

Booksy Multi-Tenant Booking API

Specyfikacja Techniczna dla Deweloperów & Agentów AI

WERSJA 1.0 (2026)

Dokumentacja Integracji API & Headless Booking

Wdrożenie rezerwacji terminów online na dowolnej stronie WWW (WordPress, Webflow, React, HTML/JS) z zachowaniem pełnej spójności wizualnej i synchronizacją 2-way z Booksy.

 **Ochrona Anti-Bot & Abuse:** Wszystkie endpointy publiczne posiadają zintegrowany filtr Honeypot, podpisany kryptograficznie token sesji formularza (Time-to-Submit heuristic), Rate Limiter (IP + Telefon) oraz obsługę Cloudflare Turnstile.

1. Informacje Ogólne & Adresy Bazowe

Środowisko	Adres URL	Autoryzacja
Publiczne API WWW	https://booking.host67.pl/api/public/salon/{slug}	Brak (Ochrona Anti-Bot)
Integracja SmartStaff	https://booking.host67.pl/api/v1/integration	Nagłówek X-API-Key
Gotowy Widget JS	https://booking.host67.pl/widget.js	Atrybut data-salon

2. Publiczne Endpointy REST API (Headless UI)

A. Pobranie Konfiguracji i Tokenu Anty-Bot

GET /api/public/salon/{slug}/config

Zwraca metadane salonu, harmonogram godzin otwarcia oraz świeży, podpisany form_session_token (wymagany przy rezerwacji).

```
{
  "salon_name": "Centrum Stopy i Ruchu",
  "slug": "centrum-stop-y-i-ruchu",
  "open_hours": [
    { "day_of_week": 1, "hour_from": "09:00", "hour_till": "19:00" },
    { "day_of_week": 2, "hour_from": "09:00", "hour_till": "19:00" }
  ],
  "cf_turnstile_sitekey": "1x000000000000000000AA",
  "form_session_token": "eyJhbGciOiJIUzI1NiIs..."
}
```

B. Katalog Usług Salonu

GET /api/public/salon/{slug}/services

Zwraca listę aktywnych usług, ceny, czas trwania w minutach oraz listę uprawnionych pracowników.

```
[
  {
    "id": 12,
    "booksy_id": "7910799",
    "name": "Konsultacja podologiczna",
    "price": 150.0,
    "currency": "zł",
    "duration_minutes": 45,
    "staff_members": [
      { "id": 5, "booksy_id": "846658", "name": "Kamil Pływaczyk" }
    ]
  }
]
```

C. Lista Personelu

GET /api/public/salon/{slug}/staff

Zwraca listę specjalistów przyjmujących w salonie (id , booksy_id , name).

D. Wyliczanie Dostępnych Slotów (Anty-kolizja Booksy)

GET /api/public/salon/{slug}/slots?service_id={id}&date={YYYY-MM-DD}&staff_id={opcjonalne}

Wylicza wolne terminy w oparciu o grafik pracy salonu, uprawnienia pracownika i istniejące rezerwacje w Booksy.

```
{
  "date": "2026-08-25",
  "service_name": "Konsultacja podologiczna",
  "duration_minutes": 45,
  "is_closed": false,
  "slots": [
    {
      "time": "10:00",
      "datetime": "2026-08-25T10:00:00+00:00",
      "available_staff_id": "846658",
      "available_staff_name": "Kamil Pływaczyk"
    },
    {
      "time": "10:45",
      "datetime": "2026-08-25T10:45:00+00:00",
      "available_staff_id": "846658",
      "available_staff_name": "Kamil Pływaczyk"
    }
  ]
}
```

E. Utworzenie Rezerwacji (2-Way Booksy Sync)

POST /api/public/salon/{slug}/book

Rezerwuje wizytę, automatycznie zakłada profil klienta w Booksy (po numerze telefonu), weryfikuje tokeny anty-bot i zwraca potwierdzenie.

Request Body (JSON):

```
{
  "service_id": "7910799",
  "staff_id": "846658",
  "start_time": "2026-08-25T10:00:00Z",
  "client_name": "Anna Nowak",
  "client_phone": "+48500100200",
  "client_email": "anna.nowak@example.com",
  "notes": "Pierwsza wizyta",
  "form_session_token": "eyJhbGciOiJIUzI1NiIs... ",
  "_hp_check": ""
}
```

Response (HTTP 200 OK):

```
{
  "success": true,
  "booking_id": "res_8492041",
  "service_name": "Konsultacja podologiczna",
  "staff_name": "Kamil Pływaczyk",
  "start_time": "2026-08-25T10:00:00Z",
  "end_time": "2026-08-25T10:45:00Z",
  "client_name": "Anna Nowak",
  "client_phone": "+48500100200",
  "message": "Rezerwacja została pomyślnie utworzona i potwierdzona."
}
```

3. Wzorcowy Kod JavaScript (Wklejenie na stronę WWW)

```
const SALON = 'centrum-stopy-i-ruchu';
const BASE = `https://booking.host67.pl/api/public/salon/${SALON}`;
let sessionToken = null;

async function init() {
  const cfg = await (await fetch(`${BASE}/config`)).json();
  sessionToken = cfg.form_session_token;
  const services = await (await fetch(`${BASE}/services`)).json();
  renderServices(services);
}

async function bookAppointment(serviceId, staffId, startTimeIso, name, phone, email, notes) {
  const res = await fetch(`${BASE}/book`, {
    method: 'POST',
    headers: { 'Content-Type': 'application/json' },
    body: JSON.stringify({
      service_id: serviceId,
      staff_id: staffId || null,
      start_time: startTimeIso,
      client_name: name,
      client_phone: phone,
      client_email: email || null,
      notes: notes || null,
      form_session_token: sessionToken,
      _hp_check: document.getElementById('hp_field')?.value || ''
    })
  });
  return await res.json();
}
```