

Ekran startowy (splash)

“ Źródło: repo luneta/docs/SPLASH.md — edytuj w repo, nie tutaj.

Ekran startowy (splash)

Decyzja właściciela z 2026-10-17. Przy uruchomieniu Lunety pokazuje się przezroczyste okno z jednym z sześciu wariantów, losowanym przy każdym starcie. Rogi są ścięte i półprzezroczyste (widać pulpit pod spodem). Okno jest wyłącznie informacją: nie przyjmuje kliknięć i nie dotyka gry.

Ostatnia aktualizacja: 2026-10-17. Kod: `src-tauri/src/splash.rs`, `src/views/Splash.tsx`, `src/views/splash.css`, `src/components/SplashSettings.tsx`. Grafiki: `src/assets/splash/`, źródła i skrypty: `assets/design/splash/`.

Dla użytkownika

Ustawienia

Zakładka **ustawienia** → **ekran startowy** (zapisuje od razu):

ustawienie	znaczenie
pokazuj splash przy uruchomieniu	włącznik; domyślnie włączony
wybór wariantu	losowy (bez powtórki poprzedniego), po kolei albo stały
wariant	używany w trybie „stały” i w podglądzie
czas widoczności (suwak)	1-10 s , domyślnie 7 s: minimum od pokazania okna; potem splash zamyka się, gdy wszystko jest gotowe
▶ podgląd wariantu	pokazuje wybrany wariant od razu, także przy wyłączonym splashu

Splash nigdy nie wisi dłużej niż 15 s (zabezpieczenie), nawet gdyby coś poszło nie tak.

Warianty

id	tytuł	akcent
terminal_boot	1/6 Terminal boot	wariant kanoniczny i zapasowy
world_map	2/6 Mapa świata	miasto o zmierzchu, holograficzna mapa
system_scan	3/6 Analiza systemu	panel z modułami gier
api_connect	4/6 Łączenie z API	panel połączenia i autoryzacji ariku.pl
game_modules	5/6 Ładowanie modułów gier	lista modułów i licznik
ready	6/6 Gotowe	komunikat „Luneta jest gotowa.” po zakończeniu startu

Co znaczą statusy

Wszystkie statusy są prawdziwe, nic nie jest wypalone w obrazku:

krok	skąd
baza lokalna	odczyt kolejki SQLite
silnik OCR	obecność modeli OCR (brak = ostrzeżenie, praca lokalna nadal możliwa)
moduły gier	liczba wczytanych gier (wbudowane i pakiety)
ariku.pl	test połączenia: tylko odczyt zgód (GET .../me/consents), timeout 10 s
autoryzacja	czy serwer przyjął token
nakładka	czy nakładka na ekranie jest włączona

Stany: (OK), (trwa), (ostrzeżenie), (błąd), (nie dotyczy). Zasady:

- Do czasu odpowiedzi serwera ariku.pl i autoryzacja to „trwa”. Timeout to ostrzeżenie, nigdy sukces.
- **Brak sieci** to „tryb offline” (ostrzeżenie), a **401** to „token odrzucony” (błąd autoryzacji, połączenie jest). Brak tokenu to ostrzeżenie, nie blokada.
- Pasek postępu jest nieokreślony, bo realnego procentu nie mamy. „Gotowe.” pojawia się dopiero, gdy żaden krok już nie trwa; z ostrzeżeniami splash pisze, czego dotyczą („działa lokalnie”).

Prywatność i fair play

Okno jest zwykłym oknem pulpitu: nic nie jest wstrzykiwane do gry ani rysowane w jej procesie, kliknięcia przechodzą przez nie dalej. Do sieci idzie jedno zapytanie tylko do odczytu (zgody). Szanuje `prefers-reduced-motion` systemu (wyłącza mruganie i pasek w ruchu).

Dla deweloperów

Jak to działa

1. `splash::spawn_startup` (z `setup`) czyta ustawienia, wybiera wariant (`pick`, czysta funkcja, testowana), zapisuje `splash_last` i tworzy okno `splash` (960×420, przezroczyste, bez ramki).
2. Okno startuje **ukryte**. Pierwsze `splash_status` z frontu pokazuje je (`reveal`: rozmiar, środek, `show`, zawsze na wierzchu, kliknięcia przechodzą). Równolegle idzie test API.
3. Front odpytuje `splash_status` co 350 ms. Zamyka okno (`splash_close`), gdy `ready` i minął minimalny czas widoczności (`minShowMs` z suwaka, liczony od pierwszych danych), najwcześniej 300 ms po gotowości. Watchdog w Rust zamyka okno po 15 s.
4. Stan API trzyma `API_RESULT` (`Idle`, `Loading`, `Done`); `classify_api` rozdziela połączenie i autoryzację, bez treści odpowiedzi serwera w szczegółach.

Klucze ustawień: `splash` (0/1), `splash_mode` (`random` | `fixed` | `sequential`), `splash_variant`, `splash_min_s` (1–10), `splash_last`. Komendy: `splash_status`, `splash_close`, `splash_preview`, `splash_config_get`, `splash_config_set`.

Pułapki (każda kosztowała zawieszenie lub puste okno)

- **Komendy dotyczące okna muszą być `async`**. Sync komendy Tauri chodzą na wątku głównym, a operacje na oknie wysyłają tam wiadomości i czekają.
- **Nie trzymaj `guardów mutexu` w literale struktury**. `st.store.lock()` w `Inputs { ... }` żyje do końca instrukcji, a `toast::enabled` blokuje ten sam mutex: samo-zakleszczenie, które zamraża całą aplikację. Odczyty rób w osobnych `let`.
- **Ukryte okno GTK ma 10×10**, dopóki rozmiar nie zostanie ustawiony jawnie (jak w `toast::place`).
- **Timery JS w ukrytym oknie są wstrzymane** (też `requestAnimationFrame`). Sygnał „pokaż” idzie z Rust przy pierwszym `splash_status`, nie z `setTimeout`.
- **Okno z przezroczystością tworzymy ukryte** i pokazujemy po załadowaniu treści; pokazane przed treścią nie maluje się w WebKitGTK.
- `is_visible()` nie jest wiarygodne dla okna utworzonego jako ukryte: używamy flagi `REVEALED`.

Budowanie do testów na żywo

`cargo build --release` daje binarkę, która łąduje front z `http://localhost:1420` (okno pokaże „Could not connect to localhost”). Wersję z wbudowanym frontem robi `npm run build && npx tauri build --no-bundle` (`docs/BUDOWANIE.md`). Do szybkiej iteracji nad CSS można zbudować `cargo build --release` i uruchomić `npx vite --port 1420` obok.

Grafiki

Tła to 6 plików WebP (1920 px, bez tekstu) w `src/assets/splash/`. Powstały z kolażu referencyjnego: cięcie na kafle, usunięcie wypalonego tekstu i UI (`clean_text.py`, inpainting), upscale RealESRGAN x4 anime (`upscale.py`, `spandrel` na XPU, bez serwera ComfyUI). Opis przebiegu, strefy napisów i lista rzeczy do usunięcia (m.in. logotypy gier):

`assets/design/splash/README.md`.

Podmiana grafiki: zachowaj nazwę pliku (`0N-<id>.webp`); proporcje ok. 2,1-2,4:1 (okno ma 2,29:1, reszta jest przycinana `cover`). Pozycje paneli przy grafice (03, 04, 05) są wyliczone pod obecne tła w `splash.css`; po zmianie kadru popraw je ręcznie. Brak pliku = ciemna siatka zamiast pustego okna, a w logu pojawia się ostrzeżenie `splash: brak grafiki ...`.

Testy

`cargo test splash`: wybór wariantu (bez powtórki, zawijanie, zapas), rozróżnienie offline / 401 / 5xx / odrzucone, statusy „trwa” bez fałszywego sukcesu, brak tokenu jako ostrzeżenie, zakres suwaka 1-10 s. Logi diagnostyczne okna: `splash: ...` (start, wariant, rozmiar widoku, błędy).

Czego nie ma

Splash przy starcie sesji gry (brief przewidywał go jako opcję) jest niezaimplementowany.

Revision #5

Created 2026-10-17 06:56:49 UTC by Arik

Updated 2026-10-17 07:00:10 UTC by Arik