

# ОГЛАВЛЕНИЕ

Введение.....	3
1. Основы работы с WebSocket .....	7
1.1. Подход к реализации на стороне клиента.....	7
1.2. Подход к реализации на стороне сервера .....	9
1.3. Простейший пример работы с WebSocketD .....	11
1.4. Просмотр процесса взаимодействия по WebSocket-протоколу с помощью Chrome Dev Tools .....	17
1.5. Задание для лабораторной работы.....	20
Контрольные вопросы .....	21
2. Websocketd и серверный код на C#.....	22
2.1. Эхо-сервер на C# и его локальное тестирование .....	22
2.2. Сетевое тестирование эхо-сервера .....	25
2.3. Закрытие соединений со стороны клиента .....	28
2.4. Задание для лабораторной работы.....	32
Контрольные вопросы .....	35
3. WebSocketD и активный код сервера .....	36
3.1. Пример генерации сервером последовательного выходного потока.....	36
3.2. Тестирование программных кодов в консоли разработчика WebSocketD.....	39
3.3. Несколько WebSocket-соединений на одной html-странице.....	42
3.4. Задание для лабораторной работы.....	46
Контрольные вопросы .....	50
4. Использование WebSocketD для удаленного выполнения программ .....	51
4.1. Общий подход к решению задачи .....	51
4.2. Расширение функциональности серверной части.....	56
4.3. Расширение функциональности клиентской части.....	59
4.4. Использование одного WebSocket-сервера для выполнения кодов как Jscript, так и VBScript .....	63
4.5. Работа с файлами на стороне сервера .....	66
4.6. Задание для лабораторной работы.....	69
Контрольные вопросы .....	73
5. Общие сведения о WebSocket-протоколе.....	74
5.1. Отличие WebSocket от других технологий.....	74
5.2. Процесс взаимодействия по WebSocket-протоколу.....	75
5.3. Установление WebSocket-соединения .....	78
5.4. Формат данных WebSocket-протокола.....	81
5.5. Процесс обмена WebSocket-сообщениями .....	83
Контрольные вопросы .....	85
6. Использование утилиты WebSocketD в среде ОС Ubuntu Server .....	87
6.1. Настройка среды виртуализации и ОС Ubuntu .....	87
6.2. Установка WebSocketD в среде ОС Ubuntu Server.....	90
6.3. Простейший пример работы с WebSocketD в ОС Ubuntu.....	93
6.4. Пример эхо-сервера на BASH.....	95

6.5.	Коды серверной обработки, использующие различные скриптовые языки ОС Ubuntu .....	97
6.6.	Несколько WebSocket-серверов на одном компьютере.....	102
6.7.	Приложение. Протокол выполнения консольных команд при начальной настройке WebSocketD в ОС Ubuntu.....	103
6.8.	Задание для лабораторной работы.....	104
	Контрольные вопросы .....	115
7.	Пример реализации простейшего WebSocket-сервера на языке Python.....	116
7.1.	Общие сведения о сетевом программировании в Python .....	116
7.2.	Разработка простейшего WebSocket эхо-сервера.....	128
7.3.	Задание для лабораторной работы.....	143
	Контрольные вопросы .....	155
8.	Использование библиотек Python3 при разработке WebSocket-приложений .....	156
8.1.	Эхо-сервер с использованием библиотеки WebSockets .....	157
8.2.	Общие сведения об асинхронном программировании .....	160
8.3.	Клиент-серверное приложение на базе WebSockets .....	162
8.4.	Пример распределенной системы с учетом синхронизации происходящих в ней процессов .....	166
8.5.	Задание для лабораторной работы.....	170
	Контрольные вопросы .....	172
9.	Использование платформы Node.js при разработке WebSocket-приложений.....	173
9.1.	Создание простого чата с использованием WebSocket-сервера на серверной платформе Node.js.....	173
9.2.	Создание WebSocket-сервера электронных торгов на серверной платформе Node.js .....	179
9.3.	Создание WebSocket-сервера фотосайта на базе серверной платформы Node.js.....	184
10.	Примеры реализации простейшего WebSocket-сервера на различных серверных языках .....	193
10.1.	Простейший WebSocket эхо-сервер на языке PHP.....	193
10.2.	Простейший WebSocket эхо-сервер на языке Perl .....	206
	Приложение. Таблица кодов закрытия WebSocket-протокола .....	208
	Заключение .....	210