

# Создание автоматизированной системы обработки обращений в службу технической поддержки (СТП)

**Проектирование информационно-управляющих систем**

**Исполнитель:**

Присталов Ярослав

# 1. Описание продукта

## 1.1. Наименование продукта:

Автоматизированная система обработки обращений в службу технической поддержки (АС ОО СТП)

## 1.2. Цель проекта:

Автоматизация и оптимизация бизнес-процессов обработки обращений клиентов в службу технической поддержки для повышения скорости реагирования, соблюдения SLA и качества обслуживания.

## 1.3. Задачи проекта:

- Построить модель автоматизируемых бизнес-процессов
- Сформулировать функциональные и нефункциональные требования к системе
- Определить ключевые показатели эффективности (KPI) процессов

## 1.4. Область автоматизации:

Процессы обработки заявок от создания до закрытия, включая контроль SLA, назначение исполнителей, уведомления и эскалацию.

## 2. Анализ предметной области

**2.1. Текущее состояние процессов:** Обработка заявок осуществляется с использованием электронной почты, чатов и электронных таблиц, что приводит к:

- Ручному отслеживанию сроков SLA
- Отсутствию единой точки контроля
- Задержкам в назначении исполнителей
- Сложностям в формировании отчетности

### Типы обращений:

1. Вопрос по функционалу (новые клиенты)
2. Консультация по конфигурации
3. Сообщение об ошибке (Bug)
4. Запрос о выходе новой версии
5. Инцидент безопасности

### 2.2. Пользователи системы:

- **Клиенты** — создают и отслеживают заявки через веб-портал.
- **Инженеры** — обрабатывают заявки, производят диагностику, взаимодействуют с Клиентом.
- **Руководитель подразделения** — контролирует распределение нагрузки, SLA, формирует отчеты.

# 3. Выбор методологии и инструментов моделирования

## 3.1. Обоснование выбора методологии BPMN 2.0:

BPMN (Business Process Model and Notation) 2.0 выбран как отраслевой стандарт по следующим причинам:

- **Стандартизация** — нотация понятна бизнес-аналитикам, разработчикам и заказчикам
- **Богатая семантика** — наличие специализированных элементов для таймеров, сообщений, событий, что соответствует требованиям автоматизации SLA и уведомлений
- **Наглядность разделения ролей** — возможность отображения пулов и дорожек (swimlanes) для клиентов, системы и сотрудников

## 3.2. Обоснование выбора CASE-средства [draw.io](https://draw.io):

- **Бесплатность и доступность** — не требует лицензионных отчислений, что соответствует учебным целям
- **Кроссплатформенность** — работает как онлайн-версия, так и desktop-приложение
- **Полная поддержка BPMN 2.0** — включает все необходимые элементы нотации
- **Простота освоения** — интуитивный интерфейс, подходит для быстрого создания качественных диаграмм

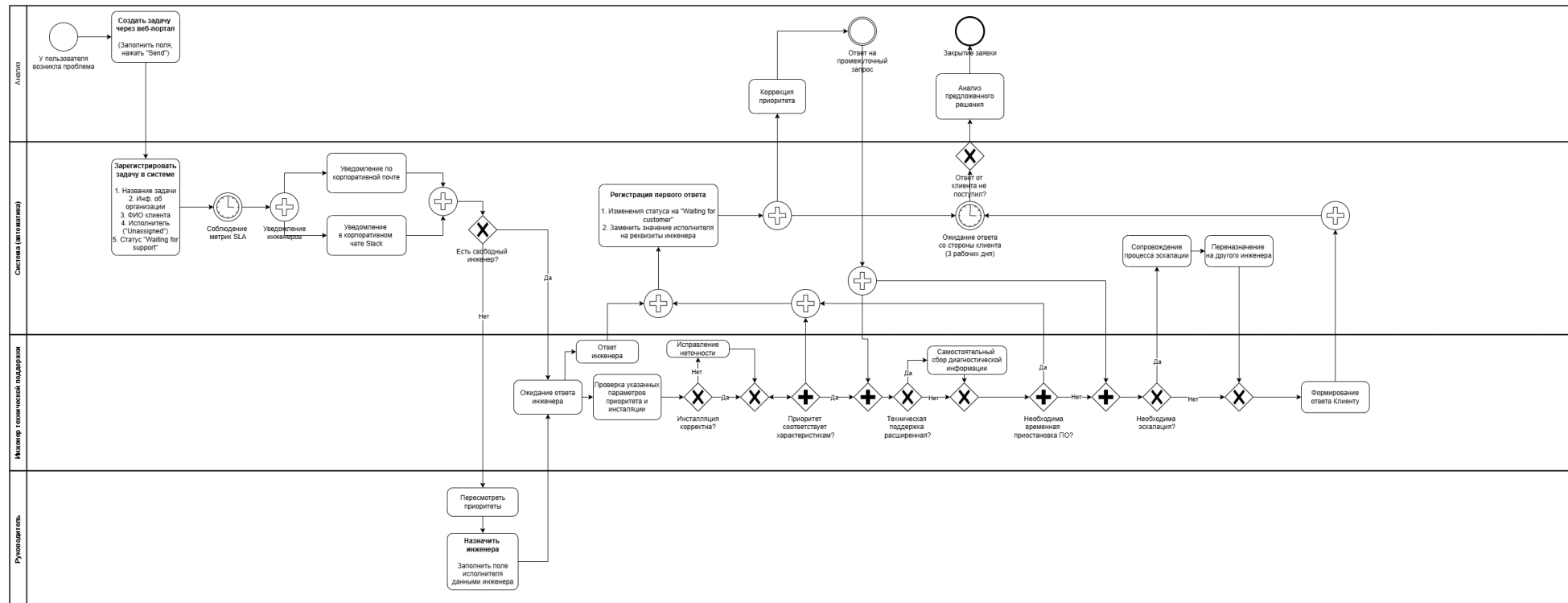
# 4. Моделирование бизнес-процессов

**Описание:** Диаграмма демонстрирует три основных этапа бизнес-процесса:

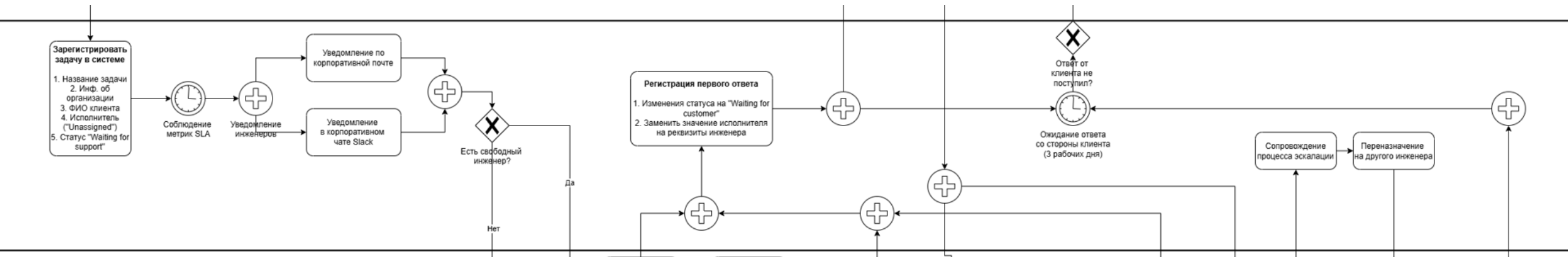
1. Создание и регистрация заявки
2. Обработка и взаимодействие
3. Закрытие и архивация

**Нотация:** BPMN 2.0

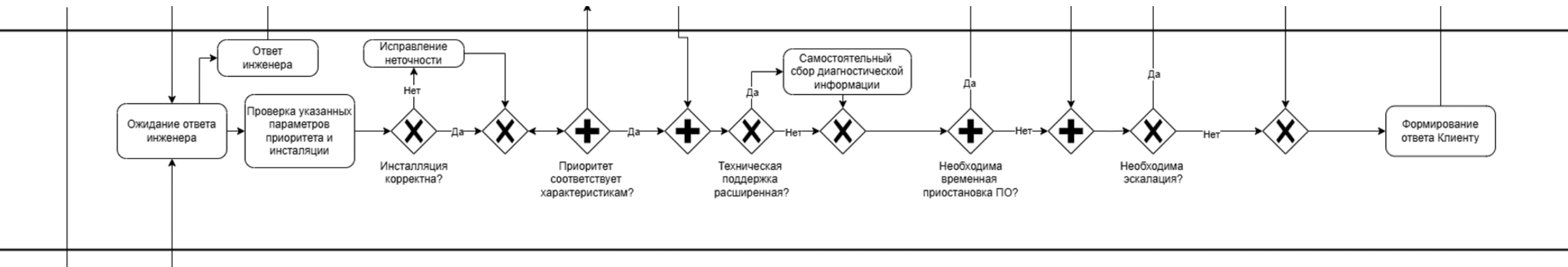
**CASE-средство:** draw.io



## 4. Моделирование бизнес-процессов



Уровень автоматизации



Уровень Инженера поддержки

# 5. Требования к информационной системе (Функциональные)

## Управление жизненным циклом заявки

- Создание заявки клиентом через веб-портал с обязательными полями: тип, приоритет, описание, вложения
- Автоматическая генерация ID формата ES-XXXX
- Автоматическое ведение статусов: «Waiting for support» → «Waiting for customer» → «Closed»
- Автоматическое назначение исполнителя при публикации первого ответа (если поле «Unassigned»)
- Ручное назначение/смена исполнителя с указанием причины (для эскалации)
- Возможность повторного открытия закрытой заявки в течение 14 дней

## Управление SLA и таймерами

- Автоматический расчет времени первого ответа в зависимости от приоритета
- Визуальная индикация оставшегося времени (цветовая маркировка)
- Автоматическая отправка предупреждений за 1 час до дедлайна
- Фиксация факта соблюдения/нарушения SLA при получении первого ответа
- Автоматическая отправка инцидентных писем при нарушении SLA

## Система уведомлений

- Отправка уведомлений о новых заявках и обновлениях в канал Slack
- Отправка email-уведомлений с деталями заявки
- Уведомление клиентов об ответах инженеров и о закрытии заявок
- Настройка правил уведомлений по типам событий и ролям пользователей

# 5. Требования к информационной системе (Функциональные)

## Отчетность и аналитика

- Отчет «Количество заявок за период в разрезе типов обращений»
- Отчет «Количество заявок за период в разрезе приоритетов»
- Отчет «Соблюдение SLA по первому ответу»
- Отчет «Нагрузка на инженеров (количество обработанных заявок)»
- Возможность экспорта отчетов в XLSX и PDF форматах

## Управление пользователями и правами доступа

- Разделение прав по ролям: Клиент, Инженер, Руководитель, Администратор
- Клиенты видят только свои заявки и заявки своей организации
- Аутентификация по логину и паролю с возможностью восстановления



# 5. Требования к информационной системе (Нефункциональные)

## Производительность

- Время отклика интерфейса при стандартных операциях:  $\leq 2$  секунд
- Поддержка одновременной работы: 1000 клиентов, 50 инженеров
- Обработка до 200 новых заявок в час в пиковые нагрузки

## Надежность и доступность

- Время доступности системы:  $\geq 99.5\%$  в рабочее время (Пн-Пт, 9:00-18:00)
- Автоматическое резервное копирование данных: ежедневно
- Время восстановления после сбоя:  $\leq 1$  часа

## Безопасность

- Использование HTTPS для всех соединений
- Шифрование паролей в базе данных
- Ведение журнала аудита всех значимых действий
- Защита от SQL-инъекций и XSS-атак

## Удобство использования (Usability)

- Интерфейс инженера: выполнение основных операций за  $\leq 3$  клика
- Интерфейс клиента: создание заявки за  $\leq 2$  минуты без обучения
- Адаптивный дизайн для работы с мобильных устройств
- Локализация интерфейса на русском языке

## Интеграционная способность

- REST API для интеграции с внешними системами
- Готовые интеграции: Slack, Email (SMTP)
- Поддержка вебхуков для отправки уведомлений

## 6. Ключевые показатели эффективности (KPI)

### 6.1. Показатели качества обслуживания

#### Процент соблюдения SLA по первому ответу:

$$\left( \frac{\text{Количество заявок с соблюдением SLA}}{\text{Общее количество заявок}} \right) \times 100\%$$

#### Целевое значение:

$\geq 95\%$

#### Среднее время первого ответа:

Рассчитывается отдельно по каждому приоритету.

#### Целевое значение:

В пределах установленного SLA для каждого приоритета.

### 6.2. Показатели операционной эффективности

#### Среднее время решения заявки:

От момента создания до закрытия.

#### Целевое значение:

$\leq 24$  часа для приоритета «Средний»

#### Коэффициент эскалации:

$$\left( \frac{\text{Количество эскалированных заявок}}{\text{Общее количество заявок}} \right) \times 100\%$$

#### Целевое значение:

$\leq 15\%$

### 6.3. Показатели удовлетворенности

#### Индекс удовлетворенности клиентов:

Измеряется через опрос после закрытия заявки (5-балльная шкала).

#### Целевое значение:

$\geq 4.2$  балла

#### Процент повторных обращений по той же проблеме:

$$\left( \frac{\text{Количество повторно открытых заявок}}{\text{Общее количество закрытых заявок}} \right) \times 100\%$$

#### Целевое значение:

$\leq 5\%$

## 7. Информационная архитектура. Модель данных

### КЛИЕНТ (Client)

- Может создавать ЗАЯВКИ (Ticket)
- Является частью ОРГАНИЗАЦИИ (Organization)
- Обладает УРОВЕНЬ ПОДДЕРЖКИ (support\_level)

### ЗАЯВКА (Ticket)

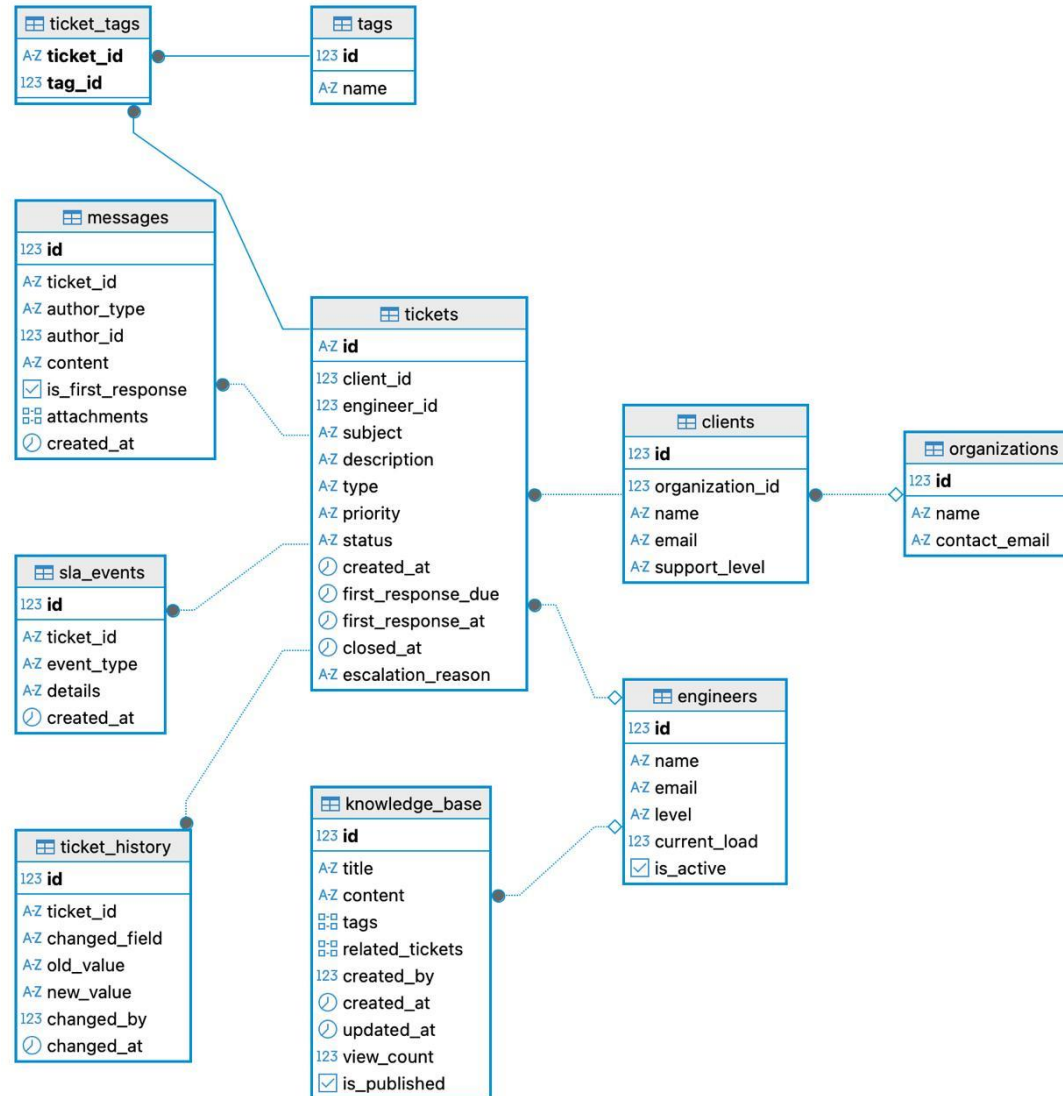
- Имеет ПРИОРИТЕТ (Priority)
- Имеет ТИП (Type)
- Имеет СТАТУС (Status)
- Назначена на ИНЖЕНЕРА (Engineer)
- Содержит СООБЩЕНИЯ (Message)
- Содержит СОБЫТИЯ SLA (SLAEvent)
- Имеет ВЛОЖЕНИЯ (Attachment)

### ИНЖЕНЕР (Engineer)

- обрабатывает много → ЗАЯВОК



# 7. Информационная архитектура. Модель данных



## 8. Информационная архитектура. Модульная структура

### Модуль веб-портала для клиентов

- Создание заявок
- Просмотр истории
- Общение с инженерами

### Модуль рабочего места инженера

- Очередь заявок
- Ответы клиентам
- Эскалация
- Закрытие

### Модуль управления (руководитель)

- Назначение исполнителей
- Мониторинг SLA
- Отчетность

### Модуль автоматизации процессов

- Контроль SLA (таймеры)
- Отправка уведомлений
- Автоназначение

### Модуль интеграций

- Slack-бот
- Email-рассылки
- REST API

### Модуль отчётности

- Готовые отчёты
- Аналитика
- Экспорт данных

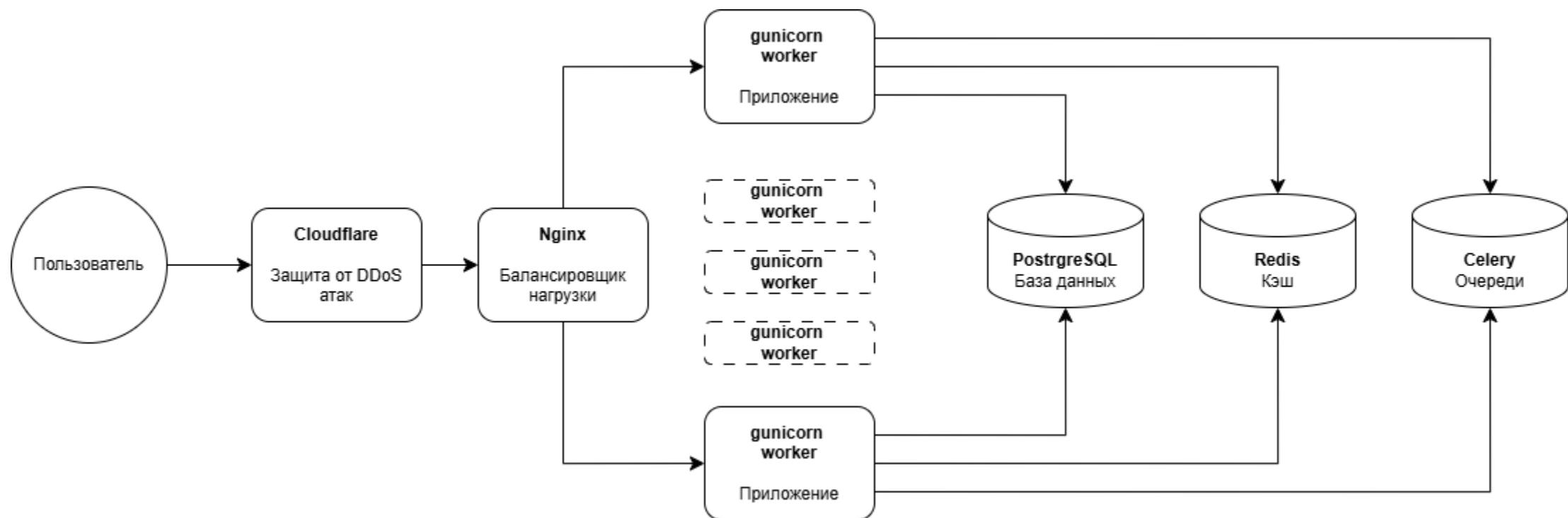
# 9. Программно-техническая архитектура.

## Технологический стек

Уровень	Технология	Обоснование выбора
Фронтенд	React + TypeScript	Готовые UI-компоненты
Бэкенд	Python (Django/Flask)	Богатый выбор библиотек
База данных	PostgreSQL	Надёжная, с свободной лицензией
Кэш	Redis	Хранение сессий
Очереди задач	Celery (для Python)	Обеспечение отказоустойчивости
Сервер	Nginx + Gunicorn	Балансировка нагрузки и обслуживание запросов
Контейнеризация	Docker	Для простого развёртывания и возможности масштабирования
Хостинг	Виртуальная машина от Yandex Cloud	Управление конфигурацией

## 9. Программно-техническая архитектура. Технологический стек

Отказоустойчивость обеспечивает микросервисная архитектура.



## 10. Оценка эффективности внедрения системы. Операционные метрики (KPI)

Показатель	Текущее значение	Целевое значение	Метод измерения
Процент соблюдения SLA по первому ответу	74%	95%	(Заявки с соблюдением SLA / Все заявки) × 100%
Среднее время первого ответа	4.5 часа	2.5 часа	Среднее арифметическое по всем заявкам
Среднее время решения заявки	24 часа	18 часов	От создания до закрытия
Коэффициент эскалации	28%	12%	(Эскалированные заявки / Все заявки) × 100%
Нагрузка на инженера (активных заявок)	8-12	4-6	Количество заявок в статусе "в работе"
Процент повторных обращений	15%	5%	(Повторные обращения по этой теме / Все обращения) × 100%



# 10. Оценка эффективности внедрения системы.

## Другие метрики

### Финансовые метрики

- Экономия на штрафах за SLA
- Экономия оплаты труда за счёт эффективности
- Стоимость обработки одной заявки
- Коэффициент использования ресурсов

### Качественные метрики

- Индекс удовлетворенности клиентов (1-10)
- Удовлетворенность сотрудников

# 11. Оценка эффективности внедрения системы.

## Оценка ресурсов на разработку и внедрение

### Трудовые ресурсы (человек, месяцев)

- Аналитик бизнес-процессов (1 чел. 2 мес.)
- Системный архитектор (1 чел. 2 мес.)
- Backend-разработчик (2 чел. 6 мес.)
- Frontend-разработчик (1 чел. 3 мес.)
- Тестировщик (QA) (1 чел. 2 мес.)
- Руководитель проекта (1 чел. 4 мес.)

### Аппаратные и программные ресурсы (спецификация)

- Сервер приложений (4 vCPU, 8 GB RAM, 100 GB SSD)
- Сервер базы данных (4 vCPU, 16 GB RAM, 200 GB SSD)
- Доменное имя и SSL ([support.company.ru](https://support.company.ru))

### Ресурсы на внедрение и сопровождение

- Обучение сотрудников СТП (12 чел)
- Миграция исторических данных
- Техническая поддержка (1-й год)
- Обновления и доработки