|  |  |
| --- | --- |
| **Project Case** |  |
| ISYS6197 | ISYS6197003  Business Application Development |
| **Computer Science** | **E222-ISYS6197003-WT01-00** |
| ***Valid on*** *Even 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 another 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. 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* |
| Eclipse 2020.6  Java 8  MySQL Java Connection Library 8.0.24  XAMPP 8.0.7 |

## Ekstensi file yang harus disertakan dalam pengumpulan tugas mandiri, proyek 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:*

|  |  |  |
| --- | --- | --- |
| **Tugas Mandiri**  *Assignment* | **Proyek**  *Project* | **UAP**  *Final Exam* |
| JAVA. CLASS | JAVA. CLASS, SQL | - |

Soal

*Case*

**JeTSki Cafe**

**JeTSki Café** is a café that have a lot of customer. To manage all of the customer and transactions, its owner asked you to create an application for it. The program should be made using **Java Programming Language**, with **MySQL Database Engine**.

There are some requirements to build this application:

Diagram

Description automatically generated

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

1. **Login Form**

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

Graphical user interface, text, application

Description automatically generated

Figure 1. Login Form

**Components**:

* **Text** for Error, Username, and Password.
* **TextField** for Username input.
* **PasswordField** for Password inputs.
* **Button** for Login and Register buttons.

**Description**:

* If the user clicks the **Login** button, then **validate**:
  + **Username** and **Password** pairing must **exist in the database**.
* If the user has successfully signed in, display a **Notification** **Alert**.

Graphical user interface

Description automatically generated with medium confidence

Figure 2. Login Successful Notification

* + After the user closes the alert, the program will **validate** role (either ‘**user**’ or ‘**administrator**’). Lastly, the **Login** **Form** will be closed, and the **Main** **Form** will be displayed.
* If the user failed to log in, display an **Error Text**.
* If the user clicks the **Register** button, the program will close the **Login** **Form**, and opens the **Registration** **Form**.

Graphical user interface, text, application, chat or text message

Description automatically generated

Figure 3. Login Failed Error

(**Additional** **note**: **every** **error** occurrence will be shown on an **Error Alert**).

1. **Registration Form**

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

Graphical user interface

Description automatically generated

Figure 4. Registration Form

**Components**:

* **Text** for Email, Name, Password, Confirm Password, Age, Gender, Male, Female.
* **TextField** for Email input, Name input
* **PasswordField** for Password and the confirm password input.
* **Spinner** for Age and validate the spinner must be 12 - 99
* **RadioButton** for Gender (‘**Male**’ or ‘**Female**’) input.
* **Button** for Register and Login buttons.

**Description**:

* If the user clicks **Register** button, then **validate**:
  + **User ID is incremented by 1** from the last User ID. If there’s no any user set the User ID to 1
  + Validate there’s no empty **field.**
  + **Username** must be between **5 - 30 characters**.
  + **Password** must be between **5 - 20 characters**.
  + **Password** cannot be same with **Username**
  + **Gender** must be selected, either ‘**Male’** or ‘**Female**’.

Graphical user interface

Description automatically generated

Figure 5. Invalid Username Format Error

(**Additional** **note**: **every** **error** occurrence will be shown on an **Error Text**).

* If the user successfully registers, **insert** a new **user** **to the database**. Program will display a **Notification Alert**. Lastly, the program will close the **Registration** **Form** and open the **Login** **Form**.

Graphical user interface, application

Description automatically generated

Figure 6. Register Success Notification

1. **Main Form**

This is the main form for the program. In this form, the user can select menus from the menu bar, such as the main **User (Logout, Exit),**and **Transaction** (**Order Menu, Transaction**). These menus are all available for the user role ‘**user**’. However, if the user role is ‘**admin**, the menu is **Admin (Logout, Exit)** and **Manage (Manage Menu, Manage User)**

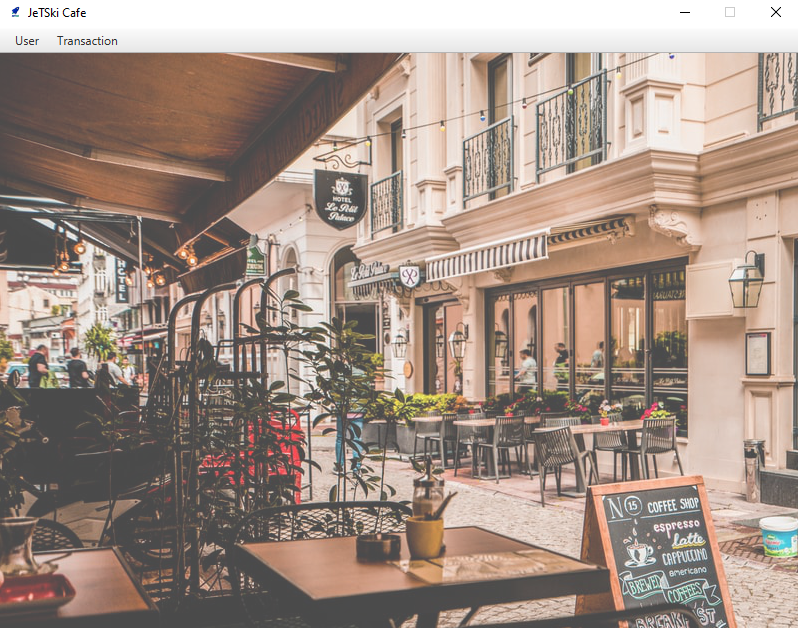


Figure 7. Main Form

**Components**:

* **‘user’** Role:
  + **Menu** for User and Transaction.
  + **MenuItem** for Logoff, Exit, Order Menu, Transcation.
* **‘administrator’** Role:
  + **Menu** for Admin and Manage.
  + **MenuItem** for Logoff, Exit, Manage Menu, Manage User.

**Description**:

* + If the user clicks the **Logoff** menu item, the program willclose the **Main Form**, and opens the **Login Form.**
  + If the user clicks the **Exit** menu item, the program will terminated.
  + If the user clicks the **Order Menu** menu item, the program will open the **Menu Form**.
  + If the user clicks the **Transaction** menu item, the program will open the **Transaction** **History Form**.

Graphical user interface, application

Description automatically generated

*Figure 8. ‘user’ role Menu items*

Graphical user interface, application

Description automatically generated

*Figure 9. ‘administrator’ role Menu items*

Graphical user interface

Description automatically generated with medium confidence

*Figure 10. ‘user’ role Menu items*

Graphical user interface, waterfall chart

Description automatically generated

*Figure 11. ‘administrator’ role Menu items*

1. **Order Menu**

This form **manages all of the menu transaction processes**. In this form, the user can click any menu and add it to the cart. In this form, cart is also shown at the bottom view of the form.

Graphical user interface

Description automatically generated

Figure 12. Order Menu Form

**Components**:

* **TableView** for Menu Tables and Cart Tables.
* **Spinner** for Menu quantity (1-99).
* **Button** for Add to Cart, Remove from Cart, and Order buttons.

**Description**:

* **Add to Cart**
  + **Adding to Cart** requires the user to first **select an menu from the Menu Table**, and **inputting the number of quantities** to purchase.
  + **Validate:**
    - If the user **did not select an menu from the Menu Table**, show an **Error Alert**.
    - If the **selected menu is already exists in the Cart Table**, show an **Error Alert**.

Graphical user interface, text, application

Description automatically generated

*Figure 13. Menu already in the cart*

Graphical user interface, application

Description automatically generated

*Figure 14. Menu not selected*

* + **Add the selected menu** into the user’s cart.

Table

Description automatically generated

*Figure 15. Cart Table after Add to Cart (added 2 menu)*

* **Remove from Cart**
  + **Removing from Cart** requires the user to first **select an menu from the Cart Table**, and **inputting the number of quantity to remove**.
  + **Validate:**
    - If the user **did not select an menu from the Cart Table**, show an **Error Alert**.
  + **Remove the selected cart from database**

Graphical user interface, text, application

Description automatically generated

*Figure 16. Cart Menu not selected Error*

Table

Description automatically generated

*Figure 17. Cart Table after Remove from Cart (removed 1 menu)*

* **Order**
  + **Ordering** requires the user to have menu inside their cart.
  + **Validate:**
    - If the user **doesn’t have any menu in their cart**, show an **Error Alert**.
  + **Add new transactions** **and transaction details** from the items in the cart.
  + **Decrease menu stock** based on cart quantity.
  + Finally, **remove all the menu** in the cart**.**

Graphical user interface, application

Description automatically generated

*Figure 18. Empty Cart Error*

Graphical user interface, application

Description automatically generated

*Figure 19. Cart Table after Order*

(**Additional** **note**: **every** **error** occurrence will be shown on an **Error Alert**)

1. **Show Transaction**

This page **show all of the transaction owned by the user**. In this form, the user can click any menu to see its details, Usercanalso **remove the transaction**

Graphical user interface, application, table

Description automatically generated

Figure 20. Show Transaction

**Components**:

* **TableView** for Transaction Table
* **TextField** for Menu Name(**read**-**only**), Menu Type(**read**-**only**), Menu Price(**read**-**only**), Quantity (**read-only**).
* **Button** for Remove Transaction

**Remove Transcation** :

* + Validate **Remove Transcation** must selected at least one transaction

Graphical user interface, text, application

Description automatically generated

*Figure 21. selected transaction not exists*

1. **Manage Menu Form**

This form **manages all of the available menu**. This menu is only available for admin roles. In this form, the admin can click any menu to see its details, **publish a new menu, update, or delete** an existing menu.

Table

Description automatically generated

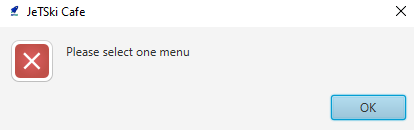
Figure 22. Menu Manager Form

**Components**:

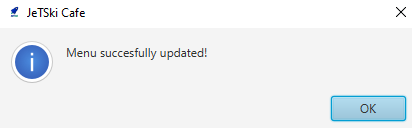
* **TableView** for Menu table.
* **TextField** for Menu ID (**read**-**only**), Menu Price, Menu Name, Menu Stock.
* **Text** for User ID, User Email, User Email, User Name, User Password, User Gender, User Age, User Role, Male, Female.
* **ComboBox** for Menu Type.
* **Button** for Insert New Menu, Update Menu, Remove Menu, Insert, Cancel.

**Description**:

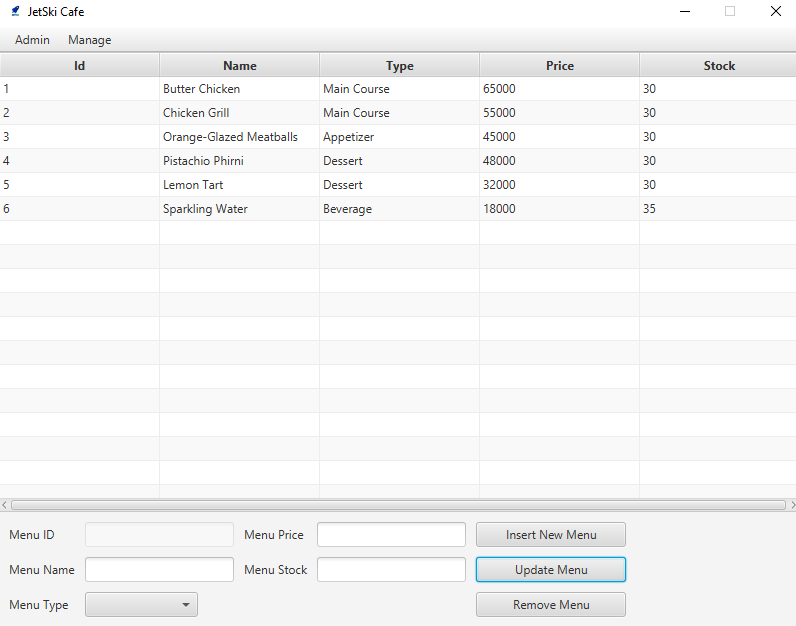
* This page only available for **admin** role
* **Create New Menu**
  + If admin click Create New Menu. Show the extra button (Insert, and Cancel Button).
  + Generate new ID for the Menu ID. **Menu** **ID is incremented by 1** from the last Menu ID. If there’s no any menu, set the Menu ID to 1
* **Update Menu**
  + **Validate:**
    - If there’s an empty field, show an **Error Allert.**
    - If there’s not selected menu, show an **Error Alert**.
    - If the **Menu Name** **length is less than 5 and more than 25**, show an **Error Alert.**
    - If the **Price is not numeric,** show an **Error Alert.**
    - If the **Price is 0,** show an **Error Alert.**
    - If the **Stock is not numeric,** show an **Error Alert.**
    - If the **Stock is 0,** show an **Error Alert.**
  + **Update the menu** using the provided data.
  + Finally, show a **Notification Alert**.



*Figure 23. Menu not picked Error*

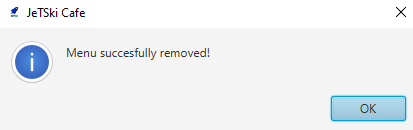


*Figure 24. Menu successfully updated*

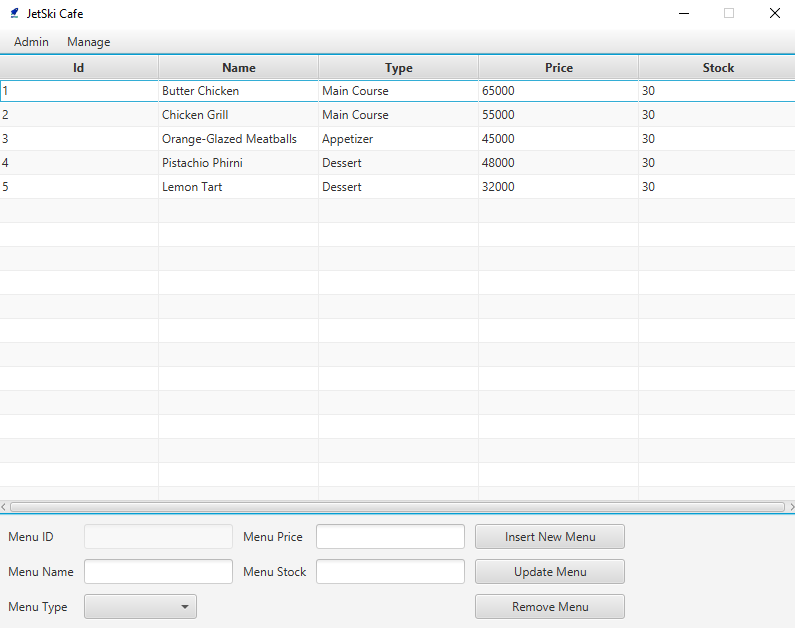


*Figure 25. Menu Updated*

* **Remove Menu**
  + **Validate:**
    - If an **menu is not selected**, show an **Error Message.**
  + **Remove the menu** from the database.
  + Finally, show a **Notification Alert**.



*Figure 26. Deleted Menu Notification*



*Figure 27. Menu Deleted*

* **Insert**
  + **Validate:**
    - If there’s empty fields, show an **Error Alert.**
    - If the **Menu Name** **length is less than 5 and more than 25**, show an **Error Alert.**
    - If the **Price is not numeric,** show an **Error Alert.**
    - If the **Price is 0,** show an **Error Alert.**
    - If the **Stock is not numeric,** show an **Error Alert.**
    - If the **Stock is 0,** show an **Error Alert.**
  + **Insert the menu** into the database.
  + Don’t forget to hide the insert and cancel button.
  + Finally, show a **Notification Alert**.

Graphical user interface, application

Description automatically generated

*Figure 28. Empty Field Error*

Graphical user interface, application

Description automatically generated

*Figure 29. Invalid Name Length*

Graphical user interface, application

Description automatically generated

*Figure 30. Invalid price*

Graphical user interface, text, application

Description automatically generated

*Figure 31. Invalid price value*

Graphical user interface, application

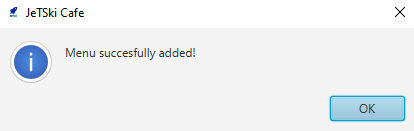
Description automatically generated

*Figure 32. Invalid stock*

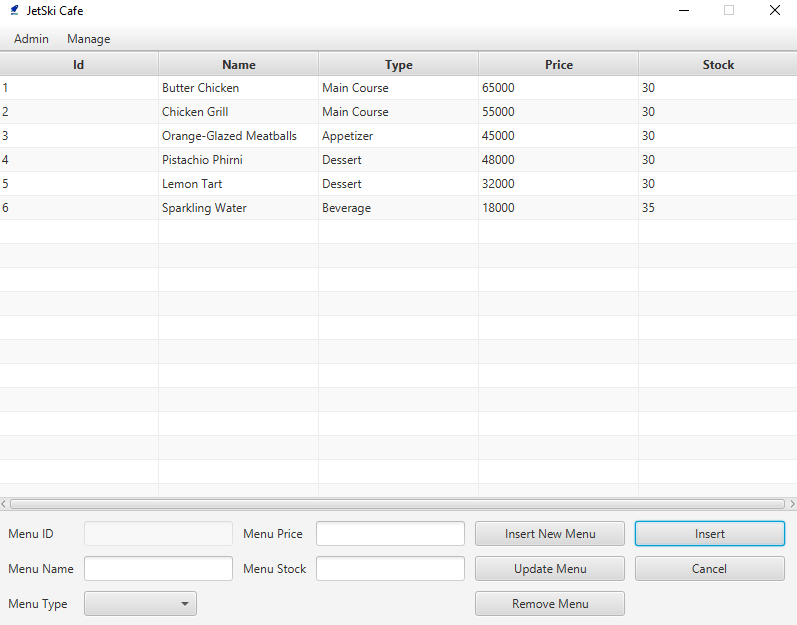
Graphical user interface, text, application

Description automatically generated

*Figure 33. Invalid stock value*



*Figure 34. Published Menu Notification*



*Figure 35. New Menu published*

* **Cancel**
* If user click cancel empty **all fields**, and hide **insert button** and **cancel button**

1. **Manage User Form**

This form **manages all of the available of user**. This menu is only available for admin roles. In this form, the admin can click any user to see its details, **make a new user, update, or delete** an existing user.

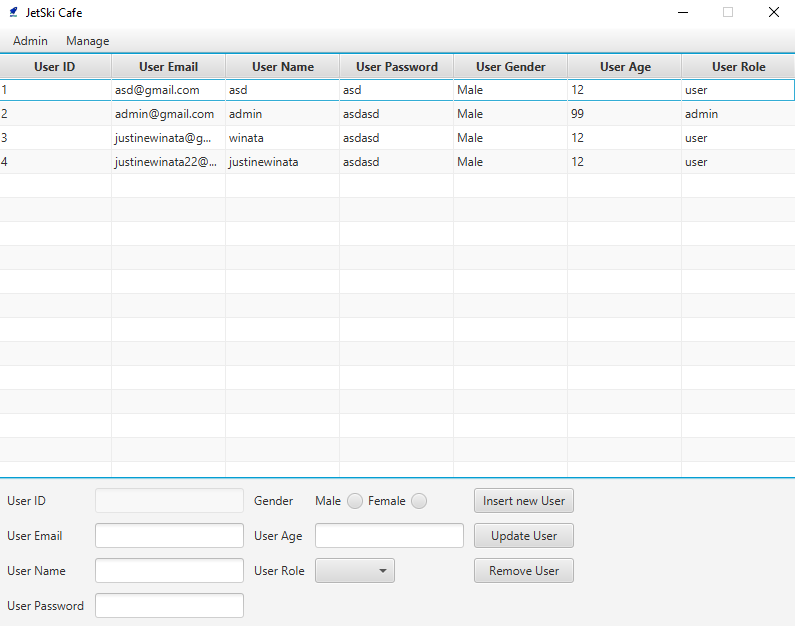


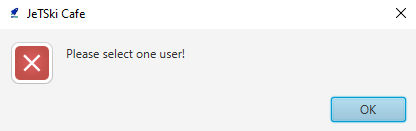
Figure 36. User Manager Form

**Components**:

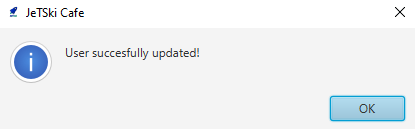
* **TableView** for User table.
* **TextField** for User ID (**read**-**only**), User Email, User Email, User Name, User Password, User Gender, User Age, User Role.
* **Text** for User ID, User Email, User Email, User Name, User Password, User Gender, User Age, User Role, Male, Female.
* **Radio Button** for Male, and Female button
* **ComboBox** for User Role.
* **Button** for Insert New User, Update User, Remove User, Insert, Cancel.

**Description**:

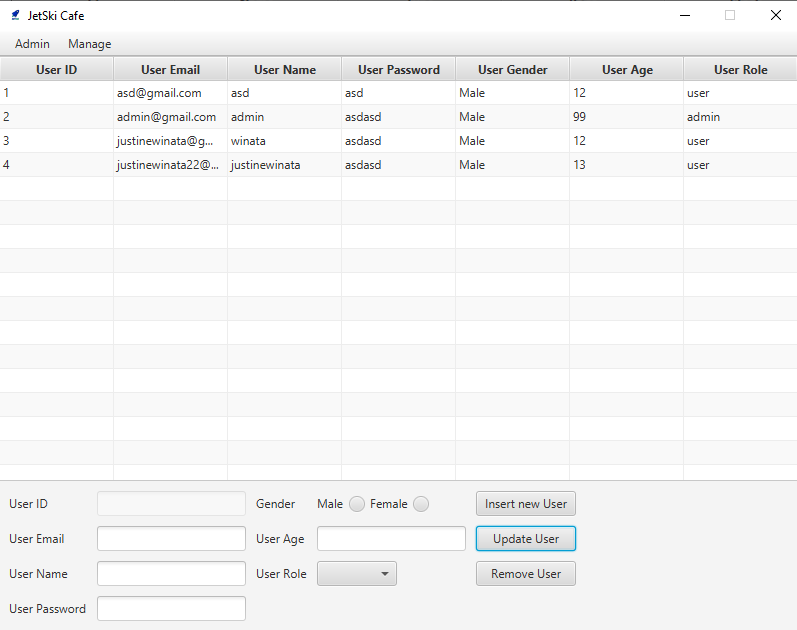
* This page only available for **admin** role
* **Create New User**
  + If admin click Create New User. Show the extra button (Insert, and Cancel Button).
  + Generate new ID for the User ID. **User** **ID is incremented by 1** from the last User ID. If there’s no any user, set the User ID to 1
* **Update Menu**
  + **Validate:**
    - If there’s an empty field, show an **Error Allert.**
    - If there’s not selected user, show an **Error Alert**.
    - If the **User Name** **length is less than 5 and more than 25**, show an **Error Alert.**
    - If the **Emails not ends with** **‘.com’**. show an **Error Alert.**
    - If the **Age length not between 12 – 99.** Show an **Error Alert.**
    - If the **Password same with username.** Show an **Error Alert.**
  + **Update the menu** using the provided data.
  + Finally, show a **Notification Alert**.



*Figure 37. User not picked Error*

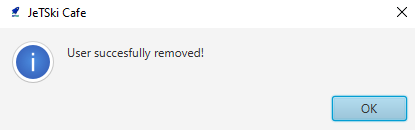


*Figure 38. User successfully updated*

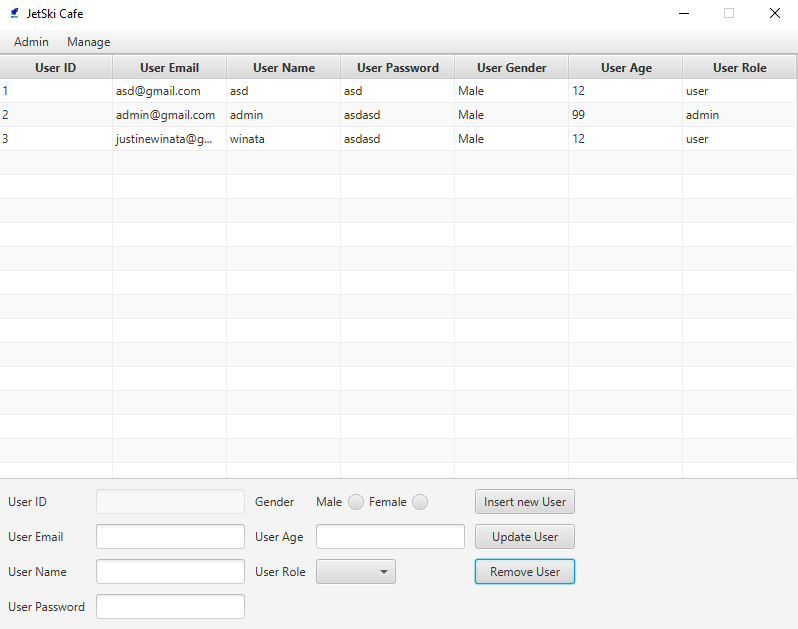


*Figure 39. User Updated*

* **Remove User**
  + **Validate:**
    - If an **user is not selected**, show an **Error Message.**
  + **Remove the user** from the database.
  + Finally, show a **Notification Alert**.

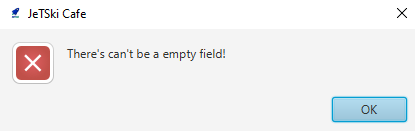


*Figure 40. Deleted User Notification*

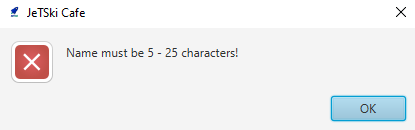


*Figure 41. User Deleted*

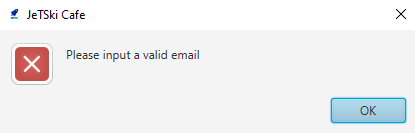
* **Insert**
  + **Validate:**
    - If there’s an empty field, show an **Error Allert.**
    - If the **User Name** **length is less than 5 and more than 25**, show an **Error Alert.**
    - If the **Emails not ends with** **‘.com’**. show an **Error Alert.**
    - If the **Age length not between 12 – 99.** Show an **Error Alert.**
    - If the **Password same with username.** Show an **Error Alert.**
  + **Insert the menu** into the database.
  + Don’t forget to **hide** the **insert** and **cancel** button.
  + Finally, show a **Notification Alert**.



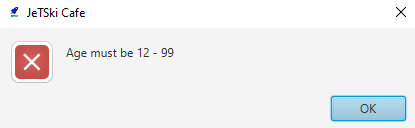
*Figure 42. Empty Field Error*



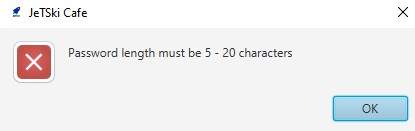
*Figure 43. Invalid Name Length*



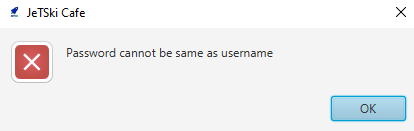
*Figure 44. Invalid Email*



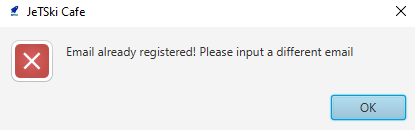
*Figure 45. Invalid Age Length*



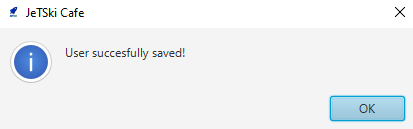
*Figure 46. Invalid Password Length*



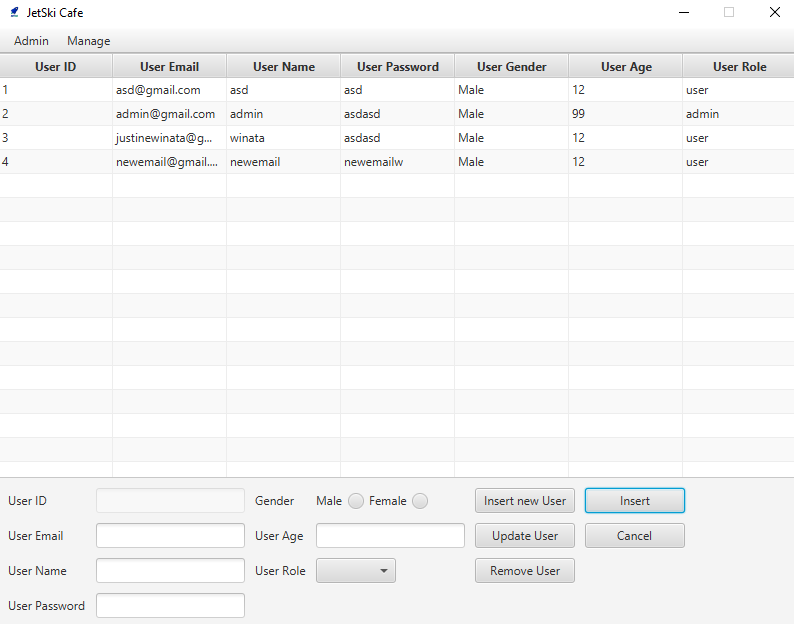
*Figure 47. Invalid password condition*



*Figure 48. Email already registered*



*Figure 49. Published User Notification*



*Figure 50. New User published*

* **Cancel**
* If user click cancel empty **all fields**, and hide **insert button** and **cancel button**

**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

**References**:

[**https://unsplash.com/photos/8IKf54pc3qk**](https://unsplash.com/photos/8IKf54pc3qk) **- Cafe Wallpaper**

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