

# High level design для сайта-блога с постами и комментариями

## Table of Contents

1. Стэк технологий .....	2
2. Схема классов back .....	2
3. Схемы бд .....	2
3.1. User .....	3
3.2. Post .....	3
3.3. Comment .....	3
3.4. Notification .....	4
3.5. PostToUserLike .....	4
3.6. UserToSubscribedUser .....	4
4. API back .....	5
4.1. Security .....	5
4.1.1. Авторизация .....	5
4.1.2. Регистрация .....	5
4.2. Профиль пользователя .....	6
4.2.1. Получение своего профиля .....	6
4.2.2. Получение чужого профиля .....	7
4.2.3. Получение профилей по логину .....	7
4.2.4. Изменение своего профиля .....	8
4.3. Пост .....	8
4.3.1. Создание поста .....	9
4.3.2. Получение всех своих постов .....	9
4.3.3. Получение всех постов пользователя .....	10
4.3.4. Получение постов фиды новостей .....	11
4.3.5. Получение поста .....	11
4.3.6. Изменение поста .....	12
4.3.7. Удаление поста .....	13
4.4. Комментарий .....	13
4.4.1. Получение комментариев для поста .....	13
4.4.2. Создание комментария .....	14
4.5. Подписка .....	15
4.5.1. Подписка/отписка на/от пользователя .....	15
4.6. Отметка "Нравится" .....	15
4.6.1. Добавление/убирание отметки "Нравится" .....	15
4.7. Уведомление .....	16

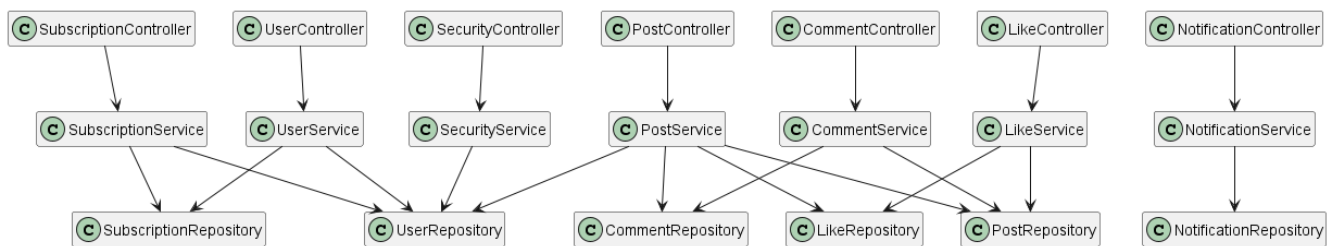
# 1. Стэк технологий

Front: TODO

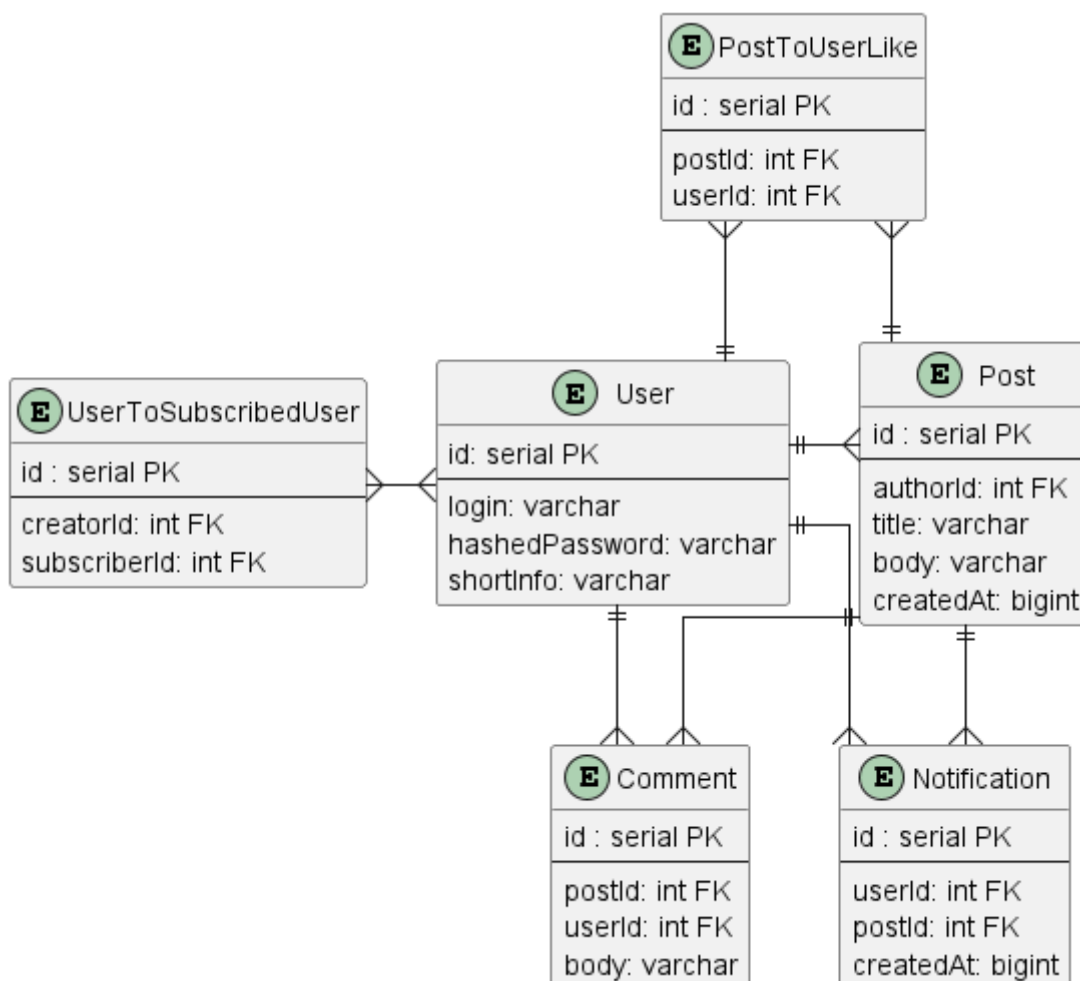
Back: Java, Spring Data/Security/MVC/Core/Boot, Maven

Остальное: Postgres, Docker, Github, Github Projects

## 2. Схема классов back



## 3. Схемы бд



## 3.1. User

Таблица профилей пользователей. Содержит логин, хешированный пароль и краткую информацию пользователя

Название поля	Тип	Ограничения
id	serial	PK - unique, not null
login	varchar	unique, not null, длина не меньше 8 и не больше 50 символов
hashedPassword	varchar	not null
shortInfo	varchar	not null, длина не меньше 1 символа и не больше 250 символов

## 3.2. Post

Таблица постов. Содержит ссылку на автора поста, название, тело поста и таймстемп создания поста в utc миллисекундах

Название поля	Тип	Ограничения	Комментарий
id	serial	PK - unique, not null	-
authorId	int	FK, not null	-
title	varchar	not null, длина не меньше 1 символа и не больше 120 символов	-
body	varchar	not null, длина не меньше 1 символа и не больше 600 символов	-
createdAt	bigint	not null	таймстемп utc в миллисекундах

## 3.3. Comment

Таблица комментариев - содержит текст комментария, ссылку на пост, под которым оставлен комментарий, и ссылку на автора, который оставил комментарий

Название поля	Тип	Ограничения
id	serial	PK - unique, not null
userId	int	FK, not null

Название поля	Тип	Ограничения
postId	int	FK, not null
body	varchar	not null, длина не меньше 1 символа и не больше 600 символов

## 3.4. Notification

Таблица уведомлений. Содержит ссылку на пользователя, которому предназначается уведомление, ссылку на пост, который вызвал создание уведомления, и таймстемп создания уведомления в utc миллисекундах. Уведомления, для которых  $createdAt + 30 \text{ дней} < \text{текущий utc таймстемп}$ , удаляются.

Название поля	Тип	Ограничения	Комментарий
id	serial	PK - unique, not null	-
userId	int	FK, not null	-
postId	int	FK, not null	-
createdAt	bigint	not null	таймстемп utc в миллисекундах

## 3.5. PostToUserLike

Таблица отметок "нравится". Содержит ссылку на пост, на который поставлена отметка, и ссылку на пользователя, поставившего отметку. Содержит уникальный индекс на {userId, postId}

Название поля	Тип	Ограничения
id	serial	PK - unique, not null
userId	int	FK, not null
postId	int	FK, not null

## 3.6. UserToSubscribedUser

Таблица подписок. Содержит ссылку на пользователя, на которого подписались, и ссылку на пользователя, который подписался. Содержит уникальный индекс {creatorId, subscriberId}

Название поля	Тип	Ограничения	Комментарий
id	serial	PK - unique, not null	-
creatorId	int	FK, not null	идентификатор пользователя, на которого подписаны

Название поля	Тип	Ограничения	Комментарий
subscriberId	int	FK, not null	идентификатор пользователя, который подписан

## 4. API back

### 4.1. Security

#### 4.1.1. Авторизация

POST /api/v1/security/auth

##### Входные данные

```
{
  "login": "string",
  "password": "string"
}
```

##### Выходные данные

###### Тело

```
{
  "token": "string"
}
```

##### Коды ошибок

Код	Причины
422	ошибка валидации
400	логин не найден, пароль не совпал

#### 4.1.2. Регистрация

POST /api/v1/security/register

##### Входные данные

```
{
  "login": "string",
  "password": "string",
  "shortInfo": "string"
}
```

```
}
```

## Выходные данные

### Тело

```
{  
  "token": "string"  
}
```

### Коды ошибок

Код	Причины
422	ошибка валидации
400	логин не уникален

## 4.2. Профиль пользователя

### 4.2.1. Получение своего профиля

GET /api/v1/user/me

### Входные данные

-

### Выходные данные

### Тело

```
{  
  "login": "string",  
  "shortInfo": "string",  
  "subscriptions": [  
    {  
      "id": 1,  
      "login": "string"  
    }  
  ]  
}
```

### Коды ошибок

Код	Причины
401	запрашивающий пользователь неавторизован

### 4.2.2. Получение чужого профиля

GET /api/v1/user/id/{id}

где {id} - идентификатор пользователя

#### Входные данные

-

#### Выходные данные

##### Тело

```
{
  "id": 1,
  "login": "string",
  "shortInfo": "string",
  "subscribed": true
}
```

#### Коды ошибок

Код	Причины
400	идентификатор запрашиваемого пользователя совпадает с идентификатором запрашивающего
401	запрашивающий пользователь неавторизован
404	пользователь не найден

### 4.2.3. Получение профилей по логину

GET /api/v1/user/login/{login}

где {login} - логин пользователя. В ответ не включается профиль запрашивающего пользователя

#### Входные данные

-

## Выходные данные

### Тело

```
[
  {
    "id": 1,
    "login": "string",
    "shortInfo": "string",
    "subscribed": true
  }
]
```

### Коды ошибок

Код	Причины
401	запрашивающий пользователь неавторизован

## 4.2.4. Изменение своего профиля

PUT /api/v1/user

### Входные данные

```
{
  "shortInfo": "string",
  "password": "string"
}
```

## Выходные данные

### Тело

-

### Коды ошибок

Код	Причины
422	ошибка валидации
401	запрашивающий пользователь неавторизован

## 4.3. Пост



### 4.3.1. Создание поста

POST /api/v1/post

#### Входные данные

```
{
  "title": "string",
  "body": "string"
}
```

#### Выходные данные

Тело

-

#### Коды ошибок

Код	Причины
422	ошибка валидации
401	запрашивающий пользователь неавторизован

### 4.3.2. Получение всех своих постов

GET /api/v1/post/filter/mine?page={page}&size={size}&sort={sort}

где {page}, {size}, {sort} - параметры пагинации

#### Входные данные

-

#### Выходные данные

Тело

```
[
  {
    "id": 1,
    "title": "string",
    "body": "string",
    "likeCounter": 1,
    "hitLike": true
    "firstComments": [
      {
        "id": 1,
        "authorLogin": "string",
```

```
    "body": "string"
  }
]
]
```

#### Коды ошибок

Код	Причины
401	запрашивающий пользователь неавторизован

### 4.3.3. Получение всех постов пользователя

GET /api/v1/post/user/{id}?page={page}&size={size}&sort={sort}

где {page}, {size}, {sort} - параметры пагинации, {id} - идентификатор пользователя

#### Входные данные

-

#### Выходные данные

##### Тело

```
[
  {
    "id": 1,
    "title": "string",
    "body": "string",
    "likeCounter": 1,
    "hitLike": true
    "firstComments": [
      {
        "id": 1,
        "authorLogin": "string",
        "body": "string"
      }
    ]
  }
]
```

#### Коды ошибок

Код	Причины
401	запрашивающий пользователь неавторизован

Код	Причины
404	пользователь не найден

#### 4.3.4. Получение постов фида новостей

GET /api/v1/post/filter/feed?page={page}&size={size}&sort={sort}

где {page}, {size}, {sort} - параметры пагинации

##### Входные данные

-

##### Выходные данные

##### Тело

```
[
  {
    "id": 1,
    "title": "string",
    "body": "string",
    "authorId": 1,
    "authorLogin": "string",
    "createdAt": 1111111111,
    "likeCounter": 1,
    "hitLike": true
    "firstComments": [
      {
        "id": 1,
        "authorLogin": "string",
        "body": "string"
      }
    ]
  }
]
```

##### Коды ошибок

Код	Причины
401	запрашивающий пользователь неавторизован

#### 4.3.5. Получение поста

GET /api/v1/post/{id}

где {id} - идентификатор поста

## Входные данные

-

## Выходные данные

### Тело

```
{
  "title": "string",
  "body": "string",
  "createdAt": 1111111111,
  "authorId": 1,
  "authorLogin": "string",
  "likeCounter": 1,
  "hitLike": true,
  "isAuthor": true
}
```

### Коды ошибок

Код	Причины
401	запрашивающий пользователь неавторизован
404	пост не найден

## 4.3.6. Изменение поста

PUT /api/v1/post/{id}

где {id} - идентификатор поста

## Входные данные

```
{
  "title": "string",
  "body": "string"
}
```

## Выходные данные

### Тело

-

### Коды ошибок

Код	Причины
422	ошибка валидации
401	запрашивающий пользователь неавторизован
403	попытка изменить пост от пользователя, не являющегося автором поста
404	пост не найден

### 4.3.7. Удаление поста

**DELETE** /api/v1/post/{id}

где {id} - идентификатор поста

#### Входные данные

-

#### Выходные данные

##### Тело

-

#### Коды ошибок

Код	Причины
401	запрашивающий пользователь неавторизован
403	попытка удалить пост от пользователя, не являющегося автором поста
404	пост не найден

## 4.4. Комментарий

### 4.4.1. Получение комментариев для поста

**GET** /api/v1/post/{id}/comment?page={page}&size={size}&sort={sort}

где {page}, {size}, {sort} - параметры пагинации, {id} - идентификатор поста

#### Входные данные

-

## Выходные данные

### Тело

```
[
  {
    "id": 1,
    "authorLogin": "string",
    "body": "string"
  }
]
```

### Коды ошибок

Код	Причины
401	запрашивающий пользователь неавторизован
404	пост не найден

## 4.4.2. Создание комментария

POST /api/v1/post/{id}/comment

где {id} - идентификатор поста

### Входные данные

```
{
  "body": "string"
}
```

## Выходные данные

### Тело

-

### Коды ошибок

Код	Причины
401	запрашивающий пользователь неавторизован
404	пост не найден

## 4.5. Подписка

### 4.5.1. Подписка/отписка на/от пользователя

POST /api/v1/subscription/{userId}

где userId - идентификатор пользователя, на которого нужно подписаться/от которого нужно отписаться

#### Входные данные

-

#### Выходные данные

##### Тело

```
{  
  "subscribed": true  
}
```

#### Коды ошибок

Код	Причины
401	запрашивающий пользователь неавторизован
404	пользователь не найден

## 4.6. Отметка "Нравится"

### 4.6.1. Добавление/убирание отметки "Нравится"

POST /api/v1/post/{id}

где {id} - идентификатор поста

#### Входные данные

-

#### Выходные данные

##### Тело

```
{  
  "hitLike": true  
}
```

#### Коды ошибок

Код	Причины
401	запрашивающий пользователь неавторизован
404	пост не найден

## 4.7. Уведомление

### 4.7.1. Получение своих уведомлений

GET /api/v1/notification

#### Входные данные

-

#### Выходные данные

##### Тело

```
[
  {
    "id": 1,
    "postId": 1
  }
]
```

#### Коды ошибок

Код	Причины
401	запрашивающий пользователь неавторизован