



**Faculty of Engineering**

**Faculty of Engineering Helwan University**

**3rd Year Computer Department**

**Computer Package (Dr. Hadeer Ahmed)**

- **Project Name: Store Management System**
- **Group Number: 5**
- **Submission Date: 28 / 12 / 2022**
- **Project Administrators**

Name	Sec
<b>Anas Hesham Mohamed</b>	<b>1</b>
<b>Toka Abd El Gawad Abd Allah</b>	<b>1</b>
<b>Habiba Emad Eldeen Abd allah</b>	<b>1</b>
<b>Hussien Mostafa Mahmoud</b>	<b>1</b>
<b>Norhan Ahmed Saber</b>	<b>3</b>

## ❖ Table of Contents:

Project Name: Store Management System .....	1
Project Administrators .....	1
Table of Contents .....	2
introduction to the Problem .....	3
purpose of the project .....	3
System Architecture .....	5
System Features .....	12
Functional Features .....	12
Non Functional Features .....	43
Architectural Design .....	78

## ❖ Introduction to the problem:

we need a system that is having a generic functionality to manage a store, fair, hypermarket or any other commercial organization.

Most of the current management systems are more likely to be custom built for specific purposes and missing some important features.

## ❖ Purpose of the project:

the purpose of the project is to have a system that can be mostly generic to fit in the market without any missing features like the following:

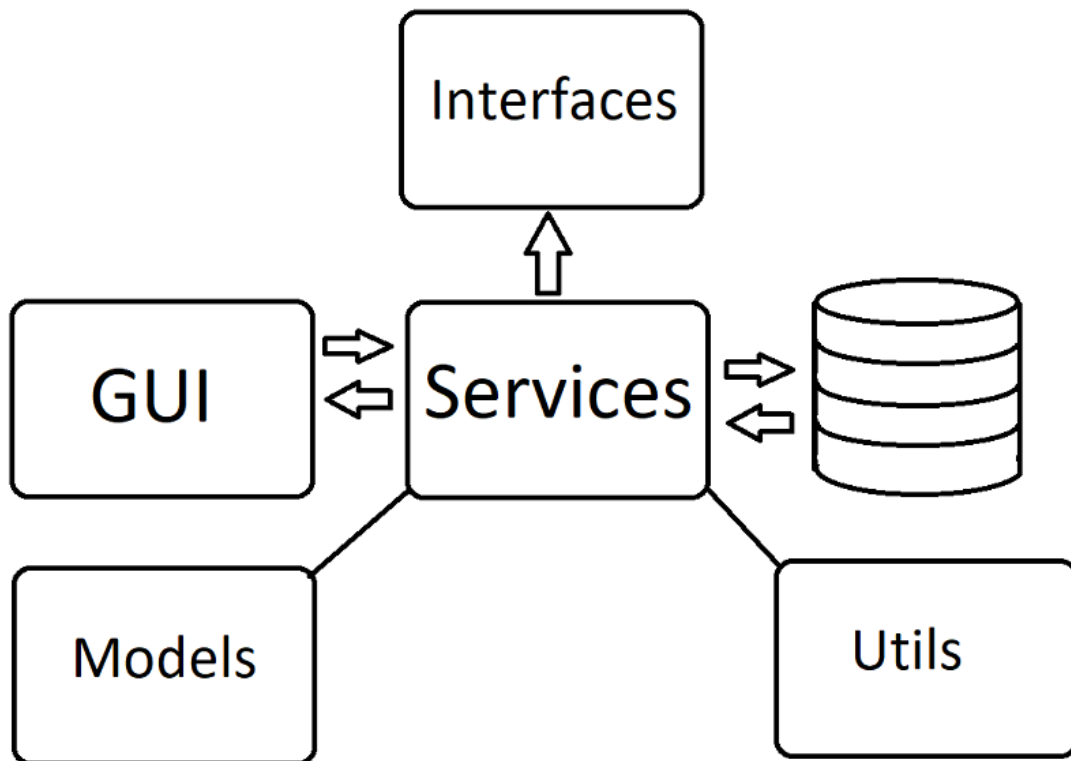
- **Creating Customers:** gathering some of their information and get them to **select** items to buy from the store inventory.
- **Importing items:** creating any type of item to be added to the store inventory
- **Uploading Pictures:** to any item while creating and updating it, to add visualization to the customers when they choose their items.
- **Creating Users:** adding users and giving them personal accounts, to contribute to **sales/management**.
- **Creating Roles:** adding a **Role** with different permissions to limit every user access to the system functionality.
- **Assigning Roles:** to users to set their permissions.
- **Creating Logs:** a detailed log is generated with every action done by any user.
- **Monitoring Logs:** to be able to see all the logs generated in the system before.

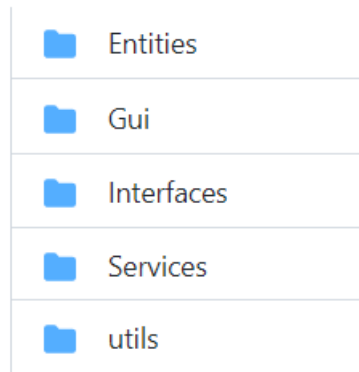
## ❖ System Architecture:

- **System Design:**

The solution is divided into 6 main abstract Components:

- Entities (Models)
- Interfaces
- Services
- GUI
- Utils
- The Database





Each one of these Projects runs on its own and transfers data through the services package to other parts of the solution so it illustrates **Abstraction** the **Separation of Concerns** concept.

- **Entities (Models):**

The Entities Project (Package) is where the data models live.

It Includes all the entities used throughout the application in every other component

The Entities are:

- Customer
- User
- Item
- Role
- Log

- **Interfaces:**

The interfaces Project (Packages) is where all the interfaces exist, the interfaces are the illustrator of every function in the system, it tells the Services what functions should be implemented and the capabilities of every Services

The Interfaces package is divided to the main Interface that all other interfaces inherit from and has all the generic function that all the services should implement.

The Interfaces are:

- IGenericInterface
- IUserInterface
- ICustomerInterface
- IItemInterface
- IRoleInterface
- ILogInterface

The generic interface defines the 5 main functions (CRUD Operations) which are:

- Get All
- Get By ID
- Create
- Update
- Delete

The derived interfaces that implement this Interface can define their own specific functions.

- **Services:**

The services project (package) is where the actual logic lives.

All services implement its interface from the interfaces package and inherit from the generic services.

The generic service is the parent of all the services, it implements all the methods in the **IGenericInterface** (CRUD) so it can be used or overridden by the child services.

The services are the way the application communicates with the database to fetch, insert or update existing data. Through it the **GUI** components can send or get data from the database.

The services use the **Java Database Connection (JDBC)** in order to communicate with the database and execute queries the SELECT, INSERT, UPDATE or DELETE queries is being generated in the database using **Statements** or **Prepared Statements**.

In order to have a connection with the SQL server a connection manager is developed.

The system can't have more than one connection running at the same time, so the **Singleton** design pattern is implemented to prevent the system from having more than one instance of the connection manager class, instead the same instance is being passed by reference throughout the whole application.

The services are:

- User Services.
- Customer Services.
- Item Services.
- Role Services.
- Log Services.

- **GUI**

The GUI or the graphical user interface is the core of the application, it holds all the panels and the main frame that are shown to the user.

Each Panel has its own **Render Data** function that communicates with the database through the services in order to view the desired data on the screen.

Some panels also can insert or update the database, this is also done through the services.

The navigation between panels is done by the switching the panel being viewed on the main JFrame by a function called **Switch Panels** that exists in the main Frame, this function and all the other main Frame function is accessible in all panels thanks to **dependency injection**.

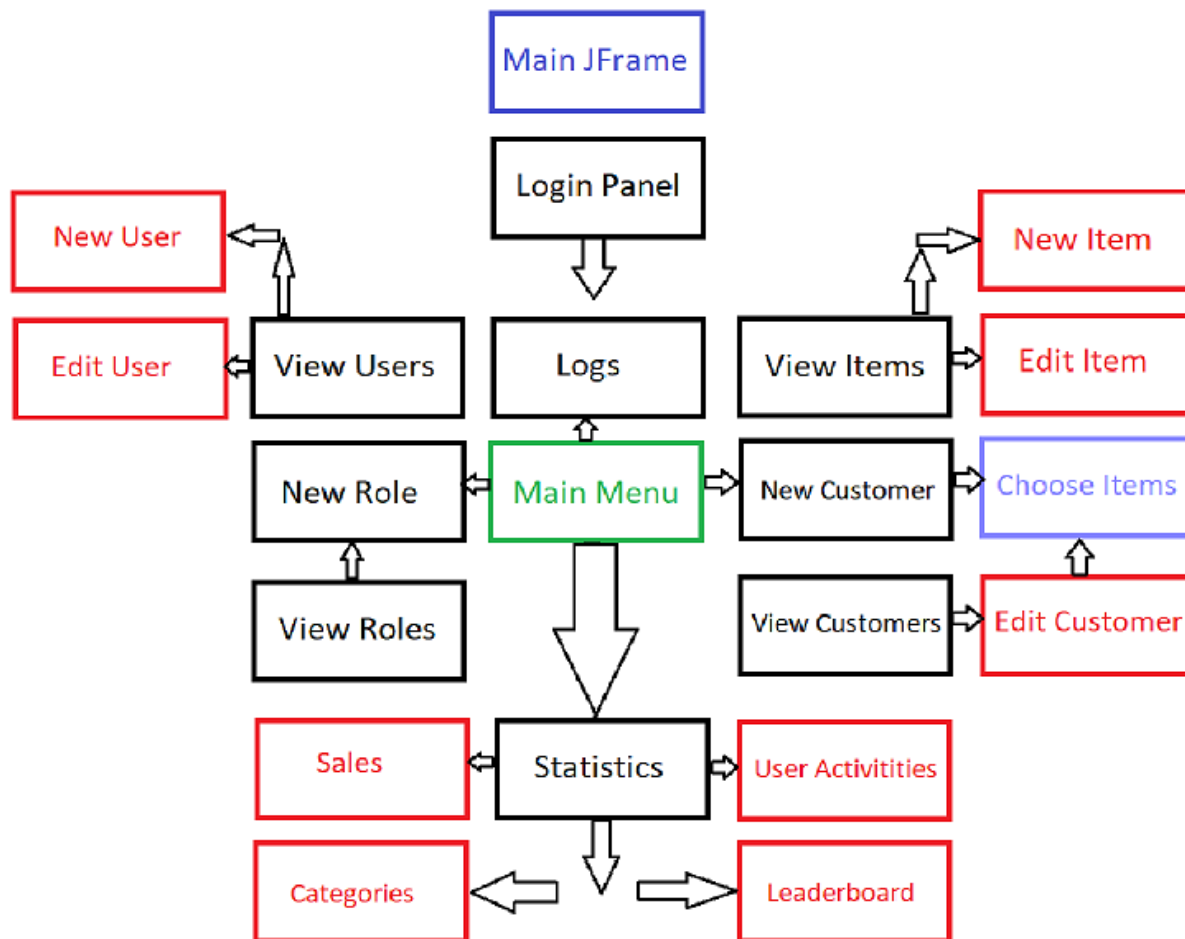
Each panel is being injected with the instance of the main Frame so it can access all the JFrame functions and vice versa, also the parent of

each panel is also injected so the panels can access each other's functions and pass data between each other.

The Panels are:

- jLoginPage
- jMainPage
- jNewCustomer
- jNewUser
- jNewItem
- jNewRole
- jViewCustomers
- jViewItems
- jViewUsers
- jViewLogs
- jViewRoles
- jEditCustomer
- jEditItem
- jEditUser
- jEditRole
- jStatistics
- jSales
- jCategories
- jUserActivities
- jLeaderboard





Some of the Panels is used for multiple purposes like:

- Choose Item: Used to assign items to customer, and to view/edit the customer purchases.
- Create Role: Used to create a new role, to edit an existing role.

These multi-purpose panels are done by injecting a different kind of parent panel to it using a generic Panel variable

Then the functionality of it is changed as a result of what type this parent is.

- **Utils:**

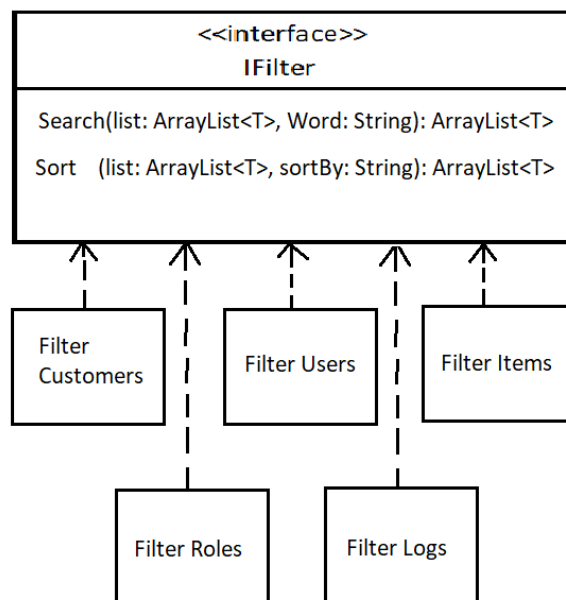
The Util package is where all the additional logical data that is used in multiple places in the application lives.

Some of the application component use the utils functions to execute some logic while rendering data to the panel.

The Util Package consists of:

- Customers Filter
- Items Filter
- Users Filter
- Logs Filter
- Role Filter
- DB Type: Enum

All the filters implement the **IFilter** Interface which defines the Search, Sort and Select Methods.



## ❖ System Features:

### -Functional Features:

- **Create Customer:**

The system can create new customer to be added in the database. The system asks for the following information about the item being added:

- Name
- Phone Number
- National id
- Gender
- E-mail
- Assigned Items
- Age

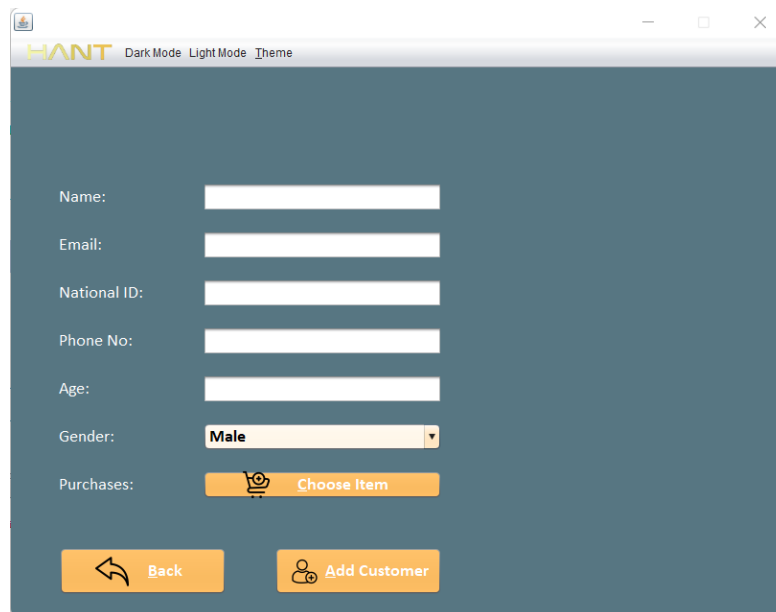


Figure: Screenshot from the Implemented Graphical User Interface showing the add customer panel

```
public class Customer {  
    public UUID id;  
    public String name;  
    public String nationalId;  
    public String phoneNumber;  
    public Date purchaseDate;  
    public String email;  
    public String gender;  
    public int age;  
    public ArrayList<UUID> itemIds;  
    public ArrayList<Item> items;  
}
```

Then a random UUID is being generated to be assigned for the id property, and the Current Date is assigned to purchase date property.

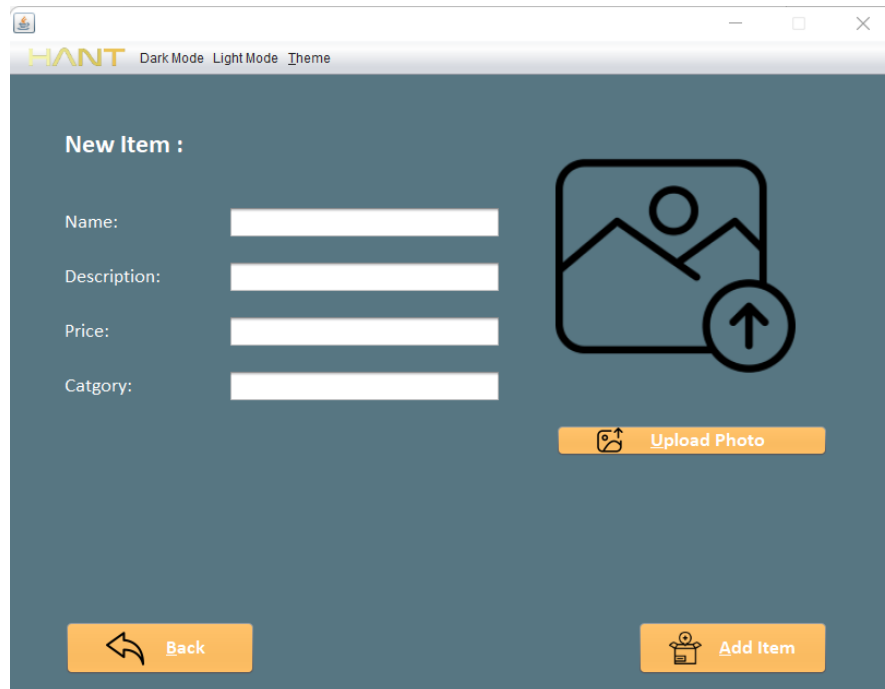
The selected items are assigned to the customer through the **assign items** function.

- **Create Item:**

The system can create new customers to be added in the database.

The system asks for the following information about the customer being added:

- Name
- Description
- Price
- Category



The screenshot shows a web browser window with the title 'HANT'. The browser's address bar shows 'Dark Mode Light Mode Theme'. The main content area is titled 'New Item :'. It contains four input fields labeled 'Name:', 'Description:', 'Price:', and 'Category:'. To the right of these fields is a large icon representing a photo upload (a square with a mountain and a circle with an upward arrow). Below this icon is a button labeled 'Upload Photo'. At the bottom of the form are two buttons: 'Back' (with a left arrow icon) and 'Add Item' (with a shopping cart icon).

```
public class Item {  
    public UUID id;  
    public String name;  
    public String description;  
    public int price;  
    public String category;  
    public Date createdAt;  
    public Date updatedAt;  
}
```

Then a random UUID is being generated to be assigned for the id property, and the Current Date is assigned to created at property.

- **Create Role:**

The system can create new role to be added in the database.

The system asks for the following information about the role being added :


- Name

And asks for the following permissions to be set:

- Can Create Customer
- Can View Customers
- Can Update Customer
- Can Delete Customer
- Can Create Item
- Can View Items
- Can Update Item
- Can Delete Item
- Can Create User
- Can View Users
- Can Update User
- Can Delete User

- Can Create Role
- Can Update Role
- Can View Roles
- Can View Logs
- Can Delete Log

Each one of the above properties determines whether the user having this role can do a certain action or not.


Dark Mode Light Mode Theme

Role Name :

☐ Create Customer
☐ Create Item
☐ Create User

☐ View Customers
☐ View Items
☐ View User


☐ Update customer
☐ Update Item
☐ Update User


☐ Delete Customer
☐ Delete Item
☐ Delete User

☐ Create Role
☐ View Logs

☐ View Roles
☐ Delete Log

☐ Update Role


Back


Create

```

public class Role {
    public UUID id;
    public String name;
    public boolean canCreateCustomer;
    public boolean canViewCustomers;
    public boolean canDeleteCustomer;
    public boolean canUpdateCustomer;
    public boolean canCreateItem;
    public boolean canViewItems;
    public boolean canDeleteItem;
    public boolean canUpdateItem;
    public boolean canViewUsers;
    public boolean canCreateUser;
    public boolean canDeleteUser;
    public boolean canViewLogs;
    public boolean canCreateLog;
    public boolean canDeleteLog;
    public boolean canUpdateRole;
    public boolean canViewRoles;
    public boolean canDeleteRole;
    public boolean canCreateRole;
}

```

- **Create User:**

The system can create new user to be added in the database. The system asks for the following information about the user being added:

- Name
- Username
- Password
- Role

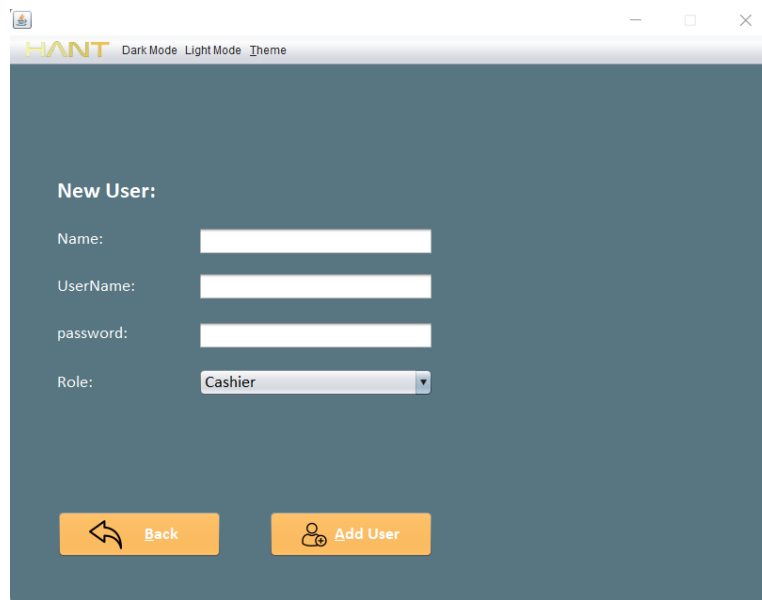


Figure: Screenshot from the Implemented Graphical User Interface showing the add customer panel

```
public class User {  
    public UUID id;  
    public String name;  
    public String userName;  
    public String password;  
    public Role role;  
}
```

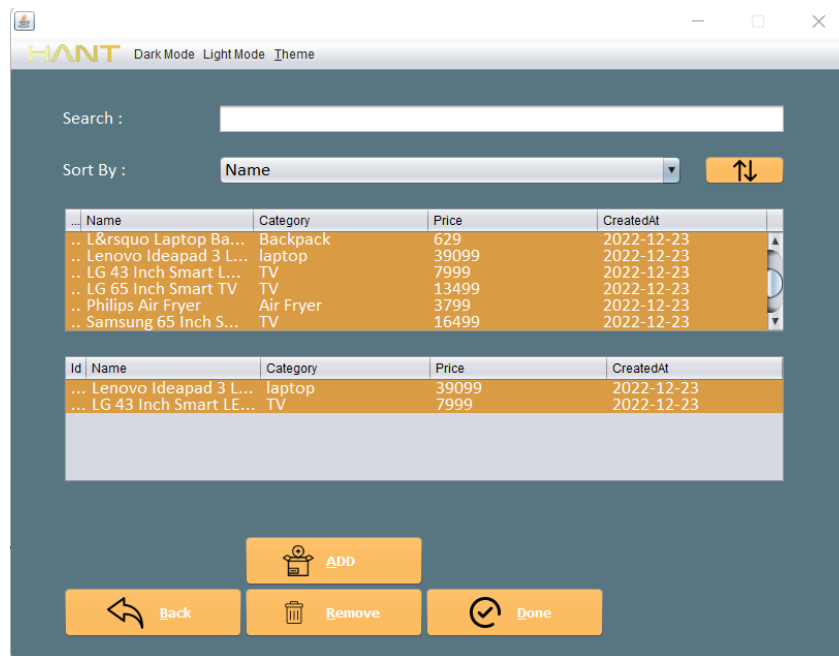
Then a random UUID is being generated to be assigned for the id property

The selected role is assigned to the user using its name.



- **Assigning Items to Customer:**

The system allows customers to choose items from the store inventory and assign it to the customer being added



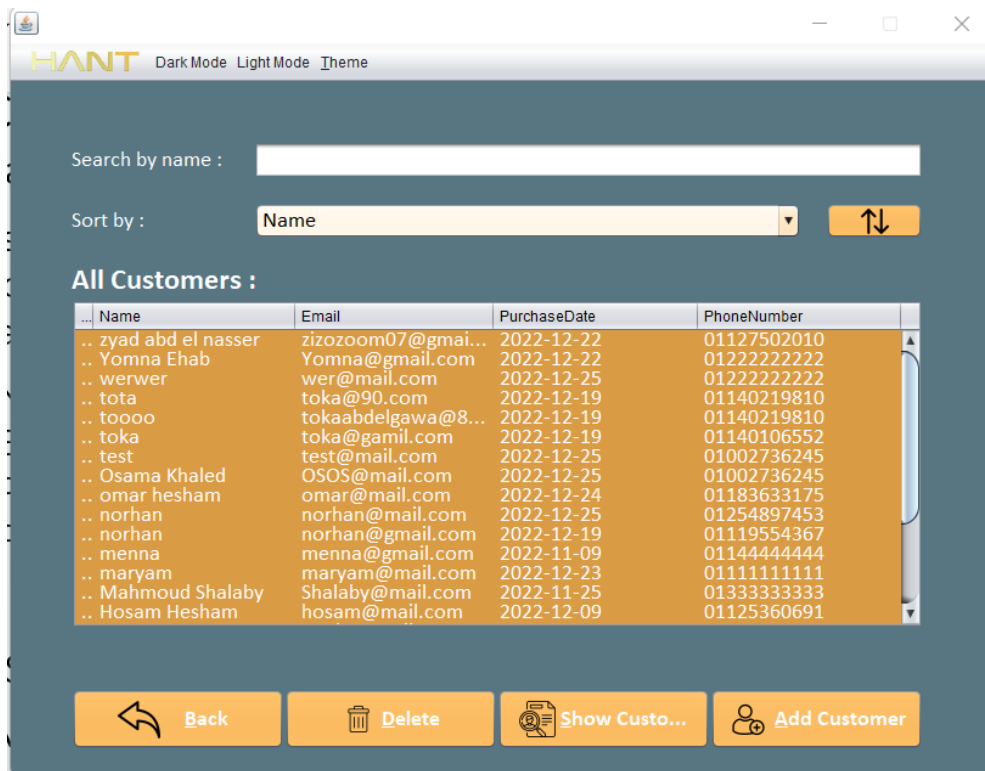
Then the selected items are assigned using its ids to the new customer through the joint table in database.

- **View Customers:**

- View all Customers:

The system allows users to view a list of all the customers stored in the database and show certain data:

- Name
- Email
- Phone Number
- Purchase Date



- View Specific Customer:

The system allows user to show all the data stored of a certain selected customer:

- Name
- Email
- National ID
- Phone No
- Age
- Gender
- Purchase Date

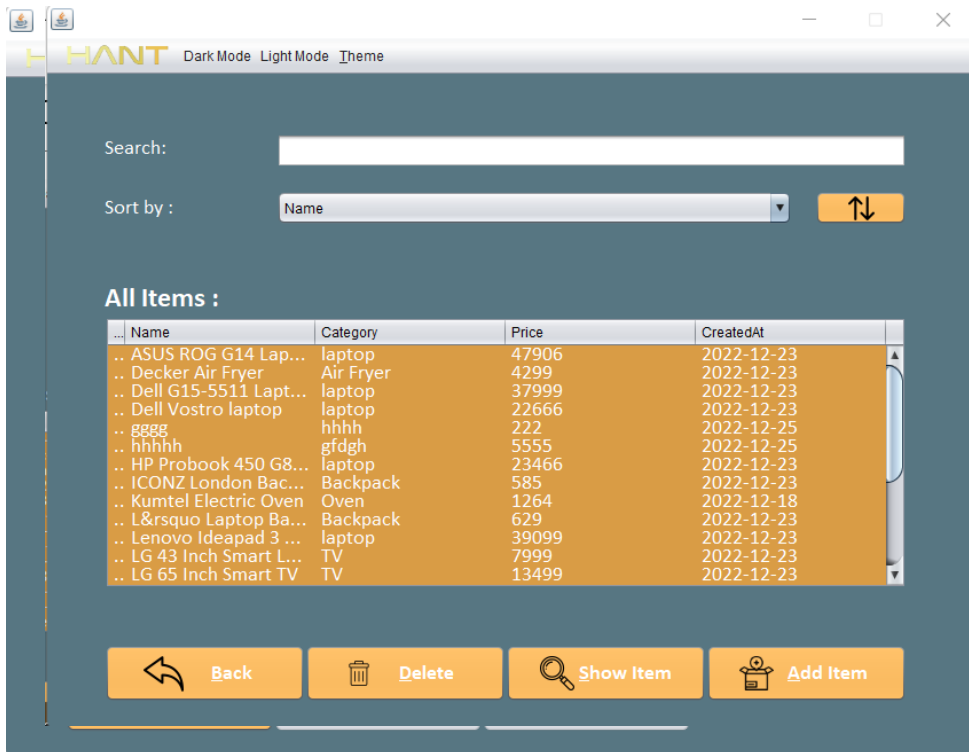
The screenshot shows a web application window titled 'HANT' with a dark theme. The window has a header bar with 'Dark Mode', 'Light Mode', and 'Theme' options. The main content area displays a user profile form with the following fields: Name (Yomna Ehab), Email (Yomna@gmail.com), National ID (12345975213214), Phone No (01222222222), Age (25), Gender (Female), and purchase date (2022-12-22 00:00:00). Below these fields is a 'Purchases' section with a 'Show purchases' button. At the bottom of the form are four buttons: 'Back', 'Edit', 'Cancel', and 'Update'.

The purchased items can be viewed or edited using the **View Assigned Items** Function.

- **View Assigned Items:**

the system allows users to view all the assigned items to a specific user using its id to search in the database then views the following information:

- Name
- Category
- Price
- Created At



- **View Items:**

- **View all Items:**

The system allows users to view a list of all the Items stored in the database and show certain data:

- Name
- Price
- Category
- Created At

- **View Specific Item:**

The system allows user to show all the data stored of a certain selected item:

- Name
- Description
- Category
- Created At
- Price
- Picture

The screenshot shows a web application window titled 'HANT' with a dark theme. The 'Item Data' form contains the following fields:

Field	Value
Name:	HP Probook 450 G8 Laptop
Description:	ntel i5, 15.6 Inch, 500GB SSD
Price:	23466
Created At:	2022-00-23
Catgory:	laptop

To the right of the form is a placeholder image of a laptop. Below the image is an 'Upload Photo' button. At the bottom of the form are four buttons: 'Back', 'Cancel', 'Update', and 'Edit'.

- **View Roles:**

The System allows the users to view all the roles saved in the database.

The screenshot shows a web application window titled 'HANT' with a dark theme. The 'View Roles' screen features a search bar at the top. Below it, the 'Roles Name:' section displays a list of roles:

- Accountant
- Admin
- Analyst
- Cashier
- Manager
- User

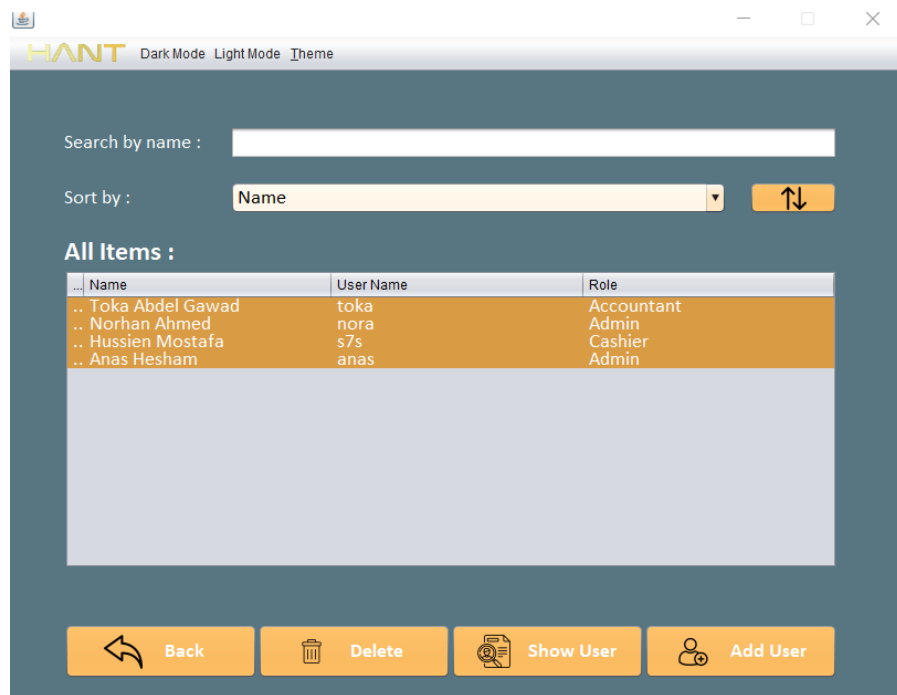
At the bottom of the screen are two buttons: 'Back' and 'Edit'.

- **View Users:**

- View all Users:

The system allows users to view a list of all the Users stored in the database and show certain data:

- Name
- Username
- Role



- View Specific User:

The system allows user to show all the data stored of a certain selected user:

- Name
- Username
- Role

**HANT** Dark Mode Light Mode Theme

**User:**

Name: Toka Abdel Gawad

UserName: toka

password:

Role: Accountant

Cancel Update

Back Edit

- **Update Customer:**

The system allows the user to update the customer's data saved in the database using the edit option.

**HANT** Dark Mode Light Mode Theme

Name: ziad abd el nasser

Email: zizozoom07@gmail.com

National ID: 23423423423423

Phone No: 01127502010

Age: 21

Gender: Male

purchase date: 2022-12-22 00:00:00

Purchases: Show purchases

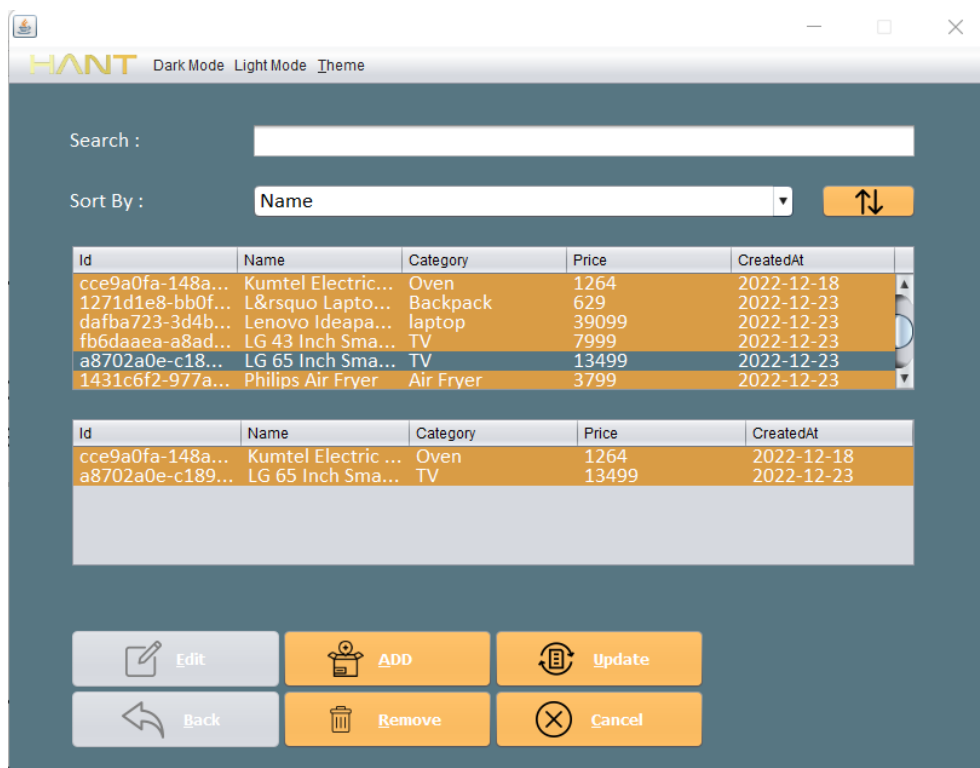
Back Edit Cancel Update

The new data will replace the old, stored data in the database

The assigned items can also be updated using the **Update Assigned Items** Function.

- **Update Assigned Items:**

The system allows users to update the assigned items to a specific customer where the user can add or remove an assigned item from the customer previous purchases.

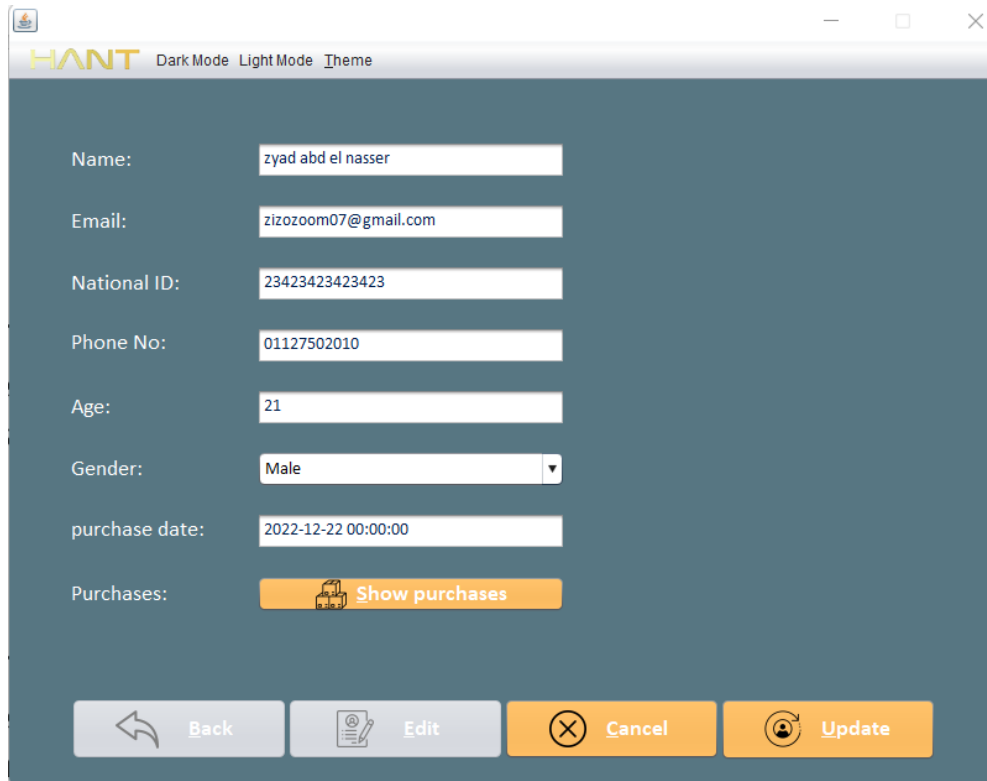


The new items after the deletion and the addition replace the old items assigned previously to this customer.




- **Update Item:**

The system allows the user to update the customer's data saved in the database using the edit option.



The screenshot displays a web application window titled "HANT" with a dark theme. The window contains a form for updating customer information. The form fields are as follows:

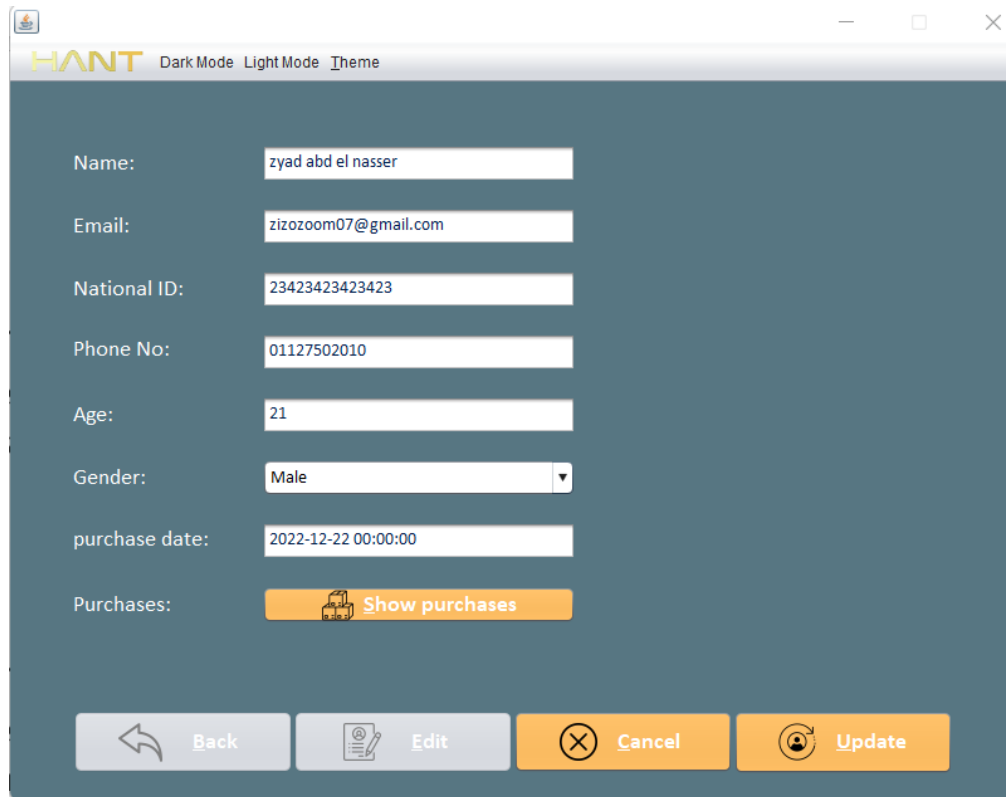
Field	Value
Name:	zyad abd el nasser
Email:	zizozoom07@gmail.com
National ID:	23423423423423
Phone No:	01127502010
Age:	21
Gender:	Male
purchase date:	2022-12-22 00:00:00
Purchases:	 <a href="#">Show purchases</a>

At the bottom of the form, there are four buttons: "Back" (with a left arrow icon), "Edit" (with a document and pencil icon), "Cancel" (with a red X icon), and "Update" (with a circular refresh icon).


The new data will replace the old, stored data in the database.

- **Update User:**

The system allows the user to update the user's data saved in the database using the edit option.



The screenshot shows a web application window titled "HANT" with a dark theme. The window has a header bar with "Dark Mode", "Light Mode", and "Theme" options. The main content area contains a form for updating user information. The form fields are as follows:

Field Label	Value
Name:	zyad abd el nasser
Email:	zizozoom07@gmail.com
National ID:	23423423423423
Phone No:	01127502010
Age:	21
Gender:	Male
purchase date:	2022-12-22 00:00:00
Purchases:	 <a href="#">Show purchases</a>

At the bottom of the form, there are four buttons: "Back" (with a left arrow icon), "Edit" (with a document and pencil icon), "Cancel" (with a red X icon), and "Update" (with a circular arrow icon).

The new data will replace the old, stored data in the database.

- **Update Role:**

The system allows the user to update the role's permissions saved in the database using the edit option.

Role Name :

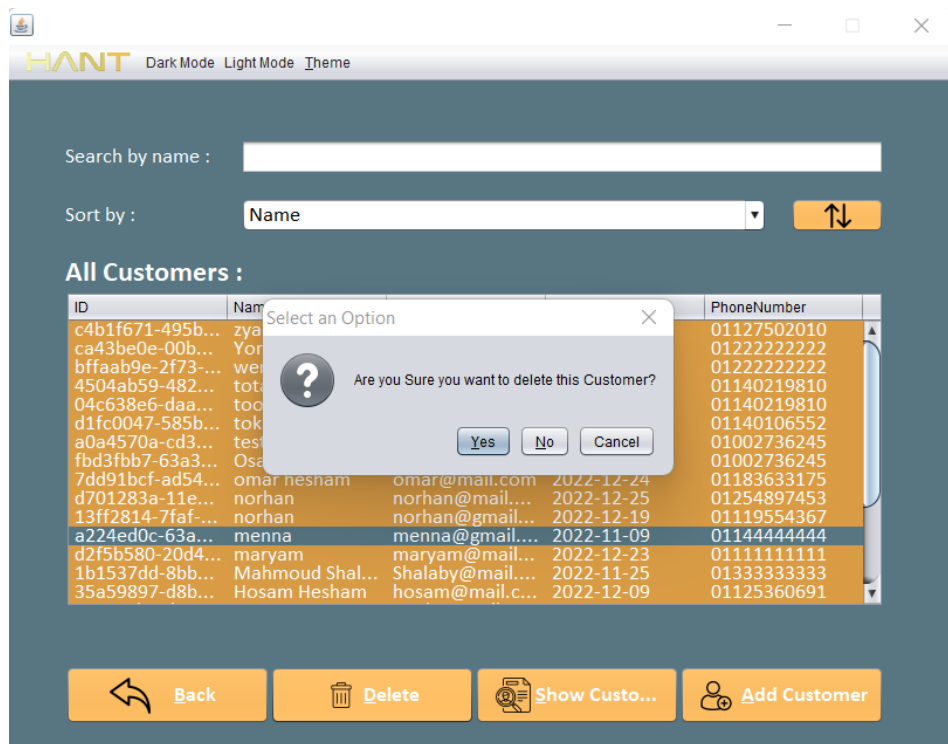
<input checked="" type="checkbox"/> Create Customer	<input type="checkbox"/> Create Item	<input type="checkbox"/> Create User
<input checked="" type="checkbox"/> View Customers	<input checked="" type="checkbox"/> View Items	<input type="checkbox"/> View User
<input checked="" type="checkbox"/> Update customer	<input type="checkbox"/> Update Item	<input type="checkbox"/> Update User
<input type="checkbox"/> Delete Customer	<input type="checkbox"/> Delete Item	<input type="checkbox"/> Delete User
<input type="checkbox"/> Create Role	<input checked="" type="checkbox"/> View Logs	
<input type="checkbox"/> View Roles	<input type="checkbox"/> Delete Log	
<input type="checkbox"/> Update Role		

The new permissions will replace the old, stored permissions in the database.

- **Delete Customer:**

The system allows the user to delete a customer that is no longer needed to be existed in the database.

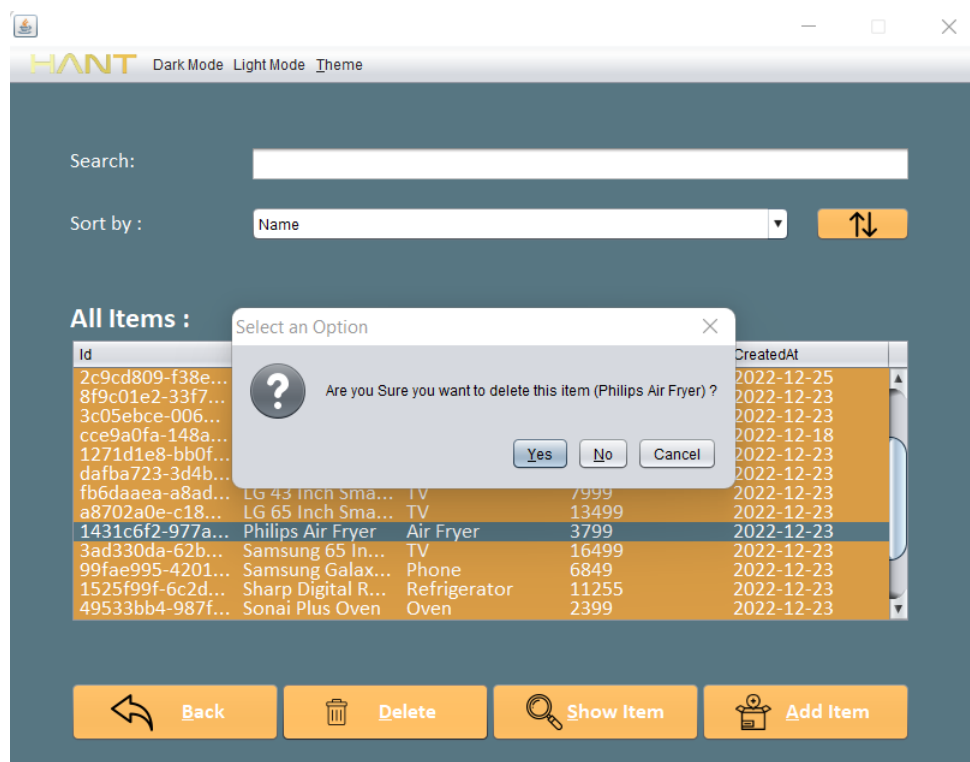
The delete operation only requires the customer id to be executed.



- **Delete Item:**

The system allows the user to delete an item that is no longer needed to be existed in the database.

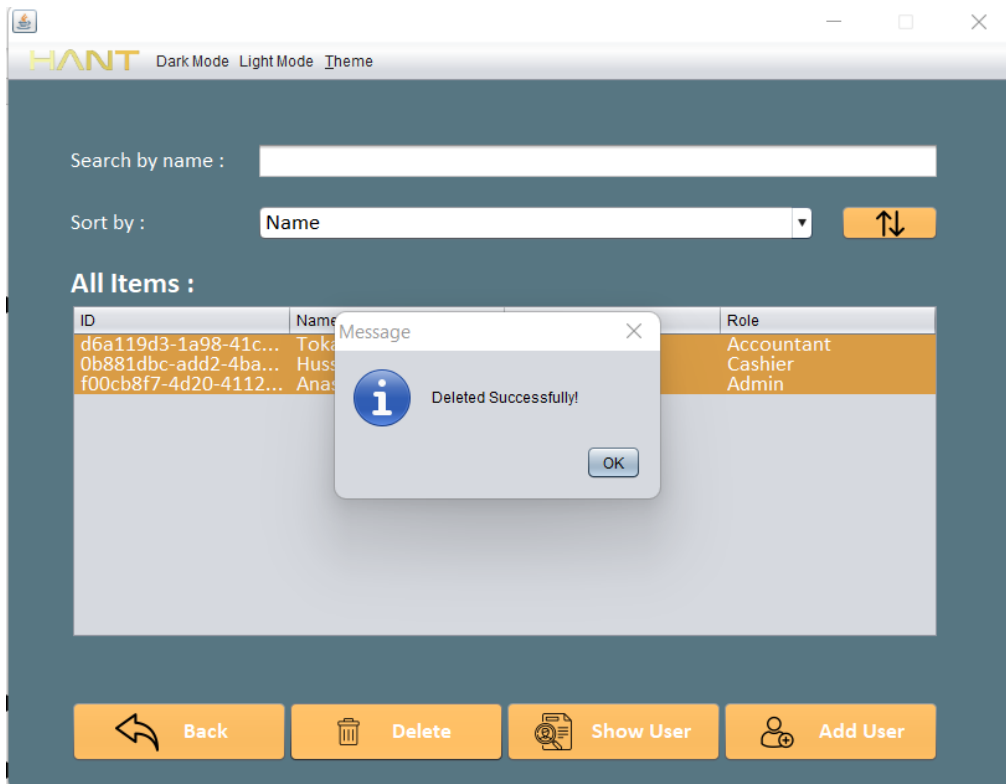
The delete operation only requires the item id to be executed.



- **Delete User:**

The system allows the user to delete a user that is no longer needed to be existed in the database.

The delete operation only requires the user id to be executed.



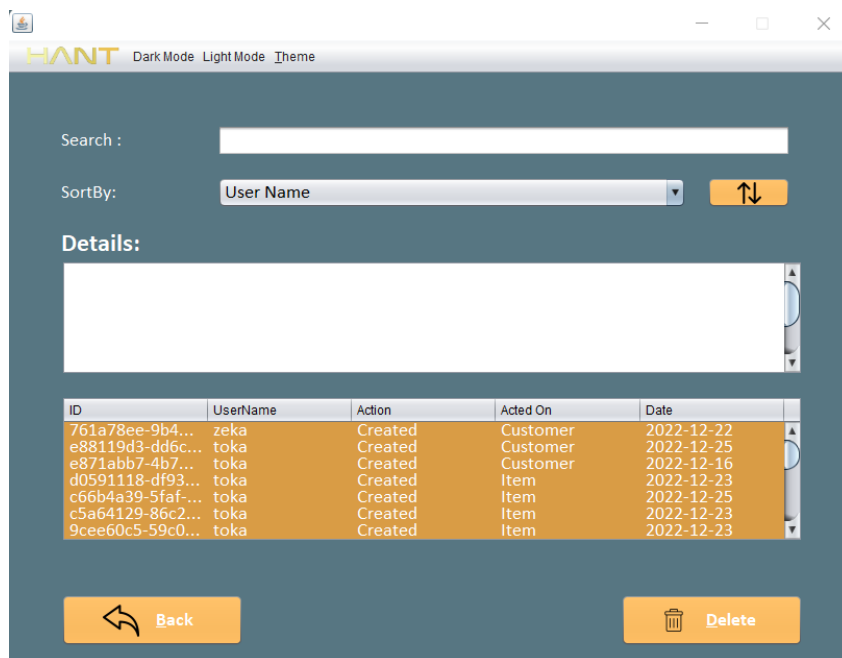
- **View Logs:**

One of the main core features in the system is the ability to store all the actions details in the form of a brief description (**log**).

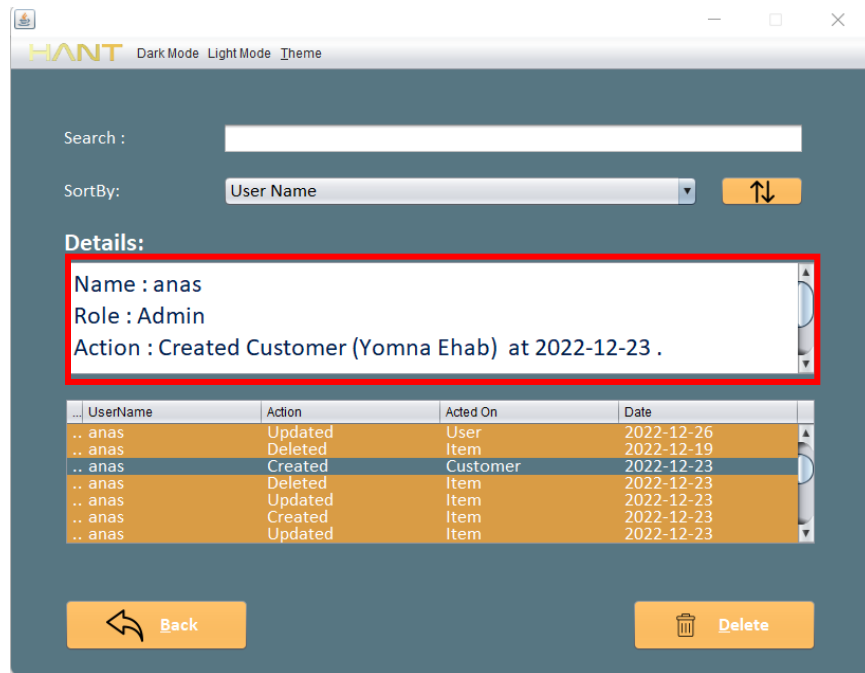
This feature gives you the ability to monitor the history of the system to check what happened during a certain period.

The system allows the users to view all the logs in a form of a list with the following information:

- Username: the username of the user did the action
- Action name: the operation name like:
  - Created
  - Updated
  - Delete
- Acted On: The type of the entity like:
  - Customer
  - Item
  - User
  - Role
- Acted On Name: the name of the entity affected by the action.
- Date: the date when the action happened.



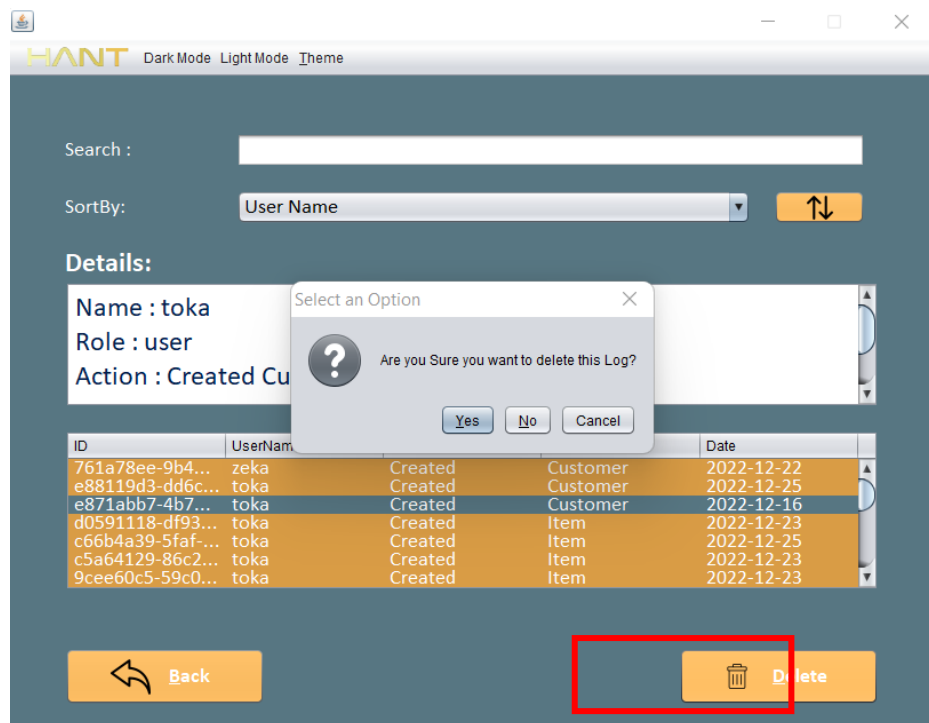
Then a specific Log can be viewed using the table selection to show more details about it.



- **Delete Log:**

The system allows the user to delete a customer that is no longer needed to be existed in the database.

The delete operation only requires the customer id to be executed.



- **Statistics:**

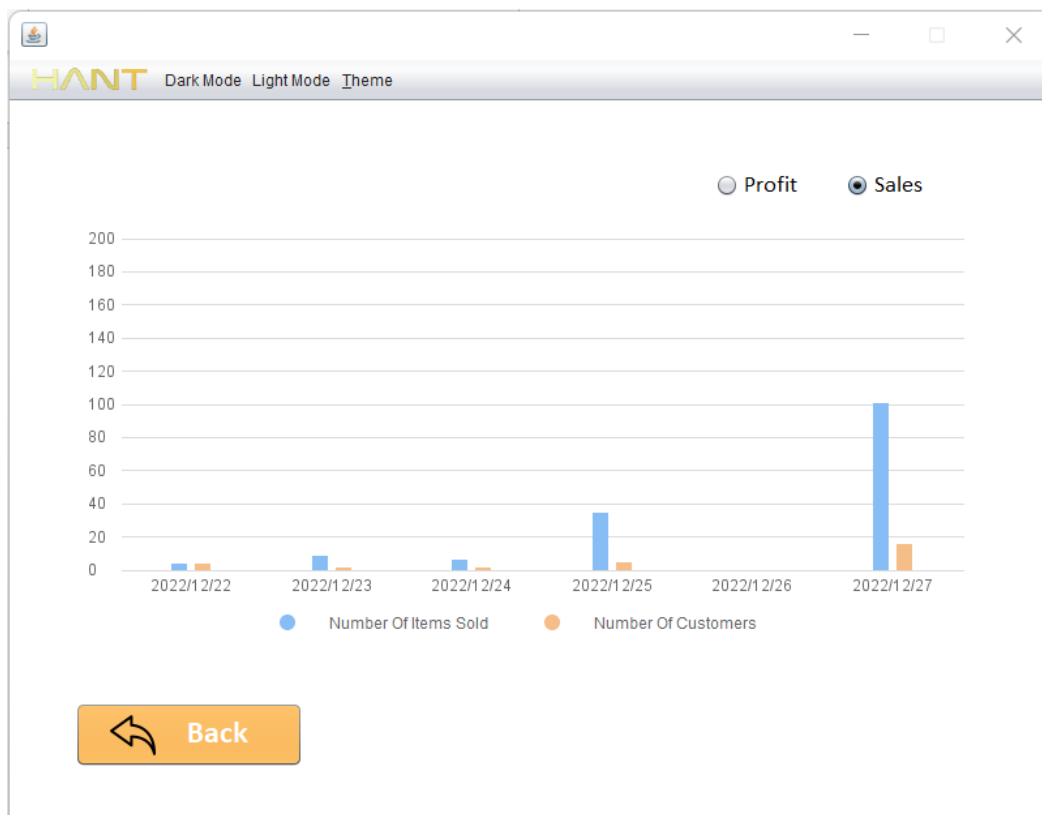
The system allows users to monitor statistics to view the work done during a certain period, the statistics is done through the database to get every action done by any user ever existed in the system.

The statistics is divided into 4 main categories:

- **Sales:**

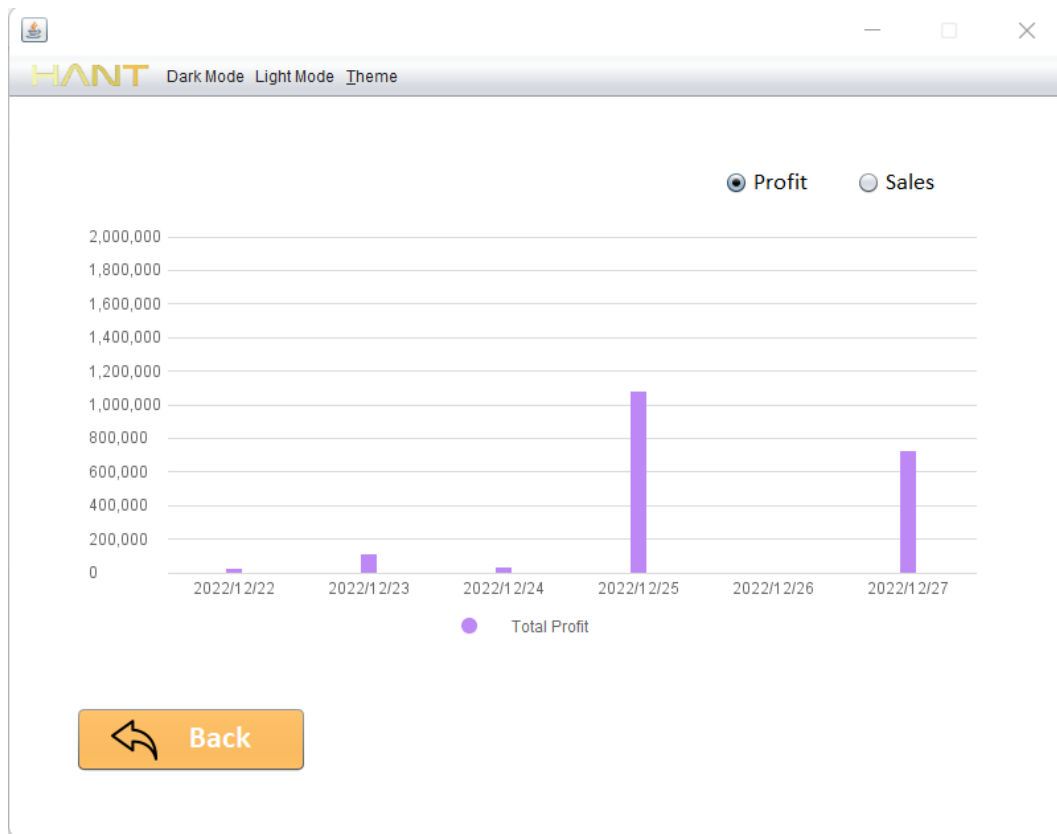
Shows the Sales Statistics comparing three types of sales throughout the day:

- Number Of Items Sold
    - Number Of Deals Done





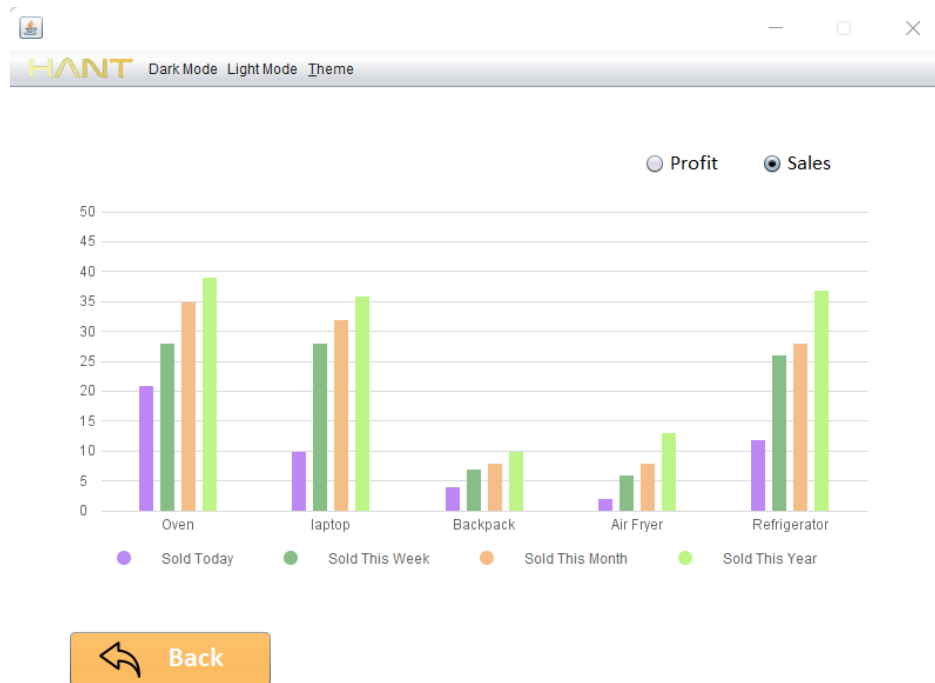
- Total Profit



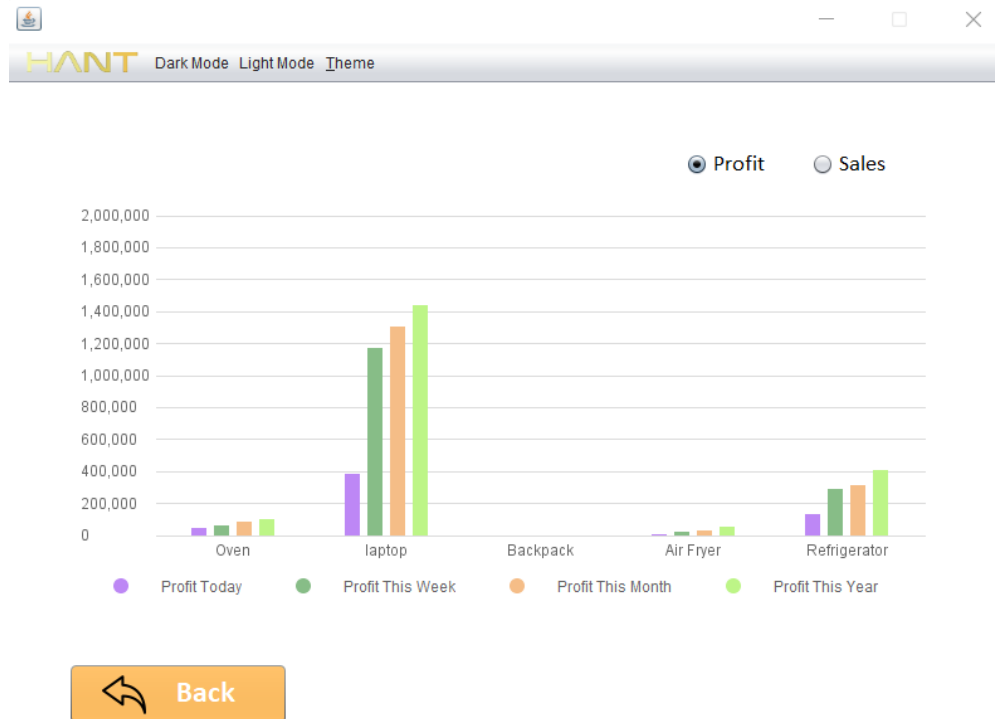
- Categories:

Shows the Categories Statistics comparing the 5 most sold category:

- Today
- This week
- This Month
- This Year



It also shows the profit for these categories during the same periods.



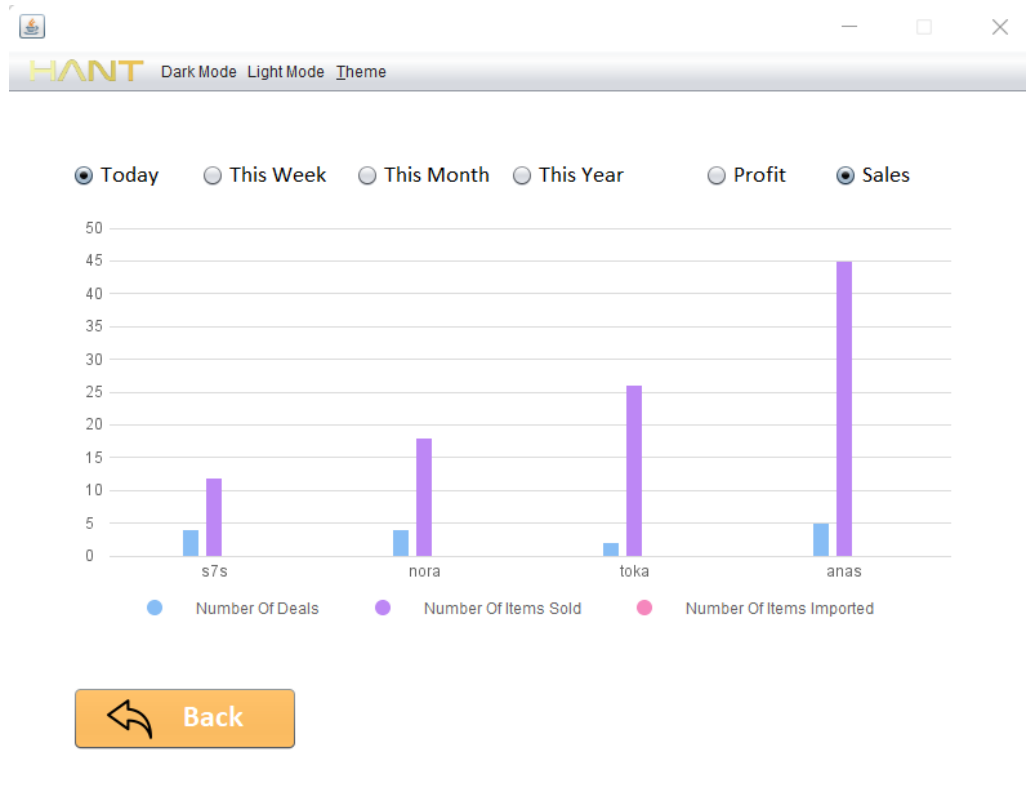
- User Activities:

Shows the Users Statistics comparing the 4 most active users:

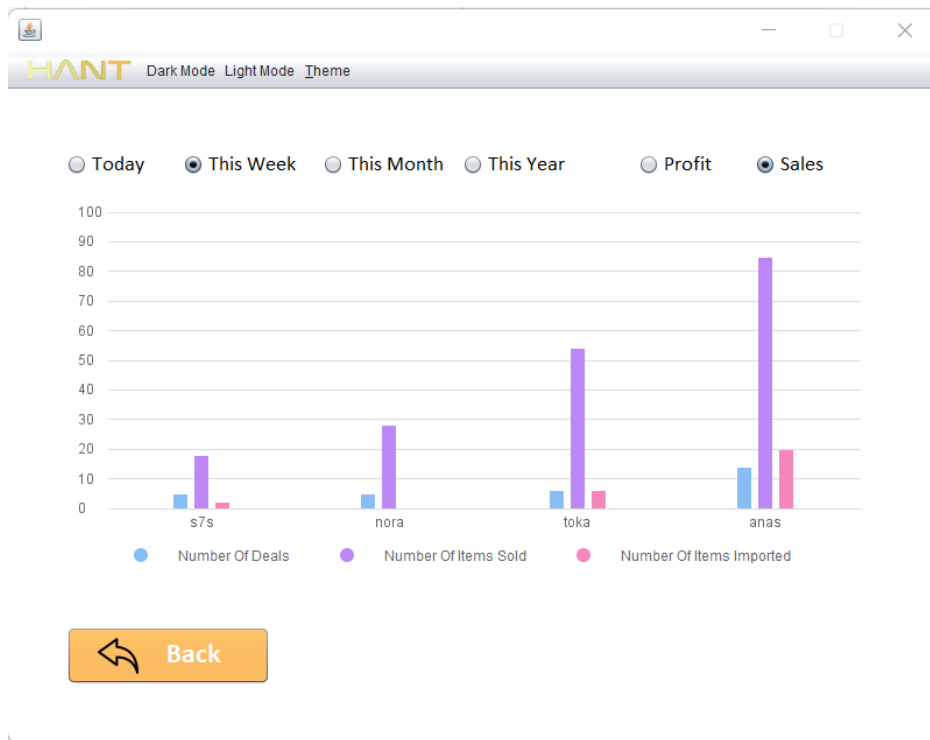
- Number Of Deals
- Number Of Sold Items
- Number Of Imported Items

These statistics is shown along different periods:

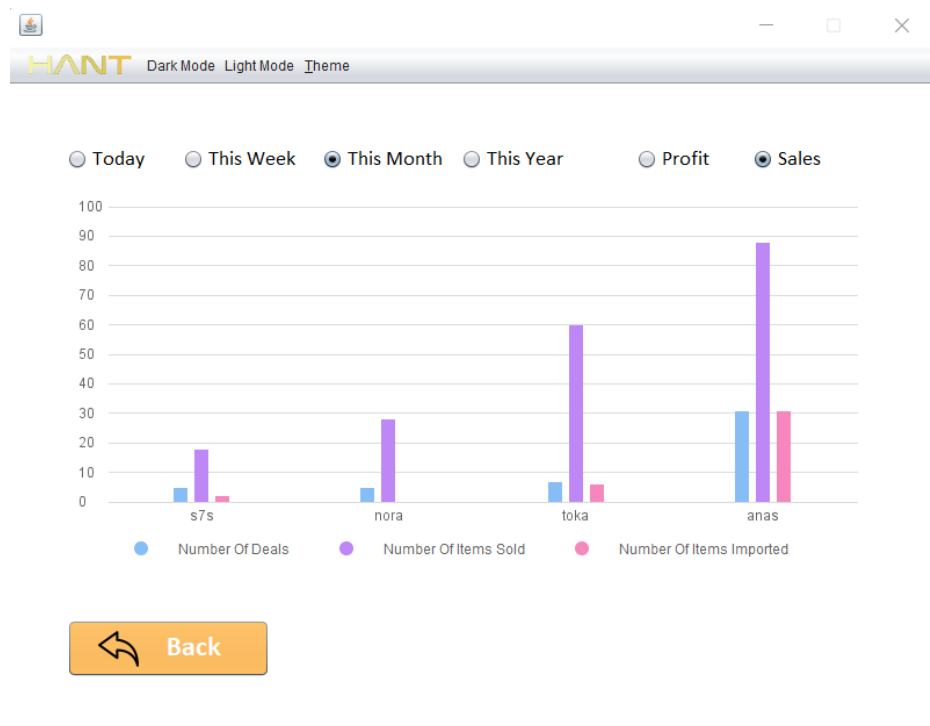
- Today



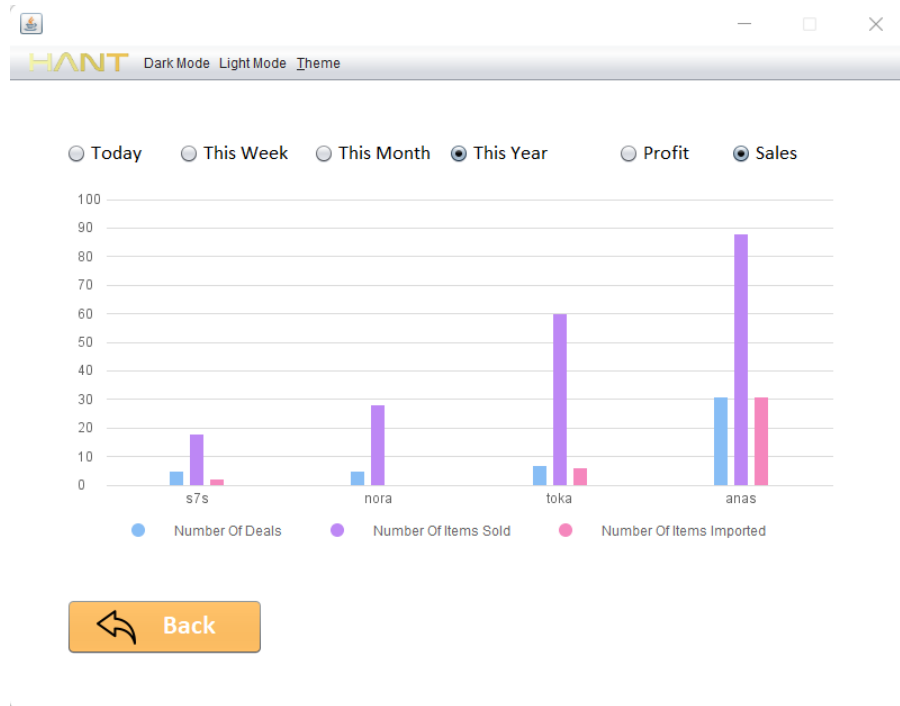
- This week



- This month

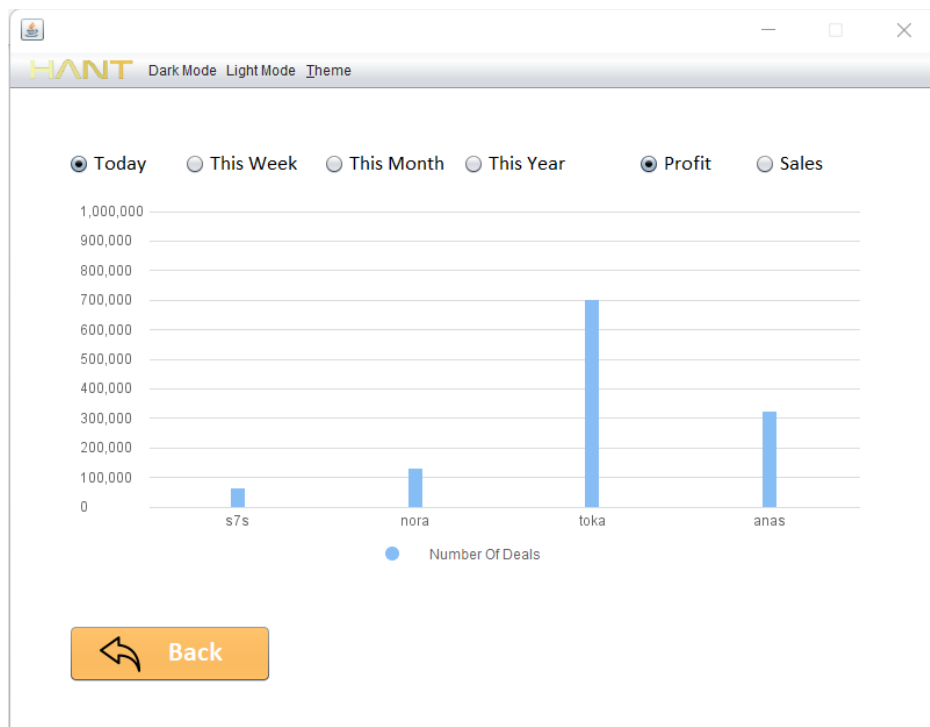


- This year

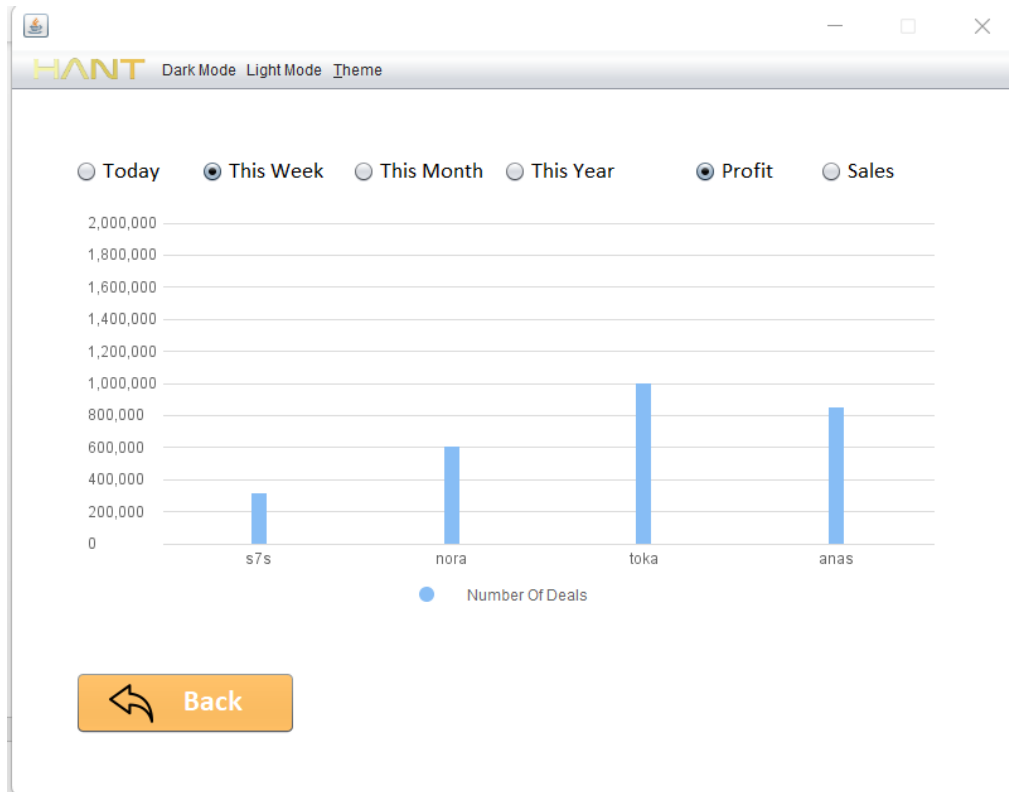


The profit done by every one of these users is also being shown throughout the same periods:

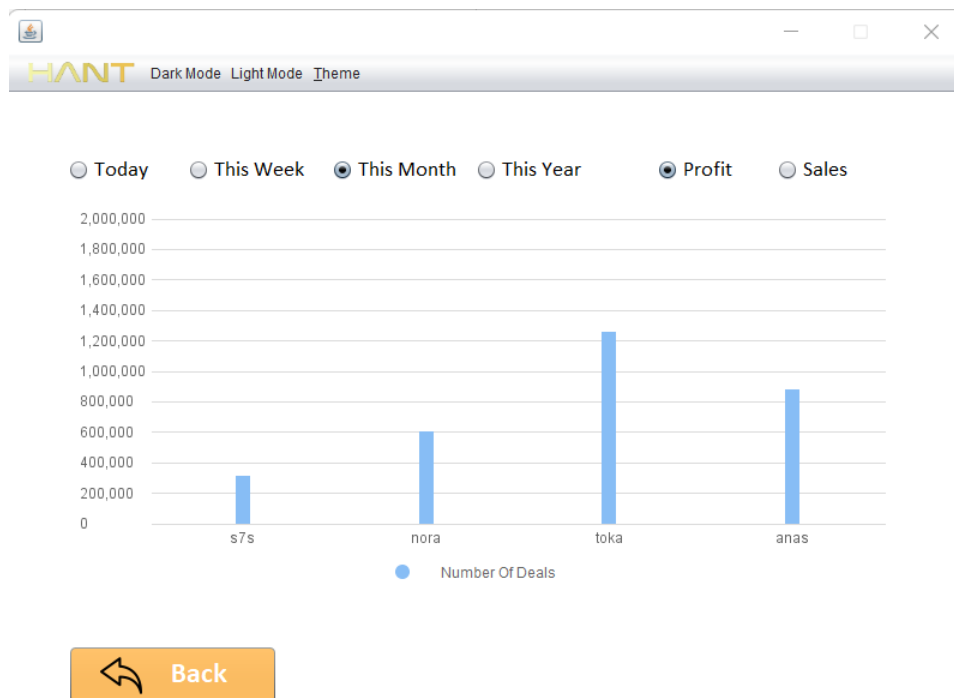
- Today



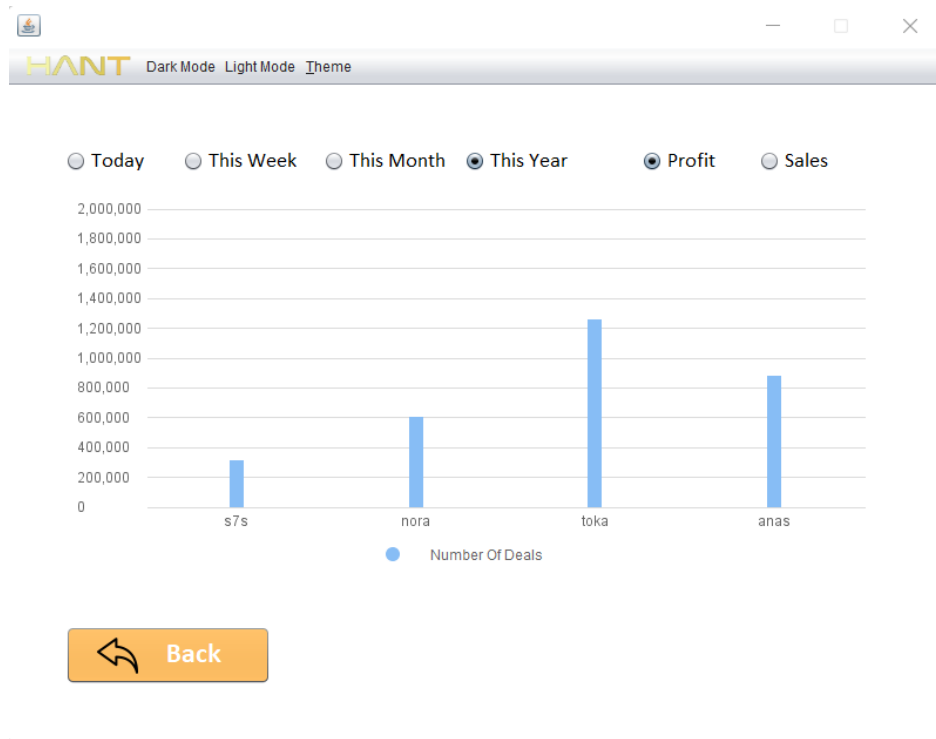
- This week



- This month



- This year



- Leaderboard:

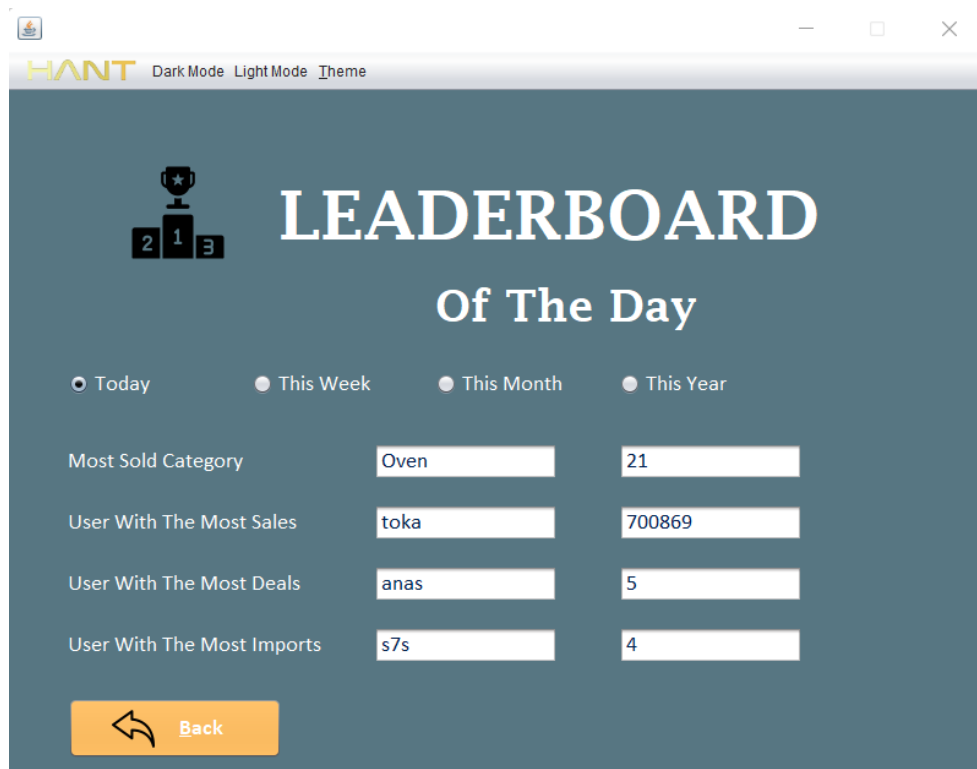
The system allows the users to view the leaderboard of all statistics.

The leaderboard is a panel that shows the leading user/category of the following statistics:

- Most Sold Category
- The User with Most Deals Done
- The User with Most Profit
- The User with Most Imported Items

The Leaderboard calculations is done to 4 periods:

- Today



- This week



- This month





- This year



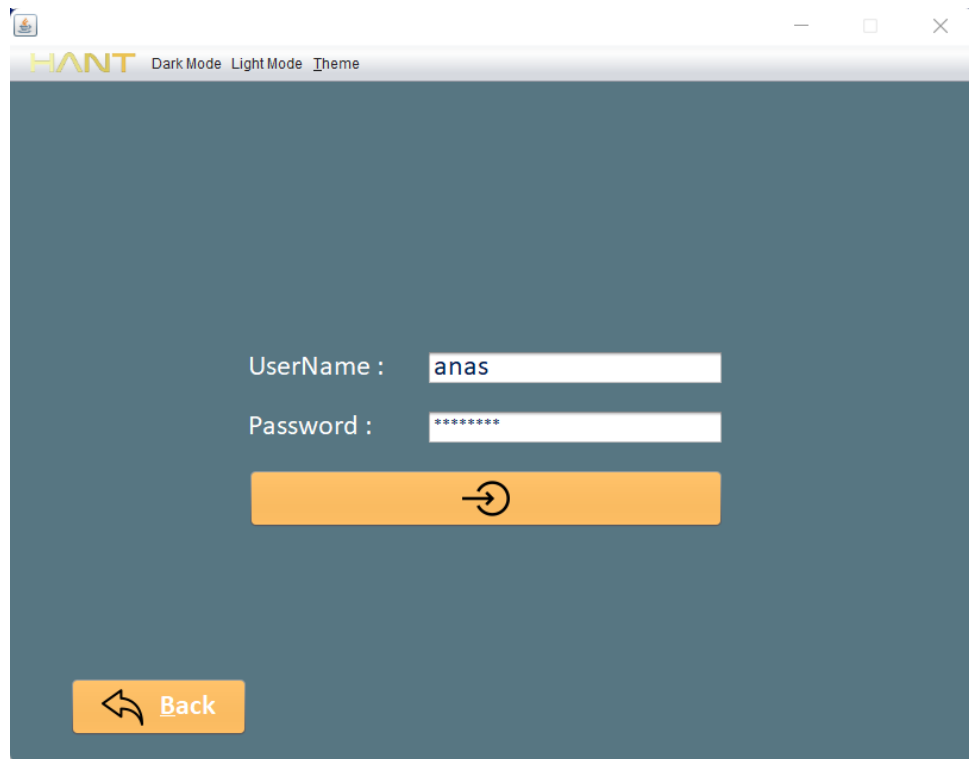
## ❖ Non-Functional Features:

- Authentication & Authorization:

- Authentication:

The authentication is done through the account services, the system authenticates the user logged after verifying the identity using the username and the password.

After the user is logged his entity is saved in the main Frame so it's data like role, username is fetched when needed



- Authorization:

The system allows the users to create roles with specific permissions, so they have access to some of the system functionality (only the assigned permissions to that role) and prevent them to access other functionality

Using the role services, the system manages the authorization so that for each function the system checks the logged in user role

to check if the specific permission is given to this role, thus it permits his access or deny it.

HANT Dark Mode Light Mode Theme

Role Name :

<input checked="" type="checkbox"/> Create Customer	<input checked="" type="checkbox"/> Create Item	<input type="checkbox"/> Create User
<input checked="" type="checkbox"/> View Customers	<input checked="" type="checkbox"/> View Items	<input type="checkbox"/> View User
<input type="checkbox"/> Update customer	<input type="checkbox"/> Update Item	<input type="checkbox"/> Update User
<input type="checkbox"/> Delete Customer	<input type="checkbox"/> Delete Item	<input type="checkbox"/> Delete User
<input type="checkbox"/> Create Role	<input checked="" type="checkbox"/> View Logs	
<input type="checkbox"/> View Roles	<input type="checkbox"/> Delete Log	
<input type="checkbox"/> Update Role		

Message

Permissions are assigned successfully!

HANT Dark Mode Light Mode Theme

**New User:**

Name:

UserName:

password:

Role: 

Analyst

Accountant

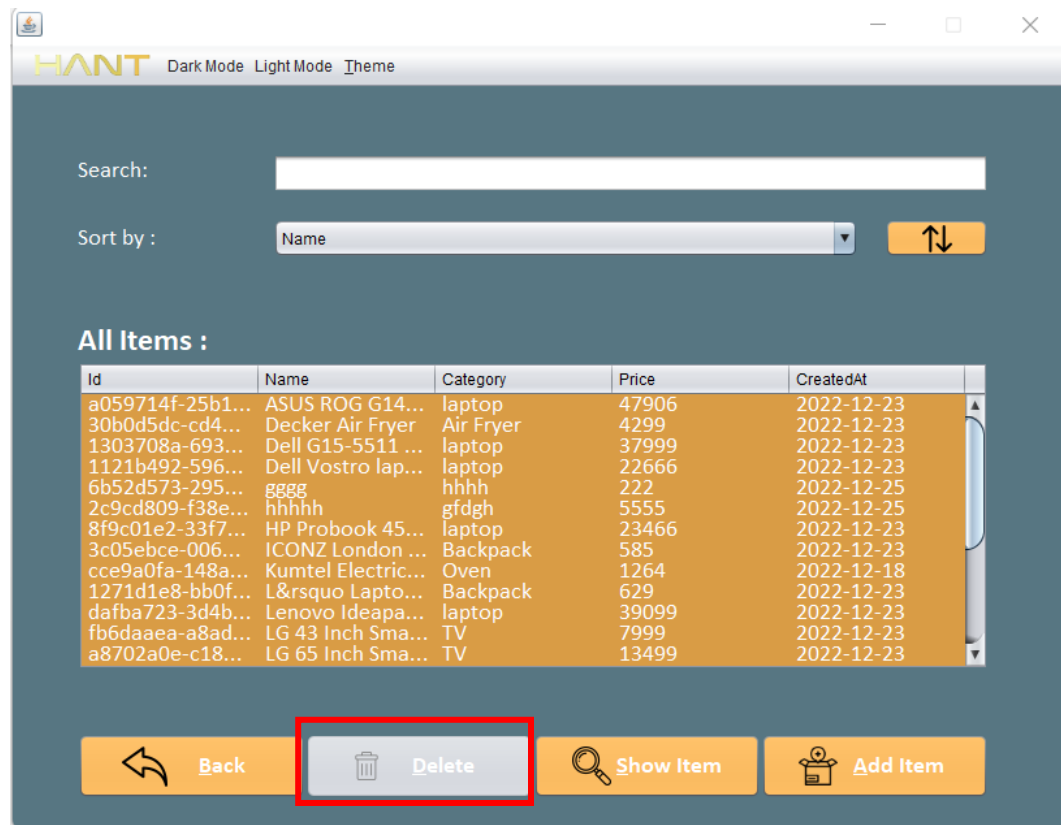
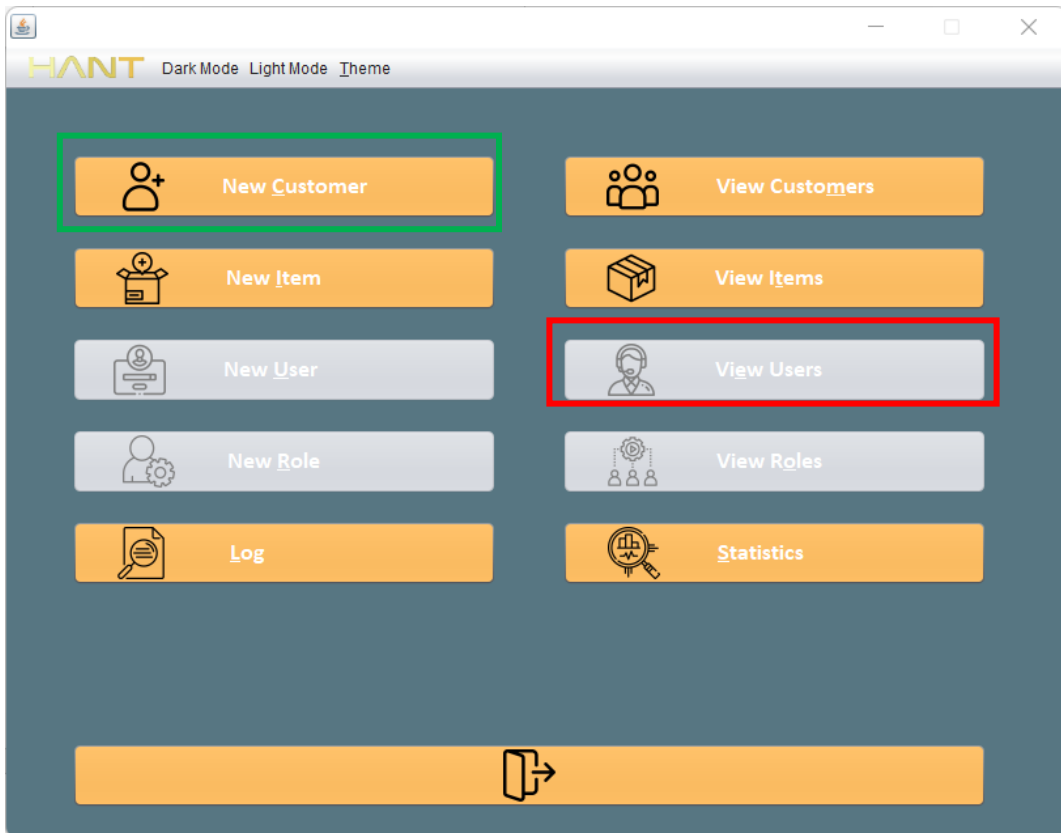
Admin

Analyst

Cashier

Manager

User



- **Filtering Customers:**

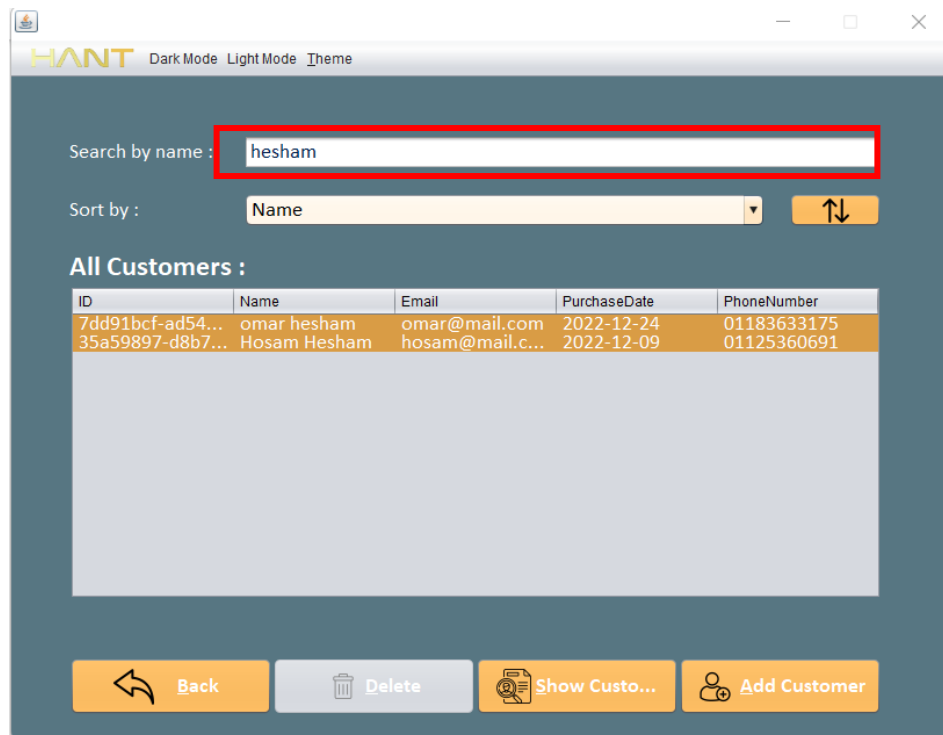
The system allows the users to filter the customers shown in the **view all customers** table.

The filtering is made of two stages:

- The Search:

The search is done using the search bar in the panel to check if a certain word exists in certain customers properties like:

- Name
- Email



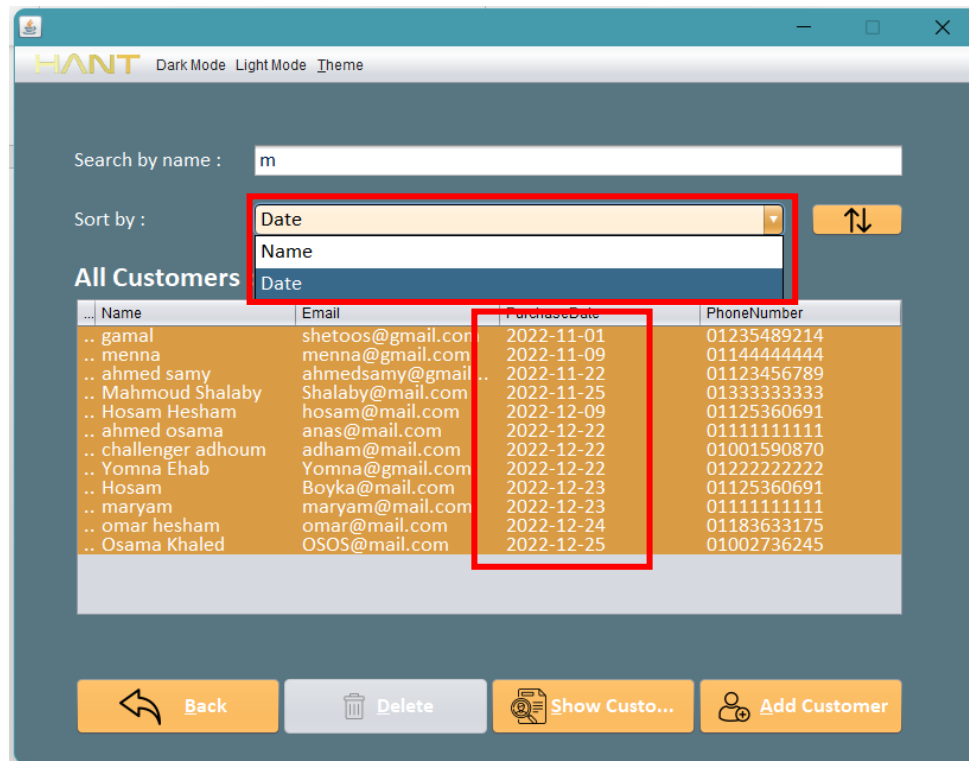
The screenshot shows the HANT application interface. At the top, there's a header with the logo and theme options (Dark Mode, Light Mode, Theme). Below the header, there's a search bar labeled 'Search by name :' with the text 'hesham' entered. To the right of the search bar is a sort dropdown menu labeled 'Sort by :' with 'Name' selected. Below the search and sort controls, there's a table titled 'All Customers :'. The table has five columns: ID, Name, Email, PurchaseDate, and PhoneNumber. The first two rows of the table are highlighted in orange. The first row shows ID '7dd91bcf-ad54...', Name 'omar hesham', Email 'omar@mail.com', PurchaseDate '2022-12-24', and PhoneNumber '01183633175'. The second row shows ID '35a59897-d8b7...', Name 'Hosam Hesham', Email 'hosam@mail.c...', PurchaseDate '2022-12-09', and PhoneNumber '01125360691'. Below the table, there's a large empty rectangular area. At the bottom of the interface, there's a navigation bar with four buttons: 'Back' (with a left arrow icon), 'Delete' (with a trash icon), 'Show Custo...' (with a magnifying glass icon), and 'Add Customer' (with a person icon).

ID	Name	Email	PurchaseDate	PhoneNumber
7dd91bcf-ad54...	omar hesham	omar@mail.com	2022-12-24	01183633175
35a59897-d8b7...	Hosam Hesham	hosam@mail.c...	2022-12-09	01125360691

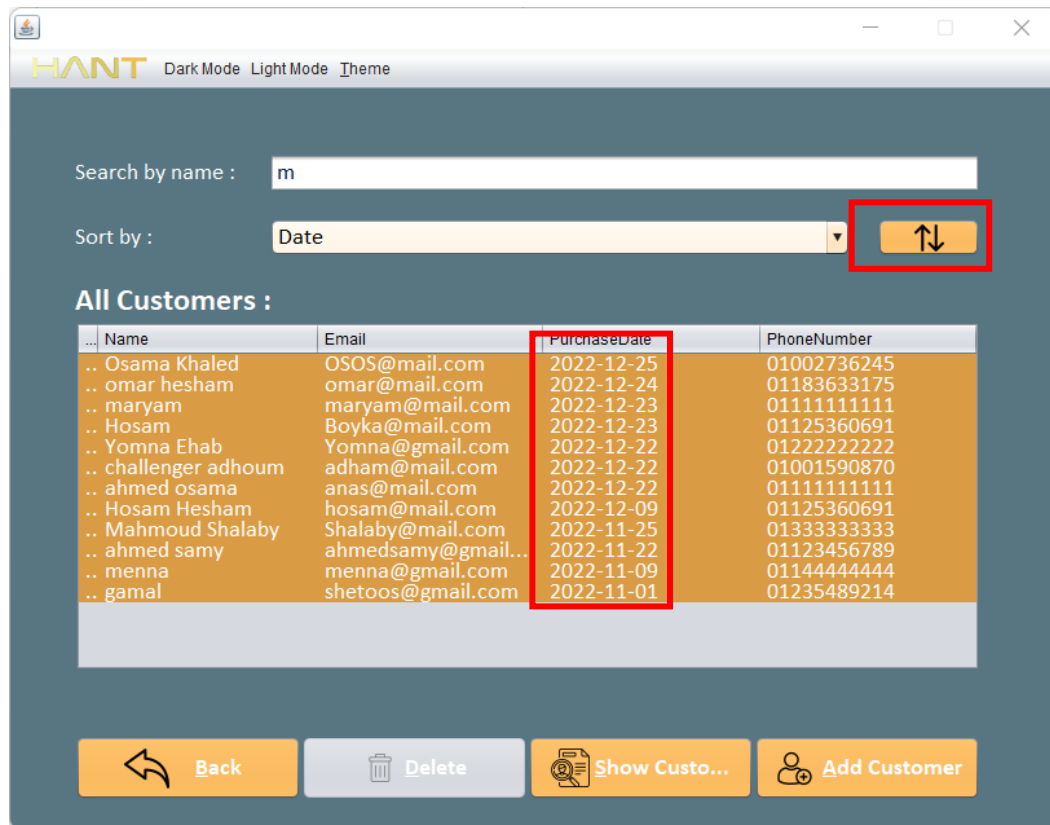
- The Sort:

The sort is done using the sort combo box in the panel to view all the filtered customer through the search stage in a certain order according to the chosen choice:

- Date
- Name



- The sort can be toggled using the toggle button the change the order of the list (ascending/descending)



- **Filtering Items:**

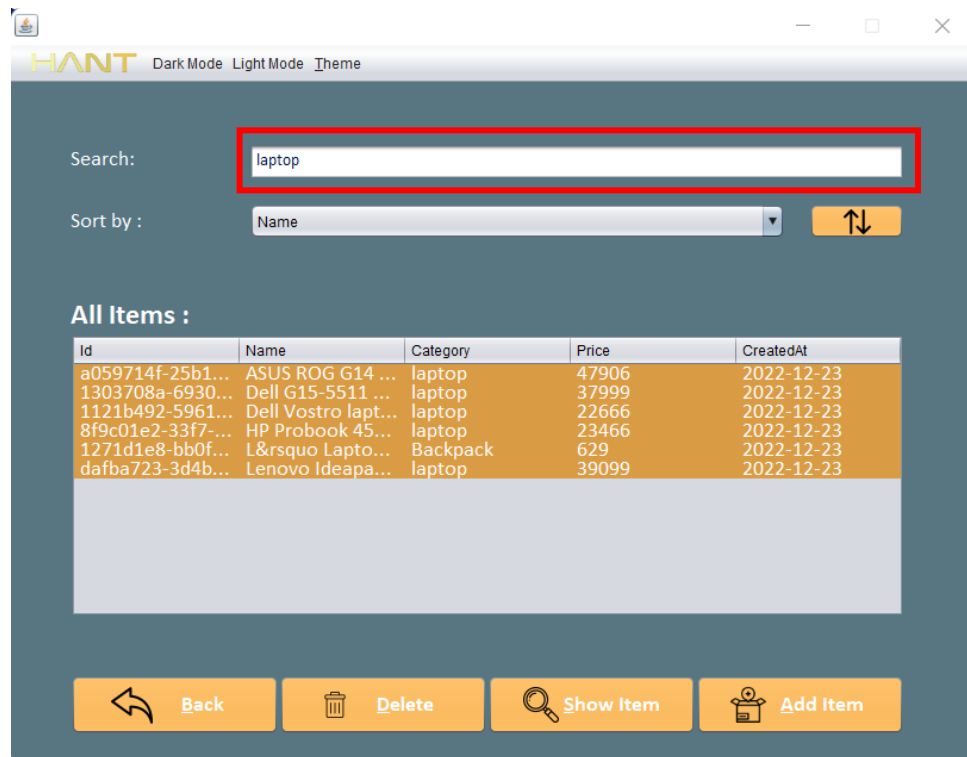
The system allows the users to filter the Items shown in the **view all Items** table.

The filtering is done through two stages:

- The Search:

The search is done using the search bar in the panel to check if a certain word exists in certain Items properties like:

- Name
- Category

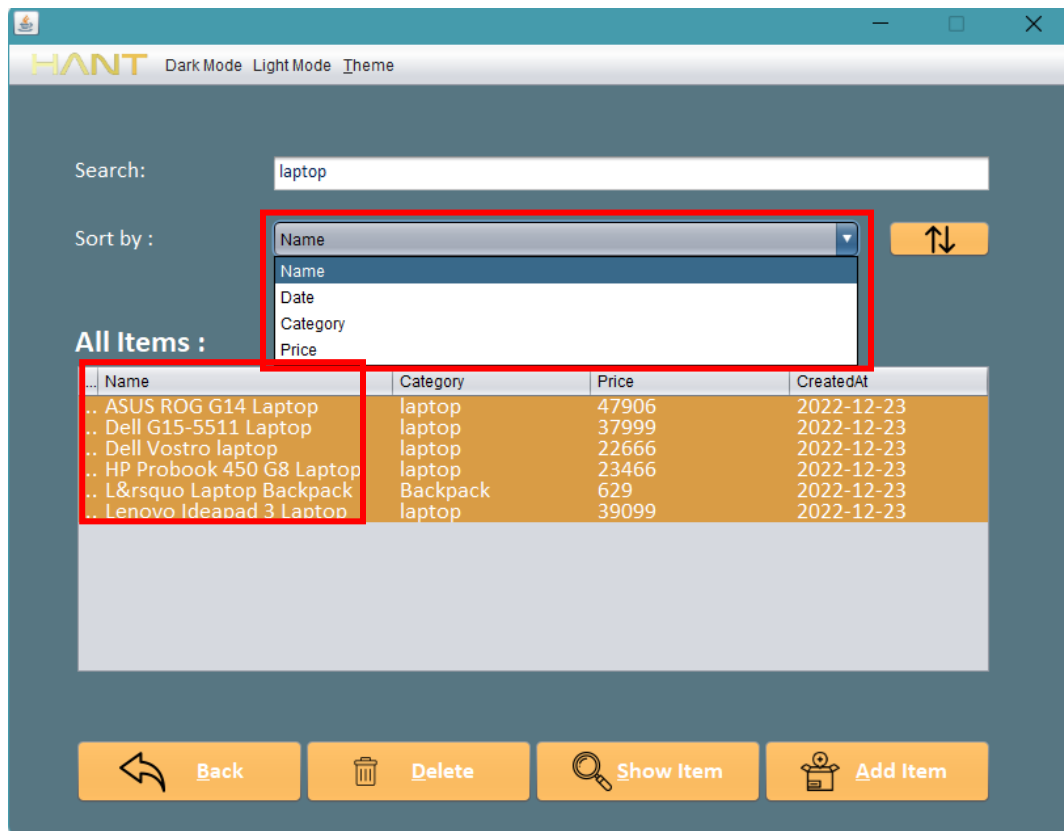


○ The Sort:

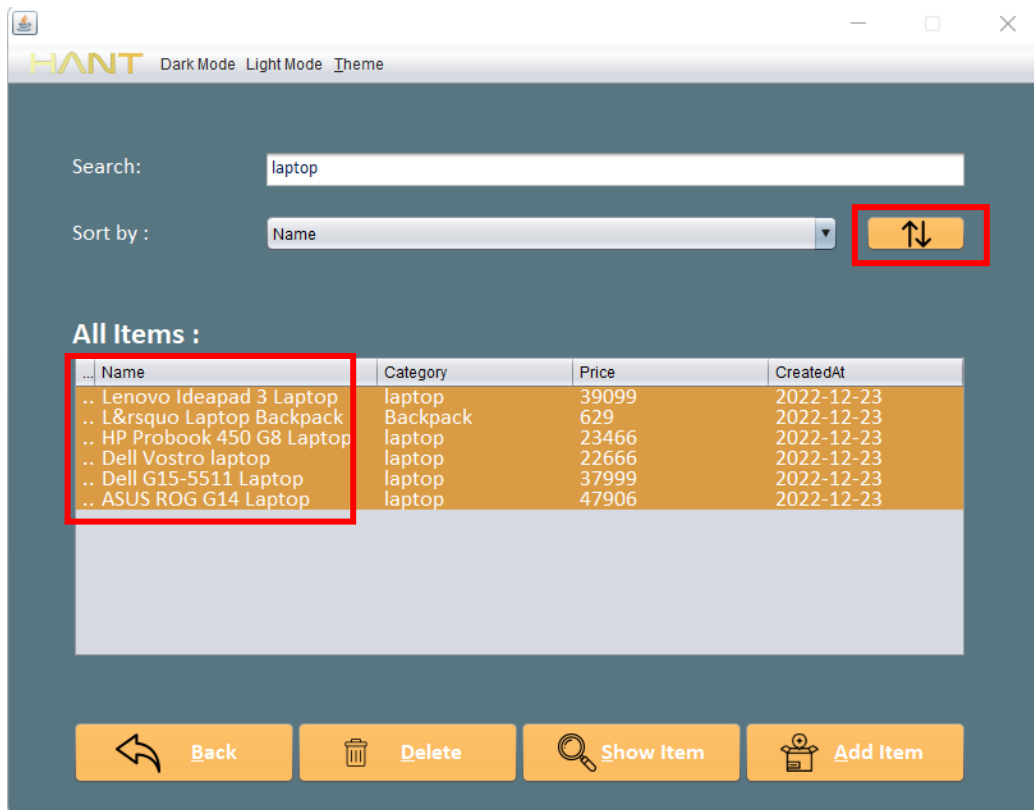
The sort is done using the sort combo box in the panel to view all the filtered items through the search stage in a certain order according to the chosen choice:

- Name
- Date (Created At)
- Category
- Price





- The sort can be toggled using the toggle button the change the order of the list (ascending/descending)



- **Filtering Users:**

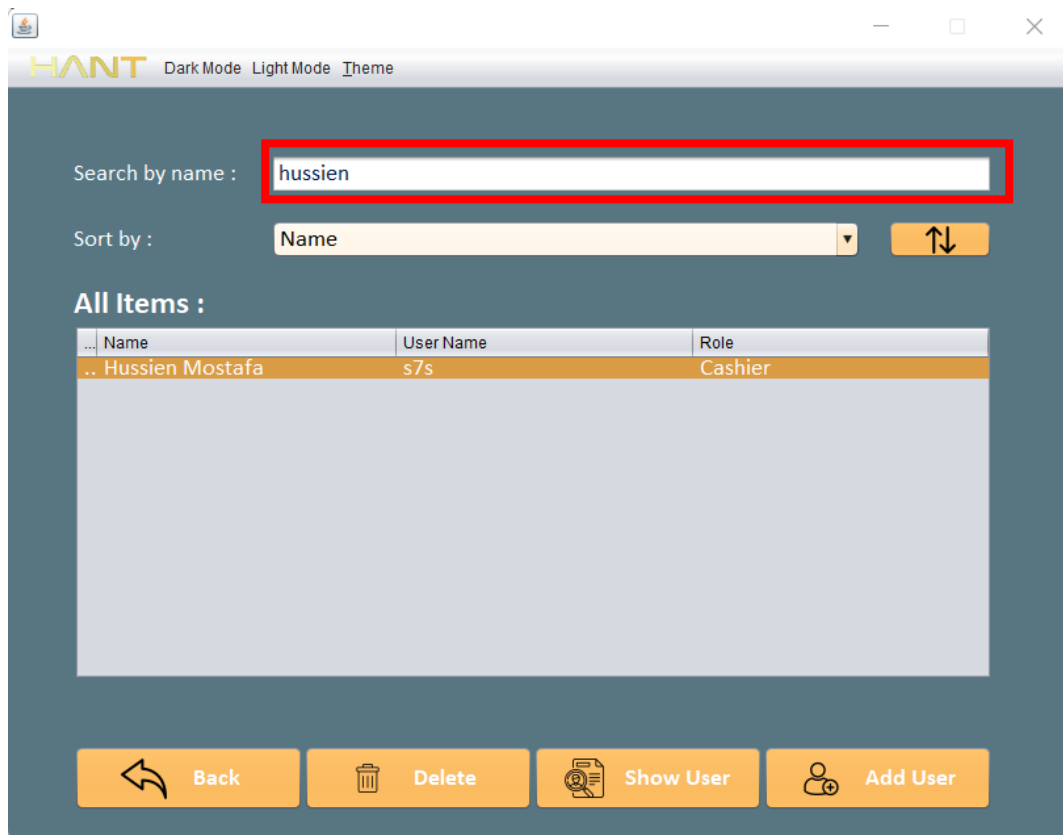
The system allows the users to filter the users shown in the **view all Users** table.

The filtering is done through two stages:

- The Search:

The search is done using the search bar in the panel to check if a certain word exists in certain Users properties like:

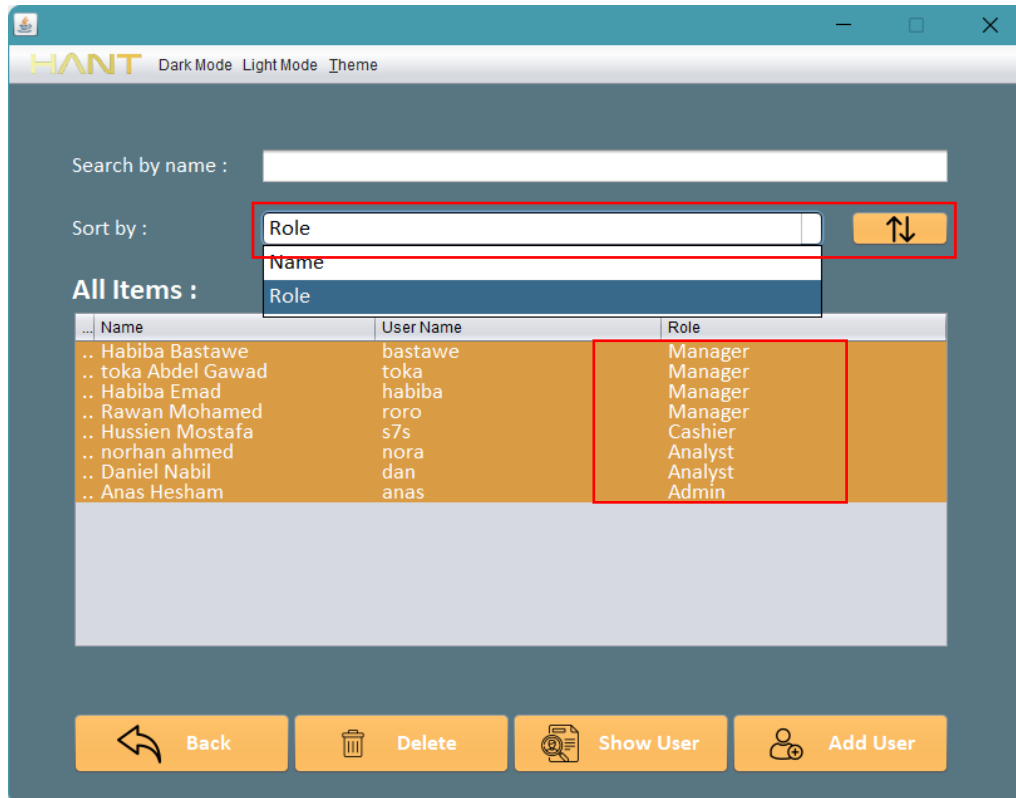
- Name
- Username



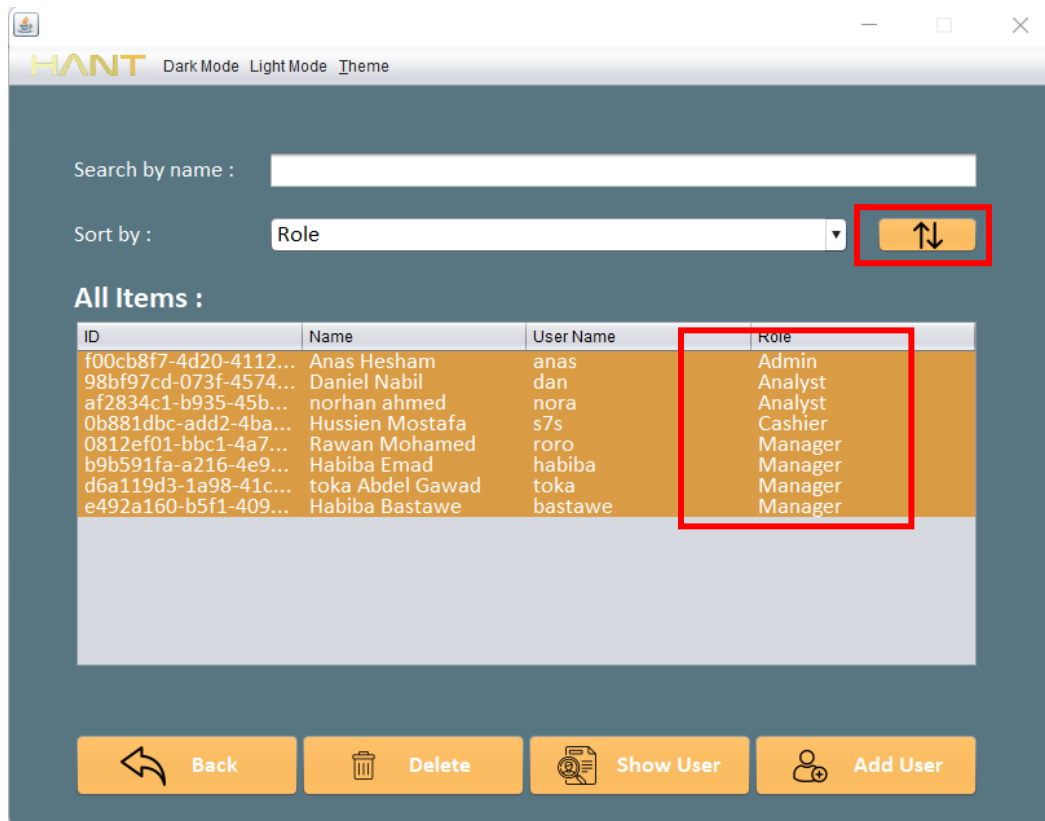
- The Sort:

The sort is done using the sort combo box in the panel to view all the filtered users through the search stage in a certain order according to the chosen choice:

- Name
- Role



- The sort can be toggled using the toggle button the change the order of the list (ascending/descending)



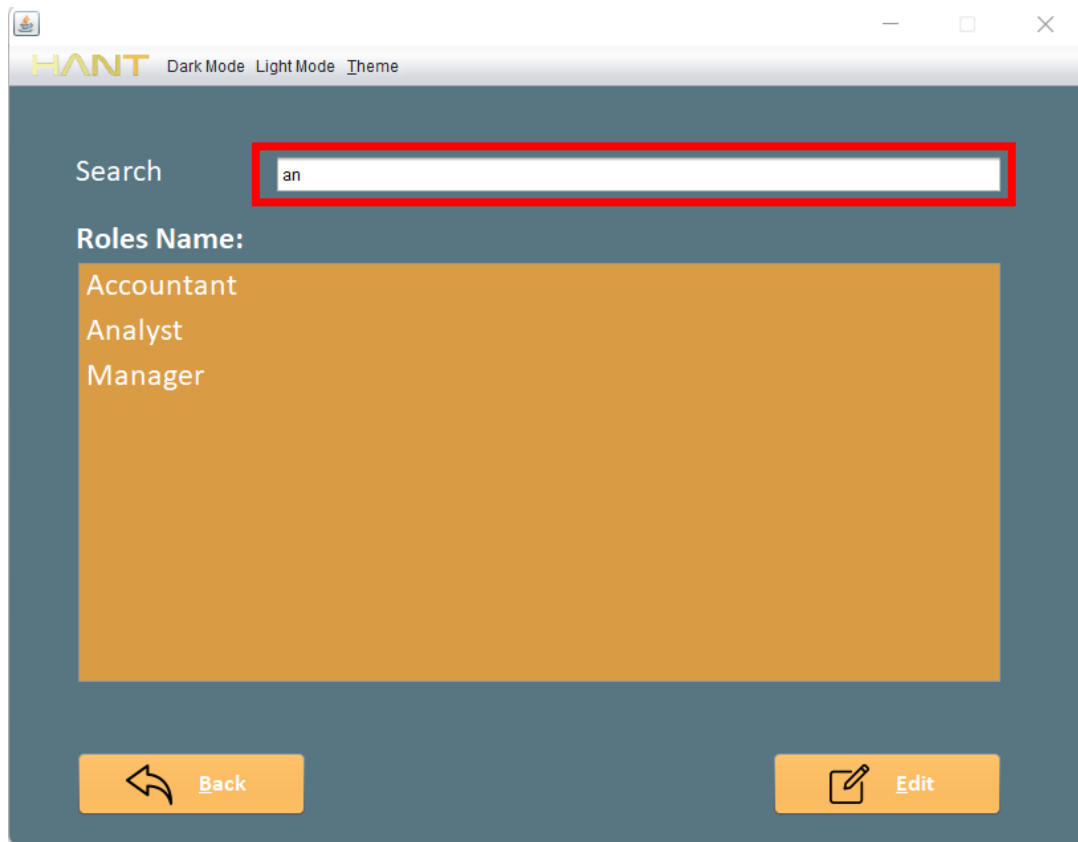
- **Filtering Roles:**

The system allows the users to filter the users shown in the **view all Role** table.

The filtering is done through the search stage:

- The Search:

The search is done using the search bar in the panel to check if a certain word exists in Role's name.



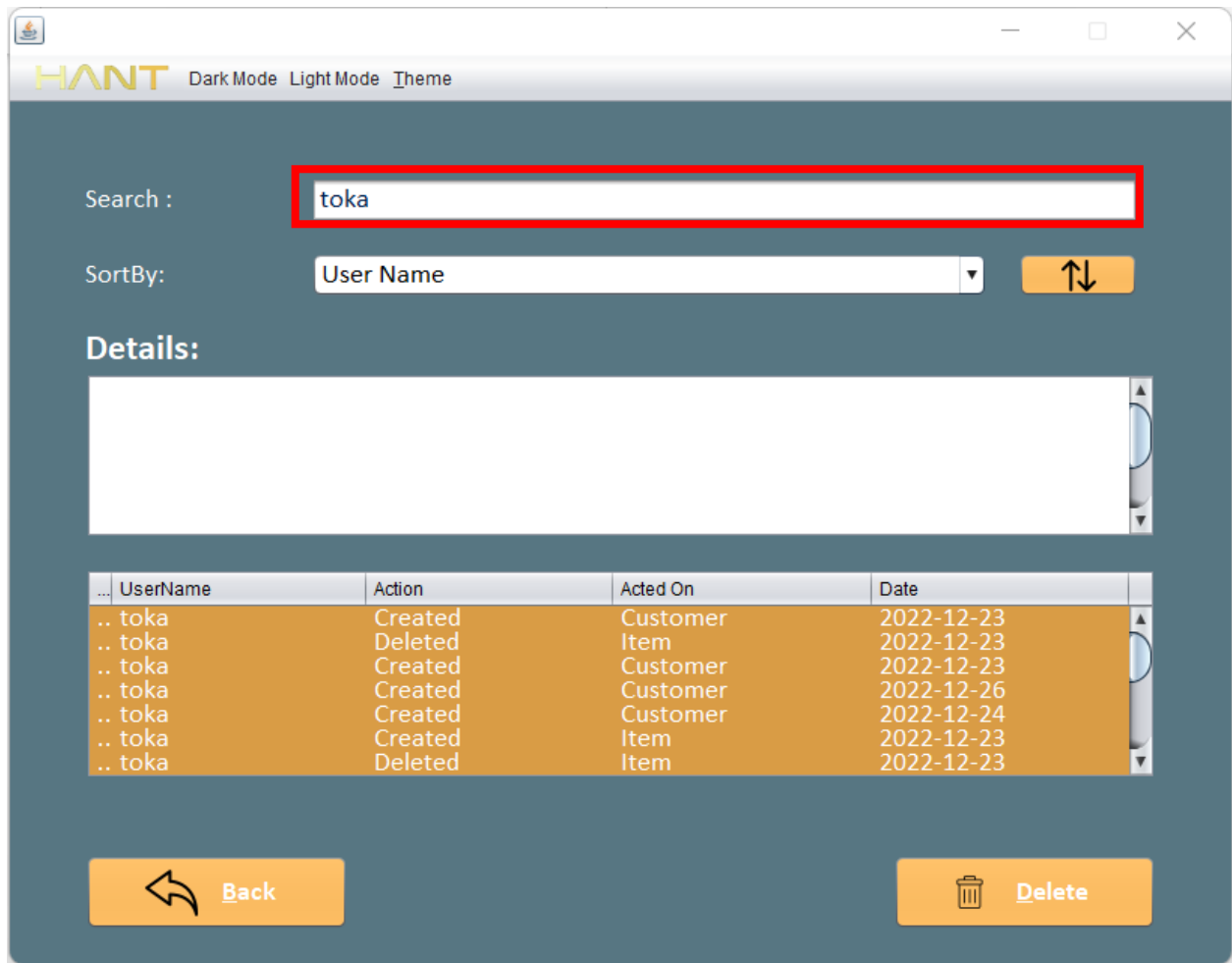
- **Filtering Logs:**

The system allows the users to filter the logs shown in the **view all Logs** table.

The filtering is done through two stages:

- The Search:

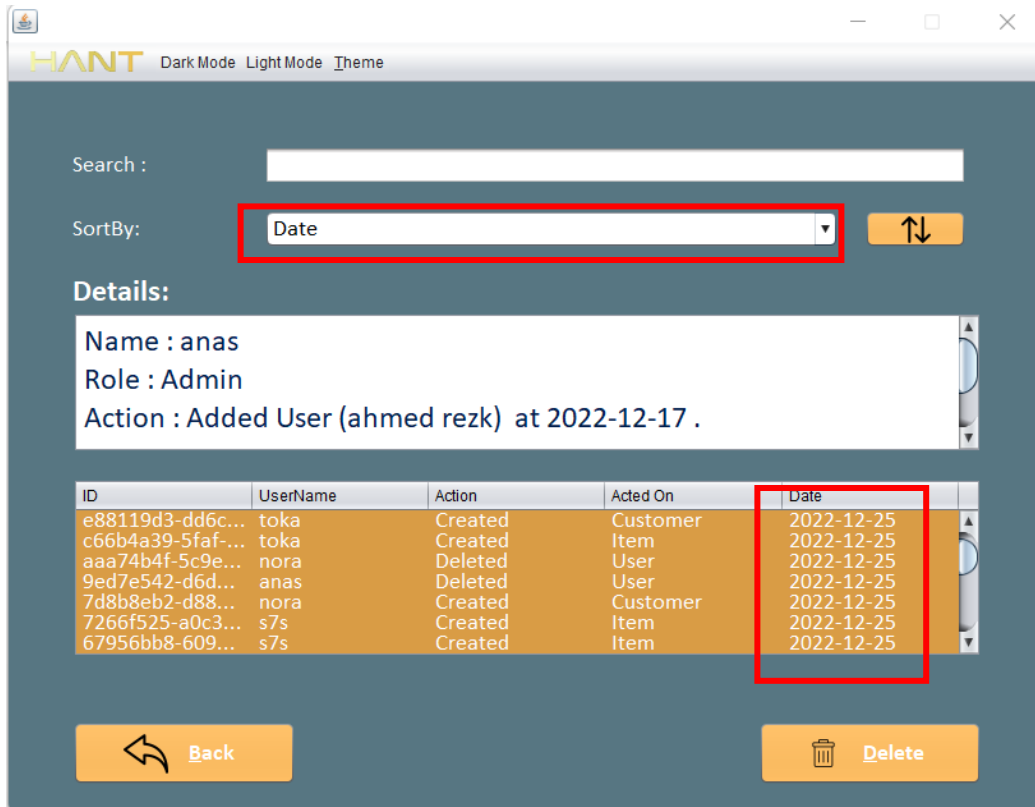
The search is done using the search bar in the panel to check if a certain word exists in username.



○ The Sort:

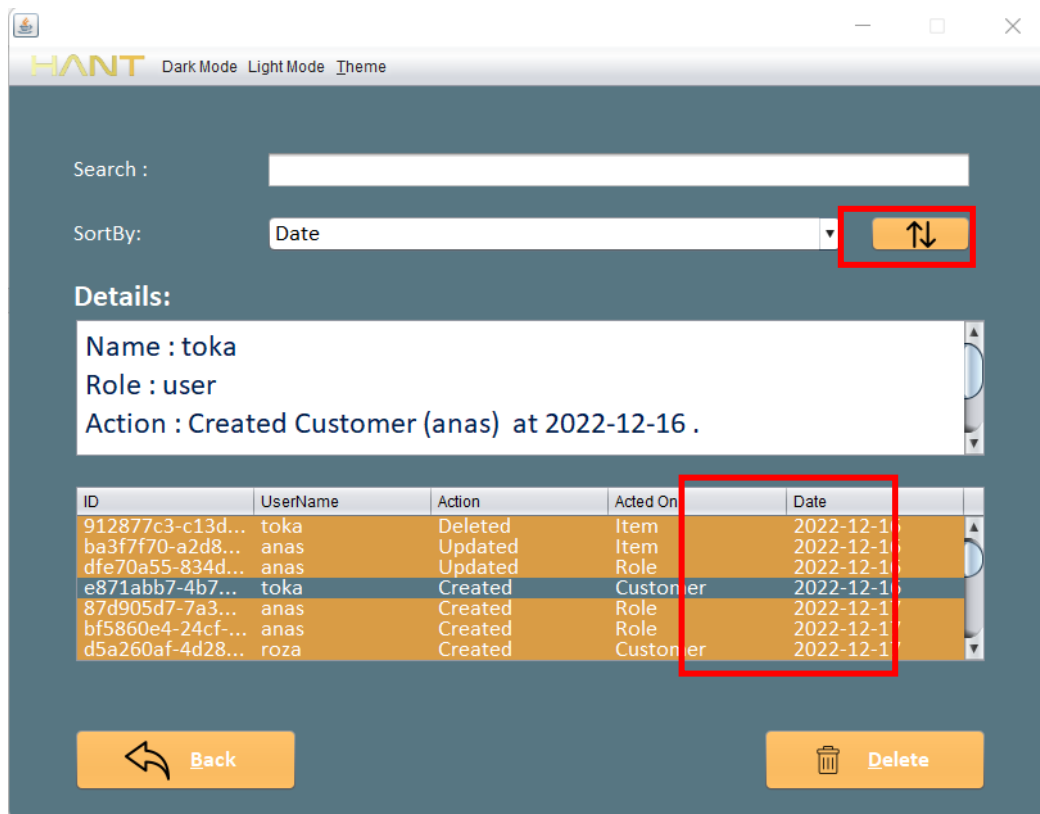
The sort is done using the sort combo box in the panel to view all the filtered logs through the search stage in a certain order according to the chosen choice:

- Username
- Action
- Acted On
- Date



- The sort can be toggled using the toggle button the change the order of the list (ascending/descending)



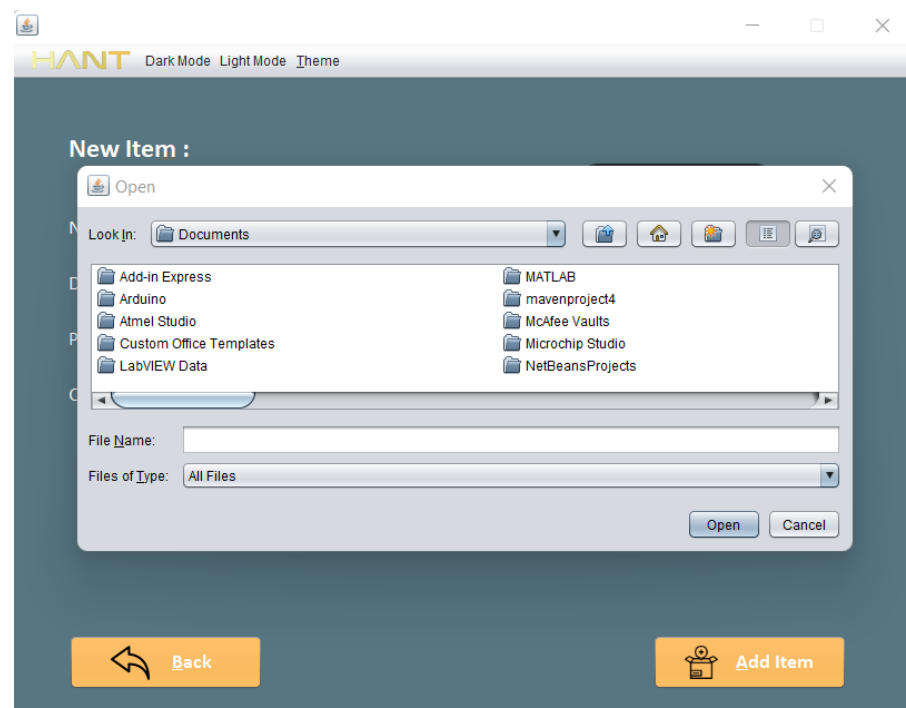
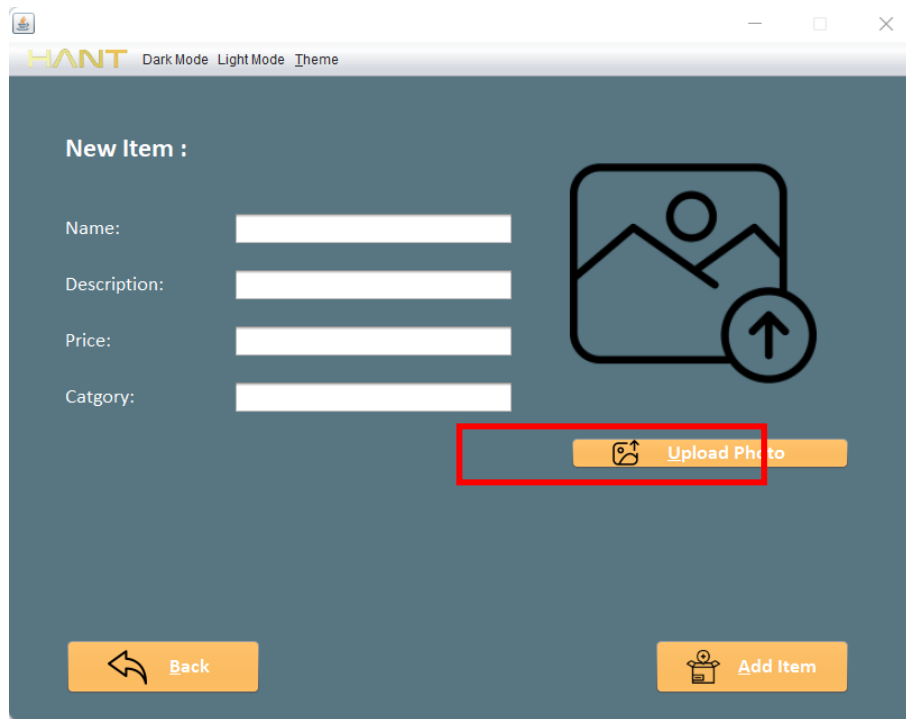


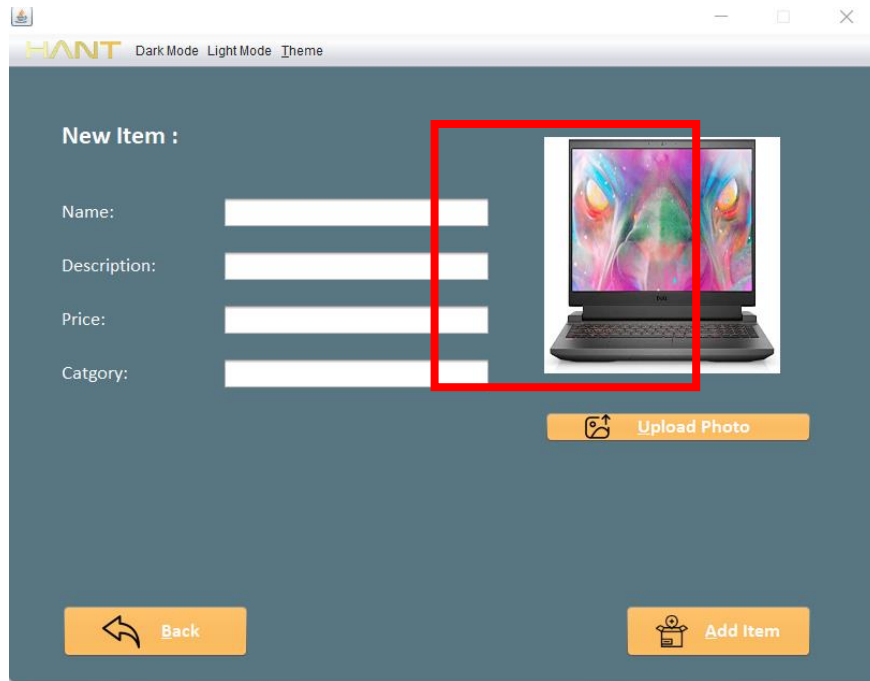
- **Upload Image to Items:**

The system allows users to upload a picture while importing new item to the store database.

Which helps customers in visualizing the items before adding it to their carts.

This operation is done using the Swing file uploader panel to make it easy for the user to select an image from a local location on the device they are on.





The selected picture will be added to the database in a sperate table called **Images**, the name of the entry is constructed using the created item id (UUID) + the Image Extension.

- **View/Update Uploaded Image:**

The system allows customers to view the image of a certain item using the **view Item** component.

The Image is fetched from the database using the id of the item through the **Images** Table -> name.

The screenshot shows a web application window titled 'HANT' with a dark theme. The 'Item Data' form is displayed, containing the following fields and values:

Field	Value
Name	Decker Air Fryer
Description	1500 Watt, 4 Liters, Black
Price	4299
Created At	2022-00-23
Category	Air Fryer

To the right of the form is a placeholder image of a black Decker air fryer. Below the image is an 'Upload Photo' button. At the bottom of the form are four buttons: 'Back', 'Cancel', 'Update', and 'Edit'.

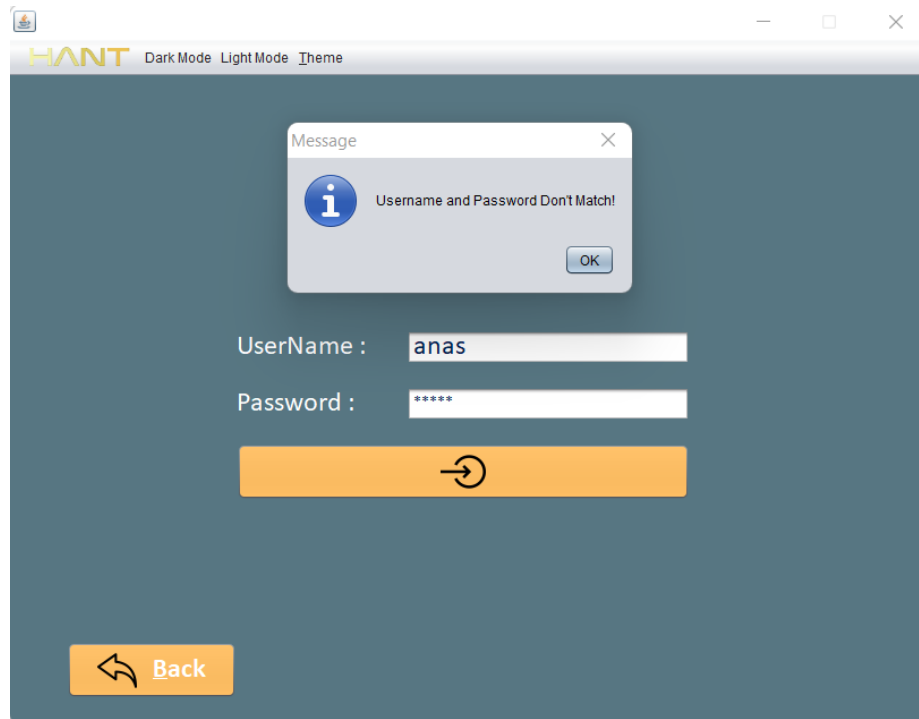
The Edit Image process is done by deleting the existed Image in the table Images and add a new entry with the same exact name. (Item id + the image extension).

- **User Input Validations:**

The validation is done to each panel to verify that the user using the application has entered correct data types and values

To inform whether the entry was accepted or not and to inform him with the constraints and the required data types we used the JOption Pane Message Dialogue.

- Login page:
  - Username Field: Can't Be Empty
  - Password Field: Can't Be Empty
  - The Username and Password must match an existing user in the database.



- Add/Update Customer page:
  - Name: Can't Be Empty and can't contain special characters or numbers in it.
  - Email: Can't Be Empty and should be in the form of an Email.
  - National Id: Can't Be Empty, should only contain digits and must be exactly 14 digits
  - Phone Number: Can't Be Empty, should only contain digits, must start with 01 and must be exactly 11 digits
  - Age: Can't Be Empty, should only contain digits and can only be positive numbers.
  - Choose Item: Every Customer should select at least One item.

The screenshot shows a web application window titled "HANT" with a dark theme. The header includes "Dark Mode", "Light Mode", and "Theme" links. The main form contains the following fields and controls:

- Name:** Text input with "ahmed" entered.
- Email:** Text input with "ahme" entered.
- National ID:** Text input with "12345" entered.
- Phone No:** Text input with "01222" entered.
- Age:** Text input with "12" entered.
- Gender:** Dropdown menu with "Male" selected.
- Purchases:** A button with a shopping cart icon and the text "Choose Item".
- Navigation:** "Back" button with a left arrow icon and "Add Customer" button with a person icon.

A modal message box is displayed in the center, titled "Message", with an information icon and the text: "You Must Choose atleast one item!". It has an "OK" button.

- Add/Update item page:
  - Name: Can't Be Empty
  - Price: Can't Be Empty, can't have letter or special characters and can't be negative
  - Category: Can't Be Empty

The screenshot shows the "New Item" form in the HANT application. The form includes the following fields and controls:

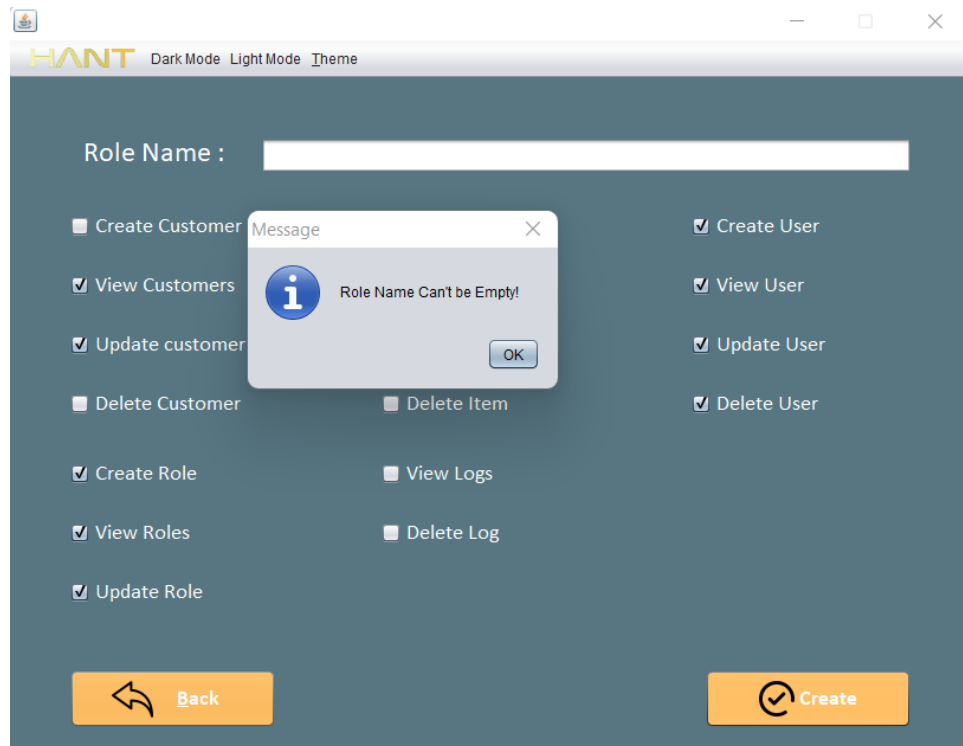
- Name:** Text input with "item" entered.
- Description:** Text input with "item" entered.
- Price:** Text input with "2000asd" entered.
- Catgory:** Text input (note the typo).
- Image Upload:** A large icon representing a photo upload and a button labeled "Upload Photo" with a camera icon.
- Navigation:** "Back" button with a left arrow icon and "Add Item" button with a box icon.

A modal message box is displayed in the center, titled "Message", with an information icon and the text: "Please Enter a valid number!". It has an "OK" button.

- Add/Update User page:
  - Name: Can't be Empty and can't contain numbers or special characters.
  - Username: Can't be Empty.
  - Password: Can't be less than 8 characters.

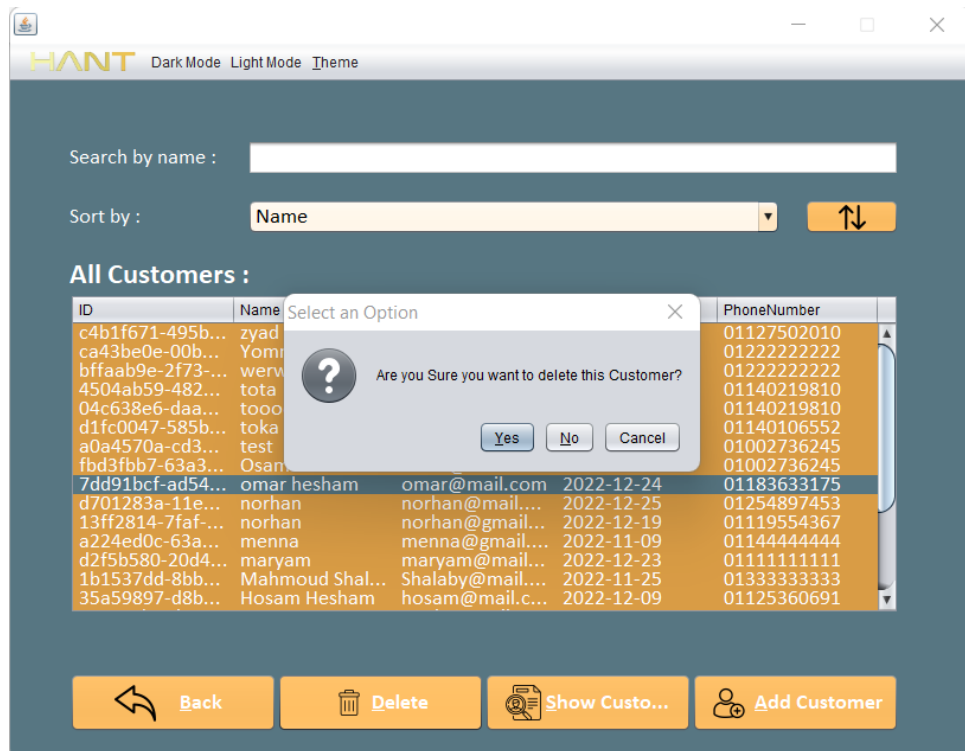
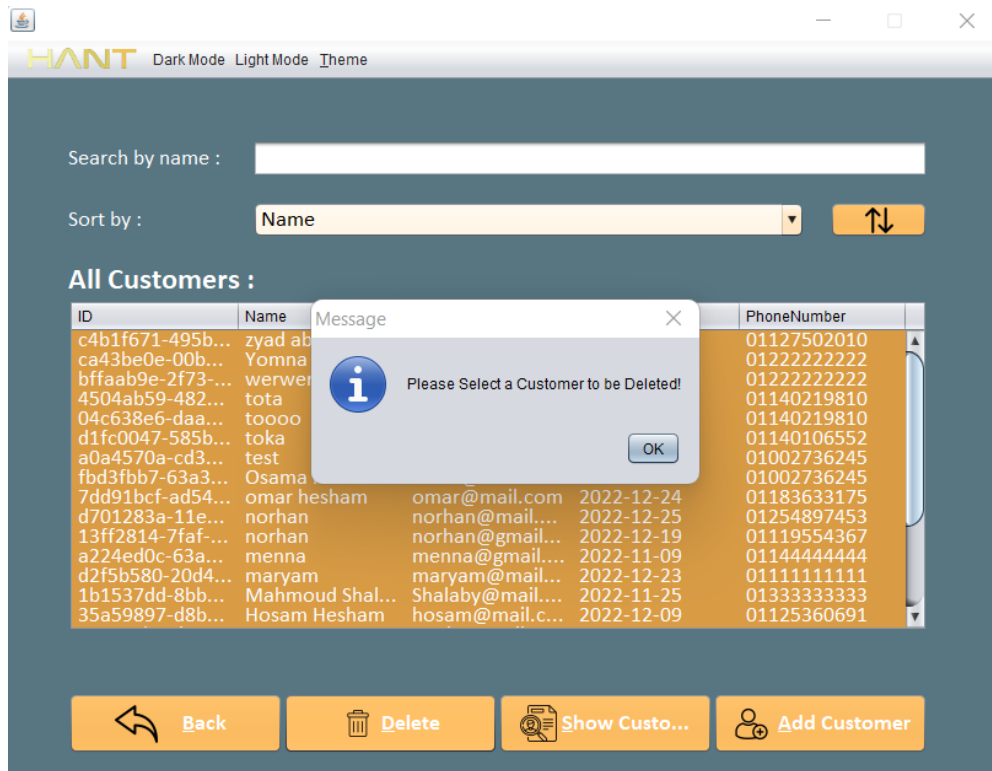
The screenshot shows a web application window titled 'HANT' with a dark theme. The main content area is titled 'New User:' and contains four input fields: 'Name:', 'UserName:', 'password:', and 'Role:'. The 'password:' field is filled with six asterisks. A modal message box is overlaid on the form, displaying an information icon and the text 'Password Can't be less than 8 Chars!'. At the bottom of the form, there are two orange buttons: 'Back' with a left arrow icon and 'Add User' with a user icon and a plus sign. The top of the window has a header bar with the 'HANT' logo and links for 'Dark Mode', 'Light Mode', and 'Theme'.

- Add Role page:
  - Name: Can't Be Empty.

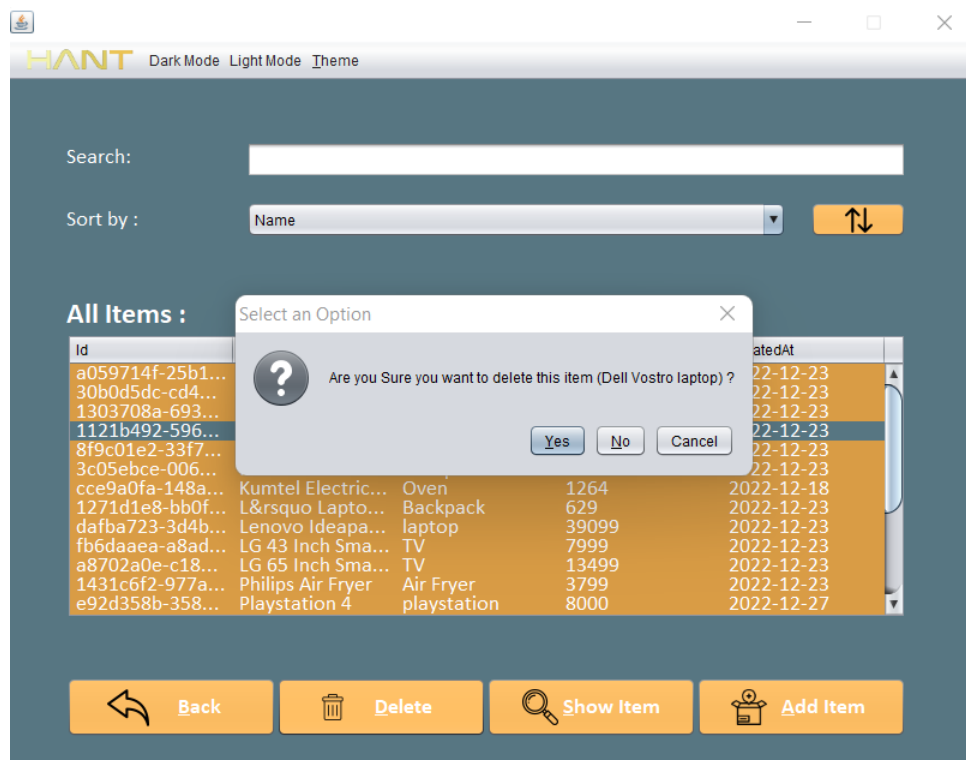


- View Customers page:
  - Delete Customer Operation: Must select a customer to be deleted, then a Confirm dialogue is shown to confirm the deletion so the accidental deletion is prevented.
  - Show Customer Operation: Must select a customer to be shown.

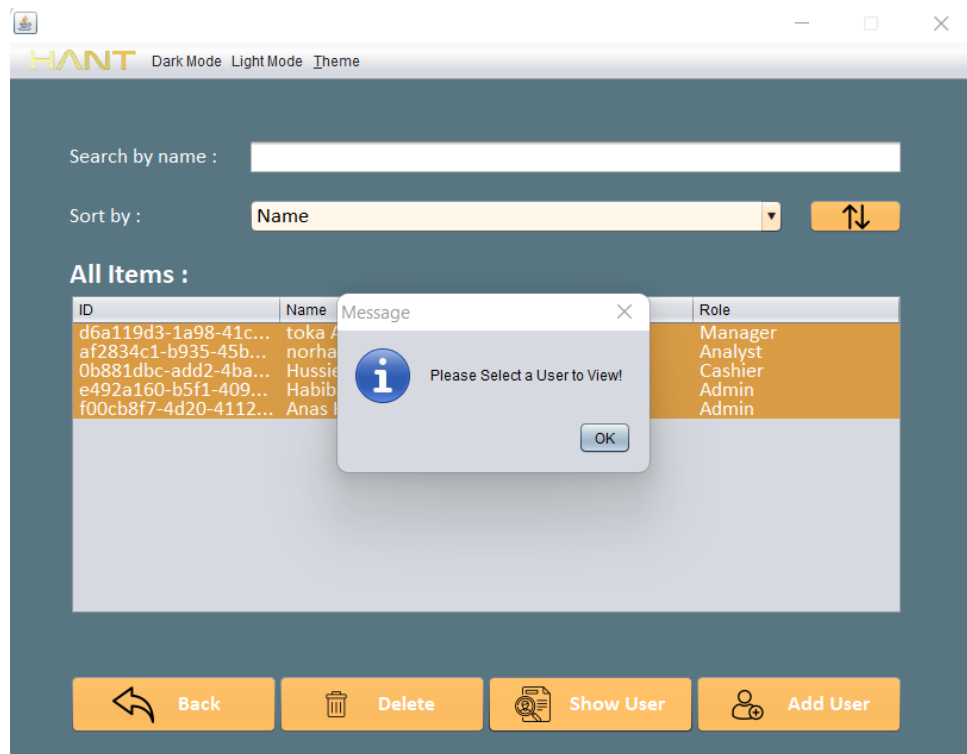




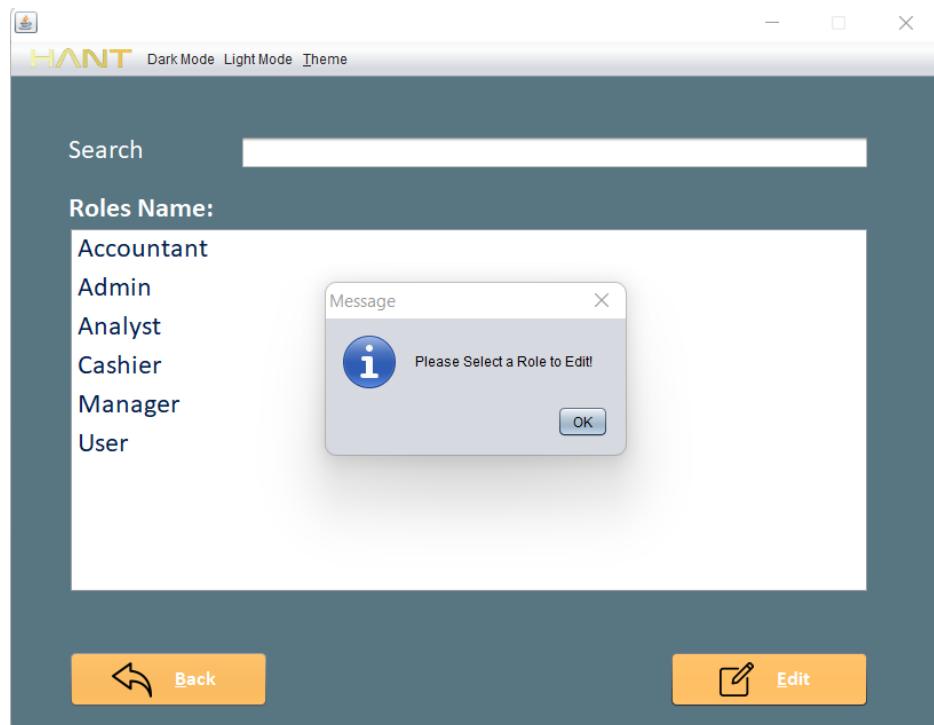
- View Items Page:
  - Delete Item Operation: Must select an item to be deleted, then a Confirm dialogue is shown to confirm the deletion so the accidental deletion is prevented.
  - Show Item Operation: Must select an Item to be shown.



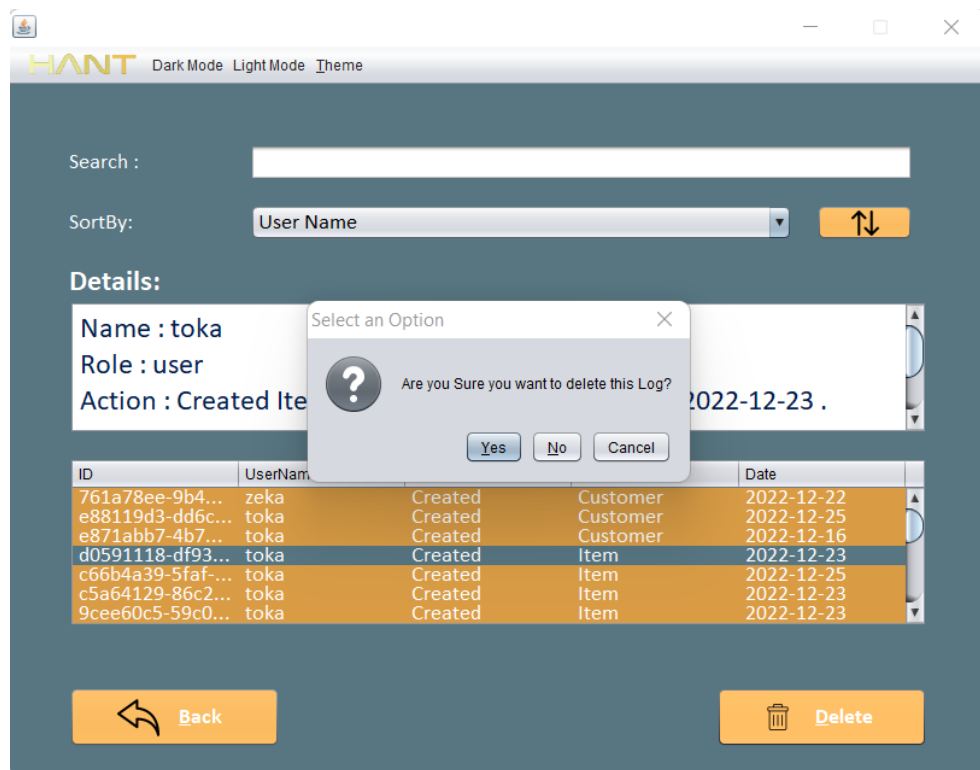
- View Users Page:
  - Delete User Operation: Must select a User to be deleted, then a Confirm dialogue is shown to confirm the deletion so the accidental deletion is prevented.
  - Show User Operation: Must select a User to be shown.



- View Roles page:
  - Edit Role Operation: Must select a Role to be shown.

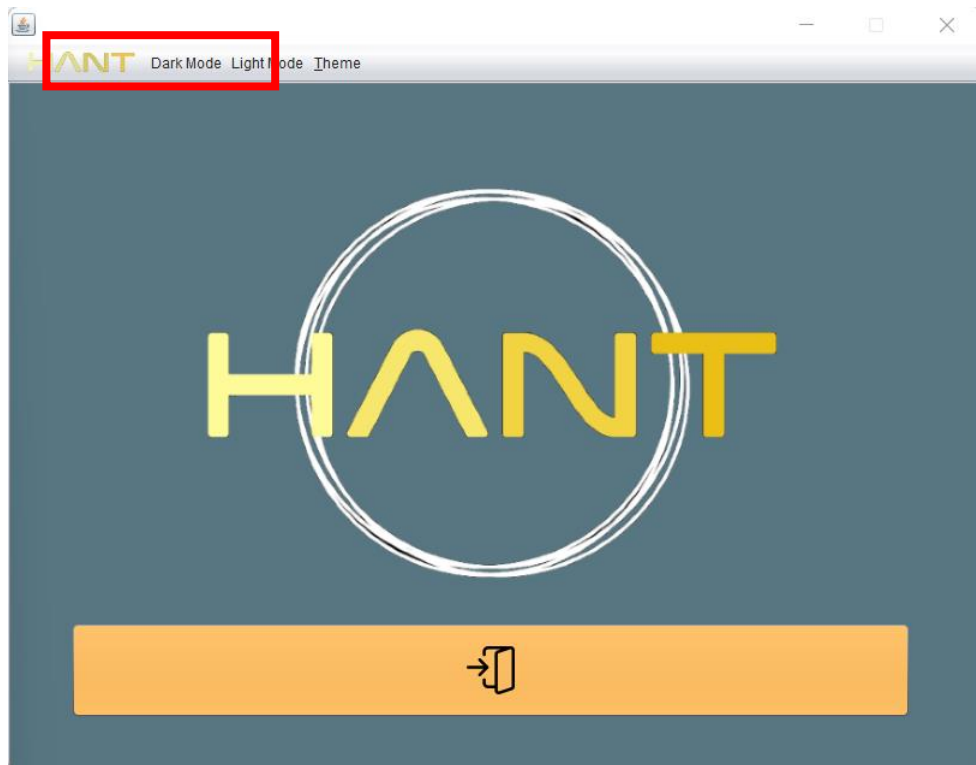


- View Logs page:
  - Delete Log Operation: Must select a Log to be deleted, then a Confirm dialogue is shown to confirm the deletion so the accidental deletion is prevented.



- **Change Theme:**

The system provides user the ability to change all the visual colors for all the panels backgrounds and the other components to have a better visual experience.



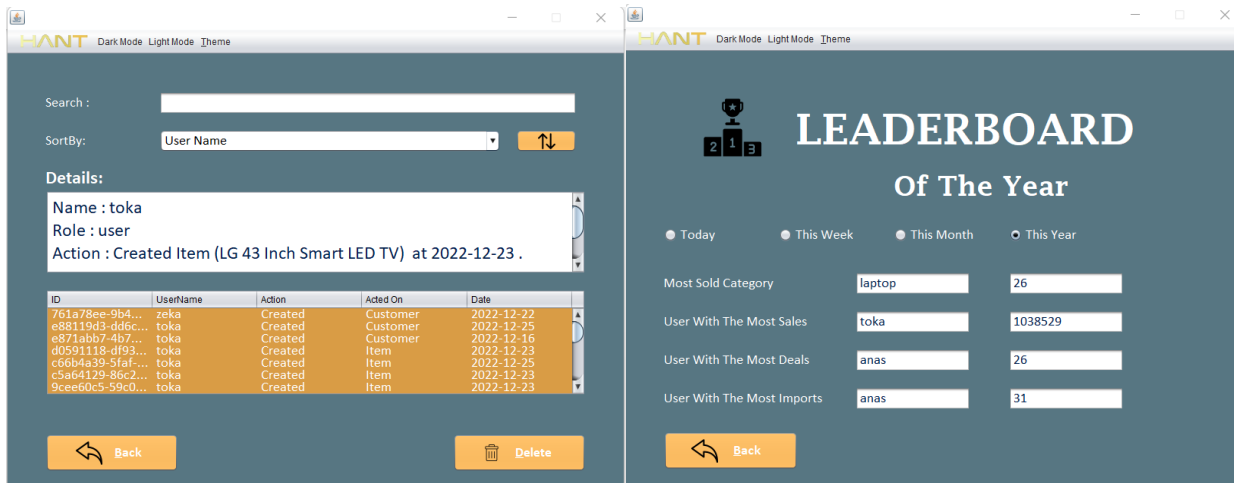
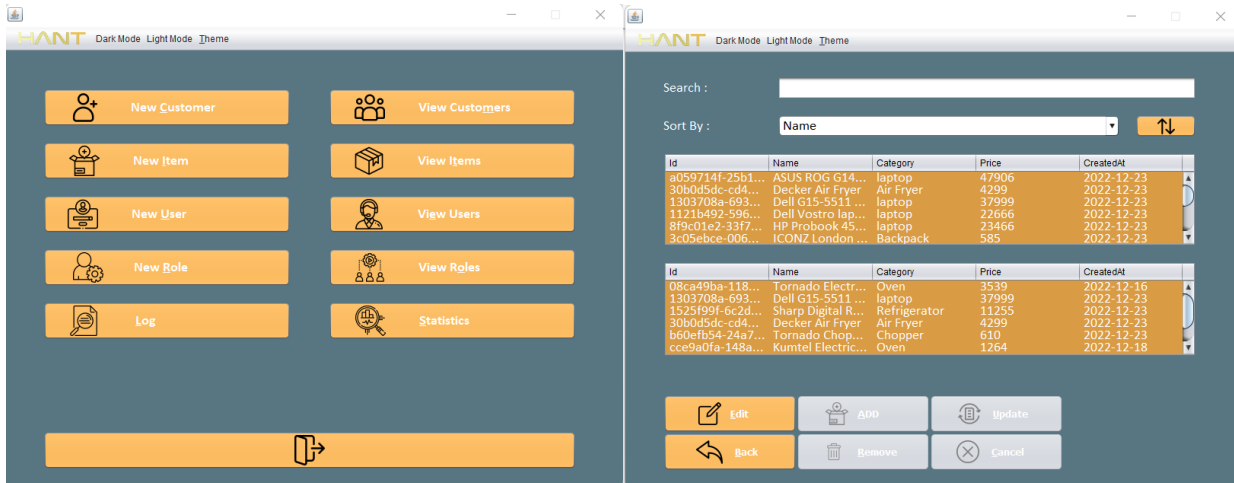
The Theme consists of 5 main Colors:

- Panels Background Color
- Buttons Color & Tables Color
- Labels Color
- Button's Text Color
- Text Fields & Combo boxes

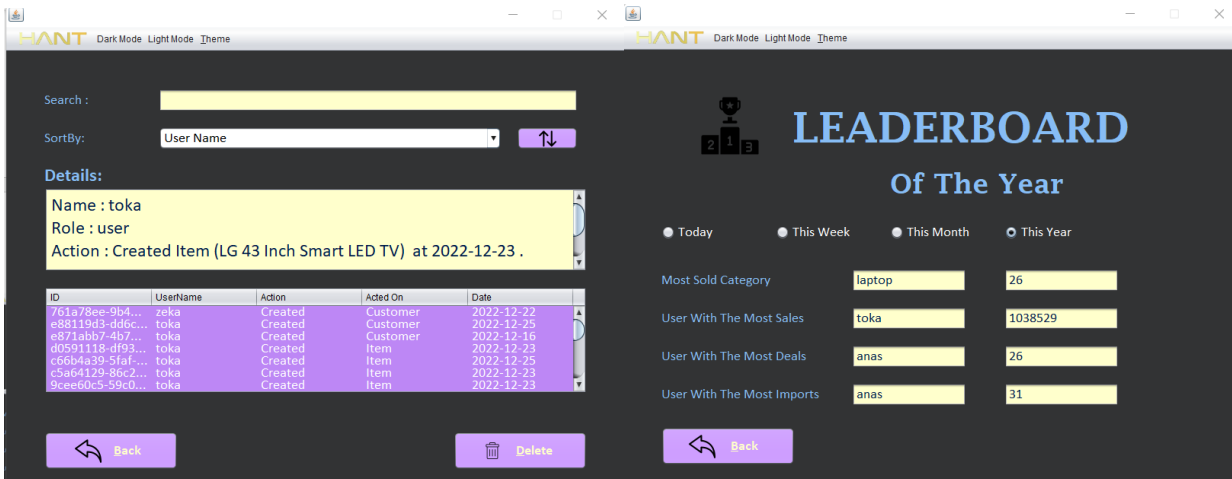
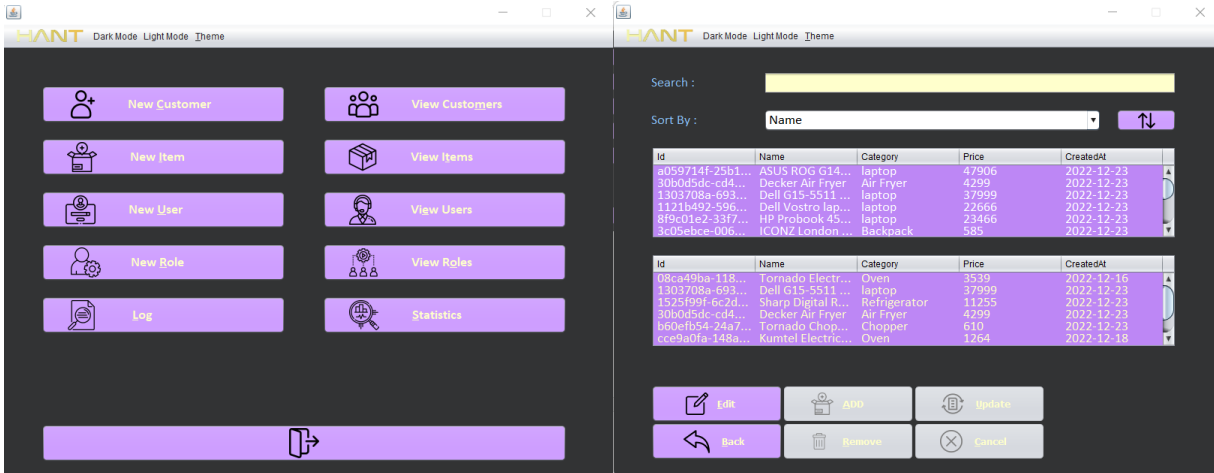
The Theme is set using Three main components:

- Light Mode (Default Theme):

The light mode is the default theme set by the application on the startup.

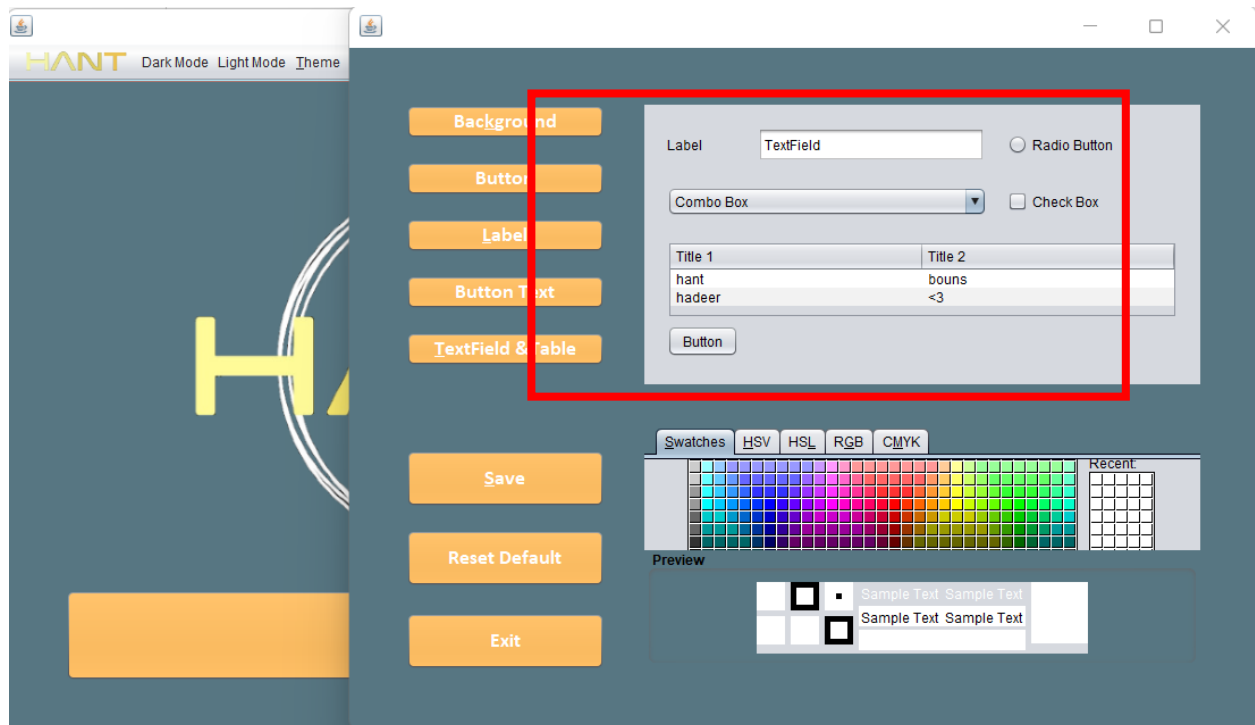


## ○ Dark Mode.



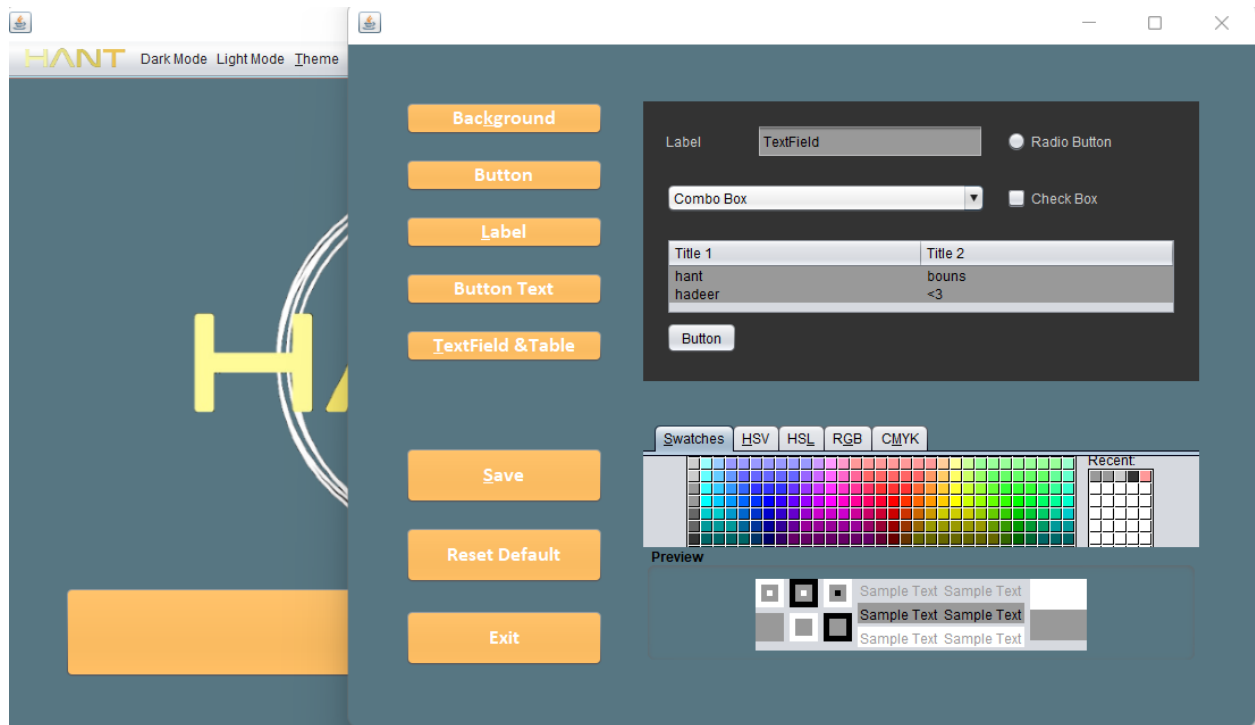
- Customized Theme

The Customized Theme option gives users the opportunity to choose each of the 5 main colors to be applied to their view. Customizing the theme is done using a **Change Theme** frame which gives users the ability to preview the full visualization before applying the theme to all panels.

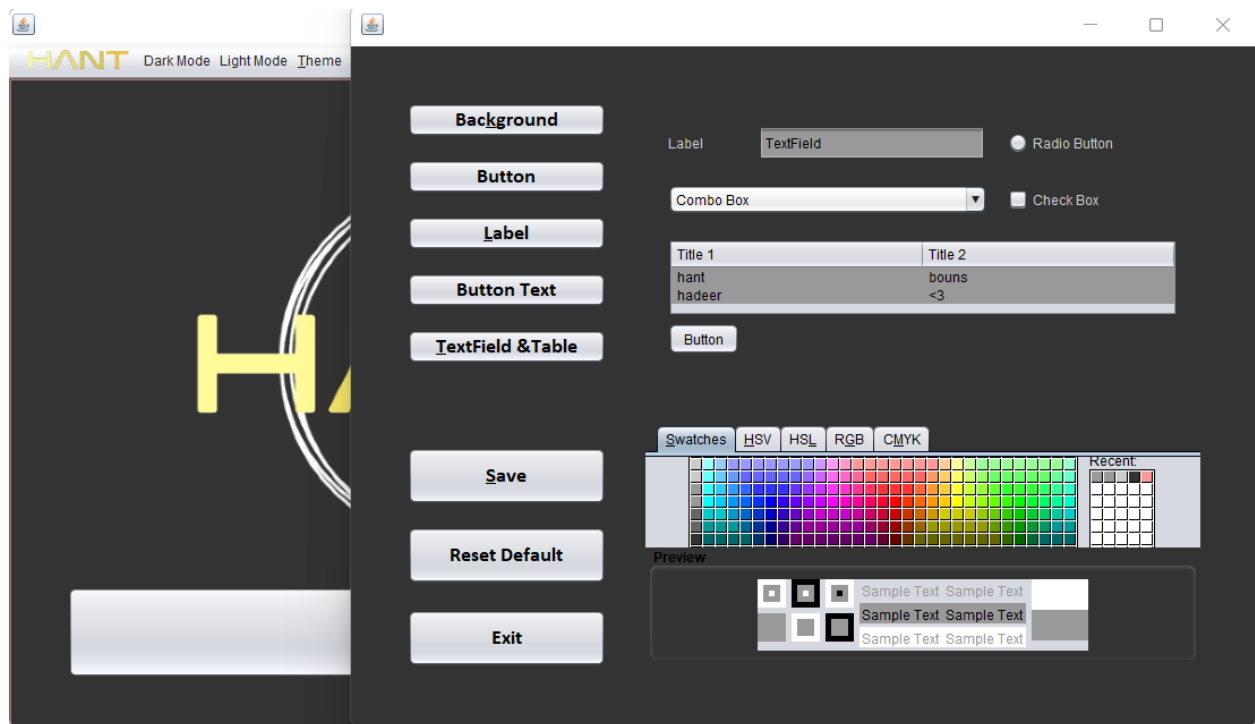


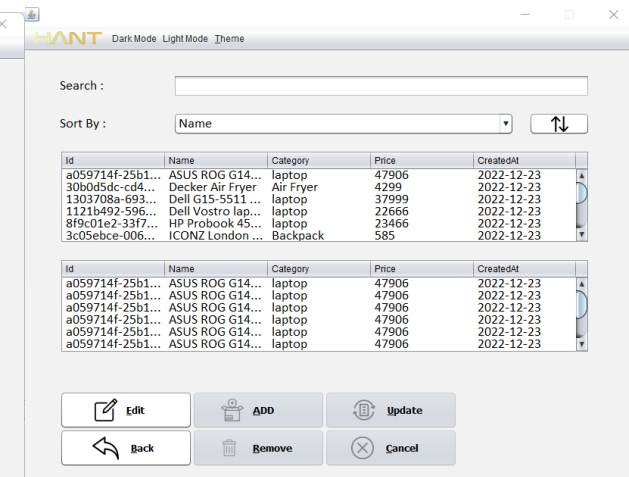
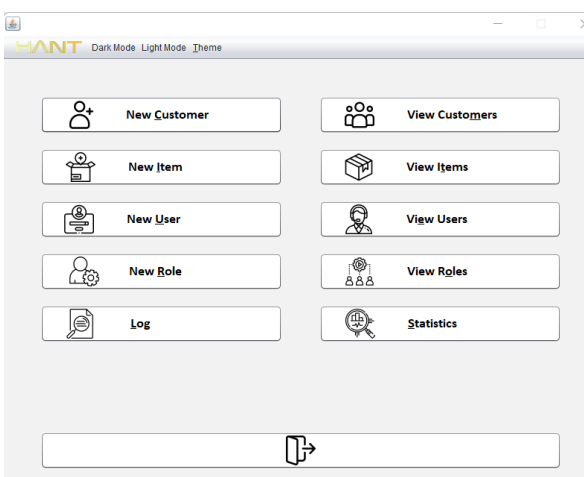
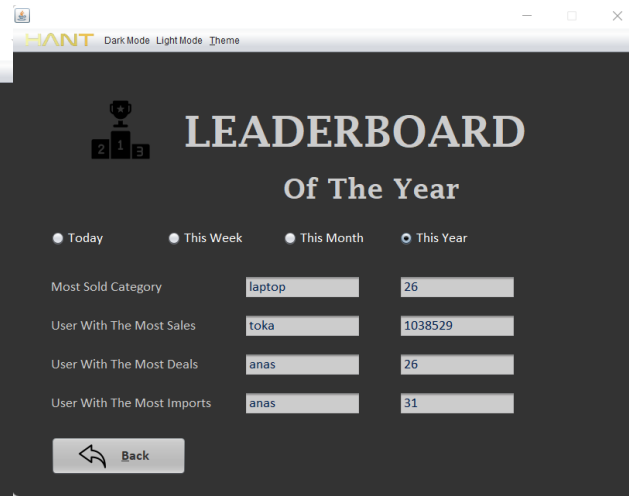
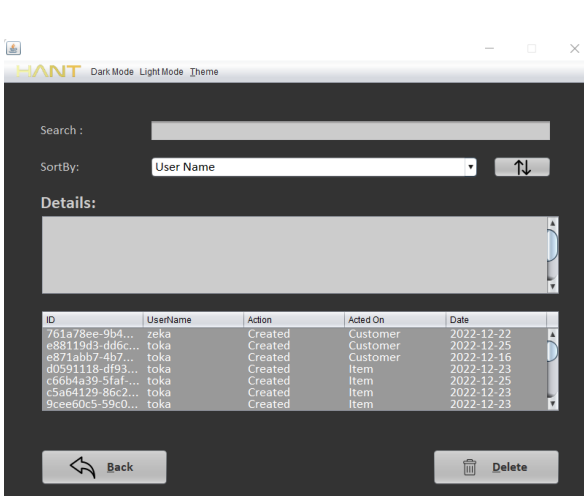
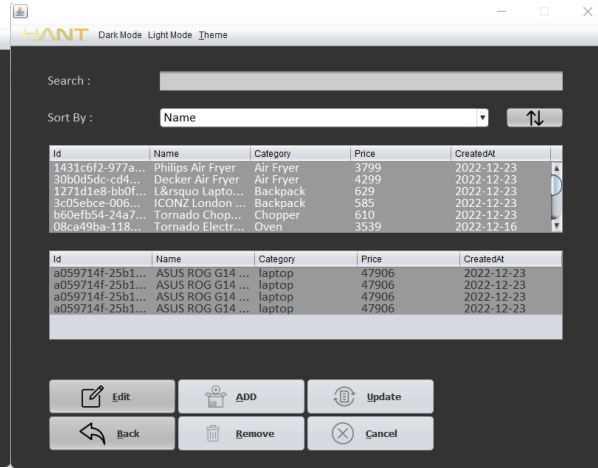
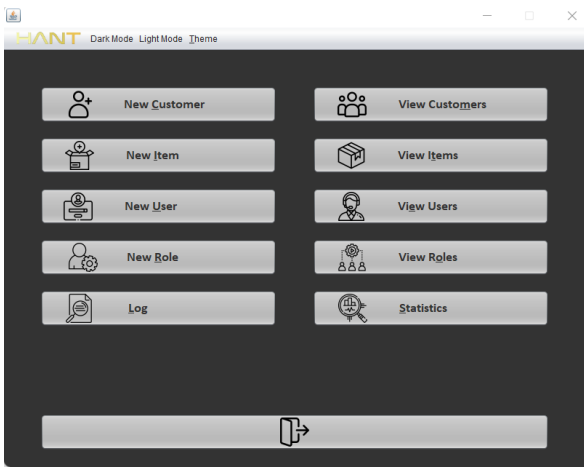
Using the Color chooser and the 5 selection buttons, the users change the test panel to see the instant changes of the selected color, the user select the desired color then apply the change to the test panel by clicking one of the 5 buttons.

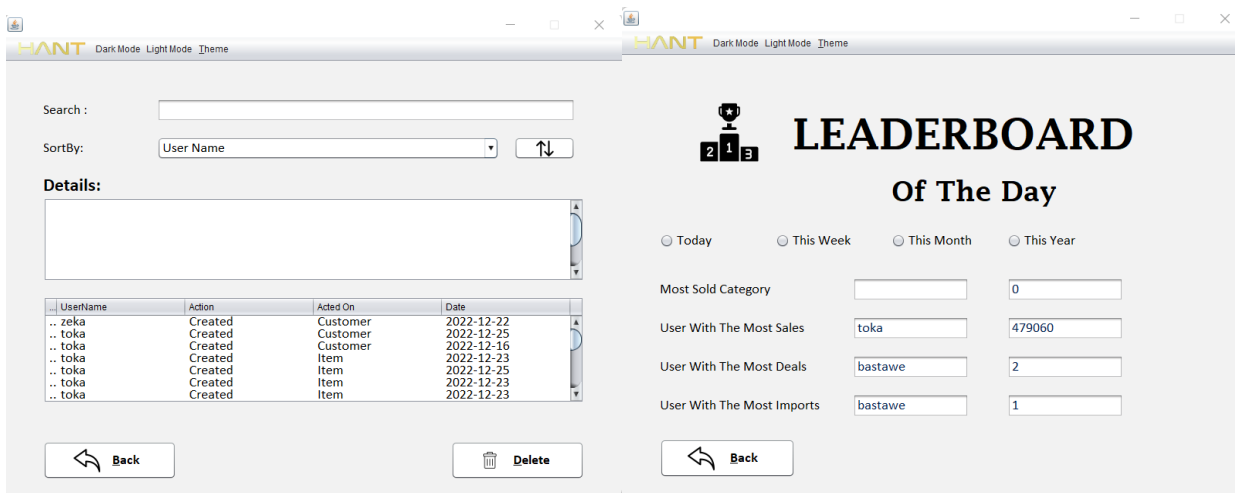




The save button applies the selected colors to the whole application.







## • Accessibility:

We added a lot of non-functional features to facilitate the user experience to make the application easier to be used, by providing different methods to access the system functionality like:

- Clicking Buttons and Submitting Data using the Enter Key:

The enter key is used to perform the click operation on any focused button in the application

It is also used to submit data in forms like New Item, Edit User or New Customer without the need of using the mouse which is easier to the users.

- Mnemonics: we added a **hotkey** to each button in every panel to give the user the ability to access it instead of using the mouse. i.e., The Mnemonics is a keyboard shortcut **Alt+Key**



Each button in every panel is given a specific key different from other buttons in the same panel

These two features gave users the opportunity to neglect the mouse role and use the keyboard in all system operations.

- Highlighting the first option in each panel so the user doesn't have to use the mouse to switch between text field instead he can use the **tab** button.

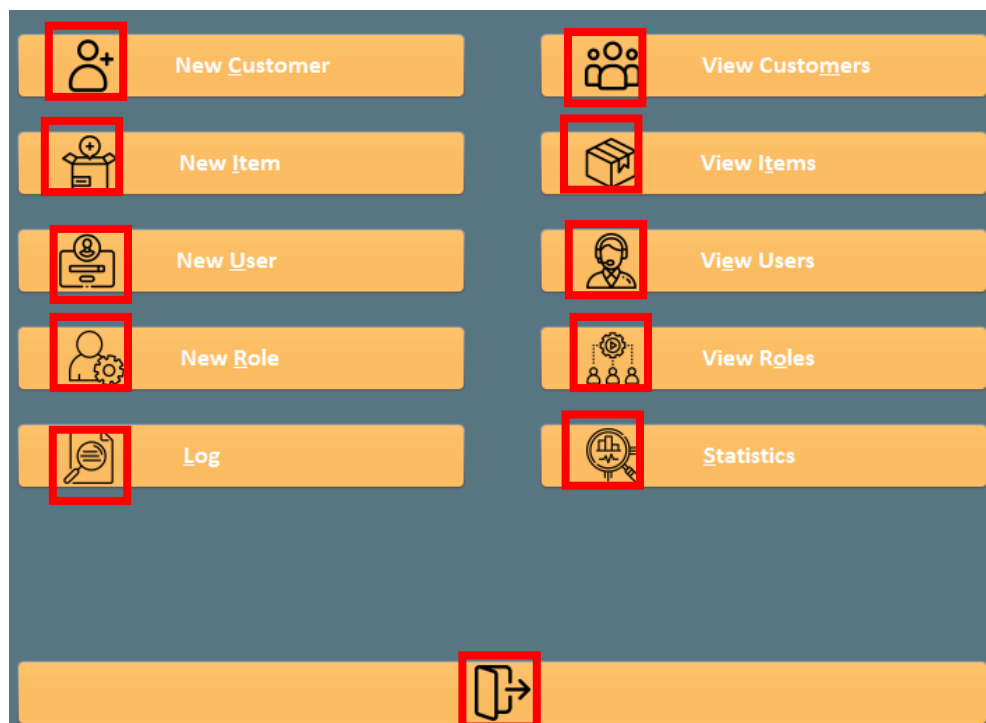
The image shows two identical form panels side-by-side. Each panel has four input fields: a top field, a 'Description:' field, a 'Price:' field, and a 'Category:' field. In the left panel, a green arrow points to the top input field. In the right panel, a red arrow points to the top input field.

- Resetting the panels to default state when the user either commits the action or clicks the back button.

- **Add-Ons:**

- Icons:

We added an icon to each button in every panel to facilitate the user decision making.



## ○ Components Sizes:

Every Component is set to fixed size to make the application more comfortable to look at.

- Every panel and the main Frame are set to 800 X 600 px
- Almost every Button is set to 175 X 50 px
- Almost every Text Field & Combo box are set to 250 X 30 px
- Almost every Label, radio button and Check box is set to 150 X 30 px
- Every Table's width is set to 700 px
- Every Panel has a Top, Right, Bottom and Left margin set to 50 px
- Every Component has Around Margin set to minimum 20 px

## ❖UML Diagram

