

Twilio Chat App — Kompleksowa dokumentacja

TC

Opis produktu, architektura, scenariusze biznesowe & wskazówki wdrożeniowe

ver3.1.0 •
2025-12-13

Przegląd rozwiązania

Twilio Chat App to dedykowane rozwiązanie dla firm, które potrzebują bezpiecznego, kontrolowanego kanału komunikacji SMS/MMS z możliwością automatyzacji i integracji z modelami AI. Produkt łączy prostotę obsługi z polityką audytowalności i pełnym zapisem danych.

Panel operatora

Intuicyjny interfejs dla zespołów obsługi

Auto reply & AI

Tryby automatycznych odpowiedzi z logiką biznesową

REST API

Integracja z systemami CRM i narzędziami zewnętrznymi

SQLite

Łatwa persystencja i prosty backup plikowy

ver3.1.0

Flask + Twilio

Workery w tle

FAISS / RAG

Multi SMS

Kluczowe funkcje

- Webhooki Twilio: przyjmowanie inbound i status callbacks
- Auto reply (klasyczny) + AI auto reply: tryby współdzielą jeden magazyn danych
- Przypomnienia: scheduler cykliczny do powiadomień
- Multi SMS: jednorazowe kampanie batchowe z historią odbiorców
- News / FAISS / RAG: scraping, indeksowanie i broadcast streszczeń

The screenshot shows the Twilio Chat application interface. At the top, there's a navigation bar with 'Twilio Chat' on the left and 'Panel' and 'Twilio Console' on the right. A 'Środowisko: DEV' button is also present. Below the navigation, the title 'Panel czatu' is displayed, followed by a subtitle 'Zarządzaj rozmowami SMS z jednego miejsca.' There are several tabs at the top: 'Wiadomości' (selected), 'Auto-odpowiedź', 'Przypomnienia', 'AI', 'News', and 'Multi-SMS'. On the left, a sidebar shows 'WSZYSTKIE WIADOMOŚCI' with a count of '55' and a note 'Łączna liczba wiadomości w bazie.'. In the center, two boxes show 'PRZYCHODZĄCE' with a count of '14' and 'WYCHODZĄCE' with a count of '41'. The main area displays a table of messages with columns: 'Kierunek', 'Odbiorca / Nadawca', 'Treść', 'Status', 'Akcja', 'Czat', and 'Czas'. The table lists several messages, mostly outgoing ('WYCHODZĄCA') to numbers like '+48880209541' with various messages like 'aaaaaa', 'elorado', 'elo', and news updates. A purple button labeled 'Wyślij wiadomość' is located at the bottom left of the message input area.

Widok wiadomości w panelu — jeden ekran do nadzoru, wysyłki i wglądu w historię

Co widzisz

- Historię wiadomości (kierunek inbound/outbound, numery, skrót treści).
- Statusy dostarczenia i znaczniki czasu.
- Szybką nawigację do widoku czatu dla konkretnego numeru.

Co możesz zrobić

- Przejść do rozmowy i kontynuować chat z klientem.
- Weryfikować skuteczność wysyłek po statusach.
- Operacyjnie diagnozować błędy (np. numer, treść, status callback).

Co dzieje się w tle

- Panel pobiera dane przez `GET /api/messages`.
- Statusy są aktualizowane callbackiem `POST /twilio/status`.
- Wszystko jest audytowane w SQLite w tabeli `messages`.

Architektura techniczna

Modularna aplikacja Flask z dedykowanymi modułami integracji i workerami. Poniżej zwięzłe mapowanie komponentów.

Komponent	Rola
`webhooks_bp`	HTTP webhooki Twilio i API
`ai_service`	Generowanie odpowiedzi AI i budowa kontekstu
`multi_sms.py`	Worker batchowy dla kampanii Multi SMS i rezerwacja zadań
`twilio_client`	Warstwa wysyłki wiadomości i zarządzania wątkami
`database.py`	SQLite, migracje, helpery biznesowe

Panel WWW — UX i scenariusze użycia

Zaprojektowany pod kątem wydajności operatora: czytelne karty, inteligentne filtry i klikalne wiersze historii upraszczają pracę.

Historia wiadomości

Jednowierszowy podgląd treści, meta kanału i rola (od/do klienta), statusy oraz czas rozbity na godzinę i datę.

The screenshot shows the Twilio Chat WWW panel. At the top, there are tabs: 'Wiadomość' (selected), 'Auto-odpowiedź', 'Przypomnienia', 'AI', 'News', and 'Multi-SMS'. Below the tabs, there are two cards: 'WSZYSTKIE WIADOMOŚCI' with the number '55' and 'Ołączna liczba wiadomości w bazie.', and 'PRZYCHODZĄCE' with the number '14' and 'Otrzymane od klientów.'. On the right, there is a card for 'WYCHODZĄCE' with the number '41' and 'Wysiane z aplikacji.'. Below these cards is a section titled 'Historia wiadomości' with a sub-note 'Ostatnie 50 pozycji (auto-odświeżanie co 15 sek.)'. It includes a table with columns: Kierunek, Odbiorca / Nadawca, Treść, Status, Akcje, Czat, and Czas. The table lists several messages, such as 'WYCHODZĄCA +48880209541 aaaaa delivered Usuń Otwórz 20:09', 'WYCHODZĄCA +48880209541 eldorado delivered Usuń Otwórz 20:09', and 'WYCHODZĄCA +48880209541 sss delivered Usuń Otwórz 19:46'. At the bottom left, there is a form for sending a new message with fields for 'Numer odbiorcy' (np. +48123456789) and 'Treść wiadomości' (Wpisz wiadomość...), and a large red button 'Wyślij wiadomość'.

Przykładowy wiersz z meta kanału oraz przyciskiem szybkiej nawigacji do czatu

Co widzisz

- Wiersze z wiadomościami i metadanymi (od/do, treść, status, data/godzina).
- Rozróżnienie inbound vs outbound (ułatwia kontrolę reakcji).

Możliwe interakcje

- Wejście do konwersacji klienta i dalsza obsługa w trybie czatu.
- Kontrola statusu dostarczenia i szybkie wykrywanie błędów.
- Praca „na historii” — obsługa wielu klientów bez utraty kontekstu.

Co się zapisuje

- Każda wiadomość jest przechowywana w SQLite wraz z SID i statusem.
- System normalizuje numery, aby spinać wątki i raportowanie.

Auto odpowiedź

Włączenie auto odpowiedzi to szybki sposób na obsłużenie powtarzalnych zapytań. Operator ustala treść i może w dowolnym momencie włączyć/wyłączyć funkcję bez restartu systemu.

The screenshot shows the Twilio Chat console interface. At the top, there's a navigation bar with 'Twilio Chat' and 'Panel - Twilio Console'. On the right, it says 'Środowisko: DEV'. Below the navigation, there's a header 'Panel czatu' with the sub-instruction 'Zarządzaj rozmowami SMS z jednego miejsca.' Underneath, there are several tabs: 'Wiadomości' (selected), 'Auto-odpowiedź' (highlighted in blue), 'Przypomnienia', 'AI', 'News', and 'Multi-SMS'. The main content area is titled 'AUTO-ODPOWIEDŹ SMS' and 'Ustaw wiadomość automatyczną'. It contains a toggle switch labeled 'Włącz automatyczne odpowiadanie na nowe wiadomości' which is turned on. Below the switch is a text input field labeled 'Treść wysyłanej wiadomości' containing the placeholder 'Np. Dziękujemy za wiadomość! Wracamy z odpowiedzią wkrótce.' At the bottom of this section is a button 'Zapisz ustawienia'. To the right of this main section, there's a sidebar titled 'Jak to działa?' (How does it work?) with the following points:

- Aplikacja zapisuje przychodzące wiadomości i wysyła SMS zwrotny, gdy funkcja jest włączona.
- Wiadomość jest wysyłana z numeru domyślnego lub Messaging Service (jeśli skonfigurowano).
- Kolejka w tle chroni przed duplikatami i obsługuje maks. 1000 ostatnich SID.
- Wyłączenie auto-odpowiedzi przywraca tryb odpowiedzi generowanej przez silnik czatu.

Przełącznik ON/OFF i pole treści — gotowe do natychmiastowego użycia.

Co widzisz

- Stan aktywacji (ON/OFF) oraz aktualną treść automatycznej odpowiedzi.
- Ustawienia działające natychmiast po zapisaniu.

Możliwe interakcje

- Włączenie/wyłączenie auto reply dla całego systemu.
- Zmiana treści odpowiedzi (np. informacja o godzinach pracy, FAQ).

AI auto reply może przejąć sterowanie, gdy jest aktywne.

Co dzieje się w tle

- Inbound webhook uruchamia logikę enqueue (bez blokowania requestu).
- System odpowiada tylko na wiadomości nowsze niż moment aktywacji (bez „retro-odpowiedzi”).
- Wysyłka realizowana przez Twilio z pełnym logiem w [messages](#).

Przypomnienia

Przypomnienia umożliwiają cykliczną wysyłkę do wybranych numerów (np. powiadomienia, statusy, nudges). Worker w tle egzekwuje harmonogram i zapisuje wyniki.

The screenshot shows the Twilio Chat console interface. At the top, there's a navigation bar with tabs: 'Panel czatu', 'Auto-odpowiedź', 'Przypomnienia' (which is currently selected), 'AI', 'News', and 'Multi-SMS'. Below the navigation, there are two main sections. On the left, under 'NOWE PRZYPOMNIENIE SMS', there's a form to 'Wyslij cyklicznie' (Send periodically) to a number '+48123456789'. It includes fields for 'Treść wiadomości' (Message content) and 'Interwał (minuty)' (Interval (minutes)) set to 60. A red 'Zapisz przypomnienie' (Save reminder) button is at the bottom. On the right, there's a table titled 'PRZYPOMNIENIA Aktywne i zaplanowane' (Active and scheduled reminders) with columns: Numer (Number), Treść (Content), Interwał (Interval), Status (Status), and Akcje (Actions). The table shows a single entry: 'Brak danych.' (No data).

Planowanie i podgląd aktywnych przypomnień w panelu.

Co widzisz

- Listę zadań cyklicznych (numer, treść, interwał, status włączenia).
- Informacje potrzebne do nadzoru (czy działa, od kiedy, do kogo).

Możliwe interakcje

- Dodanie nowego przypomnienia (treść + interwał).
- Wstrzymanie/wznowienie oraz usunięcie harmonogramu.

Co dzieje się w tle

- Scheduler sprawdza due tasks i wysyła SMS przez Twilio.
- Każda wysyłka jest zapisywana w historii, a błędy są logowane.
- Operacje wykonywane są przez API `/api/reminders`.

Multi SMS — wysyłka jednej wiadomości do wielu odbiorców

Multi SMS umożliwia operatorowi wklejenie listy numerów (dowolny format), utworzenie zadania batchowego i śledzenie postępu wysyłki realizowanej w tle.

Jak to działa (w praktyce)

1. Operator wkleja numery (przecinki / średniki / nowe linie) i treść wiadomości.
2. Backend normalizuje numery i zapisuje batch + rekordy odbiorców w SQLite.
3. Worker w tle rezerwuje najbliższy batch **pending** i przetwarza odbiorców.
4. Każda próba wysyłki aktualizuje status odbiorcy oraz liczniki batcha (sukcesy/błędy).

The screenshot shows the Twilio Chat application interface. At the top, there's a purple header bar with the text "Panel czatu" and "Multi-SMS". Below the header, the main area is divided into two sections. On the left, there's a form for sending multiple messages. It includes fields for "Numery odbiorców" (with an example value "np. +48123123123\n+441234567890") and "Treść wiadomości" (with an example value "Witaj w naszej aplikacji!"). A note below the number field says: "Wpisz każdy numer w osobnej linii lub oddziel przecinkiem; duplikaty usuwamy automatycznie." At the bottom of this section is a red button labeled "Zainicjuj wysyłkę". On the right, there's a "HISTORIA WYSYŁEK" section titled "Multi-SMS (ostatnie 20)". It shows a table with one row for "eldorado" with the status "zakończone" and the date "13.12.2025, 20:09:01". Below the table, there are buttons for "Liczne 1", "Wybrane 1", "Oczekujące 0", "Błędy 0", and "Nieprawidłowe 0".

Batch w panelu: status, liczniki sukcesów/błędów oraz rozwijana lista odbiorców.

Co widzisz

- Formularz utworzenia batcha (treść + lista numerów).
- Historię batchy z licznikami: wysłane / błędy / łącznie oraz status batcha.
- Możliwość rozwinięcia listy odbiorców i podejrzenia statusów per numer.

Możliwe interakcje

- Wklejenie numerów w różnych formatach (przecinki/linie) — system je rozbije, oczyści i znormalizuje.
- Utworzenie batcha bez czekania na wysyłkę (wysyłka idzie w tle).
- Kontrola postępu: odświeżanie historii batchy i rozwijanie odbiorców dla wglądu w statusy.
- Diagnostyka błędów: podejrzenie *error* per numer (np. niepoprawny format, błąd Twilio, brak routingu).

Co dzieje się w tle

- Backend tworzy batch i odbiorców w SQLite (kolejka DB oparta o statusy).
- Worker rezerwuje batch do przetworzenia, wysyła SMS per odbiorca i aktualizuje statusy w tabelach `multi_sms_*`.
- Każda próba wysyłki jest zapisywana w historii (`messages`), co daje pełny audyt.
- UI pobiera batchy przez `GET /api/multi-sms/batches` (opcjonalnie wraz z listą odbiorców).

Kontrola ryzyka

- **Limit odbiorców na batch:** 250 (ochrona kosztów i przypadkowej wysyłki masowej).
- **Odporność na restart:** stan utrzymywany w bazie, worker po restarcie kontynuuje kolejkę.
- **Audyt:** każda wysyłka trafia do historii wiadomości oraz ma wynik per odbiorca.

News / FAISS (RAG) — scraping, indeksowanie i dystrybucja

Moduł News automatyzuje pobieranie treści (scraping), budowę indeksu FAISS oraz generowanie streszczeń (RAG). Panel pozwala zarządzać odbiorcami i wykonywać testy bez wysyłki.

The screenshot shows the Twilio Chat News module management interface. At the top, there's a navigation bar with tabs: Wiadomości, Auto-odpowiedź, Przykładowe, AI, News (which is selected and highlighted in blue), and Multi-SMS. To the right of the tabs, it says "Panel Twilio Console" and "Indeksator: DEV".

The main area is divided into several sections:

- FAISS + OPENAI Status bazy wiedzy**: Shows the status as "gotowy" (last updated 13.12.2025, 10:08:35), the embedding type as "text-embedding-3-large", the chat node as "get-4-mlm", and vectors in the index as 884. It also indicates a backup is ready (6/6 backups available).
- NEWS / RAG Dodaj odbiorcę powiadomień**: A form to add a recipient for news notifications, including fields for phone number (+48123456789) and time (08:00). It includes a note about sending messages at the specified time.
- MENEDżER Aktywni odbiorcy (3)**: A list of three active recipients with their names and last activity times:
 - +4880802099541 [Aktywny] Wczynki kategorii (Last activity 11:00, 13.12.2025, 10:44:53, prompt for all categories)
 - +486600036933 [Aktywny] Wczynki prompt (Last activity 08:00, 13.12.2025, 19:02:46, prompt for all categories)
 - +4880802099541 [Aktywny] Wczynki kategorii (Last activity 08:00, 13.12.2025, 19:02:46, prompt for all categories)
- TEST FAISS Testuj zapytania do bazy newsów**: A section for testing AI responses to news queries. It includes a note: "Zdejmij pytanie do bazy FAISS, aby sprawdzić jak AI odpowiada na podstawie zselekowanych newsów." Below it, there's a "Zapytanie testowe" input field.
- SKŁADANIE + FAISS Uruchom pobieranie newsów**: A button to start news collection.

Widok zarządzania modułem News: konfiguracja, test i uruchomienie scrapingu.

Co widzisz

- Akcje administracyjne: test połączenia, uruchomienie scrapingu, podgląd plików źródłowych.
- Informacje o ostatnim buildzie i stanie modułu (czy dane są aktualne).

Możliwe interakcje

- Test generacji bez wysyłki (bezpieczne sprawdzenie jakości streszczeń i konfiguracji promptu).
- Uruchomienie scrapingu „na żądanie” (ręczny refresh danych) oraz szybka weryfikacja, czy pliki zostały zapisane.

Co dzieje się w tle

- Scraper pobiera treści i zapisuje pliki w katalogu danych (dla każdej kategorii osobno).
- Po pobraniu danych budowany/aktualizowany jest indeks FAISS, aby zapytania RAG miały świeży kontekst.
- API: `POST /api/news/scrape`, `GET /api/news/files`, `GET /api/news/files/<filename>`.

TEST FAISS

Testuj zapytania do bazy newsów

Zadaj pytanie do bazy FAISS aby sprawdzić jak AI odpowiada na podstawie zaśródowanych newsów.

Zaszyfruj testowe

np. Jaki są najważniejsze informacje ekonomiczne?

Wpisz pytanie lub prompt, a AI wygeneruje odpowiedź na podstawie newsów.

• Profesjonalne streszczenie wszystkich kategorii

Używa przestronnego promptu i funkcji politywnej każdej kategorii osobno sekcji.

Skraja FAISS

Ostatnia budowa —
Ostatnie powiadomienie —

ZAPISANE FAISS

Kategorie po skrapianiu

Biznes	Biznes.json	13.12.2025	Biznes.txt	813.4 kB
Gospodarka	gospodarka.json	13.12.2025	gospodarka.txt	814.1 kB
Nieruchomości	nieruchoscis.json	13.12.2025	nieruchoscis.txt	917.4 kB
Poradnik Finansowy	poradnik_finansowy.json	13.12.2025	poradnik_finansowy.txt	48.8 kB
Prawo	prawo.json	13.12.2025	prawo.txt	139.3 kB
Premium	premium.json	13.12.2025	premium.txt	82.1 kB
Technologie	technologie.json	13.12.2025	technologie.txt	102.1 kB

Znajdź indeks FAISS | Odśwież

SKRAPOWANIE = FAISS

Uruchom pobieranie newsów

news, aby pozwolić kategorie Business model, zapisz pliki i odbudować bazę FAISS.

Pobierz i obudź | Odśwież

BAZY FAISS

Odśwież | 1

Definicja odbiorców: numer, tryb promptu oraz godzina wysyłki.

Co widzisz

- Listę odbiorców powiadomień newsowych (numer + godzina + tryb).
- Flagi **enabled**, informację o ostatniej wysyłce oraz tryb generacji.

Możliwe interakcje

- Dodanie odbiorcy (numer w E.164) i wybór trybu: wszystkie kategorie (domyślnie) vs prompt niestandardowy.
- Włączenie/wyłączenie odbiorcy oraz test wiadomości bez wysyłki (preview).
- Force-send (wysyłka ręczna) dla odbiorcy w celach operacyjnych i weryfikacji routingu Twilio.

Co dzieje się w tle

- Konfiguracja odbiorców jest zapisywana w pliku **news_config.json**.
- Scheduler/worker wysyła o określonej godzinie wygenerowane streszczenia (jeśli odbiorca jest włączony).
- Wysyłka jest realizowana przez Twilio i powinna być widoczna w historii wiadomości.
- API: **GET/POST/DELETE /api/news/recipients** + akcje per odbiorca: **toggle**, **test**, **send**.

The screenshot shows a web-based application interface for managing news recipients and interacting with a FAISS index. At the top, there's a header with the text "Ostatnie powiadomienia: —". Below it, a section titled "ZAPISANE PLIKI" displays a grid of files categorized by news sources (Business, Market, etc.) and their corresponding JSON and TXT versions, along with file sizes and dates. A "Zbuduj indeks FAISS" button is visible at the top right of this section. Below this, another section titled "BAZY FAISS" allows selecting an active base or viewing existing ones. It lists an index named "faiss_openai_index" with details like creation date (13.12.2025), size (18.59 MB), status (aktywny), and actions like "Ustawić aktywną" and "Usuń".

Status indeksu i operacje: eksport/import backupu, test FAISS i budowa indeksu.

Co widzisz

- Status indeksu (czy załadowany, ile wektorów, jakie modele embeddings/chat).
- Gotowość backupu (wymagane pliki) oraz operacje import/export.

Możliwe interakcje

- Eksport backupu (ZIP) i import w celu odtworzenia środowiska (np. po wdrożeniu lub awarii).
- Budowa indeksu z istniejących plików oraz test zapytań FAISS (diagnostyka jakości i „świeżości” danych).

Co dzieje się w tle

- FAISS przechowuje wektory embeddings + mapowanie do dokumentów.
- Zapytania RAG pobierają kontekst z indeksu i opcjonalnie używają LLM do sformatowania odpowiedzi.
- API: `GET /api/news/faiss/status`, `GET /api/news/faiss/export`, `POST /api/news/faiss/import`, `POST /api/news/test-faiss`.

API modułu News (skrót)

- **Konfiguracja/odbiorcy:** `GET /api/news/config`, `GET/POST/DELETE /api/news/recipients`, `POST /api/news/recipients/<id>/(toggle|test|send)`
- **Scraping:** `POST /api/news/scrape`, pliki: `GET/DELETE /api/news/files`
- **FAISS:** `GET /api/news/faiss/status`, `GET /api/news/faiss/export`, `POST /api/news/faiss/import`, `POST /api/news/indices/build`

Import backupu ma limit bezpieczeństwa 250 MB.

Dane (SQLite) — trwałość, backup i migracje

Aplikacja przechowuje stan w SQLite. Ścieżka bazy jest konfigurowana przez `DB_PATH` (domyślnie `data/app.db`).

Co jest zapisywane

- **messages**: historia inbound/outbound, SID Twilio, status, błędy i timestamps.
- **multi_sms_***: batch + odbiorcy (statusy, liczniki, błędy per numer).
- **reminders**: harmonogramy cyklicznych wysyłek.
- **news_config.json**: konfiguracja odbiorców News (plik w katalogu danych).

Backup (praktycznie)

- Najprostszy backup to kopia pliku DB (przy zatrzymanej aplikacji lub w oknie niskiej aktywności).
- FAISS posiada dedykowany eksport/import ZIP z manifestem (API modułu News).

API — integracja i automatyzacja

Panel WWW korzysta z REST API. Poniżej najważniejsze punkty integracyjne (bez pełnego OpenAPI).

- **Wiadomości i konwersacje:** `GET /api/messages`, `GET /api/conversations`,
`GET /api/conversations/<participant>`
- **Wysyłka:** `POST /api/send-message` (SMS/MMS + Content SID)
- **Multi SMS:** `POST /api/multi-sms/batches`, `GET /api/multi-sms/batches`, `GET /api/multi-sms/batches/<id>`, `GET /api/multi-sms/batches/<id>/recipients`
- **Status callback:** `POST /twilio/status` (aktualizacja statusów wiadomości)

Glosariusz i najważniejsze pojęcia

Ten rozdział porządkuje terminologię spotykaną w panelach i API, aby PDF był samowystarczalny dla zespołów biznesowych i technicznych.

- **Batch (Multi SMS)** — zadanie wysyłki jednej wiadomości do wielu odbiorców; posiada status (`pending`, `processing`, `completed`, `failed`) i liczniki skuteczności.
- **Recipient (Multi SMS)** — pojedynczy numer przypisany do batcha; system śledzi status (`pending` / `sent` / `failed`) oraz ewentualny błąd Twilio.
- **Messaging Service SID** — identyfikator grupujący numery w Twilio; jeśli skonfigurowany, Multi SMS i UI używają go jako nadawcy zamiast pojedynczego numeru.
- **Webhook** — endpoint HTTP nasłuchujący żądań Twilio (np. inbound wiadomości, status dostarczenia). Każdy webhook jest podpisany nagłówkiem `X-Twilio-Signature`.
- **FAISS** — biblioteka wektorowa używana do szybkiego wyszukiwania podobnych dokumentów, podstawowy komponent modułu News/RAG.
- **Scraper** — proces pobierający artykuły i zapisujący je w katalogu danych; jego wynik zasila FAISS i moduł News.
- **Reminder** — harmonogram cyklicznej wysyłki SMS. Każde przypomnienie opisuje numer docelowy, treść i interwał.
- **Auto reply** — mechanizm automatycznych odpowiedzi (klasyczny lub AI). W trybie AI każde pytanie jest przekazywane do modelu OpenAI z kontekstem rozmowy.
- **Normalized number** — numer telefonu po oczyszczeniu (E.164). Pozwala scalać rekordy w historii i zapobiegać duplikatom.
- **News recipient** — wpis w `news_config.json`; określa numer, godzinę wysyłki i tryb promptu (wszystkie kategorie vs. prompt niestandardowy).

AI & Auto reply — zasady działania

AI działa jako globalny tryb auto reply; klasyczny auto reply jest wyłączany, gdy AI jest aktywne. Dodatkowo każdy tryb operuje tylko na wiadomościach nowszych niż punkt aktywacji (bez retro odpowiedzi).

1. AI buduje historię rozmowy z tabeli `messages` i wysyła zapytanie do OpenAI.
2. Odpowiedzi są wysyłane tak, aby zachować wątek konwersacji (preferowana metoda `send_reply_to_inbound`).
3. Operator może przetestować model w zakładce AI bez wysyłania SMS (endpoint `POST /api/ai/test`).

The screenshot shows the 'Panel czatu' (Chat Panel) interface. At the top, there's a navigation bar with tabs: 'Wiadomości', 'Auto-odpowiedź', 'Przypomnienia', 'AI' (which is selected and highlighted in blue), 'News', and 'Multi-SMS'. Below the navigation, there are two main sections: 'KONFIGURACJA Z AI' (AI Configuration) and 'PODGLĄD KONWERSACJI' (Conversation Preview).
Konfiguracja OpenAI: This section contains fields for 'Numer uczestnika rozmowy' (e.g., +48123456789) and 'Prompt systemowy' (e.g., 'Np. Jesteś asystentem obsługi klienta, który odpowiada krótko i po polsku.'), along with dropdowns for 'Model' (set to 'gpt-4o-mini') and 'Temperatura' (set to '0.7'). It also includes an 'OpenAI API key' input field containing 'sk-...'. A button labeled 'Zapisz konfigurację' (Save configuration) is at the bottom.
PODGLĄD KONWERSACJI: This section is titled 'Historia czatu z AI' (Conversation history with AI). It displays a message from 'Brak skonfigurowanego numeru' (No configured number) with the text 'Rozmowa z Brak skonfigurowanego numeru'. A note says 'Tylko tytuły do przyciągnięcia' (Only titles to attract). Below it, another message from 'Brak wiadomości w rozmowie z wybranym numerem.' (No messages in the conversation with the selected number).
TEST POŁĄCZENIA: This section is titled 'Wiadomość testowa' (Test message). It contains a text input field with 'Np. Cześć, to jest test. Iak się masz?' and a note 'Zostanie potraktowana jako ostatni komunikat użytkownika w rozmowie.' (Will be treated as the last message from the user in the conversation). A 'Próbuje połączenie' (Trying connection) button is at the bottom.

Interfejs konfiguracji AI oraz podgląd rozmowy w formie bąbelków

Co widzisz

- Konfigurację AI (włączenie, parametry, prompt/system) oraz stan aktywacji.
- Podgląd rozmowy w formie bąbelków, dzięki czemu operator widzi kontekst, na którym bazuje model.

Możliwe interakcje

- Włączenie AI auto reply i ustawienie promptu systemowego pod styl firmy i politykę odpowiedzi.
- Test generacji odpowiedzi bez wysyłki (bezpieczny preview) oraz dopasowanie parametrów.

Co dzieje się w tle

- AI buduje kontekst z historii wiadomości i wywołuje OpenAI wg ustawień (model/temperatura/limit historii).
- Jeśli tryb auto reply jest aktywny, inbound webhook enqueue'uje zadanie odpowiedzi (bez blokowania requestu).
- Wysyłka odpowiedzi przez Twilio jest rejestrowana w `messages`; statusy mogą się aktualizować callbackiem.
- System unika odpowiedzi na starsze wiadomości przed aktywacją trybu (brak retro odpowiedzi).

Bezpieczeństwo — kluczowe zasady

System jest zbudowany tak, aby minimalizować ryzyko nieautoryzowanych wywołań webhooków, wycieków kluczy oraz niekontrolowanej wysyłki.

Webhooki Twilio: walidacja podpisu

- Callbacki Twilio są weryfikowane nagłówkiem `X-Twilio-Signature` (RequestValidator).
- Walidację można wyłączyć tylko dla testów lokalnych przez `TWILIO_VALIDATE_SIGNATURE=false` (produkcyjnie: zostawić włączoną).

Klucze i konfiguracja

- Dane dostępowe Twilio są pobierane ze zmiennych środowiskowych i nie powinny trafiać do repozytorium.
- DB jest plikiem (SQLite) — w produkcji zadbaj o uprawnienia do katalogu `data/` oraz backup.

Kontrola kosztów

- Multi SMS ma limit odbiorców per batch.
- News wysyła wiadomości tylko do włączonych odbiorców; testy mogą działać bez wysyłki.

Operacje — uruchomienie i utrzymanie

Aplikacja uruchamia workery w tle (m.in. Multi SMS, przypomnienia, News). Najważniejsze operacyjnie jest przewidywalne środowisko, logi i backup danych.

Środowisko

- **DB:** `DB_PATH` (domyślnie `data/app.db`)
- **Twilio:** `TWILIO_ACCOUNT_SID`, `TWILIO_AUTH_TOKEN`, oraz `TWILIO_DEFAULT_FROM` lub `TWILIO_MESSAGING_SERVICE_SID`
- **AI/FAISS:** konfiguracja OpenAI (jeśli używana) + pliki indeksu w katalogu danych

Diagnostyka

- Jeśli Multi SMS nie wysyła: sprawdź logi workera oraz czy tabele `multi_sms_batches` i `multi_sms_recipients` istnieją w DB.
- Jeśli News/FAISS nie działa: sprawdź status `GET /api/news/faiss/status` i czy scraping zbudował indeks.

Backup

- SQLite: kopia pliku DB + rotacja (np. dzienna).
- FAISS: eksport ZIP przez endpoint i/lub zrzut katalogu indeksu.

Deployment & generowanie PDF

Plik HTML jest przygotowany do konwersji do PDF. Poniżej przykładowe polecenia do wygenerowania wysokiej jakości PDF lokalnie.

Jak wygenerować PDF

Rekomendowane: użyj headless Chrome lub wkhtmltopdf. Poniżej znajdują się dwie metody.

Headless Chrome

```
google-chrome --headless --disable-gpu --no-sandbox --print-to-pdf=twilio-chat-app.pdf  
file://$(pwd)/deploy/releases/full_documentation.html
```

wkhtmltopdf

```
wkhtmltopdf deploy/releases/full_documentation.html twilio-chat-app.pdf
```

Obie metody pozwalają na generację zaawansowanego, paginowanego PDF z poprawnymi stylami CSS. Jeżeli potrzebujesz, przygotuję skrypt automatyzujący proces — udostępniałem przykładowe skrypty w katalogu `scripts/`.