|  |  |
| --- | --- |
| **Project Case** |  |
| COMP6144  Pattern Software Design |
| **Computer Science** | **E202-COMP6114-BD01-00** |
| ***Valid on*** *Even Semester Year 2019/2020* | **Revision 00** |

1. Seluruh kelompok tidak diperkenankan untuk:

*The whole group is not allowed to:*

* + - Melihat sebagian atau seluruh proyek kelompok lain,

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

* + - Menyadur sebagian maupun seluruh proyek dari buku,

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

* + - Mendownload sebagian maupun seluruh proyek dari internet,

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

* + - 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,*

* + - Melakukan tindakan kecurangan lainnya,

*Committing other dishonest actions,*

* + - 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. Jangan lupa untuk melihat kriteria penilaian proyek yang ditempel di papan pengumuman, atau tanya asisten anda.

*Don’t forget to look at the project assessment criteria that posted on the announcement board, or ask your teaching assistant.*

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 Studio 2015 Community Edition (ASP.NET) * SAP Crystal Reports 13.0.16 * Firefox / Chrome |

## 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* | **UAP**  *Final Exam* |
| Folder Project (SQL, ASPX, ASPX.VB, ASPX.CS, CONFIG, ASHX, WEBINFO, DLL, PDB, DBML, DBML.LAYOUT, DESIGNER.VB, MDF, LDF, Image Files (JPG / PNG)) | Folder Project (SQL, ASPX, ASPX.VB, ASPX.CS, CONFIG, ASHX, WEBINFO, DLL, PDB, DBML, DBML.LAYOUT, DESIGNER.VB, MDF, LDF, Image Files (JPG / PNG)) | - |

## Soal

*Case*

**TokoBeDia**

**TokoBeDia** is a new e-commerce service that launched in Indonesia. To start its business, **TokoBeDia** needs to **create a website** for their service. On the website, people can find items that are being sold and purchase it. This website also allows admins to manage user, item, item type, and payment type.

As a web developer, you are asked to create a website using **ASP.NET** for this online shop. You are also **required** to use **Domain Driven Design** method to develop the website. Please note that using web service is **optional** in this project. The **required** layers are:

* **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. It also responsible for **delegating request**s from the user to the lower layer for further processing.

* **Handler**

This layer is responsible to handle all **business logic** required in the application. It will delegates the task to query from the database, including select, insert, update and delete, to the repository layer. Please notes that there can be a single handler that accesses multiple repository.

* **Repository**

Repository layer responsible for **giving access** to the database and model layer via its public interfaces to acquiring references to preexisting domain objects. It provides methods to manipulate the object, such as add and delete, which will **encapsulate the actual manipulation operation** of data in the data store (or database). Repository also provides methods that select objects based on some criteria and return fully **instantiated** objects or collection of objects whose attribute meets those criteria.

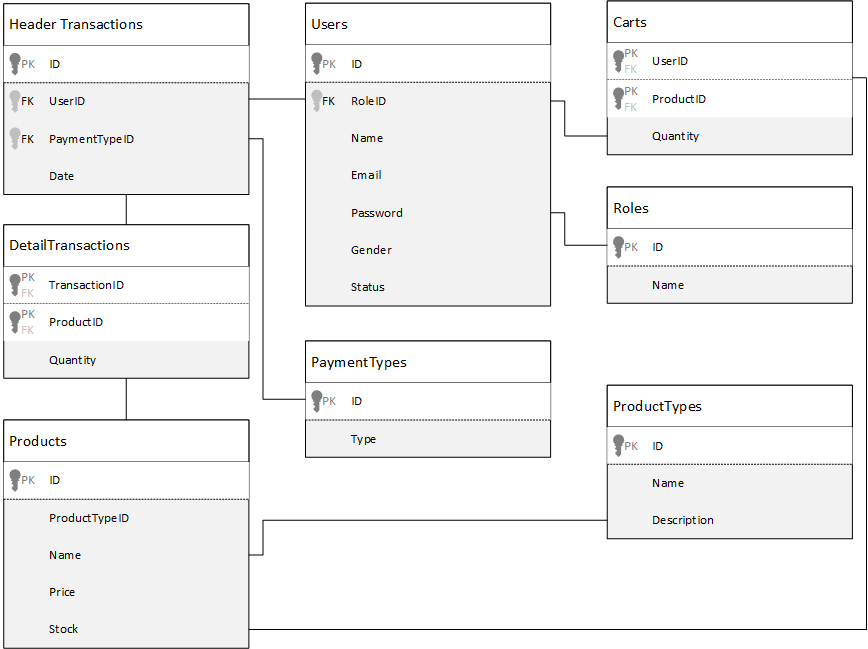
* **Factory**

You need to encapsulate all **complex object creation** in this layer. For example, when the client needs to create an aggregate object (an object that holds a reference to another object), the object factory must provide an interface for creating these objects. It is important to notes that an object returned by the factory must in **a consistent** state.

* **Model**

The model layer is responsible for **representing concepts** in the business or information about the business situation.

The following is the **Entity Relationship Diagram** for **TokoBeDia** **Database:**

****

There are three types of user roles on this website: **Administrator**, **Member**, and **Guest** (**non-logged-in user**). Below are the minimum pages you need to create for each role:

* **Administrator**
  + View User
  + View Payment Type
  + Insert Payment Type
  + Update Payment Type
  + View Product
  + Insert Product
  + Update Product
  + View Product Type
  + Insert Product Type
  + Update Product Type
  + Profile
  + Transaction Report
  + Update Profile
  + Change Password
* **Member**
  + View Product
  + View Cart
  + Update Cart
  + Add to Cart
  + Profile
  + Transaction History
  + Change password
* **Guest**
  + View Product
  + Login
  + Register

The descriptions for each page are as follows:

1. **Home**

This page is accessible by **all users**.There will be a welcome message containing the user’s username which will be **fetched using session**,displayed at the top of the homepage. **Guest** will be shown two buttons for redirect user, **View Product Button**,and **Login Button**.When the button is clicked, the user will be redirected to the respective pages based on the button.

Every logged-in user will have the following buttons available:

* **View Product**: Redirects user to View Product Page
* **View Profile**: Redirects user to Profile Page
* **Logout**:Logs out the user from the website and **destroys all cookie variables**

If the user is a member, the following buttons will become visible:

* **View Cart**: Redirects the user to View Cart Page
* **View Transaction History**: Redirects the user to View Transaction History

**Guest** and **member** will be shown 5 **randomly** picked products from the databaserecommended to them. Users with **member** roles also have an Add to Cart button at each table row**.** The Add to Cart Button will redirect the user to Add to Cart Page and the selected product’s ID will be passed using the URL’s query string parameter.

If the user is an **administrator**, the following buttons will become visible:

* **View User**: Redirects the user to View User Page
* **Insert Product**: Redirects the user to Insert Product Page
* **Update Product**: Redirects the user to Update Product Page
* **View Payment Type**: Redirects the user to View Payment Type Page
* **Insert Payment Type**: Redirects the user to Insert Payment Type Page
* **Update Payment Type**: Redirects the user to Update Payment Type Page
* **View Product Type**: Redirects the user to View Product Type Page
* **Insert Product Type**: Redirects the user to Insert Product Type Page
* **Update Product Type**: Redirects the user to Update Product Type Page
* **View Transaction Report**: Redirects the user to Transaction Report Page

1. **Login**

This page consists of a **Login Form**.Validate that this page is accessible only for **guest**. Display an error messageon the login form if the user enters the wrong combination of **Email** and **Password**.The following table shows existing fields in the login form and validation for each field:

|  |  |
| --- | --- |
| **Field** | **Information** |
| **Email** | Must be filled and appropriate with the data in the database |
| **Password** | Must be filled and appropriate with the data in the database |

Add another validation in which if the user is currently **banned** by the administrator,then the user **won’t** be permitted to log in. The login page will also have a **remember me** checkbox. The remember mecheckbox will save your login information by implementing a **cookie** which also has an **expired time** if a user had checked it so that the next time the user attempts another login, the **email** and **password** field will be **auto-filled** with the user’s previous login information, **fetched from the cookie**.

1. **Register**

This page allows guests to **register** themselves as **TokoBeDia** member. Validate that this page is accessible only for **guest**. Display an error messageon the registration form if the user inputs incorrect personal data. The following table shows existing fields in the registration form and validation for each field:

|  |  |
| --- | --- |
| **Field** | **Information** |
| **Email** | Must be filled and unique |
| **Name** | Must be filled |
| **Password** | Must be filled |
| **Confirm password** | Must be same with password |
| **Gender** | Must be chosen |

1. **View Product**

This page allows users to see products that are being sold on **TokoBeDia**.This page is accessible by **all users**. The information that must be shown is **Product ID**, **Product Name**, **Product Stock**, **Product Type**,and **Product Quantity**.

For **member** users, **Add to Cart** button will be visible. If users click Add to Cartbutton, the user will be redirected to **Add to Cart Page** and the selected product’s ID will be passedusingtheURL’s query string parameter. For userwith **administrator** role, **Insert Product**, **Update Product**, **and Delete Product** button will be visible. Below is the description for each button:

* + **Insert Product**: Redirects the user to Insert Product Page
  + **Update Product**: Redirects the user to Update Product Page and the selected product’s ID will be passed using the URL’s query string parameter
  + **Delete Product**:Deletes the selected product from the database and **refreshes** the data table.

Validate the to-be-deleted product **cannot** be referenced in another table in the database

1. **Insert Product**

This page is only accessible for **administrator.** This page allows an administrator to insert **a** new product into the database. The user will be redirected to this page when they clicked the Insert Product Buttonprovided at View Products Page**.** Display error message if the administrator inputted invaliddata.

When the user has inputted all of the fields correctly and pressedthe Insert Button, the inputted product information will be **added** and **saved** into the database. The following table shows existing fields on this page and the validation for each field:

|  |  |
| --- | --- |
| **Field** | **Information** |
| **Name** | Must be filled |
| **Stock** | Input must be 1 or more |
| **Price** | Input must be above 1000 and multiply of 1000 |

1. **Update Product**

This page is only accessible for **administrator.** This page allows an administrator to **update** **an existing** product in the database. The administrator will be redirected to this page when they clicked the Update Product Button provided at the View Product Page. All of the to-be-updated product’s current data will be shown on this page and the administrator can **change** those data before submitting it again by pressing the Update Button. The product’s data will be fetched from the database by using the product’s ID provided from the URL’s query string parameter. When the user has inputted all of the fields correctly and pressed the Update Button, the product’s information in the database will be **updated**. Display error message if the administrator inputted invaliddata.

The following table shows existing fields on this page and the validation for each field:

|  |  |
| --- | --- |
| **Field** | **Information** |
| **Name** | Must be filled |
| **Stock** | Input must be 1 or more |
| **Price** | Input must be above 1000 and multiply of 1000 |

1. **View Cart**

This page contains information about the member’s cart status. Validate this page is accessible only for **member**.This page displays **all products** that have been added into the current **logged-in user**’**s** shopping cart. The data that will be shown are **Product ID**, **Product Name**, **Product Price**, **Quantity**, and **Subtotal**.The page also shows the **Grand Total** of the cart.

The member can **update** items on cart, **delete** items from cart,and **checkout**. Below is the description for each feature:

* + **Update Product on Cart**: Redirects the user to Update Cart Page and the selected product’s ID will be passed using the URL’s query string parameter
  + **Delete Product on Cart**:Deletes the selected product from the database and **refreshes** the data table. Validate the to-be-deleted product **cannot** be referenced in another table in the database

To do a checkout, the membermust click the available checkoutbutton. The member **must** choose any **payment type** they want and you must validate that the user **cannot** checkout if the cart is **empty**. If the user success checkout, the user’s cart in the database will become **empty** and all data in the cart will be moved into **transaction table**

1. **Add to Cart**

This page is only accessible for **member**.This page allows a member to insert the quantityof products that he/she wants to add to the cart. This page alsoshowsall the informationabout the selected product,such as **Name**, **Price**, **Stock**,and **Product Type**. The product’s information will be fetched from the database by using the product’s ID provided from the URL’s query string parameter. Display an errormessage if the member inputted invalid quantity.

Required inputted **quantity validations** are as the following:

* + Must be numeric
  + Must be more than 0
  + Must be less than or equals to current stock
  + The inputted quantity and the reserved quantity in other carts combined altogether must not exceed the selected product’s current stock

Note that if the product is **already listed** in the shopping cart, its cart quantity will be **added** with the new quantity.When the user has inputted the quantity correctly and pressed the Add to Cart Button, the newly inputted cart information will be **added** and **saved** into the database, then the user will be redirected to View Cart Page.

1. **Update Cart**

This page is only accessible for **member** users. This page allows a member to **update the existing** cart item. The member will be redirected to his page when they clicked the Update button provided at the View Cart Page. All of the to-be-update cart item’s current data will be shown in this page and the member can change those data before submitting it again by pressing the Update button. The cart item’s data will be fetched from the database by using the cart item’s ID provided from the URL’s query string parameter. When the member has inputted all of the fields correctly and pressed the Update Button, the cart item’s information in the database will be **updated**. Validate that the input field **cannot** be inputted **below zero number**.If the member input 0, then the item will be **removed** from the cart. Display error message if the member inputted invalid data.

1. **View Product Type**

This page contains information about the product typethat is available on TokoBeDia. Validate that only **administrator** can access this page. There will be a table showing **Product Type ID**, **Product Type**,and **Description**.The administratorcan **insert** product type, **update** product type, and **delete** product type.Below is the description for each feature:

* + **Insert Product Type**: Redirects the user to Insert Product Type Page
  + **Update Product Type**: Redirects the user to Update Product Type Page and the selected product type’s ID will be passed using the URL’s query string parameter
  + **Delete Product Type**:Deletes the selected product type from the database and **refreshes** the data table. Validate the to-be-deleted product type **cannot** be referenced in another table in the database

1. **Insert Product Type**

This page is only accessible for **administrator.** This page allows an administrator to inserta new product type into the database. The user will be redirected to this page when they clicked the Insert Product Type Buttonprovided at View Product Type Page**.** Display error message if the administrator inputted invaliddata. When the user has inputted all of the fields correctly and pressedthe Insert Button, the inputted product type information will be **added** and **saved** into the database. The following table shows existing fields on this page and the validation for each field:

|  |  |
| --- | --- |
| **Field** | **Information** |
| **Product Type** | Must be filled, unique, and consists of 5 characters or more |
| **Description** | Must be filled |

1. **Update Product Type**

This page is only accessible for **administrator.** This page allows an administrator to **update** **an existing** product type in the database. The administrator will be redirected to this page when they clicked the Update Product Type Button provided at the View Products Type Page. All of the to-be-updated product type’s current data will be shown on this page and the administrator can change those data before submitting it again by pressing the Update Button. The product type’s data will be fetched from the database by using the product type’s ID provided from the URL’s query string parameter.

When the user has inputted all of the fields correctly and pressed the Update Button, the product type’s information in the database will be **updated**. Display error message if the administrator inputted invaliddata. The following table shows existing fields on this page and the validation for each field:

|  |  |
| --- | --- |
| **Field** | **Information** |
| **Product Type** | Must be filled, unique, and consists of 5 characters or more |
| **Description** | Must be filled |

1. **View Payment Type**

This page contains available payment type**s** in TokoBeDia.Validate that only **administrator** can access this page. There will be a table showing **the Payment Type ID** and **Payment Type** Name.The administratorcan **insert** a new payment type, **update** payment type,and **delete** payment type.

Below is the description for each feature:

* + **Insert Payment Type**: Redirects the user to Insert Payment Type Page
  + **Update Payment Type**: Redirects the user to Update Payment Type Page and the selected payment type’s ID will be passed using the URL’s query string parameter
  + **Delete Payment Type**:Deletes the selected product type from the database and **refreshes** the data table. Validate the to-be-deleted payment type **cannot** be referenced in another table in the database

1. **Insert Payment Type**

This page is only accessible for **administrator.** This page allows an administrator to inserta new payment type into the database. The user will be redirected to this page when they clicked the Insert Payment Type Buttonprovided at View Payment Type Page**.** Display error message if the administrator inputted invaliddata. When the user has inputted all of the fields correctly and pressedthe Insert Button, the inputted payment type information will be **added** and **saved** into the database. Required inputted **payment type validations** are as the following:

* + Must be filled
  + Must be unique
  + Consists of 3 characters or more

1. **Update Payment Type**

This page is only accessible for **administrator.** This page allows an administrator to **update** **the existing** payment type in the database. The administrator will be redirected to this page when they clicked the Update Payment Type Button provided at the View Payment Type Page. All of the to-be-updated payment type’s current data will be shown on this page and the administrator can **change** those data before submitting it again by pressing the Update Button. The payment type’s data will be fetched from the database by using the payment type’s ID provided from the URL’s query string parameter. When the user has inputted all of the fields correctly and pressed the Update Button, the payment’s information in the database will be **updated**. Display error message if the administrator inputted invaliddata. Required inputted **payment type validations** are as the following:

* + Must be filled
  + Must be unique
  + Consists of 3 characters or more

1. **View User**

This page is used to manage roles and status for the users that are registered to TokoBeDia.Validate that only **administrator** that can access this page. There are **2 status** that is available on **TokoBeDia**, **active** and **blocked**.

There will be a table showing users with their current role and status. The administrator can change the user's role and toggle his/her status (blocked to active and vice versa). Validate that the current administratorthat is logged in **cannot** changetheir role and status.

1. **Transaction History**

This page's purposes are for showing the transaction of the users that are registered to TokoBeDia. Validate that only **users** that are **logged in** can access this page.

For **administrator**,they can view all transactions of the users. As for **member**s,they can view all transactions that are made by the **member** that is currently logged in. Users also can view detail of the transaction, such as **Transaction Date**, **Payment Type**, **Product Name**, **Product Quantity**, and **Subtotal**. There will also be **View Transaction Report Button** when clicked, the user will be redirected to **Transaction Report Page**.

1. **Transaction Report**

This page is accessible for **administrator**.The administrator will be redirected to this page when they clicked the View Transactions Report Button provided at the Home Page. This page will display the report of all member's transactions in TokoBeDia using **Crystal Report**. The report will have to show each transaction’s **entire information** plus its **Sub Total** and **Grand Total** value.

1. **Profile**

This page is accessible for usersthat are logged in. This page will show current logged in userinformation such as **Email**, **Name,** and **Gender**. There will be an update profile button available to redirect the user to the update profile page.

1. **Update profile page**

This page is accessible for usersthat are logged in. On this page, the user can update their **Email**, **Name,** and **Gender**.There will be input fieldsthat are **already filled** with **current user information**.

The following table shows existing fields in the login form and validation for each field:

|  |  |
| --- | --- |
| **Field** | **Information** |
| **Email** | Must be filled and unique |
| **Name** | Must be filled |
| **Gender** | Must be chosen |

1. **Change Password Page**

This page is accessible for usersthat are logged in. On this page, the user can changetheircurrent password.There will be input fieldsconsists of **Old Password**, **New Password**, **Confirm Password**.The following table shows existing fields in the change password form and validation for each field:

|  |  |
| --- | --- |
| **Field** | **Information** |
| **Old Password** | Must be same with current password |
| **New Password** | Must be filled and cannot be the same with old password |
| **Confirm Password** | Must be same with new password |

**If you do not understand, please ask your assistant!**