|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  | | --- | --- | --- | |  |  |  | | МИНОБРНАУКИ РОССИИ | | | | Федеральное государственное бюджетное образовательное учреждение  высшего образования  **«МИРЭА – Российский технологический университет»**  **РТУ МИРЭА** | | |   Институт Информационных технологий | |
|  | |
| Кафедра Математического обеспечения и стандартизации информационных технологий | |
|  | |
|  | |

|  |  |
| --- | --- |
| **ОТЧЕТ ПО ПРАКТИЧЕСКОЙ РАБОТЕ № 5** | |
| **по дисциплине** | |
| «Разработка серверных частей интернет-ресурсов»  **Тема практической работы: «**Сессии. Файлы. БД.» | |
|  | |
| Выполнил студентка группы ИКБО-16-20 | Постнов Н.С. |
| Руководитель практической работы | Волков М.Ю. |

|  |  |  |
| --- | --- | --- |
| Лабораторная работа выполнена | «\_\_»\_\_\_\_\_\_\_202\_\_ г. | *(подпись студента)* |
|  |  |  |
| «Зачтено» | «\_\_»\_\_\_\_\_\_\_202\_\_ г. | *(подпись руководителя)* |

Москва 2022

Оглавление

[Цель работы 3](#_Toc115818394)

[Постановка задачи 3](#_Toc115818395)

[Результат работы 3](#_Toc115818396)

[Ответы на вопросы к практической работе 11](#_Toc115818397)

# Цель работы

Предполагается выполнить апгрейд разрабатываемого в процессе первых 4 практических работ интернет-ресурса механизмами обработки сессий и согласования контента. Предлагается добавить следующую функциональность:

1. Хранение данных сессий в БД Redis.
2. Использование данных для согласования контента на уровне

сервера для формирования контента пользователя с помощью(выбор по варианту). Требуется использовать хотя бы 3 параметра для формирования индивидуального контента, например, логин пользователя, тема(темная, светлая или для людей с цветовой слепотой) и рекомендуемый язык.

a. файлов cookie

b. файлов сессий

1. Загрузка файлов в формате pdf на сервер и хранение их(выбор

по варианту), а также их выдача обратно пользователю по запросу.

* 1. в файловой системе сервера
  2. в реляционной базе данных
  3. в не реляционной базе данных

# Постановка задачи

Предполагается создание стабильной версии интернет-ресурса и сохранение предыдущей функциональности с практических работ 1-4.

В отчете требуется отразить и описать спецификацию реализованной функциональности, провести и отразить тестирование реализованной функциональности.

# Результат работы

В данной практической работе я выбрал хранение данных сессий в БД Redis, хранение данных для согласования контента в cookie, а файлы хранятся в реляционной базе данных.

К данной практической работе мной была создана структура проекта и контейнеров (Рисунки 1-2).

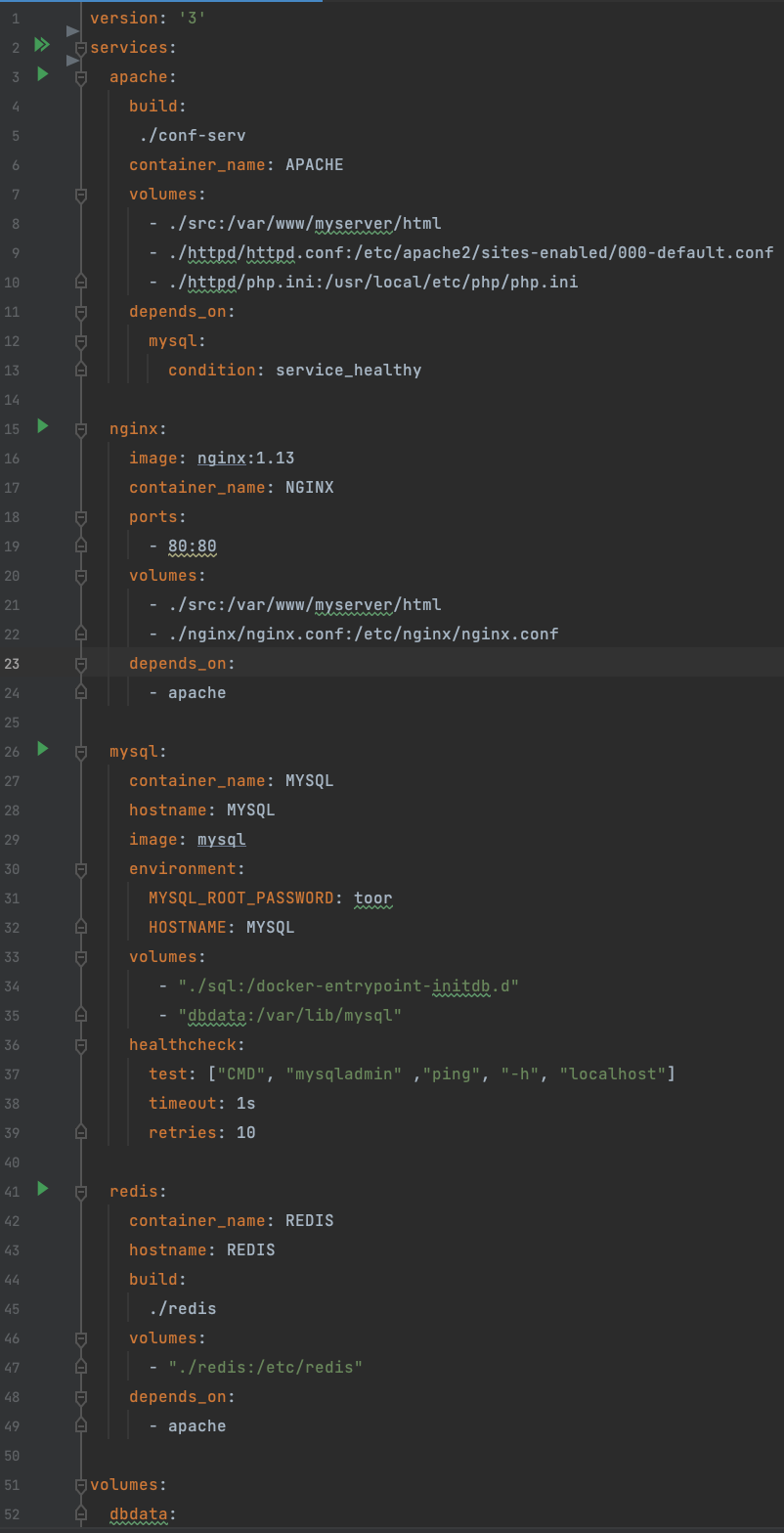


Рисунок 1 – Структура контейнеров

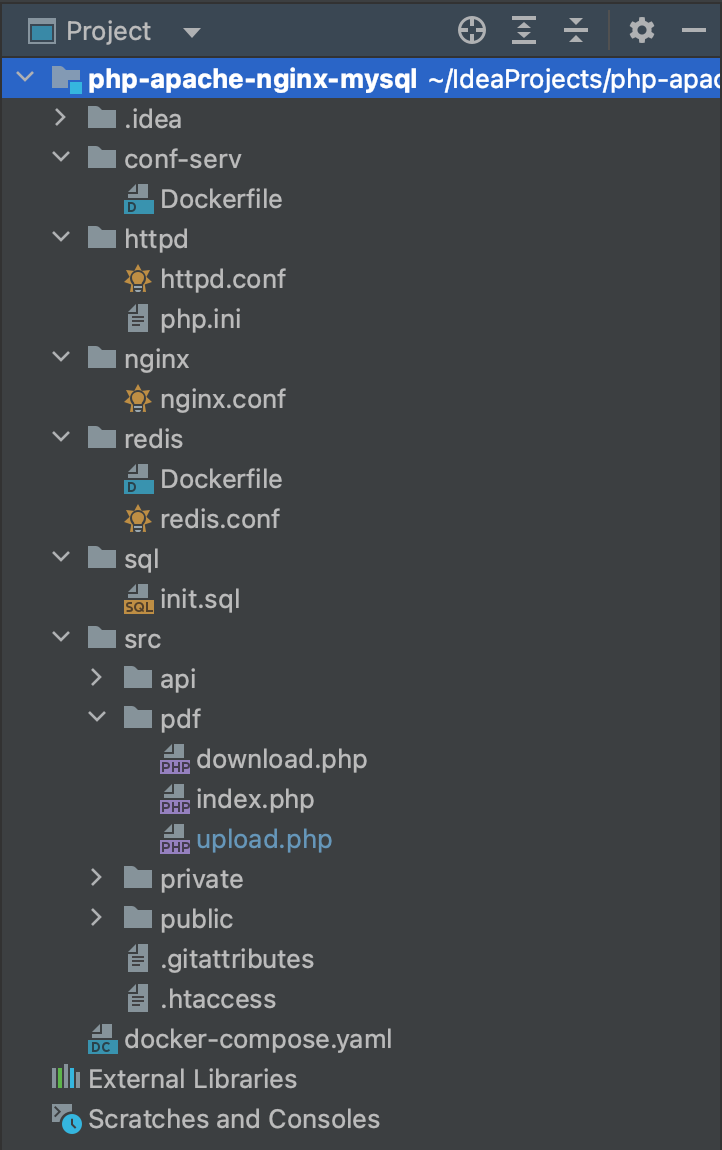


Рисунок 2 – Структура проекта

Запуск проекта осуществляется в несколько шагов. Сначала запускается база данных MYSQL и SQL-скрипт, который создает пользователя и схему с таблицей, в которой будут храниться данные пользователей. MYSQL-скрипт приведен на рисунке 3.

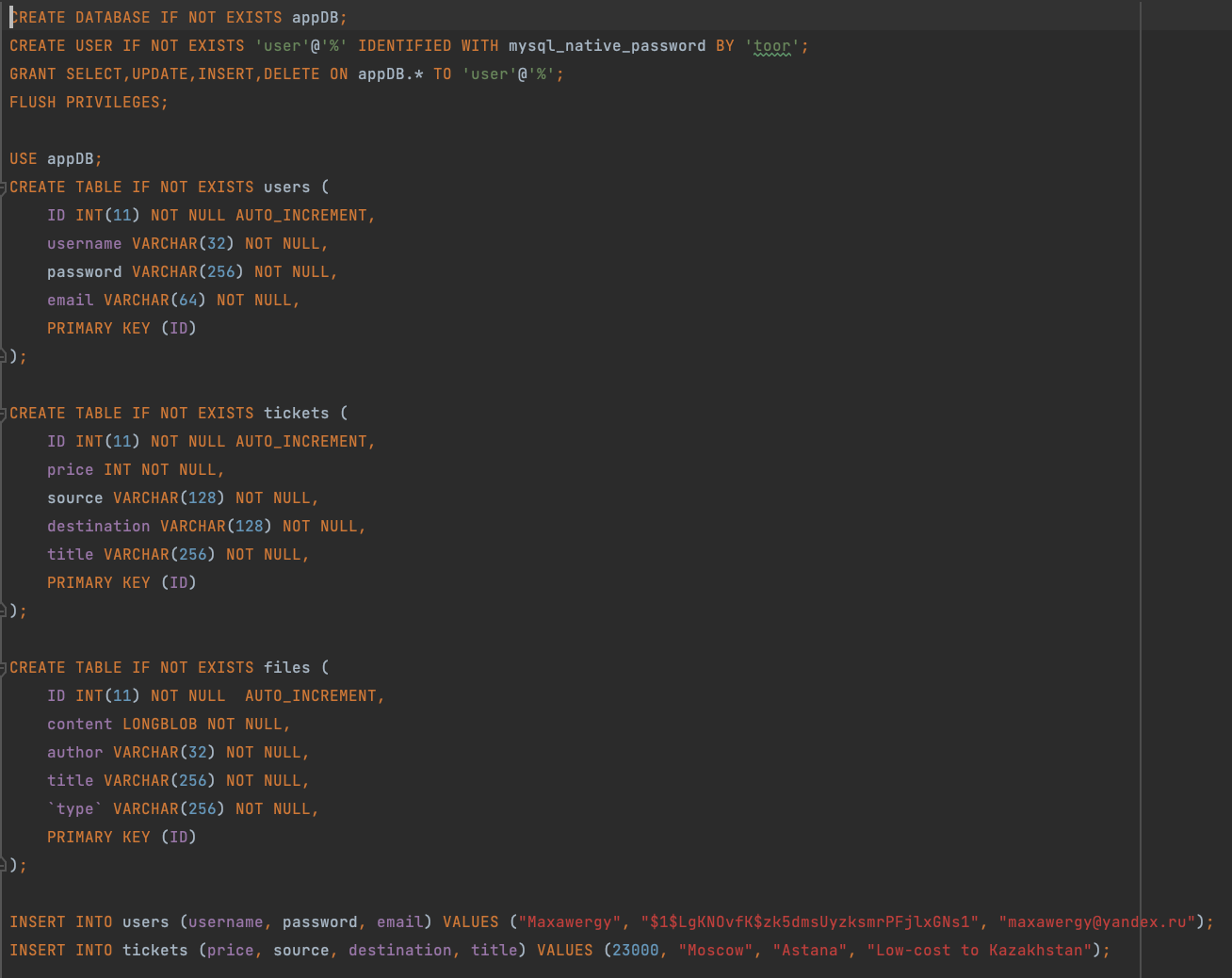


Рисунок 3 – Файл init.sql

Далее запускается образ php-apache на основе 4 практики. Был изменен php.ini.

*Листинг 1 – Файл index.php*

<?php

$ru\_lang = ["Наименование", "Автор", "Скачать", "Добавить файл", "Светлая", "Темная", "Русский", "Английский", "Файл", "Загрузить", "\*Клик\*", "Введите свое имя:", "Язык", "Тема", "Обменник", "Сменить имя"];

$en\_lang = ["Name", "Author", "Download", "Add File", "Light", "Dark", "Russian", "English", "File", "Upload", "\*Click\*", "Enter your name:", "Language", "Theme", "Exchanger", "Change name"];

$dark\_theme = ['table-dark', 'grey', 'bg-dark'];

$light\_theme = ['table-light', 'white', 'bg-light'];

if ($\_COOKIE['lan'] == 'en')

$list = $en\_lang;

else

$list = $ru\_lang;

if ($\_COOKIE['theme'] == 'dark')

$theme = $dark\_theme;

else

$theme = $light\_theme;

function addNavBar($list, $theme){

echo '<nav class="navbar navbar-expand-lg '.$theme[2].'">

<div class="container-fluid">

<a class="navbar-brand" href="index.php">'.$list[14].'</a>

<button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Переключатель навигации">

<span class="navbar-toggler-icon"></span>

</button>

<div class="collapse navbar-collapse" id="navbarSupportedContent">

<ul class="navbar-nav me-auto">

<li class="nav-item">

<a class="nav-link" href="#" onclick="var inputname = prompt(\''.$list[11].'\');document.cookie=\'name=\'+ inputname;" >'.$list[15].'</a>

</li>

<li class="nav-item dropdown">

<a class="nav-link dropdown-toggle" href="#" id="navbarDropdown" role="button" data-bs-toggle="dropdown" aria-expanded="false">

'.$list[12].'

</a>

<ul class="dropdown-menu" aria-labelledby="navbarDropdown">

<li><a class="dropdown-item" href="index.php" onclick="document.cookie=\'lan=ru\';">'.$list[6].'</a></li>

<li><hr class="dropdown-divider"></li>

<li><a class="dropdown-item" href="index.php" onclick="document.cookie=\'lan=en\';">'.$list[7].'</a></li>

</ul>

</li>

<li class="nav-item dropdown">

<a class="nav-link dropdown-toggle" href="#" id="navbarDropdown" role="button" data-bs-toggle="dropdown" aria-expanded="false">

'.$list[13].'

</a>

<ul class="dropdown-menu" aria-labelledby="navbarDropdown">

<li><a class="dropdown-item" href="index.php" onclick="document.cookie=\'theme=light\';">'.$list[4].'</a></li>

<li><hr class="dropdown-divider"></li>

<li><a class="dropdown-item" href="index.php" onclick="document.cookie=\'theme=dark\';">'.$list[5].'</a></li>

</ul>

</li>

</ul>

</div>

</div>

</nav>';

}

echo '<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css" rel="stylesheet">

<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.min.js"></script>

<script src="https://cdn.jsdelivr.net/npm/@popperjs/core@2.10.2/dist/umd/popper.min.js"></script>

<script src="https://code.jquery.com/jquery-3.5.1.slim.min.js"></script>';

echo "<body style='background-color:$theme[1]'>";

addNavBar($list, $theme);

if (!isset($\_COOKIE['name']))

getName($list[11]);

if (!isset($\_COOKIE['theme']) or ( $\_COOKIE['theme'] != 'light' and $\_COOKIE['theme'] != 'dark' ) )

setcookie("theme", "light");

if (!isset($\_COOKIE['lan']) or ( $\_COOKIE['lan'] != 'ru' and $\_COOKIE['lan'] != 'en' ) )

setcookie("lan", "ru");

function getName($string){

echo '<script type="text/javascript"> ';

echo 'var inputname = prompt("'.$string.'", "");';

echo 'document.cookie = "name=" + inputname;';

echo '</script>';

}

session\_start();

$con = new mysqli("MYSQL", "user", "toor", "appDB");

echo '<table class="table table-hover '.$theme[0].'">

<tr><th>'.$list[0].'</th><th>'.$list[1].'</th><th>'.$list[2].'</th></tr>';

$result = $con->query("SELECT \* FROM files");

foreach ($result as $row){

echo "<tr><td>{$row['title']}.{$row['type']}</td><td>{$row['author']}</td><td><a href='download.php?id={$row['ID']}'>{$list[10]}</a></td></tr>";

}

echo '</table><br>';

echo '<form enctype="multipart/form-data" action="upload.php" method="POST">

<div class="form-group">

<input type="text" required name="title" id="title" placeholder="'.$list[0].'"/>

'.$list[8].': <input name="uploadedFile" type="file" />

<input required type="submit" value="'.$list[9].'" />

</form>

</div>';

?>

*Листинг 2 – Файл download.php*

<?php

if (!isset($\_GET['id'])){

header("Location: index.php");

exit;

}

$con = new mysqli("MYSQL", "user", "toor", "appDB");

$query\_result = $con->query("SELECT \* FROM files WHERE ID='".$\_GET['id']."'");

if (mysqli\_num\_rows($query\_result) == 0)

die("File doesnt exists");

$row = mysqli\_fetch\_object($query\_result);

header("Content-type: application/" . $row->type);

header('Content-Disposition: attachment; filename="'.$row->title.'.'.$row->type.'"');

echo $row->content;

?>

*Листинг 3 – Файл upload.php*

*<?php*

*$fileName = $\_FILES['uploadedFile']['name'];*

*$fileNameCmps = explode(".", $fileName);*

*$fileExtension = strtolower(end($fileNameCmps));*

*$blob = addslashes(file\_get\_contents($\_FILES['uploadedFile']['tmp\_name']));*

*$con = new mysqli("MYSQL", "user", "toor", "appDB");*

*$con->query("INSERT INTO files (content, author, title, `type`) VALUES ('". $blob . "','" . $\_COOKIE["name"] . "','" . $\_POST["title"] . "','" . $fileExtension . "');");*

*$con->close();*

*header("Location: index.php");*

*?>*

Теперь запустим сервер и проверим работоспособность сайта (Рисунки 4-8).

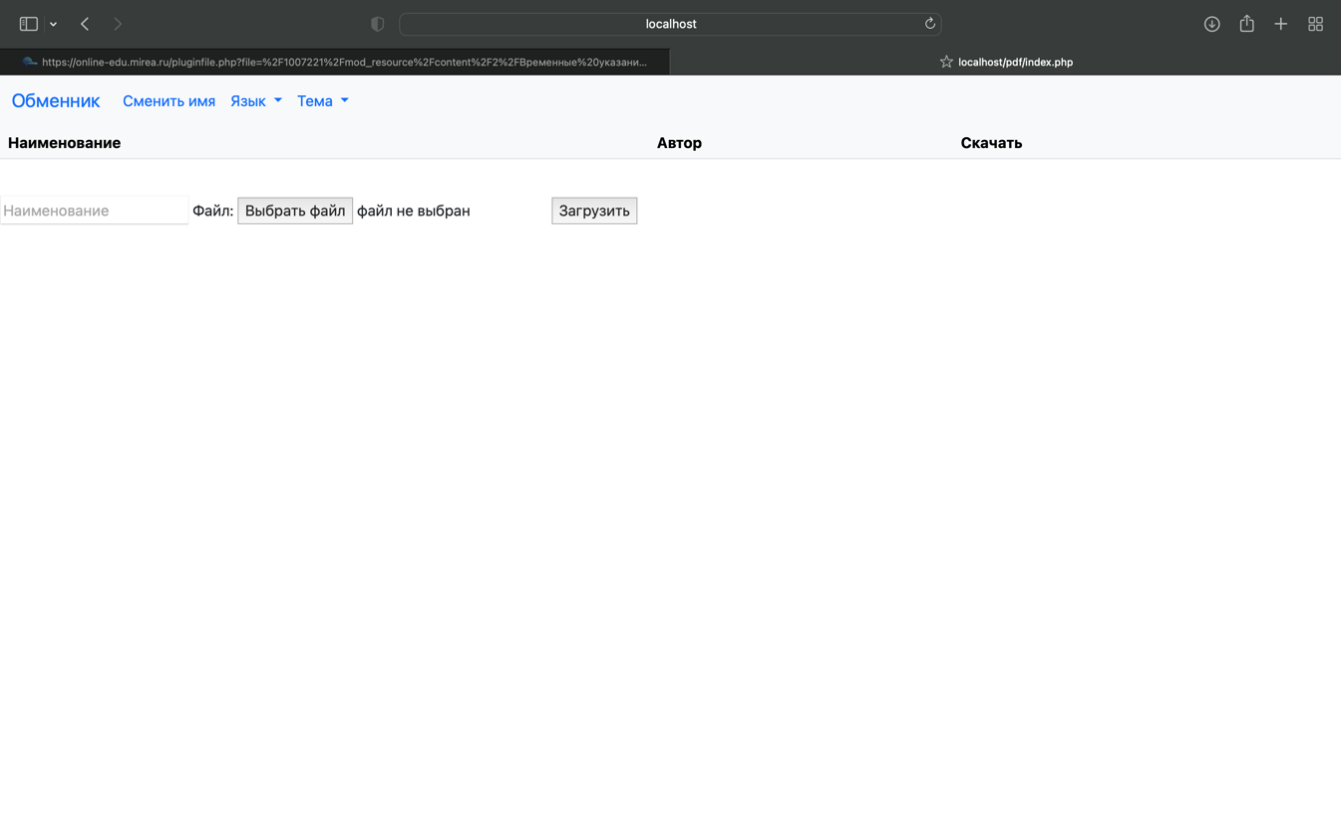


Рисунок 4 – Русский язык и светлая тема

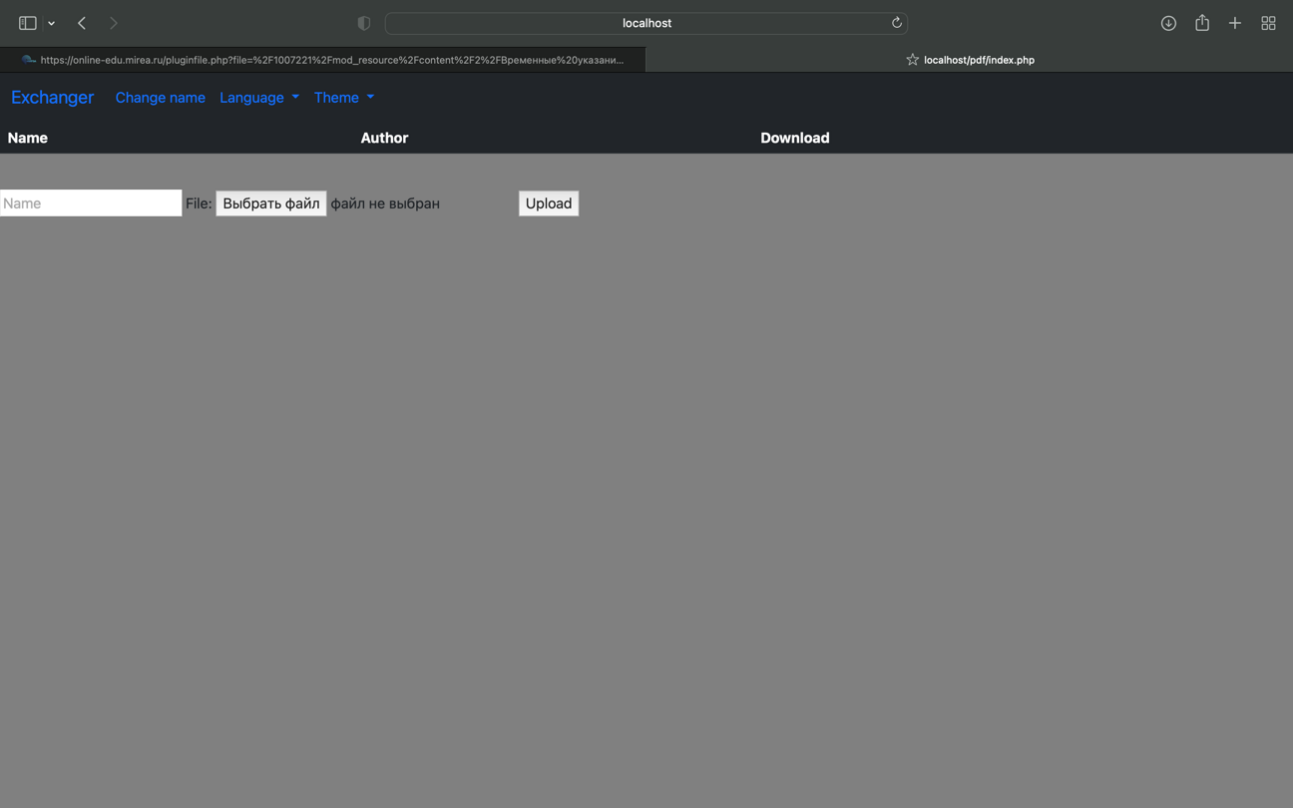


Рисунок 5 – Английский язык и темная тема

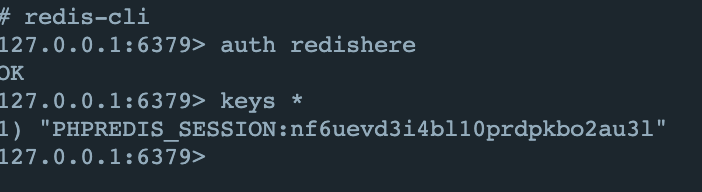


Рисунок 6 – Сохраненная сессия в REDIS

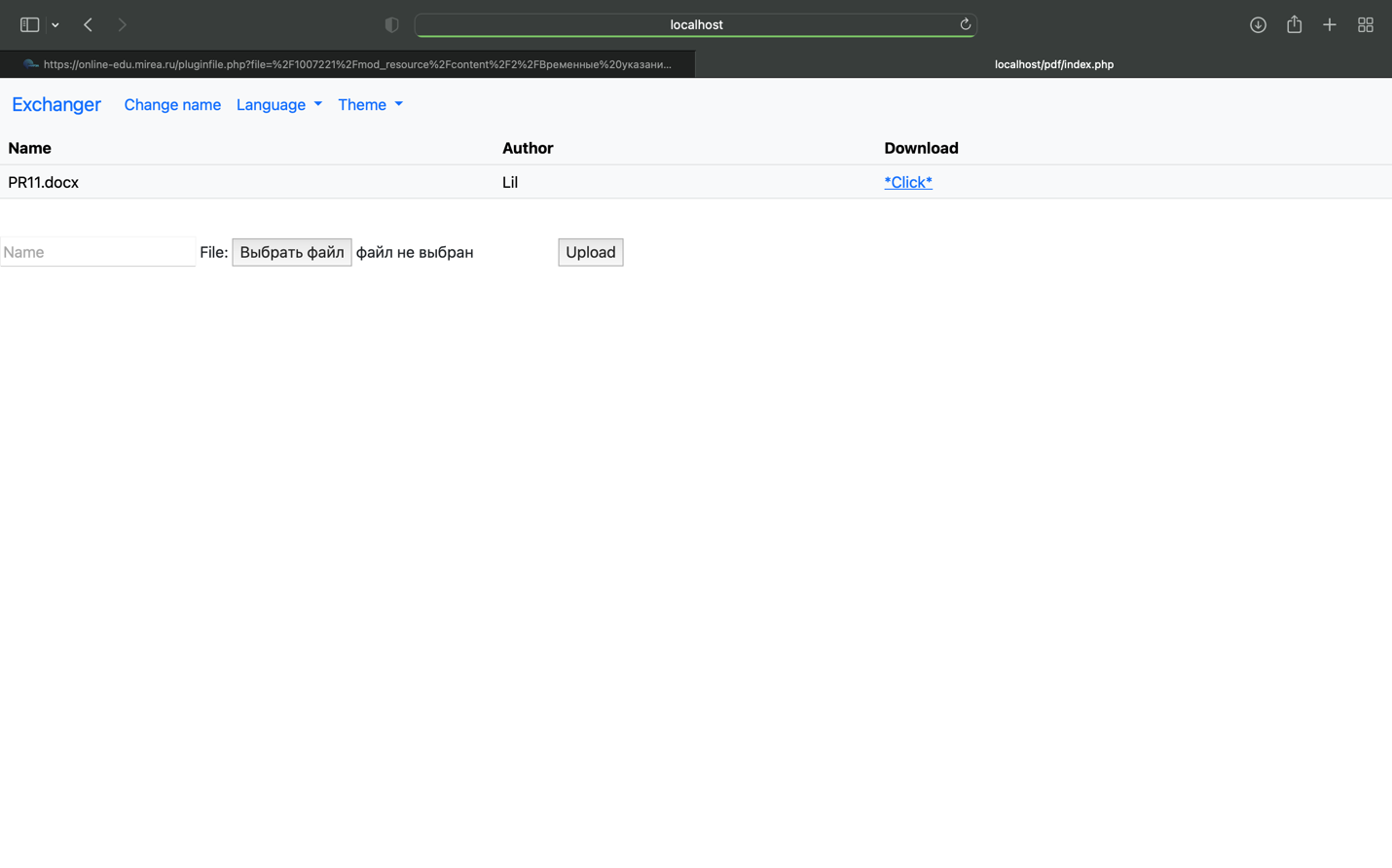


Рисунок 7 – Загрузка файла

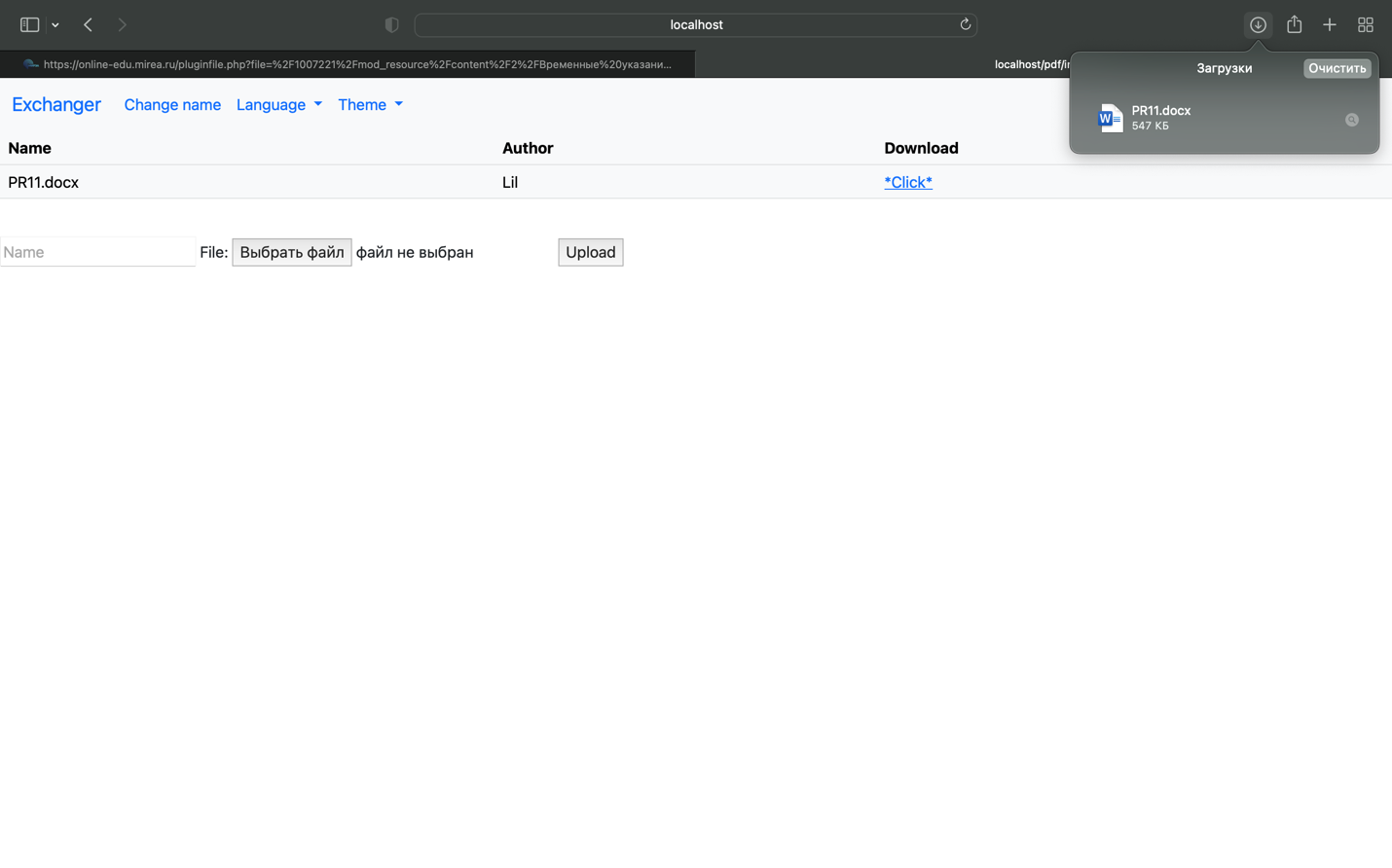


Рисунок 8 – Выкачка файла

**Вывод**

В результате выполнения работы была создана сложная серверная конфигурация, состоящая из связки apache+nginx+php+MySQL+redis, а также обменник файлов. Получены практические навыки по настройке конфигурации php (php.ini).

# Ответы на вопросы к практической работе

**1. Что такое сессия в рамках веб-разработки?**

Сессия, это абстракция, созданная для удобной работы с индивидуальными пользователями. Она используется для идентификации пользователей и позволяет отличать их друг от друга. Например, аутентификация на сайтах построена поверх механизма сессии. **2. Что такое cookie в рамках веб-разработки?**

Ку́ки — небольшой фрагмент данных, отправленный веб-сервером и хранимый на компьютере пользователя. Веб-клиент всякий раз при попытке открыть страницу соответствующего сайта пересылает этот фрагмент данных веб-серверу в составе HTTP-запроса. **3. Опишите механизм использования cookies.**

Файлы cookie используются для отслеживания привычек пользователей в Интернете. Это также можно сделать в некоторой степени, используя IP-адрес компьютера, запрашивающего страницу, или поле referer заголовка HTTP-запроса, но cookie-файлы позволяют повысить точность. Это можно продемонстрировать, если пользователь запрашивает страницу сайта, но запрос не содержит cookie, сервер предполагает, что это первая страница, которую посетил пользователь. Таким образом, сервер создаёт уникальный идентификатор (обычно последовательность случайных букв и цифр) и отправляет его в виде файла cookie в браузер вместе с запрашиваемой страницей. **4. Опишите простой пример работы сессий в PHP.**

С этого момента cookie будет автоматически отправляться браузером на сервер каждый раз, когда запрашивается новая страница с сайта. Сервер не только отправляет страницу как обычно, но также сохраняет URL запрошенной страницы, дату / время запроса и файл cookie в файле журнала. **5. Опишите способы защиты сессии пользователя.**

Сессии являются простым способом хранения информации для отдельных пользователей с уникальным идентификатором сессии. Это может использоваться для сохранения состояния между запросами страниц. Идентификаторы сессий обычно отправляются браузеру через сессионный cookie и используются для получения имеющихся данных сессии. Отсутствие идентификатора сессии или сессионного cookie сообщает PHP о том, что необходимо создать новую сессию и сгенерировать новый идентификатор сессии. **6. Верно ли, что можно хранить данные сессии в БД?**

Да

**7. Опишите жизненный цикл сессии.**

В PHP механизм управления сессиями предполагает, что только владелец знает идентификатор сессии и для получения доступа к данным не требует дополнительной авторизации. По аналогии можно представить ситуацию, при которой банк выдаёт деньги человеку, знающему номер существующего счета. Для безопасной работы с сессиями необходим дополнительный механизм идентификации, не причиняющий неудобств легальным пользователям и являющийся существенной преградой для злоумышленников.

**8. Верно ли, что можно убрать механизм обработки сессий?**

Alive - при первом запуске. Not Alive - Если повторить запрос к сервлету менее чем через 10 секунд, вывод будет идентично

повторен. Если же запрос повторить более чем через десять секунд, сессия будет

автоматически уничтожена.

**9. Опишите примеры настройки сессии во время выполнения.**

Настройку сессий можно производить с помощью файла php.ini

**10.Опишите директивы конфигурации файловой системы и потоков в PHP.**

Cписок директив включает встроенные директивы php.ini, которые вы можете использовать для настройки PHP. Директивы, которые обрабатываются модулями, перечислены и подробно описаны на страницах документаций соответствующих модулей. Потоки были введены как инструмент для работы с файлами, сетевого обмена, сжатия данных и выполнения других операций с помощью одного общего набора функций. Выражаясь простыми понятиями, поток (stream) - это ресурс (resource), который ведёт себя как источник непрерывной последовательности данных. Это означает, что из потока можно последовательно читать данные, равно как и записывать в него. В PHP нет потоков выполнения: его движок и код обходятся без потоков для распараллеливания внутренней работы. **11.Какой тип ресурса использует файловая система. Опишите данный тип.**

Файловая система использует потоки (streams) в качестве собственного типа ресурсов. **12.Как открыть и закрыть файл с помощью PHP.**

Самый лучший метод для открытия файлов является функция fopen(). Функция fread(), чтение открытого файла. Функция fclose() требует имя файла (или переменной, содержащей имя файла), который нужно закрыть. Для записи файла применяется функция fwrite(), которая записывает в файл строку. **13.Как производится чтение и запись файлов в PHP.**

Для построчного чтения используется функция fgets(), которая получает дескриптор файла и возвращает одну считанную строку. Также можно провести поблочное считывание, то есть считывать определенное количество байт из файла с помощью функции fread().

**14.Опишите как считать только часть файла, как считывать файл последовательно и считать весь файл целиком.**

Создание файла осуществляется с помощью fopen(). Функция unlink() используется для удаления файла. **15.Как производится создание и удаление файлов с помощью PHP.**

Touch – создать, rm - удалить

**16.С помощью каких функций и какую информацию о файле можно получить с помощью PHP?**

• is\_readable — Определяет существование файла и доступен ли он для чтения

• is\_writable — Определяет, доступен ли файл для записи

• stat — Возвращает информацию о файле

• chgrp — Изменяет группу файла

• chmod — Изменяет режим доступа к файлу

• chown — Изменяет владельца файла **17.Что такое DOM?**

DOM — это независящий от платформы и языка программный интерфейс, позволяющий программам и скриптам получить доступ к содержимому HTML-, XHTML- и XML-документов, а также изменять содержимое, структуру и оформление таких документов. Модель DOM не накладывает ограничений на структуру документа. **18.Как создать документ и работать с ним с помощью модуля DOM?**

DOMDocument::createAttribute

Эта функция создаёт новый объект класса DOMAttr. Этот узел не будет отображаться в документе до тех пор, пока он не будет вставлен, например, функцией DOMNode::appendChild().

**19.Что такое JSON?**

JSON — текстовый формат обмена данными, основанный на JavaScript. Как и многие другие текстовые форматы, JSON легко читается людьми. Формат JSON был разработан Дугласом Крокфордом. **20.Как декодировать строку JSON и вернуть JSON-представление данных?**

Json\_encode, json\_decode – закодировать, раскодировать соответственно **21.Как проанализировать и выявить ошибки при кодировании и декодировании JSON?**

Json\_last\_error - Возвращает последнюю ошибку (если она есть), произошедшую во время последнего кодирования/декодирования JSON если при вызове не использовался флаг JSON\_THROW\_ON\_ERROR. **22.Опишите создание, сохранение, парсинг XML-документа c помощью PHP.**

## Для этого используется сторонняя библиотека SimpleXML ¶

## Пример #2 Получение части документа <plot>

## <?php

## include 'example.php';

## $movies = new SimpleXMLElement($xmlstr);

## echo $movies->movie[0]->plot;

## ?> **23.Что такое драйвер в рамках взаимодействия с СУБД?**

Драйвер базы данных - это просто прослойка между Вашим кодом и базой данных. Она нужна по одной простой причине - все движки баз данных достаточно разные и пришлось бы для каждой базы делать свой набор классов и свою документацию. **24.Опишите добавление записи в рамках использования модуля mysqli для взаимодействия с БД MYSQL.**

Query(‘INSERT INTO TABLE (VALUE1) VALUES (“‘ . $value. ‘);” **25.Что такое постоянное соединение, опишите проблемы данного подхода и его решение в модуле mysqli.**

Connection.close() **26.Опишите основные особенности БД MongoDB.**

Это кроссплатформенная документоориентированная база данных NoSQL с открытым исходным кодом. Она не требует описания схемы таблиц, как в реляционных БД. Данные хранятся в виде коллекций и документов. Между коллекциями нет сложных соединений типа JOIN, как между таблицами реляционных БД. **27.Опишите процесс добавления новой записи в СУБД MongoDB с помощью соответствующего драйвера.**

MongoDB\Driver\Query::\_\_construct — Создаёт новый запрос **28.Опишите процесс получения и обработки записей с помощью драйвера MongoDB.**

Как MongoDB\Driver\Manager::executeCommand(), так MongoDB\Driver\Manager::executeQuery(), возвращают свои результаты в виде объекта MongoDB\Driver\Cursor.

Поскольку MongoDB\Driver\Cursor реализует интерфейс Traversable, вы можете итерировать по набору результата с помощью foreach. **29.Опишите получение записей в рамках использования модуля mysqli для взаимодействия с БД MYSQL.**

При выполнении команды SELECT метод query() объекта mysqli возвращает набор полученных строк, который мы можем перебрать с помощью цикла. **30.Опишите поиск записей, подсчет и ограничение выборки с помощью драйвера MongoDB.**

Вместо SELECT (SQL) используется FIND.

**Ссылка на удаленный репозиторий проекта**

https://github.com/NikitaPostnov101/RSCHIR\_5

# Список использованной литературы

1. Официальная документация докера [Электронный ресурс] – URL: <https://docs.docker.com> (дата обращения 03.11.2022)
2. Установка и настройка PHP [Электронный ресурс] – URL: <https://www.php.net/manual/ru/install.php> (дата обращения 03.11.2022)
3. Docker Compose [Электронный ресурс] – URL: <https://habr.com/ru/company/ruvds/blog/450312> (дата обращения 03.11.2022)
4. Docker Hub [Электронный ресурс] – URL: <https://hub.docker.com> (дата обращения 03.11.2022)
5. PHP.NET [Электронный ресурс] – URL: <https://www.php.net> (дата обращения 08.11.2022)

**ОТВЕТЫ НА ВОПРОСЫ**

**Что такое сессия в рамках веб-разработки?**

Сессия, это абстракция, созданная для удобной работы с индивидуальными пользователями. Она используется для идентификации пользователей и позволяет отличать их друг от друга. Например, аутентификация на сайтах построена поверх механизма сессии

**Что такое cookie в рамках веб-разработки?**

HTTP cookie (web cookie, куки браузера) - это небольшой фрагмент данных, который сервер отправляет браузеру пользователя.

**Верно ли, что можно хранить данные сессии в БД?**

Верно

**Как открыть и закрыть файл с помощью PHP.**

Самый лучший метод для открытия файлов является функция fopen. Функция fopen, даст вам больше вариантов, чем функция readfile()

Функция fread(), чтение открытого файла.

Первый параметр fread() содержит имя файла для чтения, а второй параметр указывает максимальное число байтов для чтения.

Следующий PHP код считывает "webdictionary.txt" файл до конца:

fread($myfile,filesize("webdictionary.txt"));

Функция fclose() используется для закрытия открытого файла.

**Как производится чтение и запись файлов в PHP.**

рочитать строку из открытого файла можно с помощью функции **fread**:

string **fread** ( int **file**, int **length** )

Эта функция возвращает строку длиной **length** символов из файла с дескриптором **file**.

**Что такое DOM?**

**DOM** – это объектное представление исходного HTML-кода документа. Процесс формирования DOM происходит так: браузер получает HTML-код, парсит его и строит DOM.

**JSON** (англ. JavaScript Object Notation) — текстовый формат обмена данными, основанный на JavaScript.