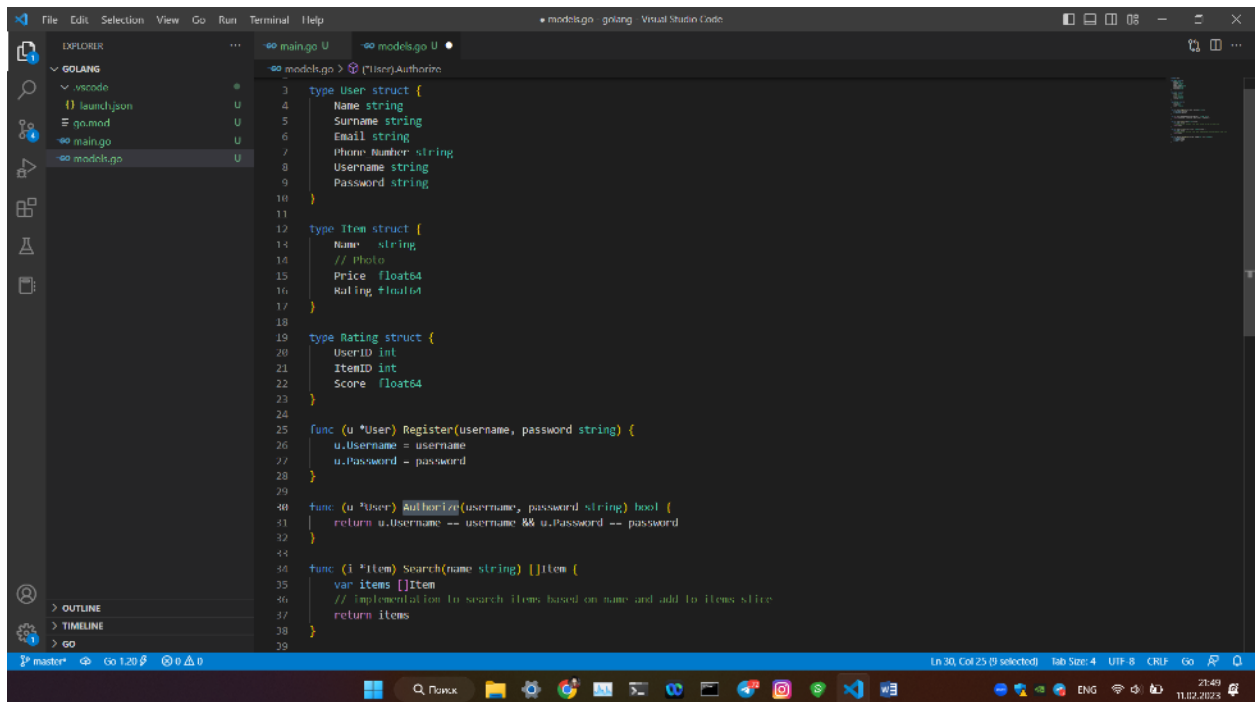
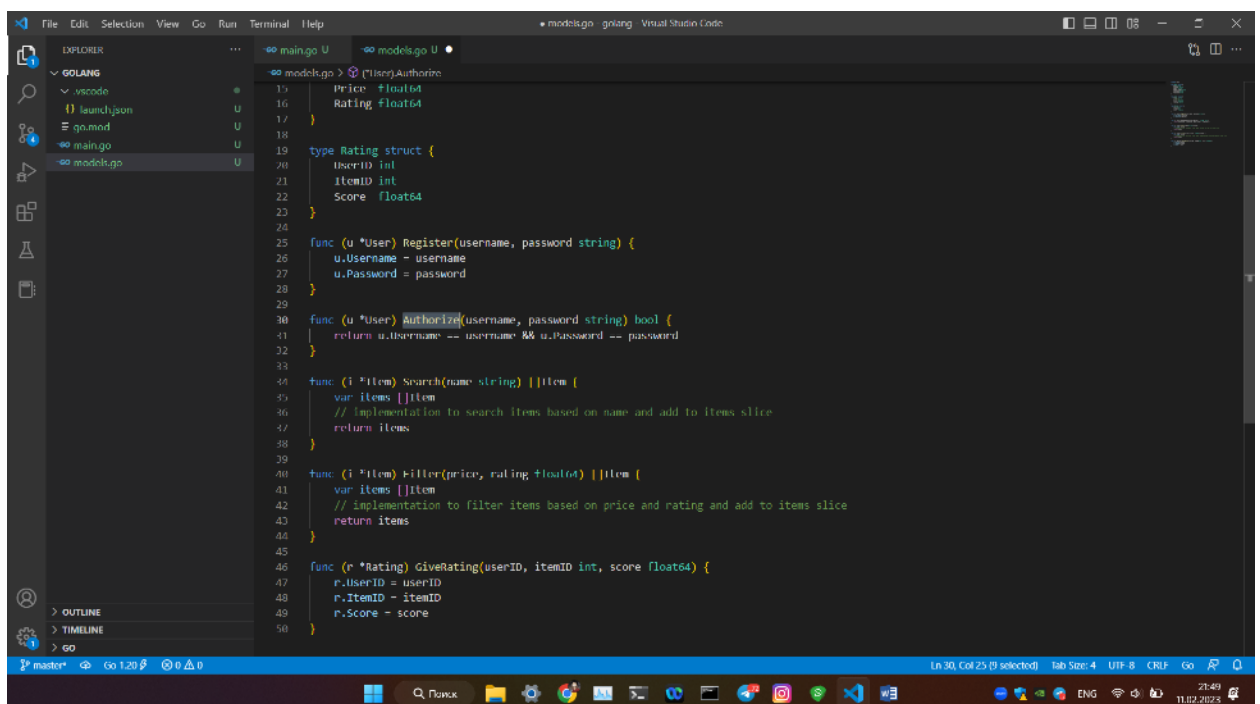


Based on requirements we added classes User, Item, Rating, func to Registration, func to Authorization, and Searching function. Every User has fields of name, surname, email, phone number and login, password. Fields of other classes you can see in this picture. Next, we are going to complete searching functions and add database.



```
1 type User struct {
2     Name string
3     Surname string
4     Email string
5     Phone Number string
6     Username string
7     Password string
8 }
9
10
11
12 type Item struct {
13     Name string
14     // Photo
15     Price float64
16     Rating float64
17 }
18
19 type Rating struct {
20     UserID int
21     ItemID int
22     Score float64
23 }
24
25 func (u *User) Register(username, password string) {
26     u.Username = username
27     u.Password = password
28 }
29
30 func (u *User) Authorize(username, password string) bool {
31     return u.Username == username && u.Password == password
32 }
33
34 func (i *Item) Search(name string) []Item {
35     var items []Item
36     // implementation to search items based on name and add to items slice
37     return items
38 }
39
```



```
15     Price float64
16     Rating float64
17 }
18
19 type Rating struct {
20     UserID int
21     ItemID int
22     Score float64
23 }
24
25 func (u *User) Register(username, password string) {
26     u.Username = username
27     u.Password = password
28 }
29
30 func (u *User) Authorize(username, password string) bool {
31     return u.Username == username && u.Password == password
32 }
33
34 func (i *Item) Search(name string) []Item {
35     var items []Item
36     // implementation to search items based on name and add to items slice
37     return items
38 }
39
40 func (i *Item) Filter(price, rating float64) []Item {
41     var items []Item
42     // implementation to filter items based on price and rating and add to items slice
43     return items
44 }
45
46 func (r *Rating) GiveRating(userID, itemID int, score float64) {
47     r.UserID = userID
48     r.ItemID = itemID
49     r.Score = score
50 }
```