**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ**

**РОССИЙСКОЙ ФЕДЕРАЦИИ**

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«МОСКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА ПО ПРОЕКТУ

«ВЕБ-ПРИЛОЖЕНИЕ ГИПП »

ПО ДИСЦИПЛИНЕ

«ПРОЕКТНАЯ ДЕЯТЕЛЬНОСТЬ»

ПРОЕКТ ВЫПОЛНЕН ПОД РУКОВОДСТВОМ

ДАНЬШИНОЙ МАРИНЫ ВЛАДИМИРОВНЫ  
СТУДЕНТАМИ ГРУППЫ 171-331,191-322,181-321

Ескин Никита Алексеевич

Голенко Вадим Владимирович

Гыждиян Дмитрий Сергеевич

Курмаз Анастасия Владиславна

Кузьменко Виктория Евгеньевна

Оспанов Сергей Александрович

Кожух Егор Константинович

Малиновская Ксения Валерьевна

# **Содержание**

**Содержание 1**

**1.Аннотация 3**

**2.Партнеры 4**

**3.План работы 5**

**4.Участники 8**

**5.Индивидуальные планы участников 10**

**6.Формирование требований к проекту 18**

6.1. Анализ предметной области, обзор современных источников 18

6.1.1. Описание организации предприятия 18

6.1.2. Общее описание предметной области 19

6.1.3. Особенности и нюансы выбранной темы 20

6.2. Анализ востребованности решения, формулировка проблемы 20

6.3. Анализ аналогов и технических решений 21

6.4. Формулирование целей и задач по проекту 23

6.5. [Исследование задачи с точки зрения использования приложения, CJM, описание целевой аудитории](https://docs.google.com/document/d/1t7BtNPZHZ4ihsv6niyLb82sNDypffapY/edit#heading=h.gjdgxs) 24

6.5.1. Классы и характеристики пользователей 24

6.5.2. Целевая аудитория 26

6.6. Бизнес процессы 27

**7. Реализация проекта 29**

7.1. Проектирование разделов, структура сайта. 29

7.2. Проектирование структуры базы данных 31

7.2.1.Основная база данных 31

7.2.2.Вспомогательная база данных для хранения файлов 32

7.3. Конкретная реализация задачи на серверной стороне 32

7.4. Конкретная реализация задачи на клиентской стороне 33

7.5. Аспекты информационной безопасности 36

7.6. Тестирование приложения 37

**8. Сопровождение проекта 38**

8.1. Администрирование проекта на сервере 38

8.2. Инструкция пользователя 39

# 

# 

# 

# 

# 

# 

# 

# **1.Аннотация**

К 2020 году и чрезвычайно быстрым развитием веб-технологий все современные цифровые СМИ нуждаются в создании частных новостных порталов. Если же у СМИ уже есть такой портал, ему необходимо продвижение, администрирование, постоянное тестирование качества работы и многое многое другое.

Было получено задание от ГИПП на создание такого веб-приложения, которые будет представлять из себя тот самый частный новостной портал, непосредственно для Гильдии Издателей Периодической Печати. Веб-приложение должно иметь удобный инструментарий для создания статей непосредственно на портале и их верстке, инструментарий для редактирования самого сайта, а также отображать все актуальные новости.

Главной целью в разработке данного веб-приложения для ГИПП было упростить работу тех изданий, которые находятся в других городах, привести их всех к единому формату. Если следовать этой цели и возможности с помощью качественного инструментария работать прямо на сайте, то в итоге должен получится равноценно удобный сайт как для издателей, так и для пользователей.

# 

# **2.Партнеры**

Партнерами при разработке проекта стали представители ГИПП, определяющие основные требования к проекту.

# 

# 

# 

# 

# 

# 

# 

# 

# **3.План работы**

|  |  |  |  |
| --- | --- | --- | --- |
| **№** | **Этап работы** | **Описание этапа** | **Дата окончания** |
| 1 | Выбор темы проектной деятельности | Выбор темы для проектной деятельности | 10.02.2020 |
| 2 | Сбор команды | Встреча участников команды для разработки проектной деятельности | 10.02.2020 |
| 3 | Встреча с заказчиком | Определение основных требований к проекту, обсуждение требований с заказчиком | 18.02.2020 |
| 4 | Разработка макета интерфейса | Разработка первоначального макета дизайна интерфейса веб-приложения | 20.03.2020 |
| 5 | Формирование технического задания | Формулировка технического задания к проекту, структурирование работы | 20.03.2020 |
| 6 | Согласование с заказчиком | Предоставление макета и разработанного технического задания заказчику |  |
| 7 | Доработка дизайн макетов | Доработка дизайн макетов интерфейса портала с учето | 1.05.2020 |
| 8 | Написание кода | Реализация поставленных заданий проекта. | 20.06.2020 |
| 9 | Создание базы данных | Добавление базы данных на портал | 20.05.2020 |
| 10 | Представление промежуточных результатов заказчику | Предоставление промежуточной версии веб-приложения заказчику, уточнение оставшихся работ, обсуждение проделанной работы | 5.06.2020 |
| 11 | Работа над ошибками | Исправление возможных ошибок в коде | 10.06.2020 |
| 12 | Тестирование | Проведение ручных тестов. Написание автоматизированныхтестов. Отладка кода с помощью вышеуказанных тестов. Рефакторинг по итогам отладки. | 15.06.2020 |
| 13 | Предоставление проекта заказчику | Предоставление проекта заказчику | 20.06.2020 |
| 14 | Составление документации | Составление пояснительной записки для проекта | 29.06.2020 |
| 15 | Подготовка презентации и видео | Подготовка презентации и видео для защиты проекта | 29.06.2020 |
| 16 | Защита проекта | Защита проектной деятельности | 1.07.2020 |

# **4.Участники**

* Ескин Никита 171-331 - подготовил Git репозиторий, составил план работы над проектом, собрал информацию, изучил, проанализировал, сравнил и выбрал лидирующую связку платформа-язык, подготовил документацию, разработал клиентскую часть приложения.
* Голенко Вадим 171-331 - ответственный за вспомогательный бекенд для хранения файлов(картинки, видео, документы и т.д.) . А именно за: настройку nginx; БД PostreSQL; создание структуры БД; создание архитектуры бэкенда; создание роутинга; настройку middleware бэкенда; создание контроллеров для соединения роутинга с управлением бд; развертывание и запуск бэкенда на сервере с использованием актуальных переменных окружения.
* Гыждиян Дмитрий 171-331 - спроектировал систему, отвечал за создание бд, создание таблиц, изучение работы с токенами, создание crud запросов для таблицы пользователей, проверку работы токенов, создание crud запросов для новостей.
* Кожух Егор 191-322 - помог в проектировании системы,проводил рефакторинг кода, изучение PHP и работы с токенами.
* Курмаз Анастасия 171-331 - работала над дизайном проекта. Разработала дизайн-макетов страницы авторизации, личного кабинета, пользовательского представления.
* Малиновская Ксения181-321 - разработала дизайн-макетов страницы добавления новостей, редактирования новостей.
* Оспанов Сергей 171-331 - производил тестирование,заполнение документаций тестирования, составление документаций для проекта,подготовка видеоролика и презентации проекта.
* Кузьменко Виктория171-331 - составление документаций для проекта, коммуникации с заказчиком, составление презентации для защиты проекта.

# 

# 

# 

# 

# 

# 

# 

# 

# **5.Индивидуальные планы участников**

## **Ескин Никита 171-331**

* Первичные консультации с заказчиком - 6 часов;
* Командное обсуждение проекта, создание бесед для проекта - 6 часов;
* Изучение ТЗ - 2 час;
* Создание базового проекта vue.js - 2 час;
* Подключение необходимых модулей (vue-modal, vue-select) и настройка vue-router - 1 час;
* Интеграция в проекта глобального хранилища vuex:
  + Изучение материалов по архитектуре и работе с хранилищем - 5 часов;
  + Интеграция хранилища в базовый проект - 1 час;
* Интеграция и настройка http-клиента Axios - 2 час ;
* Создание структуры компонентов и компонентов высшего порядка - 3 часа;
* Реализация редактора сайта:
  + Редактирование новостных рубрик - 3 часа;
  + Редактирование цветовой палитры сайта - 1 час;
  + Проверка и отладка редактора - 3 час;
* Система авторизации пользователя - 2 часа;
* Реализация добавления, изменения и удаления новости:
  + Выбор и тестирование сторонних текстовых редакторов - 4 часа;
  + Интеграция и настройка текстового редактора - 2 часа ;
  + Система хранения изображения, конвертация изображений в base64 - 3 часа;
  + Изменение статьи - 2 часа;
  + Удаление статьи - 2 час;
  + Проверка добавления, изменения и удаления - 3 час;
* Реализация кабинета администратора :
  + Разделение новостей по вкладкам “опубликовано”, “черновики” и “авторы” - 2 часа;
  + Доработка карточки новости - 2 часа;
* Реализация валидации форм:
  + Изучение материалов библиотеки vuelidate - 3 часа ;
  + Интеграция и настройка валидации - 2 часа ;
  + Проверка валидации - 1 час;
* Фильтрация новостей в личном кабинете - 3 часа;
* Отлавливание и исправление ошибок - 6 часов;

**Итого - 72 часа**

## **Вадим Голенко 171-331**

* **DevOps:**
  + Актуализация своих знаний с помощью актуальных документаций по всем примененным технологиям - 6 часов;
  + Настройка nginx - 2 часа;
  + Настройка PostrgeSQL, NodeJS, NPM, домен третьего уровня - 2 часа;
  + Реализация БД через ORM Sequelize:
    - Создание классов(таблиц) - 2 часа;
    - Настройка БД для конкретного юзера(выдача прав, создание БД и тд.) - 2 часа;
  + Перемещение продакшен кода на сервер - 1 час;
  + Запуск бекенда на сервере с актуальными ENV переменными - 1 час;
* **Архитектура:**
  + Получение ТЗ от членов команды и его изучение - 3 часа;
  + Изучение и выбор из различных типов архитектур - 4 часа;
  + Проектирование структуры базы данных - 1 час;
  + Проектирование бекенда:
    - Проектирование общей структуры - 2 часа;
    - Использование принципов архитектуры Ducks - итого 8 часов:
      * в роутинге - 2 часа;
      * в мидлварах - 2 часа;
      * в контроллерах - 2 часа;
      * в обработчиках запросов - 2 часа;
  + Проектирование системы хранения загруженных файлов - 2 часа;
* **Программирование:**
  + Актуализация своих знаний с помощью актуальных документаций по всем примененным технологиям - 6 часов;
  + Поиск и установка необходимых пакетов NPM - 1 час;
  + Инициализация шаблона для бэкенда согласно спроектированной архитектуре - 3 часа;
  + Программирование частей бэкенда - итого 12 часов:
    - в роутинге - 3 часа;
    - в мидлварах - 3 часа;
    - в контроллерах - 3 часа;
    - в обработчиках запросов - 3 часа;
* **Тестирование и отладка:**
  + Актуализация своих знаний с помощью актуальных документаций по всем примененным технологиям - 3 часа;
  + Отладка с помощью Postman - 5 часов;
  + Написание автотестов на Jest - 6 часов;

**Итого - 72 часа**

## **Гыждиян Дмитрий 171-331**

* Общение с заказчиком, согласование проекта с куратором - 2 часа;
* Первичные консультации с заказчиком - 6 часов;
* Командное обсуждение проекта- 6 часов;
* Составление ТЗ - 4 часа;
* Изучение работы авторизации через токен - 4 часа;
* Проектирование базы данных - 4 часа :
  + Создание макета таблиц - 3 часа;
  + Создание базы данных - 1 час;
* Создание файла роутинга запросов - 5 часов;
* Написание функций - 20 часов:
  + Функции авторизации - 4 часа;
  + Функции работы с пользователями - 4 часа;
  + Функции работы с новостями - 8 часов;
  + Проработка ошибок - 4 часа;
* Проверка работоспособности - 4 часа;
* Создание коллекции в Postman - 3 часа;
* Повторное обсуждение с командой - 6 часов;
* Рефакторинг кода - 10 часов;

**Итого - 72 часа**

## **Кузьменко Виктория** 171-331

* Первичные консультации с заказчиком - 6 часов;
* Командное обсуждение проекта, создание бесед для проекта - 6 часов;
* Исследование и анализ предметной области для документации - 3 часа;
* Изучение общей работы ГИПП для окончательного понимания предметной области - 3 часа;
* Анализ актуальности и востребованности темы - 4 часа;
* Анализ аналогов и технических решений для темы проекта - 4 часа;
* Анализ и описание целевой аудитории - 6 часов;
* Анализ бизнес процессов на портале - 6 часов;
* Описание реализации проекта для документации - 10 часов;
* Описание дальнейшего сопровождения проекта - 10 часов;
* Составление презентации для защиты проекта - 10 часов;
* Сбор дополнительной информации для составления документации и презентации - 4 часа;

**Итого - 72 часа**

## **Курмаз Анастасия** 171-331

* Первичные консультации с заказчиком - 6 часов;
* Командное обсуждение проекта, создание бесед для проекта - 6 часов;
* Исследование и анализ предметной области для разработки дизайна - 3 часа;
* Анализ стилистических решений и трендов дизайна - 3 часа;
* Анализ аналогов - 4 часа;
* Проработка UX стратегий - 6 часов;
* Разработка пользовательских сценариев в Figma – 4 часа;
* Анализ и выбор типографии – 2 часа;
* Проектирование интерфейса - 6 часов;
* Создание вайрфреймов - 10 часов;
* Разработка макетов - 10 часов;
* Создание постера для презентации проекта - 2 часа;
* Утверждение и согласование дизайна сайта – 4 часа;
* Валидации данных - 2 часа;
* Функции новостей и роутинг новостей - 4 часа;

**Итого - 72 часа**

## **Оспанов Сергей** 171-331

* Изучение дополнительного материала (для тестирования) - 10 часов;
* Подготовка чек-листа для тестирования - 6 часов;
* Командные консультации - 10 часов;
* Настройка среды для тестирования - 2 часа;
* Разработка базовых тестов - 6 часов;
* Тестирование модулей:
  + Меню - 1 час;
  + Личный кабинет - 2 часа;
  + Модуля авторизации - 2 часа;
  + Модуля добавления новостей - 4 часа;
  + Модуля комментариев - 4 часа;
  + Модуль социальных действий - 1 час;
  + Модуля фильтров - 2 часа;
* Тестирование ролей:
  + Администратора - 2 часа;
  + Модератора - 2 часа;
  + Редактора - 2 часа;
  + Пользователя - 2 часа;
* Тестирование мобильной версии веб приложения - 4 часа;
* Написание баг-репортов - 4 часа;
* Составление презентации для защиты проекта - 4 часа;
* Составление видеоролика для защиты проекта - 2 часа;

**Итого - 72 часа**

## **Кожух Егор** 191-322

* Общение с заказчиком, согласование проекта с куратором - 2 часа;
* Первичные коммуникации с заказчиком - 6 часов;
* Командное обсуждение проекта - 6 часов;
* Составление ТЗ - 6 часа;
* Изучение работы авторизации через токен - 6 часа;
* Изучение PHP - 20 часов;
* Повторение обсуждение с командой - 6 часов;
* Исправление ошибок кода - 10 часов;
* Рефакторинг кода - 10 часов;

**Итого - 72 часа**

## **Малиновская Ксения** 181-321

* Первичные консультации с заказником – 6 часов;
* Изучение общей работы ГИПП для разработки дизайна проекта - 4 часа;
* Знакомство с кейсом и техническим заданием – 2 часа;
* Командное обсуждение проекта - 6 часов;
* Анализ аналогов конкурентов и трендов, исследование стилей – 4 часа;
* Анализ целевой аудитории – 2 часа;
* Разработка UX стратегии и поиск креативных концепций – 10 часов;
* Разработка пользовательских сценариев в Figma – 13 часов;
* Утверждение и согласование прототипа сайта, внесение поправок – 4 часа;
* Анализ и выбор типографии – 4 часа;
* Создание дизайна страниц и визуального образа – 13 часов;
* Утверждение и согласование дизайна сайта – 4 часа.

**Итого - 72 часа**

# **6.Формирование требований к проекту**

## **6.1. Анализ предметной области, обзор современных источников**

### **6.1.1. Описание организации предприятия**

Союз предприятий печатной индустрии “Гильдия издателей периодической печати” - это проект, созданный издателями периодических печатных изданий в целях содействия своим множественным членам, которых уже порядка больше 200, в осуществлении деятельности, направленной на:

– объединение усилий организаций, работающих на российском рынке печати в том числе средств массовой информации, для защиты своих общих интересов и прав с законодательной точки зрения;

– экономическую, юридическую и социальную защиту прав и интересов издателей средств массовой информации, типографий и иных полиграфических предприятий;

– разработка и внедрение единой технической политики;

– развития и внедрения образовательных программ и стандартов в сфере издательского бизнеса;

– поддержание высокого стандарта знаний и содействие широкому внедрению передового отечественного и международного опыта подготовки специалистов в области издания и распространения средств массовой информации;

– сбор, анализ и распространение информации по проблемам и разработкам в области использования передовых технологий в полиграфии;

– содействие и оказания практической помощи членам ГИПП в широком внедрении прогрессивных разработок в области издательского бизнеса;

– поддержку перспективных в области полиграфии проектов и в установлении деловых контактов для членов ГИПП

–помощь в привлечении инвесторов, партнеров и кредитных ресурсов в российскую издательскую отрасль;

– тесное сотрудничество с международными и иностранными организациями, с целью распространения печатной продукции и принятие опыта друг у друга;

– разработку и внедрение постоянно действующей системы статистического наблюдения за издательской отраслью;

– развитие материально-технической базы ГИПП.

В результате описанных выше основных процессов деятельности ГИПП были выдвинуты следующие требования к веб-приложению:

1. Единое администрирование для всех новостных сайтов
2. Возможность для СМИ, которые до сих пор не были в интернете, упростить процесс создания сайта
3. Единая система всех сайтов, таким образом что они все выглядят примерно одинаково и смотрятся гармонично.

### **6.1.2. Общее описание предметной области**

На данный момент в системе прописана только одна роль - администратор. В веб-приложении администратор может создавать новостные записи, заниматься их версткой прямо в системе, имеет возможность редактировать сам сайт - настройка цветовой палитры веб-приложения и рубрик, ну и последнее на данный момент - добавление уже имеющихся новостей в черновики. Пользователь, который не вошел в систему может просматривать новости по рубрикам и делиться с ними в социальных сетях. В дальнейшем планируется добавление новых ролей, например такой роли как редактор, который будет иметь определенные ограничения в отличии от администратора.

### **6.1.3. Особенности и нюансы выбранной темы**

Данное веб-приложение разработано для упрощения работы Гильдии Издателей Периодической Печати, потому что в веб-приложении имеется единое администрирование всех имеющихся новостных порталов.

## **6.2. Анализ востребованности решения, формулировка проблемы**

Согласно статистике, интернет-издательства - один из основных источников информации в наше время. 47% читателей вместе с развитием информационных технологий начали обращаться к онлайн журналистике. Это удобный и доступный большому объему населения способ получения свежих новостей и интересных статей в любой точке мира и в любое время. Именно поэтому грамотная интернет модерация так важна как и для самих интернет-издательств, так и для их руководителей.

Гильдия Издателей Периодической Печати систематически сталкивается с проблемами во время модерации интернет-издательств, которые к ней относятся. Основная проблема, мешающая комфортной модерации сайта - отсутствие централизованного администрирования всех новостных порталов, размещенных на сайте. Также было бы неплохо иметь конструктор, который бы помог в решении ряда структурных вопросов, например - верстка новостной страницы.

Актуальность проекта объясняется потребностью гильдии издателей периодической печати в решении проблем, описанных выше. Портал сайтов региональных СМИ сети союза ГИПП создан именно для решения этой проблемы. С точки зрения ГИПП, данный сервис позволит систематизировать и сгруппировать множество схожих структур, обеспечив при этом централизованное, удобное и понятное управление данной системой. Помимо этого, конструктор позволит свести к минимуму множество структурных и дизайнерских различий, которыми на данный момент обладают все порталы, входящие в ГИПП, при этом не лишая уникальности каждый из Интернет-ресурсов.

Сайт разработан для упрощения деятельности Гильдии Издателей Периодической Печати, что осуществляется путем предоставления централизованного администрирования всех новостных порталов, размещенных на сайте.

После регистрации на сервисе и ее подтверждения пользователь может создать свой новостной портал, используя предоставляемый инструментарий. После этого данный сайт появляется в личном кабинете пользователя и администрации ГИПП. Используя инструментарий портала, можно также редактировать или проводить модерацию всех новостных блоков сайта.

## **6.3. Анализ аналогов и технических решений**

Для того чтобы лучше понимать как именно должно выглядеть веб-приложение и какой функционал оно должно выполнять необходимо провести анализ аналогов. Прямых аналогов у разрабатываемого проекта нет. Причина этого - узкая тематика проекта и уникальность необходимого функционала. Веб-приложение представляет собой симбиоз нескольких сайтов, поскольку выполняет роль конструктора и новостного портала одновременно.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Критерии** | **Платформа для быстрого создания новостного сайта** [**ucrawler.app/cms-ru**](https://ucrawler.app/cms-ru) | **Конструктор новостных сайтов** [**umi.ru**](https://umi.ru/) | **CMS Wordpress** | **Разрабатываемый проект** |
| **Возможность добавления фото и видео к новости** | Ограничено, видео разрешением 720p,  фото не более 10. | Только фото | Ограничено настройкой выбранной темы | Без ограничений |
| **Возможность указания издательства при добавлении новости** | Указание автора новости и издательства. | Только указание автора новости | Только указание автора новости | Указание автора новости и издательства. |
| **Наличие социальных кнопок** | Присутствует блок соц кнопок | Присутствует блок соц кнопок | Присутствует блок соц кнопок | Присутствует блок соц кнопок |
| **Возможность написание черновика будущей статьи** | Отсуствует | Отсуствует | Реализовано за счет статуса новости | Присутствует в виде отдельного раздела |
| **Наличие различных ролей на сайте** | На сайте присутствует разделение ролей | На сайте присутствует разделение ролей | На сайте присутствует разделение ролей | На сайте присутствует разделение ролей |
| **Интерфейс сайта** | Главные новости, последние новости, разделение по рубрикам | Новости по дате и времени без рубрик | Настройка по рубрикам и категориям важности | Новости разделены по рубрикам |
| **Стоимость использования** | 7000 рублей | Условно- бесплатно | В зависимости от выбранной темы | Бесплатно |

Вывод

Очевидными плюсами разрабатываемого проекта является удобство использования редактором, элементарный дизайн позволяющий строго сфокусироваться на новостях интересующих рубрик, а также нулевая стоимость использования. Таким образом разрабатываемый проект сочетает в себе не только уникальность и бесплатность использования, а призван закрывать очевидные функциональные минусы использования зачастую дорогостоящих аналогов.

## **6.4. Формулирование целей и задач по проекту**

Целью проекта является разработка для заказчика уникального веб-приложения для отображения новостей, а также включающий в себя инструментарий с помощью которого можно верстать новости, сохранять их в черновики и менять цветовую палитру сайта. Это поможет не только упростить порядок администрирования всех новостных сайтов ГИПП, но и приведет их к общему и гармоничному виду.

Конечным результатом станет программный продукт в виде веб-приложения для ГИПП, призванного помочь заказчику с решением задач администрирования и содержащий в себе конструктор для интернет-издательств.

Для достижения указанных выше целей были сформулированы следующие задачи:

1. Создание целевой страницы проекта с рабочей командой и отчетах о встречах
2. Разработка единых навигационных элементов для связи страниц портала.
3. Разработка архитектуры портала
4. Разработка базы данных с учетом особенностей предметной области и требований к системе
5. Разработка дизайн-макетов портала
6. Разработка страниц конструктора
7. Загрузка портала на рабочий сервер
8. Планирование последующего содержания проекта
9. Разработка сопутствующей документации
10. Разработка инструкции для пользователя

## **6.5.** [**Исследование задачи с точки зрения использования приложения, CJM, описание целевой аудитории**](https://docs.google.com/document/d/1t7BtNPZHZ4ihsv6niyLb82sNDypffapY/edit#heading=h.gjdgxs)

### **6.5.1. Классы и характеристики пользователей**

Можно выделить несколько основных пользователей которые могут пользоваться веб-приложением на основании требований заказчика и поставленных целей для проекта. Данные пользователи не только полностью удовлетворят требование заказчика в виде администрирования, но и не усложняют работу веб-приложения.

1. Пользователь - самый многочисленный класс пользователя, который будет в веб-приложении. Пользователями являются посетители сайта, зачастую даже не прошедшие аутентификацию. Они приходят на сайт для просмотра новостей и чтения статей по интересующей их тематике. Также пользователь может выкладывать понравившуюся новость в социальную сеть.
2. Администратор - к данной категории пользователей относятся главные редакторы, владельцы изданий и т.д. Администраторы могут использовать весь функционал доступный на сайте. Помимо просмотра новостей по рубрикам администратор может добавлять новости и редактировать их доступным инструментарием, а также редактировать сайт (добавление рубрик, изменение цветовой палитры). Администратор может добавлять новости в черновики, потом публиковать их, просматривать авторов статей.
3. Редактор, автор - данная роль является промежуточной между простым пользователем и администратором. Редактор может также добавлять новости и верстать их с помощью доступного конструктора, но он не может редактировать сайт так, как это может делать администратор. Редактор может редактировать уже опубликованную новость, если там, например, была найдена ошибка, удалять ее или же сохранять в черновики(но только собственные).

На основании данных ролей можно составить следующую диаграмму ролей для данного веб-приложения:

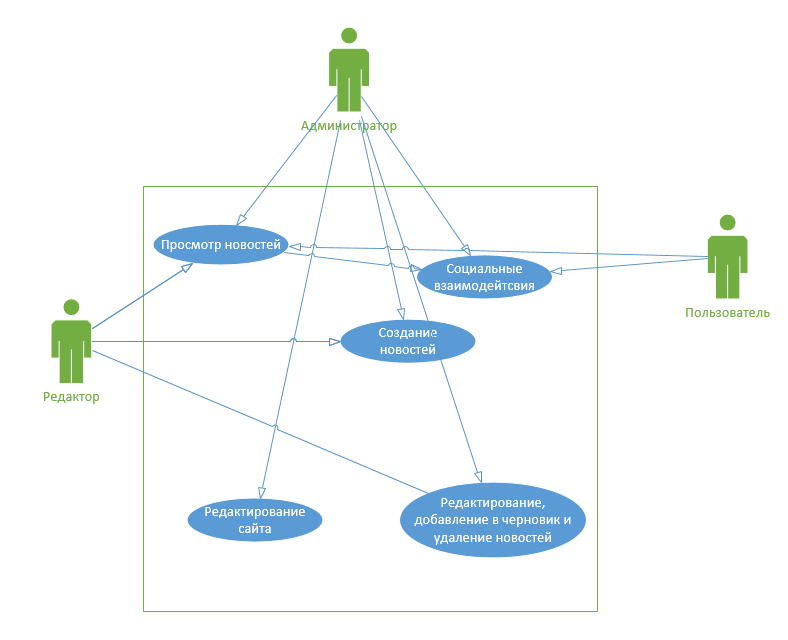


Диаграмма ролей веб-приложения

### **6.5.2. Целевая аудитория**

Среди всей целевой аудитории мы можем выделить следующие необходимые нам сегменты:

1. **Члены гильдии**

**География проживания:** маленький город России

**Возраст:** 25-55 лет

**Статус:** модератор/редактор

**С какой целью заходит на сайт?** Для создания / редактирования новостного портала

**Ожидания от сайта:** простой и доступный интерфейс, интуитивно понятный и удобный функционал конструктора, содержащий достаточный, но не избыточный инструментарий, единообразие интерфейса.

1. **Пользователи/посетители**

**География проживания:** не имеет значения

**Возраст:** 20-70 лет

**Статус:** читатель

**С какой целью заходит на сайт?** Узнать о свежих новостях и событиях

**Ожидания от сайта:** простой и доступный интерфейс,удобная навигация по сайту, регулярное обновление новостной ленты (актуальность новостей), возможность социальных взаимодействий.

Данные сегменты целевой аудитории хоть и являются абсолютно разными, они определяют необходимый для сайта функционал.

## **6.6. Бизнес процессы**

Для упрощения подхода к разработке страницы и понятия архитектуры были создана DFD диаграмма потоков данных, отображающая то, как происходят движения деятельности и данных внутри веб-приложения.

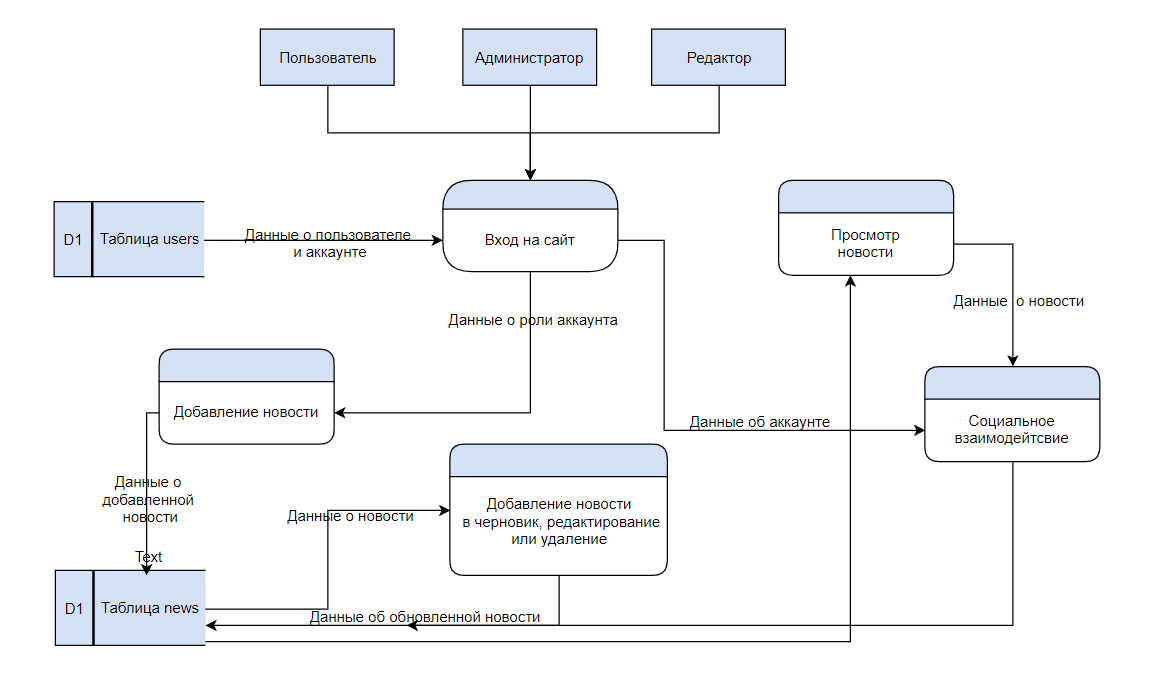


Диаграмма бизнес процессов веб-приложения

# 

# 

# 

# 

# 

# 

# 

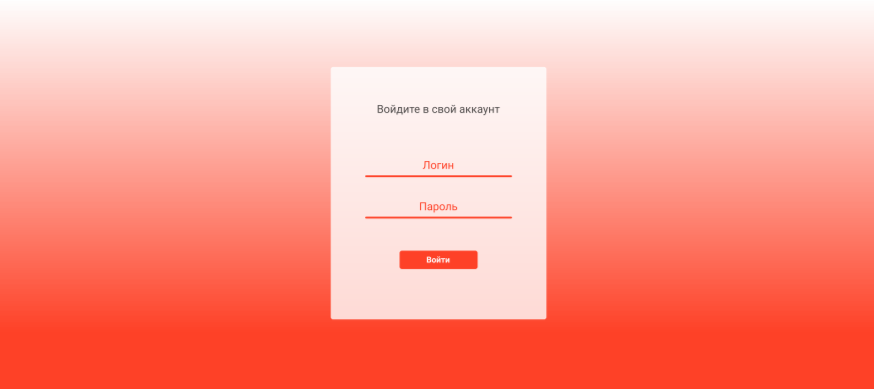
# **7. Реализация проекта**

## **7.1. Проектирование разделов, структура сайта.**

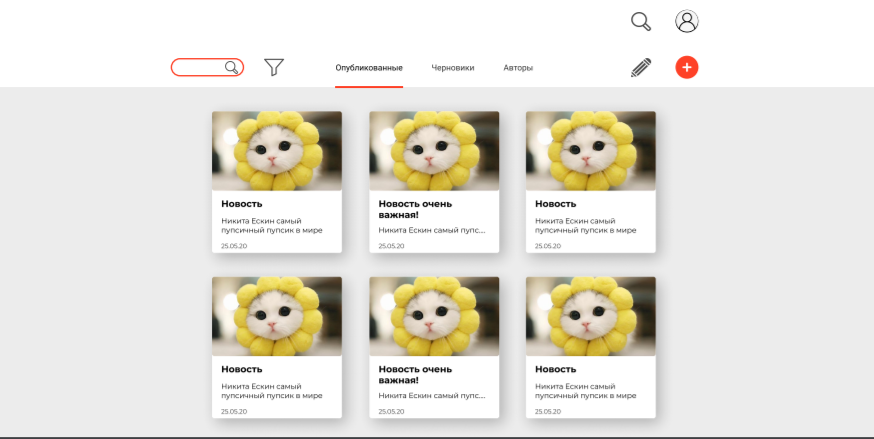
При проектировании макетов сайта был сделан упор на простой и понятный интерфейс для любого посетителя сайта. Было решение не делать конкретную цветовую гамму, так как на сайте есть редактор с помощью которого любое издание может выбрать наиболее комфортный для них цвет. В итоге, мы получили следующую структуру сайта:

* Главная - основная стартовая страница, которую видят все посетители сайта. На ней отображаются последние новости и доступные рубрики, по которым можно переходить.
* Страница входа - здесь пользователь может войти под своим логином и паролем.
* Личный кабинет - здесь пользователь может просмотреть уже созданные статьи, статьи находящиеся в черновиках, авторов. Так же пользователь может редактировать статьи, удалять их, помещать их в черновики, убирать их из черновиков. Присутствуют кнопки добавить статью и редактировать сайт.
* Страница добавлении новости - пользователь может написать здесь статью, отредактировать текст, предварительно посмотреть новость, а затем опубликовать ее или же отправить в черновики
* Страница редактирования сайта - здесь пользователь может изменить название издания, добавить или удалить рубрики, выбрать основные цвета сайта, сохранить выбранные изменения.

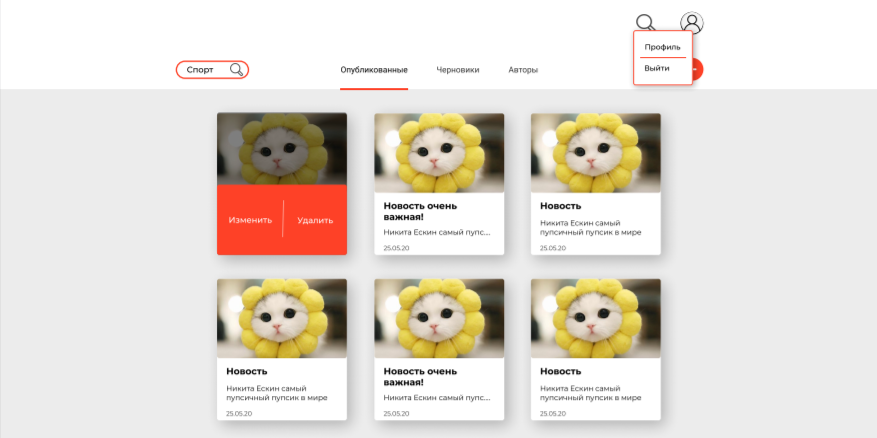
Изначальные макеты страниц сайта:



Страница авторизации



Главная страница



Выбор тематики, кнопка входа в личный кабинет и выхода

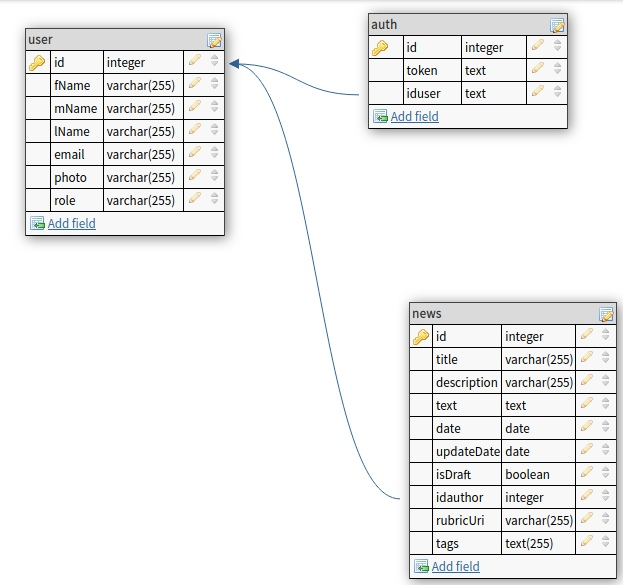
## **7.2. Проектирование структуры базы данных**

### **7.2.1 Основная база данных**

База данных разбита на 3 таблицы, каждая из которых соответствует своему функционалу на сайте:

* Таблица auth: хранит данные об авторизации пользователей
* Таблица user: хранит данные о пользователе
* Таблица news: хранит данные о статьях и новостях с сайта

Структура базы данных:



### **7.2.2 Вспомогательная база данных для хранения файлов**

Представляет из себя одну таблицу для учета сохраненных на диске файлов

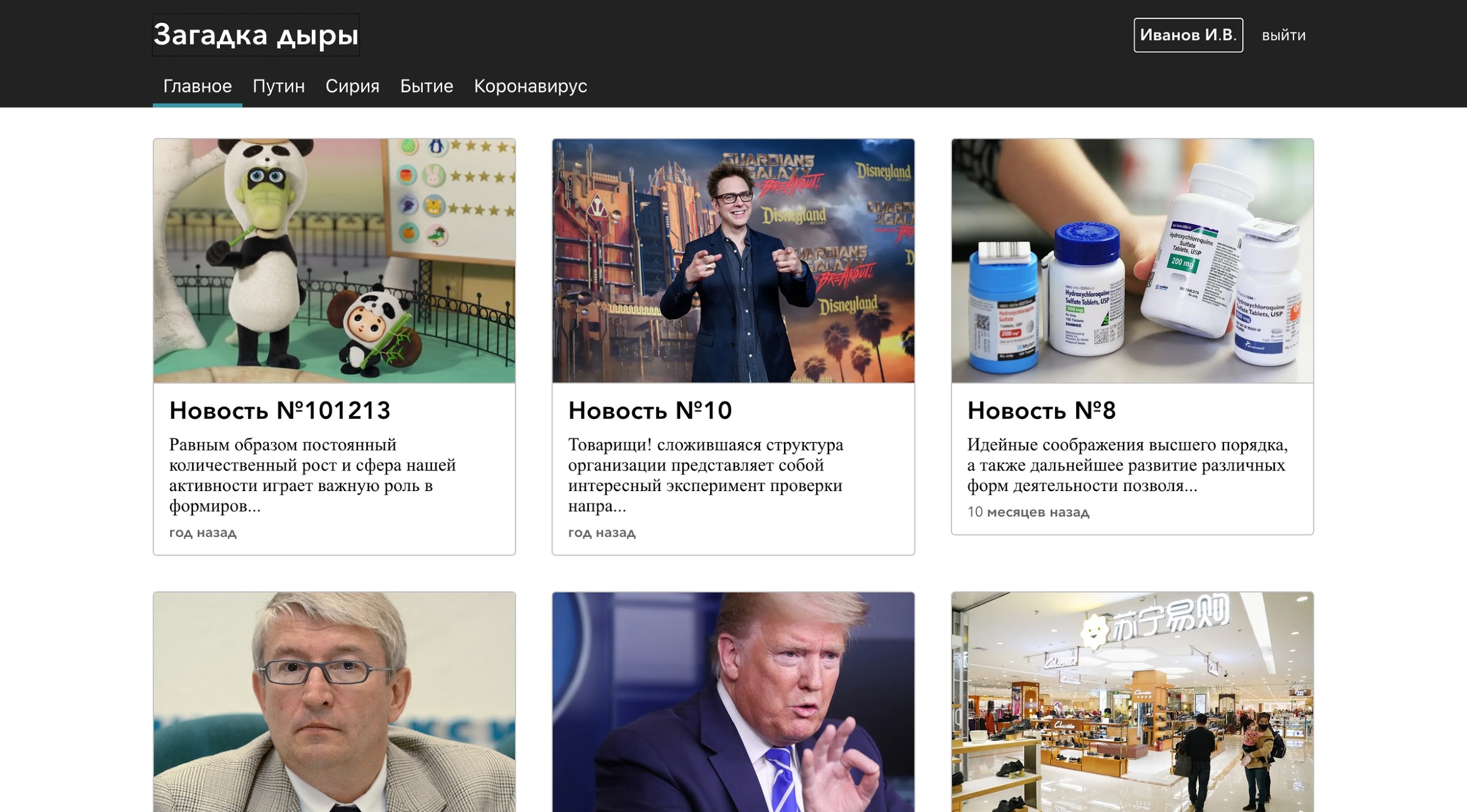


## **7.3. Конкретная реализация задачи на серверной стороне**

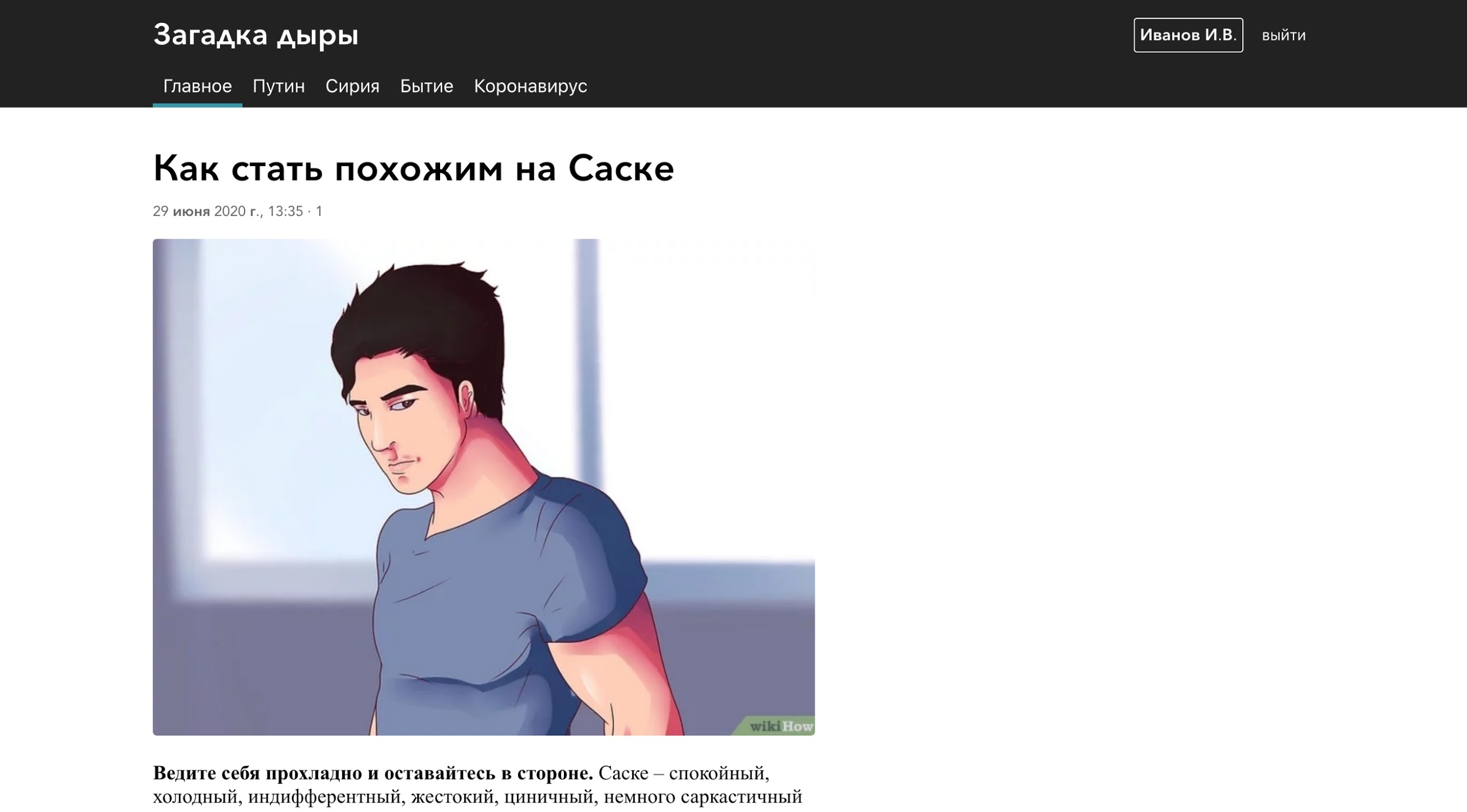
Для реализации основного функционала было написано веб-приложение на php, которое использует представленную выше базу данных (7.2.1).

Для реализации вспомогательного функционала реализован сервис хранения файлов на NodeJS , которое использует представленную выше базу данных (7.2.2).

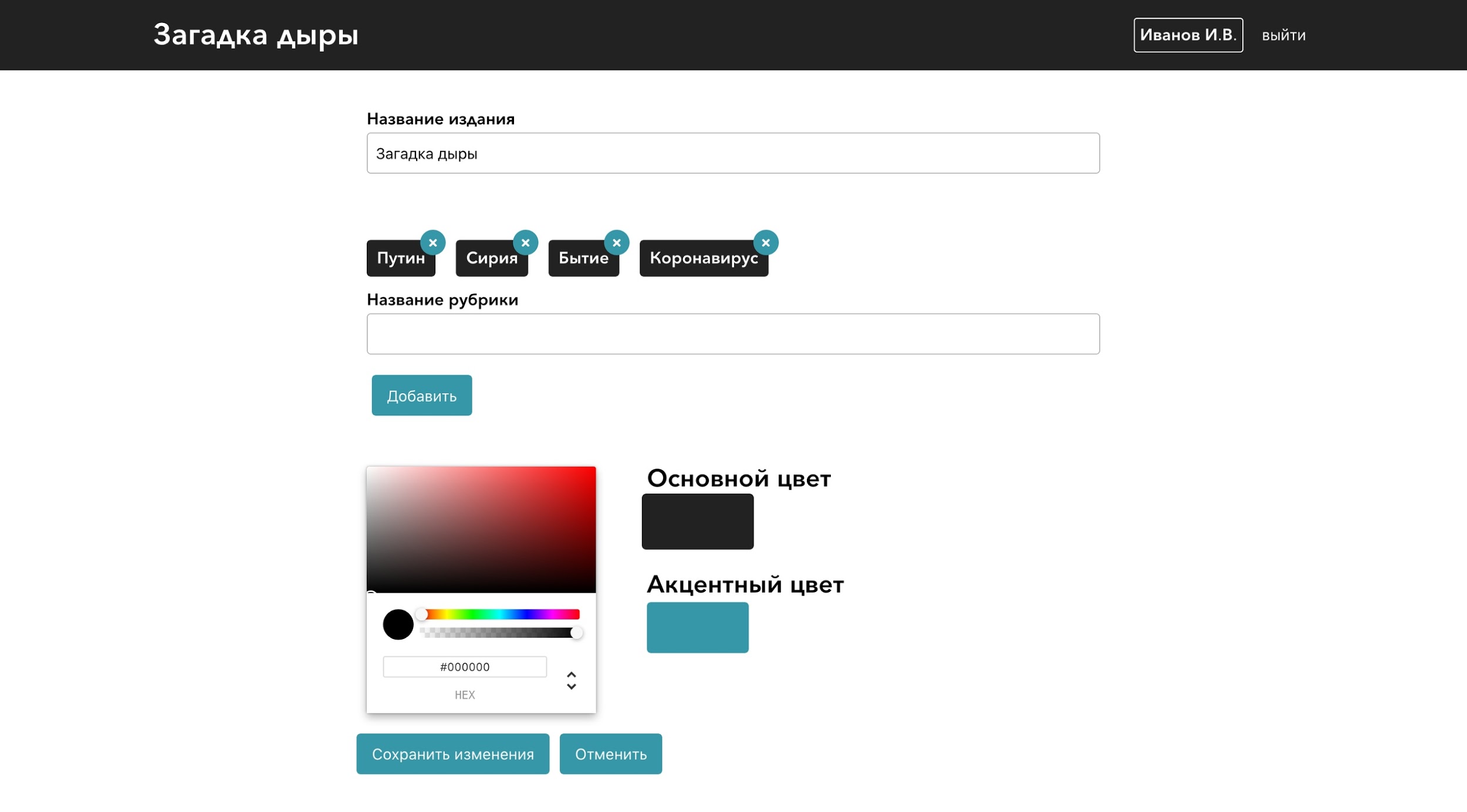
## **7.4. Конкретная реализация задачи на клиентской стороне**

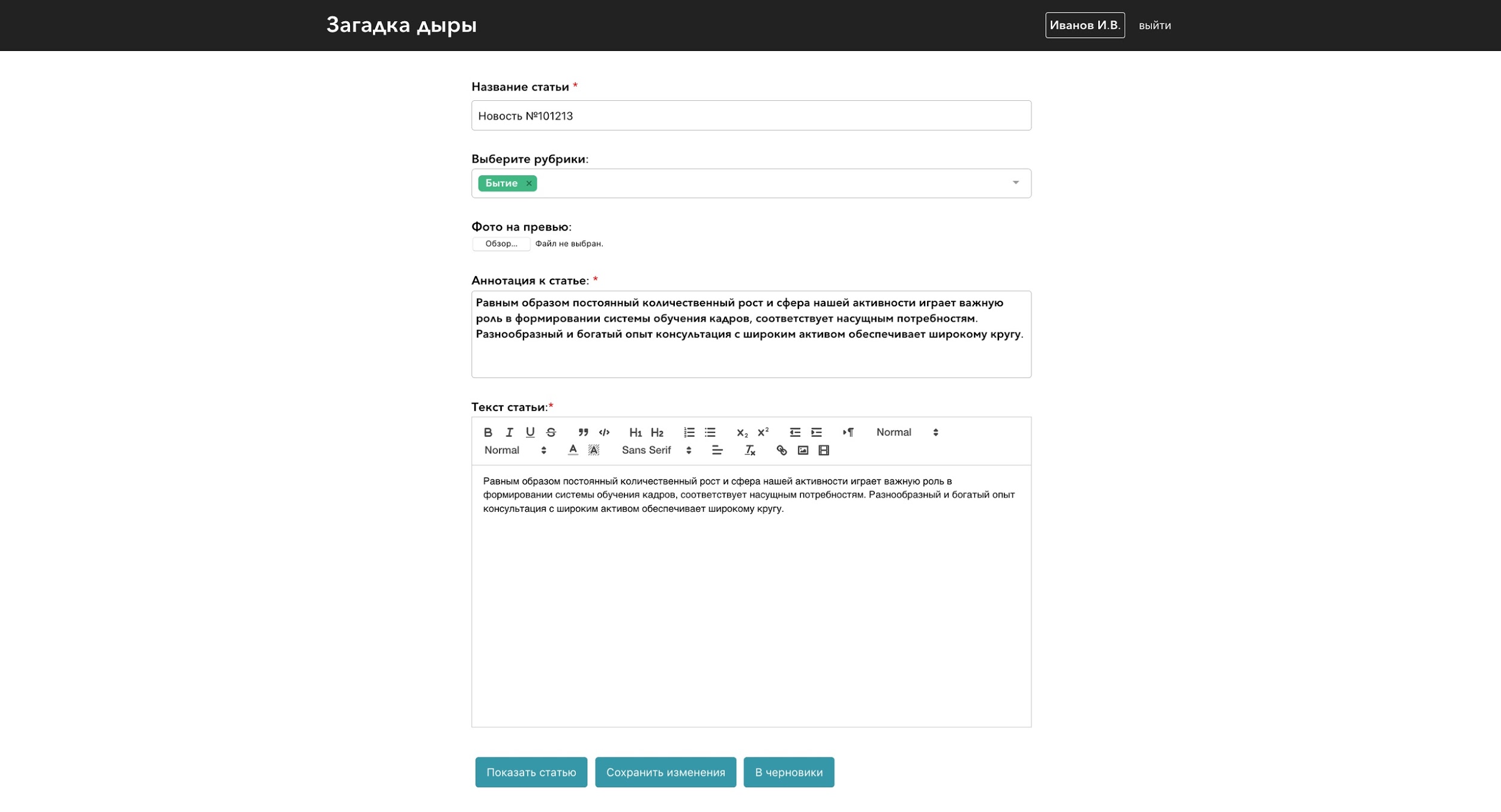
Для комфортной и быстрой реализации задачи на клиентской стороне веб-приложения был выбран VueJS. Ниже вы можете видеть уже сделанные страницы веб-приложения

Главная страница

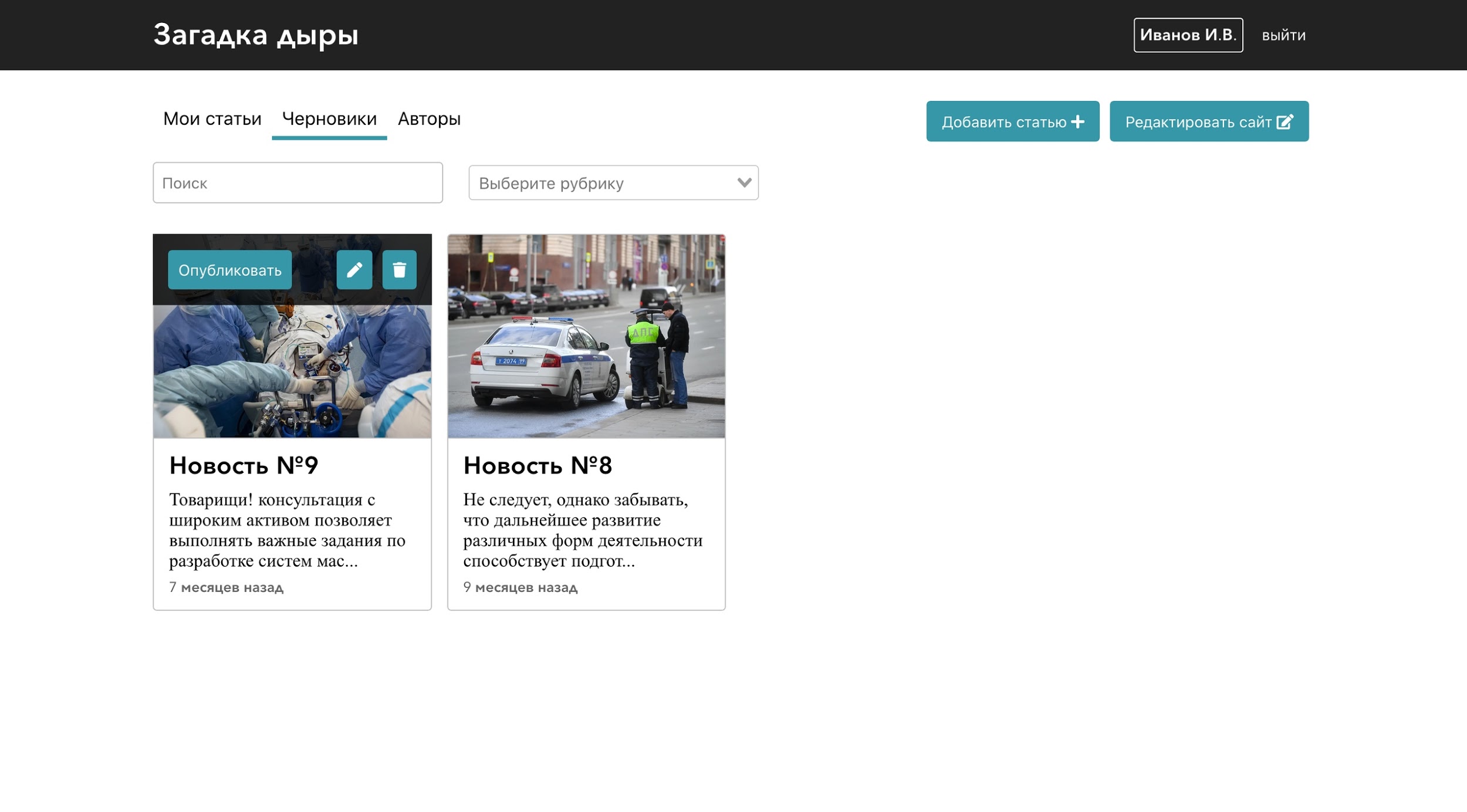


Страница новости

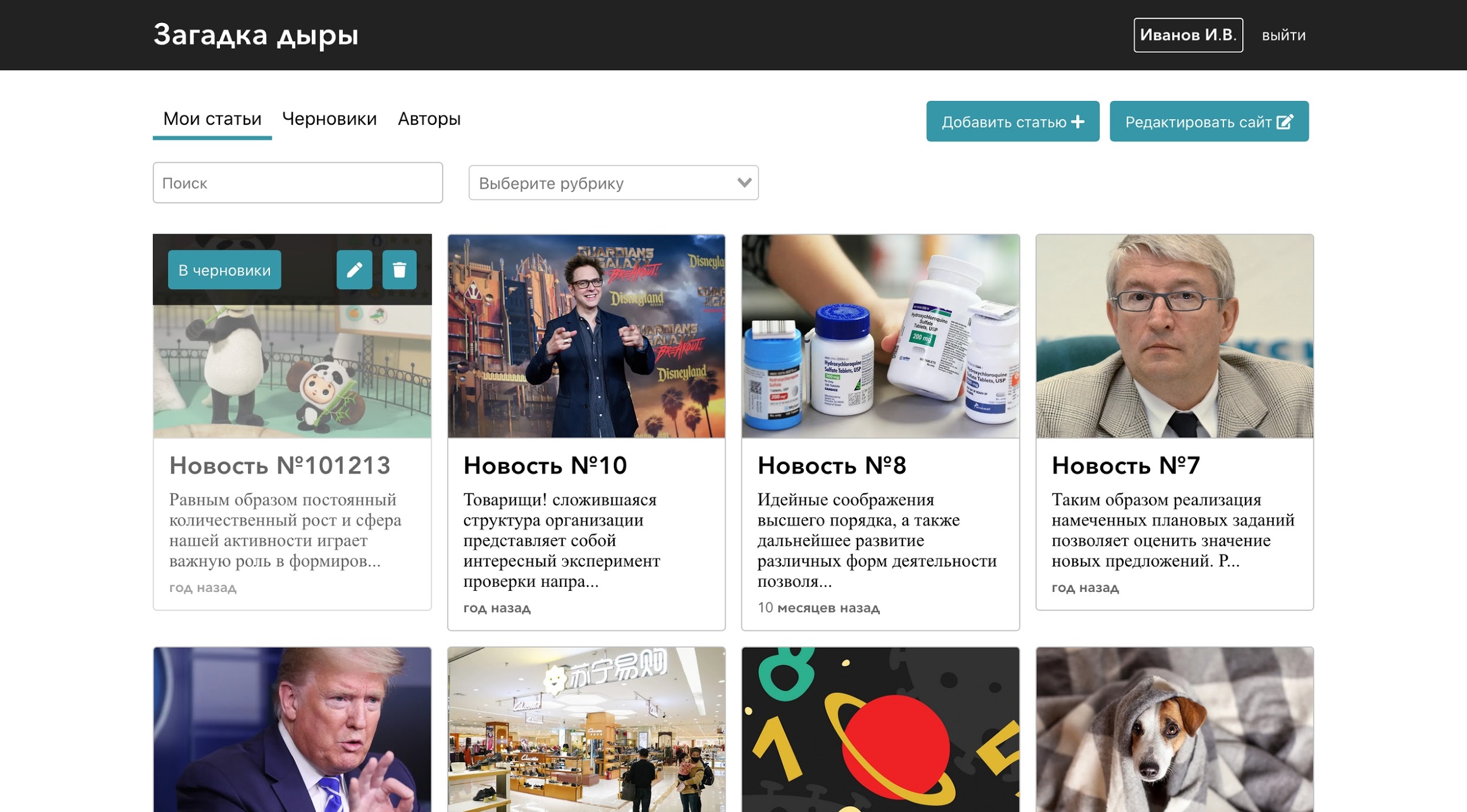
Страница редактирования сайта



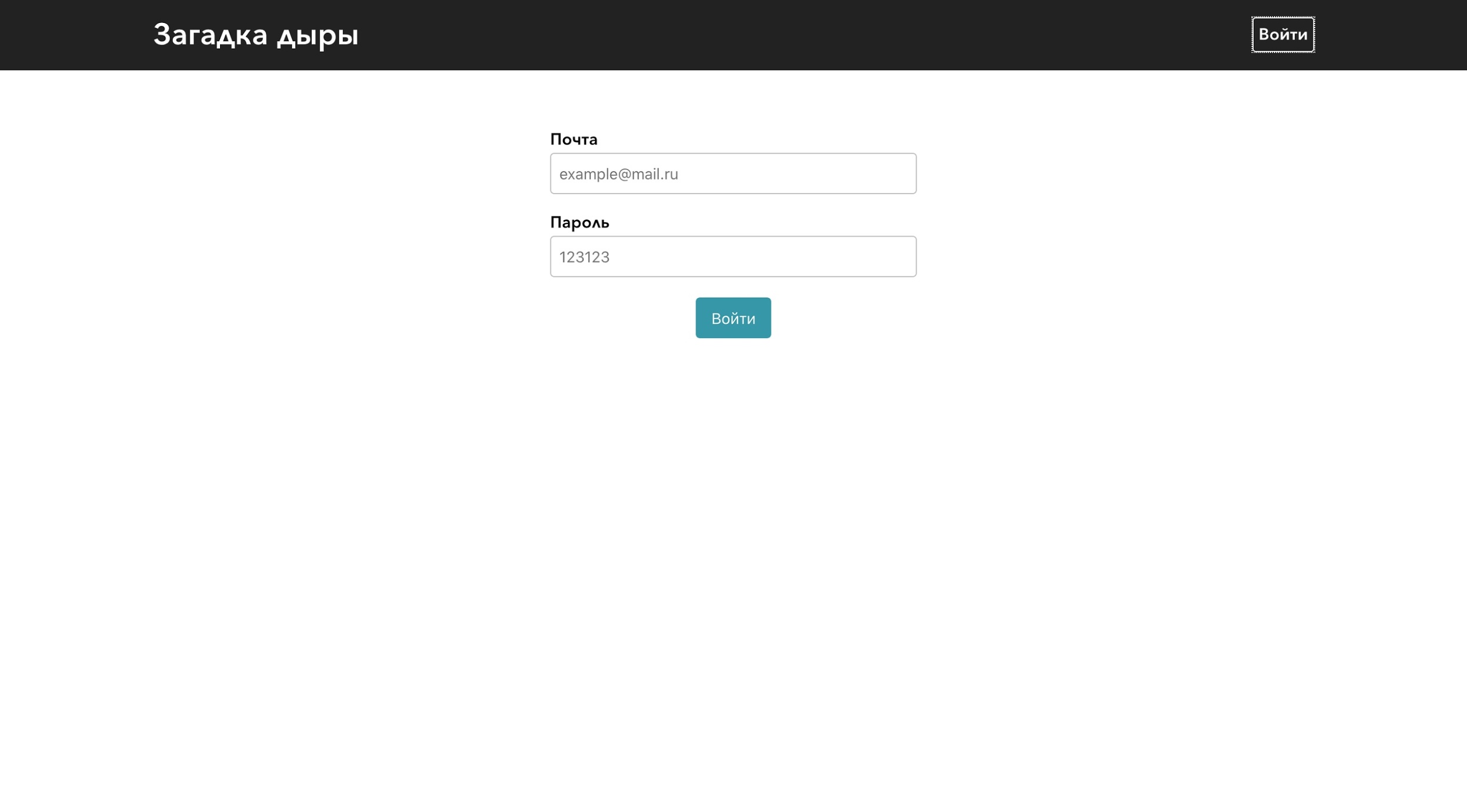
Страница создания новости



Управление статьями (Черновики)



Управление разделами (Опубликованные “Мои статьи”)



Авторизация

## **7.5. Аспекты информационной безопасности**

Как известно, один из самых главных пунктов работы веб-приложения является информационная безопасность, а именно безопасность данных пользователей сайта. Атаки на данные сайта пройдут успешно, если в самом коде веб-приложения не была настроена правильная работа с данными пользователей. При разработке был использован Node.js и большинство проблем безопасности решаются такими способами:

* Настройка и использование определенных http заголовков, которые легко подключаются с помощью модуля Helmet;
* Правильное управление сессией;
* Поиск уязвимостей и ошибок и их исправление;
* Использование технологии TLS, которая позволяет шифровать данные до того как произошла передача данных, обеспечивая их безопасность;
* Использование SQL инъекции - Используя сущности queryset/model для генерации SQL запросов, получается корректно сформированный и экранированный запрос для выбранной базы данных;

## **7.6. Тестирование приложения**

На протяжении всего проекта использовалась ручное тестирование веб-приложения. Тестировщик должен работать с проектом еще с этапа дизайна макетов, проверяя работу по требованию заказчиков.

На этапе проектирования проекта тестировщик проверяет как будет происходить навигация по сайту, валидироваться те или иные поля, как веб-приложение и его функции будут взаимодействовать с пользователями.

На этапе дизайна тестировщик уточняет работу всех элементов и то как пользователь будет взаимодействовать с ними.

Основной этап тестирования уже почти готового веб-приложения - смоук-тест. Здесь тестировщик проверяет готовы ли вообще части веб-приложения к работе, проверяет работу основных функций, правильное отображение элементов и так далее.

Для таких тестов мы использовали своеобразные чек-таблицы, в которых тестировщик писал следующий отчет

* Адрес проверяемой страницы
* Название проверки (отображение элемента, работа функции и так далее)
* Условия для успешной проверки теста
* Итог проверки

Затем таблица передавалась программистам проекта, которые распределяли между собой пункты, за которые были ответственны. Если была найдена ошибка, программист исправлял ее.

Для тестирования основной серверной части приложения использовался Postman. На вход подавался контрольный набор данных, зная какой ответ должен прийти проверялась правильность работы функций.

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# **8. Сопровождение проекта**

## **8.1. Администрирование проекта на сервере**

## Требования к клиентской части, для стабильной работы веб-приложения:

* Наличие браузера (Исключение IE < 9 версии)
* Включенный javascript
* Нет запрета на использование технологии cookie

Требования к серверной части:

* Наличие установленного на сервере Nginx, PHP 7, MySQL;
* Рекомендованная ОС - Ubuntu 18;
* Необходимое ОЗУ - 1Гб;
* Рекомендованное ОЗУ - 2Гб;
* Необходимое и достаточное кол-во ядер процессора - 1;

Вспомогательный сервис серверной части:

* Наличие установленного на сервере Nginx, NodeJS, PostgreSQL, NPM. Свободный порт 5005. При необходимости порт можно переназначить на другой в файле .env ;
* Рекомендованная ОС - Ubuntu 18;
* Необходимое ОЗУ - 1Гб;
* Рекомендованное ОЗУ - 2Гб;
* Необходимое и достаточное кол-во ядер процессора - 1;

Администраторы веб-приложения

Так как данный проект разрабатывается в качестве инструментария для создания полноценного новостного портала, то в качестве администратора сервиса будут выступать пользователи с правами “администратор”. Это позволит создать условия для всей необходимой модерации на сайте.

## **8.2. Инструкция пользователя**

|  |  |  |
| --- | --- | --- |
| № | Название | Действия пользователя |
| 1 | Предоставление информации в виде отображаемого в браузере HTML-документа | Необходимо перейти на сайт по ссылке в любом из доступных браузеров. |
| 2 | Вход на сайт с помощью логина и пароля | Необходимо нажать на кнопку “Войти” и ввести свои данные в новой вкладке |
| 3 | Переход на предпочитаемую рубрику сайта | На главной странице сайта находится строка с рубриками, можно выбрать любую и пользователь перейдет на выбранную рубрику |
| 4 | Переход на страницу личного кабинета | Пользователю необходимо нажать на кнопку со своим именем в верхнем правом углу сайта. |
| 5 | Переход на страницу черновиков | На странице личного кабинета пользователю надо выбрать пункт “Черновики” |
| 6 | Переход на страницу авторов | На странице личного кабинета пользователю надо выбрать пункт “Авторы” |
| 7 | Добавление статьи | На странице личного кабинета пользователю надо нажать на кнопку “Добавить статью” и он окажется в конструкторе по добавлению статьи |
| 8 | Опубликование новой статьи | На странице добавления новой статьи после заполнения необходимых данных пользователю надо нажать на кнопку “Опубликовать” |
| 9 | Просмотр новой статьи до публикации | На странице добавления новой статьи после заполнения необходимых данных пользователю надо нажать на кнопку “Просмотр статьи” |
| 10 | Добавление новой статьи в черновики до публикации | На странице добавления новой статьи после заполнения необходимой данных пользователю надо нажать на кнопку “В черновики” |
| 11 | Добавление уже опубликованной статьи в черновики | В личном кабинете в разделе “Мои статьи” необходимо навести на необходимую статью и нажать на кнопку “В черновики” |
| 12 | Редактирование уже опубликованной статьи | В личном кабинете в разделе “Мои статьи” необходимо навести на необходимую статью и нажать на кнопку, на которой изображена пиктограмма карандаша. |
| 13 | Удаление уже опубликованной статьи | В личном кабинете в разделе “Мои статьи” необходимо навести на необходимую статью и нажать на кнопку, на которой изображена пиктограмма мусорного бака. |
| 13 | Переход на подробную новость | В личном кабинете или на главной странице пользователю надо нажать на необходимую ему новость. |
| 14 | Переход на главную страницу | Из любой страницы сайта пользователю надо нажать на название издания сверху слева |
| 15 | Переход на страницу редактирования сайта | На странице личного кабинета пользователю надо нажать на кнопку “Редактировать сайт” и он окажется в конструкторе по изменению сайта |
| 16 | Изменение названия издания | В редакторе сайта заполнить поле “Название сайта” и нажать кнопку “Сохранить” |
| 17 | Добавление новой рубрики | В редакторе сайта заполнить поле “Название рубрики”, нажать кнопку “Добавить”, а затем кнопку “Сохранить” |
| 18 | Удаление рубрики | В редакторе сайта нажать на крестик возле той рубрики, которую необходимо удалить, а затем кнопку “Сохранить” |
| 19 | Изменение цвета сайта | В редакторе сайта есть выбор цветов с помощью палитры, также можно задать цветовой код. Пользователь может выбрать основной и акцентный цвет. После выбора надо нажать кнопку “Выход” |
| 20 | Выход из аккаунта | Пользователю необходимо нажать на кнопку “Выйти” в правом верхнем углу. |