

Санкт-Петербургский политехнический университет Петра Великого

Институт компьютерных наук и технологий

Кафедра компьютерных систем и программных технологий

**Отчет по лабораторной работе**

по дисциплине «Сети и телекоммуникации»

**Разработка HTTP-сервера**

Выполнил

студент гр. 43501/3

Преподаватель

Е.А. Никитин

К.Д. Вылегжанина

«\_\_»\_\_\_\_\_2015г.

Санкт-Петербург

2016

## Задание

Разработать приложение для операционных систем семейства Windows или Linux, обеспечивающее базовые функции сервера протокола HTTP (Web-сервера).

### Основные возможности.

Приложение должно реализовывать следующие функции:

- 1) Обработка подключения клиента
- 2) Разбор строки URL
- 3) Выдача клиенту запрошенного ресурса
- 4) Обеспечение параллельной загрузки клиенту страниц и медиа-элементов
- 5) Обеспечение параллельной работы нескольких клиентов
- 6) Формирование необходимых заголовков протокола HTTP

### Методика тестирования.

В рабочий каталог сервера помещается содержимое какого-либо Web-сайтов сети Internet. Используемый сайт должен иметь несколько уровней вложенности и содержать медиа-элементы (например, графические изображения).

## Ход работы

**HTTP** (англ. *HyperText Transfer Protocol* — «протокол передачи гипертекста») — протокол прикладного уровня передачи данных (изначально — в виде гипертекстовых документов в формате HTML, в настоящий момент используется для передачи произвольных данных). Основой HTTP является технология «клиент-сервер», то есть предполагается существование потребителей (клиентов), которые иницируют соединение и посылают запрос, и поставщиков (серверов), которые ожидают соединения для получения запроса, производят необходимые действия и возвращают обратно сообщение с результатом.

HTTP в настоящее время повсеместно используется во Всемирной паутине для получения информации с веб-сайтов. Основным объектом манипуляции в HTTP является ресурс, на который указывает URI (Uniform Resource Identifier) в запросе клиента. Обычно такими ресурсами являются хранящиеся на сервере файлы, но ими могут быть логические объекты или что-то абстрактное. Особенностью протокола HTTP является возможность указать в запросе и ответе способ представления одного и того же ресурса по различным параметрам: формату, кодировке, языку и т. д. (В частности для этого используется HTTP-заголовок.) Именно благодаря возможности указания способа кодирования сообщения клиент и сервер могут

обмениваться двоичными данными, хотя данный протокол является текстовым.

Было написано приложение - http-сервер на языке Java, который принимает сообщения от браузера и посылает по запросу от браузера данные, находящиеся в папке site, и представляющие собой заготовку сайта с различным контентом (текстовые данные и изображения). Приложение использует локальный адрес (127.0.0.1) и порт 8080. Поддерживается подключение нескольких клиентов и выдача соответствующих данных по их запросам.

### **Тестирование**

Для тестирования использовались два браузера – Google Chrome и Yandex.

Пример сообщения, которое получает сервер от браузера:

GET /images/bgslider-3.jpg HTTP/1.1

Host: localhost:8080

Connection: keep-alive

Accept: image/webp,image/\*,\*/\*;q=0.8

User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/47.0.2526.111 Safari/537.36

Referer: http://localhost:8080/

Accept-Encoding: gzip, deflate, sdch

Accept-Language: en-US,en;q=0.8

На рисунке 1 видно получение данных – заготовки сайта, полученных по запросу Yandex-браузера, на рис. 2 – тоже самое, клиент – Google Chrome.

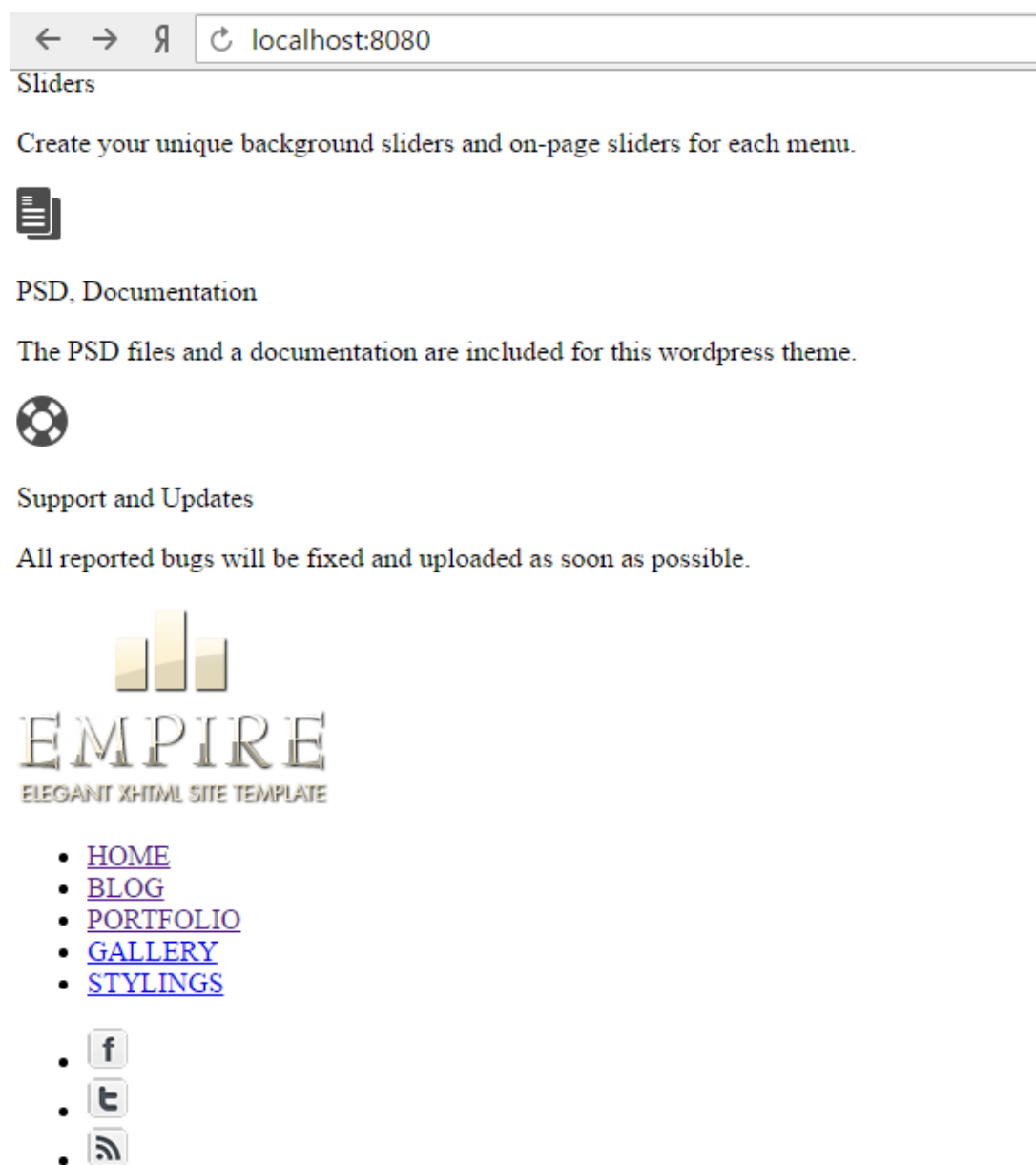


Рис. 1

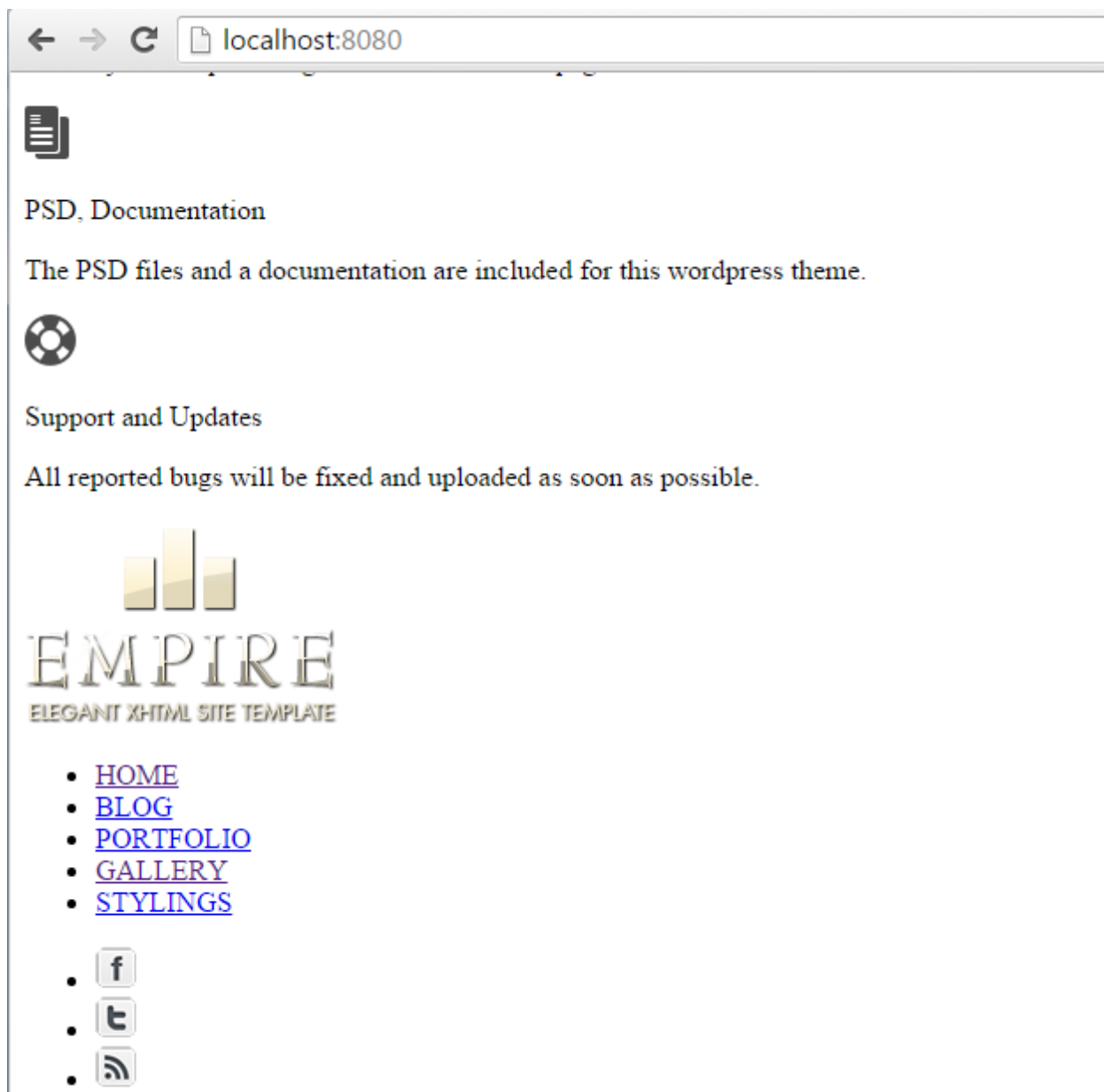


Рис. 2

На рис.3 данные полученные по переходу по ссылке с главной страницы сайта на вложенную страницу.

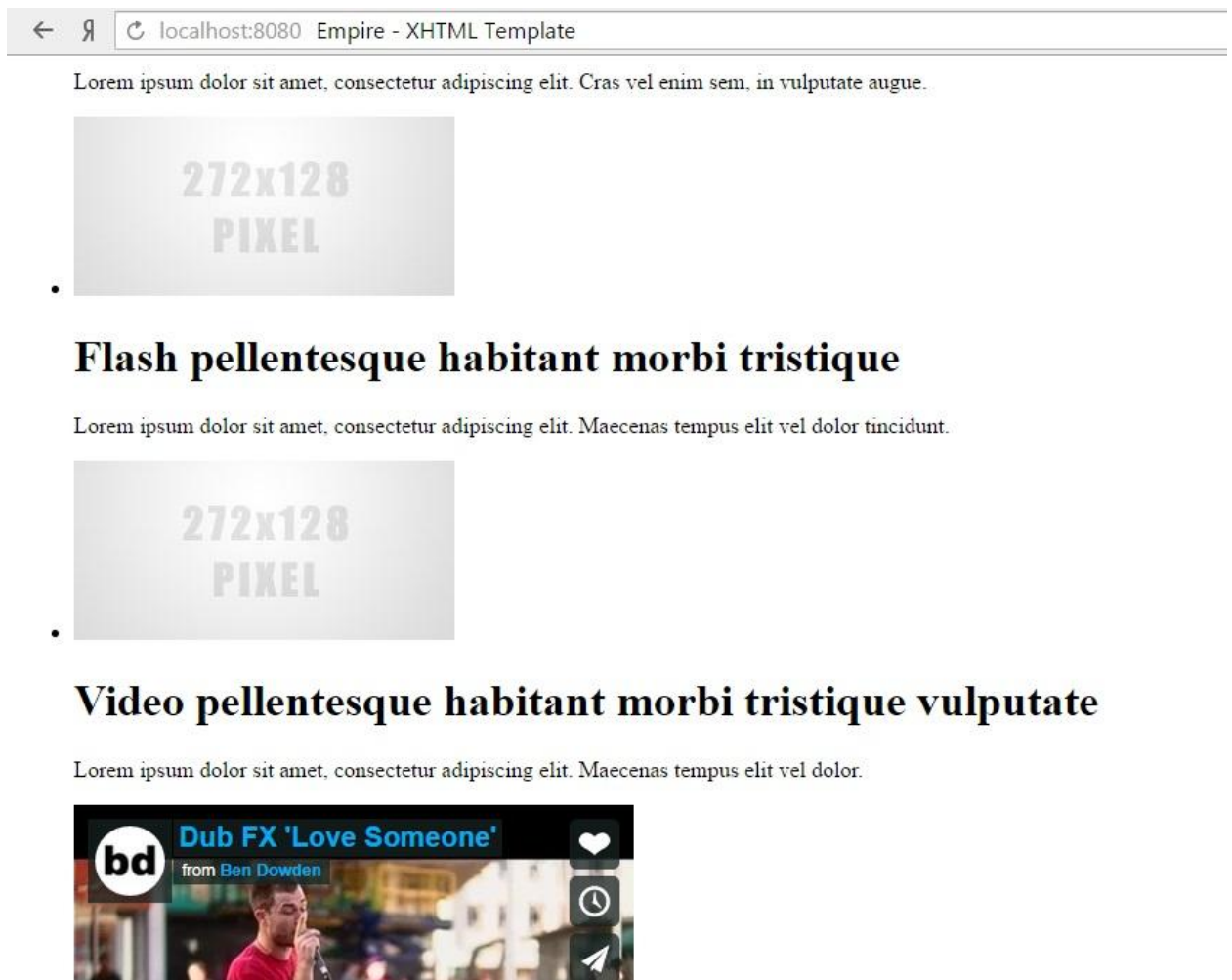


Рис. 3

Вывод:

Был изучен протокол HTTP и написан HTTP-сервер на языке Java. Данный сервер способен принимать запросы от браузера и посылать соответствующие ответы.

Для тестирования работы браузера была скачена заготовка сайта, после чего с помощью браузера посылались запросы на предоставление страниц сайта. Тестирование прошло успешно, ссылки на страницах сайта также работали. Тестирование было проведено с двух браузеров одновременно, чтобы проверить работоспособность подключения сразу нескольких клиентов одновременно.