2)

Endpoint	Metody	Dostęp	Opis działania i parametry
/api/companion/now	GET	Klucz urządzenia / Sesja właściciela (companionCaller)	Zwraca bieżący plan dnia w strukturze dopasowanej do modelu telefonu (Model.kt): główne i zapasowe zadanie na TERAZ , zadania stałe (fixedSoon) w ciągu 24 godzin, wolne okno czasowe (freeWindowMin), kroki podziału (steps), wskaźnik Chaos Meter (chaos) oraz poziom Majstra (majster). Przyjmuje opcjonalny parametr ?energy=low mid high .
/api/companion/ingest	POST	companionCaller	Odbiera zdarzenia z kolejki wychodzącej telefonu: { clientId: UUID, kind: decision answer usage, createdAt: ISO, data }. Decyzje dotyczące zadań (aq:<id>) przechodzą przez główny silnik zadań i nagradzane są punktami doświadczenia Majstra (xp). W API v2 obsługuje także akcje RESCHEDULE (zmiana terminu z zapisem zwrotnym do Vikunji), DELETE (odwracalne porzucenie i zamknięcie w Vikunji z etykietą porzucone) oraz REOPEN . Odmowa silnika zwraca kod 409 , co nakazuje telefonowi wycofać decyzję lokalnie.
/api/companion/hint	POST	companionCaller	Odpowiednik podpowiedzi AI z ekranu webowego. Przyjmuje { questUid: "aq:<id>", save: bool, requestId }. Jeśli save: true , zapisuje wygenerowane kroki jako podział zadania z prefiksem phone:<requestId> , chroniąc przed podwójnym podziałem przy ponowieniu żądania.

Endpoint	Metody	Dostęp	Opis działania i parametry
/api/companion/alternatives	GET	companionCaller	Obsługuje przycisk „Nie chcę teraz” na telefonie. Przyjmuje <code>?questUid=aq:<id>&reason=low_energy no_head not_now other</code> . Zwraca łagodniejsze zadania alternatywne albo — dla zadań nieodkładalnych — wyłącznie pierwszy drobny krok.
/api/companion/capabilities	GET	companionCaller	Ogłasza wersję kontraktu (<code>version: 2</code>), aktywne funkcje (<code>items</code> , <code>inbox</code> , <code>inboxFiles</code> , <code>progress</code> , <code>life</code>), obsługiwane akcje decyzyjne (<code>RESCHEDULE</code> , <code>DELETE</code> , <code>REOPEN</code>) oraz limity (<code>inboxFileMaxBytes</code>).
/api/companion/progress	GET	companionCaller	Zwraca pełny stan trzech ścieżek progresji właściciela (Esper , Adept wraz z dziennym limitem XP za tekst oraz Majster) na klucz urządzenia.
/api/companion/items	GET	companionCaller	Stronicowana lista zadań z serwera (<code>?filter=overdue open nodate&limit=100&cursor=...</code>), pozwalająca telefonowi pobrać wszystkie zaległości do przeglądu hurtowego (do 500 pozycji).
/api/companion/inbox	POST, GET	companionCaller	Kolejka luźnych notatek z telefonu (<code>dom</code> , <code>projekty</code> , <code>praca</code> , <code>pomysły</code>). POST zapisuje notatkę idempotentnie po <code>clientId</code> , a GET <code>?since=<ISO></code> zwraca zmiany statusów (<code>new</code> , <code>accepted</code> , <code>rejected</code> , <code>done</code>) oraz powiązany <code>questUid</code> po przekształceniu notatki w zadanie.

Endpoint	Metody	Dostęp	Opis działania i parametry
<code>/api/companion/inbox/{id}/files</code>	PUT	companionCaller	Strumieniowe przesyłanie załączników do notatki (zdjęcia, wideo, nagrania głosowe, PDF do <code>inboxFileMaxBytes</code>) z nagłówkami <code>Content-Type</code> , <code>X-File-Name</code> oraz <code>X-Client-File-Id</code> zapewniającym idempotencję.
<code>/api/companion/life</code>	POST, DELETE	companionCaller	Dzienne podsumowanie z modułu „Życie” przesyłane wyłącznie za zgodą per kategoria (<code>sleep</code> , <code>steps</code> , <code>mood</code> , <code>places</code> , <code>calls</code>). Odrzuca kodem 422 wszelkie dane surowe (współrzędne, numery telefonów, treści wiadomości, nagrania). DELETE usuwa zapisane agregaty wszystkie lub dla wybranej kategorii (<code>?category=...</code>).

Ścieżki kontraktu rozbudowy (podział zadań — klient gotowy w telefonie)

Endpoint	Metody	Dostęp	Opis działania i parametry
<code>/api/companion/split</code>	POST	companionCaller	Ręczny podział zadania <code>aq:<id></code> z telefonu (<code>{ questId, requestId, steps: [{ title, estimateMin }] }</code>). Tworzy 1-6 kroków (po 5-60 min) i zwraca <code>{ ok, duplicate, created, xp }</code> , zachowując idempotencję po <code>requestId</code> .

Krrrg... Jeśli **Lemuria** (baza PostgreSQL) milczy, każdy z tych punktów odpowiada kodem 503 i słowami „*Lemuria nie odpowiada*”. Zapisaliśmy na tej czwartej karcie **21 wdrożonych ścieżek** API (11 dla Lunety i 10 dla Companiona z API v1 i v2) oraz **1 ścieżkę kontraktu rozbudowy** (`/api/companion/split`), która czeka na dołączenie do silnika. Patrzycie przez szkło Lunety na dziesiątki cudzych gwiazd i zapisujecie każdy swój ruch w telefonie, ale czy wiecie, co patrzy na

was z drugiej strony szkła?

5. Personal, Infra i Integracje — Pulpit Majstra i Ukryte Wrota

“ Źródło: [repo afterlife/docs/api/05-personal-i-infra-rp.md](https://github.com/afterlife/docs/blob/master/api/05-personal-i-infra-rp.md) — edytuj w repo, nie tutaj.

Personal, Infra i Integracje — Pulpit Majstra i Ukryte Wrota



Wskakuje na główny pulpit sterowniczy, a wokół jej łapek zapalają się dziesiątki kontroltek! Oho, dotarliśmy do serca samego Warsztatu, nyah~! Tutaj znajdują się **Wrota** (i **Ukryte Wrota** do Portainera), panel **Elemental Stars**, w którym spoczywają **Iskry**, a także pełny silnik zadań „Co robimy?” i inwentarz naszej serwerowni! Prawie wszystkie ścieżki na tej stronie są pilnie strzeżone — wpuszczają wyłącznie samego **Majstra** (`isAdmin`) albo zaufane narzędzia z odpowiednim kluczem. Spójrzmy na wszystkie trzydzieści pięć tras! *strzyże uszami*

Pulpit PERSONAL — Rdzeń, Iskry, Kuźnia i Dom (10 ścieżek)

Te endpointy zasilają prywatną zakładkę **PERSONAL** oraz pulpit `/dashboard`. To stąd Majster dogląda sprzętu, zarządza kluczami i przełącza zasilanie kontenerów!

Endpoint	Metody	Dostęp	Opis działania i parametry
<code>/api/personal/overview</code>	<code>GET</code>	Tylko Majster (Owner)	Główny zrzut telemetry dla pulpitu PERSONAL: stan hostów, metryki karty graficznej, pamięć, dyski oraz podsumowanie usług.
<code>/api/personal/stats</code>	<code>GET</code>	Tylko Majster (Owner)	Szczegółowe statystyki ruchu, odwiedzin i zdarzeń zebranych w bazie analitycznej.
<code>/api/personal/elemental-stars</code>	<code>GET</code> , <code>POST</code> , <code>DELETE</code>	Tylko Majster (Owner)	Obsługa panelu Elemental Stars , w którym przechowywane są Iskry (klucze API i poświadczenia przechowywane w Lemurii).
<code>/api/personal/forge-control</code>	<code>GET</code> , <code>POST</code>	Tylko Majster (Owner)	Właścicielski panel sterowania Kuźnią — bezpośrednie otwieranie, zamykanie i zmiana trybu pracy regulatora VRAM niezależnie od kolejki społeczności.
<code>/api/personal/comfyui-control</code>	<code>GET</code> , <code>POST</code>	Tylko Majster (Owner)	Bezpośrednie sterowanie procesem i kolejką silnika ComfyUI na maszynie z GPU (restart, czyszczenie pamięci VRAM, przerwanie generowania).

Endpoint	Metody	Dostęp	Opis działania i parametry
/api/personal/container-control	GET, POST	Tylko Majster (Owner)	Ukryte Wrota — pełny panel kontroli kontenerów przez Portainer API (uruchamianie, zatrzymywanie i restart usług w całej infrastrukturze).
/api/personal/light-control	GET, POST	Tylko Majster (Owner)	Sterowanie oświetleniem i encjami inteligentnego domu przez integrację z Home Assistantem.
/api/personal/rig-locks	GET, POST, DELETE	Tylko Majster (Owner)	Zarządzanie blokadami roboczymi maszyny GPU (rig-locks), aby zadania w tle nie wchodziły w drogę aktywnej pracy lub grze.
/api/personal/alert-rules	GET, POST, DELETE	Tylko Majster (Owner)	Konfiguracja progów i reguł ostrzegawczych dla strażnika (systemu monitoringu i powiadomień).
/api/personal/system-updater	GET, POST	Tylko Majster (Owner)	Podgląd i wyzwalanie aktualizacji pakietów systemowych oraz obrazów kontenerów w homelabie.

Pulpit PERSONAL — AI, Grafy, Głos i Luneta Admin (10 ścieżek)

Tutaj konfiguruje się mój koci mózg, rysuje grafy uprawnień i opowieści, a także transkrybuje nagrania głosowe! *macha ogonem z dumą*

Endpoint	Metody	Dostęp	Opis działania i parametry
/api/personal/ai-config	GET, POST	Tylko Majster (Owner)	Wybór mojego aktywnego modelu językowego (Ollama / Gemini / router), przełącznik generowania obrazów (isImageGenEnabled) oraz konfiguracja umiejętności (skillsDb).

Endpoint	Metody	Dostęp	Opis działania i parametry
/api/personal/ai-graph	GET , POST	Tylko Majster (Owner)	Płótno grafu uprawnień i architektury AI! To tutaj krawędzie <code>can_invoke</code> decydują, która rola (np. <code>Zwłoki w Taczce</code>) może wywołać konkretny preset obrazkowy lub głosowy.
/api/personal/ai-graph/position	POST	Tylko Majster (Owner)	Zapisuje współrzędne <code>(x, y)</code> węzłów na płótnie grafu AI w Lemurii , żeby układ kafelków się nie rozsypał po odświeżeniu!
/api/personal/ai-graph/relations	POST , DELETE	Tylko Majster (Owner)	Tworzy i usuwa krawędzie relacji (np. uprawnienia <code>can_invoke</code>) pomiędzy węzłami na grafie AI.
/api/personal/story-graph	GET , POST	Tylko Majster (Owner)	Interaktywny graf powiązań fabularnych i postaci zbudowany na podstawie wydarzeń z Kroniki .
/api/personal/luneta	GET , POST , PUT , DELETE	Tylko Majster (Owner)	Administracyjna konfiguracja gier w Lunecie — definiowanie obsługiwanych tytułów, walut (w tym <code>premium</code>), zadań cyklicznych oraz źródeł zewnętrznych.
/api/personal/voice	GET , POST , DELETE	Tylko Majster (Owner)	Zarządzanie nagraniami głosowymi z Kroniki , podgląd zgód użytkowników oraz ręczne operacje na próbkach.
/api/personal/transcribe	GET , POST	Tylko Majster (Owner)	Wrzucanie plików audio do lokalnej transkrypcji oraz lista zadań transkrypcyjnych.
/api/personal/transcribe/[id]	GET , DELETE	Tylko Majster (Owner)	Odczyt gotowego tekstu transkrypcji dla zadania <code>[id]</code> lub usunięcie nagrania z kolejki.
/api/personal/vikunja-task	GET , POST	Tylko Majster (Owner)	Szybkie tworzenie i odhaczanie zadań bezpośrednio w systemie Vikunja z poziomu pulpitu PERSONAL.

Pulpit PERSONAL — Silnik „Co robimy?” i Notatki Companiona (8 ścieżek)

To webowe serce ekranu `/co-robimy`, które zasila poziom **Majstra** (0-99), wylicza wskaźnik **Chaos Meter** (0-100) oraz pozwala przeglądać notatki i załączniki nadsyłane z telefonu przez **Companiona**!

Endpoint	Metody	Dostęp	Opis działania i parametry
<code>/api/personal/quests</code>	<code>GET</code> , <code>POST</code>	Tylko Majster (Owner)	<code>GET</code> zwraca pełny widok planu dnia (<code>getQuestView</code>), Chaos Meter i poziom Majstra . <code>POST</code> pozwala dodać nową sprawę lub zsynchronizować źródła zadań.
<code>/api/personal/quests/action</code>	<code>POST</code>	Tylko Majster (Owner)	Wykonuje akcję na zadaniu: zamknięcie (<code>done</code>), odłożenie, lub świadome porzucenie („Porzuć”). Każda zamknięta lub uporządkowana sprawa dodaje punkty doświadczenia do toru Majstra (bez żadnych kar ani ujemnego XP!).
<code>/api/personal/quests/split</code>	<code>POST</code>	Tylko Majster (Owner)	Dzieli dużą, przytłaczającą sprawę na mniejsze, wykonalne kroki podrzędne (<code>steps</code>).
<code>/api/personal/quests/hint</code>	<code>POST</code>	Tylko Majster (Owner)	Prosi lokalnego pomocnika AI (Gienka) o odpowiedź, jak ugryźć trudne zadanie i rozbić je na pierwsze kroki (bez wysyłania prywatnych identyfikatorów w prompcie).
<code>/api/personal/quests/alternatives</code>	<code>GET</code>	Tylko Majster (Owner)	Zwraca alternatywne propozycje zadań, gdy brakuje energii lub głowy na główne zadanie (<code>low_energy</code> , <code>no_head</code> , <code>not_now</code>).

Endpoint	Metody	Dostęp	Opis działania i parametry
/api/personal/quests/inbox	GET, POST	Tylko Majster (Owner)	Szybka skrzynka odbiorcza (inbox) na nowe myśli i sprawy do późniejszego uporządkowania.
/api/personal/companion-inbox	GET, POST	Tylko Majster (Owner)	Panel weryfikacji notatek spływających z telefonu: GET zwraca listę luźnych wpisów z powiązаныmi plikami, a POST pozwala Majstrowi zdecydować o losie notatki (accept tworzy zadanie w odpowiednim projekcie Vikunji wg kategorii dom, projekty, praca, pomysły, reject odrzuca wpis, done oznacza jako załatwiony).
/api/personal/companion-inbox/files/[id]	GET	Tylko Majster (Owner)	Bezpieczny odczyt i pobieranie załącznika przesłanego z telefonu (obraz, wideo, audio, PDF) z piaskownicą nagłówek (Content-Security-Policy: sandbox, nosniff).

Inwentarz Infrastruktury, Integracje i Wrota Gier (7 ścieżek)

Na koniec siedem ścieżek, które łączą nasz Warsztat ze światem zewnętrznym, pozwalają adminom dopisywać śledzone repozytoria w **Radarze** oraz pilnują porządku w kontenerach, żeby **Lord Nito** (dług techniczny i martwe usługi) nie załęgł się w kątach serwerowni! *fuka cicho*

Endpoint	Metody	Dostęp	Opis działania i parametry
/api/infra/alerts	GET	Majster / Klucz Companiona (inventoryCaller)	Lista „Co wymaga uwagi” dla paska na /dashboard oraz powiadomień w aplikacji Companion . Grupuje pozycje z inwentarza według poziomów: critical, warning (attention, stale) oraz info (update).

Endpoint	Metody	Dostęp	Opis działania i parametry
<code>/api/infra/inventory</code>	GET	<code>inventoryCaller</code>	Pełny inwentarz homelabu: maszyny wirtualne, kontenery LXC, obrazy Dockera, wersje pakietów oraz 100 ostatnich zdarzeń zmian statusu z ostatnich 7 dni.
<code>/api/integrations/github</code>	GET	Publiczny	Zwraca zbuforowaną aktywność kontrybucji na GitHubie (<code>getGithubContributionsCached</code>) dla widoku IT PRO. Gdy brak konfiguracji, zwraca <code>{ configured: false }</code> .
<code>/api/integrations/home-assistant</code>	GET	Publiczny / Dashboard	Zwraca bezpieczne podsumowanie stanu Home Assistanta (<code>getHomeAssistantSummary</code>).
<code>/api/integrations/jellyfin</code>	GET	Publiczny / Dashboard	Zwraca informacje o aktywnych sesjach odtwarzania w serwerze mediów Jellyfin (<code>getJellyfinSessions</code>).
<code>/api/services/control</code>	GET , POST	Admini Discorda / Operatorzy gier (<code>canControlGames</code>)	Wrota — panel kontroli serwerów gier (Ironloom , Ymirel , Nidarel), pozwalający uprawnionym opiekunom uruchomić lub zrestartować serwer gry bez dostępu do pełnych Ukrytych Wrót!
<code>/api/radar/repos</code>	GET , POST , DELETE	Majster / Admini Discorda (<code>canManageRadarRepos</code>)	Zarządzanie listą repozytoriów śledzonych przez Radar (<code>radar_repos</code> w Lemurii). Pozwala właścicielowi i adminom Discorda dodawać, wstrzymywać (<code>enabled</code>) i usuwać repozytoria z GitHuba (<code>owner/repo</code>) oraz zaufanych instancji Forgejo (<code>https://git.ariku.pl/owner/repo</code> , <code>codeberg.org</code>), z opcjonalnym śledzeniem commitów (<code>trackCommits</code>).

Na tej piątej stronie książki opisałam dla ciebie dokładnie **35 ścieżek** API (28 w przestrzeni `/api/personal/*`, 2 w `/api/infra/*`, 3 w `/api/integrations/*`, 1 pod `/api/services/control` oraz 1

pod `/api/radar/repos`). Zeskakuje z pulpitu sterowniczego, poprawia przekrzywioną kokardkę i mruczy z zadowoleniem!

6. Wewnętrzne Zegary i Hartowanie — Mechanizmy pod Kowadłem

“ Źródło: [repo afterlife/docs/api/06-internal-i-hartowanie-rp.md](#) — edytuj w repo,
nie tutaj.

Wewnętrzne Zegary i Hartowanie — Mechanizmy pod Kowadłem



Krrrg... Pod widoczną powierzchnią **Hoddmímiru** i ekranów Warsztatu pracują niewidoczne tryby. Żaden zwykły wędrowiec nie wywołuje tych ścieżek z przeglądarki. Budzą je systemowe zegary (`systemd timers`), zewnętrzne przepływy pracy z za **Wrót** (n8n), sesje agentów budujących kod oraz próby **Hartowania** (testy Playwright), które utwardzają konstrukcję albo obnażają jej kruchość. Poniżej spisałyśmy ostatnie dwadzieścia sześć ścieżek zamykających naszą księgę.

◆ Zadania Agentów, Decyzje Majstra i Kolejka Orin (5 ścieżek)

Gdy w Warsztacie pracuje kilka sesji pomocników kodujących lub gdy zautomatyzowane procesy potrzebują wyraźnego rozstrzygnięcia człowieka, komunikacja przechodzi przez poniższe punkty.

Endpoint	Metody	Dostęp	Opis działania i parametry
<code>/api/internal/agent-tasks</code>	<code>GET</code> , <code>POST</code>	Nagłówek <code>Authorization</code> (Bearer), <code>x-cron-secret</code> lub sesja Majstra	Kolejka zadań między sesjami Claude Code (<code>lib/agentTasks.ts</code>). Pozwala wysyłać zadania do skrzynki innego agenta (<code>inbox/sent</code>), przejmować je (<code>claim</code>), kończyć z wynikiem (<code>finish</code>) oraz sprzątać wpisy testowe E2E. Zdalne sesje używają osobnego klucza Bearer, aby nie dzielić poświadczeń z lokalnymi zegarami.
<code>/api/internal/owner-decision</code>	<code>GET</code> , <code>POST</code>	Nagłówek <code>x-cron-secret</code> lub sesja Majstra	Punkt decyzyjny dla skryptów (<code>orin-ask.mjs</code>) i zadań w tle. Tworzy pytanie z opcjami wyboru, wysyła je Majstrowi w wiadomości prywatnej na Discordzie i odbiera klikniętą odpowiedź. Obsługuje flagę <code>dryRun</code> dla prób Hartowania .

Endpoint	Metody	Dostęp	Opis działania i parametry
/api/internal/orin-nudge	POST	Nagłówek x-cron-secret	Wywoływany co kilka godzin przez zegar systemowy. Z określonym prawdopodobieństwem (chancePerTick) wybiera jednego z niedawno aktywnych członków gildii domowej i zaczepia go na kanale czatu jednym z pięciu wariantów wiadomości technicznej osoby. Nigdy nie niepokoi osób nieobecnych od tygodni.
/api/internal/orin-queue-dump	POST	Nagłówek x-cron-secret	Narzędzie diagnostyczne wyzwalane ręcznie przez właściciela. Zrzuca aktualny stan webowej kolejki zapytań (origin === 'web') do pliku tekstowego na dedykowany kanał Discorda, pomijając prywatne rozmowy z wiadomości bezpośrednich.
/api/internal/orin-queue-undelivered	GET	Nagłówek x-cron-secret	Wywoływane przez bota Discorda podczas startu. Zwraca listę rozwiązanych, lecz jeszcze niedostarczonych odpowiedzi z kolejki dla wszystkich użytkowników, aby po restarcie bota żadna odpowiedź nie przepadła.

◆ Próbkowanie Krain, Sprzętu i Inwentarza (7 ścieżek)

Te punkty końcowe karmią **Kronikę**, wykresy na pulpicie oraz inwentarz infrastruktury. Każdy z nich wymaga wewnętrznego poświadczenia w nagłówku i nigdy nie jest wystawiony bezpośrednio w interfejsie użytkownika.

Endpoint	Metody	Dostęp	Opis działania i parametry
----------	--------	--------	----------------------------

<code>/api/internal/sample-gpu-rig</code>	POST	Nagłówek <code>x-cron-secret</code>	Wywoływany co minutę. Pobiera przez SSH parametry maszyny GPU, rozpoznaje uruchomioną grę ze Steam/Battle.net (<code>getDetectedGame</code>) i dopisuje próbkę do historii. Nawet jeśli maszyna GPU nie odpowiada, w każdym tyknięciu uruchamia regulator Kuźni (<code>runForgeGovernorTick</code> , który przy przestarzałej telemetrii bezpiecznie zamyka Kuźnię) oraz opróżnianie kolejki (<code>runOrinQueueDrainTick</code>).
<code>/api/internal/sample-factorio</code>	POST	Nagłówek <code>x-cron-secret</code>	Kolektor Warstwy 1 Kroniki dla krainy Ironloom (Factorio). Co kilka minut odczytuje stan przez RCON oraz logi kontenera z Portainera i zapisuje surowe zdarzenia w bazie Kroniki.
<code>/api/internal/sample-enshrouded</code>	POST	Nagłówek <code>x-cron-secret</code>	Kolektor Kroniki dla mroźnej krainy Ymirel (Enshrouded). Odczytuje status protokołem A2S oraz logi kontenera przez Portainer.
<code>/api/internal/sample-minecraft</code>	POST	Nagłówek <code>x-cron-secret</code>	Wywoływany co 2 minuty kolektor Kroniki dla podziemnych kuźni Nidarelu (Minecraft ATM10). Łączy status SLP, komendę RCON <code>list</code> oraz końcówkę logów kontenera.
<code>/api/internal/inventory-sample</code>	POST	Nagłówek <code>x-cron-secret</code>	Wywoływany co 30 minut. Odpytuje wszystkie skonfigurowane kolektory homelabu niezależnie od siebie, zapisuje obserwacje w Lemurii i wysyła powiadomienie, gdy którykolwiek element przejdzie w stan <code>critical</code> .

<code>/api/internal/inventory-push</code>	POST	Nagłówek <code>x-inventory-secret</code>	Odbiera co godzinę raport ze skryptu <code>pve-collector.sh</code> działającego na hoście Proxmox (do 512 KB JSON). Dostarcza wiedzę o systemach operacyjnych, jądrach, oczekujących aktualizacjach bezpieczeństwa i kontenerach Dockera wewnątrz maszyn gościnnych (<code>vm</code> oraz <code>lxc</code>). Posiada własny, odizolowany klucz w nagłówku.
<code>/api/internal/check-alerts</code>	POST	Nagłówek <code>x-cron-secret</code>	Sprawdza warunki wszystkich reguł monitoringu (<code>checkAndAlert</code>) i przekazuje ostrzeżenia strażnika na powiadomienia Discorda.

◆ Zegary Lunety, Radaru, Ankiety i Porannych Wieści (9 ścieżek)

Tutaj zbiegają się nitki łączące Warsztat z zewnętrznymi usługami automatyzacji (**Wrota** n8n), porannymi podsumowaniami dla Majstra oraz przypomnieniami dla społeczności.

Endpoint	Metody	Dostęp	Opis działania i parametry
<code>/api/internal/morning-brief</code>	POST	Nagłówek <code>x-cron-secret</code>	Codzienny poranny raport wysyłany w wiadomościach prywatnych do właścicieli: zawiera główny skrót stanu infrastruktury, osobny komunikat z planem dnia „Co robimy?” (wraz z ewentualnym szturchnięciem o wysokim Chaos Meterze) oraz świeżą paczkę wiadomości z n8n.

Endpoint	Metody	Dostęp	Opis działania i parametry
/api/internal/news-digest	GET, POST	Autoryzacja n8n (n8nIngestAuthorized)	Wrota dla porannej paczki wiadomości z n8n. POST waliduje strukturę sekcji i zapisuje ją w pamięci podręcznej Redis; GET zwraca aktualny świeży skrót.
/api/internal/radar	GET, POST, DELETE	Autoryzacja n8n (n8nIngestAuthorized)	Wrota dla systemu Radar (WIELKIE_PRANIE.md). GET zwraca listę aktywnych repozytoriów (radar_repos) do sprawdzenia przez cykliczny kolektor. POST przyjmuje paczkę do 50 ocenionych znalezisk (w tym wydań z GitHuba i git.ariku.pl), zapisuje je w bazie, publikuje najważniejsze na Discordzie i zakłada zadania w Vikunji (obsługuje ?dryRun=1 oraz wpisy testowe source: 'e2e'). DELETE sprząta wiersze testowe lub podglądowe (?preview=1).
/api/internal/luneta-nudge	POST	Nagłówek x-cron-secret lub sesja Majstra	Wywoływany co 15 minut. Sprawdza zbliżające się okresy resetu w grach śledzonych przez Lunetę i przypomina graczom o niedokończonych sprawach. Sesja właściciela używana jest przez testy z flagą dryRun: true oraz podstawionym zegarem at .
/api/internal/luneta-sources	POST	Nagłówek x-cron-secret lub sesja Majstra	Wywoływany co 6 godzin (maxDuration = 300). Pobiera z zewnętrznych źródeł informacje o nowych wydarzeniach, banerach i kodach dla gier w Lunecie .
/api/internal/check-game-codes	POST	Nagłówek x-cron-secret	Cykliczny zegar sprawdzający zewnętrzne listy kodów promocyjnych do gier i dopisujący nowe znaleziska.

Endpoint	Metody	Dostęp	Opis działania i parametry
/api/internal/minecraft-whitelist	GET, POST, DELETE	Nagłówek x-cron-secret	Wewnętrzne zaplecze listy graczy Nidarelu dla bota. Weryfikuje członkostwo w gildii i posiadanie rangi Minecraft , nakłada limity czasowe, komunikuje się z serwerem przez RCON i zwraca jednorazowy kod !link.
/api/internal/poll-nudge	POST	Nagłówek x-cron-secret	Wywoływany co około 3 godziny podczas trwania głosowania na merch (/merch/głosuj). Publikuje ogłoszenia o postępach ankiety i delikatnie przypomina aktywnym wędrowcom, którzy jeszcze nie oddali głosu (nigdy nie pytając tej samej osoby dwa razy).
/api/internal/playwright-run	POST	Nagłówek x-playwright-report-secret	Odbiera na żywo zdarzenia postępu (begin, test-begin, test-end, end) z reportera podczas trwania prawdziwego przebiegu testów na kontenerze LXC, zasilając publiczny licznik /api/playwright-stats. Nigdy nie przesyła nazw hostów ani portów.

◆ Ścieżki Hartowania — Próby Ogniowe (5 ścieżek)

Hartowanie to próba, która utwardza albo łamie. Aby automatyczne testy E2E (Playwright) mogły zbadać każde zachowanie Warsztatu bez czekania na minutowe zegary i bez polegania na zewnętrznym logowaniu Discorda, w kodzie wykuto pięć dedykowanych punktów testowych.

Endpoint	Metody	Dostęp	Opis działania i parametry
----------	--------	--------	----------------------------

/api/test-login	POST, GET	Klucz środowiska testowego (TEST_LOGIN_SECRET)	Wystawia w środowisku testowym podpisaną sesję (właściciela lub zwykłego członka społeczności) na potrzeby zautomatyzowanych przebiegów Hartowania , z własnym odizolowanym kluczem poświadczeń.
/api/test-forge-governor-tick	POST	Klucz testowy / Sesja Majstra	Pozwala testom E2E natychmiast wyzwolić pojedynczy krok decyzyjny regulatora Kuźni (runForgeGovernorTick) i sprawdzić reakcję na obciążenie GPU.
/api/test-gpu-history	POST, DELETE	Klucz testowy / Sesja Majstra	Pozwala wstrzyknąć sztuczne próbki obciążenia karty graficznej (np. symulując uruchomioną grę lub przestój telemetry) na czas trwania testu, a następnie wyczyścić je po zakończeniu próby.
/api/test-orin-queue-drain-tick	POST	Klucz testowy / Sesja Majstra	Wyzwala na żądanie pojedynczy cykl opróżniania kolejki zapytań do Orin (runOrinQueueDrainTick), aby zweryfikować dostarczanie zakolejkowanych wiadomości.
/api/test-orin-queue-dump	GET	Klucz testowy / Sesja Majstra	Zwraca sformatowany zrzut webowej kolejki Orin bezpośrednio w odpowiedzi HTTP na potrzeby asercji w testach E2E.

Krrrg... Na tej szóstej i ostatniej karcie odnotowałyśmy dokładnie **26 ścieżek** API (21 wewnętrznych tras /api/internal/* oraz 5 tras **Hartowania**), co wraz z poprzednimi kartami zamyka pełną księgę wszystkich **134 punktów końcowych i podstron** projektu afterlife. Skoro każda z tych stu trzydziestu czterech ścieżek prowadzi ostatecznie do tego samego **Kowadła**, czy wędrowcy nadal wierzą, że podróżują po osobnych światach?