МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

МЕХАНИКО-МАТЕМАТИЧЕСКИЙ ФАКУЛЬТЕТ

КАФЕДРА ВЕБ-ТЕХНОЛОГИЙ И КОМПЬЮТЕРНОГО МОДЕЛИРОВАНИЯ

**СОЗДАНИЕ ОНЛАЙН СПРАВОЧНИКА НАУЧНЫХ ЖУРНАЛОВ БЕЛАРУСИ**

Курсовая работа

Стукача

Дмитрия Павловича

студента 3 курса,

специальность 1-31 03 08-01

Математика и информационные технологии (веб-программирование и интернет-технологии)

Научный руководитель:  
доцент, кандидат физ.-мат. наук   
Ю. В. Позняк

Минск, 2021

**БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ**

Факультет \_\_\_**ММФ**\_\_\_ Кафедра \_\_\_**ВТиКМ**\_\_\_

**«УТВЕРЖДАЮ»**

Зав. кафедрой \_В.М.Волков

«\_20\_» \_\_\_октября\_\_\_ 2020\_г.

**ЗАДАНИЕ**

**НА ВЫПОЛНЕНИЕ КУРСОВОЙ РАБОТЫ**

**Студенту** \_Стукачу Дмитрию Павловичу \_\_\_**курса** \_\_3\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**1. Тема курсовой работы** \_**Создание онлайн справочника научных журналов Беларуси\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**утверждена на заседании кафедры протокол № \_\_2\_\_\_\_ от «\_30\_\_» \_\_\_\_сентября\_\_\_\_ 2015\_ года

**2. Срок сдачи студентом законченной работы** \_\_\_\_\_\_\_\_\_\_03 июня 2021 года\_\_\_\_\_\_\_\_

**3. Библиографические описания источников, рекомендуемых студентам к ознакомлению при выполнении работы:**

1) https://metanit.com/web/nodejs\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

2) www.wikipedia.org

3) www.htmlbook.ru.\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

4) www.it-academy.by

5) https://www.mysql.com\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

6) https://nodejs.org

**4. Перечень вопросов подлежащих разработке или краткое содержание работы**

1) Изучение материалов

2) Изучение Node.js и фреймворка Express\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

3) Создание базы-данных

4) Создание веб-приложения\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

5) Создание презентации и отчета

**5. Календарный график работы на весь период** (с указанием этапов работы и сроков их выполнения). **Даты консультаций и промежуточных отчетов.**

1) Изучение материалов: 10.10.2020-11.12.2020\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

2) Изучение технологий: 10.01.2021-01.03.2021

3) Постановка задачи: 02.03.2021-16.03.2021\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

4) Проектирование платформы: 17.02.2021-25.03.2021

5) Решение задач: 26.03.2021-20.05.2021\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

6) Подготовка отчета о проделанной работе: 01.05.2021-31.05.2021

**6. Задание принял к исполнению** «\_20\_» октября\_ 2020г. \_\_\_\_\_\_\_\_подпись студента

**Руководитель** \_\_\_\_\_\_Позняк.Ю.В.\_\_\_\_\_\_\_\_\_\_ / \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

ОГЛАВЛЕНИЕ

[**ВВЕДЕНИЕ** 4](#_Toc40971867)

[**ГЛАВА 1. АНАЛОГИ. ИСПОЛЬЗУЕМЫЕ ИНСТРУМЕНТЫ.** 6](#_Toc40971868)

[1.1. Обзор аналогов 6](#_Toc40971869)

[1.2. Структура веб-приложения. Паттерн MVC. 6](#_Toc40971870)

[1.3. Теоретические сведения о базе данных MySQL. 7](#_Toc40971870)

[1.4. Теоретические сведения о серверной части веб-приложения. Node.js. Фреймворк Express.js 8](#_Toc40971870)

[1.5. Теоретические сведения об интсрументе разработки представлений Handlebars. 9](#_Toc40971870)

[**ГЛАВА 2. РЕАЛИЗАЦИЯ ВЕБ ПРИЛОЖЕНИЯ** 10](#_Toc40971871)

[2.1. Разработка серверной части веб-приложения 10](#_Toc40971872)

[2.2. Разработка клиентской части веб-приложения 11](#_Toc40971873)

[**ЗАКЛЮЧЕНИЕ** 14](#_Toc40971877)

[**СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ** 15](#_Toc40971878)

### ВВЕДЕНИЕ

Актуальность. В настоящее время сфера научных коммуникаций во всем мире переживает период коренной трансформации, который характеризуется постоянным увеличением количества научных публикаций и научных журналов, расширением международного сотрудничества в рамках исследований, а также переход к использованию цифровых платформ и инструментов.В Беларуси также наблюдается постепенное изменение подходов к изданию научных журналов и их интеграции в мировое информационное пространство. Хотя Инструкция ВАК Республики Беларусь не предъявляет требований о представлении журнала в сети Интернет и наличии доступа к цифровым архивам, большинство журналов имеют сайты, некоторые из них предоставляют открытый доступ к статьям.

Кроме того, более половины журналов из Перечня ВАК Беларуси на сегодняшний день индексируются «Российским индексом научного цитирования (РИНЦ), что существенно повышает видимость их контента. Вхождение же в наиболее авторитетные в мире реферативные базы данных «Scopus» и «Web of Science» пока смогли себе обеспечить лишь несколько белорусских периодических изданий.

Таким образом, перед автором сегодня часто возникает вопрос: насколько результаты его исследования будут иметь распространение после публикации и какое они окажут влияние на развитие исследований по данному направлению? Все более очевидным становится то, что от того, какой у журнала сайт, существует ли его англоязычная версия, предоставляется ли доступ к архивам, осуществляется ли индексирование различными базами данных, агрегаторами и каталогами зависит охват целевой аудитории, влияние и цитируемость опубликованной статьи. В свою очередь, повышение видимости и цитируемости статей ведет к росту репутации ученого и вносит свой вклад в показатели учреждения, где он работает. Для учреждений образования эти показатели прямо влияют на их позиции в международных рейтингах университетов.

Цель - для упрощения процедуры выбора журнала, оценки его открытости и интегрированности в систему международных научных коммуникаций создать онлайн справочник научных журналов, включенных в Перечень ВАК, основываясь на предоставленных данных [1].

В рамках данной работы, Фундаментальной библиотекой БГУ были предоставлены данные в виде таблицы обо всех научных журналах в Республике Беларусь. Поэтому задача создания онлайн справочника разбивается на следующие подзадачи:

* Создание базы данных научных журналов
* Разработка серверной части веб-приложения
* Разработка клиентской части веб-приложения
* Разработка интерфейса
* Размещение справочника на сайте Фундаментальной библиотеки БГУ

По функциональности справочник должен выполнять следующее:

* Вывод наиболее важной информации о научных журналах на страницу
* Вывод на страницу всей информации о конкретном журнале при выборе его из списка
* Поиск журнала по названию
* Поиск журналов по научному направлению

Для корректировки и обновления информации со стороны работников библиотеки в справочнике должна быть предусмотрена роль «администратор». Функции, реализуемые пользователем с ролью «администратор»:

* Возможность безопасной авторизации с определенным логином и паролем для получения полномочий редактирования
* Добавление нового журнала в базу данных и внесение информации о нем
* Редактирование информации о текущих журналах в базе данных
* Безопасный выход из роли «администратор»

### 

### ГЛАВА 1 АНАЛОГИ. ИСПОЛЬЗУЕМЫЕ ИНСТРУМЕНТЫ.

## **Обзор аналогов**

По реализуемой функциональности в Беларуси прямых аналогов нет.

Согласно проведенным исследованиям можно выделить следующие похожие электронные источники информации:

* «Журналы БГУ» [2],
* «Портал научных журналов БНТУ» [3],
* Раздел «Журналы для опубликования результатов научных исследований» БГУИР [4],
* «Национальный правовой интернет портал РБ» [5]

Стоит отметить, что приведенные выше источники информации затрагивают лишь некоторую часть научных журналов Беларуси. Это освещение информации либо в каком-то определенном научном направлении, либо освещение исключительно наиболее популярных научных журналов. Поэтому онлайн справочник научных журналов Беларуси – достаточно полезный информационный ресурс, из-за использования информации абсолютно обо всех журналах Беларуси.

## **Структура веб-приложения. Паттерн MVC.**

Веб-приложение – клиент-серверное приложение, в котором клиент взаимодействует с веб-сервером при помощи браузера. [Логика](https://ru.wikipedia.org/wiki/%D0%9B%D0%BE%D0%B3%D0%B8%D0%BA%D0%B0) веб-приложения распределена между сервером и клиентом, хранение данных осуществляется, преимущественно, на сервере, обмен информацией происходит по [сети](https://ru.wikipedia.org/wiki/%D0%9A%D0%BE%D0%BC%D0%BF%D1%8C%D1%8E%D1%82%D0%B5%D1%80%D0%BD%D0%B0%D1%8F_%D1%81%D0%B5%D1%82%D1%8C). Одним из преимуществ такого подхода является тот факт, что клиенты не зависят от конкретной [операционной системы](https://ru.wikipedia.org/wiki/%D0%9E%D0%BF%D0%B5%D1%80%D0%B0%D1%86%D0%B8%D0%BE%D0%BD%D0%BD%D0%B0%D1%8F_%D1%81%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0) пользователя, поэтому веб-приложения являются межплатформенными службами. При реализации онлайн справочника клиент использует сайт для посылания HTTP-запросов на сервер, который взаимодействует с базой данных и отправляет результат обратно на страницу пользователю. При разработке приложений часто используются общепринятые принципы или паттерны, применяя которые можно понятно разбить весь проект на слои, каждый из которых отвечает за определенную функцию.

Паттерн MVC(Model-View-Controller) – схема разделения данных приложения, пользовательского интерфейса и управляющей логики на три компонента для удобной модификации в процессе разработки каждого из них. Как видно на рисунке 1, пользователь в интерфейсе вводит запрос, который реализуется контроллером при взаимодействии с базой данных, и получает результат в виде обновленной информации в представлении. Если пользовательский запрос не содержит обращения к базе данных, то контроллер обновляет представление, согласно введенному запросу, например, переход на другую страницу [6].

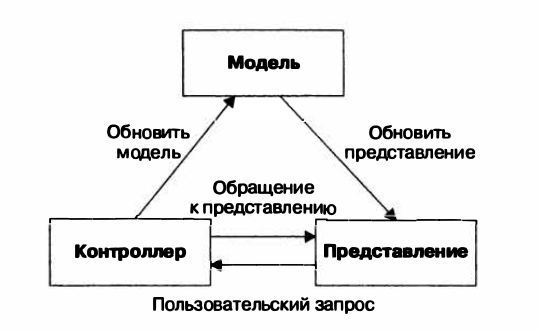


Рисунок 1 Структура веб-приложения с использованием паттерна MVC

В онлайн справочнике паттерн MVC используется следующим образом:

* модель – журнал в базе данных,
* контроллер – логика на Node.js,
* интерфейс – страница с использованием движка представлений Handlebars.
  1. **Теоретические сведения о базе данных MySQL**

В качестве исходных данных о научных журналах использовалась таблица, где находилась информация о каждом журнале. Для организации взаимодействия информации с клиентом в веб-приложения вся таблица Excel была переведена в таблицу базы данных MySQL.

MySQL – свободная реляционная система управления базами данных. Разработку и поддержку MySQL осуществляет корпорация Oracle. Гибкость СУБД MySQL обеспечивается поддержкой большого количества типов таблиц: пользователи могут выбрать как таблицы типа [MyISAM](https://ru.wikipedia.org/wiki/MyISAM" \o "MyISAM), поддерживающие полнотекстовый поиск, так и таблицы [InnoDB](https://ru.wikipedia.org/wiki/InnoDB" \o "InnoDB), поддерживающие транзакции на уровне отдельных записей. Более того, СУБД MySQL поставляется со специальным типом таблиц EXAMPLE, демонстрирующим принципы создания новых типов таблиц. Благодаря открытой архитектуре и GPL-лицензированию, в СУБД MySQL постоянно появляются новые типы таблиц [7].

В качестве основной единицы в таблице используется научный журнал с его подробной информацией. Для взаимодействия с Node.js используется драйвер mysql2, который предоставляет функции и обеспечивает более высокую производительность. Возможность взаимодействия с базой данных через коннекторы различных языков позволит не создавать новую базу данных, а использовать одну для нескольких приложений. Для создания онлайн справочника база данных MySQL была выбрана по следующим причинам:

* Удобство в администрировании
* Доступность взаимодействия с логикой справочника
* Общеизвестность в среде применения
* Удобство для малых и средних типов приложений, реализующееся через достаточную вместимость данных в каждую таблицу и высокую производительность в таких рамках
* Удобство взаимодействия с инструментом реализации серверной части справочника
  1. **Теоретические сведения о серверной части веб-приложения. Node.js. Фреймворк Express.js.**

Node.js – программная платформа, основанная на движке V8, расширяющая возможности языка JavaScript. Node.js добавляет возможность JavaScript взаимодействовать с устройствами [ввода-вывода](https://ru.wikipedia.org/wiki/%D0%92%D0%B2%D0%BE%D0%B4-%D0%B2%D1%8B%D0%B2%D0%BE%D0%B4) через свой [API](https://ru.wikipedia.org/wiki/API), подключать другие внешние библиотеки, обеспечивая вызовы к ним из JavaScript-кода. Node.js применяется преимущественно на сервере, выполняя роль [веб-сервера](https://ru.wikipedia.org/wiki/%D0%92%D0%B5%D0%B1-%D1%81%D0%B5%D1%80%D0%B2%D0%B5%D1%80), но есть возможность разрабатывать на Node.js и десктопные оконные приложения и даже программировать микроконтроллеры. Модульная структура языка позволяет включать зависимости по мере развития приложения от взаимодействия с файлами до создания токенов для обеспечения безопасного подключения. В основе Node.js лежит [событийно-ориентированное](https://ru.wikipedia.org/wiki/%D0%A1%D0%BE%D0%B1%D1%8B%D1%82%D0%B8%D0%B9%D0%BD%D0%BE-%D0%BE%D1%80%D0%B8%D0%B5%D0%BD%D1%82%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%BD%D0%BE%D0%B5_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5) программирование с [неблокирующим вводом/выводом](https://ru.wikipedia.org/wiki/%D0%9D%D0%B5%D0%B1%D0%BB%D0%BE%D0%BA%D0%B8%D1%80%D1%83%D1%8E%D1%89%D0%B0%D1%8F_%D1%81%D0%B8%D0%BD%D1%85%D1%80%D0%BE%D0%BD%D0%B8%D0%B7%D0%B0%D1%86%D0%B8%D1%8F) [8]. В рамках онлайн справочника Node.js выполняет роль веб-сервера, по которому происходит взаимодействие логики с базой данных. Для облегчения написания приложения и упрощения создания API использовался фреймворк Express.js.

Express.js, или просто Express, [фреймворк web-приложений](https://ru.wikipedia.org/wiki/%D0%A4%D1%80%D0%B5%D0%B9%D0%BC%D0%B2%D0%BE%D1%80%D0%BA) для Node.js, реализованный как [свободное и открытое программное обеспечение](https://ru.wikipedia.org/wiki/%D0%A1%D0%B2%D0%BE%D0%B1%D0%BE%D0%B4%D0%BD%D0%BE%D0%B5_%D0%B8_%D0%BE%D1%82%D0%BA%D1%80%D1%8B%D1%82%D0%BE%D0%B5_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%BD%D0%BE%D0%B5_%D0%BE%D0%B1%D0%B5%D1%81%D0%BF%D0%B5%D1%87%D0%B5%D0%BD%D0%B8%D0%B5) под [лицензией MIT](https://ru.wikipedia.org/wiki/%D0%9B%D0%B8%D1%86%D0%B5%D0%BD%D0%B7%D0%B8%D1%8F_MIT). Он спроектирован для создания [веб-приложений](https://ru.wikipedia.org/wiki/%D0%92%D0%B5%D0%B1-%D0%BF%D1%80%D0%B8%D0%BB%D0%BE%D0%B6%D0%B5%D0%BD%D0%B8%D0%B5) и [API](https://ru.wikipedia.org/wiki/API). Де-факто является стандартным каркасом для Node.js. Express может являться backend'ом для программного стека [MEAN](https://ru.wikipedia.org/wiki/MEAN_(%D0%B2%D0%B5%D0%B1-%D1%80%D0%B0%D0%B7%D1%80%D0%B0%D0%B1%D0%BE%D1%82%D0%BA%D0%B0)), вместе с базой данных [MongoDB](https://ru.wikipedia.org/wiki/MongoDB" \o "MongoDB) и каркасом [Vue.js](https://ru.wikipedia.org/wiki/Vue.js), [React](https://ru.wikipedia.org/wiki/React" \o "React) или [AngularJS](https://ru.wikipedia.org/wiki/AngularJS" \o ") для frontend'а.

Так как Node.js имеет модульную структуру, то опишем функции, выполняемые следующими модулями:

* JWT (JSON Web Token) — модуль для создания токенов доступа, основанный на формате JSON. Используется для передачи данных для аутентификации в клиент-серверных приложениях,
* mysql2 — драйвер для организации соединения с базой данных MySQL и выполнения команд в базу данных при написании http-запросов
* body-parser — промежуточное программное обеспечении, анализирующее тело запроса для получения данных,
* password-hash — модуль для хэширования пароля при авторизации пользователя внутри системы, поддерживает несколько функций хеширования чем достаточно удобен при снижении нагрузки на сервер,
* nodemon — модуль для беспрерывной разработки приложения внутри локального сервера.
  1. **Теоретические сведения об инструменте разработки представлений Handlebars**

Движок представлений Handlebars предоставляет возможности, необходимые для эффективного создания **семантических шаблонов**. Это включает в себя HTML с встроенными выражениями-помощниками, которые реализуют взаимодействие с основной логикой приложения. При грубом тесте производительности предварительно скомпилированные шаблоны Handlebars.js (в исходной версии Handlebars.js) обрабатывались примерно вдвое быстрее, чем шаблоны Mustache. Этот факт также идет в пользу использования Handlebars для создания справочника, где используется вывода достаточно большого количества информации на страницу [9]. В справочнике, помимо шаблона, используются хелперы для вывода результата запроса в базу данных, а также для сокрытия информации, предназначенной к использованию лишь авторизованному пользователю. Разработан для работы в любой среде ECMAScript 3. Это включает в себя:

* Node.js 10+
* Хром
* Fire Fox
* Safari 5+
* Opera 11+
* IE 6+

Однако в плане удобства дизайна Handlebars.js имеет некоторый минус из-за неудобного прописывания стилей и скриптов для веб-сайта. Однако данная проблема решается путем подключения хелперов, обеспечивающих взаимодействие c css и js скриптами. Таким образом, движок представлений Handlebars полностью выполняет функции по дизайну и разработке клиентской части веб-приложения, поддерживая HTML, CSS и JS скрипты, а также обладая взаимодействием с серверной частью, написанной на Node.js.

### ГЛАВА 2 РЕАЛИЗАЦИЯ ВЕБ-ПРИЛОЖЕНИЯ

### 2.1 Разработка серверной части веб-приложения

### Для создания серверной части онлайн справочника используется Node.js с фреймворком Express.js. Создается локальный сервер с определенным портом, на котором выполняется тестовый вариант справочника. Код функции создания сервера описан в листинге 2.1.1.

### Листинг 2.1.1 Код функции создания локального сервера

const PORT = 3000;

const app = express();

const start = () => {

    try {

        app.listen(PORT, () => console.log("Server started at 3000 port"));

    } catch (error) {

        console.log(error);

    }

}

start();

Здесь app – название созданного приложения на фреймворке Express.js, которое прослушивает свободный локальный порт. В данном случае, это порт 3000. И в качестве подтверждения успешного выполнения операции выводит в логи соответствующее сообщение. При ошибке, выводится сообщение ошибки. Для подключения к базе данных требуется создание пула. Пул соединений с базой данных - набор заранее открытых соединений с базой данных используемый для предоставления соединения в тот момент, когда оно требуется. Код создания пула с базой данных представлен в листинге 2.1.2.

### Листинг 2.1.2 Код создания подключения к базе данных

const mysql = require("mysql2");

const pool = mysql.createPool({

    connectionLimit: 5,

    host: "localhost",

    user: "root",

    database: "sciencemagazines",

    password: ""

});

Использование драйвера mysql2 с помощью команды require позволяет использовать метод, который позволяет создать подключение с локального хоста к базе данных с именем «sciencemagazines» при желании можно поставить пароль для безопасности при вход в базу данных через редактор, однако тогда придется вводить его и при создании подключения, поэтому в тестовом варианте пароль отсутствует.

Для соблюдения паттерна MVC используется роутер, с помощью которого по определенному пути идет вызов определенной функции из файла контроллера. А сама логика экспортируется в основной файл, для уменьшения нагрузки на понимание со стороны разработчика. Фреймворк Express.js имеет метод Router(), благодаря которому происходит выполнение данного принципа. Код логики работы роутера представлен в листинге 2.1.3.

### Листинг 2.1.2 Код создания подключения к базе данных

const router = express.Router();

const controller = require("./controller.js");

router.get(“/search”, controller.getMagazines);

router.get(“/individualInfo/:id”, controller.getIndividualMagazine);

router.post(“/searchByName”, controller.findMagazinesByName);

router.post(“/searchByScientificDirections”, controller.findMagazinesByScientificDirections);

router.get(“/create”, controller.redirectForCreateMagazinePage);

router.post(“/create”, controller.createMagazine);

router.get(“/edit/:id”, controller.getMagazineToEditById);

router.post(“/edit”, controller.editMagazine);

router.post(“/delete/:id”, controller.deleteMagazineById);

router.post(“/logIn”, controller.logInUser);

router.get(“/logout”, controller.logOutUser);

module.exports = router;

### 2.2 Разработка клиентской части веб-приложения

### Для выполнения всех функций онлайн справочника научных журналов используются http-запросы, которые бывают в основном двух видов:

### GET-запросы, которые в основном производят вывод данных

### POST-запросы, которые часто взаимодействуют с базой данных, редактируя в ней информацию или принимая данные от пользователя производят некоторые действия с ними.

### Ниже представлен код POST-запроса, который выполняет функцию поиска научного журнала по имени в онлайн справочнике. Код представлен в листинге 2.2.1.

### Листинг 2.2.1 Код функции поиска журнала по названию

async findMagazinesByName(req, res) {

        if (!req.body) return res.sendStatus(400);

        const name = req.body.name;

        pool.query(`SELECT \* FROM sciencemagizenes WHERE nameEng LIKE "%"?"%"`, [name], function(err, result) {

            if (err) return console.log(err.message);

            res.render("search.hbs", {

                users: result,

                adminStatus

            });

        });

    }

Данная функция получает данные с поля поиска по имени и с помощью модуля body-parser передает их в переменную «name». Далее с помощью метода запроса в базу данных производится выборка полных данных всех журналов, чье имя на английском содержит значение из переменной «name». С использованием метода render() результаты выборки выводятся пользователю на страницу. Таким образом, даже введя неточное название или название журнала не полностью можно с высокой вероятностью найти нужный. Это был пример POST-запроса, так как принимались данные, вводимые пользователем.

Для примера GET-запроса приведен код функции вывода полной информации о конкретном журнале на страницу в листинге 2.2.2.

### Листинг 2.2.2 Код функции вывода полной информации о выбранном журнале

async getIndividualMagazine(req, res) {

        const id = req.params.id;

        pool.query("SELECT \* FROM sciencemagizenes WHERE id=?", [id], function(err, data) {

            if (err) return console.log(err.message);

            res.render("individualInfo.hbs", {

                user: data[0]

            });

        });

    }

### Здесь уже пользователь не вводит никакие данные для функции, он просто выбирает область на странице, которая занимается краткой информацией конкретного журнала. При нажатии на область, занимаемой конкретным журналом, происходит событие, при котором пользователь переносится на страницу, на которую выводиться полная информация о выбранном журнале. Таким образом, по «id» определяется журнал в базе данных и выводится информация о нем на страницу «individualInfo» с помощью render().

### Подобными функциями реализован весь онлайн справочник по научным журналам Беларуси.

### ЗАКЛЮЧЕНИЕ

В качестве перспектив развития следует заметить, что поиск авторитетных научных журналов весьма важная задача для ученых, поэтому основным направлением развития функциональности приложения может стать ранжирование журналов по их популярности, доступности и цитируемости. Для решения этой задачи может быть введен рейтинг журналов, ведь важно не только найти список всех доступных журналов, но и выделить самые влиятельные, что значительно сэкономит время, направленное на выбор журнала для публикации.

Справочник планируется разместить на сайте Фундаментальной библиотеки БГУ, а также он будет предложен для размещения на сайтах других вузовских библиотек Беларуси. Подобный сервис, несомненно, окажет помощь ученым в подборе наиболее авторитетных научных журналов Беларуси, особенно полезным он будет для молодых ученых [1].

### СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Научная статья «Справочник журналов ВАК» [Электронный ресурс]. — 2021. — Режим доступа URL:

<https://docs.google.com/document/d/1q_hLNg03Da5PiX4FDLxiis_zL5-eVXwp-K6Ec-jtw5w/edit?ts=607c4e7e>.

Дата доступа: 13.04.2021

1. Сайт «Журналы БГУ» [Электронный ресурс]. — 2021. — Режим доступа URL:

<https://journals.bsu.by/>

Дата доступа: 19.05.2021

1. Сайт «Порталы научных журналов БНТУ» [Электронный ресурс]. — 2021. — Режим доступа URL:

<https://journals.bntu.by/ru/>

Дата доступа: 14.05.2021

1. Официальный сайт БГУИР, раздел «Журналы для опубликования результатов научных исследований» [Электронный ресурс]. — 2021. — Режим доступа URL:

<https://www.bsuir.by/ru/kaf-piks/dokumenty-i-ssylki>

Дата доступа: 17.05.2021

1. Сайт «Национальный правовой интернет портал РБ» [Электронный ресурс]. — 2021. — Режим доступа URL:

<https://pravo.by/gosudarstvo-i-pravo/nauka-i-obrazovanie/vak-informiruet/perechen-nauchnykh-izdaniy-respubliki-belarus-dlya-opublikovaniya-rezultatov-dissertatsionnykh-issle/>

Дата доступа: 23.05.2021

1. Сайт Википедия, раздел «MVC»[Электронный ресурс]. — 2021. — Режим доступа URL: <https://ru.wikipedia.org/wiki/Model-View-Controller>

Дата доступа: 13.05.2021

1. Официальный сайт реляционной базы данных MySQL [Электронный ресурс]. — 2021. — Режим доступа URL: https://www.mysql.com/

Дата доступа: 12.05.2021

1. Официальный сайт программной платформы Node.js [Электронный ресурс]. — 2021. — Режим доступа URL: https://nodejs.org/en/docs/

Дата доступа: 12.05.2021

1. GitHub аккаунт с открытым исходным кодом [Электронный ресурс]. — 2021. — Режим доступа URL: https://github.com/handlebars-lang/handlebars.js#differences-between-handlebarsjs-and-mustache

Дата доступа: 01.06.2021