



Team: NDY corp.

Progress Report 5

Date: March 12, 2023

		
Amangeldin Erasyly	Amanzhol Nursultan	Shulanbay Darkhan
id:200103252	id:200103444	id:200103038

We have refactored so that we can move on more effectively. Our application had two main problems, the first was the structure of our API server, since it was sufficiently closed and all the initialization of all dependencies took place inside the structure itself, so we did not have a normal opportunity to replace some dependencies in tests and so on.

And we ran our two tests “userrepository_test.go” and they passed successfully.

```
20 s := teststore.New()
PROBLEMS 2 OUTPUT TERMINAL GITLENS DEBUG CONSOLE Test Output at 21:14:39 + v
Running tool: C:\Program Files\Go\bin\go.exe test -timeout 30s -run ^TestUserRepository_Create$ github.com/ndy-corp/1.src/midterm-1/src-code/internal/app/store
=== RUN TestUserRepository_Create
--- PASS: TestUserRepository_Create (0.00s)
PASS
ok      github.com/ndy-corp/1.src/midterm-1/src-code/internal/app/store/teststore    0.185s

> Test run finished at 3/12/2023, 9:14:41 PM <
Ln 4, Col 14 Tab Size: 4 UTF-8 CRLF Go Go Live Go Upd
```

```
PROBLEMS 2 OUTPUT TERMINAL GITLENS DEBUG CONSOLE Test Output at 21:14:39 + v
> Test run finished at 3/12/2023, 9:14:41 PM <
Running tool: C:\Program Files\Go\bin\go.exe test -timeout 30s -run ^TestUserRepository_FindByEmail$ github.com/ndy-corp/1.src/midterm-1/src-code/internal/app/store/teststore
=== RUN TestUserRepository_FindByEmail
--- PASS: TestUserRepository_FindByEmail (0.00s)
PASS
ok      github.com/ndy-corp/1.src/midterm-1/src-code/internal/app/store/teststore    0.053s

> Test run finished at 3/12/2023, 9:15:13 PM <
Ln 4, Col 14 Tab Size: 4 UTF-8 CRLF Go Go Live Go Upd
```

We have created a special type of error that will be in the store package and it will be a common error for all implementations.

```
1 package store
2
3 import "errors"
4
5 var (
6     // ErrRecordNotFound ...
7     ErrRecordNotFound = errors.New("record not found")
8 )
9
```

Then we created "server_internal_test.go". We have an http request here, and we have announced our server. Then we called ServerHTTP and we passed the recorder there and then requested it. Then we checked our status, which we recorded in recorder, which is equal to the ok status. After running our test, our server is running and everything is fine.

```
1 package apiserver
2
3 import (
4     "net/http"
5     "net/http/httptest"
6     "testing"
7
8     "github.com/ndy-corp/1.src/midterm-1/src-code/internal/app/store/tests"
9     "github.com/stretchr/testify/assert"
10 )
11
12 run test | debug test
13 func TestServer_HandleUsersCreate(t *testing.T) {
14     rec := httptest.NewRecorder()
15     req, _ := http.NewRequest(http.MethodPost, "/users", nil)
16     s := newServer(teststore.New())
17     s.ServeHTTP(rec, req)
18     assert.Equal(t, http.StatusOK, rec.Code)
19 }
```

PROBLEMS 2 OUTPUT TERMINAL GITLENS DEBUG CONSOLE Go Tests

Running tool: C:\Program Files\Go\bin\go.exe test -timeout 30s -run
^TestServer_HandleUsersCreate\$ github.com/ndy-corp/1.src/midterm-1/src-code/internal/app/apiserver

ok github.com/ndy-corp/1.src/midterm-1/src-code/internal/app/apiserver (cached)