

MinskMir Voice Agent - Архитектура

Техническая документация

Ноябрь 2025

Contents

1 MinskMir Voice Agent	2
1.1 Детальная Архитектура Системы	2
1.2 1. Технологический стек	2
1.2.1 Frontend	2
1.2.2 Backend	2
1.2.3 База данных	2
1.2.4 Voice AI & LLM	2
1.3 2. Поток данных	3
1.3.1 2.1. Инициация голосового звонка	3
1.3.2 2.2. ElevenLabs Pipeline - Обработка голоса	3
1.4 3. Компоненты системы	5
1.4.1 3.1. Frontend Компоненты	5
1.5 4. API Эндпоинты	6
1.5.1 4.1. ElevenLabs Integration	6
1.5.2 4.2. Webhook Эндпоинты	6
1.5.3 4.3. Admin Эндпоинты	6
1.5.4 4.4. RAG & Knowledge Base	6
1.6 5. База данных	7
1.6.1 5.1. Supabase PostgreSQL Schema	7
1.6.2 5.2. RAG Vector Search	8
1.7 6. Автоматическое обновление данных	10
1.7.1 6.1. Vercel Cron Job - Каждый час	10
1.7.2 6.2. Web Scraping Pipeline	10
1.7.3 6.3. Мониторинг и Статистика	11
1.8 7. Безопасность и обработка персональных данных	12
1.8.1 7.1. Соответствие законодательству РБ	12
1.8.2 7.2. Категории обрабатываемых данных	12
1.8.3 7.3. Меры защиты данных	12
1.8.4 7.4. Права субъектов данных	13
1.8.5 7.5. Трансграничная передача данных	13
1.8.6 7.6. Контактная информация	13
1.9 8. Заключение	15
1.9.1 Последнее обновление	15
1.9.2 Политика конфиденциальности	15

1 MinskMir Voice Agent

1.1 Детальная Архитектура Системы

1.2 1. Технологический стек

1.2.1 Frontend

Технология	Описание
Next.js 13+	React framework с App Router
TypeScript	Типизированный JavaScript
Tailwind CSS	Utility-first CSS framework
React Hooks	useState, useEffect, custom hooks

1.2.2 Backend

Технология	Описание
Next.js API Routes	Serverless функции
Node.js	Runtime environment
TypeScript	Типизация на стороне сервера

1.2.3 База данных

Технология	Описание
Supabase	PostgreSQL как сервис
pgvector	Векторный поиск для RAG
Real-time	Подписки на изменения данных
RLS	Row Level Security

1.2.4 Voice AI & LLM

Технология	Описание
ElevenLabs	Полный стек голосового AI
GPT-5	Language Model для диалогов
TTS	Text-to-Speech синтез речи
STT	Speech-to-Text распознавание
WebSocket	Real-time аудио стриминг

1.3 2. Поток данных

1.3.1 2.1. Инициация голосового звонка

```
(bir.by)

"      "

VoiceWidget.tsx (Frontend)
-
-

POST /api/elevenlabs/conversation-token

API: conversation-token.ts
1.   agentId   env
2.   ElevenLabs API
3.   signed JWT token

{ conversationId, token }

VoiceWidget.tsx
-   ElevenLabs SDK
-   WebSocket

ElevenLabs Platform
real-time
```

1.3.2 2.2. ElevenLabs Pipeline - Обработка голоса

```
ELEVENLABS PLATFORM

1. AUDIO INPUT
-
-   WebSocket
-       : ~200-300ms

2. SPEECH-TO-TEXT (STT)
   : ElevenLabs Turbo STT
   : 30+
```

Output: " "

- 3. CONTEXT & KNOWLEDGE BASE
 - (RAG)
 -
 - Custom tools & functions

- 4. LLM GPT-5 PROCESSING
 - : GPT-5
 - Input: +
 - Processing: + RAG
 - Output: " ..."

- 5. TEXT-TO-SPEECH (TTS)
 - : ElevenLabs TTS v3

Streaming audio generation

- 6. AUDIO OUTPUT
 - WebSocket
 -
-

1.4 3. Компоненты системы

1.4.1 3.1. Frontend Компоненты

Компонент	Путь	Назначение
VoiceWidget	src/components/VoiceWidget	Главный виджет голосовых звонков
AINotificationWidget	src/components/AINotificationWidget	Уведомление о звонках лидах
AudioPlayer	src/components/AudioPlayer	Воспроизведение записей
CallButton	src/components/CallButton	Кнопка инициации звонка
Navigation	src/components/Navigation	Навигационное меню

1.4.1.1 VoiceWidget.tsx - Детали Основные функции: - Инициация звонка через Eleven-Labs - Управление состоянием разговора - WebSocket соединение - Обработка ошибок

Состояния:

```
type CallStatus =
  | 'idle' //
  | 'requesting' //
  | 'connecting' //
  | 'connected' //
  | 'disconnecting' //
  | 'error' //
```

1.5 4. API Эндпоинты

1.5.1 4.1. ElevenLabs Integration

Метод	Эндпоинт	Описание
POST	/api/elevenlabs/conversation-token	Получение токена для разговора
GET	/api/elevenlabs/conversations	Список всех разговоров
GET	/api/elevenlabs/signed-url	Signed URL для аудио записи
GET	/api/elevenlabs/health	Проверка статуса интеграции

1.5.2 4.2. Webhook Эндпоинты

Метод	Эндпоинт	Описание
POST	/api/webhook/voice-lead-enhanced	Webhook от ElevenLabs (основной)
POST	/api/webhook/proxy-crm	Проксирование в 1C CRM
POST	/api/webhook/1c-apartment-update	Обновления квартир из 1C

1.5.3 4.3. Admin Эндпоинты

Метод	Эндпоинт	Описание
GET	/api/admin/leads	Список всех лидов
GET	/api/admin/enhanced-leads	Лиды с полной информацией
GET	/api/admin/stats	Статистика системы
GET	/api/admin/quiz-submissions	Результаты квизов

1.5.4 4.4. RAG & Knowledge Base

Метод	Эндпоинт	Описание
POST	/api/tools/search-apartments	Векторный поиск квартир
GET	/api/knowledge-base/list	Список документов в базе знаний
POST	/api/knowledge-base/create-file	Загрузка файла в базу
POST	/api/knowledge-base/create-text	Создание текстового документа
DELETE	/api/knowledge-base/delete	Удаление документа

1.6 5. База данных

1.6.1 5.1. Supabase PostgreSQL Schema

1.6.1.1 Таблица: Lead

```
CREATE TABLE "Lead" (  
  id UUID PRIMARY KEY DEFAULT gen_random_uuid(),  
  name TEXT,  
  phone TEXT UNIQUE,  
  email TEXT,  
  status TEXT DEFAULT 'new',  
  source TEXT, -- 'voice', 'form', 'quiz'  
  notes TEXT,  
  createdAt TIMESTAMP DEFAULT NOW(),  
  updatedAt TIMESTAMP DEFAULT NOW()  
);
```

Статусы лидов: - new - Новый лид - contacted - Был контакт - qualified - Квалифицирован - converted - Конвертирован - lost - Потерян

1.6.1.2 Таблица: Conversation

```
CREATE TABLE "Conversation" (  
  id UUID PRIMARY KEY DEFAULT gen_random_uuid(),  
  conversationId TEXT UNIQUE,  
  leadId UUID REFERENCES "Lead"(id),  
  transcript JSONB,  
  recordingUrl TEXT,  
  duration INTEGER,  
  sentiment TEXT, -- 'positive', 'neutral', 'negative'  
  createdAt TIMESTAMP DEFAULT NOW()  
);
```

1.6.1.3 Таблица: apartments (RAG)

```
CREATE TABLE apartments (  
  id UUID PRIMARY KEY DEFAULT gen_random_uuid(),  
  external_id TEXT UNIQUE NOT NULL,  
  address TEXT NOT NULL,  
  price INTEGER NOT NULL,  
  area FLOAT NOT NULL,  
  rooms INTEGER NOT NULL,  
  floor INTEGER,  
  total_floors INTEGER,  
  district TEXT,  
  description TEXT,  
  photos TEXT[],  
  url TEXT,  
  embedding vector(3072), -- pgvector  
  source TEXT DEFAULT 'website',  
  status TEXT DEFAULT 'active',  
  scraped_at TIMESTAMP,  
  created_at TIMESTAMP DEFAULT NOW(),  
  updated_at TIMESTAMP DEFAULT NOW()  
);
```

-- HNSW

```
CREATE INDEX idx_apartments_embedding ON apartments
USING hnsw (embedding vector_cosine_ops)
WITH (m = 16, ef_construction = 64);
```

1.6.2 5.2. RAG Vector Search

Модель embeddings: text-embedding-3-large (OpenAI)

Размерность: 3072

Метрика: Cosine Similarity

PostgreSQL функция для поиска:

```
CREATE OR REPLACE FUNCTION match_apartments(
    query_embedding vector(3072),
    match_threshold float DEFAULT 0.75,
    match_count int DEFAULT 10,
    filter_min_price int DEFAULT NULL,
    filter_max_price int DEFAULT NULL,
    filter_rooms int DEFAULT NULL,
    filter_district text DEFAULT NULL
)
RETURNS TABLE (
    apartment_id uuid,
    address text,
    price int,
    area float,
    rooms int,
    floor int,
    description text,
    similarity float,
    last_updated timestamp
)
LANGUAGE plpgsql AS $$
BEGIN
    RETURN QUERY
    SELECT
        a.id,
        a.address,
        a.price,
        a.area,
        a.rooms,
        a.floor,
        a.description,
        1 - (a.embedding <=> query_embedding) as similarity,
        a.updated_at
    FROM apartments a
    WHERE
        (filter_min_price IS NULL OR a.price >= filter_min_price)
        AND (filter_max_price IS NULL OR a.price <= filter_max_price)
        AND (filter_rooms IS NULL OR a.rooms = filter_rooms)
        AND (filter_district IS NULL OR a.district = filter_district)
        AND 1 - (a.embedding <=> query_embedding) > match_threshold
    ORDER BY a.embedding <=> query_embedding
    LIMIT match_count;
END;
$$;
```

1.7 6. Автоматическое обновление данных

1.7.1 6.1. Vercel Cron Job - Каждый час

Расписание: 0 * * * * (каждый час в 0 минут)

Эндпоинт: /api/cron/scrape-apartments

1.7.2 6.2. Web Scraping Pipeline

VERCEL CRON ()

1. WEB SCRAPER (Puppeteer)

- bir.by/apartments
-
- : , ,
-

2. DIFF ENGINE

-
- :
- * (added)
- * (updated)
- * (removed)

3. (OpenAI)

- embeddings
- : text-embedding-3-large
- : 3072

4. (Supabase pgvector)

- Upsert
- HNSW
-

5. (ElevenLabs KB)

-
-

1.7.3 6.3. Мониторинг и Статистика

Логируемые метрики: - Количество обработанных объявлений - Количество новых/обновленных/удаленных
- Время выполнения scraping - Ошибки и исключения

Алерты: - Если scraped = 0 (сайт недоступен) - Если errors > 10 (проблемы с парсингом) - Если duration > 60s (медленная работа)

1.8 7. Безопасность и обработка персональных данных

1.8.1 7.1. Соответствие законодательству РБ

Нормативно-правовая база:

1. Закон Республики Беларусь № 99-3 «О защите персональных данных» (7 мая 2021 г.)
2. Указ Президента РБ № 383 (7 ноября 2022 г.)
3. Постановление Совета Министров РБ № 568 (30 августа 2022 г.)

Основной сервер: Республика Беларусь (дата-центр на территории РБ)

Статус оператора: ООО “МинскМир” зарегистрировано в качестве оператора персональных данных

1.8.2 7.2. Категории обрабатываемых данных

Категория	Данные	Срок хранения
Общедоступные	ФИО, телефон, email	До конвертации + 1 год
Специальные	Аудио записи	90 дней
Специальные	Транскрипты	2 года
Технические	IP адрес, User Agent	90 дней (обезличенный)
Обезличенные	Статистика, метрики	Бессрочно

1.8.3 7.3. Меры защиты данных

1.8.3.1 Шифрование

Тип	Метод	Описание
В транзите	TLS 1.3	Все HTTPS/WSS соединения
В покое	AES-256	База данных Supabase
Аудио файлы	AES-256	S3-совместимое хранилище

1.8.3.2 Контроль доступа

```
-- Row Level Security (RLS) Supabase
CREATE POLICY "Managers can view leads"
ON "Lead" FOR SELECT
TO authenticated
USING (
  auth.jwt() ->> 'role' = 'manager' OR
  auth.jwt() ->> 'role' = 'admin'
);

CREATE POLICY "Only admins can delete"
ON "Lead" FOR DELETE
TO authenticated
USING (auth.jwt() ->> 'role' = 'admin');
```

1.8.3.3 Аудит и логирование

```
CREATE TABLE "DataAccessLog" (
  id UUID PRIMARY KEY DEFAULT gen_random_uuid(),
  userId UUID NOT NULL,
  action TEXT NOT NULL, -- 'read', 'update', 'delete', 'export'
  tableName TEXT NOT NULL,
```

```

recordId UUID,
ipAddress INET,
userAgent TEXT,
timestamp TIMESTAMP DEFAULT NOW(),
dataSnapshot JSONB
);

CREATE INDEX idx_access_log_user ON "DataAccessLog"(userId);
CREATE INDEX idx_access_log_time ON "DataAccessLog"(timestamp DESC);

```

1.8.4 7.4. Права субъектов данных

Право	Срок ответа	Реализация
Доступ к данным	15 дней	GET /api/gdpr/export-my-data
Копия данных	30 дней	PDF/JSON экспорт
Исправление	10 дней	PATCH /api/gdpr/update-my-data
Удаление	3 дня	DELETE /api/gdpr/delete-my-data
Отзыв согласия	Немедленно	POST /api/gdpr/withdraw-consent

1.8.5 7.5. Трансграничная передача данных

1.8.5.1 ElevenLabs (США/ЕС) Цель: Обработка голосовых данных (TTS/STT)

Срок хранения: 72 часа (автоматическое удаление)

DPA: Заключен Data Processing Agreement

Уведомление: ОАЦ при Президенте РБ

1.8.5.2 OpenAI (США) Цель: Обработка текстовых данных (анализ транскриптов)

Использование для обучения: НЕТ (opt-out включен)

DPA: Заключен Data Processing Agreement

Уведомление: ОАЦ при Президенте РБ

1.8.6 7.6. Контактная информация

Оператор персональных данных:

- **Наименование:** ООО "МинскМир"
- **УНП:** [указывается УНП]
- **Адрес:** 220000, РБ, г. Минск, ул. [адрес]

Ответственное лицо:

- **ФИО:** [Фамилия Имя Отчество]
- **Email:** privacy@bir.by
- **Телефон:** +375 (XX) XXX-XX-XX
- **Часы:** понедельник-пятница, 9:00-18:00

Запросы:

- Доступ к данным: data-access@bir.by
- Удаление данных: data-update@bir.by
- Отзыв согласия: consent-withdrawal@bir.by
- Жалобы: complaints@bir.by

Уполномоченный орган:

- **Название:** Оперативно-аналитический центр при Президенте РБ
 - **Адрес:** 220013, г. Минск, ул. Я. Коласа, 37
 - **Телефон:** +375 (17) 374-91-51
 - **Email:** info@oac.gov.by
 - **Сайт:** <https://oac.gov.by>
-

1.9 8. Заключение

1.9.1 Последнее обновление

Дата: Ноябрь 2025

Версия: 2.0

1.9.2 Политика конфиденциальности

Полная версия доступна:

- **На сайте:** <https://bir.by/privacy>
- **В офисе:** По запросу
- **Email:** privacy@bir.by

Документ обновлен: 2025-11-03