

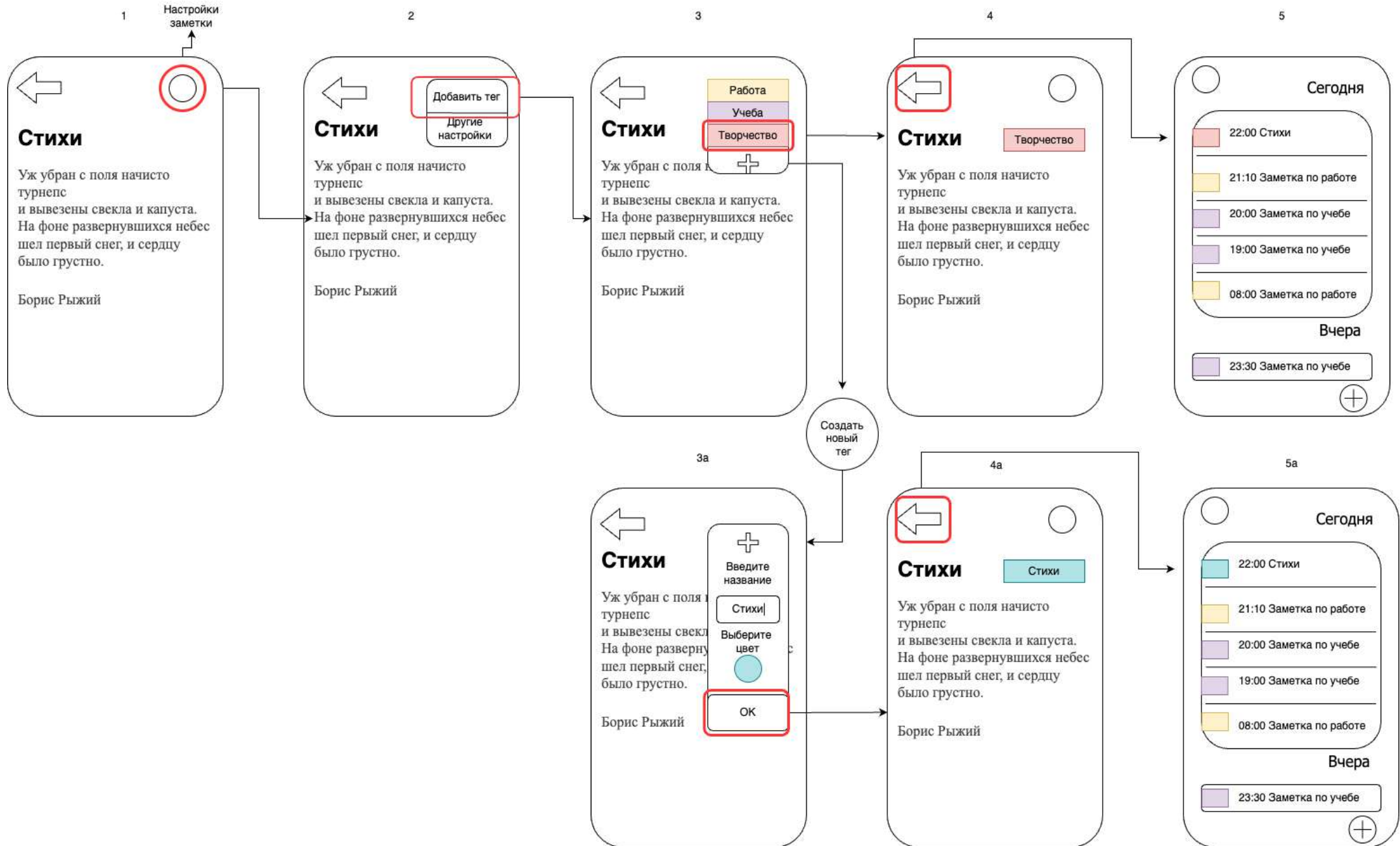
1. Название продукта

Заметки

2. User story:

Я, как пользователь приложения "Заметки", хочу добавлять теги. Чтобы упорядочить заметки по собственным категориям.

3. Макет



4. Use case

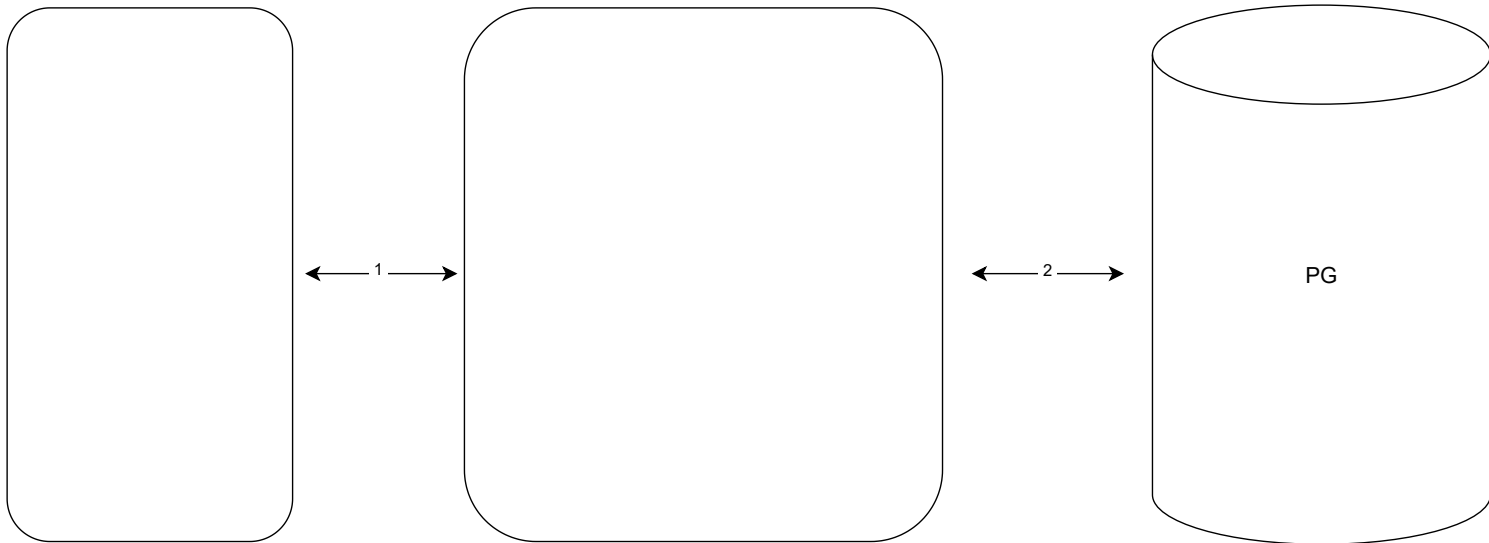
Заголовок	Добавление тега в уже созданную заметку
Акторы	Пользователь
Предусловие	Пользователь находится на экране конкретной заметки
Ограничения	1. Не больше одного тега для одной заметки 2. Название тега должно быть уникальным
Триггер	Пользователь нажимает кнопку "Настройки" на экране конкретной заметки (экран 1)
Основной сценарий	1. Система отображает выбранную заметку и меню ее настроек 2. Пользователь нажимает на меню настроек 3. Система отображает меню настроек этой заметки 4. Пользователь выбирает в меню настроек заметки "добавить тег" 5. Система отображает меню выбора тега 6. Пользователь выбирает тег 7. Система отображает выбранный тег внутри заметки 8. Пользователь возвращается на экран со списком созданных заметок 9. Система отображает присвоенный тег заметке в списке заметок
Альтернативный сценарий	6а. Пользователь нажимает в меню выбора тега "создать тег" 7а. Система отображает экран ввода данных для нового тега 8а. Пользователь вводит данные 9а. Система проверяет уникальность тега и отображает новый тег в меню настроек заметки - переход к 6 шагу основного сценария
Исключительный сценарий	8б. Пользователь вводит название тега, который уже существует 9б. Система проверяет уникальность тега и сообщает о том, что такой тег уже есть и нужно создать другой
Результат	Заметке успешно присвоен тег, в поиске по выбранному тегу отображается добавленная заметка

A

B

5. Архитектура

C



A - Фронтенд, мобильное приложение

B - Бэкэнд

C - База данных, PG

6. Модель данных

Объект:
Пользователь

Родительская сущность	Атрибут	Описание
Пользователь	Имя	Имя или ник
	e-mail	Почтовый адрес и связанное с ним облачное хранилище
	Гендер	Местоимение, которым приложение будет обращаться к пользователю
	Тег	Дочерняя сущность
	Заметка	Дочерняя сущность

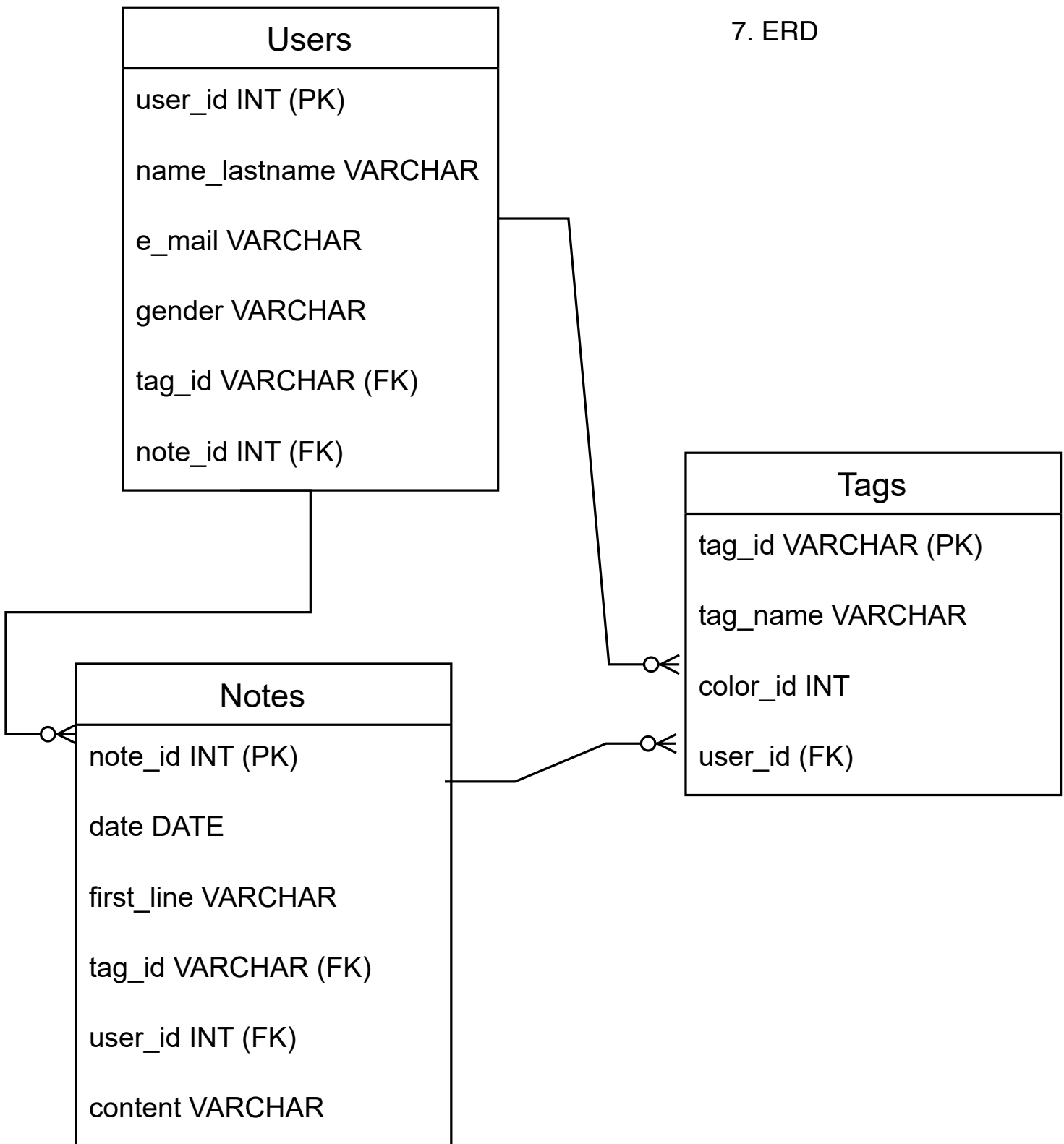
Объект:
Тег

Дочерняя сущность	Атрибут	Описание
Тег	Название	Название тега
	Цвет	Цвет, присвоенный тегу
	Пользователь	Родительская сущность

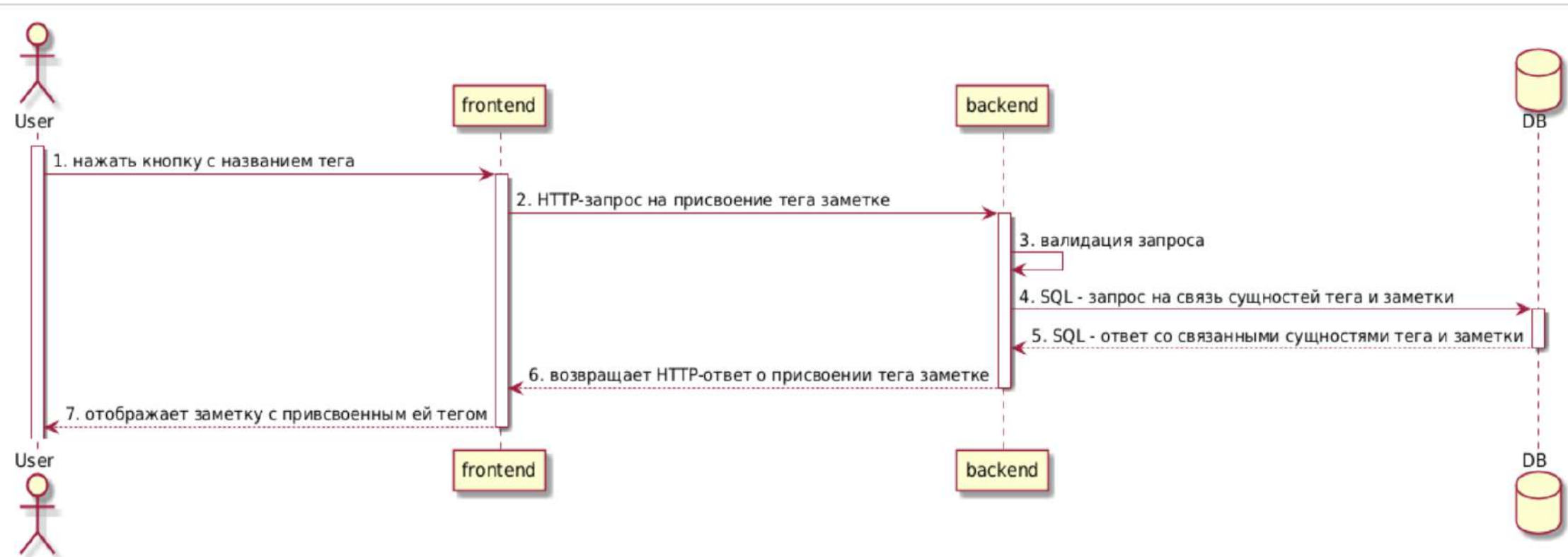
Объект:
Заметка

Родительская сущность	Атрибут	Описание
Заметка	Название	В случае отсутствия пользовательского названия заметке присваивается название в виде первых пяти слов заметки
	Дата и время создания	
	Тег	Дочерняя сущность
	Пользователь	Родительская сущность
	Содержание заметки	

7. ERD



8. Sequence diagram



Номер	Описание
1	Нажать кнопку с названием тега
2	HTTP-запрос на присвоение тега заметке
3	Валидация запроса
4	SQL - запрос на связь сущностей и заметки
5	SQL - ответ со связанными сущностями тега и заметки
6	Возвращает HTTP - ответ о присвоении тега заметке
7	Отображает заметку с присвоенным ей тегом

Присвоить заметке тег

POST /notes

Request

Название параметра		Находится в	Тип данных	Описание	Обязательность
note_id		body	int	Идентификатор заметки	Да
tags		body	array	Массив тегов со всеми параметрами	Да
	tag_id	body	int	Уникальный идентификатор тега	Да

Response 201 Created

Тело ответа пустое

Название параметра	Находится в	Тип данных	Описание	Обязательность

10. Swagger

```
1 openapi: 3.0.3
2 info:
3   title: Notes
4   description: I-
5     Made for SA course
6   license:
7     name: Apache 2.0
8     url: http://www.apache.org/licenses/LICENSE-2.0.html
9   version: 1.0.11
10 servers:
11   - url: https://notes.swagger.io/notes
12 tags:
13   - name: notes
14     description: Everything about notes
15
16
17 paths:
18   /notes:
19     post:
20       tags:
21         - notes
22       summary: Tag a note
23       description: User assigns a tag to a note
24       requestBody:
25         required: true
26         content:
27           application/json:
28             schema:
29               $ref: '#/components/schemas/NotesTag'
30       responses:
31         '200':
32           description: Successful response with no content
33
34
35
36 components:
37   schemas:
38     NotesTag:
39       type: object
40       properties:
41         note_id:
42           type: integer
43           description: Unique note id
44           example: '5'
45       tags:
46         type: array
47         items:
48           type: object
49           properties:
50             tag_id:
51               type: integer
52               description: Unique tag identifier
53               example: 1
```

Notes 1.0.11 OAS 3.0

Made for SA course

[Apache 2.0](#)

Servers

<https://notes.swagger.io/notes> ▼

notes Everything about notes

POST /notes Tag a note

User assigns a tag to a note

Parameters

Try it out

No parameters

Request body required

[application/json](#) ▼

Example Value Schema

```
{
  "note_id": 5,
  "tags": [
    {
      "tag_id": 1
    }
  ]
}
```

Responses

Code	Description	Links
200	Successful response with no content	No links

11. Критерии приемки

US: Я, как пользователь приложения Заметки, хочу добавлять теги.
Чтобы упорядочить заметки по собственным категориям.

Кейс №1

Функциональность: Создание нового тега

Дано: Клиент заполняет форму создания тега

Когда: Клиент вводит уже существующее имя тега в поле "Введите название" на экране 0

Тогда: Система выводит сообщение об ошибке "Это название уже используется"

Экран 0

The screenshot shows a mobile application interface. On the left, there is a list of notes under the heading "До пятни". The notes include: "Договориться о вре встречи 6 февраля могли Катя и Рома)", "СЕКРЕТАРЕМ", "Сходить к Павле", "Подготовить новые выгрузке на платф", and "Перенести колонку брали на НГ". On the right, there is a modal overlay for creating a new tag. It has a title "Введите название" with a plus icon, a text input field containing "Работа", a label "Выберите цвет" with a pink circle, and an "OK" button.

Кейс №2

Функциональность: Создание нового тега

Дано: Клиент заполняет форму создания тега

Когда: Клиент пытается сохранить новый тег не заполнив форму "Введите название" на экране 0

Тогда: Система выводит сообщение об ошибке "Введите, пожалуйста, название"

Экран 0

This screenshot is identical to the one in the previous block, showing the same list of notes and the tag creation modal. However, the text input field for the tag name is now empty, reflecting the state where the user has not yet entered a name.

Нефункциональные требования

Требования надежности:

1. Система должна быть доступна 99% времени.

Требования производительности:

1. Список доступных тегов должен открываться не более 2 секунд.
2. Запрос присвоения заметке тега POST/notes должен выдерживать нагрузку 1 rps.