



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 № 0
по курсу «Компьютерные сети»
«Разработка простейшего web-сервера»

Студент группы ИУ9-31Б Горбунов А. Д.

Преподаватель Посевин Д. П.

Москва 2023

1 Задание

Рассматривается задача разработки web-сервера на языке GO на основе пакета net/http.

0.1:

Задача 1: Реализовать web-сервер и запустить на заданном порте.

Задача 2: Изучить принимаемые web-сервером параметры, реализовать передачу данных методом GET.

Задача 3: Реализовать вывод форматированного гипертекста с контекстным меню в виде гиперссылок, при клике на гиперссылку должна выполняться подмена контента;

0.2:

Задача: Реализовать получение данных из различных RSS-каналов по вариантам. Сравнить результаты разбора и сделать выводы.(<https://news.rambler.ru/rss/Namibia/>)

0.3:

Задача: необходимо разработать web-сервер, который выполняет соединение с удаленным (удаленными) серверами RSS-новостей и возвращает результаты обработки данных в структурированном виде (страница гипертекста) web-клиенту, в нашем случае в браузер по вариантам.

2 Результаты

Исходный код программы представлен в листинге 1, 2

Листинг 1 — web.go

```
1 package main
2
3 import (
4     "encoding/xml"
5     "fmt"
6     "io/ioutil"
7     "log"
8     "net/http"
9
10 )
11
12 type RSS struct {
13     Channel Channel `xml:"channel"`
14 }
15
16 type Channel struct {
17     Title      string `xml:"title"`
18     Description string `xml:"description"`
19     Items      []Item `xml:"item"`
20 }
21
22 type Item struct {
23     Title      string `xml:"title"`
24     Description string `xml:"description"`
25     Link       string `xml:"link"`
26 }
27
28 func HomeRouterHandler(w http.ResponseWriter, r *http.Request) {
29     r.ParseForm()
30
31     fmt.Fprintf(w, "<p>"+r.FormValue("")+"</p>")
32     fmt.Fprintf(w, "<a href='/'>main pagexml</a><br><a href='/rss'>rss</a><br><a href='/about'>about</a>")
33 }
```

Листинг 2 — web.go(продолжение)

```

1 func rssHandler(w http.ResponseWriter, r *http.Request) {
2     resp, err := http.Get("https://news.rambler.ru/rss/Namibia/")
3     if err != nil {
4         return
5     }
6     defer resp.Body.Close()
7     body, err := ioutil.ReadAll(resp.Body)
8     if err != nil {
9         fmt.Println("error", err)
10        return
11    }
12    rss := RSS{}
13    err = xml.Unmarshal(body, &rss)
14    if err != nil {
15        fmt.Println("error", err)
16        return
17    }
18    fmt.Println("Channel title:", rss.Channel.Title)
19    fmt.Println("Channel description:", rss.Channel.Description)
20    fmt.Println("Amount of elements:", len(rss.Channel.Items))
21    for _, item := range rss.Channel.Items {
22        fmt.Println("-----")
23        fmt.Println("Heading:", item.Title)
24        fmt.Println("Description:", item.Description)
25        fmt.Println("Link:", item.Link)
26    }
27    r.ParseForm()
28    fmt.Fprintf(w, "<p>Channel title:"+rss.Channel.Title+"</p>")
29    fmt.Fprintf(w, "<p>Channel description:"+rss.Channel.Description+"</p>")
30    for _, item := range rss.Channel.Items {
31        fmt.Fprintf(w, "<p>-----</p>")
32        fmt.Fprintf(w, "<p>Heading:"+item.Title+"</p>")
33        fmt.Fprintf(w, "<p>Description:"+item.Description+"</p>")
34        fmt.Fprintf(w, "<p>Link:"+item.Link+"</p>")
35    }
36 }
37 func aboutHandler(w http.ResponseWriter, r *http.Request) {
38     r.ParseForm()
39     fmt.Fprintf(w, "<p>"+r.FormValue("")+"</p>")
40     fmt.Fprintf(w, "<a href='/' >main</a><br>about")
41 }
42
43 func main() {
44     http.HandleFunc("/", HomeRouterHandler)
45     http.HandleFunc("/rss", rssHandler)
46     http.HandleFunc("/about", aboutHandler)
47     err := http.ListenAndServe(":9000", nil)
48     if err != nil {
49         log.Fatal("ListenAndServe: ", err)
50     }
51 }

```

Результат запуска представлен на рисунке 1, 2, 3

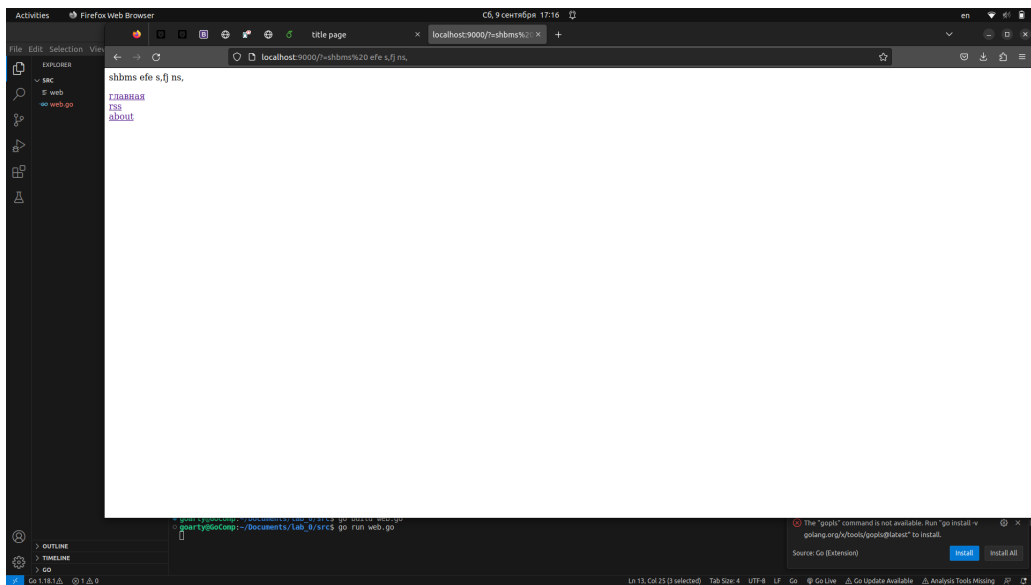


Рис. 1 — Реализация главной страницы

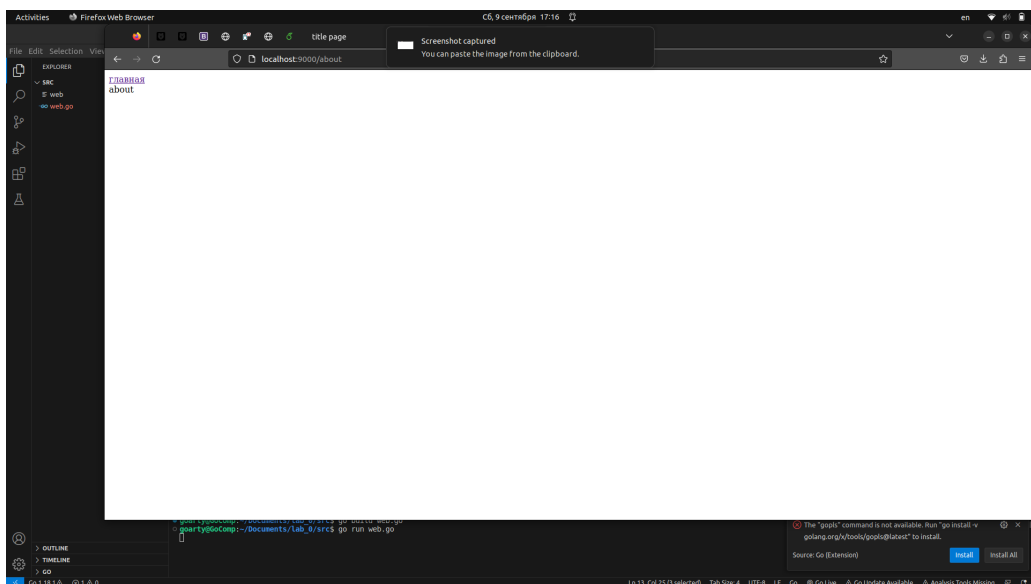


Рис. 2 — Реализация страницы about

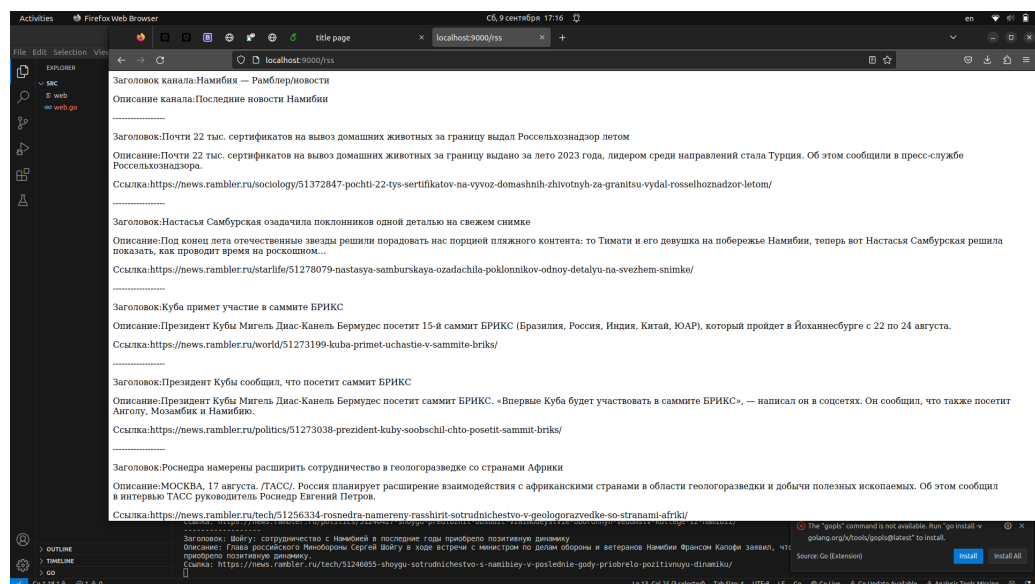


Рис. 3 — Реализация страницы rss