Project Case	
COMP6115 Object Oriented Analysis & Design	BINUS UNIVERSITY Software Laboratory Center
Computer Science	**C202-COMP6115-SW02-PPTI 8**
***Valid on** Compact 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.

2. 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"

3. 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

4. Persentase penilaiaan untuk matakuliah ini adalah sebagai berikut:
   Marking percentage for this subject is described as follows:

UTP *Mid Exam*	**UAP** *Final Exam*
40%	60%

5. 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 Neon 3
XAMPP 7.47
MySQL Java Connection Library 5.1.49

6. Ekstensi file yang harus disertakan dalam pengumpulan utp dan uap untuk matakuliah ini adalah sebagai berikut:
    File extensions should be included in assignment, project, and final exam collection for this subject are described as follows:

UTP	**UAP**
Mid Exam	*Final Exam*
VPP	JAVA, CLASS, SQL

Soal
Case

SWeeBook Library

SWeeBook Library is a new library in Jakarta, Indonesia that is opened by Mr. Kevin Sudut. As the business grows, Mr. Kevin plans to improve its service by developing a system that can help to manage their business activity within the library. Below is the business flow that is required to be covered by the system.

- **Membership**

    The membership role is created for the people that use the service at SWeeBook library to borrow the available books in the library. Before using the service, the member must be registered on the system.

Variable	Validation
Member Name	- Cannot be empty
Member Username	- Cannot be empty - Must be unique
Member Gender	- Cannot be empty - Must be between "Male" and "Female"
Member Address	- Cannot be empty

    After the member has been registered to the system, the member can view all the books in the library and borrow them. The member can borrow as many as **10 books** including the book that has not been returned. Besides that, the member cannot borrow the same book, if the book has not been returned. For the return book process, the member can return the book one by one, although in one transaction the member can borrow more than one book. However, the library has a fine when the members are late to return the book. The member can borrow a book for two weeks and will pay the fine with a total of **1000 IDR** for each delay day.

    Also, the member can view borrow book status and the borrow book history with the detail book items of each transaction.

- **Administrator Staff**

    The employee with the administrator role can view all the books in the library and all members that is registered in the system. Besides that, the staff can view the report of borrowing transactions that has been accepted or still pending requests. For the accepted transaction, the staff can choose month and year when the borrowing transaction occur, and the system will show all borrowed history based on the chosen month and year. To view all the borrowed book items for each transaction, the staff may click the borrowed history cell in the table and the system will show all the borrowed book items based on the chosen borrowed id in the cell of the table.

Variable	Validation
Month	- Must be valid month
Year	- Must be valid year

    For pending requests, the system will show all pending transactions that need to be accepted by the staff. The staff can also view all the borrowed book for each pending transaction.

- **Purchasing Staff**

    The employee with the purchasing role is able to view all books and each product that has its own information. Sometimes, the staff can insert or update information about the book. The staff inputs all information about the book and check the ISBN number has been registered or not. If the ISBN number exists, then update the information about the book and add the quantity on the system with input quantity. Otherwise, the system will insert a new book record. The staff also can delete the book when the book is no longer exists in the library or borrowed books record.

Variable	Validation
Book Title	- Cannot be empty
Book Genre	- Cannot be empty - Must be exists
ISBN Number	- Must be exists (for delete book) - Must be numeric - Length must be between 10 and 13
Quantity	- Must be numeric - Must be above zero

    Sometimes, the staff purchases a new book that genre type does not exist in the system, so the staff can insert a new genre to the system then the staff can add a new book to the system and view available genre type in the system.

Variable	Validation
Genre Type	- Cannot be empty - Must be unique

- **Human and Resource Management Staff**

  Human and resource management staff are responsible to handle all the employees data. They can view all employees and hired a new employee. When the staff hired a new employee, they must tell the manager to approve the request.

Variable	Validation
Employee Name	- Cannot be empty
Employee Username	- Cannot be empty - Must be unique
Employee Gender	- Cannot be empty - Must be between "Male" and "Female"
Employee Salary	- Cannot be empty - Must be numeric - Must be above 10000

- **Manager**

  To make the business run smoothly, Kevin will hire a manager for supervising and motivating employees. The manager can view report of the borrowing transactions generated by the system. The transaction report includes all borrowed history recorded by the system each month for each year. The manager needs to choose month and year when the borrowed happened, and the system will show all borrowed history based on the chosen month and year. To view all the borrowed book items each transaction, the manager may click the borrowed history cell in the table and the system will show all the borrowed books based on the chosen borrowed id in the cell.

Variable	Validation
Month	- Must be valid month
Year	- Must be valid year

  Besides that, the manager can also view employees, hired employees, accept a new employee request, fired employees, and view membership.

Variable	Validation
Employee Id (Accept & Fired)	- Must be chosen - Must be exists
Employee Status	- Must be "Pending" (for accept employee) - Must be "Active" (for fired employee)

***

To help you design the program, Mr. Kevin already hire Budi, a System Analyst, to provide you with the design of the program based on the requirement and the business flow above. You can open "**sweebook.vpp**" to view the diagram designed by Mr. Budi.

***

As a programmer, Mr. Kevin 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, the 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 engine must be using **MySQL** and using the provided **SQL file**
- The application must have an **authenticated user based on roles**
- For better security, all **id** must be using **UUID** (**Universally Unique Identifier**) and **do not store the password as plain text**
- The application must minimalize human error with great user experiences
- The application must be made based on the analysis diagram that provided by Mr. Budi, 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.