

## Test Description

In ./docs directory, there are some sample Test Files generated by running program in Edlab. In this document, I will describe HTTP requests that I have tested and explain them.

Table1. Test File

Input File	Command List File	"/docs/edlab_test_client_command_list.csv"
	Book Data File	"/docs/edlab_test_book_data.csv"
Output File	Book Data File	"/docs/edlab_test_book_data.csv"
	Client Log File	"/docs/edlab_test_client_log_file.csv"
	FrontEnd Log File	"/docs/edlab_test_frontend_log_file.csv"
	Catalog Log File	"/docs/edlab_test_catelog_log_file.csv"
	Order Log File	"/docs/edlab_test_order_log_file.csv"

Book Data File [after the test]:

Every line represents a book data. For each line, from left to right, each element means: itemNumber, name, topic, stock, cost

```
1 "0","How to get a good grade in 677 in 20 minutes a day","distributed systems","1497","12"
2 "1","RPCs for Dummies","distributed systems","1443","22"
3 "2","Xen and the Art of Surviving Graduate School","graduate school","0","33"
4 "3","Cooking for the Impatient Graduate Student","graduate school","0","42"
```

Command List File:

Client will read command list file, and send these HTTP requests to FrontEndService.

```
1 /search/distributed systems
2 /search/graduate school
3 /lookup/0
4 /lookup/1
5 /lookup/2
6 /lookup/3
7 /buy/0
8 /buy/1
9 /buy/2
10 /buy/3
```

After running the test, Book Data File should be changed. And Client, FrontEndServer, CatalogServer, OrderServer will generate Log File separately.

Client Log File:

Client Log File records every HTTP request it sends and their respond. We can verify the test by reading client's request and its respond. One thing worth to be mentioned is that, the Client succeed to buy item 0 and 1. But fail to buy item 2 and 3. The reason is that item 2 and 3 are run out of stock. We can see the stock of item 2 and 3 are 0 on Book Data File.

```
1 "Run command:http://128.119.243.164:5018/search/distributed%20systems"
2 "Result:{"itemNumber":"0","name":"How to get a good grade in 677 in 20 minutes a day","topic":"distributed systems","stock":1497,"cost":12}"
3 "Run command:http://128.119.243.164:5018/search/graduate%20school"
4 "Result:{"itemNumber":"2","name":"Xen and the Art of Surviving Graduate School","topic":"graduate school","stock":0,"cost":33}"
5 "Run command:http://128.119.243.164:5018/lookup/0"
6 "Result:{"itemNumber":"0","name":"How to get a good grade in 677 in 20 minutes a day","topic":"distributed systems","stock":1497,"cost":12}"
7 "Run command:http://128.119.243.164:5018/lookup/1"
8 "Result:{"itemNumber":"1","name":"RPCs for Dummies","topic":"distributed systems","stock":1443,"cost":22}"
9 "Run command:http://128.119.243.164:5018/lookup/2"
10 "Result:{"itemNumber":"2","name":"Xen and the Art of Surviving Graduate School","topic":"graduate school","stock":0,"cost":33}"
11 "Run command:http://128.119.243.164:5018/lookup/3"
12 "Result:{"itemNumber":"3","name":"Cooking for the Impatient Graduate Student","topic":"graduate school","stock":0,"cost":42}"
13 "Run command:http://128.119.243.164:5018/buy/0"
14 "Result:{"success":true}"
15 "Run command:http://128.119.243.164:5018/buy/1"
16 "Result:{"success":true}"
17 "Run command:http://128.119.243.164:5018/buy/2"
18 "Result:{"success":false}"
19 "Run command:http://128.119.243.164:5018/buy/3"
20 "Result:{"success":false}"
21 "Command List Tested"
```

OrderServer Log File:

Client will not send “reload” signal to increase the stock of books. So we let OrderServer to send “reload” request to CatalogServer. Also, OrderServer will record every “buy” request, no matter it success or not.

1	"reload item 0 5 more success"
2	"buy 0 success"
3	"buy 1 success"
4	"buy 2 fail"
5	"buy 3 fail"

Catalog Log File:

1	"reload item 0 5 more", " 1498 left"
2	"query subject distributed systems success"
3	"query subject graduate school success"
4	"query item 0 success"
5	"query item 1 success"
6	"query item 2 success"
7	"query item 3 success"
8	"request item 0 success", " 1497 left"
9	"request item 1 success", " 1443 left"
10	"request item 2 fail", " 0 left"
11	"request item 3 fail", " 0 left"