

# НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО

Факультет инфокоммуникационных технологий

Направление подготовки 11.03.02 «Инфокоммуникационные технологии и  
системы связи»

Лабораторная работа №3

Выполнил:

Маноменов Иван Андреевич

Группа №3323

Проверил:

Марченко Елена Вадимовна

Санкт-Петербург

2024

# СОДЕРЖАНИЕ

Стр.

<b>1</b>	<b>Цель работы.....</b>	<b>3</b>
<b>2</b>	<b>Ход работы .....</b>	<b>4</b>
2.1	Упражнение 1 .....	4
2.2	Упражнение 2 .....	7
2.3	Упражнение 3 .....	9
	<b>ЗАКЛЮЧЕНИЕ .....</b>	<b>12</b>

## **1 Цель работы**

Освоить навыки работы с `gulp`, `get` и `post` запросами и их обработку через `php`.

## 2 Ход работы

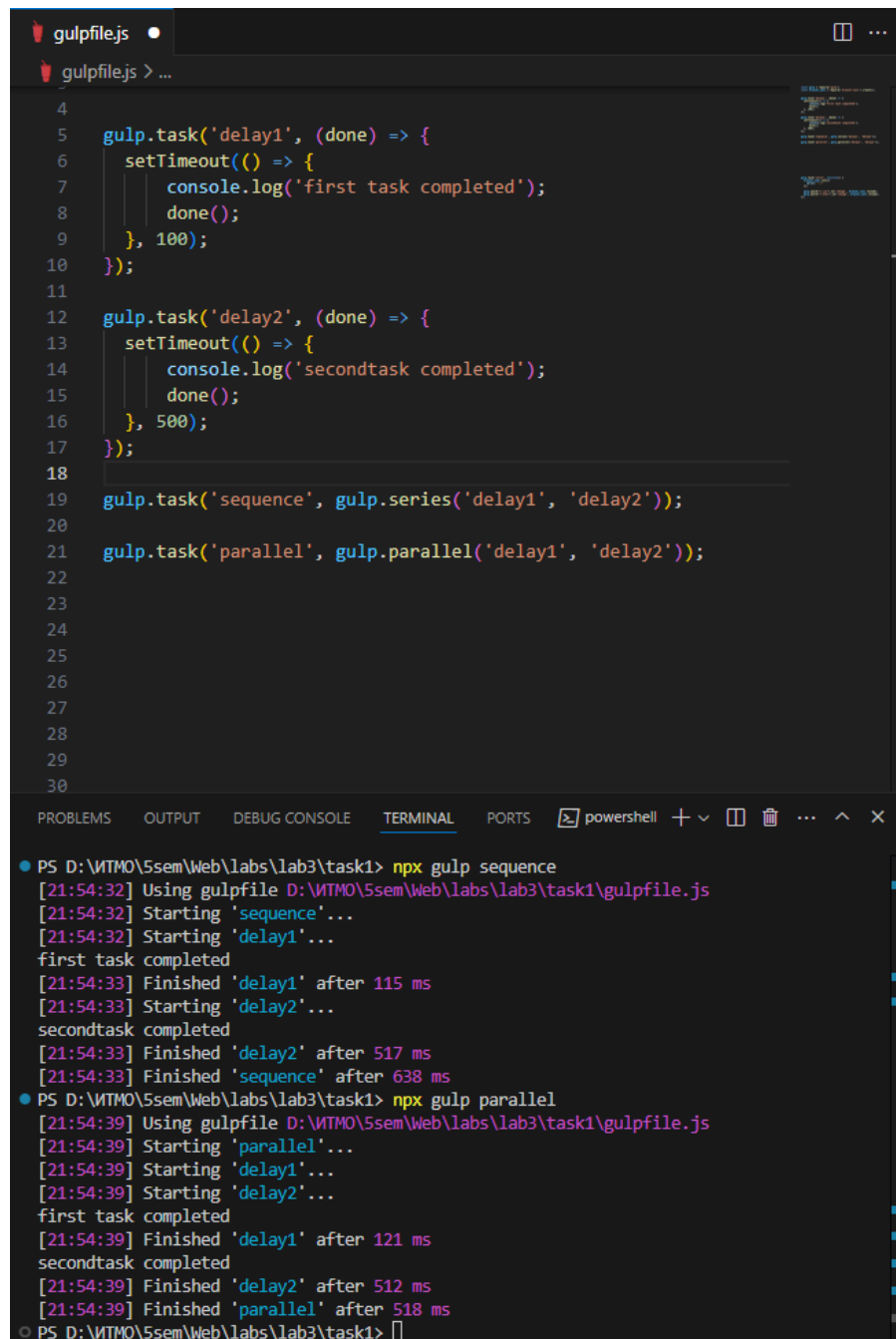
### 2.1 Упражнение 1

Задача упражнения:

а) Создать 2 таска, настроить их на последовательное и параллельное выполнение

б) Настроить отображение файлов проекта в браузере и автоматическую перезагрузку при изменении одного из контролируемых файлов проекта

Для выполнения первой части задания был написан файл *gulpfile.js* с двумя простыми тасками: одна ожидает 500 мс, другая - 100 мс. Как можно видеть, при параллельном и последовательном выполнении время выполнения изменилось (рисунок 2.1).



```
gulpfile.js
gulpfile.js > ...

4
5  gulp.task('delay1', (done) => {
6    setTimeout(() => {
7      console.log('first task completed');
8      done();
9    }, 100);
10  });
11
12  gulp.task('delay2', (done) => {
13    setTimeout(() => {
14      console.log('secondtask completed');
15      done();
16    }, 500);
17  });
18
19  gulp.task('sequence', gulp.series('delay1', 'delay2'));
20
21  gulp.task('parallel', gulp.parallel('delay1', 'delay2'));
22
23
24
25
26
27
28
29
30

PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  powershell + - [ ] [ ] ... ^ X

• PS D:\VTMO\5sem\Web\labs\lab3\task1> npx gulp sequence
[21:54:32] Using gulpfile D:\VTMO\5sem\Web\labs\lab3\task1\gulpfile.js
[21:54:32] Starting 'sequence'...
[21:54:32] Starting 'delay1'...
first task completed
[21:54:33] Finished 'delay1' after 115 ms
[21:54:33] Starting 'delay2'...
secondtask completed
[21:54:33] Finished 'delay2' after 517 ms
[21:54:33] Finished 'sequence' after 638 ms
• PS D:\VTMO\5sem\Web\labs\lab3\task1> npx gulp parallel
[21:54:39] Using gulpfile D:\VTMO\5sem\Web\labs\lab3\task1\gulpfile.js
[21:54:39] Starting 'parallel'...
[21:54:39] Starting 'delay1'...
[21:54:39] Starting 'delay2'...
first task completed
[21:54:39] Finished 'delay1' after 121 ms
secondtask completed
[21:54:39] Finished 'delay2' after 512 ms
[21:54:39] Finished 'parallel' after 518 ms
○ PS D:\VTMO\5sem\Web\labs\lab3\task1> [ ]
```

Рисунок 2.1 — Параллельное и последовательное выполнение задач

Для выполнения второй части задания был написан простой html и scc файл, отображающие сайт, показанный на рисунке 2.3 (слева), а также функция serve, отслеживающая изменения css и html файлов (рисунок 2.2). Теперь при изменении html и css файлов браузерная страница автоматически обновляется (рисунки 2.3 и 2.4)

```
function serve() {
  browser_sync.init({
    server: "./"
  });

  gulp.watch("*.css").on('change', browser_sync.reload);
  gulp.watch("*.html").on('change', browser_sync.reload);
};

gulp.task('sequence', gulp.series('delay1', 'delay2'));
gulp.task('parallel', gulp.parallel('delay1', 'delay2'));
gulp.task('default', serve);
```

Рисунок 2.2 — Функция serve

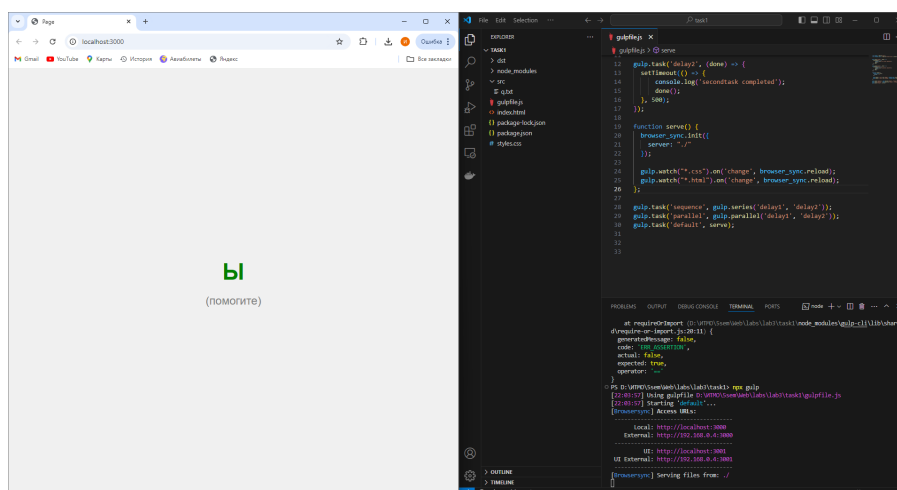


Рисунок 2.3 — Изначальный вид страницы index.html

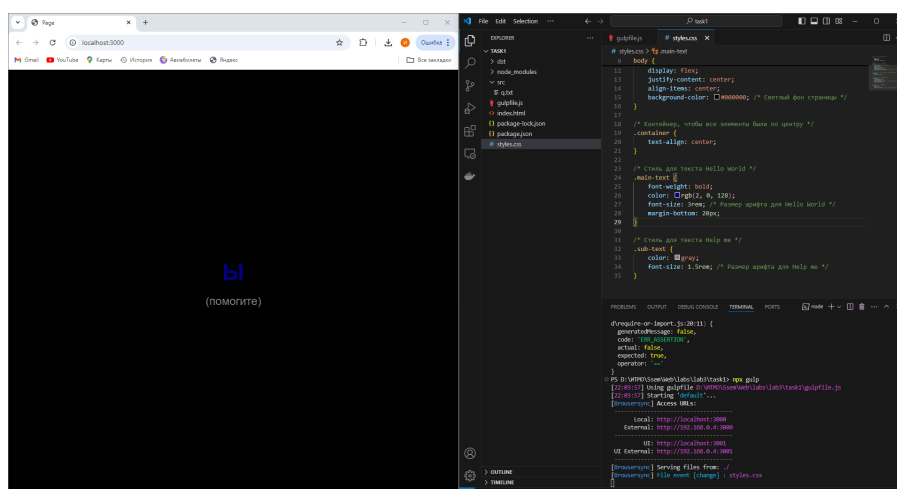


Рисунок 2.4 — Вид страницы index.html после изменений

## 2.2 Упражнение 2

Целью второго упражнения было создать форму для отправки обратной связи и php скрипт с обработкой информации из формы.

Был написан файл *feedback\_form.html* (рисунок 2.5) и php-скрипт *process\_feedback.php* (рисунок 2.6). Для работы с php был установлен и включен сервер ХАМПП

```
task2 > feedback_form.html > html > body > form > br
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>Обратная связь</title>
7 </head>
8 <body>
9
10 <h2>Форма обратной связи</h2>
11 <form action="process_feedback.php" method="POST">
12 <label for="first_name">Имя:</label>
13 <input type="text" id="first_name" name="first_name" required><br><br>
14
15 <label for="last_name">Фамилия:</label>
16 <input type="text" id="last_name" name="last_name" required><br><br>
17
18 <label for="email">Электронная почта:</label>
19 <input type="email" id="email" name="email" required><br><br>
20
21 <label for="message">Сообщение:</label><br>
22 <textarea id="message" name="message" rows="4" cols="50" required></textarea><br><br>
23
24 <label>Оценка работы сервиса:</label><br>
25 <input type="radio" id="5" name="grade" value="excellent" required>
26 <label for="5">Отлично</label><br>
27 <input type="radio" id="4" name="grade" value="good" required>
28 <label for="4">Хорошо</label><br>
29 <input type="radio" id="3" name="grade" value="norm" required>
30 <label for="3">Нормально</label><br>
31 <input type="radio" id="2" name="grade" value="bad" required>
32 <label for="2">Плохо</label><br>
33 <input type="radio" id="1" name="grade" value="awful" required>
34 <label for="1">Отвратительно</label><br>
35
36 <label>Дополнительные опции:</label><br>
37 <input type="checkbox" id="newsletter" name="options[]" value="newsletter">
38 <label for="newsletter">Подписка на новости</label><br>
39 <input type="checkbox" id="updates" name="options[]" value="updates">
40 <label for="updates">Получать обновления</label><br>
41 <input type="checkbox" id="offers" name="options[]" value="offers">
42 <label for="offers">Получать предложения</label><br><br>
43
44 <input type="submit" value="Отправить">
45 </form>
46 </body>
47 </html>
```

Рисунок 2.5 — Файл *feedback\_form.html*

```

task2 > process_feedback.php > ...
1 <?php
2 $first_name = $_POST['first_name'];
3 $last_name = $_POST['last_name'];
4 $email = $_POST['email'];
5 $message = $_POST['message'];
6 $grade = $_POST['grade'];
7 $options = isset($_POST['options']) ? implode(separator: ", ", array: $_POST['options']) : 'Нет дополнительных опций';
8
9
10 echo "<h2>Спасибо за ваш отзыв!</h2>";
11 echo "<p><strong>Имя:</strong> " . htmlspecialchars(string: $first_name) . "</p>";
12 echo "<p><strong>Фамилия:</strong> " . htmlspecialchars(string: $last_name) . "</p>";
13 echo "<p><strong>Оценка работы сервиса:</strong> " . htmlspecialchars(string: $email) . "</p>";
14 echo "<p><strong>Сообщение:</strong> " . nl2br(string: htmlspecialchars(string: $message)) . "</p>";
15 echo "<p><strong>Предпочтительный способ связи:</strong> " . htmlspecialchars(string: $grade) . "</p>";
16 echo "<p><strong>Дополнительные опции:</strong> " . htmlspecialchars(string: $options) . "</p>";
17 ?>
18

```

Рисунок 2.6 — Файл *process\_feedback.php*

Теперь при заполнении формы и ее отправке (рисунок 2.7) открывается страница с полученной информацией (рисунок 2.8). Информация из формы передается в php файл при помощи метода POST

← → ↻ ⓘ localhost/lab3/task2/feedback\_form.html

☰ | Gmail YouTube Карты История Авиабилеты Яндекс

## Форма обратной связи

Имя:

Фамилия:

Электронная почта:

Сообщение:

Оценка работы сервиса:

- ☐ Отлично
- ☐ Хорошо
- ☐ Нормально
- ☒ Плохо
- ☐ Отвратительно

Дополнительные опции:

- ☒ Подписка на новости
- ☐ Получать обновления
- ☒ Получать предложения

Рисунок 2.7 — Страница *feedback\_form.html*



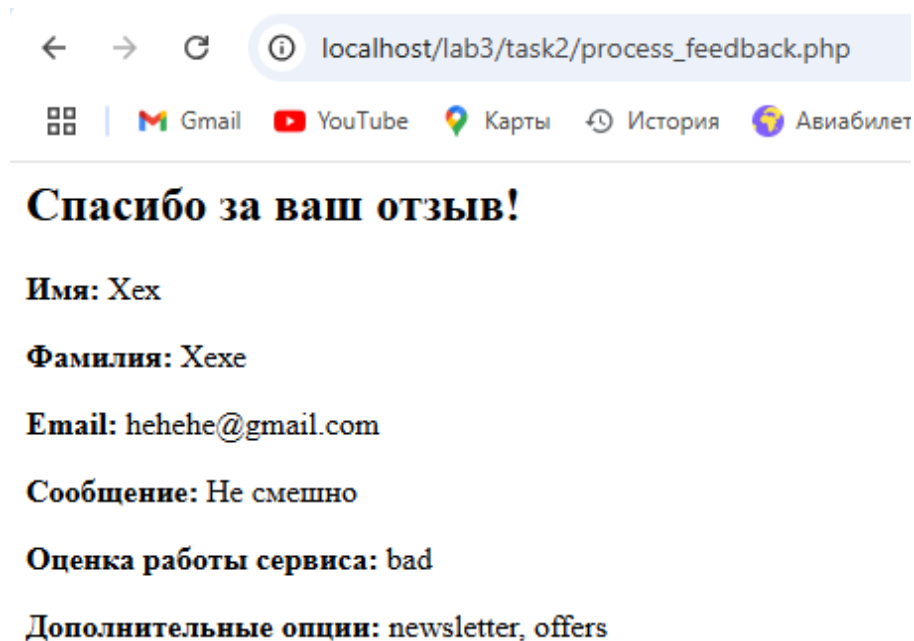


Рисунок 2.8 — Страница с выводом полученной информации

## 2.3 Упражнение 3

В этом упражнении целью было настроить портал `http://test.site` при помощи wordpress.

Помимо установленного ранее XAMPP был также установлен архив Wordpress.

Далее в XAMPP были запущены Apache и MySQL сервера (рисунок 2.9)

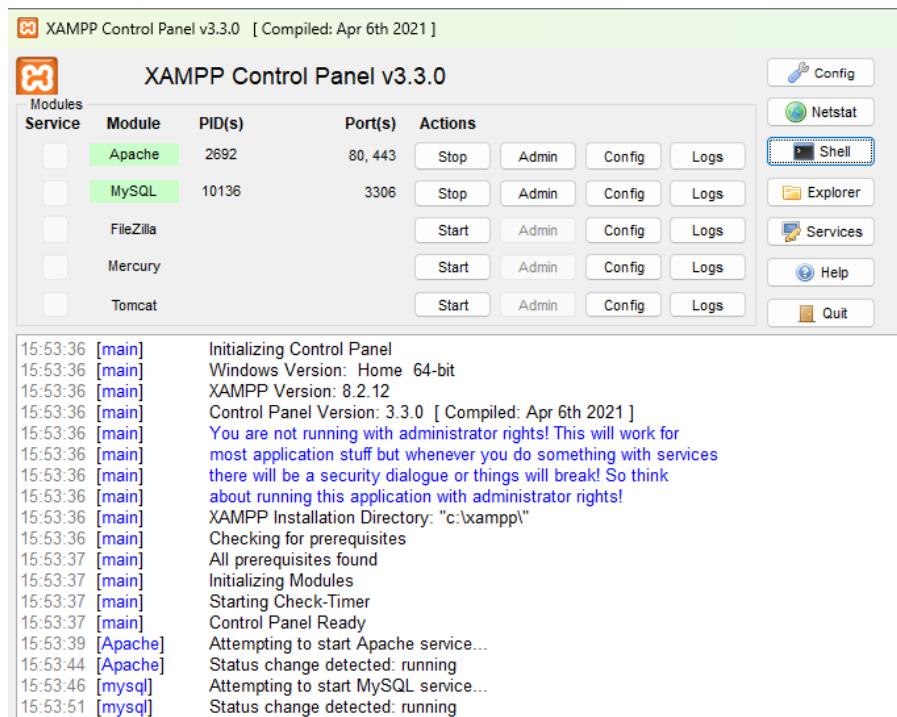


Рисунок 2.9 — Запуск XAMPP

Был запущен wordpress, и в нем заполнена соответствующая форма (рисунок 2.10). Далее была изменена ссылка на сайт

## Добро пожаловать

Добро пожаловать в знаменитую пятиминутную установку WordPress! Просто заполните поля — и вперёд, к использованию самой мощной и гибкой персональной платформы для публикаций в мире!

## Требуется информация

Пожалуйста, укажите следующую информацию. Не переживайте, потом вы всегда сможете изменить эти настройки.

Название сайта	<input type="text" value="Web lab3 task3"/>
Имя пользователя	<input type="text" value="IvanMan"/> <small>Имя пользователя может содержать только латинские буквы, пробелы, подчёркивания, дефисы, точки и символ @.</small>
Пароль	<input type="password" value="11IF212ENKspTHuJR("/> <div>Надёжный</div> <div><a href="#">Скрыть</a></div> <p><b>Важно:</b> Этот пароль понадобится вам для входа. Сохраните его в надёжном месте.</p>
Ваш e-mail	<input type="text" value="i.manomenov@gmail.com"/> <small>Внимательно проверьте адрес электронной почты, перед тем как продолжить.</small>
Видимость для поисковых систем	<input checked="" type="checkbox"/> Попросить поисковые системы не индексировать сайт <small>Будет ли учитываться этот запрос — зависит от поисковых систем.</small>
<div>Установить WordPress</div>	

Рисунок 2.10 — Настройка Wordpress

Результат на рисунке 2.11

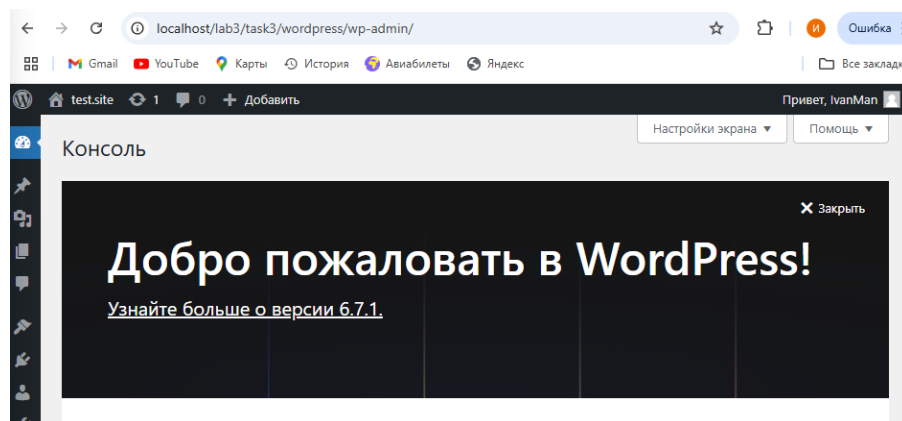


Рисунок 2.11 — Результат

## **ЗАКЛЮЧЕНИЕ**

При выполнении работы были освоены навыки работы с `gulp`, `get` и `post` запросами и их обработки через `php.p`. Цель работы достигнута.