

**МОСКОВСКИЙ АВИАЦИОННЫЙ ИНСТИТУТ  
(НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ)**

**Институт №8 «Компьютерные науки и прикладная математика»  
Кафедра №806 «Вычислительная математика и программирование»**

**Курсовая работа  
по курсу «Базы Данных»**

**Система управления библиотекой научных статей**

**Выполнил: Кострюков Е. С.  
Группа: М8О-307Б-22  
Преподаватель: Малахов А. В.**

**Москва, 2024**

## Оглавление

<b>СХЕМА БАЗЫ ДАННЫХ.....</b>	<b>3</b>
<b>СХЕМА ПРИЛОЖЕНИЯ. МОДЕЛИ.....</b>	<b>4</b>
<b>ОПИСАНИЕ ОСНОВНЫХ МОДЕЛЕЙ.....</b>	<b>5</b>
<b>КРАТКОЕ ОПИСАНИЕ РАБОТЫ ПРИЛОЖЕНИЯ.....</b>	<b>5</b>
<b>QR-КОД НА GIT.....</b>	<b>6</b>
<b>ВЫВОД.....</b>	<b>6</b>

## Схема базы данных

Таблицы:

1. **article\_authors** - содержит информацию о связи между статьями и авторами. Включает идентификаторы статьи и автора, обеспечивая связь "многие ко многим" между таблицами.
2. **articles** - таблица, содержащая основную информацию о статьях, включая идентификатор статьи, название, год публикации, ссылку на статью и идентификатор пользователя, который загрузил статью.
3. **authors** - таблица авторов, содержащая информацию о каждом авторе, включая его идентификатор, имя и фамилию.
4. **article\_keywords** - таблица связи между статьями и ключевыми словами, содержащая идентификаторы статьи и ключевого слова, обеспечивая связь "многие ко многим" между таблицами articles и keywords.
5. **keywords** - таблица, содержащая ключевые слова, включающие идентификатор ключевого слова и само ключевое слово.
6. **users** - таблица пользователей, содержащая информацию о каждом пользователе, включая его идентификатор, электронную почту, пароль, имя, фамилию, дату регистрации и роль пользователя.
7. **user\_favorites** - таблица избранных статей пользователя, содержащая идентификаторы пользователя и статьи, обеспечивая связь "многие ко многим" между таблицами users и articles.



## Схема приложения. Модели

Приложение состоит из следующих компонентов:

### 1. Frontend (Streamlit)

Пользовательский интерфейс реализован с использованием библиотеки **Streamlit**. Приложение предоставляет следующие страницы:

- **Главная страница:** позволяет пользователю просматривать доступные статьи с их основной информацией, включая название, год публикации и ссылку на статью.
- **Профиль пользователя:** отображает информацию о пользователе, позволяет просматривать все добавленные в избранное статьи.
- **Избранные статьи:** позволяет пользователю просматривать и управлять статьями, которые он добавил в избранное.
- **Панель администратора** (только для администраторов): предоставляет функционал для управления статьями, включая редактирование данных о статьях, добавление и удаление статей.
- Каждая из страниц имеет отдельный файл, отвечающий за логику и взаимодействие с пользователем.

### 2. Backend

Взаимодействие с базой данных происходит через слой **репозиториев**. Репозитории содержат функции для выполнения SQL-запросов, таких как выборка данных о статьях, пользователях, добавление в избранное и управление администрированием.

**Backend структурирован следующим образом:**

- **repositories:** модуль для работы с БД, включает файлы для управления статьями, пользователями, избранными статьями и администраторами.
- **business:** уровень бизнес-логики, где выполняются вычисления и операции с данными перед передачей в репозиторий, такие как регистрация пользователей, авторизация и управление ролями.

### 3. База данных

База данных разработана с учетом нормализации данных. Основные таблицы включают:

- **users:** хранит информацию о пользователях, включая email, имя, фамилию, роль (пользователь или администратор), и другие данные.
- **articles:** хранит информацию о статьях, включая название, год публикации, ссылку на статью и id пользователя, который загрузил статью.

- **authors:** содержит данные об авторах статей, включая их имя и фамилию.
- **keywords:** хранит ключевые слова, связанные с статьями.
- **user\_favorites:** хранит данные о том, какие статьи были добавлены в избранное пользователями.
- **article\_authors:** связывает статьи с их авторами.
- **article\_keywords:** связывает статьи с ключевыми словами.

## Описание основных моделей

В приложении используются следующие модели данных:

- **Пользователь (User):** описывает данные о пользователе, включая email, имя, фамилию, роль (администратор/пользователь), дату регистрации.
- **Статья (Article):** содержит информацию о статье, включая название, ссылку, год публикации и пользователя, который загрузил статью.
- **Автор (Author):** содержит данные о авторе, такие как имя и фамилия.
- **Ключевое слово (Keyword):** описывает ключевые слова, которые можно привязать к статьям для улучшения поиска.
- **Избранные статьи (UserFavorites):** связывает пользователей и статьи, добавленные в избранное.
- **Статья-Автор (ArticleAuthors):** связывает статьи с авторами.
- **Статья-Ключевое слово (ArticleKeywords):** связывает статьи с ключевыми словами.

## Краткое описание работы приложения

1. **Авторизация:**
  - Пользователь вводит email и пароль. Если email найден в базе данных и пароль верен, приложение предоставляет доступ к функционалу пользователя или администратора в зависимости от роли.
2. **Функционал пользователя:**
  - Просмотр доступных статей с их основной информацией, такой как название, год публикации и ссылка на статью.
  - Просмотр профиля пользователя с возможностью редактирования данных.

- Управление избранными статьями (добавление и удаление из избранного).

### **3. Функционал администратора:**

- Управление статьями, включая добавление, редактирование и удаление статей.
- Просмотр доступных статей с их основной информацией, такой как название, год публикации и ссылка на статью.
- Просмотр профиля пользователя с возможностью редактирования данных.
- Управление избранными статьями (добавление и удаление из избранного).

### **QR-код на Git**



### **Вывод**

В ходе работы было разработано веб-приложение для управления научными статьями с использованием Streamlit. Реализованы страницы для регистрации, авторизации, просмотра статей и избранных материалов, а также панель администратора для управления статьями. В проекте использована система сессий для хранения данных о пользователе, а также обеспечены проверки на корректность ввода данных. Это позволяет пользователю с удобством взаимодействовать с приложением в зависимости от его роли (пользователь или администратор).