

### **Бизнес-требования**

1. Название
2. User Story
3. Use Case
4. Макет
5. BPMN

### **Функциональные требования**

1. Архитектура
2. Модель данных
3. ER-диаграмма
4. Sequence-диаграмма (Диаграмма последовательности)
5. REST
6. Swagger

### **Критерии приёмки и нефункциональные требования**

1. Критерии приёмки
2. Нефункциональные требования

1. Название.

AI-агент: Smart Support Assistant

2. User Story.

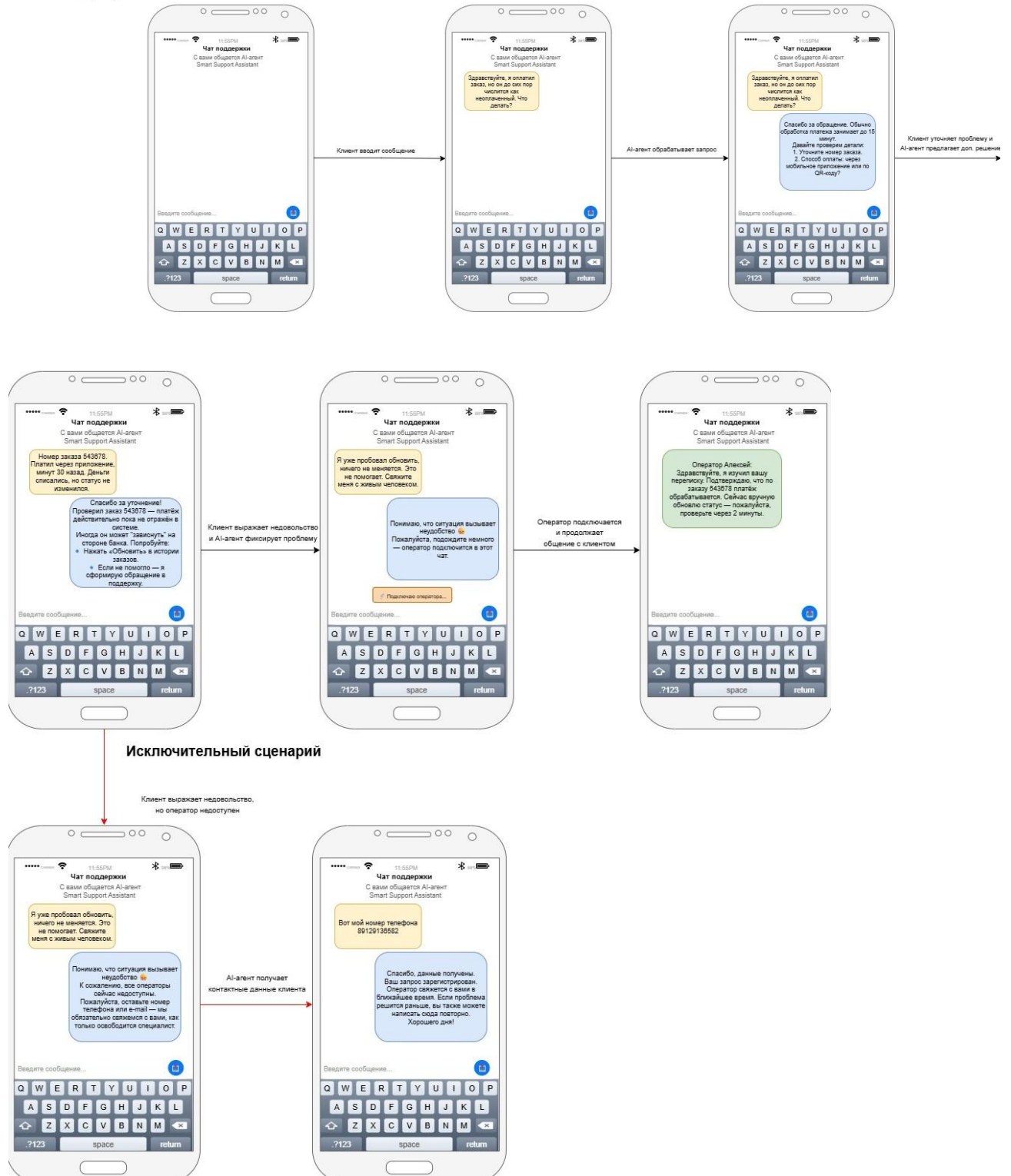
Как оператор службы поддержки, я хочу, чтобы AI-агент обрабатывал обращения до моего подключения, используя базу знаний и историю общения, и передавал чат мне, если клиент выражает недовольство, чтобы я подключался только при необходимости

3. Use Case.

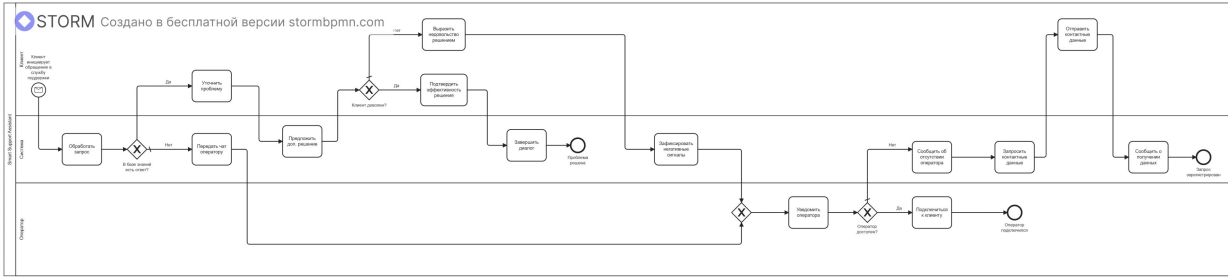
Элемент	Описание
Заголовок	Обработка обращений AI-агентом и передача оператору при недовольстве клиента
Акторы	Оператор службы поддержки, AI-агент, Клиент
Предусловия	AI-агент обучен и интегрирован с чатом поддержки
	В системе доступна база знаний и история общения с клиентами
	Оператор онлайн и готов принять чат при необходимости
Ограничения	AI-агент не принимает финальные решения по чувствительным вопросам (возвраты, жалобы на сотрудников и т.п.)
	Передача оператору осуществляется только при наличии явных индикаторов недовольства (например, негативные фразы, повторяющиеся жалобы)
	Максимальное время автономной работы AI без передачи оператору — 10 минут
Триггер	Клиент инициирует обращение в службу поддержки через чат
Основной сценарий	1. Клиент открывает чат и пишет сообщение с вопросом или проблемой
	2. AI-агент обрабатывает запрос, используя базу знаний и историю общения
	3. Клиент получает первый ответ и уточняет проблему, если решение не подходит
	4. AI-агент предлагает дополнительное решение или пояснение, опираясь на контекст
	5. Клиент продолжает диалог, указывает, что он недоволен (напр. "это не помогает", "свяжите меня с человеком")
	6. AI-агент фиксирует негативные сигналы, классифицирует их как маркер недовольства
	7. AI-агент уведомляет оператора и передает ему диалог с краткой сводкой
	8. Оператор подключается и продолжает общение с клиентом в том же чате
Альтернативный сценарий	2а. AI-агент не может найти подходящий ответ в базе знаний
	3а. AI-агент сообщает клиенту, что подключит оператора для дальнейшей помощи
	4а. AI-агент передаёт чат оператору по причине недостатка информации
	2b. AI-агент находит точный ответ на вопрос клиента в базе знаний и формирует персонализированный ответ с учётом истории обращений.
	3b. Клиент получает ответ и подтверждает, что проблема решена (например, "спасибо, всё понятно", "это помогло").
Исключительный сценарий	4b. AI-агент завершает диалог, предлагает оценить помощь и при необходимости сохраняет детали обращения в историю.
	5а. Оператор недоступен (офлайн или перегружен)
Результат	AI-агент сообщает клиенту о задержке и предлагает оставить контактные данные или дождаться ответа

## 4. Макет

### Основной сценарий

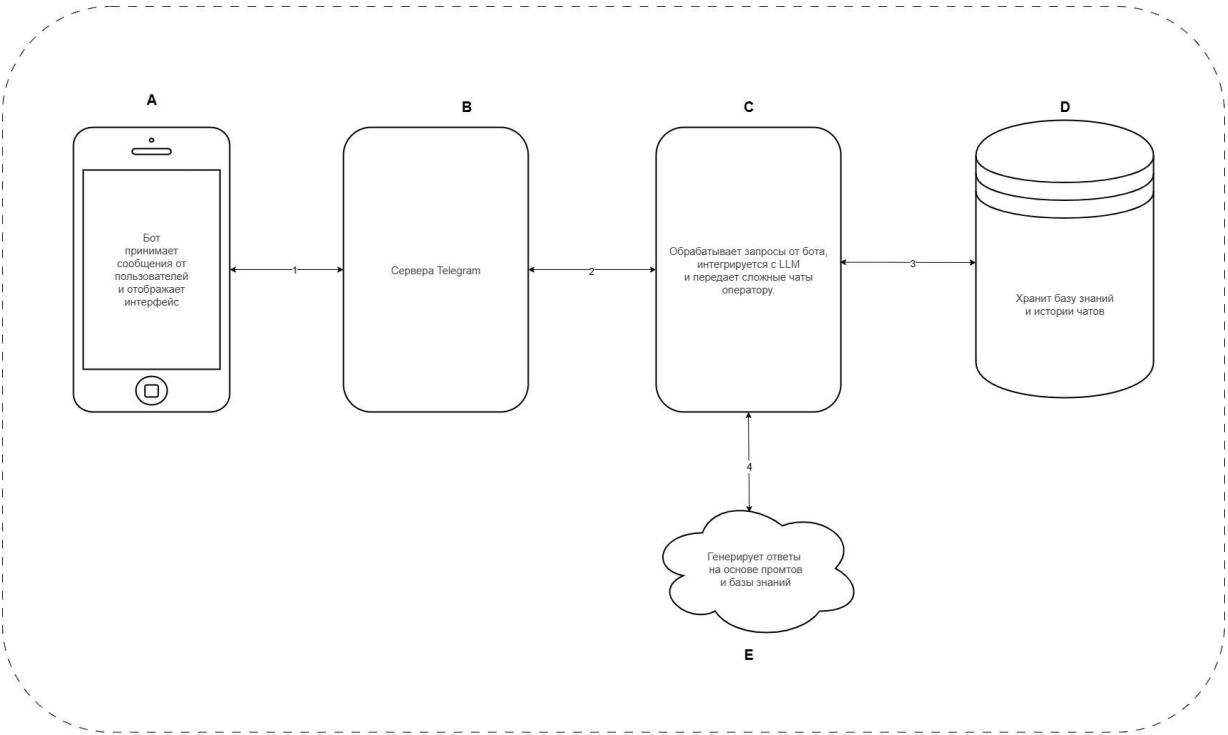


5. BPMN



Функциональные требования.

1. Архитектура.

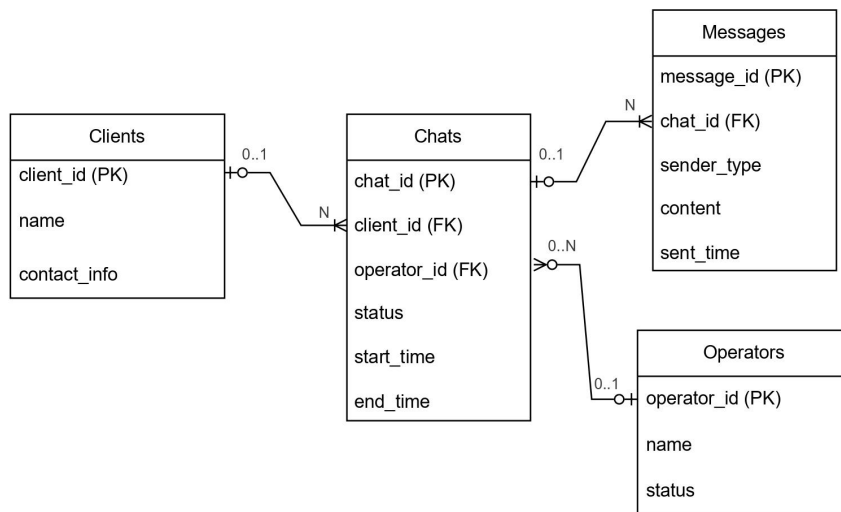


- A - Frontend  
B - Сервера Telegram  
C - Backend  
D - Реляционная БД, PG (PostgreSQL)  
E - LLM
- 1 - протокол взаимодействия HTTPS  
2 - протокол взаимодействия HTTPS  
3 - протокол взаимодействия TCP/IP  
4 - протокол взаимодействия HTTPS

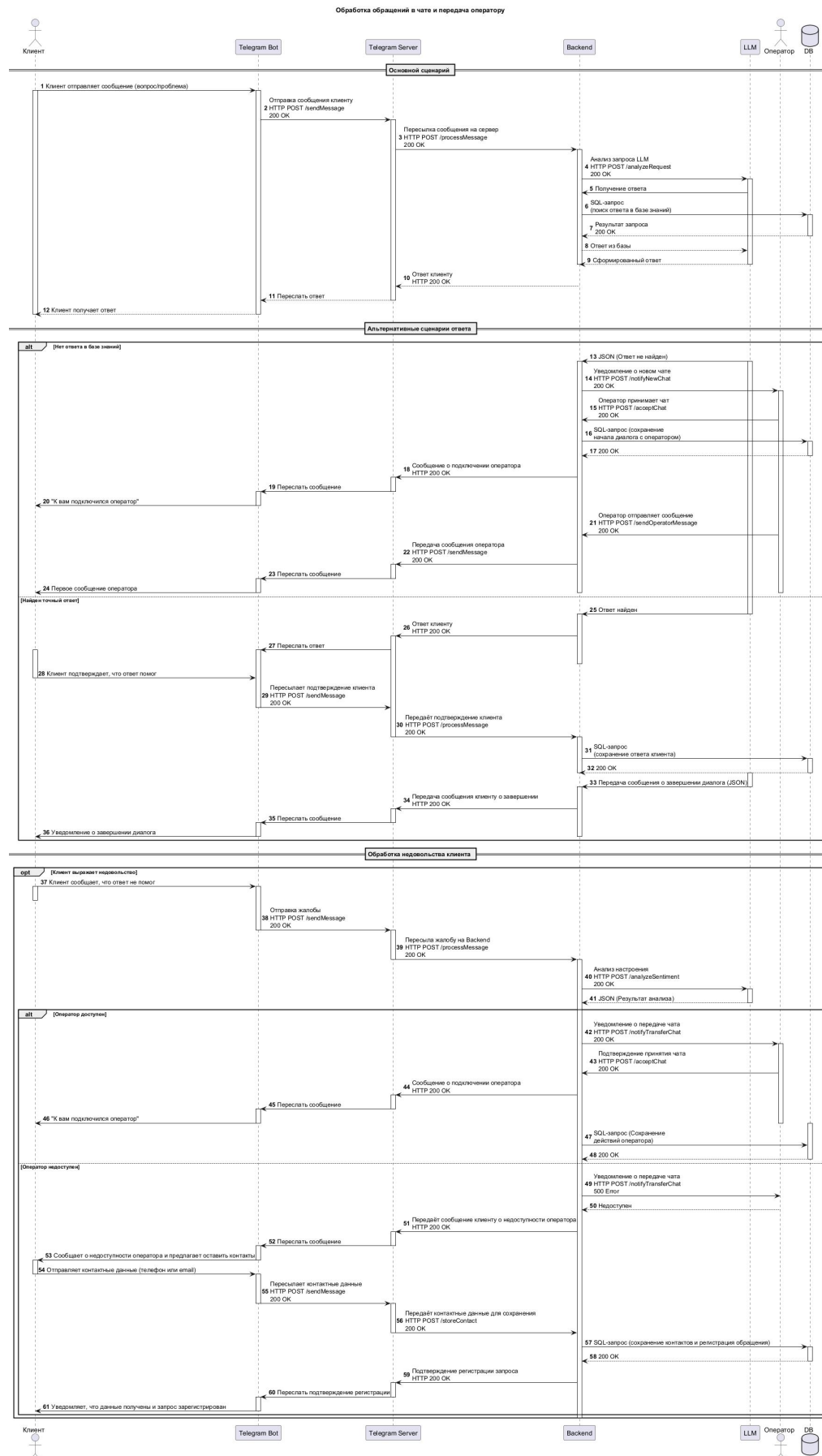
## 2. Модель данных.

Объект Clients		
Родительская сущность	Атрибут	Описание
Clients	-	Объект клиента, который инициирует обращения в службу поддержки.
	client_id	Уникальный идентификатор клиента
	name	Имя клиента
	contact_info	Контактная информация
Объект Operators		
Родительская сущность	Атрибут	Описание
Operators	-	Объект оператора, который отвечает за ведение чата.
	operator_id	Уникальный идентификатор оператора
	name	Имя оператора
	status	Статус (online/offline/busy)
Объект Chats		
Родительская сущность	Атрибут	Описание
Chats	-	Объект чата, который связывает клиента и оператора
	chat_id	Уникальный идентификатор чата
	client_id	Ссылка на клиента
	operator_id	Ссылка на оператора
	status	Статус чата
	start_time	Время начала чата
	end_time	Время завершения чата
Объект Messages		
Родительская сущность	Атрибут	Описание
Messages	-	Объект сообщения, которое создаётся в рамках чата
	message_id	Уникальный идентификатор сообщения
	chat_id	Ссылка на чат
	sender_type	Тип отправителя (клиент/оператор)
	content	Текст сообщения
	sent_time	Время отправки сообщения

### 3. ER-диаграмма.



## 4. Sequence-диаграмма (Диаграмма последовательности).



## 5. REST

### GET /v1/clients/{clientId}

Получение информации о конкретном клиенте по его идентификатору

#### Request

Название параметра	Тип данных	Находится в	Описание	Обязательность параметра
clientId	int	path	Уникальный идентификатор клиента	да

#### Response

Название параметра	Тип данных	Находится в	Описание	Обязательность параметра
client_id	int	body	Уникальный идентификатор клиента	да
name	string	body	Имя клиента	да
contact_info	string	body	Контактная информация	да

Response code: <200>

### POST /v1/operators

Создание нового оператора в системе

#### Request

Название параметра	Тип данных	Находится в	Описание	Обязательность параметра
name	string	body	Имя оператора	да
status	string	body	Статус (online/offline/busy)	нет

#### Response

Название параметра	Тип данных	Находится в	Описание	Обязательность параметра
operator_id	int	body	Уникальный идентификатор оператора	да
name	string	body	Имя оператора	да
status	string	body	Статус оператора	да

Response code: <201> (Created)



**GET /v1/chats/{chatId}/messages**

Получение всей переписки (всех сообщений) из определенного чата

**Request**

Название параметра	Тип данных	Находится в	Описание	Обязательность параметра
chatId	int	path	Уникальный идентификатор чата	да

**Response**

Название параметра		Тип данных	Находится в	Описание	Обязательность параметра
chat_id		int	body	Уникальный идентификатор чата	да
messages		array	body	Массив объектов сообщений	да
	message_id	int	body	Уникальный идентификатор сообщения	да
	sender_type	string	body	Тип отправителя (client/operator)	да
	content	string	body	Текст сообщения	да
	sent_time	string	body	Время отправки сообщения	да

Response code: <200>

6. Swagger

Smart Support Assistant API 1.0.0 OAS 3.0

API для системы поддержки с AI-агентом и эскалацией к операторам.

Servers

https://api.smartsupport.example.com - Production server

Clients

Операции с клиентами

GET

/v1/clients/{clientId}

Получение информации о конкретном клиенте

Operators

Управление операторами

POST

/v1/operators

Создание нового оператора

Chats

Работа с чатами и сообщениями

GET

/v1/chats/{chatId}/messages

Получение переписки чата

# Критерии приёмки и нефункциональные требования

## 1. Критерии приёмки

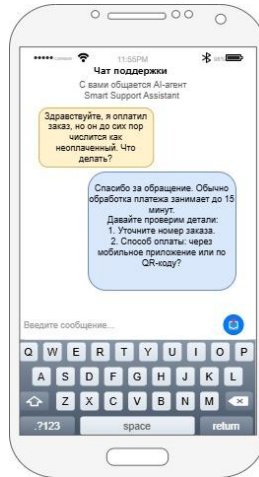
Номер кейса: 1

**Функциональность:** AI-агент обрабатывает первое обращение клиента.

**Дано:** Клиент открывает чат поддержки и пишет первое сообщение с вопросом.

**Когда:** AI-агент получает сообщение клиента.

**Тогда:** Система обрабатывает запрос, анализируя базу знаний и историю предыдущих обращений клиента, и формирует релевантный ответ.



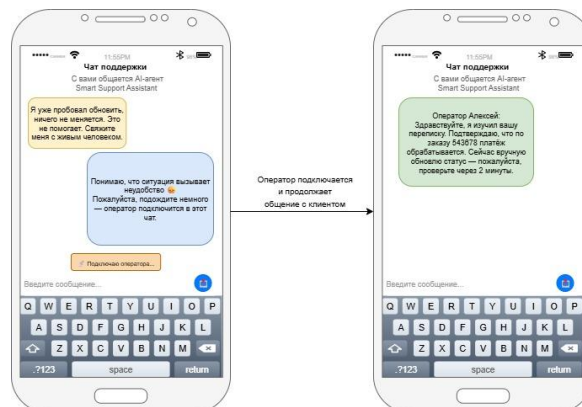
Номер кейса: 2

**Функциональность:** AI-агент передает чат оператору с контекстом.

**Дано:** AI-агент зафиксировал недовольство клиента и инициировал передачу чата.

**Когда:** Происходит передача чата оператору.

**Тогда:** Система автоматически создает и присоединяет к чату краткую сводку диалога (контекст, проблема, предложенные решения, причина эскалации) для оператора.



## **2. Нефункциональные требования**

### **Требования надежности:**

Система AI-агента должна быть доступна 99,9% времени (24/7)

В случае полного отказа AI-агента все входящие чаты должны автоматически и немедленно перенаправляться операторам без прерывания работы сервиса.

Система должна фиксировать и сохранять все диалоги в режиме реального времени, чтобы исключить потерю данных общения при аварийном завершении работы.

### **Требования производительности:**

Ответ AI-агента на первое сообщение клиента должен быть <3 секунд после получения запроса.

Ответ AI-агента на уточняющие вопросы в середине диалога должен быть <2 секунд.

Генерация сводки для оператора при передаче чата должна занимать <2 секунд.

### **Требования безопасности:**

Все личные данные клиента (история обращений, контакты), обрабатываемые AI-агентом, должны быть анонимизированы при использовании в моделях машинного обучения.

Доступ к настройке и обучению AI-агента должен быть ограничен кругом уполномоченных администраторов.