

Рис. 3.1. Процесс обработки запросов данных на веб-странице

## 1.1. КОНЦЕПТУАЛЬНОЕ ПРОЕКТИРОВАНИЕ

Концептуальное проектирование используется на этапе предварительного описания будущей системы и позволяет сформировать общее представление о ее логике, составе и взаимодействии между компонентами. Этот этап играет важную роль в формировании основы, на которую опирается вся последующая разработка. В процессе проектирования выделяются ключевые блоки, определяются связи между ними, а также описываются типы обрабатываемых данных и предполагаемые функции.

В рамках данной выпускной квалификационной работы концептуальное проектирование применялось для определения структуры информационной системы, предназначенной для внутреннего использования в IT-отделе организации. На этом этапе были описаны основные функциональные элементы, включая модуль аутентификации, систему управления материалами, механизм ролевого доступа и модуль полнотекстового поиска. Для каждого из компонентов определялись входные и выходные данные, а также логика взаимодействия с другими частями системы.

Особенностью концептуального проектирования является то, что на этом этапе не рассматриваются конкретные реализации. Важно определить, какие компоненты должны существовать в системе, и каким образом они взаимодействуют, не углубляясь в детали кода или технологии. В проекте были разработаны схемы, отражающие логику обработки данных, движение

пользовательских запросов и реакции со стороны серверной части. Это позволило сформировать целостное понимание архитектуры будущей системы и упростило последующую детализацию в рамках логического и физического проектирования.

Ключевым инструментом на этапе концептуального проектирования является диаграмма «use case» (варианты использования). Она представляет собой описание взаимодействия между пользователем и системой, отражающее процесс, который осуществляется в рамках использования.

В ходе анализа разрабатываемой системы было выделено два действующих лица – IT-специалист и администратор.

IT-специалисту доступны следующие функции:

1. поиск материалов в системе;
2. добавление, удаление собственных материалов из системы (включает предварительную выдачу роли в системе);
3. добавление, удаление собственных комментариев к материалам системе
4. изменение собственных персональных данных профиля

Администратору доступны следующие функции:

1. регистрация пользователей в информационной системе;
2. поиск материалов в системе;
3. добавление, удаление собственных материалов из системы;
4. добавление, удаление собственных комментариев к материалам системе
5. изменение собственных персональных данных профиля
6. модерация материалов и комментариев в системе;
7. выдача роли в системе;
8. блокировка пользователя в системе.

Далее изображена общая диаграмма вариантов использования (рис. 3.2):

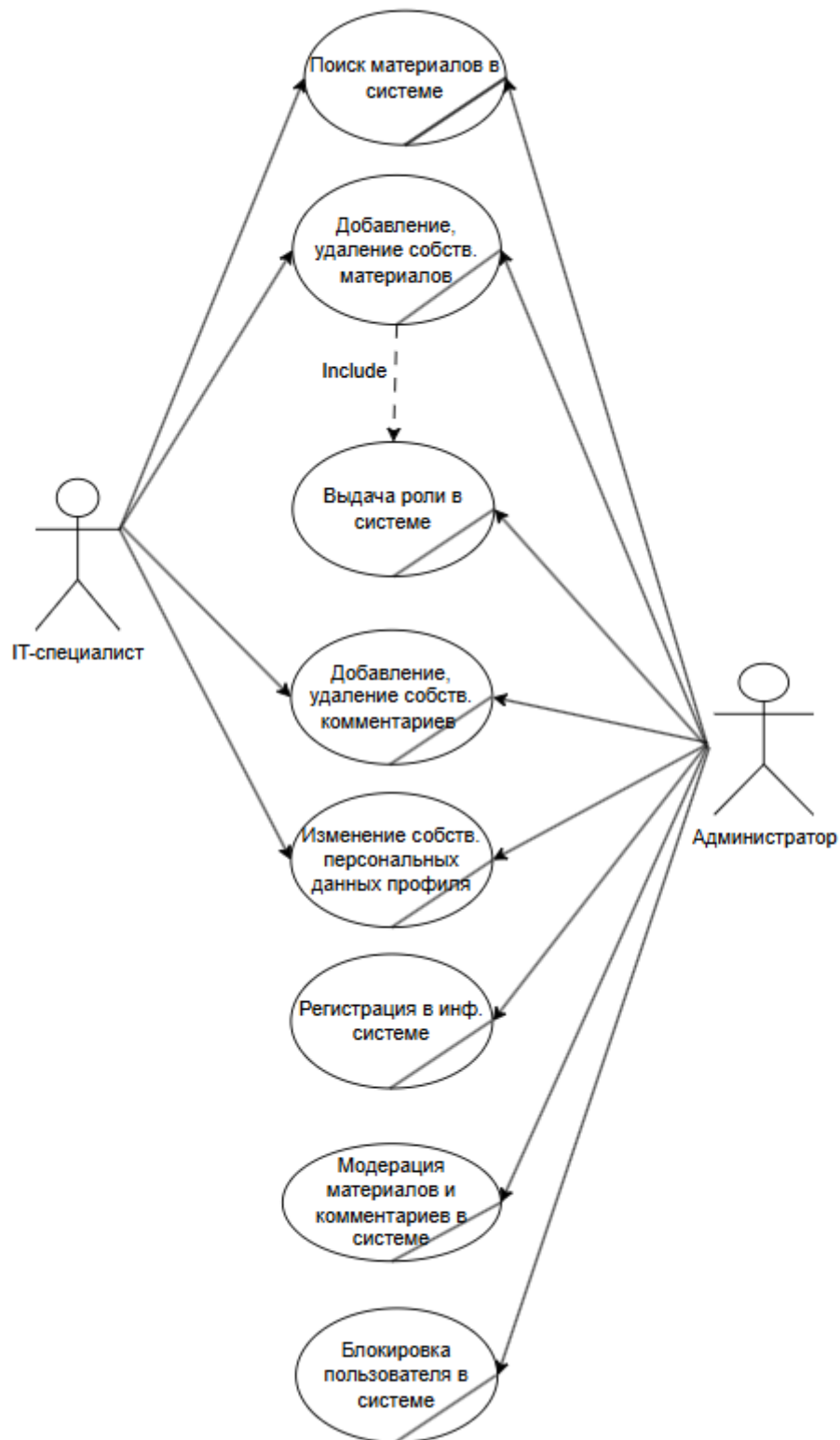


Рис. 3.2. Диаграмма вариантов использования

Для решения задач, связанных с разработкой информационной системы поддержки деятельности IT-отдела, необходимо хранение данных о следующих объектах:

1. Материалы базы знаний – статьи, инструкции, руководства, которые

добавляются, редактируются и хранятся в системе.

2. Пользователи – сотрудники IT-отдела, имеющие доступ к системе, включая их роли.

3. История изменений материалов – информация о том, кто и когда вносил изменения в материалы.

4. Категории материалов – различные типы информации, например, инструкции по решению проблем, технические статьи и т. д.

5. Комментарии – оставленные пользователями комментарии под материалами базы знаний.

6. Вложения – прикрепленные к материалам документы и изображения, привязанные к записям базы знаний.

## **1.2. ЛОГИЧЕСКОЕ ПРОЕКТИРОВАНИЕ**

Логическим проектированием базы данных является преобразование требований к данным в структуры данных. Результатом логического проектирования является СУБД-ориентированная структура базы данных и спецификации прикладных программ. Целью логического проектирования служит определение состава и структуры таблицы БД на основе результатов концептуального проектирования и проверка полученной модели с помощью методов нормализации.

В стадии логического проектирования входит:

- формирование отношений на основе логической модели данных;
- проверка отношений с использованием средств нормализации;
- определение ограничений целостности.

Для построения отношений используются диаграммы ER-типа сущность - связь. Основные ее определения:

- Сущность – объект, информация о котором должна храниться в базе данных. Для ее изображения используется прямоугольный блок с именем сущности в виде существительного;

- связь – ассоциация между двумя сущностями, она предполагает наличие общих атрибутов. Для ее изображения используется ромб с именем связи в виде глагола.

Связи бывают типов: 1:1, 1:M (или M:1), M:M. Класс принадлежности может быть обязательным и необязательным.

На предыдущем этапе были выделены объекты, которые необходимо хранить в БД. Эти объекты становятся сущностями при ER-моделировании.

Построим ER-диаграммы всех сущностей и связей между ними.

ER-диаграмма «Пользователь создает материал» (рис. 3.3).

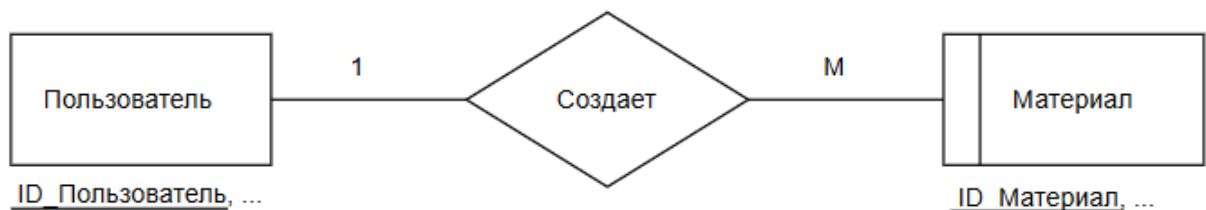


Рис. 3.3. Связь «Создает»

Связь «Создает» имеет тип связи 1:M, так как каждый пользователь может создать несколько материалов, а каждый материал связан только с одним пользователем. Сущность «Пользователь» имеет необязательный класс принадлежности, так как пользователь может не создать ни одного материала. Сущность «Материалы» имеет обязательный класс принадлежности, так как каждый материал должен быть связан с хотя бы одним пользователем.

ER-диаграмма «История изменений связана с материалом» (рис. 3.4).

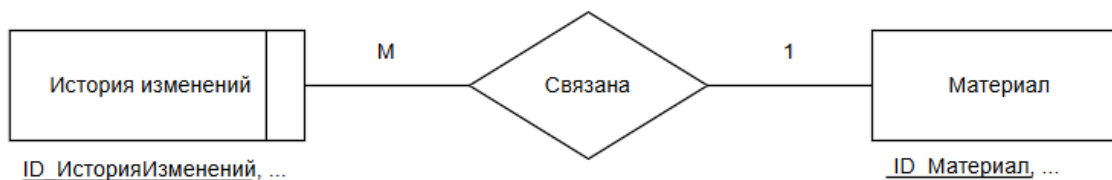


Рис. 3.4. Связь «Связана»

Связь «Связана» имеет тип связи 1:M, так как каждый материал может иметь несколько изменений, а каждое изменение связано с одним конкретным

материалом. Сущность «Материалы» имеет необязательный класс принадлежности, так как каждый материал может не иметь изменений. Сущность «История изменений» имеет обязательный класс принадлежности, так как каждое изменение обязательно должно быть связано с каким-либо материалом.

ER-диаграмма «Пользователь формирует историю изменений» (рис. 3.5).



Рис. 3.5. Связь «Формирует»

Связь «Формирует» имеет тип связи 1:M, так как один пользователь может внести несколько изменений, но каждое изменение формируется только одним пользователем. Сущность «Пользователь» имеет необязательный класс принадлежности, так как пользователь может не вносить изменения. Сущность «ИсторияИзменений» имеет обязательный класс принадлежности, так как каждое изменение должно быть связано с конкретным пользователем.

ER-диаграмма «Категория включает материал» (рис. 3.6).



Рис. 3.6. Связь «Включает»

Связь «Включает» имеет тип связи M:M, так как каждый материал может быть отнесен к одной или нескольким категориям, а каждая категория может включать несколько материалов. Сущность «Материалы» имеет обязательный класс принадлежности, так как каждый материал должен

принадлежать к какой-либо категории. Сущность «Категория» имеет обязательный класс принадлежности, так как категория должна обязательно включать материал.

ER-диаграмма «Пользователь оставляет комментарий» (рис. 3.7).



Рис. 3.7. Связь «Оставляет»

Связь «Оставляет» имеет тип связи 1:M, так как один пользователь может оставить несколько комментариев, но каждый комментарий связан только с одним пользователем. Сущность «Пользователь» имеет необязательный класс принадлежности, так как не каждый пользователь может оставить комментарий. Сущность «Комментарий» имеет обязательный класс принадлежности, так как каждый комментарий обязательно должен быть связан с пользователем.

ER-диаграмма «Комментарий принадлежит материалу» (рис. 3.8).



Рис. 3.8. Связь «Принадлежит»

Связь «Принадлежит» имеет тип связи M:1, так как каждый комментарий связан с одним материалом, но каждый материал может иметь несколько комментариев. Сущность «Комментарий» имеет обязательный класс принадлежности, так как каждый комментарий обязательно должен

быть привязан к материалу. Сущность «Материал» имеет необязательный класс принадлежности, так как материал может не иметь комментариев.

ER-диаграмма «Материал содержит файл» (рис. 3.7).

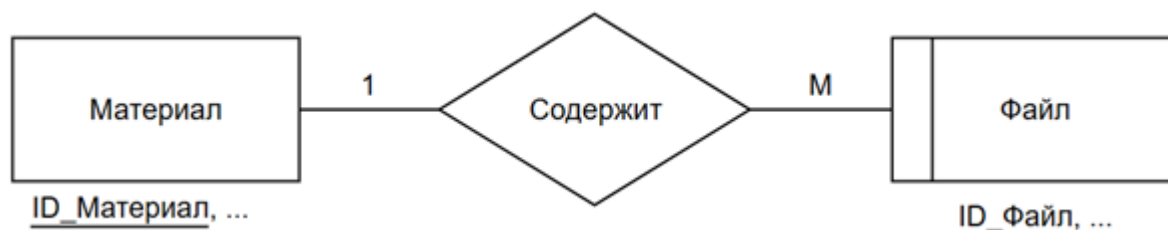


Рис. 3.9. Связь «Содержит»

Связь «Содержит» имеет тип 1:M, так как один материал может содержать несколько вложений, а каждое вложение относится ровно к одному материалу. Сущность «Материал» имеет необязательный класс принадлежности, так как материал может не иметь вложений. Сущность «Файл» имеет обязательный класс принадлежности, так как каждое вложение обязательно должно быть связано с каким-то материалом.

Сформируем набор предварительных отношений с указанием предполагаемого первичного ключа для каждого отношения.

Связь «Создает» (рис. 3.3.) удовлетворяет условиям правила 4, в соответствии с которым получается 2 отношения:

1. Пользователь (Id\_Пользователь, ...).
2. Материал (Id\_Материал, Id\_Пользователь, ...).

Связь «Связана» (рис. 3.4.) удовлетворяет условиям правила 4, в соответствии с которым получается 2 отношения:

1. Материал (Id\_Материал, ...).
2. ИсторияИзменений (Id\_ИсторияИзменений, Id\_Материал, ...).

Связь «Формирует» (рис. 3.5.) удовлетворяет правилу 4, поэтому получаются два отношения:

1. Пользователь (Id\_Пользователь, ...).
2. ИсторияИзменений (Id\_ИсторияИзменений, Id\_Пользователь, ...).



Связь «Включает» (рис. 3.6.) удовлетворяет условиям правила 6, в соответствии с которым получается 3 отношения:

1. Материал (Id\_Материал, ...).
2. Категория (Id\_Категория, ...).
3. Материал\_Категория (Id\_Материал, Id\_Категория, ...).

Связь «Оставляет» (рис. 3.7.) удовлетворяет условиям правила 4, в соответствии с которым получается 2 отношения:

1. Пользователь (Id\_Пользователь, ...).
2. Комментарий (Id\_Комментарий, Id\_Пользователь, ...).

Связь «Принадлежит» (рис. 3.8.) удовлетворяет условиям правила 4, в соответствии с которым получается 2 отношения:

1. Материал (Id\_Материал, ...).
2. Комментарий (Id\_Комментарий, Id\_Материал, ...).

Связь «Содержит» (рис. 3.9.) удовлетворяет условиям правила 4, в соответствии с которым получается 2 отношения:

1. Материал (Id\_Материал, ...).
2. Файл (Id\_Файл, Id\_Материал, ...).

Далее рассмотрено добавление в сформированные отношения не ключевых атрибутов.

- Пользователь (Id\_Пользователь, ФИО, Почта, Пароль, Роль, ДатаРегистрации)
- Материал (Id\_Материал, Заголовок, Содержание, ДатаСоздания, Id\_Пользователь)
- ИсторияИзменений (Id\_ИсторияИзменений, ДатаИзменения, Id\_Материал, Id\_Пользователь)
- Категория (Id\_Категория, НазваниеКатегории)
- Материал\_Категория (Id\_Материал, Id\_Категория)
- Комментарий (Id\_Комментарий, ТекстКомментария, ДатаКомментария, Id\_Пользователь, Id\_Материал)

- Файл (Id\_Файл, Путь, Оригинальное\_имя, Размер, Тип, Дата\_создания, Id\_Материал)

### 1.3. ФИЗИЧЕСКОЕ ПРОЕКТИРОВАНИЕ

Физическое проектирование – это процесс подготовки описания реализации базы данных. На этом этапе рассматриваются основные отношения и организация, предназначенных для обеспечения эффективного доступа к данным, а также все связанные с этим ограничения целостности и средства защиты.

Физическая модель данных представляет собой детализированное воплощение логической модели, адаптированное к особенностям конкретной системы управления базами данных. Модель разрабатывается с учетом конкретных потребностей проекта и может быть объединена с другими моделями при необходимости масштабирования системы. При определении связей между таблицами учитывается количество записей и возможность использования NULL для отдельных атрибутов, что способствует сохранению целостности данных. Физическое проектирование проводится с учетом версии СУБД, конфигурации серверного оборудования и условий хранения, что позволяет оптимизировать производительность системы. Каждый столбец таблиц получает точный тип данных, фиксированную длину и значение по умолчанию, а также определяется набор первичных и внешних ключей, представлений, индексов, а также методов контроля доступа и аутентификации. В результате данного этапа была сформирована физическая схема базы данных, включающая семь таблиц (табл. 3.1-3.6)

Таблица 3.1

Структура таблицы «Пользователь»

Столбец	Тип	Ноль?	Ключ	Значение по умолчанию	Ограничение	Ссылка
ID	Целое	NOT NULL	Первичный			

Столбец	Тип	Ноль?	Ключ	Значение по умолчанию	Ограничение	Ссылка
ФИО	Строка	NOT NULL				
Почта	Строка	NOT NULL			Уникальный	
Пароль	Строка	NOT NULL				
Роль	Строка	NOT NULL		Пользователь		
Дата_регистрации	Дата	NOT NULL		Текущая дата		
Заблокирован	Булевый	NOT NULL				

Таблица 3.2

## Структура таблицы «Материал»

Столбец	Тип	Ноль?	Ключ	Значение по умолчанию	Ограничение	Ссылка
ID	Целое	NOT NULL	Первичный			
Заголовок	Строка	NOT NULL				
Содержание	Строка	NOT NULL				
Дата_Создания	Дата	NOT NULL		Текущая дата		
ID_Пользователь	Целое	NOT NULL	Внешний			Пользователь (ID)
Search_vector	Tsvector	NOT NULL				

Таблица 3.3

## Структура таблицы «История изменений»

Столбец	Тип	Ноль?	Ключ	Значение по умолчанию	Ограничение	Ссылка
ID	Целое	NOT NULL	Первичный			
ID_Пользователь	Целое	NOT NULL	Внешний			Пользователь (ID)

Столбец	Тип	Нуль?	Ключ	Значение по умолчанию	Ограничение	Ссылка
ID_Материал	Целое	NOT NULL	Внешний			Материал (ID)
ДатаИзменения	Дата	NOT NULL		Текущая дата		

Таблица 3.4

## Структура таблицы «Категория»

Столбец	Тип	Нуль?	Ключ	Значение по умолчанию	Ограничение	Ссылка
ID	Целое	NOT NULL	Первичный			
Название	Строка	NOT NULL			Уникальный	

Таблица 3.5

## Структура таблицы «Материал\_Категория»

Столбец	Тип	Нуль?	Ключ	Значение по умолчанию	Ограничение	Ссылка
ID_Материал	Целое	NOT NULL	Первичный			Материал (ID)
ID_Категория	Целое	NOT NULL	Первичный			Категория (ID)

Таблица 3.6

## Структура таблицы «Комментарий»

Столбец	Тип	Нуль?	Ключ	Значение по умолчанию	Ограничение	Ссылка
ID	Целое	NOT NULL	Первичный			
ТекстКомментария	Строка	NOT NULL				
ДатаКомментария	Дата	NOT NULL		Текущая дата		
ID_Материал	Целое	NOT NULL	Внешний			Материал (ID_Материал)
ID_Пользователь	Целое	NOT NULL	Внешний			Пользователь (ID_Пользователь)

Структура таблицы «Файл»

Столбец	Тип	Ноль?	Ключ	Значение по умолчанию	Ограничение	Ссылка
ID	UUID	NOT NULL	Первичный			
Путь	Строка	NOT NULL				
Оригинальное_имя	Строка	NOT NULL				
Размер	Целое	NOT NULL				
Тип	Строка	NOT NULL				
Дата_создания	Дата	NOT NULL		Текущая дата		
ID_Материал	Целое	NOT NULL	Внешний			Материал (ID_Материал)

## 1.4. ОПИСАНИЕ РАБОТЫ С СИСТЕМОЙ

Прежде чем приступить к использованию системы, пользователь должен войти в систему, подтвердив свои учетные данные (рис. 3.10).

**Добро пожаловать!**

E-mail

Пароль

**Войти**

[Забыли пароль?](#)

Нажимая «Войти», вы принимаете пользовательское соглашение и политику конфиденциальности

Рис. 3.10. Окно авторизации

При вводе некорректных данных система выдаст сообщение об ошибке входа (рис. 3.11).

**Добро пожаловать!**

**Войти**

Неверный email или пароль

[Забыли пароль?](#)

Нажимая «Войти», вы принимаете пользовательское соглашение и политику конфиденциальности

Рис. 3.11. Ошибка при входе

После корректного ввода логина и пароля пользователь авторизуется и попадает на стартовый экран. На нем отображается поисковая строка, категории системы и профиль пользователя (рис. 3.12).

IT Support System

Привет, Иван Иванович

Выйти

**Поиск по базе знаний**

Выберите категорию, чтобы просмотреть материалы

Поиск

Сетевые технологии

Тестовая Категория

Информационная безопасность

Операционные системы

Системное администрирование

Работа с ПО и лицензиями

Устранение неполадок

Поддержка пользователей

DevOps и автоматизация

Базы данных

Документация по внутренним системам

Рис. 3.12. Начальная страница

При нажатии на кнопку раскрывается список возможностью добавления материала, категории или пользователя (рис. 3.13).

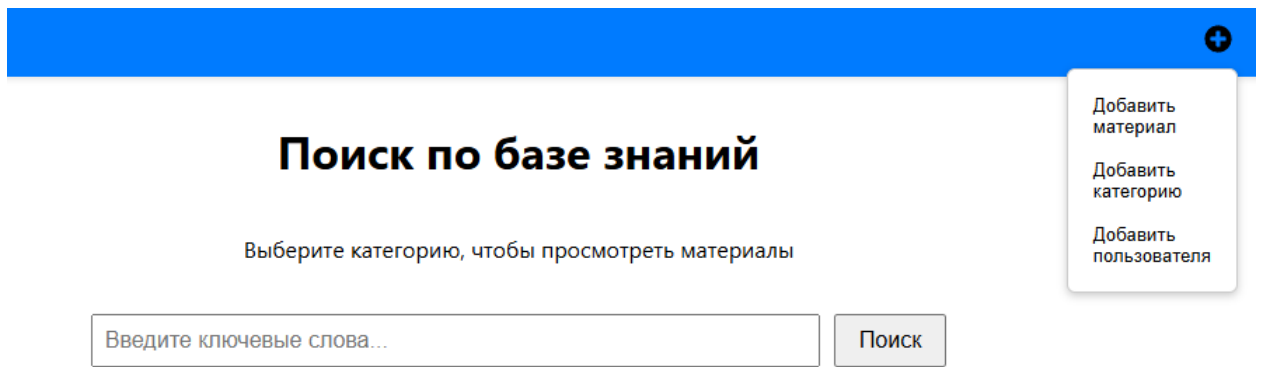


Рис. 3.13. Кнопки добавления материала и категории

При нажатии на кнопку добавления материала открывается страница с полями для создания материала. Пользователь указывает заголовок, текстовую часть и категорию. Категория заполняется с помощью выпадающего списка (рис. 3.14).

### Создание нового материала

#### Заголовок

Установка Docker



Docker — это платформа для быстрой разработки, тестирования и развертывания приложений в изолированных контейнерах.

Установка Docker проста и занимает всего несколько шагов.

#### Инструкция по установке:

- Перейдите на официальный сайт [docker.com](https://docker.com).
- Выберите подходящую версию для вашей операционной системы (Windows, macOS или Linux).
- Скачайте установочный файл.
- Запустите установку и следуйте инструкциям мастера установки.
- После завершения установки перезагрузите компьютер (для Windows и macOS).

#### Файлы

Прикрепить файлы

#### Категории

Сетевые технологии x

x | v

Рис. 3.14. Страница создания материала

Добавление таблицы к материалу происходит при нажатии на кнопку «добавить таблицу». Выдается окно с указанием размеров таблицы (рис. 3.15).

The image shows a dialog box titled "Размер таблицы" (Table Size) overlaid on a blurred background of a document. The dialog has two input fields: "Строки:" (Rows) with the value "6" and "Столбцы:" (Columns) with the value "2". At the bottom right, there are two buttons: "Создать" (Create) and "Отмена" (Cancel).

Рис. 3.15. Окно указания размеров таблицы

Далее заполняется таблица (рис. 3.16).

Требования к системе для установки Docker:

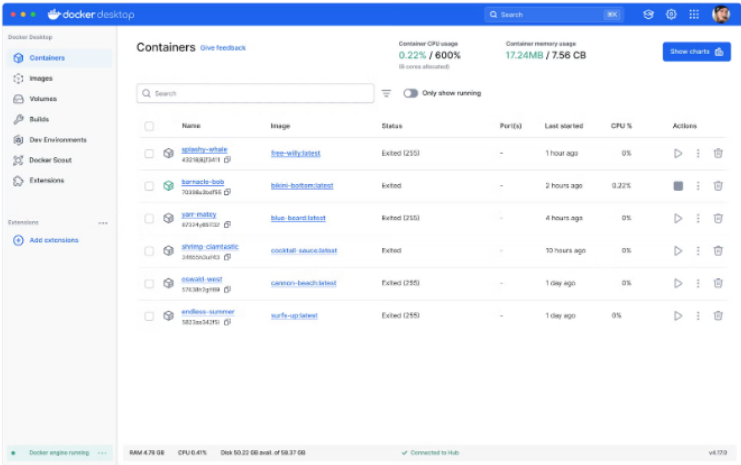
Требование	Минимальные параметры
Операционная система	Windows 10/11 (Pro/Enterprise) или macOS 10.15+
Процессор	Поддержка виртуализации (Intel VT-x или AMD-V)
Оперативная память	От 4 ГБ
Свободное место	<ul style="list-style-type: none"> <li>От 5 ГБ на диске</li> </ul>

Рис. 3.16. Созданная таблица

При нажатии на кнопку «Прикрепить файлы» пользователь имеет возможность прикрепить файлы или изображения (рис. 3.17).



Окно docker desktop:



Файлы

Прикрепить файлы

Docker.pdf (691.6 KB)

Рис. 3.17. Прикрепленные файлы

При нажатии на кнопку «Создать» материал публикуется в системе (рис. 3.18-3.19).

Установка Docker

Автор: Иванов Иван Иванович    Создан: 20.06.2025    Обновлено: 20.06.2025  
Категория: Сетевые технологии

Docker — это платформа для быстрой разработки, тестирования и развертывания приложений в изолированных контейнерах.

Установка Docker проста и занимает всего несколько шагов.

Инструкция по установке:

- Перейдите на официальный сайт [docker.com](https://docker.com).
- Выберите подходящую версию для вашей операционной системы (Windows, macOS или Linux).
- Скачайте установочный файл.
- Запустите установку и следуйте инструкциям мастера установки.
- После завершения установки перезагрузите компьютер (для Windows и macOS).
- Требования к системе для установки Docker:

Требование	Минимальные параметры
Операционная система	Windows 10/11 (Pro/Enterprise) или macOS 10.15+
Процессор	Поддержка виртуализации (Intel VT-x или AMD-V)
Оперативная память	От 4 ГБ
Свободное место	От 5 ГБ на диске

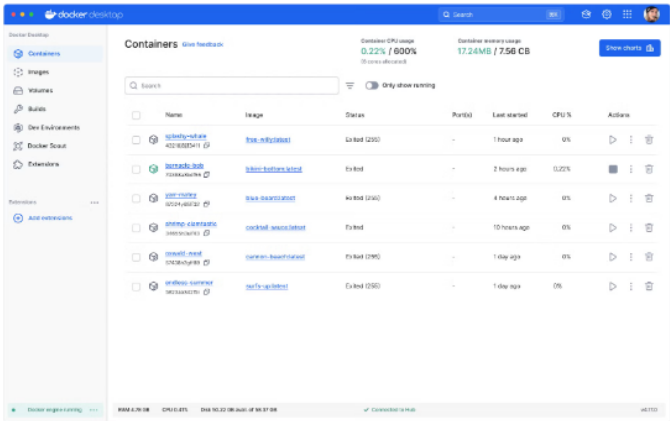
Окно docker desktop:



Рис. 3.18. Добавленный материал

Свободное место	От 5 Гб на диске
-----------------	------------------

Окно docker desktop:



Вложения

- [Docker.pdf](#)

Назад

Редактировать

Удалить

Комментарии

Напишите комментарий...

Отправить

Рис. 3.19. Добавленный материал

Редактирование материала происходит при нажатии на кнопку «Редактировать» (рис. 3.20).

The screenshot shows the Docker Desktop application window. On the left is a sidebar with navigation icons for Containers, Images, Volumes, Builds, Dev Environments, Docker Scout, and Extensions. The main area is titled 'Containers' and shows a table of running containers. At the top right, it displays 'Container CPU usage: 0.22% / 600%' and 'Container memory usage: 17.24MB / 7.56 GB'. The table has columns for Name, Image, Status, Ports, Last started, CPU %, and Actions. Several containers are listed, including 'nginx-proxy', 'nginx-proxy', 'nginx-proxy', 'nginx-proxy', 'nginx-proxy', 'nginx-proxy', and 'nginx-proxy'. At the bottom, there is a status bar showing 'Docker engine running' and system information.

Теперь вы готовы использовать Docker для создания и запуска контейнеризированных приложений.

Рис. 3.20. Редактирование материала

При нажатии кнопки «Сохранить изменения» статья обновляется и добавляется дата обновления (рис. 3.21).

## Установка Docker

Автор: Иванов Иван Иванович   Создан: 20.06.2025   Обновлено: 20.06.2025

Категории: Сетевые технологии

Рис. 3.21. Дата обновления

После публикации материала у пользователей есть возможность добавления комментариев к материалу (рис. 3.22).

### Вложения

- [Docker.pdf](#)

Назад

Редактировать

Удалить

### Комментарии

Напишите комментарий...

Отправить

Иванов Иван Иванович   20.06.2025, 16:25:11

отличный материал!

Удалить

Рис. 3.22. Добавление комментария

Удаляется материал при нажатии на кнопки «Удалить» (рис. 3.23).

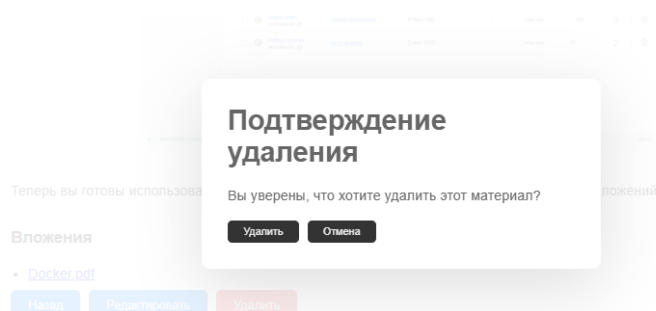


Рис. 3.23. Удаление материала

В случае отсутствия у пользователя прав на удаление материала, выдается ошибка (рис. 3.24).

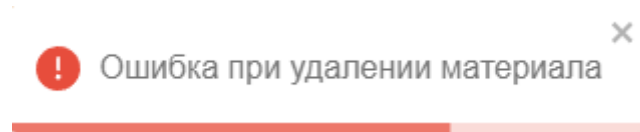


Рис. 3.24. Ошибка удаления материала

Пользователь также может просмотреть материалы, относящийся к конкретной категории, нажав на иконку нужной категории (рис. 3.25-3.27).

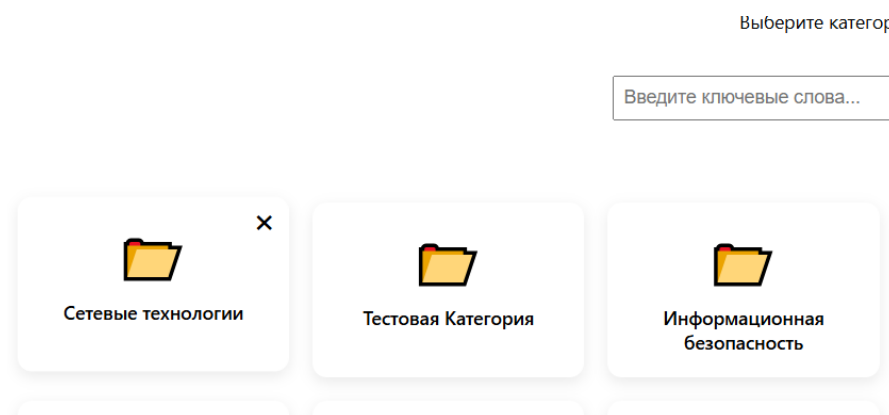


Рис. 3.25. Категории

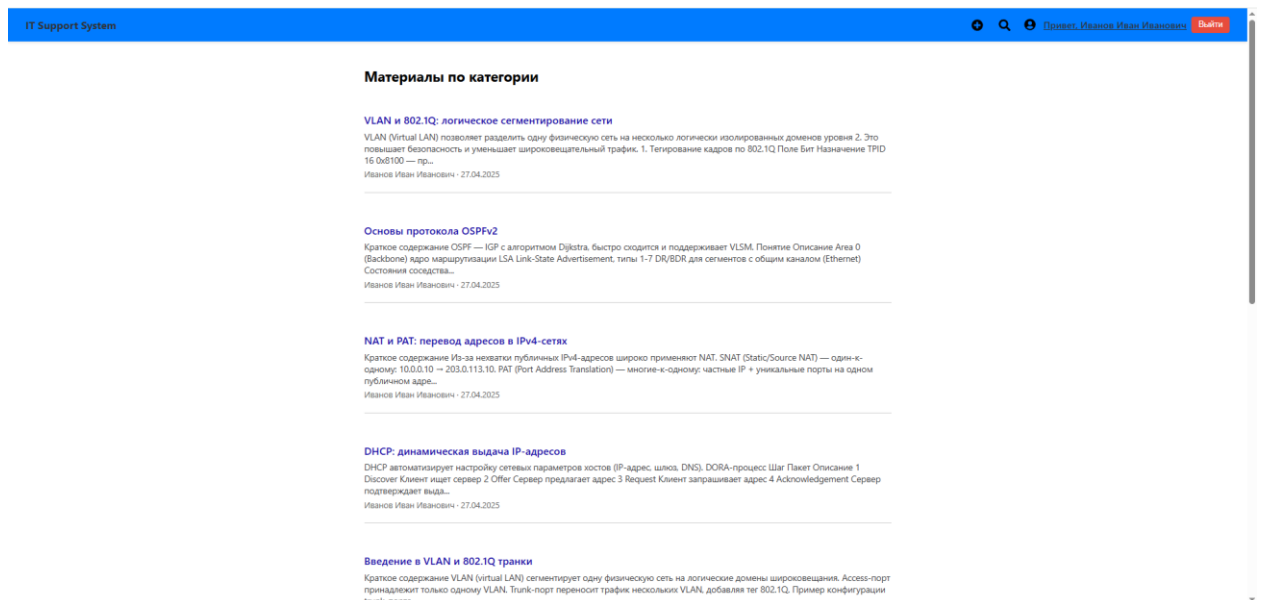


Рис. 3.26. Переход к материалам категории

### Docker введение

Чтобы понять суть докера, сперва углубимся в историю. Раньше доставка грузов по всему миру осуществлялась в бочках, коробках и прочих ёмкостях. Проблема была в том, что все они были разных размеров и форм. Для разгрузки корабля требовалась уйма време...

Иванов Иван Иванович · 19.04.2025

### Информационные технологии

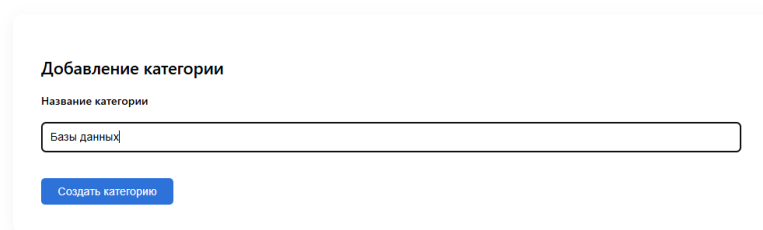
Информационные технологии — область знаний и практических навыков, связанных с использованием компьютеров и программного обеспечения для сбора, обработки, хранения, передачи и анализа информации.

Иванов Иван Иванович · 10.04.2025

Назад 1 / 2 Вперёд

Рис. 3.27. Пагинация при просмотре материалов

Добавим категорию, нажав на кнопку «Добавить категорию» на главной странице (рис. 3.28).



Добавление категории

Название категории

Базы данных

Создать категорию

Рис. 3.28. Создание категории

После добавления категория отобразится на главной странице (рис. 3.29).

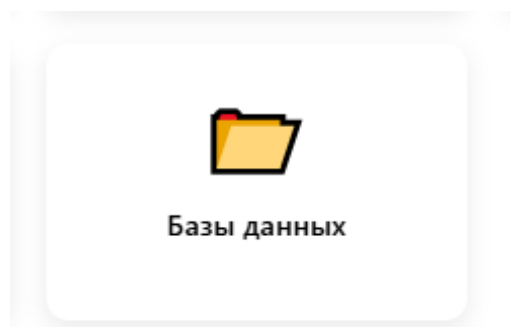


Рис. 3.29. Добавленная категория

При отсутствии материалов в категории показывается соответствующее уведомление (рис. 3.30).

## Материалы по категории

В этой категории пока нет материалов.

Рис. 3.30. Надпись об отсутствии материалов в категории

В свой профиль можно перейти по клику на иконку или ФИО в шапке системы (рис. 3.31).

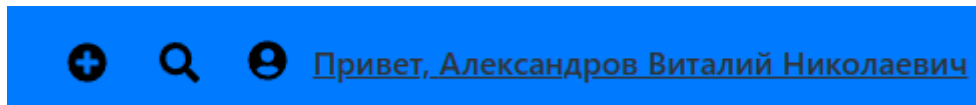


Рис. 3.31. Шапка системы

В профиле находится информация о пользователе, с возможностью редактирования имени, смены почты и пароля (рис. 3.32).

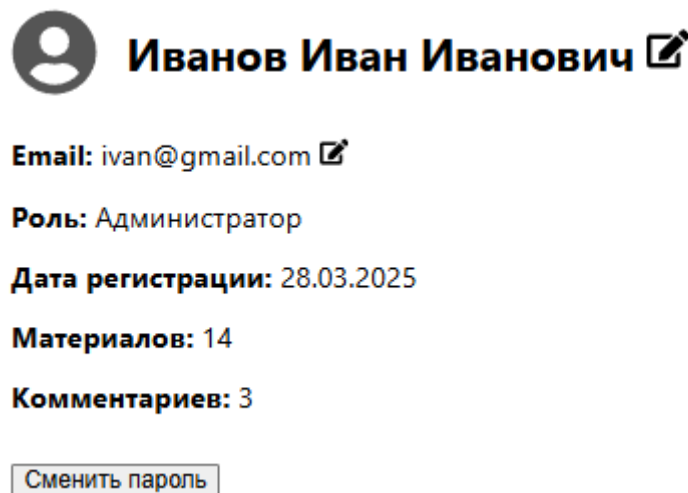


Рис. 3.32. Профиль пользователя

Поиск пользователей осуществляется по клику соответствующей иконки в шапке системы. Введем ключевые слова в строку поиска (рис. 3.33).

## Поиск пользователей

Поиск

 Иванов Александр Иванович



 Иванов Иван Иванович


 Иванов Николай Николаевич


 Иванов Олег Федорович

Рис. 3.33. Поиск пользователей

При переходе в профиль пользователя, помимо просмотра информации о пользователе администратор может задать ему роль и сменить имя (рис. 3.34).

 **Иванов Олег Федорович** 

**Email:** oleg@gmail.com 

**Роль:** Читатель 

**Дата регистрации:** 26.04.2025

**Материалов:** 0

**Комментариев:** 0

Заблокировать

Рис. 3.34. Просмотр чужого профиля

Поиск материалов в системе осуществляется с помощью ввода ключевых слов в строку поиска (рис. 3.35).

## Поиск по базе знаний

Выберите категорию, чтобы просмотреть материалы

установка docker

### Результаты поиска:

#### Установка Docker

платформа для быстрой разработки, тестирования и развертывания приложений в изолированных контейнерах. **Установка Docker** проста и занимает всего несколько шагов. Инструкция по **установке**: [Перейдите на официальный ... установки](#) перезагрузите компьютер (для Windows и macOS). Требования к системе для **установки Docker**: Требование Минимальные параметры Операционная система Windows 10/11 (Pro/Enterprise) или macOS 10.15+ Процессор

Иванов Иван Иванович · 20.06.2025

Рис. 3.35. Поиск материалов