



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 информационной системы сбора данных из
блокчейна Ethereum и сохранение их в Firebase Realtime
Database»

Студент группы ИУ9-32Б Лавров Р. Д.

Преподаватель Посевин Д. П.

Москва 2025

1 Задание

Реализовать информационную систему, которая будет собирать данные из блокчейна Ethereum и сохранять их в Firebase Realtime Database. Для этого необходимо использовать язык программирования Go и соответствующие библиотеки для работы с Ethereum и Firebase.

2 Результаты

Исходный код 1.

Листинг 1: Пример реализации класса

```
1 package main
2
3 import (
4     "context"
5     "fmt"
6     "log"
7     "math/big"
8
9     "github.com/ethereum/go-ethereum/ethclient"
10
11     firebase "firebase.google.com/go"
12     "firebase.google.com/go/db"
13     "google.golang.org/api/option"
14 )
15
16 type FireDB struct {
17     *db.Client
18 }
19
20 var fireDB FireDB
21
22 func (db *FireDB) Connect() error {
23     ctx := context.Background()
24     opt := option.WithCredentialsFile("bmstu-sem2-firebase-adminsdk-fbsvc-159e4e53ed.json")
25     config := &firebase.Config{DatabaseURL: "https://bmstu-sem2-default-rtdb.firebaseio.com/"}
26     app, err := firebase.NewApp(ctx, config, opt)
27     if err != nil {
28         return fmt.Errorf("error initializing app: %v", err)
29     }
30     client, err := app.Database(ctx)
31     if err != nil {
```

```

32     return fmt.Errorf("error initializing database: %v", err)
33 }
34 db.Client = client
35 return nil
36 }
37
38 func FirebaseDB() *FireDB {
39     return &fireDB
40 }
41
42 func main() {
43     ctx := context.Background()
44
45     client, err := ethclient.Dial("https://mainnet.infura.io/v3/6
        f1c6a8fd16b4d9ba6d2565dc3c47920")
46     if err != nil {
47         log.Fatalln(err)
48     }
49
50     header, err := client.HeaderByNumber(ctx, nil)
51     if err != nil {
52         log.Fatal(err)
53     }
54
55     blockNumber := big.NewInt(header.Number.Int64())
56     block, err := client.BlockByNumber(ctx, blockNumber)
57     if err != nil {
58         log.Fatal(err)
59     }
60
61     err = FirebaseDB().Connect()
62     if err != nil {
63         log.Println(err)
64         return
65     }
66
67     var txData map[string]interface{}
68
69     for _, tx := range block.Transactions() {
70         txData = map[string]interface{}{
71             "chainId": tx.ChainId().String(),
72             "hash":    tx.Hash().Hex(),
73             "value":   tx.Value().String(),
74             "cost":    tx.Cost().String(),
75             "to":      tx.To().String(),
76             "gas":     tx.Gas(),

```

```

77     "gasPrice": tx.GasPrice().String(),
78     "block":    block.Number().Uint64(),
79 }
80 break
81 }
82
83 ref := fireDB.NewRef("lastTransaction")
84
85 if err := ref.Set(ctx, txData); err != nil {
86     log.Fatalf("Firebase write error: %v", err)
87 }
88 fmt.Println(txData)
89 }

```

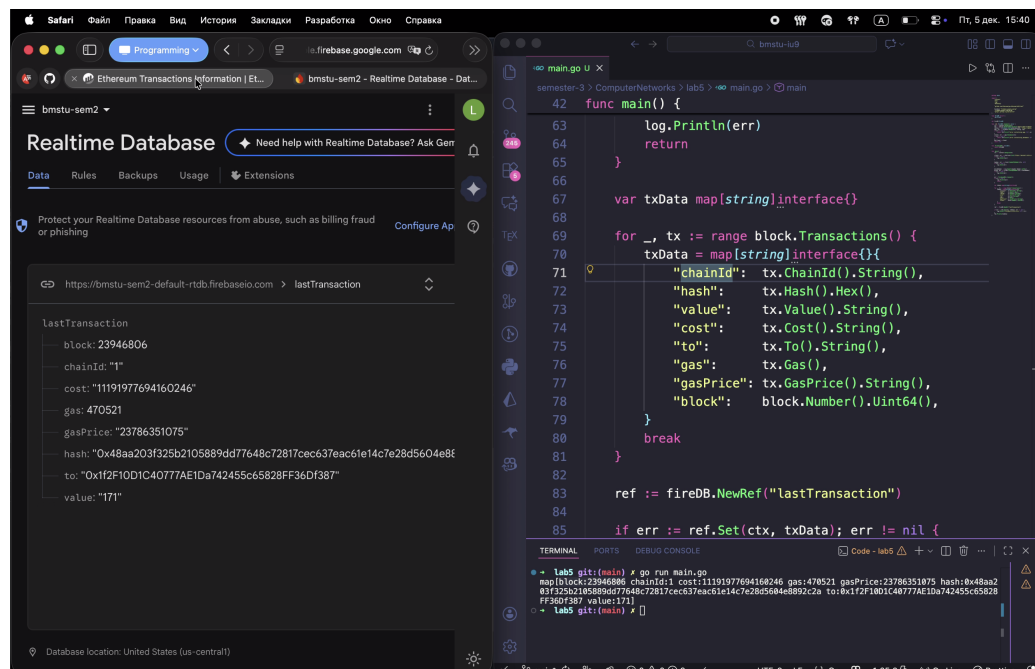


Рис. 1 — Результат работы

3 Вывод

В ходе выполнения лабораторной работы была реализована информационная система, которая собирает данные из блокчейна Ethereum и сохраняет их в Firebase Realtime Database с использованием языка программирования Go и соответствующих библиотек для работы с Ethereum и Firebase.