



Министерство науки и высшего образования Российской Федерации  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Московский государственный технический университет  
имени Н.Э. Баумана  
(национальный исследовательский университет)»  
(МГТУ им. Н.Э. Баумана)

---

ФАКУЛЬТЕТ \_\_\_\_\_ «Информатика и системы управления»

КАФЕДРА \_\_\_\_\_ «Теоретическая информатика и компьютерные технологии»

**Лабораторная работа № 5**  
**по курсу «Компьютерные сети»**  
**«Реализация WebSocket клиента и сервера на языке Golang»**

Студент группы ИУ9-32Б Волохов А. В.

Преподаватель Посевин Д. П.

*Москва 2023*

# 1 Задание

Исходный код программы представлен в листингах 1– 2– 3– ??– ??– ??.

## Листинг 1 — client.go

```
1 package main
2 import (
3     "bufio"
4     "fmt"
5     "github.com/gorilla/websocket"
6     "os"
7     "strconv"
8 )
9 type Process struct {
10     PID          int    `json:"pid"`
11     ProcessName  string `json:"processName,omitempty"`
12     Error        string `json:"error,omitempty"`
13 }
14 func main() {
15     conn, _, err := websocket.DefaultDialer.Dial("ws://localhost:8080",
16         nil)
17     if err != nil {
18         fmt.Println(err)
19         return
20     }
21     defer func(conn *websocket.Conn) {
22         err := conn.Close()
23         if err != nil {
24             }
25     }(conn)
26     for {
27         fmt.Print("Enter PID: ")
28         scanner := bufio.NewScanner(os.Stdin)
29         scanner.Scan()
30         pidStr := scanner.Text()
31         pid, err := strconv.Atoi(pidStr)
32         if err != nil {
33             fmt.Println("Error: Invalid PID. Please enter a valid number.")
34             continue
35         }
36         processData := Process{
37             PID: pid,
38         }
39         err = conn.WriteJSON(processData)
40         if err != nil {
41             fmt.Println(err)
42             break
43         }
44         var responseData Process
45         err = conn.ReadJSON(&responseData)
46         if err != nil {
47             fmt.Println(err)
48             break
49         }
50         if responseData.Error != "" {
51             fmt.Printf("Error: %s\n", responseData.Error)
52         } else {
53             fmt.Printf("Process Name: %s", responseData.ProcessName)
54         }
55     }
56 }
```

## Листинг 2 — server.go

```

1 package main
2
3 import (
4     "fmt"
5     "github.com/gorilla/websocket"
6     "io/ioutil"
7     "net/http"
8     "path/filepath"
9     "runtime"
10    "strconv"
11 )
12
13 var upgrader = websocket.Upgrader{
14     CheckOrigin: func(r *http.Request) bool {
15         return true
16     },
17 }
18
19 type Process struct {
20     PID          int    `json:"pid"`
21     ProcessName  string `json:"processName,omitempty"`
22     Error        string `json:"error,omitempty"`
23 }
24
25 func getProcessName(pid int) (string, error) {
26     var processName string
27
28     switch runtime.GOOS {
29     case "linux", "darwin":
30         commPath := filepath.Join("/proc", strconv.Itoa(pid), "comm")
31         data, err := ioutil.ReadFile(commPath)
32         if err != nil {
33             return "", err
34         }
35         processName = string(data)
36     default:
37         return "", fmt.Errorf("unsupported operating system")
38     }
39
40     return processName, nil
41 }
42
43 func handleConnection(w http.ResponseWriter, r *http.Request) {
44     conn, err := upgrader.Upgrade(w, r, nil)
45     if err != nil {
46         fmt.Println(err)
47         return
48     }
49     defer func(conn *websocket.Conn) {
50         err := conn.Close()
51         if err != nil {
52             return
53         }
54     }(conn)

```

### Листинг 3 — server.go - продолжение

```
1 for {
2     var process Process
3
4     //
5     err := conn.ReadJSON(&process)
6     if err != nil {
7         fmt.Println(err)
8         break
9     }
10
11     processName, err := getProcessName(process.PID)
12     if err != nil {
13         //
14         OutputData
15         process.Error = err.Error()
16     } else {
17         //
18         process.ProcessName = processName
19     }
20
21     //
22     err = conn.WriteJSON(process)
23     if err != nil {
24         fmt.Println(err)
25         break
26     }
27 }
28
29 func main() {
30     http.HandleFunc("/", handleConnection)
31     fmt.Println("Server is listening on :8080")
32     err := http.ListenAndServe(":8080", nil)
33     if err != nil {
34         return
35     }
36 }
37
```

```
Enter PID: 12
Process Name: rcu_tasks_rude_kthread
Enter PID: 1
Process Name: systemd
Enter PID: 3
Process Name: rcu_gp
Enter PID: 100
Process Name: blkcg_punt_bio
Enter PID: 0
Error: open /proc/0/comm: no such file or directory
Enter PID:
```

Рис. 1 — Пример работы