

Министерство науки и высшего образования Российской Федерации
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО
ITMO University

Отчет по лабораторной работе 3

По дисциплине WEB-программирование

Тема работы «Gulp и веб-сервера»

Обучающийся Петрова Наталья Глебовна

Факультет факультет инфокоммуникационных технологий

Группа K3321

Направление подготовки 11.03.02 Инфокоммуникационные технологии и системы связи

Образовательная программа Программирование в инфокоммуникационных системах

Обучающийся	<u>14.12.2024</u> (дата)	<u> </u> (подпись)	<u>Петрова Н. Г.</u> (Ф.И.О.)
Руководитель	<u> </u> (дата)	<u> </u> (подпись)	<u>Марченко Е.В.</u> (Ф.И.О.)

СОДЕРЖАНИЕ

Стр.

ВВЕДЕНИЕ	3
1 Задание 1	4
1.1 Написание gulp файла	4
1.1.1 Последовательное и параллельное выполнение задач ..	4
1.1.2 Выполнение задач с помощью инструмента BrowserSync	5
2 Задание 2	7
2.0.1 Методы GET и POST	9
3 Задание 3	11
ЗАКЛЮЧЕНИЕ	14

ВВЕДЕНИЕ

Цель работы

Целью данной работы является освоение навыков работы с gulp, а также настройки локальных веб-серверов.

Задачи:

1. Настроить gulp: а) создать два таска – настроить на последовательное и параллельное выполнение; б) настроить отображение файлов проекта в браузере и автоматическую перезагрузку при изменении одного из контролируемых файлов проекта.

2. Создать форму для отправки информации по обратной связи от пользователя сайта – передает информацию о себе: имя, фамилия, электронная почта, поле с обратной связью, должны быть радиокнопки (по меньшей мере 2 шт.) и должны быть чекбоксы (не менее трех). Разработать файл с формой и php скрипт по образцу. В отчете привести информацию о использовании методов get и метод post.

3. Установить инструментарий для отладки проектов (LAMP, Денвер - локальный сервер или что-то другое аналогичное на ваше усмотрение). С портала wordpress берете движок и устанавливаете его. Настраиваете портал <http://test.site> При вводе данного адреса отвечает ваш портал. При желании можете поставить одну из тем, которая Вам понравится.

1 Задание 1

1.1 Написание gulp файла

1.1.1 Последовательное и параллельное выполнение задач

В первом задании необходимо было написать gulp-файл таким образом, чтобы при его запуске выполнялись два таска, сначала последовательно, а потом параллельно.

Gulp — это инструмент для автоматизации задач в веб-разработке, основанный на Node.js. Это своего рода сборщик задач, который помогает упростить процесс разработки и повысить производительность.

Файл **gulpfile.js** — это основной конфигурационный файл для Gulp. В нем определяются задачи, которые будут выполняться инструментом.

При **последовательном** выполнении одна задача выполняется после завершения другой. Это удобно, когда одна задача зависит от результата предыдущей.

Параллельное выполнение позволяет выполнять несколько задач одновременно, что может значительно ускорить процесс сборки. Однако этот подход подходит только для тех задач, которые не зависят друг от друга.

```

D: > WEB > Лаба_3 > gulpfile.js > ...
1  const gulp = require('gulp');
2  const {series, parallel} = require('gulp');
3
4
5  function Natasha(cb) {
6
7      console.log("I invite you to my house!");
8      cb();
9  }
10
11 function Dachа(cb) {
12
13     console.log('Welcome to my Dachа!');
14     cb();
15 }
16
17
18 exports.ser = series(Natasha,Dacha);
19 exports.par = parallel(Dacha,Natasha);
20

```

Рисунок 1.1 — Содержимое gulpfile

```

D:\WEB\Лаба_3>gulp par
[00:28:54] Using gulpfile D:\WEB\Лаба_3\gulpfile.js
[00:28:54] Starting 'par'...
[00:28:54] Starting 'Dacha'...
[00:28:54] Starting 'Natasha'...
Welcome to my Dachа!
[00:28:54] Finished 'Dacha' after 3.83 ms
I invite you to my house!
[00:28:54] Finished 'Natasha' after 5.14 ms
[00:28:54] Finished 'par' after 15 ms

D:\WEB\Лаба_3>gulp ser
[00:29:10] Using gulpfile D:\WEB\Лаба_3\gulpfile.js
[00:29:10] Starting 'ser'...
[00:29:10] Starting 'Natasha'...
I invite you to my house!
[00:29:10] Finished 'Natasha' after 3.04 ms
[00:29:10] Starting 'Dacha'...
Welcome to my Dachа!
[00:29:10] Finished 'Dacha' after 2.28 ms
[00:29:10] Finished 'ser' after 12 ms

```

Рисунок 1.2 — Результат выполнения через командную строку

1.1.2 Выполнение задач с помощью инструмента BrowserSync

BrowserSync-это инструмент, который позволяет автоматически обновлять страницу при изменении файлов проекта. В данном случае, при малейшем изменении файла html сразу изменится вид веб-страницы. При запуске через командную строку файла, достаточно прописать только gulp, так как

по умолчанию будет вызвана функция `serve`, так как в коде файла она задана как основная.

После запуска открывается страница, которая отображает содержание `html` файла. Теперь можно параллельно редактировать `html` документ и наблюдать за изменениями сразу на странице браузера.

Большое преимущество данного инструмента в том, что он позволяет работать таким образом не только с `html` файлами, но и `css` и `js`.

```
D: > WEB > Лаба_3 > gulpfile.js > ...
1  const gulp = require('gulp');
2  const browserSync = require('browser-sync').create();
3
4  function serve() {
5    browserSync.init({
6      server: {
7        baseDir: 'D:/WEB/Лаба_3'
8      }
9    });
10
11    gulp.watch('*.html').on('change', browserSync.reload);
12  }
13  exports.serve = serve;
14  exports.default = serve;
15
```

Рисунок 1.3 — Содержимое `gulpfile`

```
D: > WEB > Лаба_3 > index.html > html
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4    <meta charset="UTF-8">
5    <meta name="viewport" content="width=device-width, initial-scale=1.0">
6    <title>Document</title>
7
8
9  </head>
10 <body>
11   <h3>Приветствую Вас на моей страничке!!!</h3>
12   <p>Меня зовут Петрова Наталья Глебовна</p>
13 </body>
14 </html>
```

Рисунок 1.4 — `html` файл

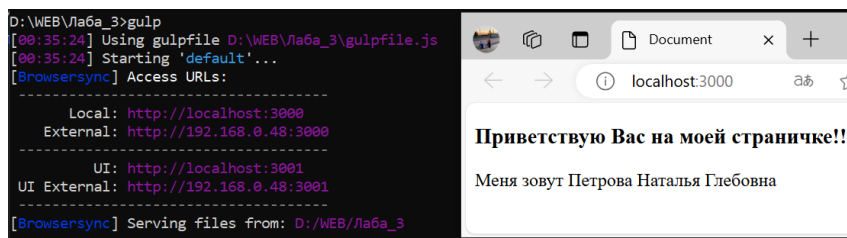


Рисунок 1.5 — Результат выполнения

2 Задание 2

Второе задание было посвящено созданию формы обратной связи, а также тестирование двух вариантов отправки данных: через POST и GET запросы.

При выполнении данного задания был написан простой html файл, который позволил создать небольшую форму. Затем с помощью php скрипта появляется возможность управлять html файлом, а точнее данными, которые пользователь вводит в форму.

```
index.html > ...
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4    <meta charset="UTF-8">
5    <meta name="viewport" content="width=device-width, initial-scale=1.0">
6    <title>Задание 2</title>
7  </head>
8  <body>
9    <h1>Анкета пользователя</h1>
10   <div>
11     <form method="post" action="script.php">
12       <label for="name">Имя</label><br>
13       <input type="text" id="name" name="name" required><br>
14       <label for="lastname">Фамилия</label><br>
15       <input type="text" id="lastname" name="lastname" required><br>
16       <label for="email">Электронная почта</label><br>
17       <input type="email" id="email" name="email" required><br>
18       <label for="txtarea">Сообщение</label><br>
19       <textarea rows="4" cols="50" id="txtarea" name="txtarea" required>Напишите свой комментарий здесь</t
20       <label>Какой любимый цвет?</label><br>
21       <p><input name="radiobtn" type="radio" value="Черный"> Черный</p>
22       <p><input name="radiobtn" type="radio" value="Белый"> Белый</p>
23       <label>А тут какой?</label><br>
24       <p><input name="checkboxbtn[]" type="checkbox" value="Розовый"> Розовый</p>
25       <p><input name="checkboxbtn[]" type="checkbox" value="Зеленый"> Зеленый</p>
26       <p><input name="checkboxbtn[]" type="checkbox" value="Желтый"> Желтый</p>
27       <input type="submit" value="Отправить Post">
28       <button id="btn-get" formaction="script.php" formmethod="get">Отправить Get</button>
29     </form>
30   </div>
31 </body>
32 </html>
33
```

Рисунок 2.1 — html файл

```

script.php
1  <?php
2  if ($_SERVER['REQUEST_METHOD'] == 'POST') {
3      $name = htmlspecialchars($_POST['name']);
4      $last_name = htmlspecialchars($_POST['lastname']);
5      $email = htmlspecialchars($_POST['email']);
6      $msg = htmlspecialchars($_POST['txtarea']);
7      $radiobtn = htmlspecialchars($_POST['radiobtn']);
8      $checkbtn = isset($_POST['checkbtn']) ? $_POST['checkbtn'] : [];
9      echo "<h1>Спасибо, данные зафиксированы! Форма отправлена</h1>";
10     echo "<p>Имя: $name </p>";
11     echo "<p>Фамилия: $last_name </p>";
12     echo "<p>Электронная почта: $email </p>";
13     echo "<p>Сообщение пользователя: $msg </p>";
14     echo "<p>Любимый цвет: $radiobtn </p>";
15     if (!empty($checkbtn)) {
16         echo "<p> Любимые цвета: " . implode(", ", $checkbtn) . "</p>";} else {
17         echo "<p> Любимые цвета: не выбрано";}
18
19
20
21
22 } elseif ($_SERVER["REQUEST_METHOD"] == "GET") {
23     $name = htmlspecialchars($_GET['name']);
24     $last_name = htmlspecialchars($_GET['lastname']);
25     $email = htmlspecialchars($_GET['email']);
26     $msg = htmlspecialchars($_GET['txtarea']);
27     $radiobtn = htmlspecialchars($_GET['radiobtn']);
28     $checkbtn = isset($_GET['checkbtn']) ? $_GET['checkbtn'] : [];
29     echo "<h1>Спасибо, данные зафиксированы! Форма отправлена</h1>";
30     echo "<p>Имя: $name </p>";
31     echo "<p>Фамилия: $last_name </p>";
32     echo "<p>Электронная почта: $email </p>";
33     echo "<p>Сообщение пользователя: $msg </p>";
34     echo "<p>Любимый цвет: $radiobtn </p>";
35     if (!empty($checkbtn)) {
36         echo "<p> Любимые цвета: " . implode(", ", $checkbtn) . "</p>";} else {
37         echo "<p> Любимые цвета: не выбрано";}
38 } else {
39     echo "<h1> Ошибка </h1>";
40 }
41 ?>

```

Рисунок 2.2 — php скрипт

The screenshot shows a web browser window with the address bar displaying 'localhost/lab_3_task2/'. The page title is 'Анкета пользователя'. The form contains the following elements:

- Input fields for 'Имя' (Name) and 'Фамилия' (Surname).
- An input field for 'Электронная почта' (Email).
- A text area for 'Сообщение' (Message) with the placeholder text 'Напишите свой комментарий здесь'.
- A label 'Какой любимый цвет?' (What is your favorite color?) followed by three radio buttons: 'Черный' (Black), 'Белый' (White), and 'А тут какой?' (What about this?).
- Below the radio buttons are three checkboxes: 'Розовый' (Pink), 'Зеленый' (Green), and 'Желтый' (Yellow).
- At the bottom, there are two buttons: 'Отправить Post' (Send Post) and 'Отправить Get' (Send Get).

Рисунок 2.3 — Запуск формы

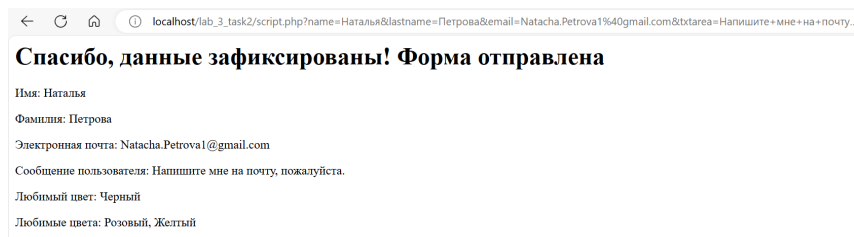


Рисунок 2.4 — Результат выполнения метода get

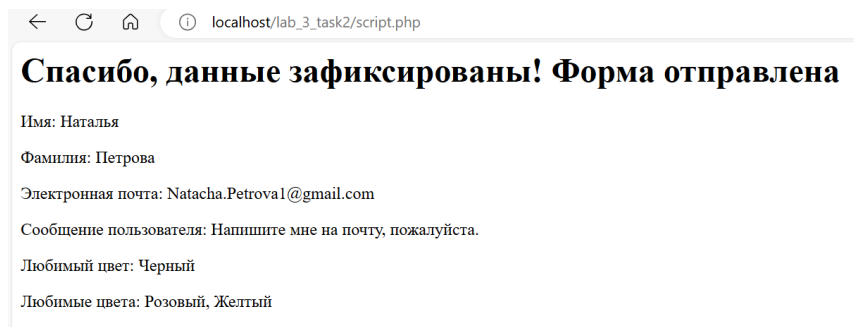


Рисунок 2.5 — Результат выполнения метода post

2.0.1 Методы GET и POST

После отправки формы двумя методами можно заметить различия между ними:

Метод GET:

1. **Передача данных:** Данные передаются в URL-адресе, добавляясь к нему после символа ?.
2. **Ограничения:** Максимальная длина данных ограничена длиной URL-адреса (обычно около 2048 символов).
3. **Видимость:** Все данные видны в адресной строке браузера, что может быть небезопасно при передаче конфиденциальной информации.

Метод POST:

1. **Передача данных:** Данные передаются внутри тела HTTP-запроса.

2. **Ограничения:** Нет ограничений по длине данных, но есть ограничения по размеру заголовков HTTP-запросов.

3. **Видимость:** Данные не видны в адресной строке браузера, что повышает безопасность.

3 Задание 3

В данном задании было необходимо установить локальный сервер, для операционной системы Windows был выбран **wamp**. Помимо этого был установлен движок с портала Wordpress, который в последствии будет помогать управлять сайтом. Движок был установлен по пути D:/WAMP/www.

После установки всех необходимых компонентов в адресной строке был прописан адрес `localhost/phpmyadmin` и создана БД `lab3`.

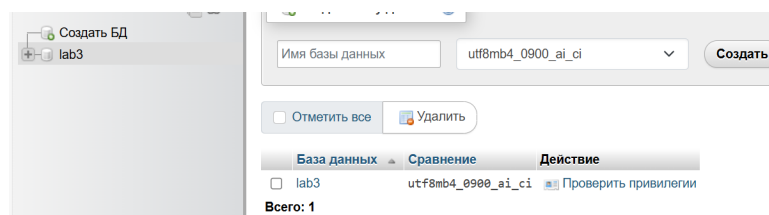


Рисунок 3.1 — Создание БД под сайт test.site

После успешной попытки создания сайта открывается страничка администратора Wordpress.

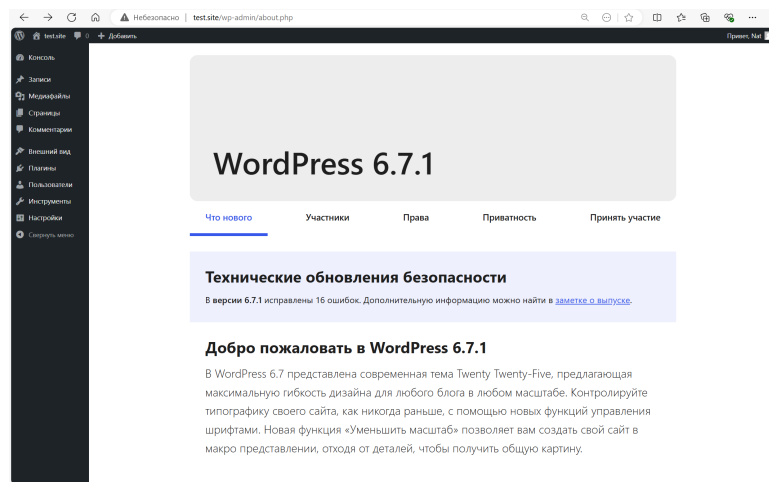


Рисунок 3.2 — Регистрация в wordpress

Согласно заданию необходимо настроить портал `http://test.site`. То есть при вводе данного адреса в адресную строку должен отвечать мой портал.

Чтобы выполнить поставленную задачу необходимо немного видоизменить файлы проекта:

1. Была добавлена строка 127.0.0.1 test.site в конце файла (это IP-адрес, который используется компьютером для ссылки на самого себя) в файл hosts, который расположен по пути: C:/Windows/System32/drivers/etc

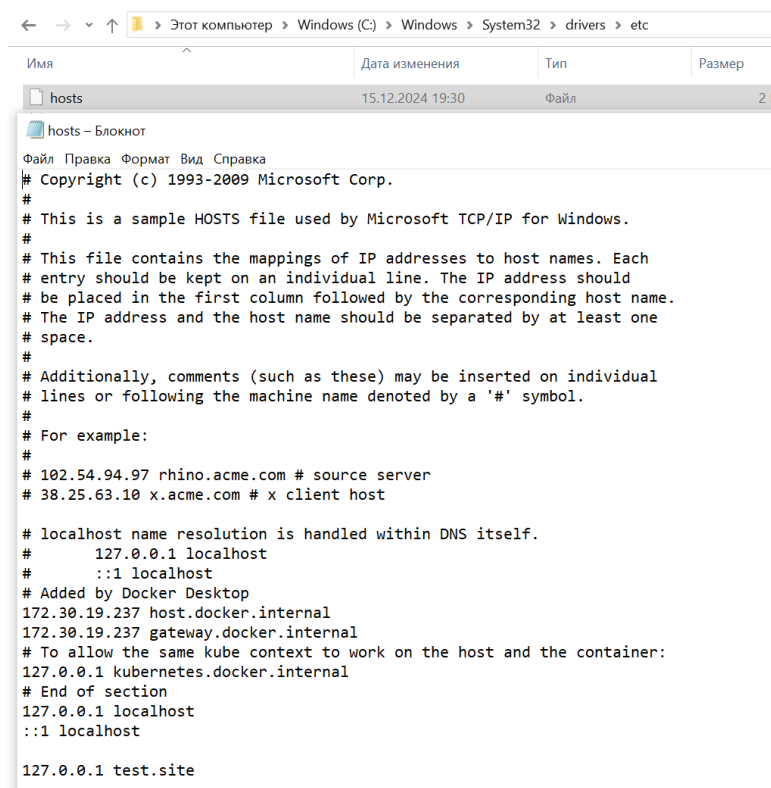


Рисунок 3.3 — Изменение файла hosts

2. В файле httpd.conf по пути D:/WAMP/bin/apache/apache2.4.62.1/conf были раскомментированы строки, которые запрещали настройку виртуальных хостов.

3. В файле httpd-vhosts.conf в D:/WAMP/bin/apache/apache2.4.62.1/conf/extra был настроен локальный хост для запуска на нем сайта.

```

D: > WAMP > bin > apache > apache2.4.62.1 > conf > extra > httpd-vhosts.conf
1  # Virtual Hosts
2  #
3  <VirtualHost _default_:80>
4      ServerName localhost
5      ServerAlias localhost
6      DocumentRoot "${INSTALL_DIR}/www"
7      <Directory "${INSTALL_DIR}/www/">
8          Options +Indexes +Includes +FollowSymLinks +MultiViews
9          AllowOverride All
10         Require local
11     </Directory>
12 </VirtualHost>
13
14 <VirtualHost *:80>
15     ServerName test.site
16     DocumentRoot "D:/WAMP/www/wordpress"
17     <Directory "D:/WAMP/www/wordpress/">
18         AllowOverride All
19         Require all granted
20     </Directory>
21 </VirtualHost>
22

```

Рисунок 3.4 — Настройка виртуального хоста

После всех проделанных манипуляций сервер WAMP был перезагружен. И в таблице `wp_options` БД `lab3` значения полей `siteurl` и `home` было изменено на `test.site`. `Siteurl` определяет полный URL сайта, включая протокол, а `home` указывает на главную страницу сайта (как правило они совпадают, но могут отличаться если главная страница расположена по другому адресу)

	option_id	option_name	option_value	autoload
<input type="checkbox"/>	1	cron	a:10:{i:1734302588;a:1:{s:34:"wp_privacy_delete_o...	on
<input type="checkbox"/>	2	siteurl	http://test.site	on
<input type="checkbox"/>	3	home	http://test.site	on
<input type="checkbox"/>	4	blogname	test.site	on
<input type="checkbox"/>	5	blogdescription		on
<input type="checkbox"/>	6	users_can_register	0	on
<input type="checkbox"/>	7	admin_email	Natacha.Petrova1@gmail.com	on

Рисунок 3.5 — Изменение полей таблицы

В результате при вводе в адресную строку `http://test.site` успешно открывается портал, более того, на сайте `wordpress` от имени администратора я изменила тему сайта.

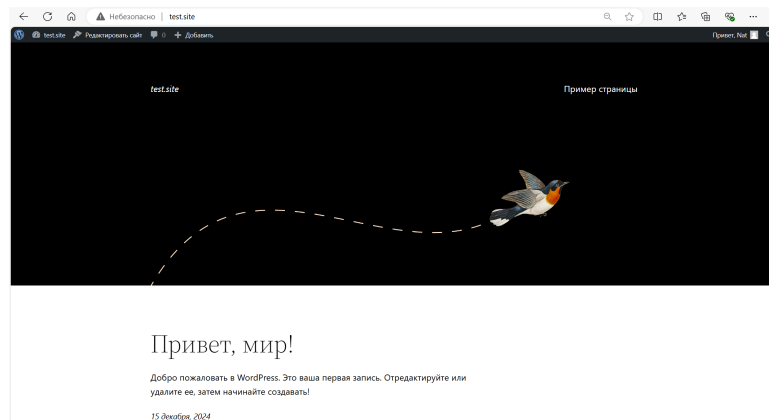


Рисунок 3.6 — Результат работы

ЗАКЛЮЧЕНИЕ

В результате выполнения лабораторной работы были получены навыки работы с gulp файлами, а также приобретено умение настройки локальных серверов.