# **Сервис отправки push-нотификаций**

| **Версия документа** | **Дата изменений** | **Описание** | **Автор изменений** | **Ссылки на Confluence** | **Ссылка на задачу в Jira** |
| --- | --- | --- | --- | --- | --- |
| *0.1 Мажорная версия должна соответствовать версии сервиса* | *28/09/2024* | *Создание первой версии* | *Юрий Бленегапце* |  |  |

## **Контекст**

*Сейчас в банке единственные каналы коммуникации с клиентами — SMS и email. Банк хочет проверить гипотезу, что push-уведомления можно использовать как альтернативный способ отправки сообщений, особенно маркетинговых уведомлений. Однако особенностью push-уведомлений является обязательное разрешение на их получение через мобильное приложение.*

## **Описание бизнес-домена**

*Push-уведомления — это уведомления, которые приходят в мобильное приложение клиента.*

*Управление Push-уведомлениями возможно только при наличии разрешения клиента на их получение.*

## **Бизнес-требования**

* *Поддержка отправки push-уведомлений.*
* *Проверка наличия разрешений у клиентов на получение уведомлений.*
* *Интеграция с CRM и облаком для отправки уведомлений на устройства.*
* *Сбор данных о статусе доставки уведомлений.*

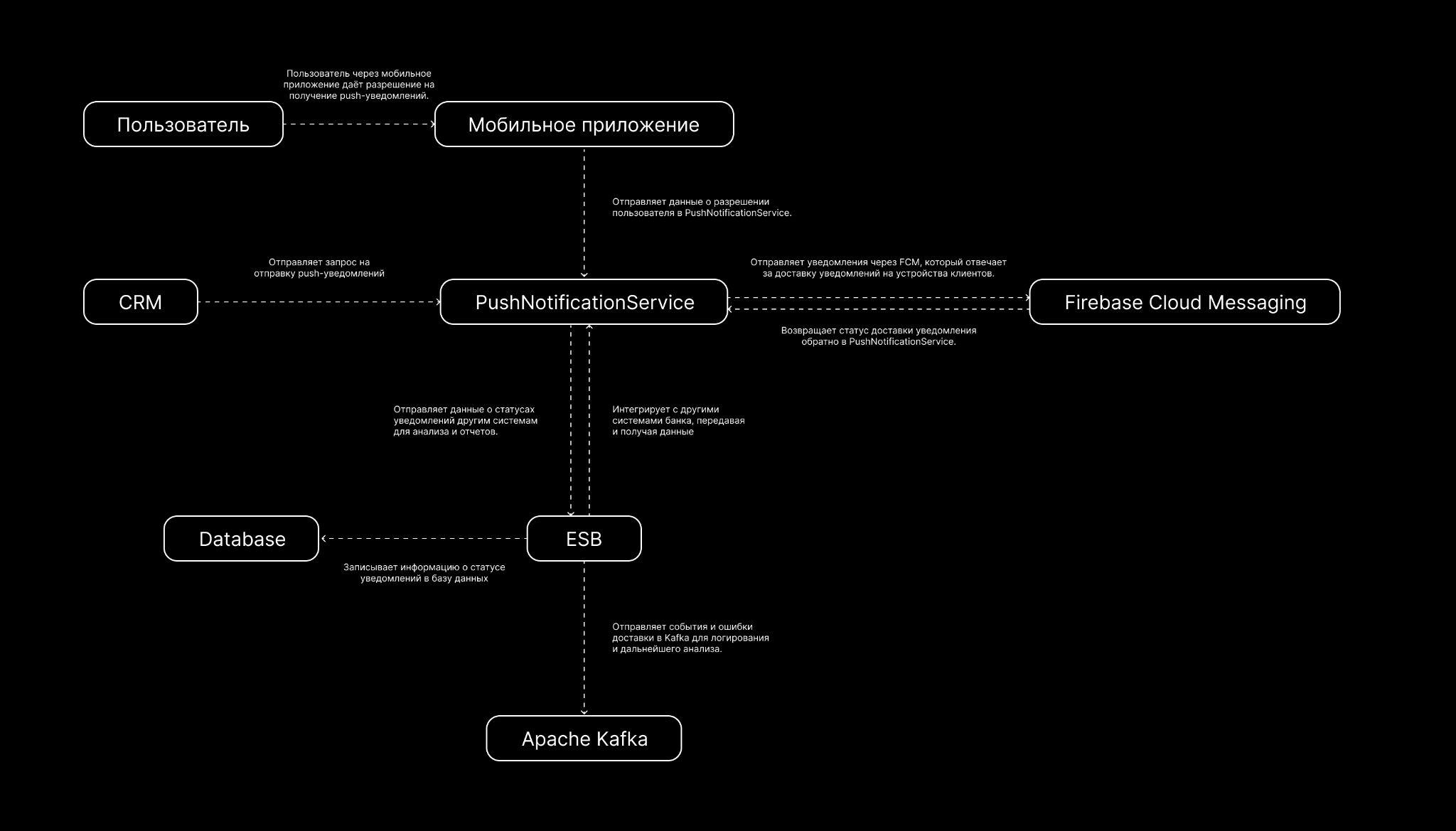
## **Технические требования**

* *Поддержка различных платформ (iOS, Android, веб).*
* *Используемые компоненты:*
  + *CRM*
  + *ESB*
  + *База данных*
  + *Apache Kafka*
  + *Новый сервис*
  + *Firebase Cloud Messaging (FCM) или аналоги*

## **Ограничения сервиса**

* *Управление SMS-подписками.*
* *Взимание оплат (но есть проверка оплат).*
* *Непосредственная отправка SMS (только триггеры для соответствующего сервиса отправки).*
* *Регистрация клиентов.*
* *Тарификация.*

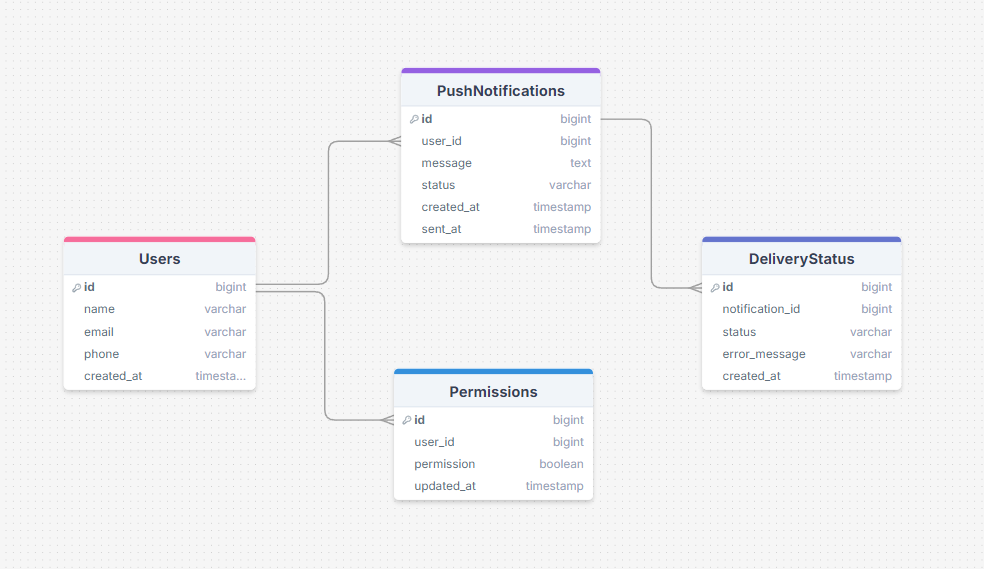
## **Карта взаимодействий**

**

| *Инициатор* | *Адресат* | *Описание для чего строится это взаимодействие* | *Тип взаимодействия (синх/асинх)* | *Протокол (HTTP, HTPS, др) или технология (Kafka, ESB)* |
| --- | --- | --- | --- | --- |
| *Пользователь* | *Мобильное приложение* | *Пользователь даёт разрешение на получение push-уведомлений* | *Синхронное* | *HTTP/HTTPS* |
| *Мобильное приложение* | *Сервис Push-уведомлений* | *Отправка данных о разрешениях* | *Синхронное* | *HTTP/HTTPS* |
| *CRM* | *Сервис Push-уведомлений* | *CRM отправляет триггер для отправки push-уведомлений* | *Синхронное* | *HTTP/HTTPS* |
| *Сервис Push-уведомлений* | *Firebase Cloud Messaging* | *Отправка push-уведомления клиенту через FCM* | *Асинхронное* | *HTTP/HTTPS* |
| *Firebase Cloud Messaging* | *Сервис Push-уведомлений* | *Возвращение статуса доставки уведомления* | *Асинхронное* | *HTTP/HTTPS* |
| *Сервис Push-уведомлений* | *Kafka* | *Логирование статуса доставки и ошибок через топики Kafka* | *Асинхронное* | *Kafka* |
| *Сервис Push-уведомлений* | *ESB* | *Отправка событий и данных в другие системы через ESB* | *Асинхронное* | *ESB* |
| *ESB* | *Система аналитики/мониторинга* | *Передача данных для дальнейшей аналитики* | *Асинхронное* | *ESB* |

## **Описание базы данных**

### **Схема базы данных**

**

### **Описание таблиц**

#### **1. Users**

Содержит информацию о пользователях, которым могут быть отправлены push-уведомления. Каждая запись представляет уникального клиента банка.

| **Поле** | **Описание** | **Тип данных** | **Обязательное** | **Уникальное** | **Источник данных для всех событий** | **Пример** |
| --- | --- | --- | --- | --- | --- | --- |
| *id* | *Уникальный идентификатор уведомления* | *BIGHT* | *Да* | *Да* | *Автоматически генерируется базой данных* | *123* |
| *name* | *Имя пользователя* | *VARCHAR(100)* | *Да* | *Нет* | *Вводится клиентом при регистрации или обновляется через CRM* | *Иван Иванов* |
| *email* | *Электронная почта* | *VARCHAR(255)* | *Да* | *Да* | *Вводится клиентом при регистрации или обновляется через CRM* | *iviv@mail.ru* |
| *phone* | *Номер телефона* | *VARCHAR(15)* | *Нет* | *Нет* | *Вводится клиентом при регистрации или обновляется через CRM* | *+79991234567* |
| *created\_at* | *Дата и время регистрации пользователя* | *TIMESTAMP* | *Да* | *Нет* | *Автоматически фиксируется при добавлении записи* | *2024-09-09 00:00:00* |

#### **2. Permissions (Разрешения)**

| **Поле** | **Описание** | **Тип данных** | **Обязательное** | **Уникальное** | **Источник данных для всех событий** | **Пример** |
| --- | --- | --- | --- | --- | --- | --- |
| *id* | *Уникальный идентификатор разрешения* | *BIGINT* | *Да* | *Да* | *Автоматически генерируется* | *12345* |
| *user\_id* | *Идентификатор пользователя* | *BIGINT* | *Да* | *Нет* | *Взят из системы* | *12345* |
| *permission* | *Статус разрешения (true/false)* | *BOOLEAN* | *Да* | *Нет* | *Указан пользователем* | *delivered* |
| *updated\_at* | *Дата и время изменения разрешения* | *TIMESTAMP* | *Да* | *Нет* | *Автоматически при обновлении* | *"2024-09-01 10:00:00"* |

#### 

#### **3. PushNotifications**

Содержит записи обо всех отправленных уведомлениях. Каждое уведомление связано с конкретным пользователем через внешний ключ user\_id.

| **Поле** | **Описание** | **Тип данных** | **Обязательное** | **Уникальное** | **Источник данных для всех событий** | **Пример** |
| --- | --- | --- | --- | --- | --- | --- |
| *id* | *Уникальный идентификатор уведомления* | *BIGHT* | *Да* | *Да* | *Автоматически генерируется базой данных* | *123* |
| *user\_id* | *Идентификатор пользователя* | *BIGINT* | *Да* | *Нет* | *Извлекается из таблицы Users* | *123* |
| *message* | *Текст уведомления* | *TEXT* | *Да* | *Нет* | *Генерируется CRM на основании маркетинговых или сервисных событий* | *Ваш заказ готов* |
| *status* | *Статус уведомления* | *VARCHAR(50)* | *Да* | *Нет* | *Устанавливается сервисом на основании ответа от FCM* | *delivered* |
| *created\_at* | *Дата и время регистрации пользователя* | *TIMESTAMP* | *Да* | *Нет* | *Автоматически фиксируется при создании записи* | *2024-09-09 00:01:00* |
| *sent\_at* | *Дата и время отправки уведомления* | *TIMESTAMP* | *Нет* | *Нет* | *Заполняется после успешной отправки уведомления через FCM* | *2024-09-09 00:02:00* |

## 

#### **4. DeliveryStatus**

Хранит данные о статусах доставки каждого push-уведомления. Это позволяет отслеживать, было ли уведомление доставлено, есть ли ошибки, и в какое время был обновлен статус.

| **Поле** | **Описание** | **Тип данных** | **Обязательное** | **Уникальное** | **Источник данных для всех событий** | **Пример** |
| --- | --- | --- | --- | --- | --- | --- |
| *id* | *Уникальный идентификатор уведомления* | *BIGHT* | *Да* | *Да* | *Автоматически генерируется базой данных* | *123* |
| *notification\_id* | *Идентификатор уведомления* | *BIGINT* | *Да* | *Нет* | *Извлекается из таблицы PushNotifications* | *123* |
| *status* | *Статус доставки* | *VARCHAR(50)* | *Да* | *Нет* | *Обновляется на основе данных, полученных от FCM* | *delivered* |
| *error\_message* | *Описание ошибки* | *VARCHAR(255)* | *Нет* | *Нет* | *Генерируется системой на основании ответа от FCM при ошибке* | *Ошибка сети* |
| *created\_at* | *Время обновления статуса* | *TIMESTAMP* | *Да* | *Нет* | *Автоматически фиксируется при обновлении статуса* | *2024-09-09 00:03:00* |

## 

## 

## **Описание REST-интерфейса**

### **Swagger**

*openapi: 3.0.1*

*info:*

*title: Сервис Push-уведомлений API*

*version: 1.0.0*

*servers:*

*- url: https://api.pushservice.com*

*description: Основной сервер службы push-уведомлений*

*paths:*

*/notifications/permissions:*

*post:*

*summary: Проверка разрешения на отправку уведомлений*

*requestBody:*

*required: true*

*content:*

*application/json:*

*schema:*

*type: object*

*properties:*

*client\_id:*

*type: string*

*description: Уникальный идентификатор пользователя*

*example: "12345"*

*responses:*

*'200':*

*description: Разрешение успешно проверено.*

*content:*

*application/json:*

*schema:*

*type: object*

*properties:*

*client\_id:*

*type: string*

*description: Уникальный идентификатор пользователя*

*example: "12345"*

*permission:*

*type: boolean*

*description: Статус разрешения*

*example: true*

*'400':*

*description: Ошибка в запросе.*

*/notifications/send:*

*post:*

*summary: Отправка push-уведомления через FCM*

*requestBody:*

*required: true*

*content:*

*application/json:*

*schema:*

*type: object*

*properties:*

*client\_id:*

*type: string*

*description: Уникальный идентификатор пользователя*

*example: "12345"*

*message:*

*type: string*

*description: Текст уведомления*

*example: "Ваш заказ доставлен"*

*responses:*

*'200':*

*description: Уведомление успешно отправлено.*

*'400':*

*description: Ошибка в запросе.*

*/notifications/delivery-status:*

*post:*

*summary: Запись статуса доставки уведомления*

*requestBody:*

*required: true*

*content:*

*application/json:*

*schema:*

*type: object*

*properties:*

*notification\_id:*

*type: string*

*description: Идентификатор уведомления*

*example: "notif\_67890"*

*status:*

*type: string*

*description: Статус доставки*

*example: "delivered"*

*timestamp:*

*type: string*

*format: date-time*

*description: Время обновления статуса*

*example: "2024-09-09T10:00:00Z"*

*responses:*

*'200':*

*description: Статус доставки записан.*

*'400':*

*description: Ошибка в запросе.*

### 

#### ***Метод POST /notifications/permissions***

***Описание****: Проверяет, есть ли у клиента разрешение на получение push-уведомлений.*

***Запрос:***

*{*

*"client\_id": "12345"*

*}*

***Ответ:***

*{*

*"client\_id": "12345",*

*"permission": true*

*}*

#### 

#### ***Метод POST /notifications/send***

***Описание****: Отправляет push-уведомление клиенту.*

***Запрос:***

*{*

*"client\_id": "12345",*

*"message": "Ваш заказ доставлен"*

*}*

***Ответ:***

*{*

*"status": "Уведомление отправлено",*

*"message\_id": "notif\_67890"*

*}*

#### ***Метод POST /notifications/delivery-status***

***Описание****: Записывает статус доставки push-уведомления в базе данных.*

***Запрос:***

*{*

*"notification\_id": "notif\_67890",*

*"status": "delivered",*

*"timestamp": "2024-09-09T10:00:00Z"*

*}*

***Ответ:***

*{*

*"status": "Статус успешно записан"*

*}*

**Интеграции сервиса**

### **Kafka**

# **Топик push\_delivery\_status**

*Этот топик логирует успешные и неудачные попытки доставки уведомлений. Данные, отправленные в этот топик, содержат информацию о статусе уведомления и времени доставки.*

| *Параметр* | *Наименование* | *Тип данных* | *Обязательность* | *Описание использования* | *Пример* |
| --- | --- | --- | --- | --- | --- |
| *notification\_id* | *Идентификатор уведомления* | *BIGINT* | *Да* | *Уникальный идентификатор отправленного уведомления* | *1234* |
| *user\_id* | *Идентификатор пользователя* | *BIGINT* | *Да* | *Идентификатор клиента, которому отправлено уведомление* | *1234* |
| *status* | *Статус уведомления* | *VARCHAR(50)* | *Да* | *Статус доставки уведомления (например, delivered, failed)* | *delivered* |
| *timestamp* | *Время события* | *TIMESTAMP* | *Да* | *Время события, фиксирующее статус уведомления* | *2024-09-09 10:00:00* |

**Пример запроса:**

*{*

*"notification\_id": 12345,*

*"user\_id": 67890,*

*"status": "delivered",*

*"timestamp": "2024-09-0910:00:00"*

*}*

# **Топик push\_error\_log**

*Этот топик используется для записи ошибок, возникающих при попытке доставки уведомлений (например, проблемы с подключением к FCM).*

| *Параметр* | *Наименование* | *Тип данных* | *Обязательность* | *Описание использования* | *Пример* |
| --- | --- | --- | --- | --- | --- |
| *error\_id* | *Идентификатор ошибки* | *BIGINT* | *Да* | *Уникальный идентификатор ошибки* | *12345* |
| *notification\_id* | *Идентификатор уведомления* | *BIGINT* | *Да* | *Уникальный идентификатор отправленного уведомления* | *12345* |
| *client\_id* | *Идентификатор клиента* | *BIGINT* | *Да* | *Идентификатор клиента* | *12345* |
| *error\_message* | *Описание ошибки* | *VARCHAR(255)* | *Да* | *Сообщение об ошибке (например, Ошибка соединения с FCM)* | *Ошибка соединения с FCM* |
| *timestamp* | *Время события* | *TIMESTAMP* | *Да* | *Время события, фиксирующее статус уведомления* | *2024-09-09T10:05:00Z* |

**Пример запроса:**

*{*

*"error\_id": 12345,*

*"notification\_id": 12345,*

*"client\_id": 12345,*

*"error\_message": "Ошибка соединения с FCM",*

*"timestamp": "2024-09-09T10:05:00Z"*

*}*

### **ESB**

# **SendNotificationData**

*Указывают название очереди, особенности использования данного метода, а также описание получаемых/передаваемых сервисом данных*

| *Параметр* | *Наименование* | *Тип данных* | *Обязательность* | *Описание использования* | *Пример* |
| --- | --- | --- | --- | --- | --- |
| *notification\_id* | *Идентификатор уведомления* | *BIGINT* | *Да* | *Уникальный идентификатор отправленного уведомления* | *12345* |
| *client\_id* | *Идентификатор клиента* | *BIGINT* | *Да* | *Идентификатор клиента* | *12345* |
| *status* | *Статус уведомления* | *VARCHAR(50)* | *Да* | *Статус доставки уведомления (например, delivered, failed)* | *delivered* |
| *timestamp* | *Время события* | *TIMESTAMP* | *Да* | *Время события, фиксирующее статус уведомления* | *2024-09-09T10:01:00* |

## **Пример запроса:** *{*

*"notification\_id": "12345",*

*"client\_id": "12345",*

*"status": "delivered",*

*"timestamp": "2024-09-09T10:01:00"*

*}*

## 

## 

## 

## 

## 

## **Варианты использования**

### **Основной сценарий: Отправка push-уведомления**

#### **Описание сценария**

### **Предусловие:**Пользователь зарегистрирован в системе и дал разрешение на получение push-уведомлений.CRM имеет триггер, инициирующий отправку уведомления.

### **Триггер:** Событие в CRM (например, день рождения клиента или акция).

### **Гарантия успеха:** Уведомление успешно отправлено и доставлено клиенту.

#### **Основной сценарий**

| **Шаг** | **Описание шага** | **Примечание** |
| --- | --- | --- |
| **1** | CRM отправляет запрос на отправку уведомления в сервис Push-уведомлений | Запрос содержит идентификатор клиента и текст уведомления |
| **2** | Сервис Push-уведомлений проверяет разрешение клиента | Сервис проверяет наличие активного разрешения |
| **3** | При наличии разрешения сервис отправляет уведомление через FCM | Уведомление отправляется через Firebase Cloud Messaging |
| **4** | FCM возвращает статус доставки уведомления | Сервис Push-уведомлений получает от FCM статус доставки |
| **5** | Статус доставки записывается в базу данных | Статус сохраняется в таблице DeliveryStatus |
| **6** | Сервис Push-уведомлений отправляет событие в Kafka | Логирование события доставки через Kafka |

### 

### **Альтернативные сценарии**

#### **Альтернативный сценарий 1: Клиент не дал разрешение на уведомления**

| **Шаг** | **Описание шага** | **Примечание** |
| --- | --- | --- |
| **1** | CRM отправляет запрос на отправку уведомления в сервис Push-уведомлений | Запрос содержит идентификатор клиента и текст уведомления |
| **2** | Сервис Push-уведомлений проверяет разрешение клиента | Разрешение отсутствует в базе данных |
| **3** | Сервис Push-уведомлений возвращает ошибку 403 в CRM | CRM уведомляется о невозможности отправки |

### 

#### **Альтернативный сценарий 2: Ошибка доставки уведомления**

| **Шаг** | **Описание шага** | **Примечание** |
| --- | --- | --- |
| **1** | CRM отправляет запрос на отправку уведомления в сервис Push-уведомлений | Запрос поступает в сервис |
| **2** | Сервис Push-уведомлений отправляет уведомление через FCM | Попытка отправки уведомления |
| **3** | FCM возвращает ошибку доставки | Ошибка доставки по сети или с устройством клиента |
| **4** | Статус ошибки записывается в базу данных | Логируется ошибка в таблице DeliveryStatus |
| **5** | Сервис Push-уведомлений отправляет событие об ошибке в Kafka | Логирование ошибки в топике push\_error\_log |

### 

#### **Альтернативный сценарий 3: Ошибка при подключении к FCM**

| **Шаг** | **Описание шага** | **Примечание** |
| --- | --- | --- |
| **1** | CRM отправляет запрос на отправку уведомления в Сервис Push-уведомлений | Запрос поступает как в основном сценарии |
| **2** | Сервис Push-уведомлений пытается отправить уведомление через FCM | Попытка подключения к Firebase Cloud Messaging |
| **3** | Происходит ошибка соединения | FCM временно недоступен |
| **4** | Сервис Push-уведомлений сохраняет статус ошибки в базу данных | Ошибка записывается в таблице DeliveryStatus |
| **5** | Сервис Push-уведомлений отправляет событие ошибки в Kafka | Логирование ошибки |
| **6** | Сервис инициирует повторную попытку подключения | Повторная попытка через определённый интервал |

### 