

МИНОБРНАУКИ РФ
ФГБОУ ВО «ИжГТУ имени М.Т. Калашникова»
Кафедра «Программное обеспечение»

Отчет
по лабораторной работе №2
по дисциплине «Проектирование и конструирование
программного обеспечения»

Выполнил

студент гр. Б20-191-2:

Вотинцев Я.А.

Принял:

Еланцев М.О.

Ижевск 2023

Прототипы экранных форм

Форма авторизации пользователя. Используется пользователем для входа в систему. После корректности ввода данных пользователь получает доступ ко всем остальным функциям системы.

Вход

Введите email




Введите пароль

☐ Запомнить

Войти

[Зарегистрироваться?](#)

Войти с помощью:

The image is a wireframe of a login screen. It features a blue header with the word 'Вход' (Login) in white. Below the header, there are two input fields for 'email' and 'password'. A checkbox labeled 'Запомнить' (Remember me) is positioned below the password field. A blue button labeled 'Войти' (Login) is centered below the inputs. A link 'Зарегистрироваться?' (Sign up?) is located below the button. A horizontal line separates the login section from the social login section, which is headed 'Войти с помощью:' (Login with). Below this header are three circular icons for Facebook, VK, and Twitter.

Форма регистрации пользователя. Используется для создания нового аккаунта пользователя.

Регистрация

Email используется для подтверждения аккаунта

[Войти](#)

Форма публикации. Пользователь может просмотреть контент поста.

Заголовок заголовок заголовок

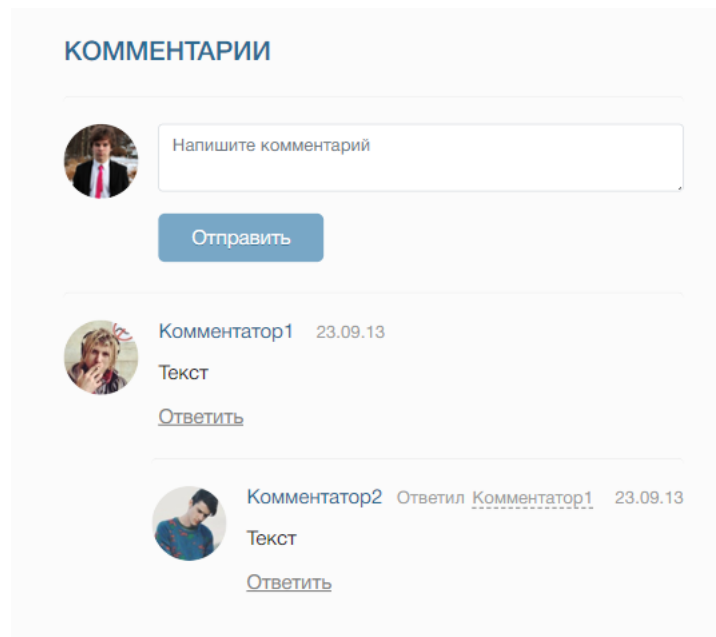


Иван Иванов

Текст поста. Текст текст текст текст текст текст текст

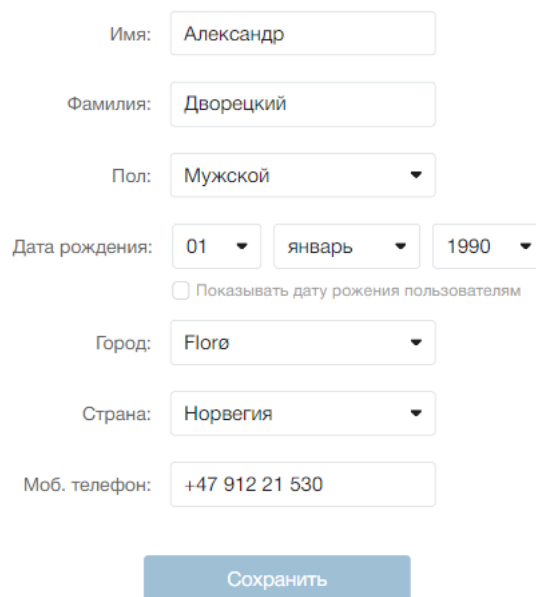
Ключевые слова: [дизайн](#), [простота](#),

Форма комментариев. Пользователь может просмотреть комментарии других пользователей или оставить свой.



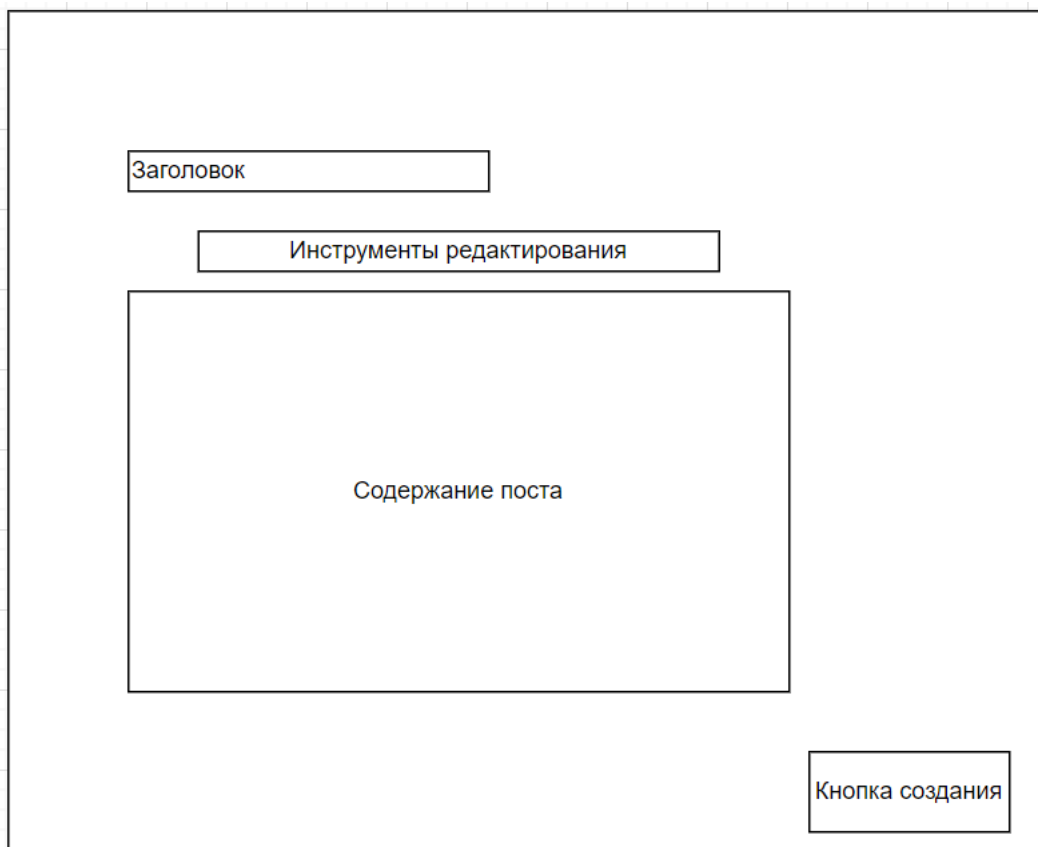
The screenshot shows a web interface for comments. At the top, there's a section titled "КОММЕНТАРИИ". Below it is a form for adding a new comment, featuring a user profile picture, a text input field with the placeholder "Напишите комментарий", and a blue "Отправить" button. Below the form is a list of existing comments. The first comment is from "Комментатор1" dated "23.09.13", with the text "Текст" and a link to "Ответить". The second comment is from "Комментатор2" dated "23.09.13", with the text "Текст" and a link to "Ответить". The second comment also includes a link to "Ответил Комментатор1".

Форма ввода данных профиля пользователя. Пользователь может указать и сохранить личную информацию для её отображения другим пользователям.



The screenshot shows a user profile form. It includes fields for "Имя:" (Alexandr), "Фамилия:" (Дворецкий), "Пол:" (Мужской), "Дата рождения:" (01, январь, 1990), "Город:" (Florø), "Страна:" (Норвегия), and "Моб. телефон:" (+47 912 21 530). There is a checkbox labeled "Показывать дату рождения пользователям" which is currently unchecked. A blue "Сохранить" button is at the bottom.

Форма создания поста. На ней пользователь может создать новую публикацию и добавить в неё контент.

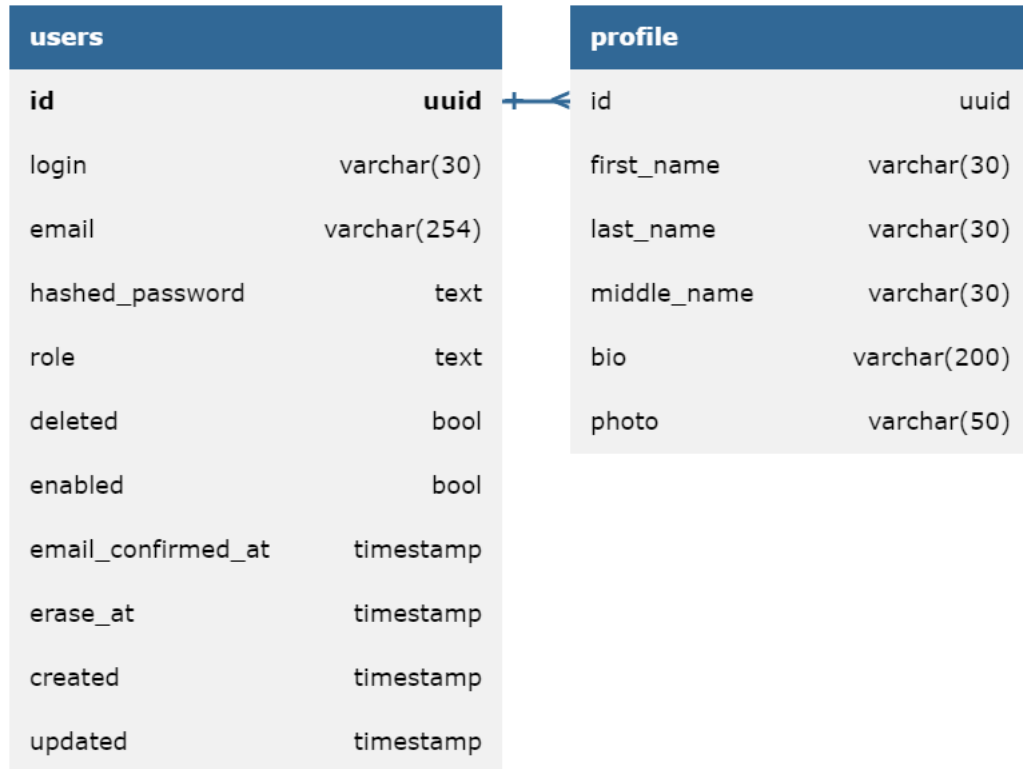


Форма журнала событий системы. Пользователь-администратор может увидеть логи всей системы.

Журнал событий					
ДАТА	УРОВЕНЬ	СЕРВИС	ПОЛЬЗОВАТЕЛЬ	ОПИСАНИЕ	ДЕЙСТВИЕ
События системы					

Диаграммы сущностей

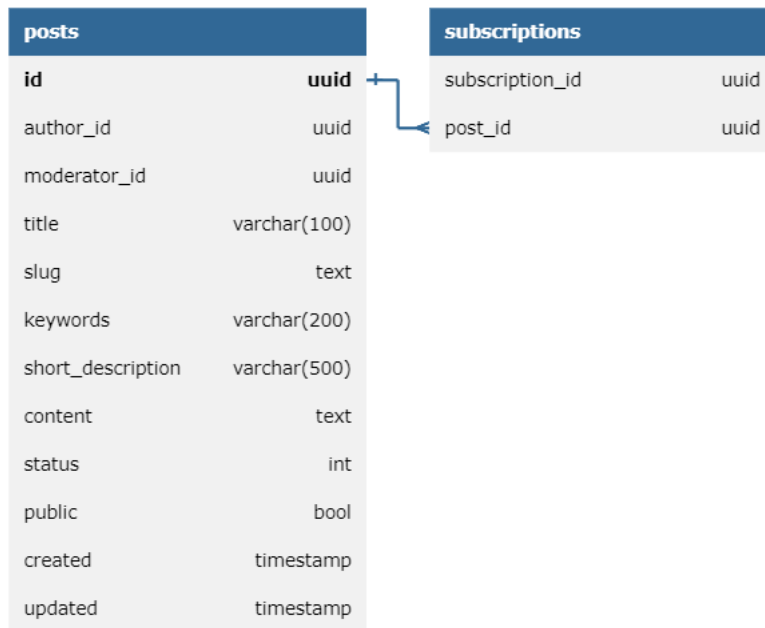
Сервис пользователей и авторизации:



Сервис логирования системы:

history_logs	
id	uuid
level	text
service	text
user_id	uuid
data	json
created	timestamp

Сервис постов:



Сервис комментариев

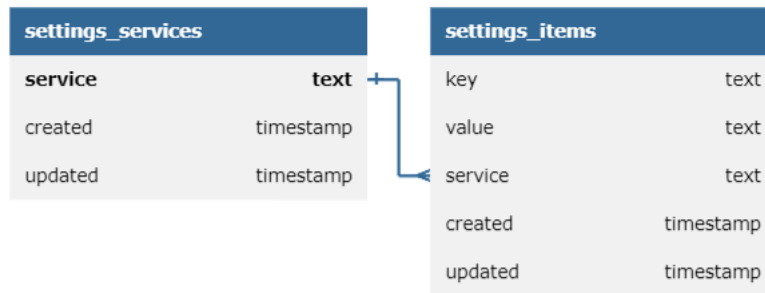
comments	
id	uuid
post_id	uuid
comment_id	uuid
author_id	uuid
content	text
created	timestamp
updated	timestamp

Сервис лайков

posts_likes	
id	uuid
post_id	uuid
user_id	uuid
removed	bool
created	timestamp
updated	timestamp

comment_likes	
id	uuid
post_id	uuid
comment_id	uuid
user_id	uuid
removed	bool
created	timestamp
updated	timestamp

Сервис конфигурации системы



Разработка API системы

1. Login. Этот метод принимает на вход логин и пароль пользователя. Если пользователь существует и пароль правилен – возвращает токен для входа в систему.

Входная информация: логин и пароль пользователя

Выходная информация: токен авторизации

2. User-info. Этот метод принимает на вход токен авторизации. Если проблем с токеном нет, то возвращает информацию о пользователе из токена.

Входная информация: токен авторизации

Выходная информация: базовая информация о пользователе

3. AllUsers/UserByID. Этот метод возвращает администратору список всех пользователей системы или пользователя по его ID.

Входная информация: токен авторизации.

Выходная информация: список всех пользователей системы.

4. CreateUser/UpdateUser. Этот метод принимает информацию о пользователе, которого нужно зарегистрировать или информацию для обновления существующего пользователя. Сервис обрабатывает информацию, валидирует её и создаёт запись в БД.

Входная информация: токен авторизации, информация о новом пользователе/новая информация о существующем пользователе.

Выходная информация: информация о создании/обновлении

5. Logs. Этот метод собирает всю накопленную информацию с других сервисов по совершенным HTTP-запросам и возвращает её.

Входная информация: токен авторизации, фильтры и параметры выборки логов.

Выходная информация: логи системы.

6. Settings. К этому методу все микросервисы обращаются с определённой периодичностью для того чтобы получить свои настройки.

Входная информация: код сервиса.

Выходная информация: список настроек сервиса.

7. Методы получения, изменения, создания, удаления списка сервисов из настроек. По аналогии работы с пользователями в сервисе конфигурации можно манипулировать информацией о текущих микросервисах проекта.

Входная информация: токен авторизации, имя сервиса, ключ и значение.

Выходная информация: информация о проделанной работе: получение/создание/обновление/удаление.

8. AllPosts/PostByID. Этот метод возвращает пользователю список постов.

Входная информация: токен авторизации, id поста

Выходная информация: список постов.

9. CreatePost. Этот метод создаёт новый пост.

Входная информация: токен авторизации, информация о новом посте

Выходная информация: сообщение о создании нового поста.

10. Comments. Этот метод возвращает комментарии поста.

Входная информация: токен авторизации, id поста.

Выходная информация: комментарии поста.

11. CreateComment. Этот метод создаёт новый комментарий поста от текущего пользователя.

Входная информация: токен авторизации, id поста, содержание комментария

Выходная информация: сообщение об успешности операции.

12.Likes. Этот метод возвращает оценки поста.

Входная информация: токен авторизации, id поста.

Выходная информация: количество лайков поста.

13.SetLike. Этот метод оставляет от текущего пользователя лайк или убирает его.

Входная информация: токен авторизации, id поста.

Выходная информация: сообщение об успешности операции.

Иерархическая структура работ



Время выполнения проекта по методу PERT

Для того, чтобы выполнить данную оценку, необходимо сначала определить количественный состав работ. Имеем на данный момент:

- Сущности БД: $N_{БД} = 15$
- Методы API: $N_{API} = 40$
- Тесты методов API: $N_T = 40$

Оценим количество часов для выполнения одной единицы работы (оптимистично O_i , пессимистично P_i и среднее M_i):

- Для одной сущности БД:
 - $O_{БД} = 1\text{ч}$
 - $P_{БД} = 6\text{ч}$
 - $M_{БД} = 2\text{ч}$
- Для одного метода API:
 - $O_{API} = 1\text{ч}$
 - $P_{API} = 12\text{ч}$
 - $M_{API} = 4\text{ч}$
- Для одного теста:
 - $O_T = 1\text{ч}$
 - $P_T = 12\text{ч}$
 - $M_T = 4\text{ч}$

Оценку средней трудоемкости по каждому элементарному пакету можно определить по формуле: $E_i = (P_i + 4M_i + O_i)/6$

Рассчитаем для проекта:

- $E_{БД} = (1 + 4*2 + 6)/6 = 2,5 \text{ чел.*час}$
- $E_{API} = (1 + 4*4 + 12)/6 = 4,84 \text{ чел.*час}$
- $E_T = (1 + 4*4 + 12)/6 = 4,84 \text{ чел.*час}$

Суммарная трудоемкость проекта может быть рассчитана по формуле: $E = \sum E_i N_i$

$$E = 2,5 * 15 + 4,84 * 40 + 4,84 * 40 = 427,7 \text{ чел.*час}$$

Для расчета среднеквадратичного отклонения используется формула:
 $CKO_i = (P_i - O_i)/6$

- $CKO_{\text{БД}} = (6 - 1)/6 = 0,83 \text{ чел.} \cdot \text{час}$
- $CKO_{\text{API}} = (12 - 1)/6 = 1,83 \text{ чел.} \cdot \text{час}$
- $CKO_{\text{T}} = (12 - 1)/6 = 1,83 \text{ чел.} \cdot \text{час}$

$$CKO = \sqrt{\sum N_i CKO_i^2} = 16,7 \text{ чел.} \cdot \text{час}$$

Тогда для оценки суммарной трудоемкости проекта, которую мы не превысим с вероятностью 95%, получим

$$E_{95\%} = E + 2 * CKO = 427,7 + 2 * 16,7 = 461,1$$

Так как кодирование составляет лишь 25% трудозатрат проекта, общая трудоемкость всего проекта будет:

$$E_{\text{total}} = 461 * 4 = 1844 \text{ чел.} \cdot \text{час.}$$

Так как человек работает не 100% своего времени над проектом, в месяц он будет работать не 168 часов, а примерно 132 часа. Тогда трудоемкость проекта в чел. * мес. Будет равна:

$$P_{\text{м}} = 1844 / 132 = 14 \text{ чел.} \cdot \text{мес.}$$

Базовое расписание в виде диаграммы Ганта

