|  |  |
| --- | --- |
| **Project Case** |  |
| COMP6115  Object Oriented Analysis & Design |
| **Computer Science** | **E212-COMP6115-DU01-00** |
| ***Valid on*** *Even Semester Year 2020/2021* | **Revision 00** |

1. Seluruh kelompok tidak diperkenankan untuk:

*The whole group is not allowed to:*

* + 1. Melihat sebagian atau seluruh proyek kelompok lain,

*Seeing a part or the whole project from another groups*

* + 1. Menyadur sebagian maupun seluruh proyek dari buku,

*Adapted a part or the whole project from the book*

* + 1. Mendownload sebagian maupun seluruh proyek dari internet,

*Downloading a part or the whole project from the internet,*

* + 1. Mengerjakan soal yang tidak sesuai dengan tema yang ada di soal proyek,

*Working with another theme which is not in accordance with the existing theme in the matter of the project,*

* + 1. Melakukan tindakan kecurangan lainnya,

*Committing other dishonest actions,*

* + 1. Secara sengaja maupun tidak sengaja melakukan segala tindakan kelalaian yang menyebabkan hasil karyanya berhasil dicontek oleh orang lain / kelompok lain.

*Accidentally or intentionally conduct any failure action that cause the results of the project was copied by someone else / other groups.*

1. Jika kelompok terbukti melakukan tindakan seperti yang dijelaskan butir 1 di atas, maka **nilai kelompok** yang melakukan kecurangan (menyontek maupun dicontek) akan di – **NOL** – kan.

*If the group is proved to the actions described in point 1 above, the score of the group which committed dishonest acts (cheating or being cheated) will be “Zero”*

1. Perhatikan jadwal pengumpulan proyek, segala jenis pengumpulan proyek di luar jadwal tidak dilayani.

*Pay attention to the submission schedule for the project, all kinds of submission outside the project schedule will not be accepted*

1. Bila Anda tidak membaca peraturan ini, maka Anda dianggap telah membaca dan menyetujuinya

*If you have missed to read these regulations, so you are considered to have read and agreed on it*

1. Persentase penilaiaan untuk matakuliah ini adalah sebagai berikut:

*Marking percentage for this subject is described as follows:*

|  |  |  |
| --- | --- | --- |
| **Tugas Mandiri**  *Assignment* | **Proyek**  *Project* | **UAP**  *Final Exam* |
| 40% | 60% | - |

1. Software yang digunakan pada matakuliah ini adalah sebagai berikut:

*Software will be used in this subject are described as follows:*

|  |
| --- |
| **Software**  *Software* |
| Visual Paradigm Community Edition 16.1 Java 8  Eclipse 2020.6  XAMPP 7.4.7  MySQL Java Connection Library 5.1.49 |

## Ekstensi file yang harus disertakan dalam pengumpulan tugas mandiri, dan proyek untuk matakuliah ini adalah sebagai berikut:

*File extensions should be included in assignment and project collection for this subject are described as follows:*

|  |  |
| --- | --- |
| **Tugas Mandiri**  *Assignment* | **Proyek**  *Project* |
| VPP | JAVA, CLASS, SQL |

## Soal

*Case*

**Just DU It !**

Just Du It ! is a famous Indonesian technology company specializing in branded shoe. To expand its business, Just Du It ! wants to improve its service by developing a system that can help them to manage their business activity within the shop. Below is the business flow that required to be covered by the system.

* **Transaction Management**

Every transaction the customer made will be recorded in the shop system. The transaction occurs when the customer is buying the products available in the shop. Before the transaction proceeds to the payment section, each product customer bought **must be recorded in the temporary object called cart**. The only employee with the Cashier role **can add a product to the cart, remove product from cart** and **confirm the transaction to proceed to payment**. Please notes that there is **no duplicate product in the cart object**. It means that when the cashier adding an existing product to the cart, it will **update the quantity of the existing product** object instead of creating a new product object in the cart which will result in duplicated products in the cart. Do not forget to **calculate a new total price** each time the cashier adds the product to the cart.

For the payment section, Just DU It! accepts two kinds of payment, **cash** and **credit card**.

After the payment is done and the cashier confirms to checkout, a new transaction object and **each transaction item** object will be **created** and **recorded** in the database. Do not forget to **insert a timestamp** each time the transaction object is recorded in the database because the Cashier can view all **today transaction based on the transaction timestamp**. Then, product stock corresponding to the **quantity** in the transaction will be **decreased**.

|  |  |
| --- | --- |
| **Variable** | **Validation** |
| Product ID | * Cannot be empty * Must exist in the database |
| Product Quantity | * Cannot below zero * Must be less than the product stock |
| Payment Method | * Must be chosen (between credit or cash) |

* **Product Management**

The employee with the **Product Management** role can **view** the shop storage, including all the products in the database. Each product has its information, and the staff can **insert** **a product,** **update** the information, or **delete** the product from the database.

|  |  |
| --- | --- |
| **Variable** | **Validation** |
| Product ID (Update & Delete) | * Cannot be null * Must be exists * Cannot be updated |
| Product Name | * Cannot be empty |
| Product Description | * Cannot be empty |
| Product Price | * Must be numeric * Must be above zero |
| Product Stock | * Must be numeric * Must be above zero |

* **Human Resource Management**

**Human Resource** staff is responsible to handle all the staff data. They can **insert new employees**, **update employee** data, **fire employees** (this only update the employee status from **“Active”** to **“Not Active”** and **do** **not delete the data from database**) and **view all employees**. When insert new employee, default employee password is his / her **username**.

|  |  |
| --- | --- |
| **Variable** | **Validation** |
| Employee ID (Update & Delete) | * Cannot be empty * Must be exists |
| Employee Role | * Cannot be empty * Role ID must be exists |
| Employee Name | * Cannot be empty |
| Employee Username | * Cannot be empty * Must be unique |
| Employee Status | * Cannot be empty * Must be “Active” or “Not Active” |
| Employee Salary | * Cannot be empty * Must be numeric * Must be above zero |

* **Manager**

To make the business run smoothly, Just DU It hire a manager for supervising and motivating employees. The **manager** can **view transaction reports generated by the system**. The transaction report includes **all transactions** recorded by the system **each month for each year**. The manager needs to choose month and year when the transaction happened, and the system will show all transactions based on the chosen month and year. To view all transaction detail, including the transaction item list, the manager may **click the transaction cell in the table** and the system will show all transaction items based on the chosen transaction id in the cell. Besides that, the manager can also **view**, **fire** (same as Human Resource Department), **update**, and **insert employee**.

\*\*\*

To help you design the program, Just DU It already hire Daniel, a System Analyst, to provide you with the design of the program based on the requirement and the business flow above. You can open “**diagram.vpp**” to view the diagram that designed by Mr. Daniel.

\*\*\*

As a programmer, Just DU It wants you to develop the application with the following requirement:

* The application must be build using Java-based Programming with **MVC** (**Model View Controller**) architecture for a better development process
* Model

The model layer is responsible for **representing concepts** in the business or information about the business situation. Besides that, model layer also responsible for **giving access to the database** via its public interfaces to acquiring and manipulating references to preexisting domain objects.

* View

View layer, or Presentation Layer, is responsible for showing information to the user and interpreting the user's commands. This layer is the home for **all user interfaces** in the project.

* Controller

This layer is responsible to **validate** all input from the view layer and **all business logics** are implemented in the controller layer. It also responsible for **delegating request**s from the user to the lower layer for further processing.

* The database must be using **MySQL**
* The application must have an **authenticated user based on roles**
* The application must minimalize human error with great user experiences
* The application must be made based on the analysis diagram that provided by Mr. Daniel, but you may add additional components in the system based on your assumption
* Documentation for the application:
* The guide for using the application
* Any additional assumption that you make to develop the program

**Please ask your teaching assistant if there are any related questions.**

Here are the rules that you must follow to create your project:

1. Use appropriate software for this subject based on **Sistem Praktikum** that can be downloaded from Binusmaya
2. Use the techniques taught during practicum
3. Collect appropriate files for this subject based on **Sistem Praktikum** that can be downloaded from Binusmaya
4. Include the other files that can support your project, such as:
   * All files in your project
   * Other files (image, audio, video, etc.) used in your project
   * \*.DOC file (documentation of your project) that contains all pages in your project, reference links of additional files (image, audio, video, etc.) used in your project, the description about how to use your application, etc.