

## CHAPTER V

### IMPLEMENTATION AND TESTING

#### 5.1 IMPLEMENTATION

This quiz using java socket is created so that can see the communication between client and server.

So first we need to start the server via terminal, the port address already define on program so server just need to be start with: java Server

```
[edwinsan@edwinsan ~]$ cd /home/edwinsan/
[edwinsan@edwinsan FIX]$ java Server
MiniServer active 8080
```

Figure 10. Server active

and can be seen here that server use port 8080 and already waiting the connection from client .

Next on client side, first open the browser and type server's IP address and set the port too like this <http://192.168.41.100:8080> and after port number type '/menu' so we can go to menu page, example



Figure 11. Client side

After that the menu page will appear .

In server side can be seen the process of split the line from client's request like this.

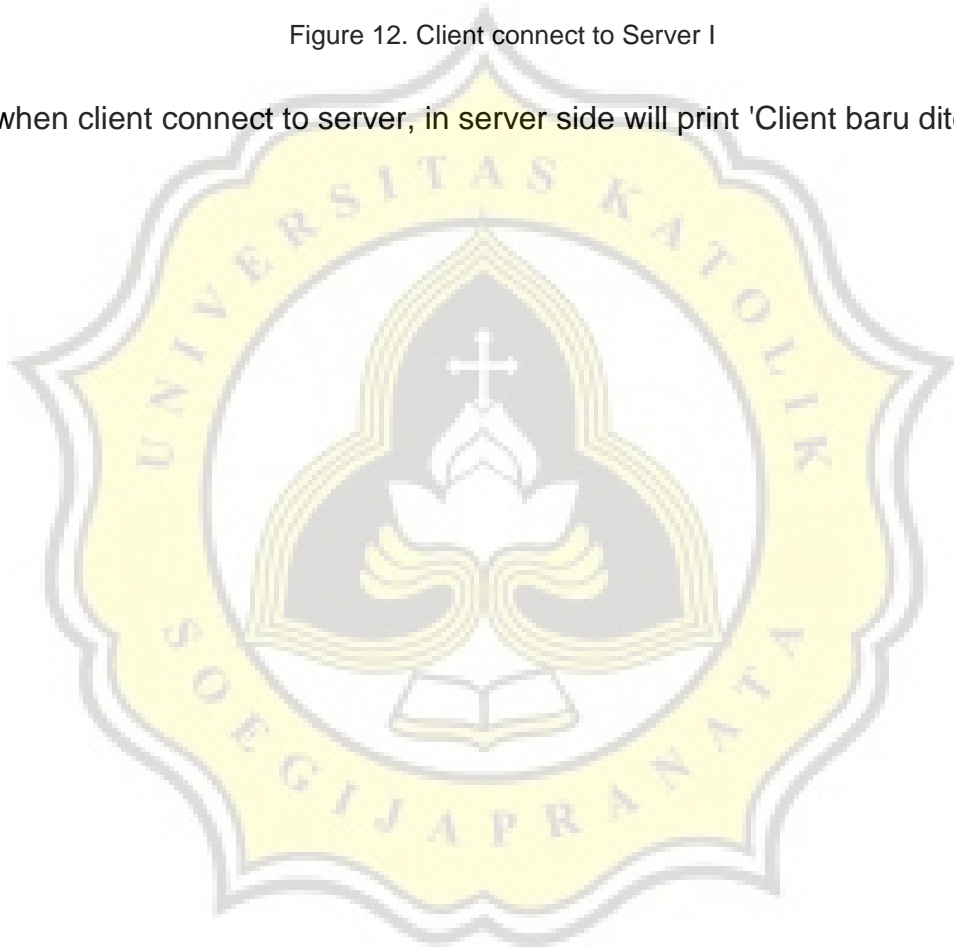
```
[edwinsan@edwinsan FIX]$ java Server
MiniServer active 8080

Client baru diterima.

Abaikan
GET /menu HTTP/1.1
/menu
```

Figure 12. Client connect to Server I

when client connect to server, in server side will print 'Client baru diterima'.



## 5.2 TESTING

Now after some implementation then start on testing.

```
[edwinsan@edwinsan FIX]$ java Server
MiniServer active 8080

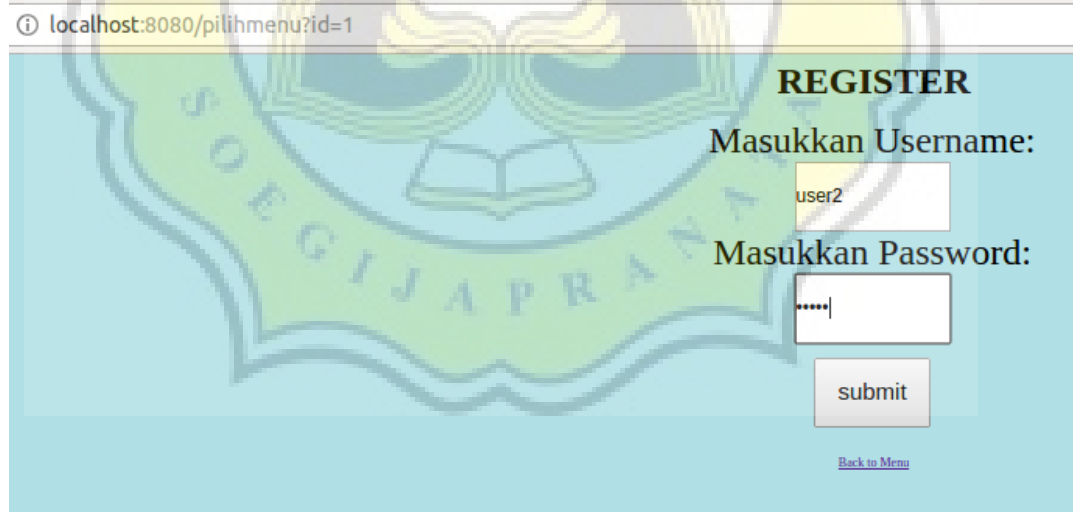
Client baru diterima.

Abaikan
GET /menu HTTP/1.1
/menu
```

Figure 13. Client connect to Server II

when client request to the server, it give some line like GET /menu HTTP/1.1 and server process it to reply with menu page. If client request to server which not suitable with protocol, then server do nothing .

After that, client can input '1' so client can register username. After type '1' form Register will appear



localhost:8080/pilihmenu?id=1

**REGISTER**

Masukkan Username:

Masukkan Password:

[Back to Menu](#)

Figure 14. Register form

localhost:8080/menu?id=1 its mean that client request register form with input '1' in the menu page .

Then client input or set username and password and after that will appear text like this.

Register berhasil silahkan klik [disini](#)

Figure 15. Register Success

Its mean that client success to register the username and then click 'disini' to go to first menu for login.

In server side, output can be seen like this

```
GET /REGISTER?username=yey&password=yey HTTP/1.1
/REGISTER
yey
yey
```

Figure 16. Register Success Server side

Server record the username and password into database so client can login with that username.

Back again on client side, client input menu '2' to login and form login will appear

localhost:8080/pilihmenu?id=2

**LOGIN**

Masukkan Username:

Masukkan Password:

[Back to Menu](#)

Figure 17. Login Form

Then client input username and password which already registered. If username and password not match, client will redirect to menu page again but if username and password are suitable with database, server will give the quiz page to client.



Figure 18. Quiz page

The page of quiz will appear like that and then client answer on the blank textfield. After answer the first quiz, the second quiz will appear until fifth quiz and quiz will appear one by one so client will get more concentration on answering the quiz, and every user will get different question.

In Server side will look like this

```
GET /SOAL?user=yey&id=1&jawaban=25 HTTP/1.1
/SOAL
yey
1
25
```

Figure 19. Server side

Server will record client's username, question's id and client's answer in to the database.

And then after client answer all quiz, server will checks client's answer with real answer, if same then client will get score 20 in every question.

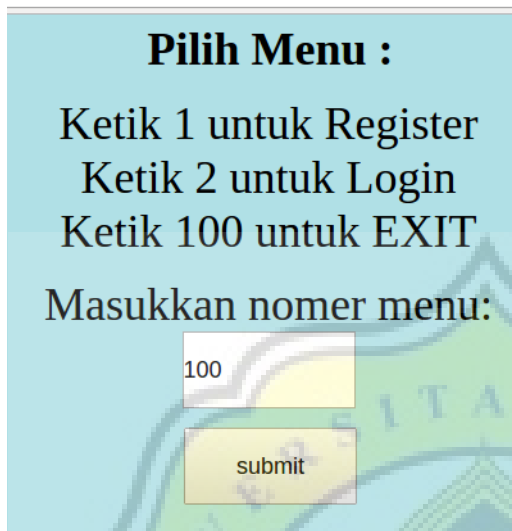
After the server checks the answer, then server will send the score with the question, real answer and user's answer so client can do some correction by themselves and the page will look like this.



Figure 20. Client get score

Then client should click 'disini' if client will play some quiz again or for exit.

For exit, client can input '100' on menu page and then server stop to listen on this client



**Pilih Menu :**

Ketik 1 untuk Register  
 Ketik 2 untuk Login  
 Ketik 100 untuk EXIT

Masukkan nomer menu:



Figure 21. Client input '100' for exit

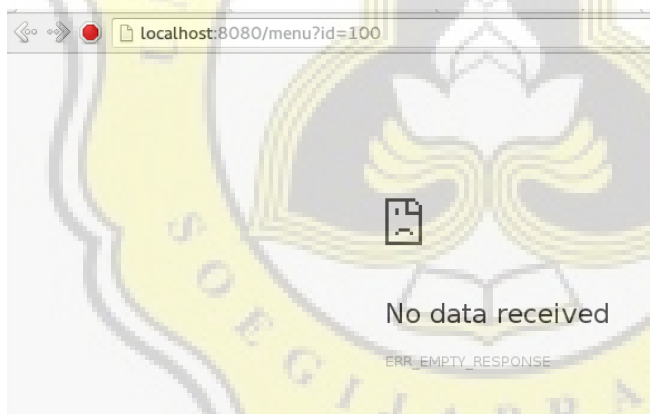


Figure 22. Client success to exit

And on Server side when client input '100' server will give notifications 'User exit' and server will end the thread on this client.

```
GET /menu?id=100 HTTP/1.1
/menu
100

User exit!
```

Figure 23. Client exit(Server side)