

# Лабораторная работа №2

---

## Основное задание

Спроектировать и разработать систему для анонимного общения в сети интернет.

Интерфейс системы должен представлять собой веб-страницу с лентой заметок, отсортированных в обратном хронологическом порядке и форму добавления новой заметки. В ленте отображаются последние 100 заметок.


Возможности:

1. Добавление текстовых заметок в общую ленту
2. Реагирование на чужие заметки (лайк, дизлайк)
3. Добавление комментариев
4. Добавление изображений к заметкам

## 1. Создание пользовательского интерфейса и описание пользовательских сценариев работы

← → ↺ ⬆ 127.0.0.1:8000

Купить молоко



👍 1

👎 0

Введите комментарий

Отправить

Комментарии:

Хорошая идея!

Добавить заметку:

Выберите файл

Файл не выбран

Введите название

Сохранить

---

4

**На сайте пользователю доступны следующие возможности:**

Добавление текстовых заметок в общую ленту

Реагирование на чужие заметки (лайк, дизлайк)

Добавление комментариев

Добавление изображений к заметкам

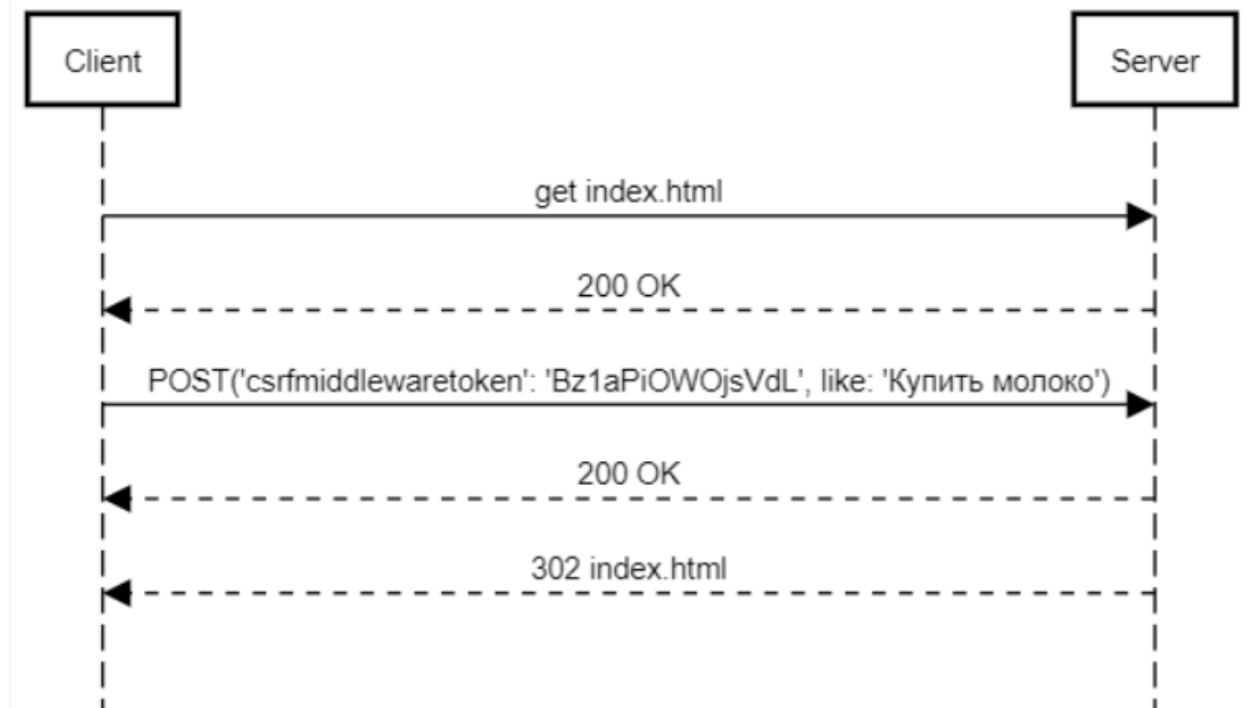
При создании новой заметки пользователь может создать ее как с фотографией, так и без нее.

## 2.Описание API сервера и хореографии

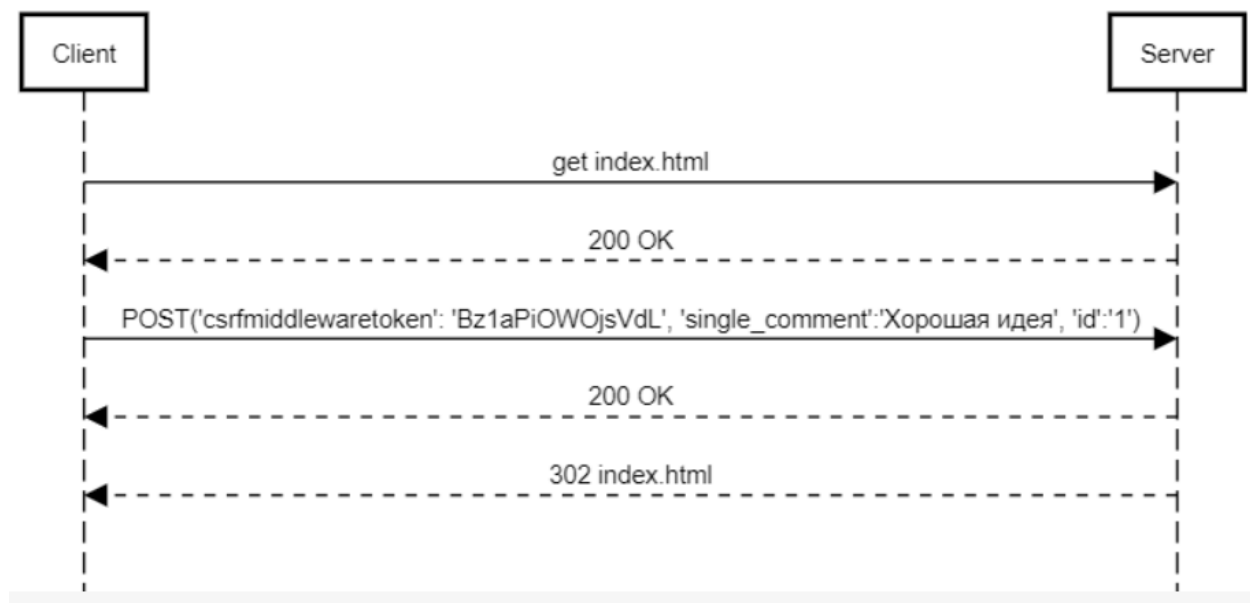
Добавление новой заметки:



## Добавление лайка/дизлайка



## Добавление комментария



### 3. Описание структуры базы данных

В качестве базы данных используется `sqlite`. Имеются 2 таблицы.

В первой хранятся заметки, их фото и количество лайков с дизлайками. Модель описана следующим образом:

```
class notes(models.Model):
    title = models.CharField(max_length = 100)
    image = models.FileField(upload_to='photos/', blank=True)
    like = models.IntegerField(blank=True, default=0)
    dislike = models.IntegerField(blank=True, default=0)
```

	id	title	like	dislike	image
	Фи...	Фильтр	Фи...	Фильтр	Фильтр
1	16	Купить ...	8	7	photos/451701-PhotoRoom.png

Комментарии хранятся в отдельной таблице. Каждая запись хранит в себе `id` поста, с которым он связан и содержание комментария. Модель описана следующим образом.

```
class comment(models.Model):
    note = models.ForeignKey(notes, default=None, on_delete=models.CASCADE,
    verbose_name='Название товара')
    single_comment = models.CharField(max_length = 500)
```

	id	single_comment ▼ <sup>1</sup>	note_id
	Фи...	Фильтр	Фильтр
1	10	Хорошая заметка!	21

### 4. Описание алгоритмов

- Алгоритм действий пользователя

