|  |  |
| --- | --- |
| |  | | --- | | МИНОБРНАУКИ РОССИИ  Федеральное государственное бюджетное образовательное учреждение  высшего образования  **«МИРЭА – Российский технологический университет»**  **РТУ МИРЭА** | |

Институт Информационных Технологий

Кафедра корпоративных информационных систем (КИС)

**ОТЧЕТ**

по Лабораторной работе №3

по дисциплине

«Разработка программного обеспечения для корпоративных информационных систем»

Выполнил студент группы ИКБО-08-18 Валяев Д.А.

Принял ассистент Масленников В.В.

Выполнено «\_\_\_» \_\_\_\_\_\_\_\_\_2021 г.

Зачтено «\_\_\_» \_\_\_\_\_\_\_\_\_2021 г.

**Задание**

Используя любой набор фреймворков (фронтэнд, бэкэнд) создать простейший форум. Функциональные требования (динамическое обновление, пометка новых сообщений) и оформление использовать от лабораторной работы №2. Модифицировать БД для реализации функциональности форума.

Дополнительные задания (обязательно на автомат):

Реализовать счетчики «лайков» для отдельных сообщений.

Реализовать возможность безопасного использования html-оформления в сообщениях (можно через bb-code)

Реализовать удаление учетной записи с уничтожением личной информации их профиля, но сохранением всех отправленных сообщений.

**Содержание работы**

1. Реализация основного функционала
   1. Динамическое обновление
   2. Пометка новых сообщений
   3. Модификация базы данных
2. Реализация дополнительного функционала
   1. Счетчик лайков
   2. Оформление текста
   3. Удаление учетной записи

**Выполнение работы**

В качестве фреймворка для выполнения работы был выбран FastAPI, язык программирования Python 3.7 для серверной части и javascript дддля клиентской.

1. **Основной функционал**

1.1 Код для динамического обновления страницы при получении новых сообщений был взят из лабораторной работы №2 практически без изменений. В лабораторной работе №1 при удалении элемента происходила перезагрузка страницы, сейчас это происходит с использование javascript и XMLHttpRequest, что позволяет удалить элемент без перезагрузки страницы.

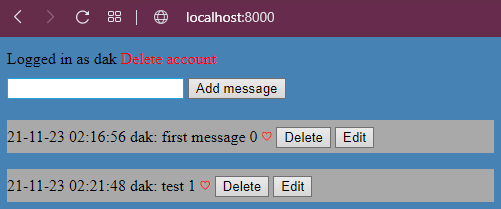


Рис 1.1 – Оформление страницы

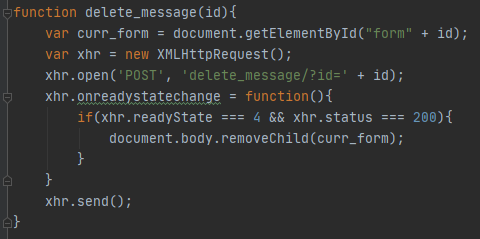


Рис 1.2 – Код удаления сообщения

На рисунке 1.2 можно видеть, как происходит удаление сообщения. Для начала находится элемент, который будет удален по id, затем создается запрос на сервер с указанием id удаляемого сообщения, в случае если ответ от сервера успешный, сообщение удаляется со страницы динамически.

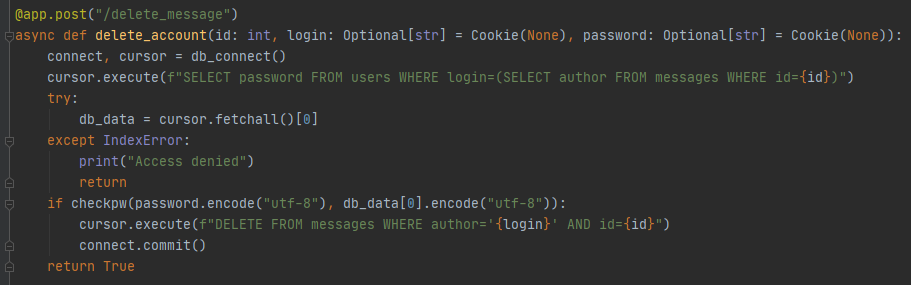


Рис 1.3 – Серверный код удаления сообщения

На рисунке 1.3 изображен код, который обрабатывает удаление сообщения из базы данных. Для начала происходит получение данных о том, что запрос отправил, если это владелец, тогда сообщение удаляется. Серверный код практически без изменений был взять из лабораторной работы №1.

1.2 Для оформления новых сообщений использовался псевдоелемент ::before и анимация, так же как и в лабораторной работе№2.

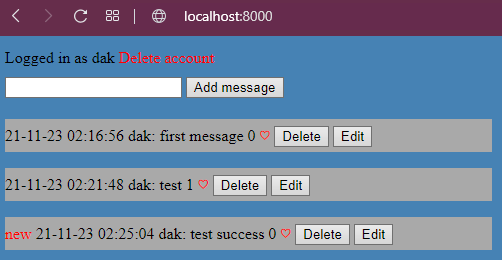


Рис 1.4 – Оформление новых сообщений

Чтобы разделить сообщения на новые и старые было добавлено 2 класса «message» и «messageNew», код оформления новых сообщений можно увидеть на рисунке 1.3.

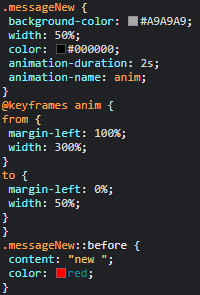


Рис 1.5 – Код оформления новых сообщений

1.3 Для выполнения лабораторной работы была модифицирована база данных. В таблице с сообщениями «messages» появилось поле «likes» целочисленного типа, для отображения лайков сообщения. Также чтобы один и тот же пользователь не мог ставить лайк на одну и ту же записть несколько раз, была создана таблица «marks» для хранения пары пользователь-запись.

1. **Дополнительный функционал**

Кроме возможности авторизации и добавления записи были реализованы дополнительные функции, такие как:

* 1. Для реализации возможность поставить лайк был написан код на языке программирования javascript с использованием XMLHttpRequest. Чтобы поставить лайк необходимо нажать на красное сердце рядом с сообщением.

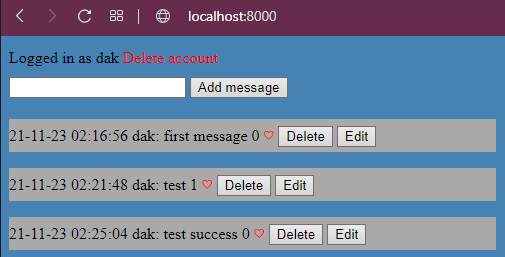


Рис 2.1 – Кнопка лайка

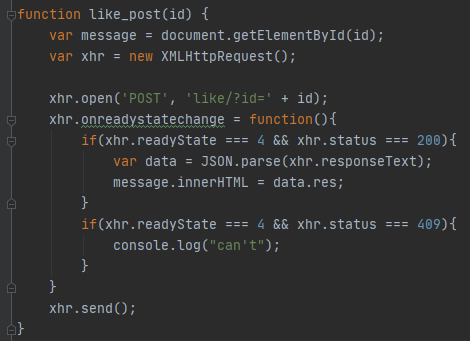


Рис 2.1 – Код для установки лайка на сообщение

На рисунке 2.1 можно видеть функцию отправки запроса на сервер для установки лайка на запись, для начала мы находим сообщение на странице, чтобы далее изменить его, после этого создается запрос на сервер с id записи, на которую необходимо установить оценку. При получении успешного ответа от сервера, сообщение обновляется динамически.

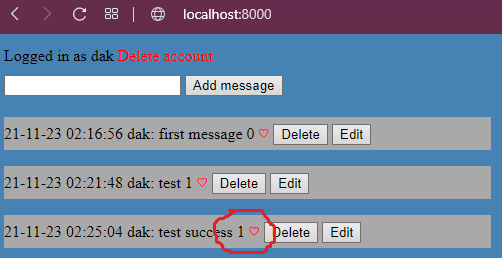


Рис 2.2 – Отображение количества лайков

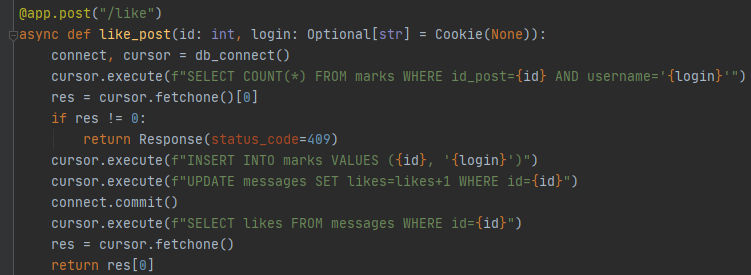


Рис 2.3 – Серверный код добавления лайка

На сервере происходит проверка ставил ли данный пользователь лайк на эту запись, если да, тогда возвращается ответ с кодом 409, если нет, то пара пользователь-запись записывается в таблицу «marks» и в таблице «messages» обновляется количество лайков.

2.2 Для оформления текста использовалась библиотека BBCodeHTML, которая, которая преобразует теги в HTML разметку, например, [b]text[/b] будет преобразовано в <strong>text</strong>.



Рис 2.4 – Вывод сообщений с использованием bbcode

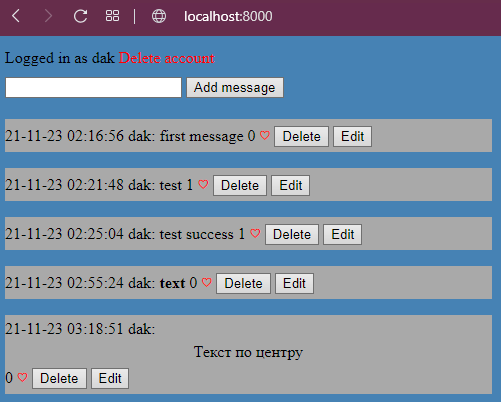


Рис 2.5 – Пример оформления сообщений

На рисунке 2.5 для оформления использовались теги [b][/b] – для четвертого сообщения и [center][/center] для пятого сообщения.

2.3 Также была реализована возможность удаления аккаунта, чтобы это сделать необходимо нажать красную кнопку «Delete account» рядом со своим именем пользователя, после чего высветится повторный вопрос, чтобы избежать случайных нажатий.

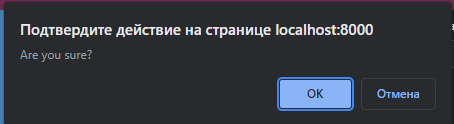


Рис 2.6 – Форма для подтверждения удаления

Если удаление аккаунта подтверждено, то выполнятся код внутри javasript функции, который вызывает функцию на сервере

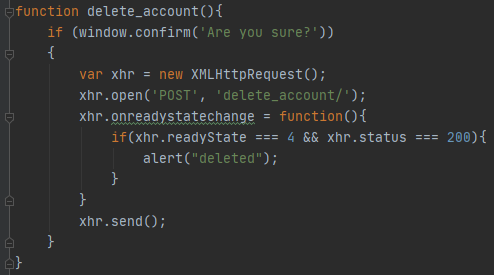


Рис 2.7 – Вызов серверной функции удаления аккаунта

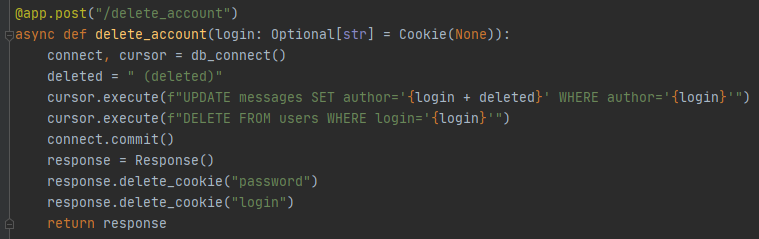


Рис 2.8 – Удаление аккаунта



Рис 2.9 – Отображение сообщений после удаления аккаунта

Функция удаления аккаунта, полностью стирает запись об аккаунте из таблицы «users» и обновляет автора сообщений в таблице «messages», на «имя пользователя (deleted)»

**Вывод**

В данной лабораторной работе были приобретены навыки работы с оформлением сообщений при помощи BBcode, динамическим обновлением контента страницы и работы с XMLHttpRequest.