|  |  |
| --- | --- |
| **Project Case** |  |
| ISYS6197 | ISYS6197003  Business Application Development |
| **Computer Science** | **O222-ISYS6197-DP03-00** |
| ***Valid on*** *Odd Semester Year 2021/2022* | **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* |
| Java 8  Eclipse Neon 2  XAMPP 3.22 (MySQL)  MySQL Java Connection Library |

1. 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* |
| JAVA, CLASS | JAVA, CLASS, SQL, image files(JPG/PNG/GIF) |

## Soal

*Case*

**DoPe Game Store**

**DoPe Game Store** is a shop that sells many types of game. To manage all the transactions and manage system of the beverage, DoPe Game Store’s owner asked you to create the program for the store. The program should be made using **Java Programming Language**.

There are some requirements to build this application:

Diagram

Description automatically generated

Execute the provided create and insert query file on “**dopegamestore**” database! (“create+insert.sql”)

1. **Login Form**

This is the first form which will appear when the program is run. **Login** **Form** is used by users to **Login** so the users can access the **Main** **Form**.

Graphical user interface, application

Description automatically generated

Figure 1. Login Form

**Components**:

* **JLabel** for Title, Email, Password, Sign Up.
* **JTextField** for input Email.
* **JPasswordField** for input Password.
* **JButton** for Login.

**Description**:

* If user clicks **Login** button, then **validate**:
  + **Email** field **must** **be** **filled**.
  + **Password** field **must** **be** **filled**.
  + **Email** and **Password** **must** be **correct** according to the **existing data in database**.
* If user has successfully signed in, **Login Form** will be closed and **Main Form** will be displayed.
* If user failed to login then display warning message.
* If user click “**Sign** **up** **here**.” label, then program will close **Login** **Form** and open **Register** **Form**.

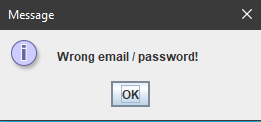


Figure 2. Failed Login Warning

(**additional** **note**: **every** **error** occurrence will be shown on error message **JOptionPane**).

1. **Register Form**

**Register Form** is used by users to **Register**.

Graphical user interface

Description automatically generated

Figure 3. Register Form

**Components**:

* **JLabel** for Title, ID, User Name, Email, Phone, Address, Password, Gender, Role, and Sign In.
* **JTextField** for input ID (**read**-**only**), User Name, Email, and Phone.
* **JTextArea** for input Address.
* **JPasswordField** for input Password.
* **JRadioButton** for input Gender (“**Male**” or “**Female**”).
* **JComboBox** for choose user Role which has 2 options:
  + **Admin**
  + **Customer**
* **JButton** for Register.

**Description**:

* If user clicks **Register** button, then **validate**:
  + **ID is generated** when Register Form is opened with **format**:

**USXXX**

Where **X**: Increment digit

Example: US001, US002, US003

* + **User** **Name** must be between **5 - 30 characters**.
  + **Email** must be in valid **format**:

**[email]@[provider].[domain]**

* + - Character ‘@’ must **not be next to** ‘.’.
    - **Must not starts** and **ends** with ‘@’ nor ‘.’.
    - **Must contain** **exactly** **one** ‘@’.
    - **Must contain** **exactly** **one** ‘.’ after ‘@’ for separating [**provider**] and [**domain**].

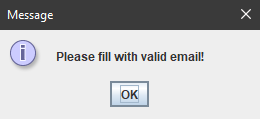


Figure 4. Invalid Email Format Warning

* + **Phone** number must be numeric and more than equals **12 digits**.
  + **Address** must consist of **10 or more characters** and **ends with** ‘ Street’.
  + **Password** must **5 - 30 length of character** and **digit** (must at least contain 1 character and 1 digit).
  + **Gender** must be selected either “**Male**” or “**Female**”.
  + **Role** must be chosen either “**Admin**” or “**Customer**”.

(**additional** **note**: **every** **error** occurrence will be shown on error message **JOptionPane**).

* If user successfully register, **insert** new **user** data **to database**, then program will display a success insert message. Lastly, program will close **Register** **Form** and open **Login** **Form**.

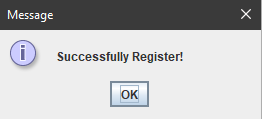


Figure 5. Register Success Message

1. **Main Form**

This is the main form for the program. At this form the user can select menu from the menu bar, such as **User** **Menu** (**Profile** and **Exit**), **Transaction** **Menu** (**Buy Game** and **View Transaction History**) and **Manage** **Menu** (**Manage Game**). The menu that will show is determined by the user role. If the user role is “**Customer**”, then the menus that available are **User** **Menu** and **Transaction** **Menu** only. If the user role is “**Admin**”, then the menus that available are **User** **Menu** and **Manage** **Menu**.

Graphical user interface, text, application, email

Description automatically generated

Figure 1. Admin Main Form

Graphical user interface, text, application, email

Description automatically generated

Figure 2. Customer Main Form

**Components**:

* **User** Role:
  + **JMenu** for Menu Profile and Menu Transaction.
  + **JMenuItem** for Edit Profile, Exit, Buy Game, and View Transaction History.
* **Admin** Role:
  + **JMenu** for Menu Profile and Menu Manage.
  + **JMenuItem** for Edit Profile, Exit and Manage Game.

**Description**:

There are some **JMenuItem** that available (**to be clicked**):

* **Both** Role
  + If user clicks **Edit Profile** Menu Item, then the program will open **Profile Form**.
  + If user clicks **Exit** Menu Item, then the program will be **terminated**.
* **Customer** Role

There will be two **JMenu** displayed:

* + - * **Profile**

Graphical user interface

Description automatically generated with medium confidence

Figure 3. Customer: Profile Menu

There are 2 **JMenuItem** available **(to be clicked**) in **Profile Menu** (**Edit Profile** and **Exit**).

* + - * + If user clicks **Edit Profile**, program will open **Profile** **Form** inside the **Main** **Form**.
        + If user clicks **Exit**, program will be **terminated**.
      * **Transaction**

Graphical user interface, application

Description automatically generated

Figure 4. Customer: Transaction Menu

There are 2 **JMenuItem** available (**to be clicked**) in **Transaction Menu** (**Buy Game** and **View Transaction History**).

* + - * + If user click **Buy Game**, program will open **Buy Game Form** inside the **Main Form**.
        + If user click **View Transaction History**, program will open **Transaction Form** inside the **Main** **Form**.
* **Admin** Role

There will be two **JMenu** displayed:

* + - * **Profile**

A picture containing graphical user interface

Description automatically generated

Figure 5. Admin: User Menu

There are 2 **JMenuItem** available (**to be clicked**) in **Profile** **Menu** (**Profile** and **Exit**).

* + - * + If user clicks **Edit Profile**, program will open **Profile** **Form** inside the **Main** **Form**.
        + If user clicks **Exit**, program will be **terminated**.
      * **Manage**

Diagram

Description automatically generated with low confidence

Figure 6. Admin: Manage Menu

There are 1 **JMenuItem** available (**to be clicked**) in **Manage** **Menu** (**Manage** **Game**).

* + - * + If user clicks **Manage** **Game**, program will open **Manage** **Game Form** inside the **Main** **Form**.

1. **Buy Game Form**

This form **displays all the currently available game**. In this form, user can click any product to see the game detail and add it to cart. But if the game Stock is 0, so the game detail won’t be shown in the form and can’t be added to cart. In this form, cart is also shown at the bottom view of the form. This form is made with **JInternalFrame**.

Graphical user interface

Description automatically generated with medium confidence

Figure 13. Buy Game Form

**Components**:

* **JTable** for Game Table and Cart Table.
* **JScrollPane** for Game and Cart Table Container.
* **JLabel** for Title, Game ID, Game Name, Game Type, Game Price, Game Stock, and Game Quantity
* **JTextField** (**read**-**only**) for Game ID, Game Name, Game Type, Game Price, and Game Stock
* **JSpinner** for input Game Quantity.
* **JButton** for Add to Cart, Remove Selected Cart, Clear Cart, and Checkout.

**Description**:

User can **only edit quantity field** and the **other fields can’t be edited**. If user clicks **Add to Cart** button, program will **validate** is there any **game chosen** or not. If the game already chosen, then program will **validate** the **quantity** that **can’t be less than 1** and **can’t be more than its Game Stock**. If user choose a game that currently **stock is 0**, then program will **show warning message**.

Graphical user interface, application, Word

Description automatically generated

Figure 14. Choose Beverage Message

Graphical user interface, application

Description automatically generated

Figure 15. No Stock Message

If all conditions are fulfilled, then the game will be added to the user’s cart:

* If the game **hasn’t existed yet**, **insert** new **game** to user’s cart and show message.

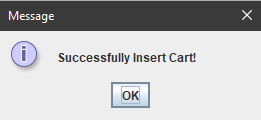


Figure 16. Success Insert Cart Message

* If the game **has existed**, **update** game **quantity** **with the addition of quantity**
* User can also remove game form cart by using **Remove Selected Cart** button. **Validate** that the **game** from cart **must be chosen** first by click the game row that want to be deleted. After remove game from cart, shows message.
* If user clicks **Clear Cart** button, program will **ask** **confirmation** first and if user choose **yes**, then program will **clear the cart**.

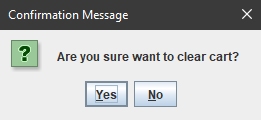


Figure 18. Clear Cart Confirmation

* If user clicks **Checkout** button, program will **ask** **confirmation** first and if user choose **yes**, then program will **decrease Game Stock** equals to the quantity, **clear the cart** and **insert all beverage** to **Transaction** (**Header** and **Detail**) and shows message.

Graphical user interface, application, Excel

Description automatically generated

Figure 19. Checkout Confirmation

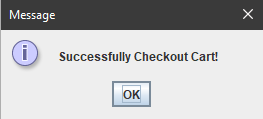


Figure 20. Checkout Success Message

(**additional** **note**: **every** **error** occurrence will be shown on error message **JOptionPane**).

1. **Transaction Form**

**Transaction** **Form** allows user to **view all the transaction history** or all transaction that user has made. This form is made with **JInternalFrame**.

Graphical user interface, application

Description automatically generated

Figure 21. Transaction Form

**Components**:

* **JLabel** for Title, Selected ID, Grand Total.
* **JTable** for Header Transaction Table and Detail Transaction Table.
* **JScrollPane** for Header and Detail Transaction Table container.
* **JTextField** (**read**-**only**) for Selected ID and Grand Total.

**Description**:

If user **clicks** one of the **header transaction table**, it will **show** all the **detail** **transaction** that user bought on that transaction. It also **triggers** program to **update** the **Selected** ID based on user clicks and **Grand** **Total** by **sum** all the **Sub** **Total** of the detail transaction. **Sub** **Total** in detail transaction is obtained from **calculation** between **Quantity** times **Beverage** **Price**.

Graphical user interface

Description automatically generated with medium confidence

Figure 22. Detail Transaction View

1. **Manage Game Form**

**Manage** **Game Form** is made for admin to manage game. Admin can do **insert** new game, **update** any existing game data, **remove**/**delete** game, and **add** **game stock**. This form is made with **JInternalFrame**.

Graphical user interface, application, table, Excel

Description automatically generated

Figure 23. Manage Game Form

**Components**:

* **JLabel** for Title, New Game ID, New Game Name, New Game Type, New Game Price, New Game Stock, Game ID, Game Name, Game Type, Game Price, Game Stock, and Add Stock.
* **JTable** for Game Table.
* **JScrollPane** for Game Table container.
* **JTextField** for input Game ID (**read**-**only**), Game Name, Game Price, Game Stock (**read**-**only**), New Game ID (**read**-**only**), New Game Name, and New Game Price.
* **JComboBox** for choose Game Type and New Game Type.
* **JSpinner** for input Add Stock and New Game Stock.
* **JButton** for Update Game, Delete Game, Add Stock, Insert Game, and Reset.

**Description**:

* All the fields can be edited **except** for Game ID, New Game ID, and Game Stock.
* Admin can do **Insert**, **Update**, **Delete**, and **Add Stock** Game from database.
  + If admin clicks Update Game button, then **validate**:
    - **Game** **ID** is **not** **empty** (by click a row in Game table).

Graphical user interface, application, Word

Description automatically generated

Figure 24. Choose Beverage Message

* + - **Game** **Name** must consist of **5 - 30 characters**.
    - **Game Type** must be **chosen** either **TPS**, **MOBA**, **FPS** or **RPG**.
    - **Game Price** must more than **0 and numeric**.
  + If all conditions for **Update** **Game** are fulfilled, then program will **ask** **confirmation** first and if use choose **yes**, program will **update** **game’s** **data** **in** **database**.

Graphical user interface, application

Description automatically generated

Figure 25. Update Game Confirmation

* + If admin clicks **Remove** **Game** button, then **validate**:
    - **Game ID** is **not** **empty** (by click a row in beverage table).
  + If all conditions for **Remove** **Game** are fulfilled, then program will **ask** **confirmation** first and if use choose **yes**, program will **delete** **game from** **database**.

Graphical user interface, application

Description automatically generated

Figure 26. Remove Game Confirmation

* + If admin clicks **Add** **Stock** button, then **validate**:
    - **Game** **ID** is not **empty** (by click a row in beverage table).
    - **Add** **Stock** must be more than **0**.
  + If all conditions for **Add** **Stock** are fulfilled, then program will **ask** **confirmation** first and if use choose **yes**, program will **update game’s stock in database with the addition of Added Stock**.

Graphical user interface, application, Word

Description automatically generated

Figure 27. Add Game Stock Confirmation

* + If admin clicks **Insert** **Game** button, then **validate**:
    - **New Game ID** is generated when **Manage** **Beverage** **Form** open with format:

**GAXXX**

Where **X**: Increment digit

Example: GA001, GA002, GA003

* + - **New Game Name** must consist of **5 - 30 characters**.
    - **New Game Type** must be **chosen** either **TPS**, **MOBA**, **FPS** or **RPG**.
    - **New Game Price** must be more than **0 and numeric**.
    - **New Game Stock** must be more than **0 and numeric**.
  + If all conditions for **Insert** **Game** are fulfilled, then program will **ask** **confirmation** first and if use choose **yes**, program will **insert** **new** **game to** **database**.

Graphical user interface, application

Description automatically generated

Figure 28. Insert New Game Confirmation

* + If admin clicks **Reset** button, then **reset all field**.

(**additional** **note**: **every** error occurrence will be shown on error message **JOptionPane**).

1. **Edit Profile Form**

**Edit Profile** **Form** allows users to see and update their profile. This form is made with **JInternalFrame**.

Graphical user interface, application

Description automatically generated

Figure 29. Edit Profile Form

**Components**:

* **JLabel** for Update Profile Title, Username, User Email, User Phone, User Address, User Gender.
* **JTextField** for input Username, User Email, and User Phone.
* **JTextArea** for input User Address.
* **JRadioButton** for choose User Gender (“Male” or “Female”).
* **JButton** for Update Profil.

**Description**:

* If user clicks **Update** **Profile** button, program will **validate**:
  + **Username** must be between **5 - 30 characters**.
  + **Email** must be in valid **format**:

**[email]@[provider].[domain]**

* + - Character ‘@’ **must not be next to** ‘.’.
    - **Must not starts** and **ends** with ‘@’ nor ‘.’.
    - **Must contain exactly one** ‘@’.
    - **Must contain exactly one** ‘.’ after ‘@’ for separating [**provider**] and [**domain**].
  + **Phone** number must be **numeric** and more than equals **12 digits**.
  + **Address** must consist of **10 or more characters** and **ends** **with** ‘ Street’.
  + **Gender** must be **selected** **either** “**Male**” or “**Female**”.
* If all conditions for **Update** **Profile** are fulfilled, then program will **ask** **confirmation** first and if user choose **yes**, program will **update** **user’s** **data** **in** **database**.

Graphical user interface, application

Description automatically generated

Figure 30. Update Profile Confirmation

(**additional** **note**: **every** **error** occurrence will be shown on error message **JOptionPane**).

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. Additional rules:
   * All connections and data manipulation to the database must use the JDBC driver. The database file (using Microsoft Access 2003, .mdb) must be submitted as well.
   * Use **LayoutManager** to design your application (You are **not allowed** to use **free** / **absolute** / **null** layout!).
   * All validations must use the techniques taught during practicum. (Another language or Regular Expression is prohibited).
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.

**If there is something you don’t understand, feel free to ask your Assistant**