**BTEC Level 5 HND Computing and Systems Development**

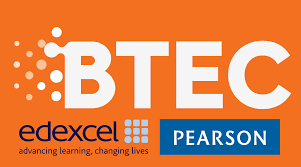


A Project Distributed Software Applications Presented

by

Nguyễn Đình Tùng (GC00952)

Assignment 2



Submitted to the Graduate School of the

University of Greenwich Vietnam in partial fulfillment

Wednesday, 20th June 2018

Lecturer: Đỗ Quốc Bình

**Table of Contents**

[**Introduction** 5](#_Toc523989276)

[**Task 1: (L03 3.1, 3.2, 3.3, 3.4, M1, M3, D1, and D3) Software Implementation Stage** 6](#_Toc523989277)

[1.1: Implement the components and data and file structures required to implement a given design 6](#_Toc523989278)

[1.2:Define relationships between components to implement design requirements 19](#_Toc523989279)

[1.3: Identify and implement opportunities for error handling and reporting 20](#_Toc523989280)

[1.4: Make effective use of an Integrated Development Environment (IDE) including code and screen templates 22](#_Toc523989281)

[**Task 2: (L04 4.1, 4.2, 4.3, D1 and D2) Software Testing Stage and Software Evaluation Stage** 26](#_Toc523989282)

[2.1: Critically review and test your solution 26](#_Toc523989283)

[2.2: Analysis actual test results against expected results to identify discrepancies 29](#_Toc523989284)

[2.3: Evaluate independent feedback with recommendations for improvement 33](#_Toc523989285)

[**Task 3: (L04 4.4, 4.5 M3) Software Documentation Stage (User guide & Technical)** 37](#_Toc523989286)

[2.4: Create user documentation for the developed application 37](#_Toc523989287)

[2.5: Create documentation for the support and maintenance of your application 39](#_Toc523989288)

**Table of figures**

[Figure 1. Wireframe register 9](#_Toc524080376)

[Figure 2. Wireframe login 10](#_Toc524080377)

[Figure 3. Wireframe view all products 11](#_Toc524080378)

[Figure 4. Wireframe reserve product 14](#_Toc524080379)

[Figure 5. View products reserved 17](#_Toc524080380)

[Figure 6. Wireframe view products hired 18](#_Toc524080381)

[Figure 7. Relationships between components 19](#_Toc524080382)

[Figure 8. Error handing 21](#_Toc524080383)

[Figure 9. Check null 22](#_Toc524080384)

[Figure 10. Check quantity in stock 22](#_Toc524080385)

[Figure 11. Language tool 23](#_Toc524080386)

[Figure 12. Invoking code completion 23](#_Toc524080387)

[Figure 13. Smart suggestions at the top 23](#_Toc524080388)

[Figure 14. Camel case completion 24](#_Toc524080389)

[Figure 15. Completing keywords 24](#_Toc524080390)

[Figure 16. Suggesting names for variable and fields 24](#_Toc524080391)

[Figure 17. Suggesting parameters 24](#_Toc524080392)

[Figure 18. Common prefix completion 25](#_Toc524080393)

[Figure 19. Subword completion 25](#_Toc524080394)

[Figure 20. Register of customers 26](#_Toc524080395)

[Figure 21. Login to system 27](#_Toc524080396)

[Figure 22.View all products from customers 27](#_Toc524080397)

[Figure 23. Customers reserve product 27](#_Toc524080398)

[Figure 24. Customers hire product 28](#_Toc524080399)

[Figure 25. Customers return product 28](#_Toc524080400)

[Figure 26. Staffs add new product 28](#_Toc524080401)

[Figure 27. Staffs view report in duration they want 29](#_Toc524080402)

[Figure 28. Staffs management account of customers 29](#_Toc524080403)

[Figure 29. First questionnaire 34](#_Toc524080404)

[Figure 30. Second questionnaire 34](#_Toc524080405)

[Figure 31. Third questionnaire 35](#_Toc524080406)

[Figure 32. Fourth questionnaire 35](#_Toc524080407)

[Figure 33. how to go to page user guide 37](#_Toc524080408)

[Figure 34. Screen register 37](#_Toc524080409)

[Figure 35. Check enter wrong 38](#_Toc524080410)

[Figure 36. Reserve 38](#_Toc524080411)

[Figure 37. Hire product 38](#_Toc524080412)

[Figure 38. Return product 39](#_Toc524080413)

**Table of tables**

[Table 1. Table test case for system. 33](#_Toc524080414)

[Table 2. Result and solution for questionnaire 36](#_Toc524080415)

# **Introduction**

This report I will write about implement the components and data and file structures required to implement a given design, next is define relationships between components to implement design requirements, next is identify and implement opportunities for error handling and reporting and make effective use of an Integrated Development Environment (IDE) including code and screen templates, critically review and test your solution, analyze actual test results against expected results to identify discrepancies, evaluate independent feedback with recommendations for improvements, create user documentation for the developed application and finally is create documentation for the support and maintenance of your application.

# **Task 1: (L03 3.1, 3.2, 3.3, 3.4, M1, M3, D1, and D3) Software Implementation Stage**

## 1.1: Implement the components and data and file structures required to implement a given design

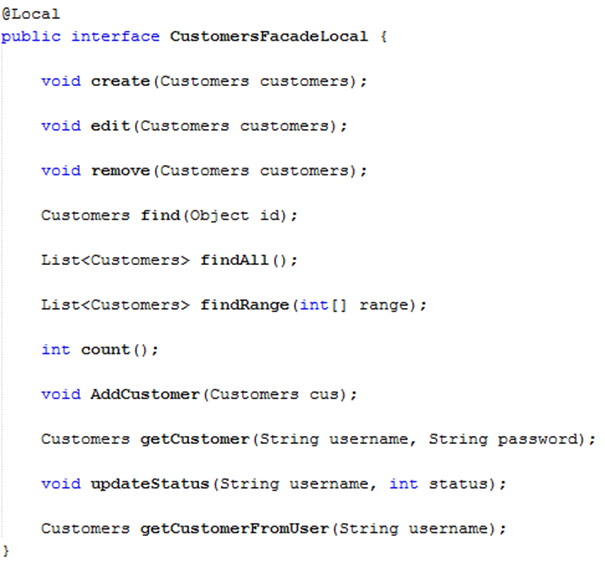
With requirements of company ABC this is customer can reserve cameras and lens, customer can hire cameras and lens based on reservation or at store and customer can return hired cameras and lens. I designed main functions of program as follows, continue customers can login to system store online, next Customers can view all products and can reserve a product (cameras or lens), Next Customers can view all products that they reserved and customers can hire these products, finally customers can return products that they hired. Below I will give code snippet for each function and give wireframes to you can easily visualize them.

1. **Register from customers**

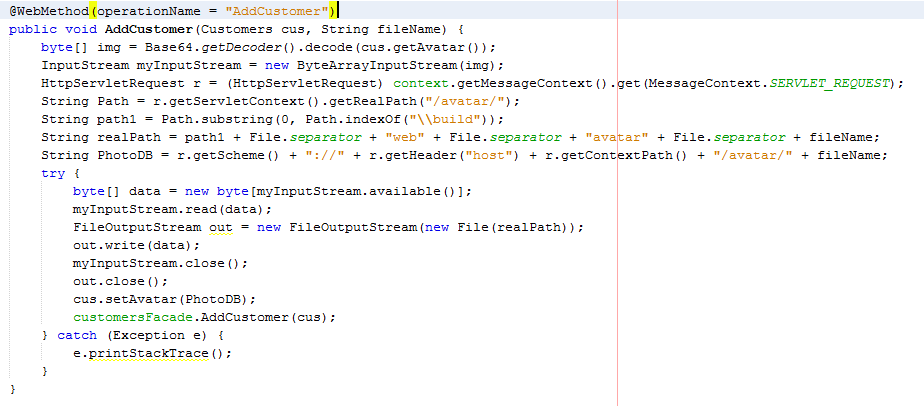
First, we will create function register on classes CustomersFacade, at this function we implement manipulate with database to can save information to database as image below



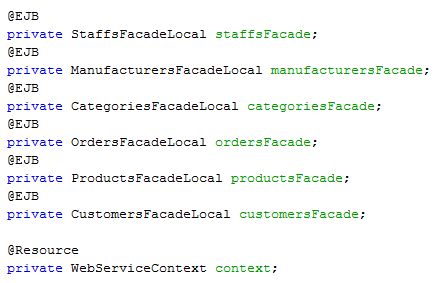
Continue we will create a function at CustomersFacadeLocal same with function register in class CustomersFacade but it only is interface to can call them at class web service. When we created anything function in class CustomersFacade we also must create interface for it in class CustomersFacadeLocal to can call them in class web service as image below



Continue at class web server is DAPD1525Service we will create function register as image below



At this function we implement write code to receive parameters this is a Customers and a fileName from client-side and call function manipulate with database to save this information to database. We passing a fileName is name of image and we save this image to folder in project, this folder is “avatar” to save image from client-side send on. We convert this image to byte and save folder “avatar” as image above and save address localhost of image in database. After we call function just created at class CustomersFacadeLocal to save them to database. Below is how to call FacadeLocals in this project



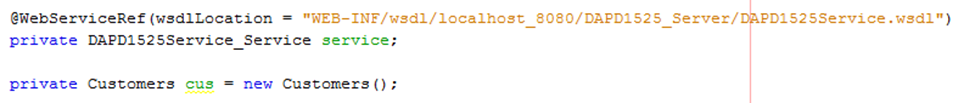
At Client-side. First we create interface for user this is a JSF page. It includes some tags inputText to help user enter data, a tag choose file image to help user choose image they want set up avatar and button “Register” to after they enter full data they will implement manipulate register account, at this button we have attribute actionListener to call function from controller and implement this function as image below



At class CustomersManagement mean is class controller we write code for function register as image below



At this function we implement add data to Customer and passing this Customer to server by call service as image below



**This is model wireframe to simulation to users**

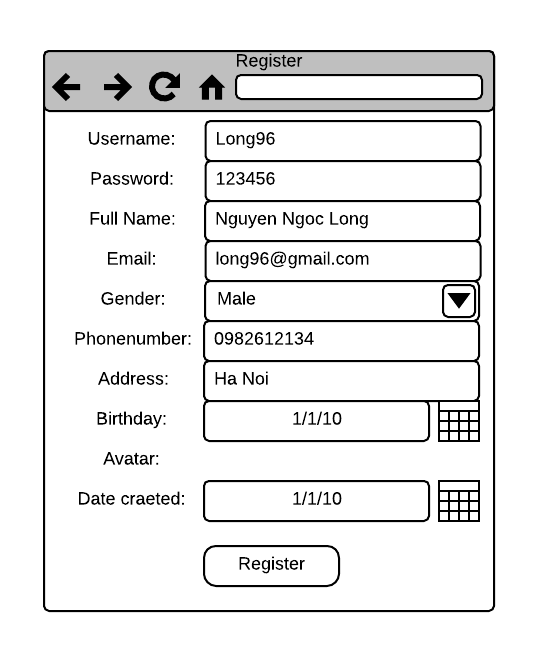
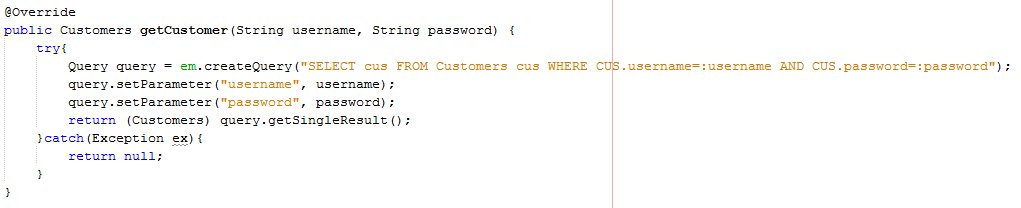


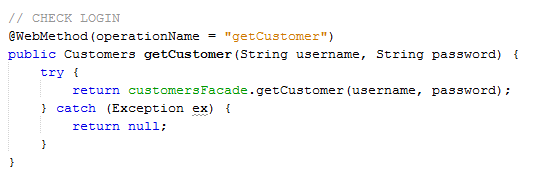
Figure 1. Wireframe register

1. **Login to system store online**

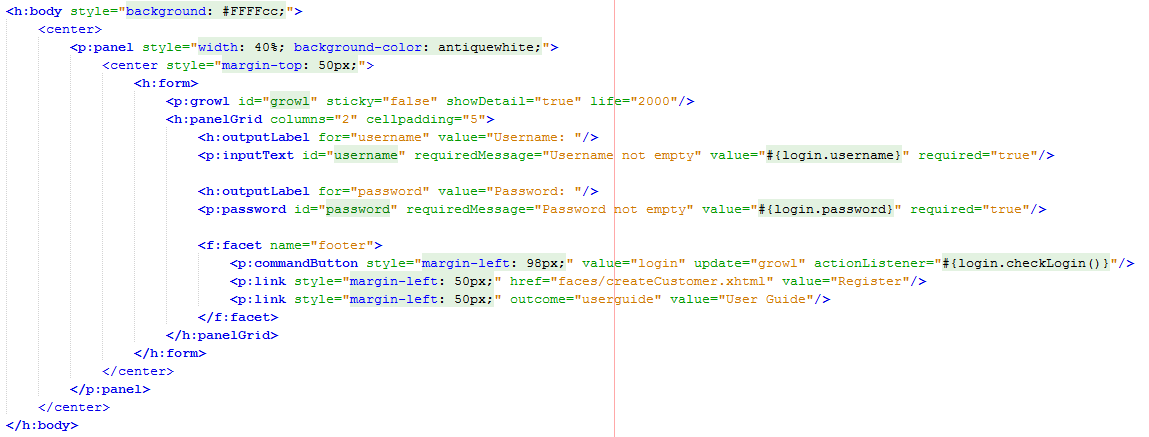
Same with function register process work of system also same like that. First is create function to get data Username and Password in database as image below



Continue at class DAPD1525Service we create a function, this function passing username and password from client-side and we call function get data from database through username and password just passed and return a Customer as image below



At client-side we create a JSF page login and requirement user enter username and password and set up an event for button “Login” this is actionListener checkLogin() as image below



At class controller we created a function name is checkLogin() to implement check if username and password not exist already we will info error and end program, if username and password exist already we will continue check username and password depend account of customers or staffs and implement move other interfaces as image below



**This is wireframe for function login**

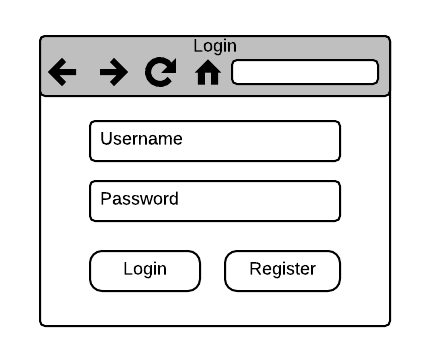
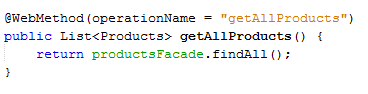


Figure 2. Wireframe login

1. **View all products**

Because we use technology Entity Framework support for us very many about many aspect, so we easy write code and faster in project implementation as current we don’t need write code to get data in table Products form database instead we just call this feature in class web service and return a list products as image below



At Client-side we only write code to call this function from server-side and display data to users as image below



**This is wireframe for view all products**

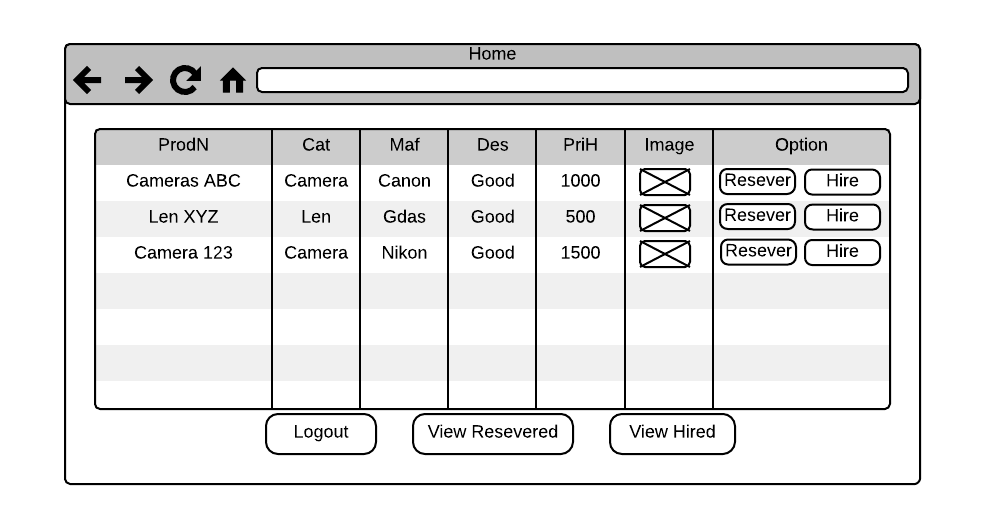
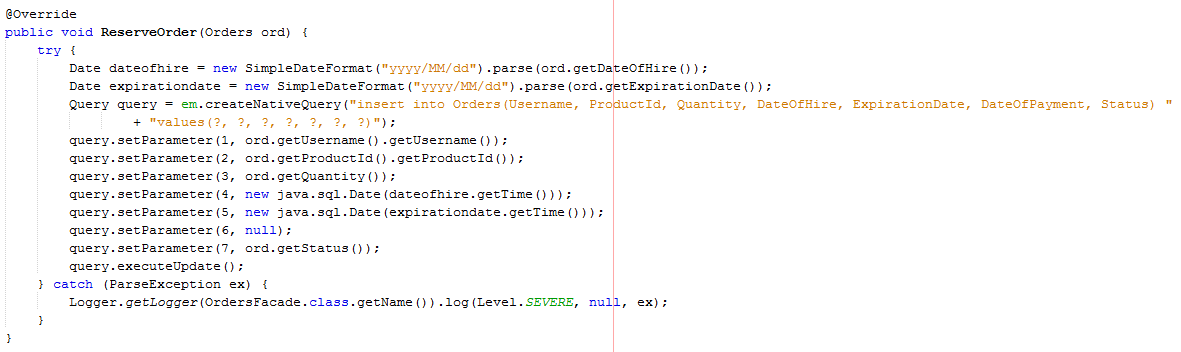


Figure 3. Wireframe view all products

1. **Reserve products**

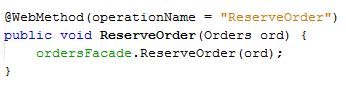
Same with function register. First we write to manipulate with database mean is insert information from client-side to database as image below



And this is class OrdersFacadeLocal to storage interface be created in class OrdersFacade to we can call them in class web service DAPD1525Service as image below



Continue at class web service we write a function to receive from client-side and call interface just created to save information of order to database as image below



At client-side we created a JSF page to users can enter quantity that they want hire and requirement they choose date of hire mean is day they want hire product and choose day they want end hire product and a button to user click to requirement system implement process reserve product as image below



At this button we have set up event actionListener reserve() to call function from controller and implement check information and passing information to server. We checked if quantity equal 0 system will info error if users want hire this product, if users enter quantity want hire more than quantity in stock system will info error to users and check date of hire, expiration date is not less more date now. Outside we also check expiration date is not less more date of hire, if satisfies the above requirements system will send information order to server and server implement save data to database and info to success users as image below



Same with function reserve product, the functions hire product and return product just update order and change status, change quantity of product in stock.

**This is wireframe for reserve**

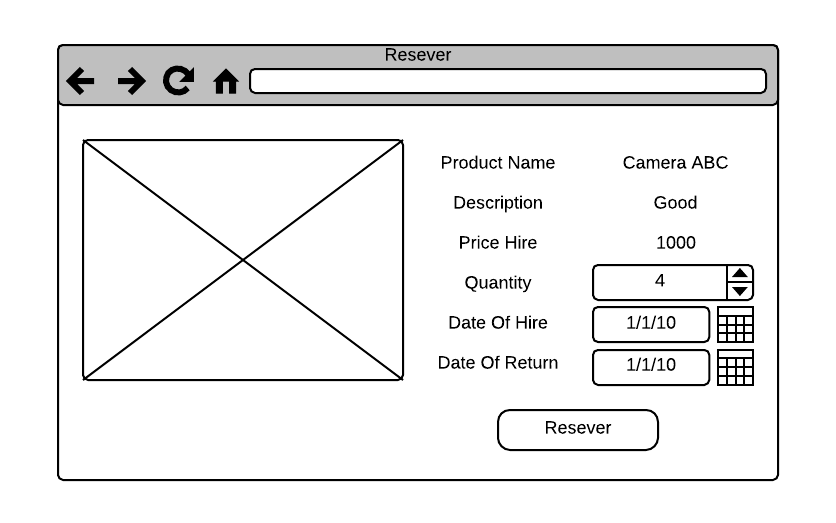
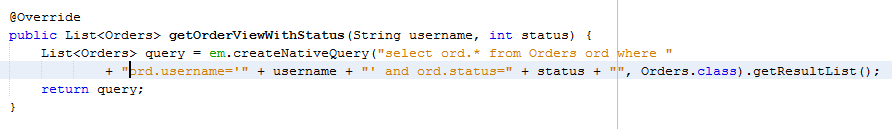


Figure 4. Wireframe reserve product

1. **View all products that customer reserved and hire products reserved**

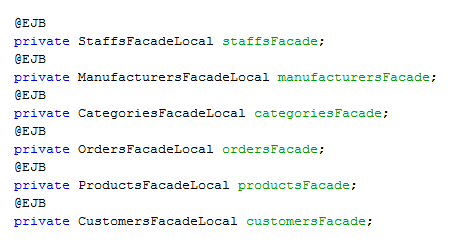
Same with function view all product. First we get data from database in class OrdersFacade depend package session with condition is username must like with username of user are logging and status of that products are reserved status as image below



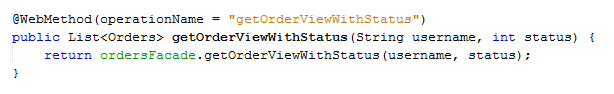
Continue step is create interface like with function just create in class interface OrdersFacadeLocal to we can call this function at class service as image below



Continue step at class web service DAPD1525Service we will call class interface to can use functions have in that classes as image below to from that that classes interface we will create function at here and call function created from that classes



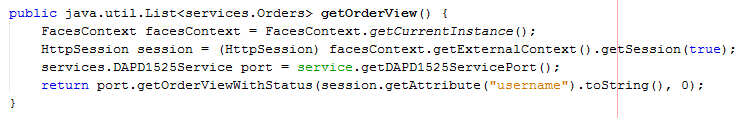
Continue step we will write code create a new function at class web service DAPD1525Service to return list order that this customer reserved use function we just create to manipulate with database with condition is passing two parameters username and status as image below



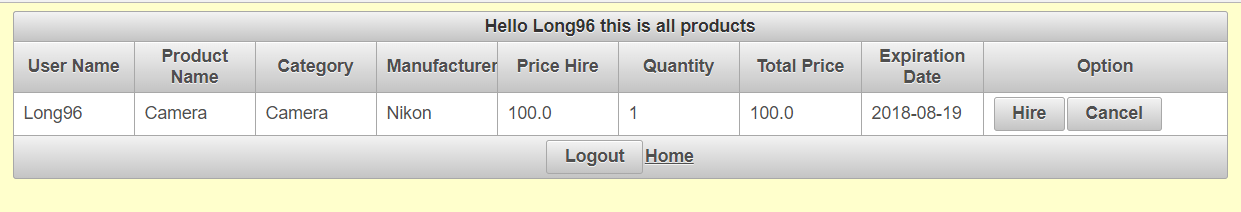
At client-side we created controller JSF Manage bean to call web service as image below



After we call service complete we will create function get all product that this customer reserved and implement call functions from web service and passing parameters as image below

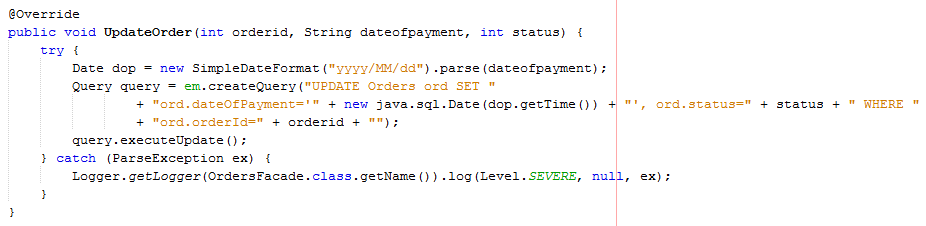


Continue step we created JSF page to display information got from database as image below

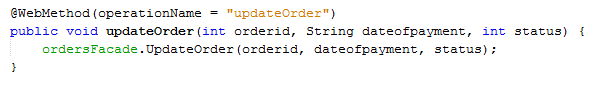


At this page customers can hire products that they reserved or cancel them

Continue to hire product we need write code from server-side and client as follow. To implement function hire product of customers we need update status of this orders. So we need write function in class OrdersFacade to update status of this orders as image below



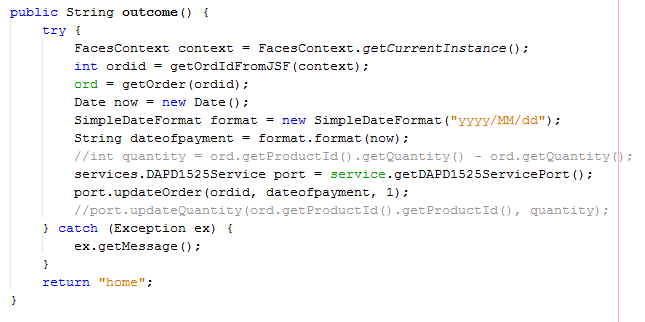
Continue at class web service DAPD1525Service we will write function update order to call function just create at classes interfaces and implement passing parameters for function requirement passing parameters to manipulate with database as image below



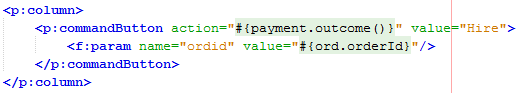
At client-side, at class controller HireManagement we implement call service as image below



Continue we create function hire product, at this function we use try{}catch to match error exception as convert date, calculate and some errors null from server-side. Also this function we implement function update status for this orders and this products as image below



And call this function at JSF page as image below



**This is wireframe view all products that customer reserved and hire products reserved**

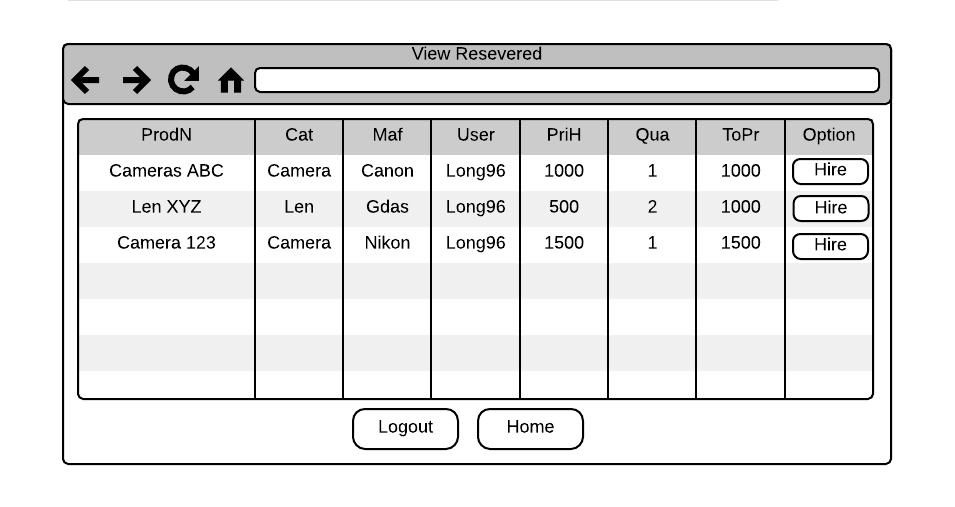
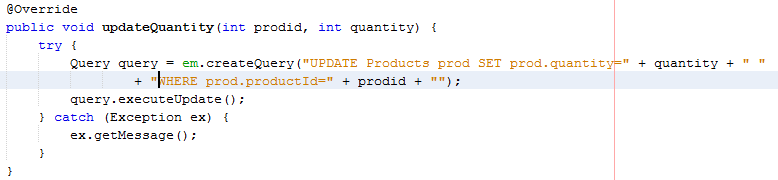


Figure 5. View products reserved

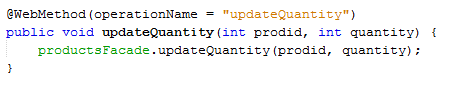
1. **Return products**

Same with function hire product. Function return product only need add function update quantity of this products after then customers returned products for store

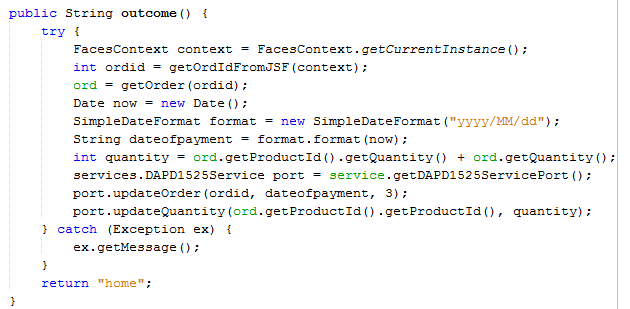
Below is function update quantity at class ProductsFacade to manipulate with database



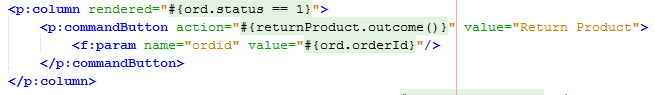
Below is function update quantity at class web service DAPD1525Service to passing parameters from customers and implement manipulate with database



At class controller we will implement call functions update order and update quantity and passing parameters as image below



Below is way we call function just created at class controller to implement manipulate return product of users



**This is wireframe for return product**

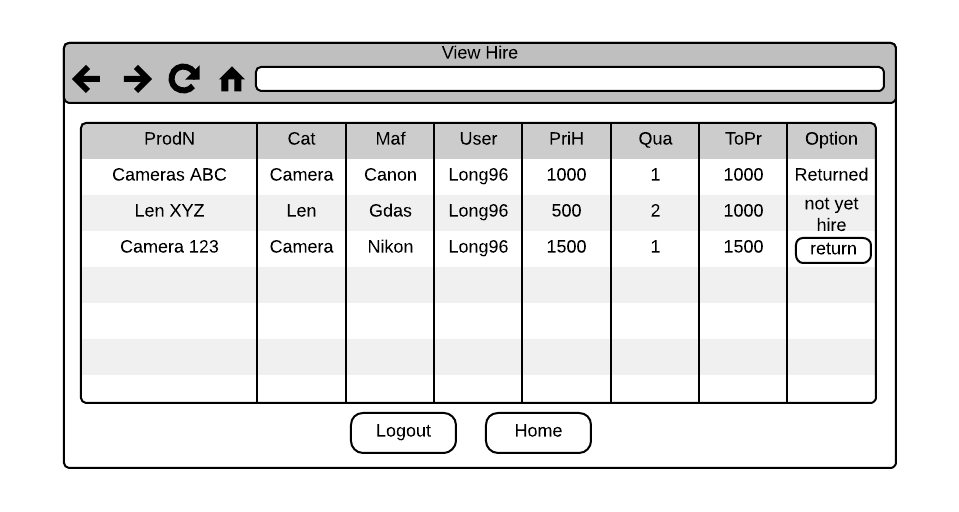


Figure 6. Wireframe view products hired

* **Conclusions:** This task I wrote about implement code for main core this is customers register -> login -> view all products -> reserve product -> hire product -> return product, I given code snippet for each functions and draw wireframe to you can easy imagine my design. Overall my program uses web services to design that help my easier in management and solve errors when program happened problems, because my program only spent for large enterprises, it is effective more when use for large enterprises, performances will faster and less faced problems more when use for small businesses or shops.

## 1.2:Define relationships between components to implement design requirements

Below is image describe process enforcement of my program, it describes detail components as component of database, component of entity bean and session bean, component of web services, component of controller (JSF Manage bean) and pages.

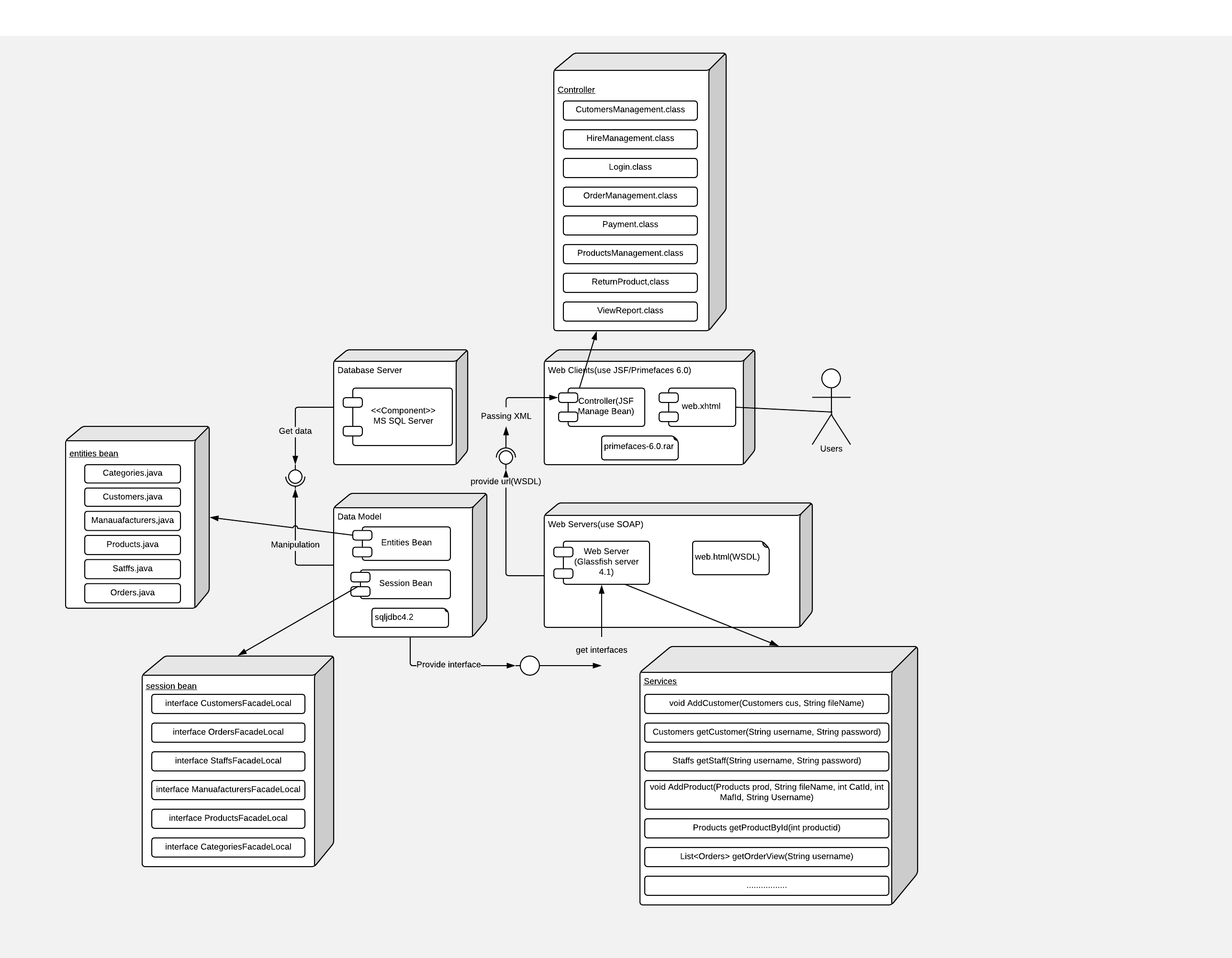


Figure 7. Relationships between components

As you can see component database have relationship direct with component session bean and entity bean because at component session bean I written code to directly interact with the database to you can understand more I will for example perform view products from a customer will have to call for the get all products and must use the select all to the database.

Similarly, other features that you want to perform must be through the session bean to manipulate the database

To client can use function in component session through web services is component session bean must provide interfaces to component web services can call them and create function help client can call them. Below is code snippet on web services to get all products.

* Code interfaces: List**<**Products**>** findAll**();**
* Code web services:

@WebMethod**(**operationName **=** "getAllProducts"**)**

public List**<**Products**>** getAllProducts**()** **{**

**return** productsFacade**.**findAll**();**

**}**

After have function on web services, on controller (JSF Manage Bean) I need call web method getAllProducts() as code snippet below

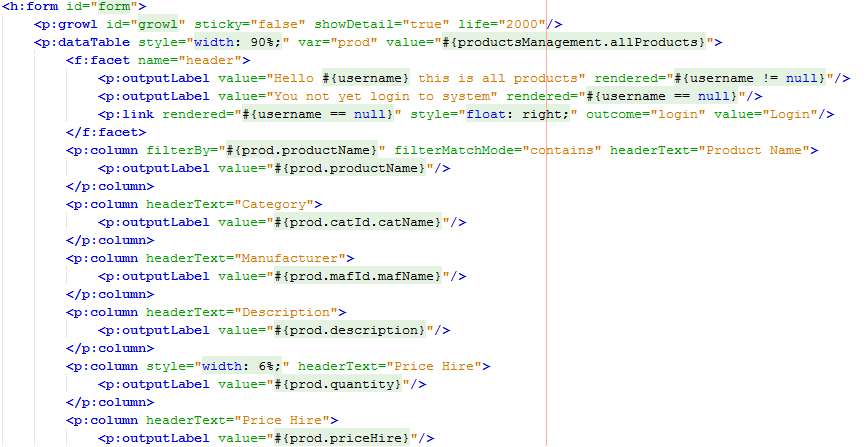
public java**.**util**.**List**<**services**.**Products**>** getAllProducts**()** **{**

services**.**DAPD1525Service port **=** service**.**getDAPD1525ServicePort**();**

**return** port**.**getAllProducts**();**

**}**

At page jsf I just call to JSF Manage Bean and can use function getAllProducts() to display information to customers. Below is code snippet display for customers.

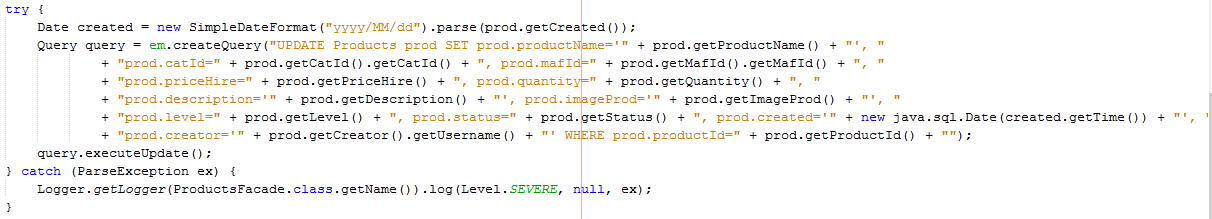
****

Similarly with other features such as reserve, hire, register ... the operation of the features is the same.

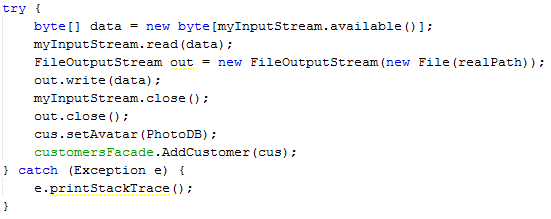
## 1.3: Identify and implement opportunities for error handling and reporting

To can design a program that is not faced with errors is very difficult, so to can discovered the errors in my program I used errors handing **try{}catch** to can discovered and solve errors by away fast. Outside I also use **if{}else** to check error when users manipulate with system. Below is some code snippet **try{}catch** I used.

Code snippet **try{}catch** below used to check error when convert data to database or check data null or fail sql query.



Code snippet **try{}catch** to check manipulate with file as file null, file too long….



Code snippet **if{}else** combined with **try{}catch** below to info error to users manipulate illegal on website and check error from form user



This image info to user

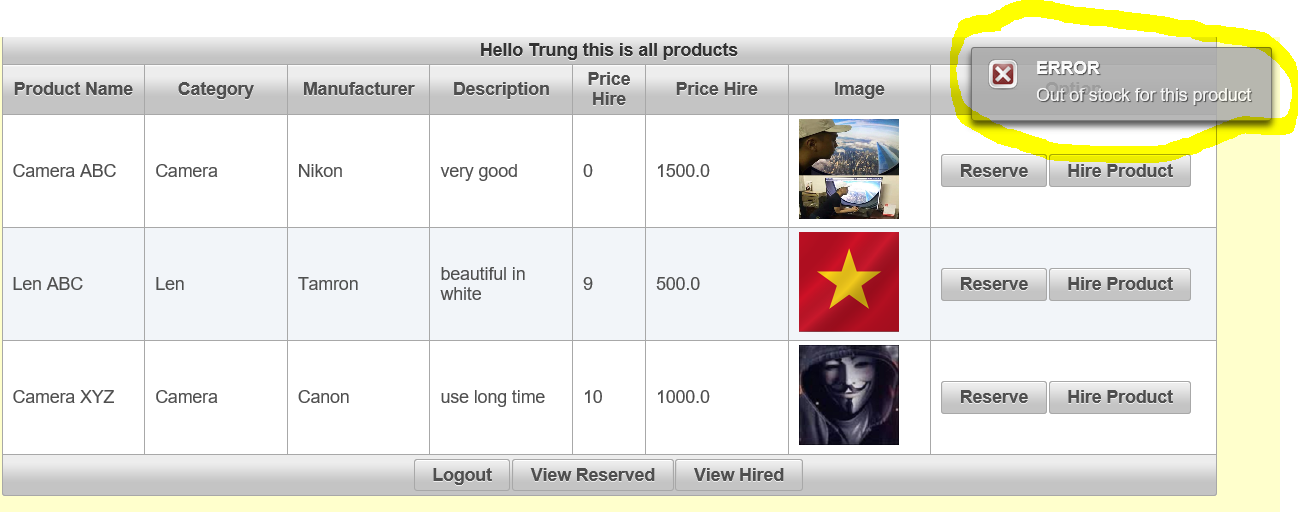


Figure 8. Error handing

This is image to check null when users do not enter to form

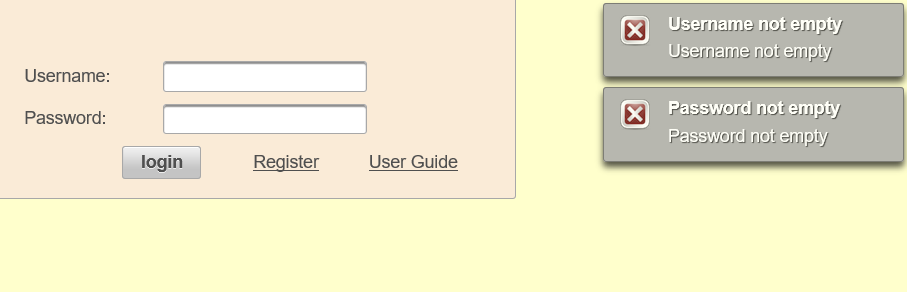


Figure 9. Check null

Check quantity when users enter over quantity in store

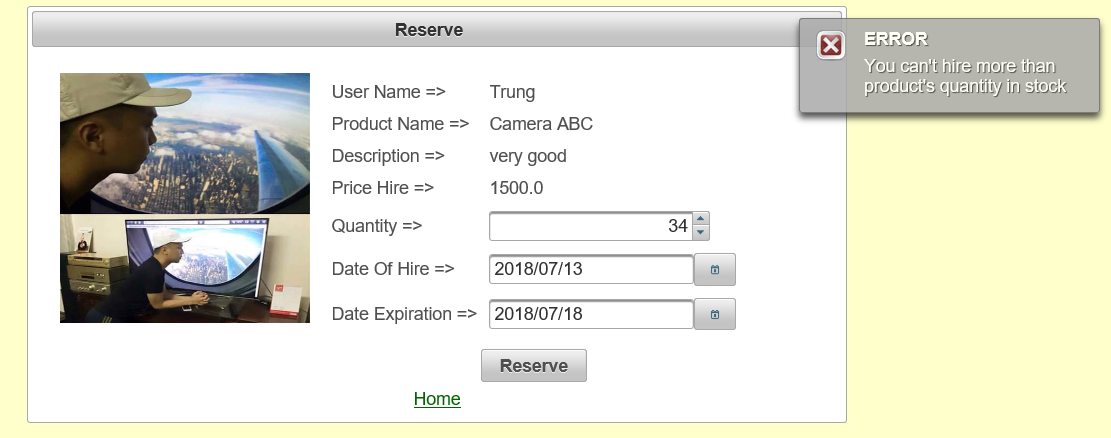


Figure 10. Check quantity in stock

## 1.4: Make effective use of an Integrated Development Environment (IDE) including code and screen templates

With this system I used IED netbeans to design for my program

With database I used tool MS SQL Server 2014 to design for database of program

Outside I also used some other tools as

* JSF / Primefaces to design for website, page display to users or function to controller manipulate of users to website.
* As I wrote above I used automation as: Entity Classes from Database and Session bean from Entity Classes.

NetBeans IDE is the official IDE for Java 8. With its editors, code analyzers, and converters, I can quickly and smoothly upgrade my applications to use new Java 8 language constructs, such as lambdas, functional operations, and method references.

Batch analyzers and converters are provided to search through multiple applications at the same time, matching patterns for conversion to new Java 8 language

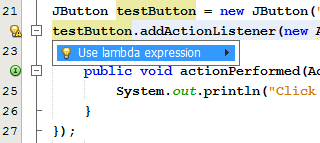


Figure 11. Language tool

The NetBeans IDE's Java Editor helps you quickly complete and generate code through the "smart" code completion feature. In a general sense, code completion is very useful when you want to fill in the missing code, look at the options available in the context of your application, and generate blocks of code when needed. See below for examples of how to use code completion.

* **Invoking Code Completion**

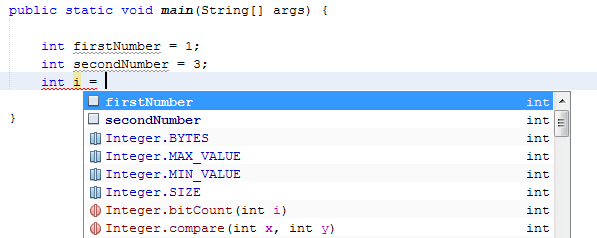


Figure 12. Invoking code completion

* **Smart Suggestions at the Top**

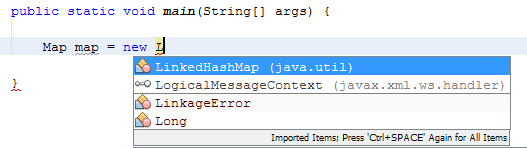


Figure 13. Smart suggestions at the top

* **Camel Case Completion**

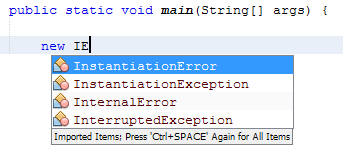


Figure 14. Camel case completion

* **Completing Keywords**

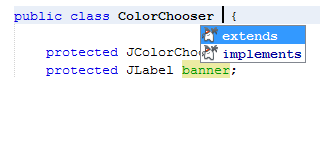


Figure 15. Completing keywords

* **Suggesting Names for Variable and Fields**

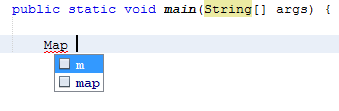


Figure 16. Suggesting names for variable and fields

* **Suggesting Parameters**

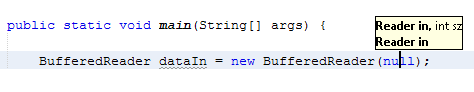


Figure 17. Suggesting parameters

* **Common Prefix Completion**

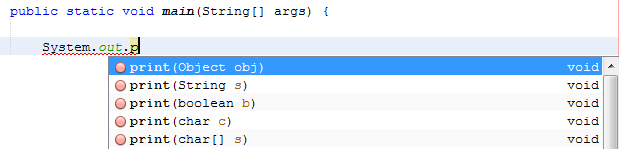


Figure 18. Common prefix completion

* **Subword Completion**

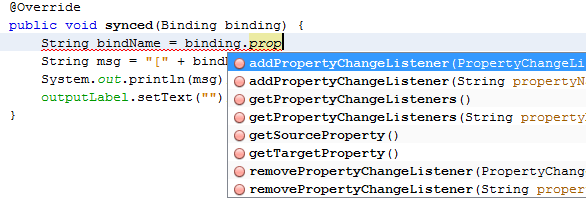


Figure 19. Subword completion

* **Conclusions:** This task wrote about explain relationships between components in program, given example and image UML deployment of program, I also wrote about explain error handing I used in design program given code snippet and image info error. Outside I wrote about I used tools to design program, some tools and automation to support for design.

# **Task 2: (L04 4.1, 4.2, 4.3, D1 and D2) Software Testing Stage and Software Evaluation Stage**

## 2.1: Critically review and test your solution

Looking back at the whole system I have designed with the appropriate features for the customer that is

* Customers can register their account and log on to the system
* Customers can view the products and order the products they want according to the quantity
* Customers can rent products they have previously booked
* Customers can return the product they rented after they have used it

The features on the system have met all the requirements of the ABC company. Also the system needs more features for the staff that is

* Employees can add new products on the online store
* Employees can manage customer accounts: such as accepting customer account creation or locking customer accounts
* Employees can view store statistics for a period of time they want.

The features shown on the site are as shown below

* Feature customers register account

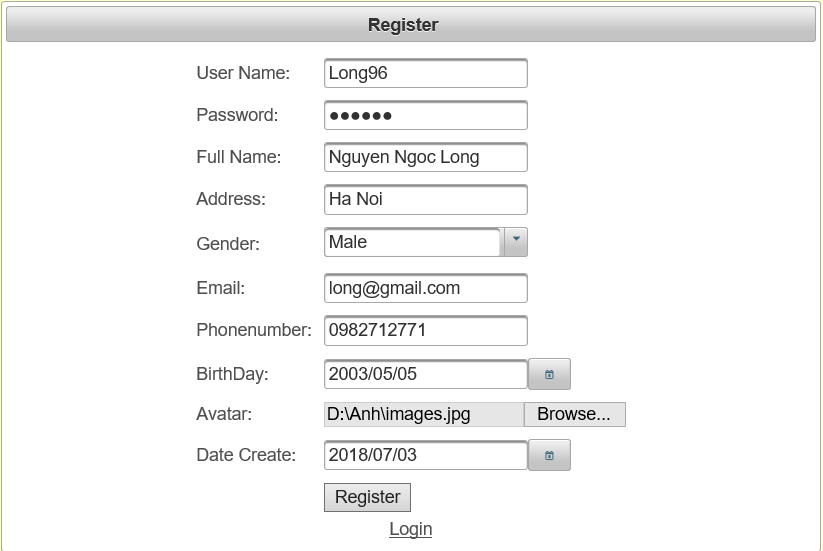


Figure 20. Register of customers

* Customers login to system

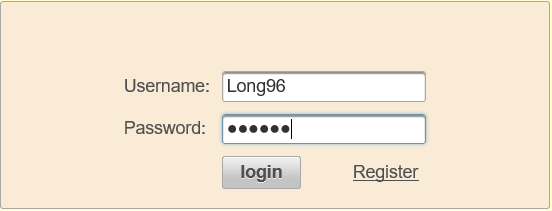


Figure 21. Login to system

* Customers view products



Figure 22.View all products from customers

* Customers reserve product

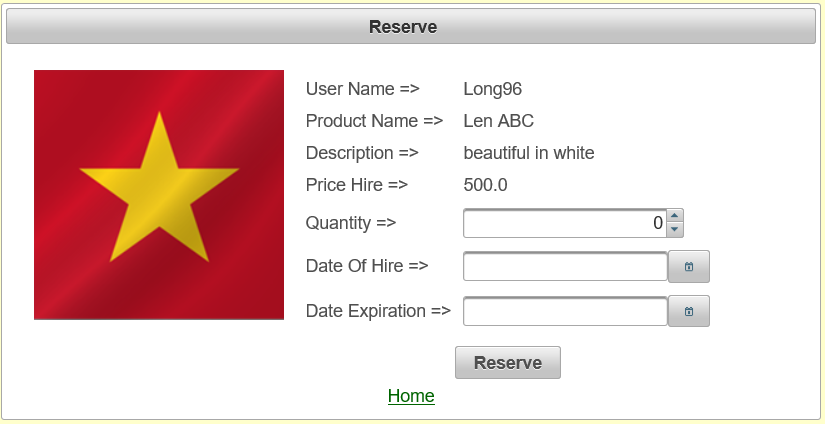


Figure 23. Customers reserve product

* Customers hire product

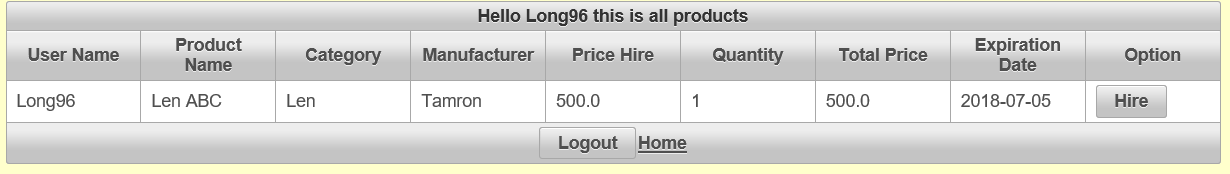


Figure 24. Customers hire product

* Customers return product

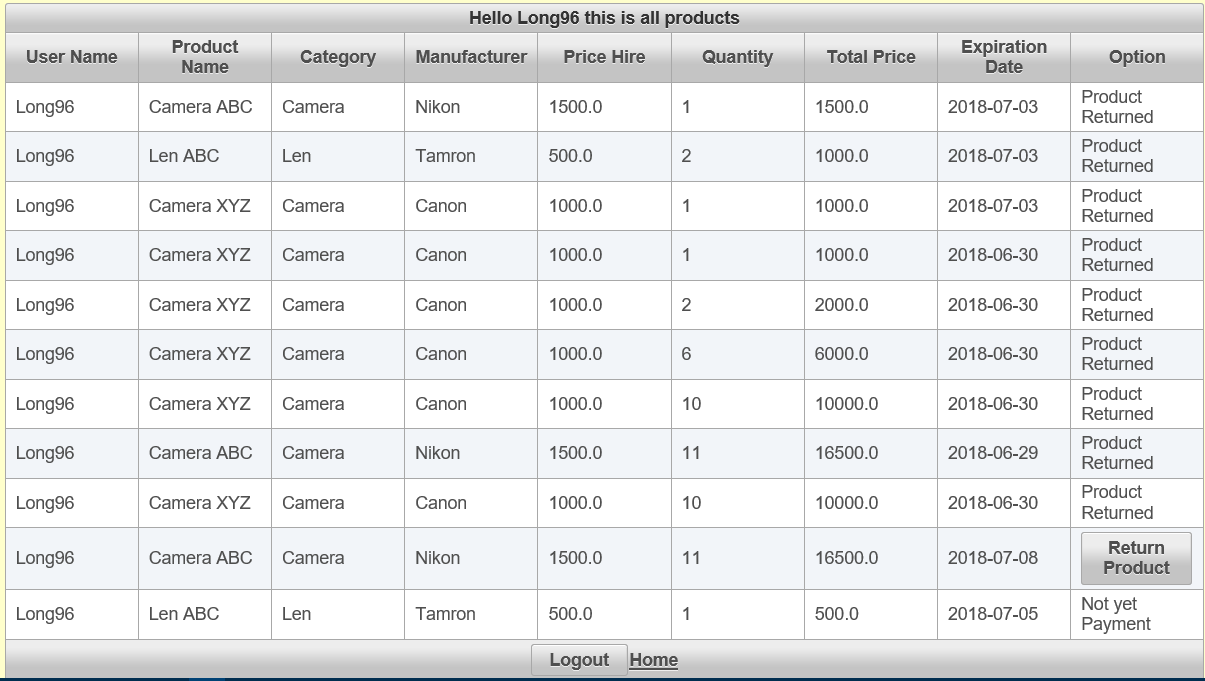


Figure 25. Customers return product

* Staffs add new product

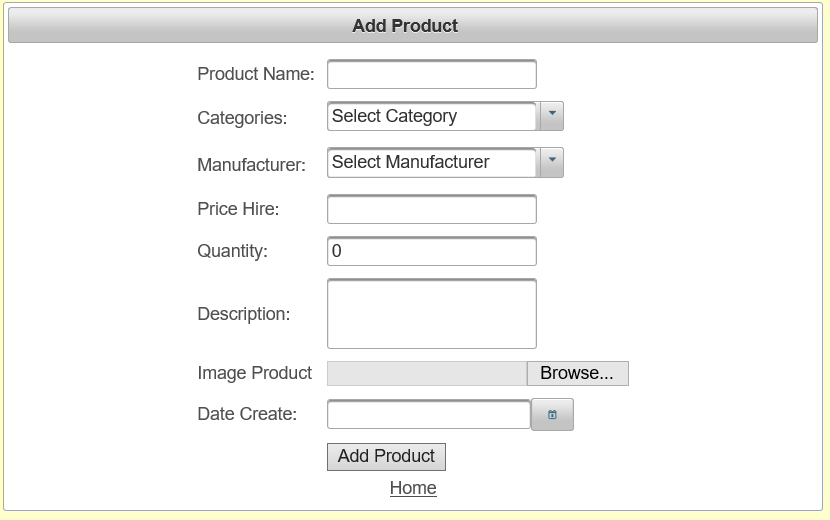


Figure 26. Staffs add new product

* Staffs view report in duration they want

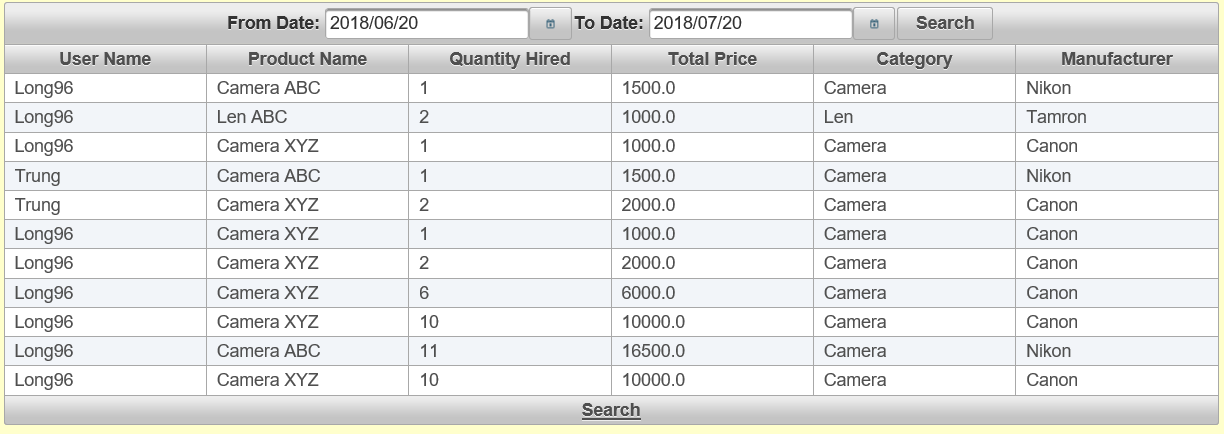


Figure 27. Staffs view report in duration they want

* Staffs management account of customers



Figure 28. Staffs management account of customers

The system still has some weaknesses as the site design is not very user-friendly and requires some additional features to help users and managers alleviate the problem. Payment methods should be added so that users and administrators can chat online so managers can advise on products for users, so there is a feedback section with products from customers. Customers can edit personal information on the site this feature needs to be added. In the future, the most important thing is to redesign a website that is relevant and beautiful to the customer, such as changing the way the homepage is displayed, customers can search for products through category and manufacturers ...

* **Conclusions:** This task I wrote about review program and wrote about weakness of program, I will need to do to improve the site and need develop some new features to help for customers and staffs.

## 2.2: Analysis actual test results against expected results to identify discrepancies

Below is table test plan for my system, before starting the test plan, you need to prepare a programmable computer, which includes test manager and tester components.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| No | Scope | Type | Input | Actual output | Expected output | Status |
| 1 | Register | Normal scenarios | Username: Long96  Password: 123456  FullName: Nguyen Ngoc Long  Address: Ha Noi  Gender: Male  Email: long96@gmail.com  PhoneNumber: 0981234432  Birthday: 12-27-1996  Avatar: avatar2.png  Created: 07-03-2018 | Insert table Customers successful (register successfully) | Insert table Customers successful (register successfully) | Pass |
| 2 | Register | Data validation | Username: Quan  Password: 123456  FullName: Nguyen Duy Quan  Address: Ha Noi  Gender: Male  Email: quannd@gmail.com  PhoneNumber: wqewqw  Birthday: 05-15-1980  Avatar: avatar2.png  Created: 05-25-2018 | Insert tables Customers is not successful (register not successfully because phone is not format and not number) | Insert tables Customers is not successful (register not successfully because phone is not format and not number) | Pass |
| 3 | Add Product | Normal scenarios | ProductName: Camera ABC  Category: Camera  Manuafacturers: Canon  Description: Good  Quantity: 20  PriceHire: 1000  ImageProduct: camera.png  Create date: 05-25-2018  Creator: Admin1 | Insert table Products successful (add new product successfully) | Insert table Products successful (add new product successfully) | Pass |
| 4 | Add product | Data validation | ProductName: Camera ABC  Category: Camera  Manuafacturers: Canon  Description: Good  Quantity: 20  PriceHire: weqeq  ImageProduct: camera.png  Create date: 05-25-2018  Creator: Admin1 | Insert table Products is not successful (add product not successfully field price hire is not number) | Insert table Products is not successful  (add product not successfully field price hire is not number) | Pass |
| 5 | Reserve | Normal scenarios | Username: Long96  ProductId: 1  Quantity: 2  DateOfHire: 05-25-2018  ExpirationDate: 06-01-2018  DateOfPayment: 05-28-2018  Status: 0 | Insert table Orders successful | Insert table Orders successful | Pass |
| 6 | Reserve | Data validation | Username: Long96  ProductId: 1  Quantity: 2  DateOfHire: 05-25-2018  ExpirationDate: 05-23-2018  Status: 0 | Insert table Orders is not successful (reserve not successfully because date of hire can’t less more expiration date) | Insert table Orders is not successful  (reserve not successfully because date of hire can’t less more expiration date) | Pass |
| 7 | Hire product | Normal scenarios | OrderId: 1  DateOfPayment: 05-27-2018  Status: 1  Quantity: 1 | update table Orders and Products successful (hire product successfully) | update table Orders and Products successful (hire product successfully) | Pass |
| 8 | View all Products | Normal scenarios |  | select table Products is successful  (Display all products to users) | select table Products is successful  (Display all products to users) | Pass |
| 9 | View report | Normal scenarios | From date: 06-20-2018  To date: 07-20-2018  Status: 3 | select table Orders with condition from date to date and status is successful (display the orders to staffs) | select table Orders with condition from date to date and status is successful (display the orders to staffs) | Pass |
| 10 | Return product | Normal scenarios | Status: 3  Quantity: 1 | update table Orders and Products is successful (return product successfully) | update table Orders and Products is successful (return product successfully) | Pass |
| 11 | Login | Normal scenarios | Username: Long96  Password: 123456 | Check account is true (login successfully) | Check account is true (login successfully) | Pass |
| 12 | Login | Data validation | Username: Long96  Password: 123455 | Check account is fail because password is not like with database (login is not successfully) | Check account is fail because password is not like with database (login is not successfully) | Pass |
| 13 | Search Name Product | Normal scenarios | Text entered: Camera | System will display products have name are Camera to users | System will display products have name are Camera to users | Pass |
| 14 | Search Name Product | Data validation | Text entered: 12 | System will not display information about product for keyword you searched | System will not display information about product for keyword you searched | Pass |
| 15 | Update Product | Normal scenarios | ProductName: Camera XYZ  Category: Camera  Manuafacturers: Canon  Description: Good  Quantity: 10  PriceHire: 1500  ImageProduct: camera.png  Create date: 05-25-2018  Creator: Admin1 | Update data changed from staff successful and save data changed in database (update table product successful) | Update data changed from staff successful and save data changed in database (update table product successful) | Pass |
| 16 | Update Product | Data validation | ProductName: Camera XYZ  Category: Camera  Manuafacturers: Canon  Description: Good  Quantity: sdasd  PriceHire: 1500  ImageProduct: camera.png  Create date: 05-25-2018  Creator: Admin1 | Update table product not successful because field quantity can’t is characters, it is only number | Update table product not successful because field quantity can’t is characters, it is only number | Pass |

Table 1. Table test case for system.

* **Conclusions:** After test case for my system, there are a total of sixteen test cases for 16 features, the result of the test case is that all 16 features are pass, no features fail, good operating system is easy to use. Beside I also not yet test case for function cancel product from customers because I still yet write code for this function, next time I will write code so customers can cancel the product when they do not want to place the product.

## 2.3: Evaluate independent feedback with recommendations for improvement

To get a sense of the value of my program, I have prepared the following questions to consult with my classmates and teachers so that I can draw strengths and weaknesses. of the program.

First question and the chart statistics the results of the question

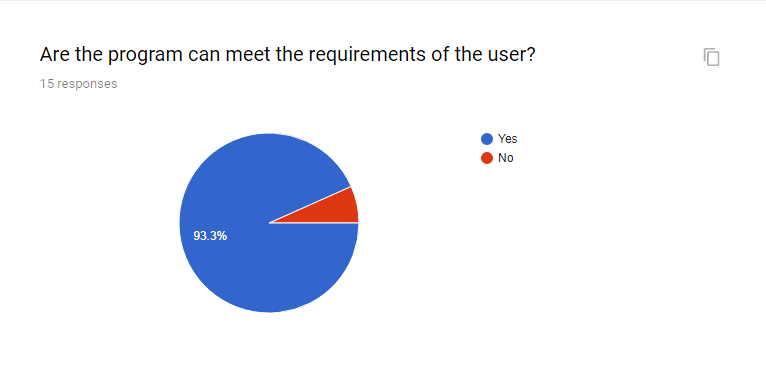


Figure 29. First questionnaire

Second question and the chart statistics the results of question

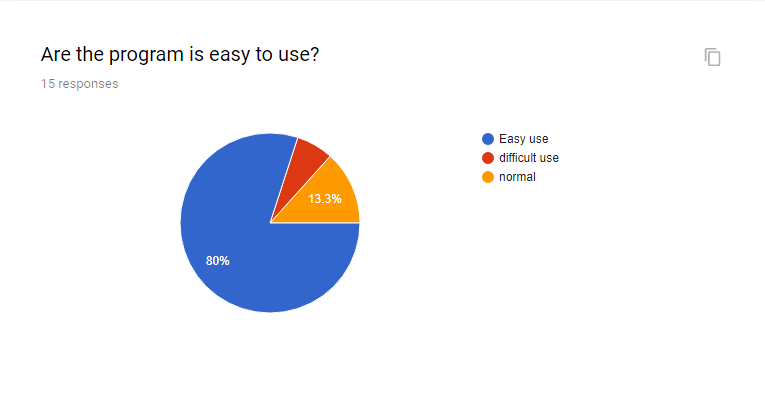


Figure 30. Second questionnaire

Third question and the chart statistics the results of question

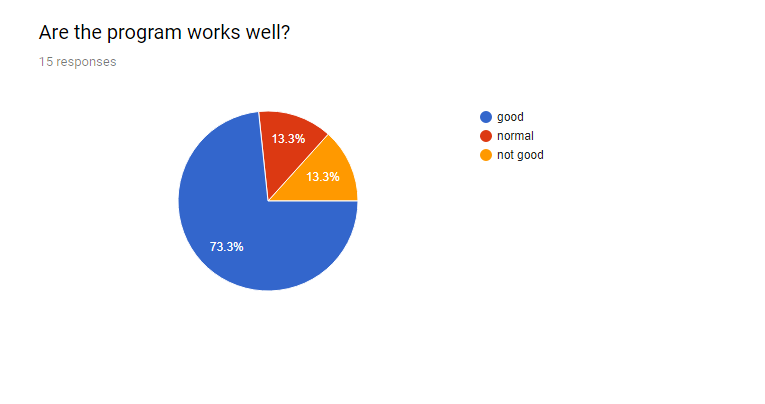


Figure 31. Third questionnaire

Fourth question and the chart statistics the results of question

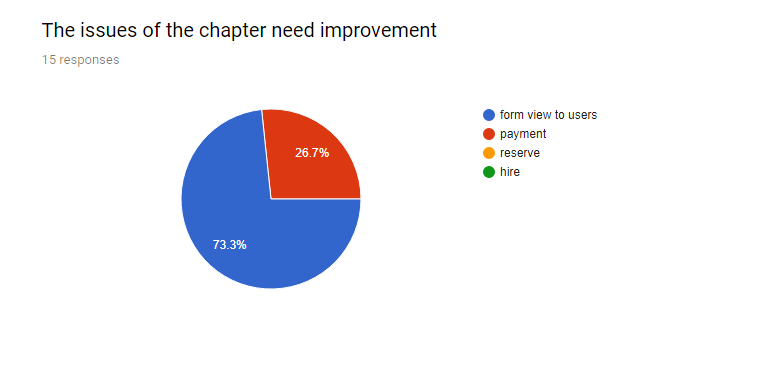


Figure 32. Fourth questionnaire

* **Table result for feedbacks and solution**

|  |  |  |  |
| --- | --- | --- | --- |
| No | Questions | Result | Solution |
| 1 | Are the program can meet the requirements of the user? | Yes: 93.3%  No: 6.7% | Results show that the features of the program meet the requirements of the user, the features are necessary and useful to the user. So we should not change the features of the program |
| 2 | Are the program is easy to use? | Easy use: 80%  Difficult use: 6.7%  Normal: 13.3% | The results show that the program is quite easy to use, but besides that the website design is quite simple and difficult to see so need to improve in terms of user interface. |
| 3 | Are the program works well? | Good: 73.3%  Normal: 13.3%  Not good: 13.3% | The program performs well without any problems, but there are still some flaws that need to be repaired as the performance of the user needs to be responded more quickly. |
| 4 | The issues of the chapter need improvement | Form view to user: 73.3%  Payment: 26.7%  Reserve: 0%  Hire:0% | The evaluation and evaluation of the program requires a change of the look of the viewer as soon as possible, and a need to upgrade the payment facility with the expansion of payment capacity and payment methods. |

Table 2. Result and solution for questionnaire

* **Conclusions:** This task I wrote feedbacks of classmate and teacher for my program, analysis result and given solution for problems had in program. Overall my program work pretty ok, program was adapted requirements from customers, features are good, less happened errors and problems in work, my program is easy use. But, some problems need improve as interfaces view with user need improve as soon as possible and need add feature payment with many types of payment as payment by banks, payment by cards…

# **Task 3: (L04 4.4, 4.5 M3) Software Documentation Stage (User guide & Technical)**

## 2.4: Create user documentation for the developed application

Below as a documentation for the manual for the user usage of the web site features, out of the also create a page manual website using the page, to access you see the image below.

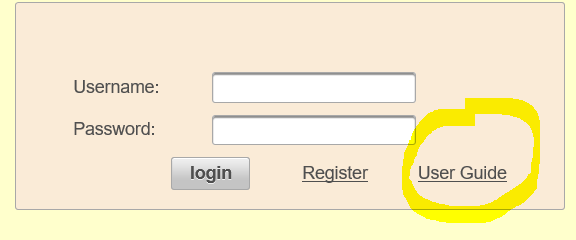


Figure 33. how to go to page user guide

Below is some guides support for customers can use features on our web site

1. **How to register a new account?**

**Step 1:** To register the account you need to click the register button on the login page and visit the account registration page on the site

**Step 2:** Next you need to enter the complete information form the system requirements as image below

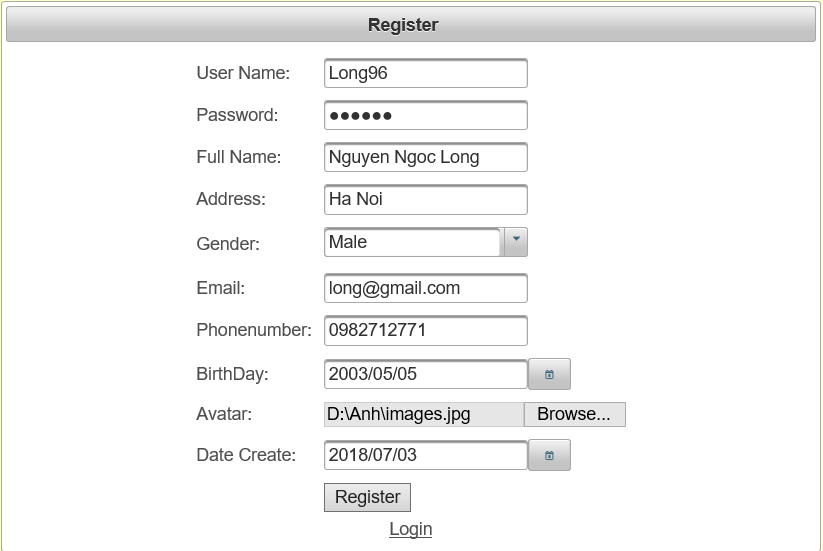


Figure 34. Screen register

**Step 3:** Finally, after you have entered the full information, you need to click the register button below to system register account.

**2. How to login to system?**

To login to the system you need to register the account as we have guided above

**Step 1:** After registering your account at the login page, you will need to enter your username and password for verification.

**Step 2:** Next click on the Login button to system login account. If you don’t enter username and password or you enter wrong username or password system will info error as image below.

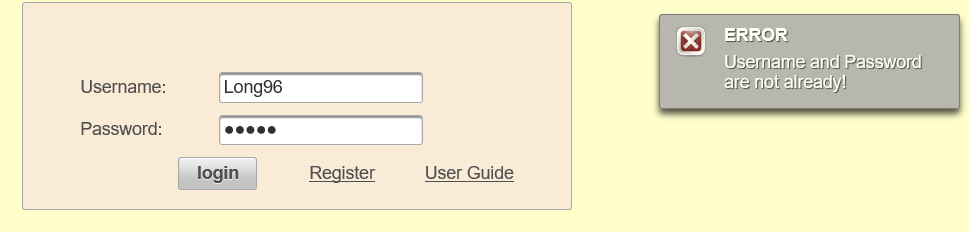


Figure 35. Check enter wrong

**3. How to reserved products on web site?**

**Step 1:** After you have logged in successfully, the system will go to the home page, where you select the product you want to rent and click on the reserve button.

**Step 2:** At reserve you need to enter the number of products you want to rent, then choose the start date and return the product and click on the reserve button to make the system to order products for you as image below

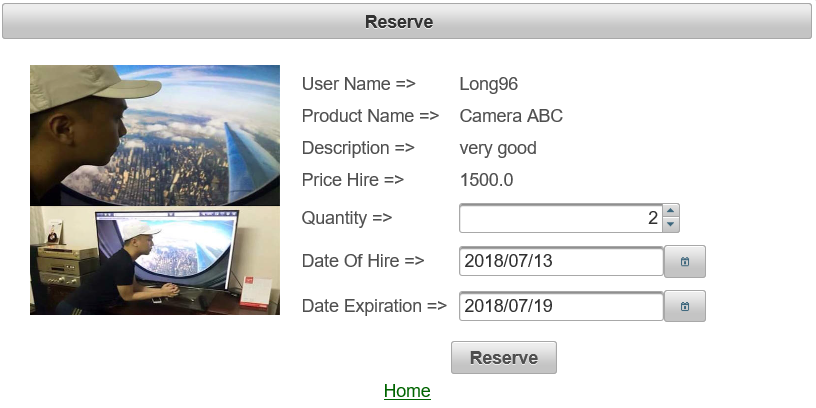


Figure 36. Reserve

1. **How to view products reserved and hire this products?**

**Step 1:** On the home page you click on the reserved view button to view the products you have reserved

**Step 2:** After going to the reserved view page, to hire a product, click on the hire button to get your product hired, the date paid is the current date you are hiring the product. As image below

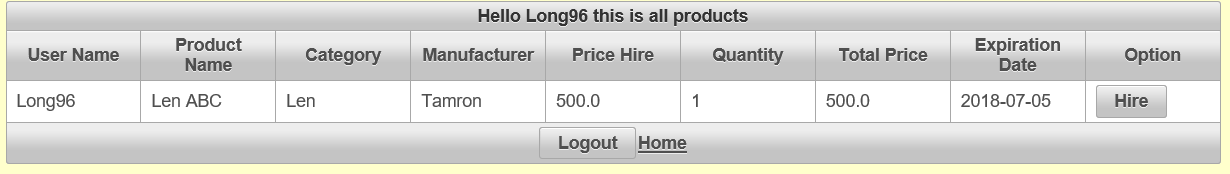


Figure 37. Hire product

1. **How to view products hired and return this products?**

**Step 1:** On the home page click on the view hired to see the products you have rented

**Step 2:** After you have used the product, if you want to return the product, please click on the return product button at the product page for the product to return your product. As image below

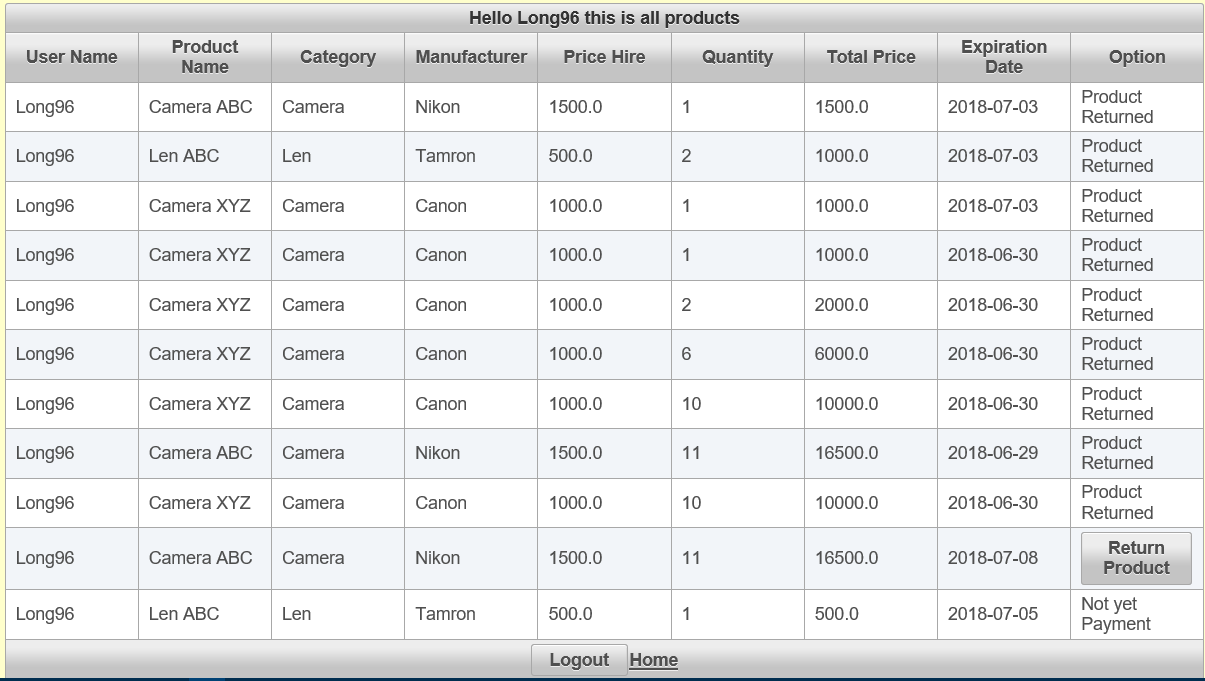


Figure 38. Return product

* **Conclusions:** this task I wrote about a document to guide users about features on web site and how to design for site include given code.

## 2.5: Create documentation for the support and maintenance of your application

1. **Introduction**

My program is designed to meet the customer's need for camera and lens rentals with the following customer features

* Customers can register their account and log on to the system
* Customers can view all products and order products
* After ordering the product the customer can rent the product
* Customers can return products they have rented

In addition, the program has the following features for employees

* Employees can add new products to the online store
* Employees can manage client accounts
* Employees can view revenue reports within the time they want

1. **Application overview**

I have just made a contract with ABC company, which provide hiring cameras and lens services in the city. ABC Company wants to develop a program that can help them to manage their job. Basically, they want to manage:

* Customer and staff
* Camera and Lens
* Customer can reserve cameras and lens
* Customer can hire cameras and lens based on reservation or at store
* Customer can return hired cameras and lens

Customer information: Name, Address, Telephone.

Staff information: Name, Phone

Camera/Len Type: Name (i.e: Canon, Nikon, …)

Camera: Camera type, Model, Status, Color, Hiring Price (per day), Paying Price (in case of loss). Example: (Canon, 5DmkII, Good, Black, 10, 1000)

Len: Len Type, Model, Status, Hiring Price (per day), Paying Price (in case of loss). Example (Tamron, 100mm Macro, New, 8, 500)

Order: customer, staff, number of cameras, lens, total price, etc.

This application will need to provide the functionalities via a Web-based application or Desktop-based application of your selected technologies. Database can be stored in Microsoft SQL Server, MySQL or other DBMS.

All database related operations must be done via web services. Web services technology can be RESTful or SOAP (selection must be explained).

1. **Testing plan used and test results**

At task one I tested plan for my program, here I will write result of test plan again. there are a total of sixteen test cases for 16 features, the result of the test case is that all 16 features are pass and not features fail.

1. **Possible improvements**

The system still has some weaknesses as the site design is not very user-friendly and requires some additional features to help users and managers alleviate the problem. Payment methods should be added so that users and administrators can chat online so managers can advise on products for users, so there is a feedback section with products from customers. Customers can edit personal information on the site this feature needs to be added. In the future, the most important thing is to redesign a website that is relevant and beautiful to the customer, such as changing the way the homepage is displayed, customers can search for products through category and manufacturers ...

1. **Current version numbers and Date last changed**

My current program has not been put into operation and has only been tested for the first time and has not detected any errors, but the program needs to be improved to be able to put into use.

1. **Deployment of your web services and Web service usages**

Two part I wrote above. So, I will not write again here

* **Conclusions:** This task I wrote about a document technical booklet that will explain to other developers what your solution is all about, how it is supposed to work and how to maintain it.