|  |  |  |
| --- | --- | --- |
|  |  |  |
| МИНОБРНАУКИ РОССИИ | | |
| Федеральное государственное бюджетное образовательное учреждение  высшего образования  **«МИРЭА**– **Российский технологический университет»**  **РТУ МИРЭА** | | |

Институт Информационных технологий

Кафедра Инструментального и прикладного программного обеспечения

**ПРАКТИЧЕСКАЯ РАБОТА №6**

по дисциплине «Разработка серверных частей интернет-ресурсов»

**Тема практической работы:** Сторонние модули PHP

**Студент группы** ИКБО-24-20 Постнов Никита Сергеевич

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(подпись студента)

**Руководитель практической работы** преподаватель Волков М.Ю.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(подпись руководителя)

Работа представлена «\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_ 2022 г.

Допущен к работе «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_ 2022 г.

Москва 2022

**Оглавление**

[Цель работы 3](#_Toc113143392)

[Ход работы 3](#_Toc113143393)

[Вывод 7](#_Toc113143394)

[Ответы на вопросы к практической работе 8](#_Toc113143395)

[Ссылка на удалённый репозиторий проекта 15](#_Toc113143396)

[Список использованной литературы 15](#_Toc113143397)

# Цель работы

Получить навыки работы со сторонними модулями.

# Задание

Для выполнения каждой части задания предполагается использование различных расширений и сторонних модулей PHP.

Предполагается выполнить модификацию разрабатываемого в процессе первых 5 практических работ интернет-ресурса. Для этого выполнить следующие действия:

1. Сгенерировать набор фикстур с помощью выбранного стороннего модуля. Каждая фикстура должна содержать не менее 5 полей. Всего фикстур должно быть не менее 50.

2. На основе этих фикстур построить минимум 3 графика различного типа с помощью сторонних модулей PHP.

3. График превратить в картинку и добавить на него ваш личный полупрозрачный водяной знак с помощью библиотеки GD(строго рекомендуется) или другой библиотеки для работы с изображениями в PHP.

4. Отобразить полученные графики на странице статистики.

Предполагается создание стабильной версии интернет-ресурса и сохранение предыдущей функциональности с практических работ 1-5.

В отчете требуется отразить и описать спецификацию реализованной функциональности, провести и отразить тестирование реализованной функциональности. А также описать инструменты для установки и настройки сторонних модулей и процесс их использования для установки инструментов, используемых в данной практической работе.

# Ход работы

В качестве библиотек для разработки установим следующие зависимости:

* Faker – генерация тестовых данных
* jpGraph – генерация графиков и изображений по графикам
* GD – наложение водяного знака

В соответствии с этим необходимо в Dockerfile внести зависимости для установки вспомогательного модуля – composer и GD(Рисунок 1) и отдельный файл для установки зависимости (Рисунок 2).

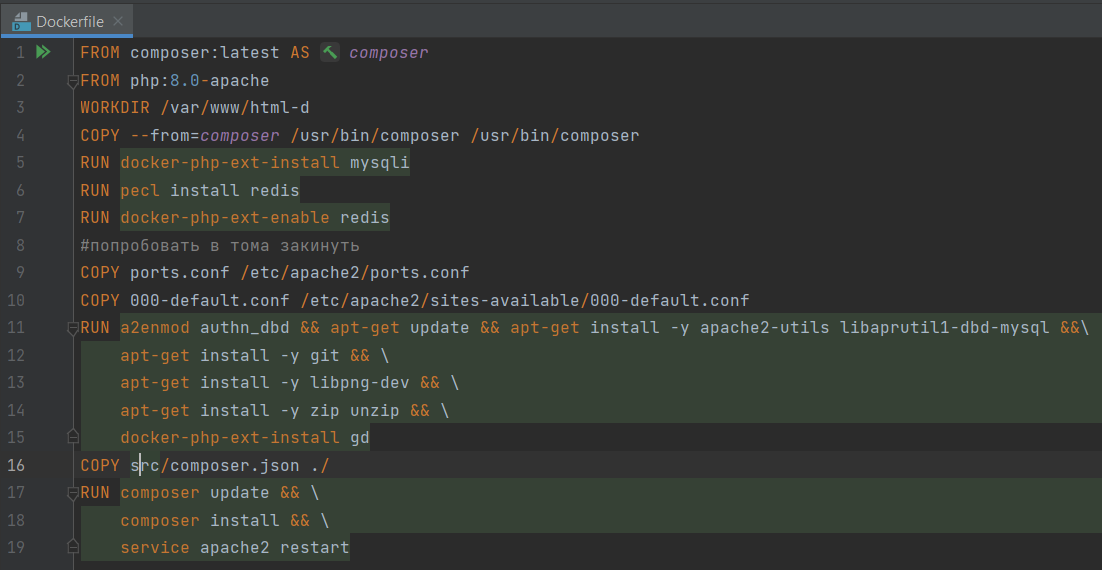


Рисунок 1 – Dockerfile

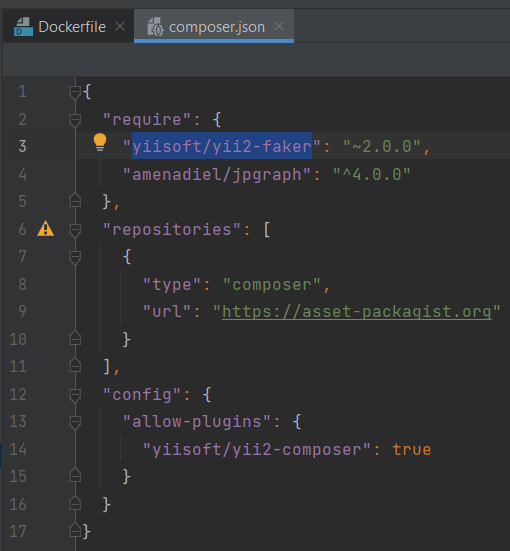
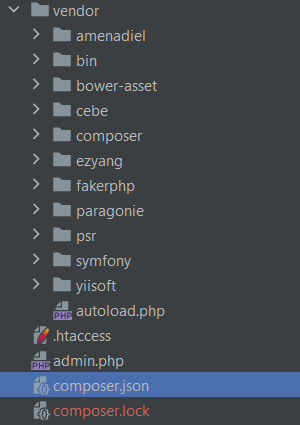


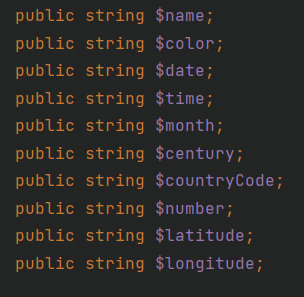
Рисунок 2 – composer.json

Далее нужно при запуске в контейнере запустить команду composer install, т.к при сборке не все директории смонтированы. Другой путь – прописать скрипт запуска и включить его в docker-compose.

В результате будут скачены зависимости (Рисунок 3).

  
Рисунок 3 – Скачанные файлы

Для генерации используем следующую структуру (Рисунок 4).

  
Рисунок 4 – Структура текстовых данных

Далее для каждого типа диаграмм создается отдельный класс. В результате получим следующую функциональность (Рисунок 5 – 6).

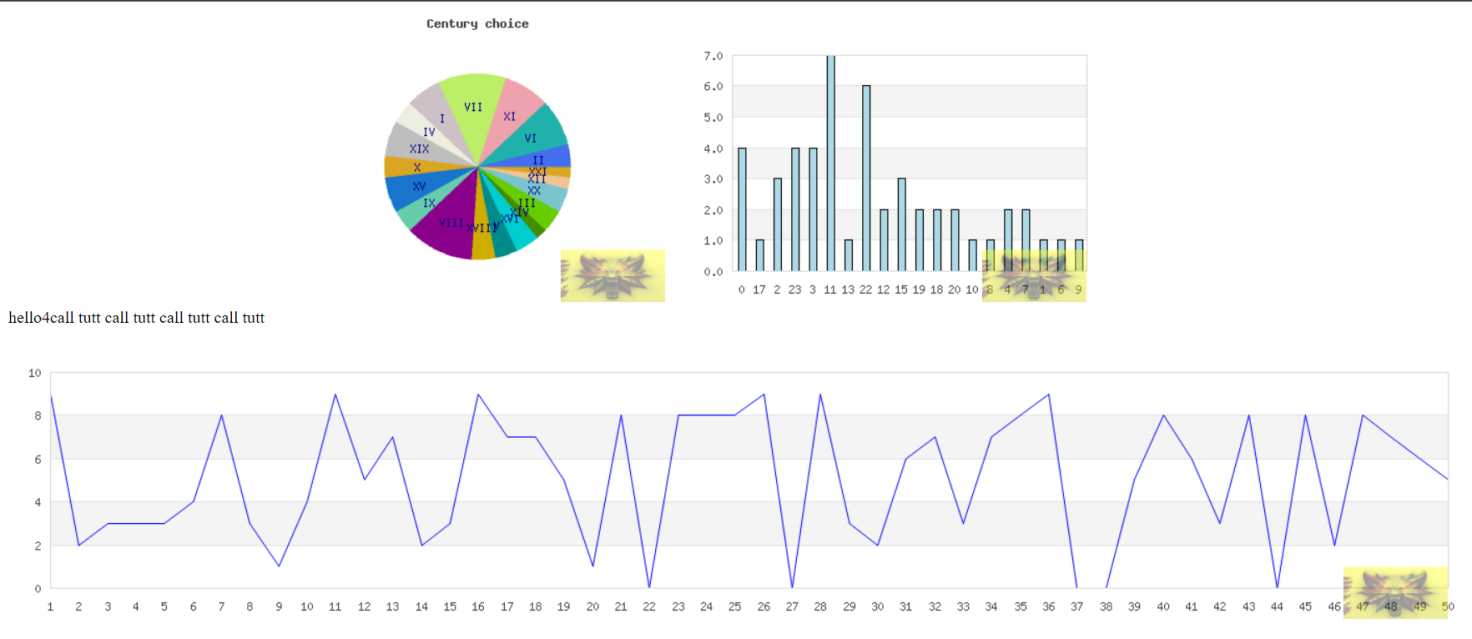


Рисунок 5 – Тестирование функциональности

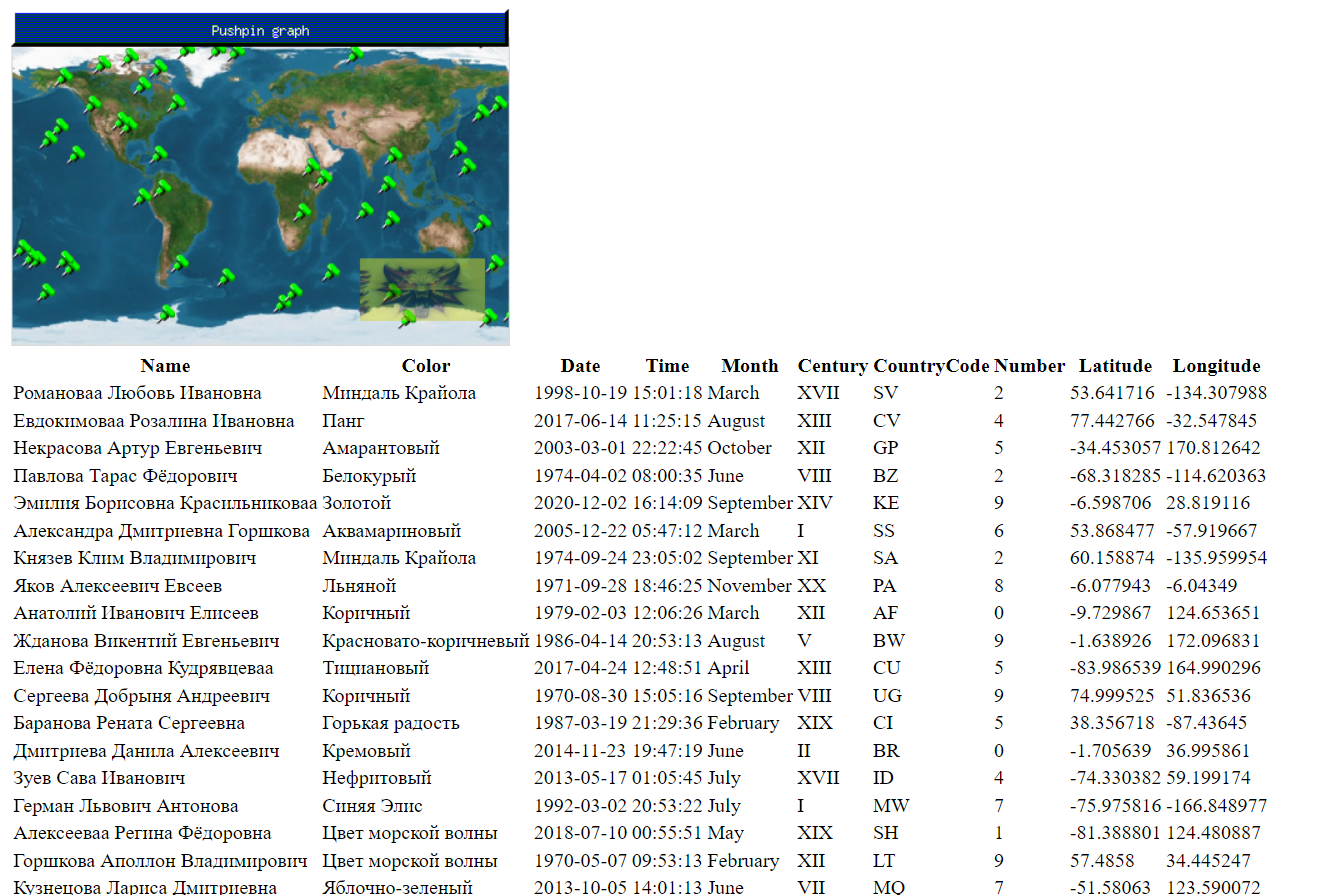


Рисунок 6 – Тестирование функциональности

Пропишем спецификацию (http://localhost:8006/faker/statistic\_page.php). Представим ответ в виде таблицы (Таблица 1)

Таблица 1 – Спецификация

|  |  |  |  |
| --- | --- | --- | --- |
| CRUD-операция | HTTP-метод | URL: | Входные Данные/Ответ от сервера |
| Read | GET | создать нового файла  http://localhost:8006/faker/statistic\_page.php | Входные данные:  Выходные данные: html-ответ о загрузке  Успех: 201  Неудача:500 |

# Вывод

В ходе выполнения данной практической работы были получены навыки работы со сторонними модулями (бибилиотеками) PHP. Использовалась следующая сборка: Faker для тестовых данных, jpGraphдля построения графиков и графическая библиотека GD для генерации водяного знака.

# Ответы на вопросы к практической работе

1. Что такое Composer и для чего он используется?

Современным решением для управления пакетами для проектов на языке программирования PHP является Composer. Нужно понимать, что Composer это более продвинутая система пакетного менеджмента на уровне одного проекта. С помощью данного средства невозможно установить глобальный пакет, только пакет для определенного проекта. Хотя существуют некоторые исключения для модулей анализа и статистики.

1. Опишите работу с календарем в PHP: опишите преобразование дат из одного календаря в другой.

Модуль календарь предоставляет ряд функций для упрощения перевода дат из одного формата в другой.

Для конвертации между календарными форматами сначала нужно перевести дату в Юлианское исчисление, а затем в любой другой календарный формат на ваш выбор.

1. Опишите понятие серверное время и процесс форматирования времени для разных часовых поясов.

Серверное время - это время региона (часового пояса), где находится сервер.

gmdate(string $format, ?int $timestamp = null): string

Эта функция идентична функции date() за исключением того, что возвращает время по Гринвичу (GMT).

1. Опишите способ получения времени заката и рассвета с использованием языка программирования PHP.

date.sunrise\_zenith - Угол, под которым солнце светит во время восхода;

date.sunset\_zenith - Угол, под которым солнце светит во время заката;

1. Опишите константы, используемые в модуле “Время и дата”.

[https://www.php.net/manual/ru/class.datetimeinterface.php#datetimeinterface.constants.types](https://www.php.net/manual/ru/class.datetimeinterface.php" \l "datetimeinterface.constants.types)

Следует упомянуть основные поддерживаемые календари, определенные в виде предопределенных констант: CAL\_GREGORIAN=0, CAL\_JULIAN=1, CAL\_JEWISH=2 и CAL\_FRENCH=3.

0 or CAL\_GREGORIAN - Gregorian Calendar

1 or CAL\_JULIAN - Julian Calendar

2 or CAL\_JEWISH - Jewish Calendar

3 or CAL\_FRENCH - French Revolutionary Calendar

1. Приведите принципы арифметики даты и времени.

Также данный класс реализует базовую временную арифмети-ку: сложение и вычитание дат, - с помощью функций add и sub, а вход которым подаются объекты типа DateInterval, обозначающие временной интервал.

1. Чем отличается фреймворк от библиотеки? Приведите пример.

«Фреймворк» отличается от понятия библиотеки тем, что библиотека может быть использована в программном продукте просто как набор подпрограмм близкой функциональности, не влияя на архитектуру программного продукта и не накладывая на неё никаких ограничений.

Очень часто фреймворки и библиотеки характеризуют как "упрямые" (opinionated) или "не упрямые" (un-opinionated). Эта оценка основывается на уровне свободы, который имеет разработчик при структурировании кода.

Примером очень "упрямого" фреймворка может служить Angular, а вот Vue.js считается очень свободным.

1. Опишите возможные форматы даты и времени и примеры их использования.

<https://www.php.net/manual/en/datetime.formats.time.php>

<https://www.php.net/manual/en/datetime.formats.date.php>

Класс DateTimeZone представляет собой часовой пояс и различные информационные методы о часовых поясах: информация о расположении часового пояса, имя часового пояса, смещение часового пояса от UTC (GMT), все переходы для часового пояса и другая сопутствующая информация.

1. Опишите работу с датой и временем в подходе ООП.

Используются объекты, содержащие всю информацию о конкретном моменте времени (год, месяц, число, часовой пояс, час, минута, секунда) представленные в виде полей, и методы их обработки.

1. Опишите использование класса DateTime.

Для DateTimeImmutablecуществует изменяемый аналог класс DateTime

1. Опишите использование класса DateTimeImmutable.

Класс DateTimeImmutable является неизменяемым представлением даты и времени.. Создание экземпляраданного класса возможно в процедурном (используется функция date\_create\_immutable) и объектно-ориентированном стиле. По умолчанию создается объект, содержащий текущее время и в текущем часовом поясе, но данные моменты настраиваются. Также возможно создание нового объекта типа DateTimeImmutable на основе разбора даты некоторого формата с помощью функциональности date\_create\_immutable\_from\_format. Важной особенностью данного класса является метод modify, который изменяет временную метку, на который указывал изначальный объект, и возвращает новый объект, оставляя старый без изменения.

1. Опишите использование класса DateTimeZone.

Класс DateTimeZone представляет собой часовой пояс и различные информационные методы о часовых поясах: информация о расположении часового пояса, имя часового пояса, смещение часового пояса от UTC (GMT), все переходы для часового пояса и другая сопутствующая информация.

1. Какие библиотеки используются для работы с изображениями в PHP.

Язык программирования PHP предоставляет множество стандартных расширений для обработки изображений, самые известные из них упомянуты в официальной документации: exif (изменяемая информация об изображении), gd (мощный инструмент обработки изображений), Gmagick, ImageMagick и другие. Самой известной и широко используемой является библиотека GD, на основе ее возможностей реализовано множество пакетов для продвинутой и упрощенной обработки и генерации изображений. Например, ее функциональ-ность использует вышеописанная библиотека для построения графиков JpGraph.

1. Опишите основные возможности библиотеки GD.

PHP не ограничен созданием только HTML страничек. Он так же позволяет создавать и работать с файлами изображений в различных форматах включая GIF, PNG, JPEG, WBMP, и XPM. Что более удобно, PHP позволяет выводить изображение в потоке непосредственно в браузер. Для этого необходимо скомпилировать PHP с графической библиотекой GD, содержащей функции для работы с изображениями.

1. Как использовать Composer для подключения библиотек к проекту?

Для того, чтобы использовать модули, установленные с помощью composer нужно подключить к скрипту файл vendor/autoload.php.

1. Что такое PEAR? В чём разница работы PEAR и Composer?

PEAR (PHP Extension and Application Repository) - эторепозиторийпри-ложенийимодулей PHP. Включает обширный репозиторий PHP-пакетов, си-стему для управления пакетами (многими разработчиками признана устарев-шей, но до сих пор используется), стандарт написания кода и субрепозитории PECL, App, Gtk. Данная функциональность входит в поставляемую базовую конфигурацию PHP и можно использовать инструмент “из-коробки”. Если по какой-то причине используется сборка без данного пакетного менеджера, то его можно установить из репозитория дистрибутива.

1. Как использовать PEAR для установки библиотек?

Для того, чтобы установить какое-либо расширение, нужно воспользоваться командой: pearinstall<имя модуля>. При появлении новой версии пакет можно обновить с помощью команды: pearupgrade<имя модуля>.

1. Как использовать Composer для обработки зависимостей PEAR?

Если вы уже используете Composer и желаете установить какой-то код из PEAR, вы можете использовать Composer для обработки зависимостей PEAR.

{

"repositories": [

{

"type": "pear",

"url": "http://pear2.php.net"

}

],

"require": {

"pear-pear2/PEAR2\_Text\_Markdown": "\*",

"pear-pear2/PEAR2\_HTTP\_Request": "\*"

}

}

1. Что такое PECL?

PECL - это репозиторий расширений PHP, доступ к которым предоставляется через систему» PEAR

1. Как называется репозиторий, содержащий Composer-совместимые библиотеки?

На данный момент существует много PHP библиотек, которые совместимы с Composer, готовых для использования в вашем проекте. Список этих «пакетов» есть на Packagist, официальном репозитории для Composer-совместимых PHP библиотек.

1. С помощью какой библиотеки можно получить детальный отчет о работе приложения?

PHP Benchmark является удобным инструментом получения статистики приложения и ее визуализации. Можно получать не только конечные данные, но и собирать обширную промежуточную статистику.

1. С помощью какой библиотеки можно упростить себе работу с регулярными выражениями в PHP?

RegExpBuilder - полноценный аналог одноименной библиотеки для языка программирования JavaScript. RegExpBuilder интегрирует регулярные выражения в язык программирования, тем самым облегчая их чтение и обслуживание.

1. С помощью какой библиотеки возможны быстрые и эффективные запросы на PHP, данная библиотека является аналогом jQuery.

PhpQuery существует множество реализаций данного расширения. В данном обзоре описывается расширение, которое устанавливается пакетным менеджером Composer. Это простая оболочка библиотеки xpath, которая позволяет выполнять запросы на html-странице с более удобным синтаксисом jquery.

1. С помощью какой библиотеки возможно простое и эффективное генерирование документов в формате PDF?
2. С помощью какой библиотеки возможен эффективный парсинг HTML/XML?

mPDF − это библиотека PHP, которая генерирует PDF-файлы из HTML-кода в кодировке UTF-8. Она основана на инструментах FPDF и HTML2FPDF с рядом улучшений, позволяющих данной библиотеке являться одним из лучших инструментов генерации PDF-файлов для языка программирования PHP.

1. С помощью какой библиотеки возможно создание диаграмм на движке GOOGLE?

GoogChart является классом PHP для создания динамичных диаграмм с помощью GoogleCharts. Не распространяется на весь API GoogleCharts, но де-лает их очень простыми в использовании.

1. С помощью какой библиотеки возможно сканирование конфигурационного файла PHP на предмет безопасности?

Iniscan — это инструмент, предназначенный для сканирования конфигура-ционного файла php.ini на предмет общих методов обеспечения безопасности и выдачи отчета о результатах сканирования

1. С помощью какой библиотеки возможен анализ HTML и удаление вредоносного кода для защиты от XSS атак.

HTML Purifier − библиотека для фильтрации HTML, разработанная для удаления всего вредоносного кода. Может быть использована как мощный ин-струмент для защиты кода от XSS атак.

1. С помощью какой библиотеки возможно построение графиков, диаграмм и другого структурированного контента на PHP?

JpGraph − PHP библиотека для создания объектно-ориентированных графиков. Эта библиотека позволяет строить линейные диаграммы, гистограммы, пироги, карты, свечи, полярные, радар, круговые и контурные диаграммы любого размера. Является одним из мощнейших инструментов построения графиков для языка программирования PHP.

1. С помощью какой библиотеки облегчается процесс загрузки и валидации файлов на PHP?

Upload − библиотека, которая облегчает загрузку и валидацию файлов. Ко-гда форма отправлена, библиотека может проверить тип файла и его размер, что облегчает работу с функциональностью загрузки файлов на сервер и их дальнейшей обработки.

# Ссылка на удалённый репозиторий проекта

Код решения доступен по ссылке: https://github.com/IvanCherepanov/RSChIR\_backend/tree/master/task\_6

# 

# Список использованной литературы

1. Официальный репозиторий образов Docker, URL: https://hub.docker.com/
2. Как сделать Composer Install внутри Docker-контейнера, URL: https://highload.today/composer-install-docker/
3. Faker, URL: https://fakerphp.github.io/
4. FakerPHP/Faker, URL: URL: https://github.com/FakerPHP/Faker/tree/main/src/Faker/Core
5. Installing GD in Docker, URL: https://stackoverflow.com/questions/39657058/installing-gd-in-docker
6. Куки, document.cookie, URL: https://learn.javascript.ru/cookie
7. Отправка файлов на сервер, URL: https://metanit.com/php/tutorial/5.4.php
8. Загрузка файлов методом POST, URL: https://www.php.net/manual/ru/features.file-upload.post-method.php
9. Docker самый простой и понятный туториал. Изучаем докер, так, если бы он был игровой приставкой, URL: https://badcode.ru/docker-tutorial-dlia-novichkov-rassmatrivaiem-docker-tak-iesli-by-on-byl-ighrovoi-pristavkoi/amp/
10. Официальная документация языка PHP, URL: https://www.php.net/manual/ru/index.php
11. Docker-compose, URL: https://habr.com/ru/company/ruvds/blog/450312/
12. Docker-compose: идеальное рабочее, URL: окружение https://habr.com/ru/post/346086/
13. Beginner’s Guide, URL http://nginx.org/en/docs/beginners\_guide.html#conf\_structure
14. docker-compose build nginx + php + mysql, URL https://russianblogs.com/article/6388179240/
15. Настраиваем контейнеры Docker, Docker-compose (PHP + MySQL + Nginx + phpMyAdmin), URL: https://infostart.ru/1c/articles/1690130/
16. Готовим локальную среду Docker для разработки на PHP, URL: https://phptoday.ru/post/gotovim-lokalnuyu-sredu-docker-dlya-razrabotki-na-php
17. НАСТРОЙКА ВИРТУАЛЬНЫХ ХОСТОВ APACHE, URL: https://losst.ru/nastrojka-virtualnyh-hostov-apache