

Section 1: Metadata

to be filled by the student

1.1. Project Information to be filled by the student

Title: Safatique Jewellery Store Management System	
Section: T-6	Instructor: Ms. Maria Samad

1.2. Student(s) Information

Name: Hania Kashif	ID: hk08454
Section: T-6	Batch: 2026

Name: Sajal Fatima	ID: sf08108
Section: T-6	Batch: 2026

Name: Shaaf Farooque	ID: sf08405
Section: T-6	Batch: 2026

Submission guideline: Save your project proposal as a pdf file and rename as Project Proposal_L1_ProposedTitle where L1 is to be replaced with your section

Section 2: The Project

to be filled by the student

2.1. Project Description: *Please provide a brief introduction of the project including its scope.*

For our Database Systems course project, we are developing a comprehensive database management system for Safatique, an online jewelry store. This project aims to create an efficient and user-friendly platform that streamlines both customer interactions and administrative operations. Safatique is a small business owned by students who are also working on this project, which ensures that our final product is centered around our primary stakeholders.

Project Scope:

- **Customer Interface:** This involves creating a user-friendly online platform where customers can browse, select, and purchase jewelry items. In real life, this would be the website or app that customers interact with.
- **Inventory Management:** The system will track jewelry stock and raw materials, allowing for adding, updating, and removing products. This reflects the real-world need for efficient stock control in the jewelry business.
- **Sales and Analytics:** The project includes real-time sales tracking and generation of analytical reports on sales trends and popular products. This mirrors the data-driven decision-making processes in modern retail businesses.
- **Order Management:** The system will track customer orders from placement to delivery, maintaining detailed customer profiles and purchase histories. This represents the end-to-end order fulfillment process in a real e-commerce operation.
- **Admin Tools:** Store administrators will have access to tools for managing inventory, analyzing operations, and making data-driven decisions. This reflects the back-end management needs of an actual online store.

This project will integrate database design principles, SQL querying, and user interface development to create a fully functional e-commerce system tailored for the jewelry industry.

2.2 Functional Requirements

This section describes each function/feature provided by your system. These functions are logically grouped into modules based on their purposes. The users in your system must be categorized such as administrator, customer or administrator etc. These users will be accessing the database with the level of access that they are authorized with.

Module 1: Registrations

- **Function 1:** Register an account

The system lets users register an account on the website as an administrator and as a customer.

- **Function 1a:** Register as another administrator. The register form prompts the user to enter their details i.e. Name, Email, Password. The form is submitted and an unverified administrator account is created. The user receives a link on their email address which completes account verification.

- **Function 1b:** Register as a customer. The register form prompts the user to enter their details i.e. Name, Email, Password. The form is submitted and an unverified customer account is created. The user receives a link on their email address which completes account verification.

Module 2: Buying and Selling

- **Function 1:** Only administrator can put item(s) to sell. A registered administrator can upload an item to sell. The system prompts the administrator for item photos, item category and item details and the item are added to the item inventory.

- **Function 2:** Add item to cart. A certain button on the item page prompts the system to add the item to the user's cart. All consequent item(s) added without checkout, are added to the same cart unless cleared.

- **Function 3:** Clear item from cart. The system allows the user to remove a previously added item from the cart. If the cart is empty, the checkout link is no longer accessible.

Module 3: Order Tracking

The user (buyer and seller both) can track orders by inputting the Order number (Order ID) and view the status i.e. Processing, Cleared, Out for Delivery, Delivered.

Module 4: Logistics

The seller can use this Function to view the logistical details of their orders. They can sign in using their account details and then view all orders made from their shop/account. They can view details such as area of delivery, truck number, estimated delivery time, etc. The user can also print out the pdf of their order summary.

Module 6: User Database

This function can only be accessed by the application's moderators to keep track of all users (buyers and sellers). It will fetch all details of any selected user and can moderate their activities, for example verifying or deleting accounts.

Module 7: Complaints/Feedback

For both sellers and customers, this function can be used to give any feedback on the application, it will ask for a rating and comments where users can also report certain sellers or customers. This will store the complaint in the database, which can be accessed by employees, and they can then email the complainant with a potential fix to their problem for e.g. Refund vouchers etc.

Module 8: Financial Management

This would help you manage sales data, expenses, profit margins, and generate financial reports.

Module 9: Inventory Management

This module would help you keep track of your jewelry stock, including details like item descriptions, quantities, materials used, and reorder points.

Module 10: Supplier Management

Use this to manage information about your suppliers, including contact details, materials they provide, and order history. This includes courier and raw materials.

Module 10: Customer Relationship Management (CRM)

This module would store customer information, purchase history, preferences, and allow you to manage loyalty programs or special offers.

Access Levels:

Administrators: These users have the highest level of access. They can perform functions such as:

- Registering other administrators
- Uploading items for sale
- Accessing and moderating the user database
- Managing financial data and generating reports

Customers: These are the end-users of the online store. Their functions include:

- Registering as customers
- Adding items to cart and clearing items from cart
- Tracking their orders
- Providing feedback or complaints

Moderators: These users have special access to:

- User database for account verification and moderation
- Handling customer complaints and feedback

2.3. Planned Schedule: *Kindly list the start/end dates and the timeline for the achievement of any intermediate milestones and the expected contribution to be made by the participant(s).*



Week	Task	Contributions
4	Proposal Submission	Hania: Draft, Sajal: Review, Shaaf: Finalize
4-5	Improving screens	Hania: UI design, Sajal: Functionality, Shaaf: User testing
5-6	Working on ERD	Hania: Entity identification, Sajal: Relationship mapping, Shaaf: Attribute definition
6	Interim ERD deadline	All: Collaborative review and submission
7-8	Connecting and refining screens	Hania: Frontend integration, Sajal: Backend connectivity, Shaaf: Error handling
8-9	Implementing SQL	Hania: Query writing, Sajal: Database setup, Shaaf: Testing queries
9	Getting dummy data	Hania: Data generation, Sajal: Data validation, Shaaf: Data import
9-12	Backend development and database functionality	Hania: API development, Sajal: Database optimization, Shaaf: Security implementation
13	Interim Demo	All: Presentation preparation and delivery
14	Finalizing ERD, SQL, DB Schema	Hania: ERD refinement, Sajal: SQL optimization, Shaaf: Schema validation
15	Finalizing front and back end	Hania: Frontend polishing, Sajal: Backend optimization, Shaaf: Integration testing
16	Final Presentation	All: Presentation preparation, demo, and documentation

2.4. Technology Stack: *If you are utilizing any language or database besides PyQt and SQL Server, please complete this section; otherwise, leave it blank. Specify the programming language and database management system intended for constructing this application and the application type (Desktop, Web, or Mobile).*

Software:

1. Db Designer
2. Qt Designer
3. MySQL
4. Visual Studio Code
5. GitHub

Programming Languages:

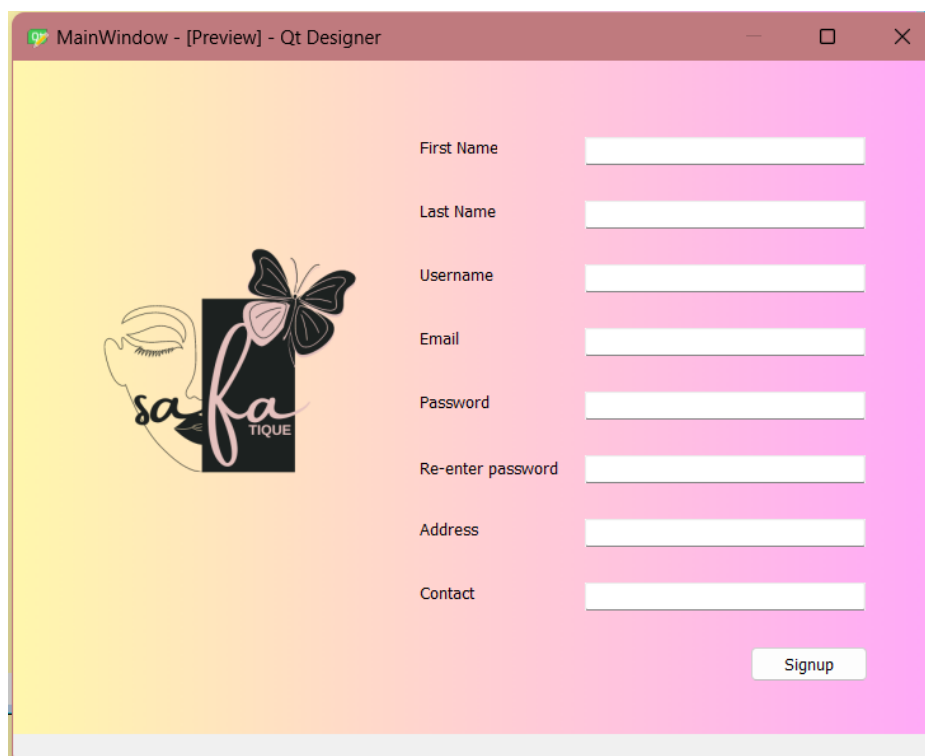
1. Python (PyQt)
2. SQL

2.5. Screens: *Provide images of all application screens, showcasing clear input and corresponding outputs. Ensure each image includes a concise caption explaining user action and expected/observed output. You can create these screens using Qt Designer.*



The image shows a Qt Designer window titled 'MainWindow - [Preview] - Qt Designer'. The design is a login screen with a yellow-to-pink gradient background. On the left is a logo for 'sa fa TIQUE' featuring a stylized face and a butterfly. To the right of the logo are two text input fields labeled 'Username' and 'Password'. Below the 'Password' field is a 'login' button. At the bottom left, there is a link 'Don't have an account?' followed by a 'signup' button.

Fig 1: The sign-in screen where the user can log in and if not, there is a button to redirect the user to the sign-up page where they can register a new account. The details are verified from the Customer Database



The image shows a Qt Designer window titled 'MainWindow - [Preview] - Qt Designer'. The design is a registration screen with a yellow-to-pink gradient background. On the left is the same 'sa fa TIQUE' logo as in Fig 1. To the right of the logo are seven text input fields labeled 'First Name', 'Last Name', 'Username', 'Email', 'Password', 'Re-enter password', and 'Address'. Below the 'Address' field is a 'Contact' field. At the bottom right is a 'Signup' button.

Fig 2: This is the registration page where the user can sign-up and create an account with the relevant details. The form will store this in the Customer Database.

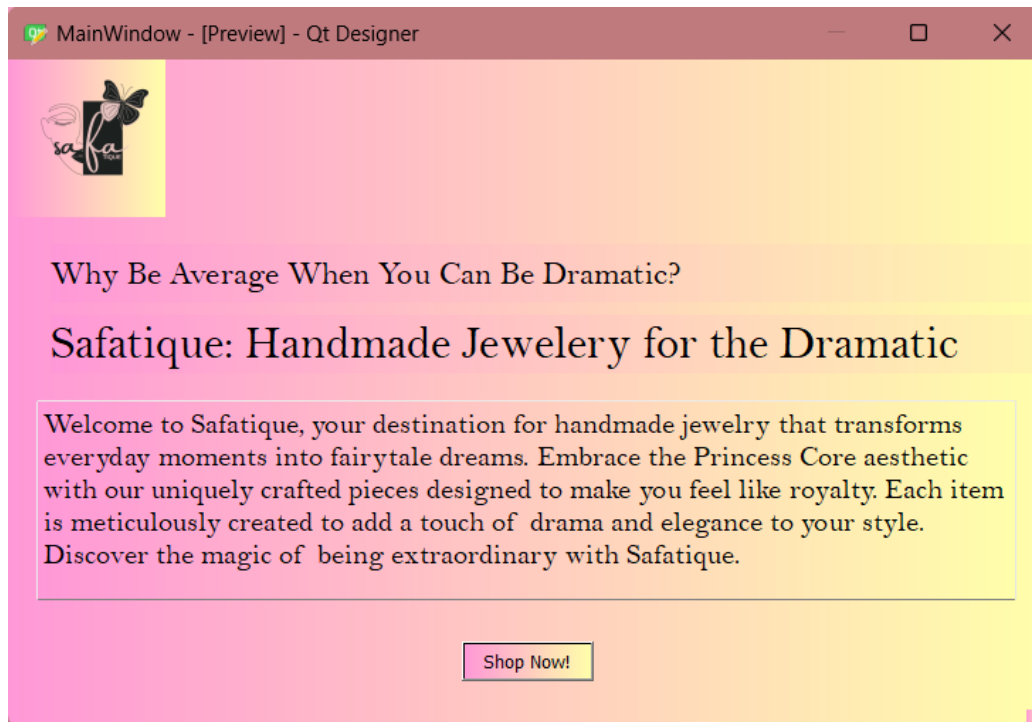


Fig 3: This is a homepage screen that will be visible after login or registration. The shop now button will open the catalogue.

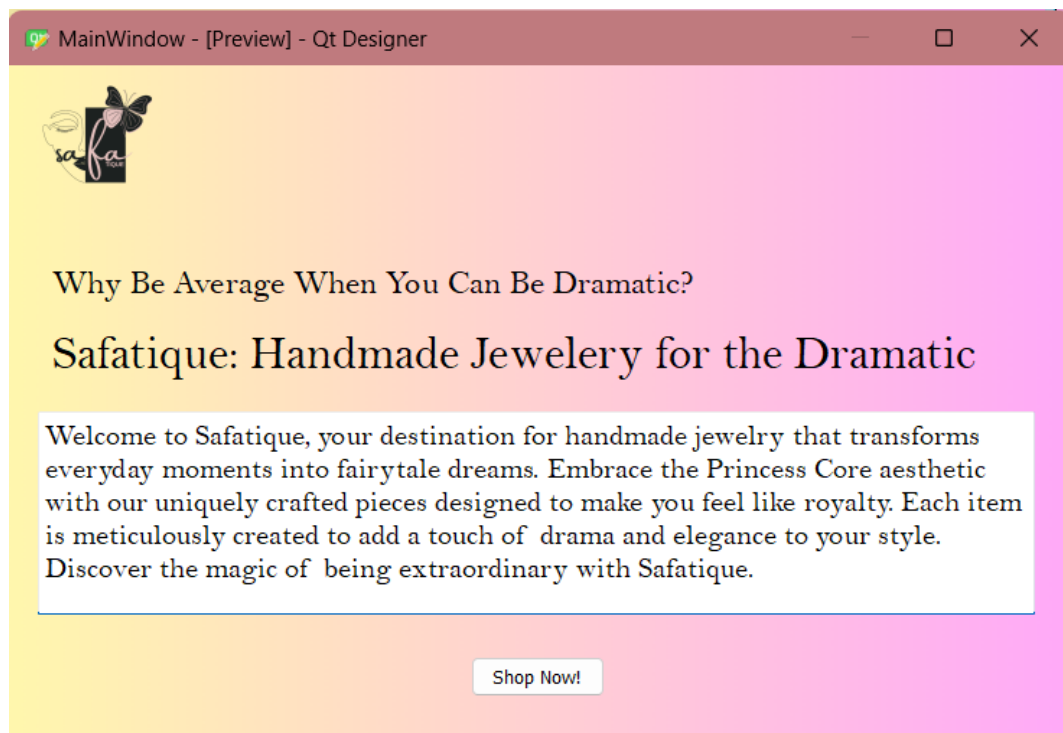


Fig 3.1: This is another version of the screen above

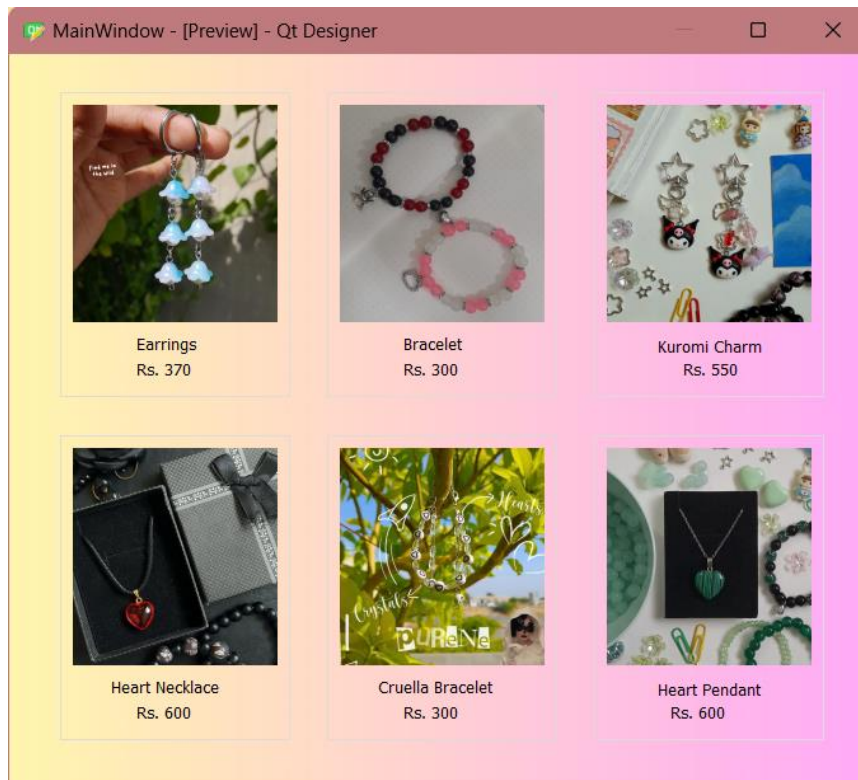


Fig 4: User will be able to click on the image to add to cart or view them in a detailed format in another pop-up screen.

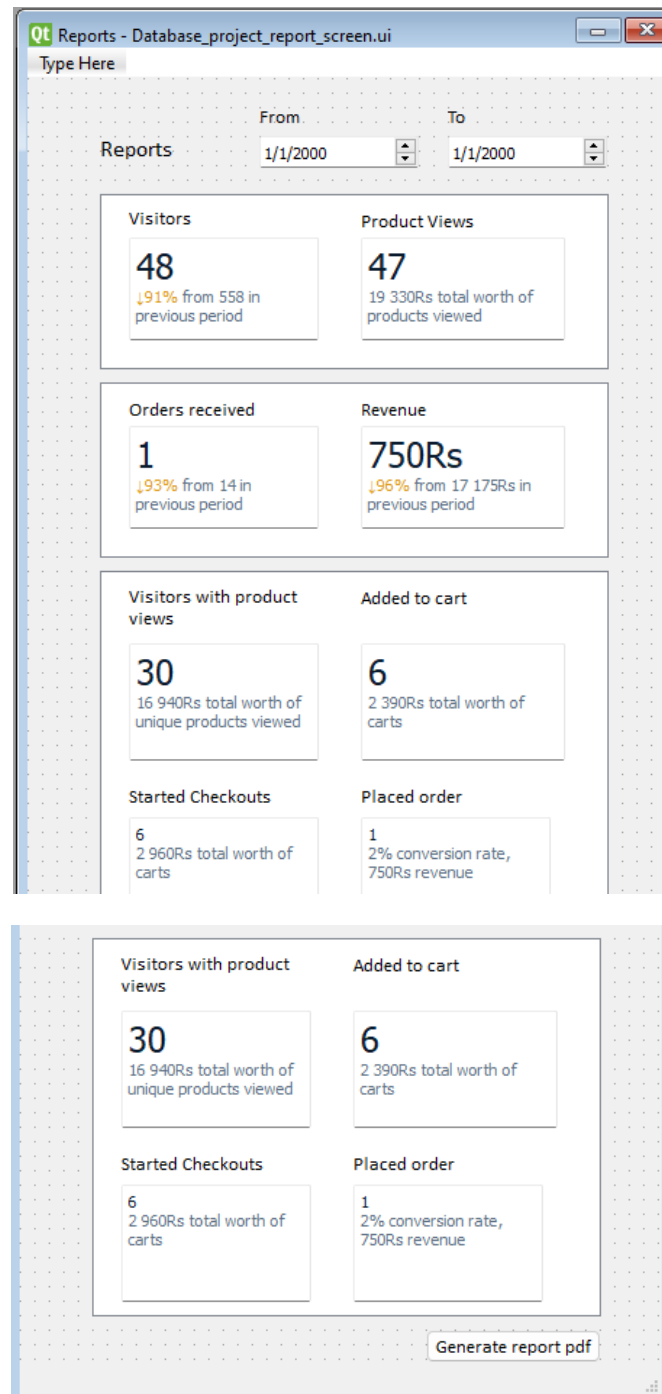


Fig 5: The Reports Dashboard allows users to generate customized reports by selecting specific date ranges and report types. The resulting display provides a comprehensive overview of store performance, including visitor data, product engagement, order information, and revenue figures. Users can easily compare current data with previous periods and download the report in PDF format for further analysis or presentation.

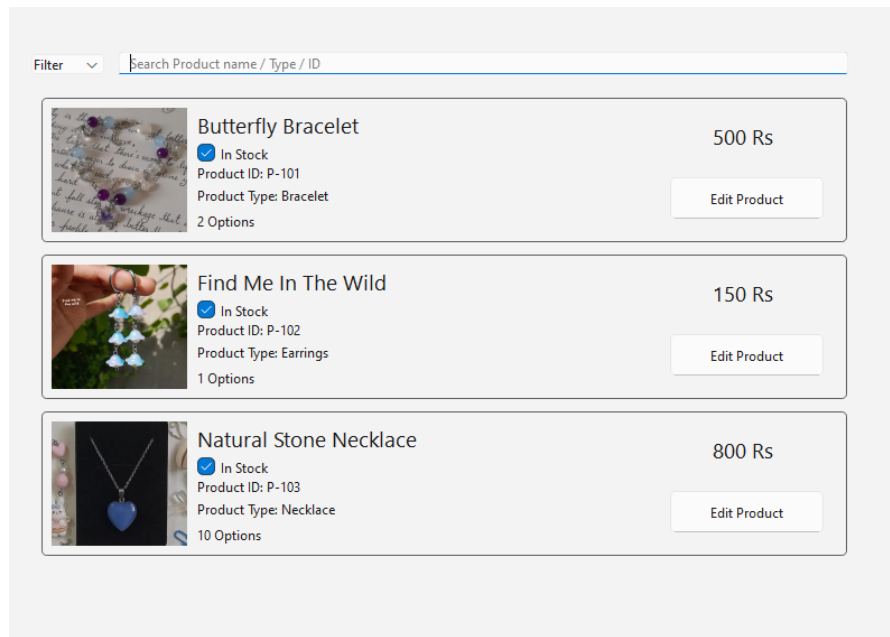
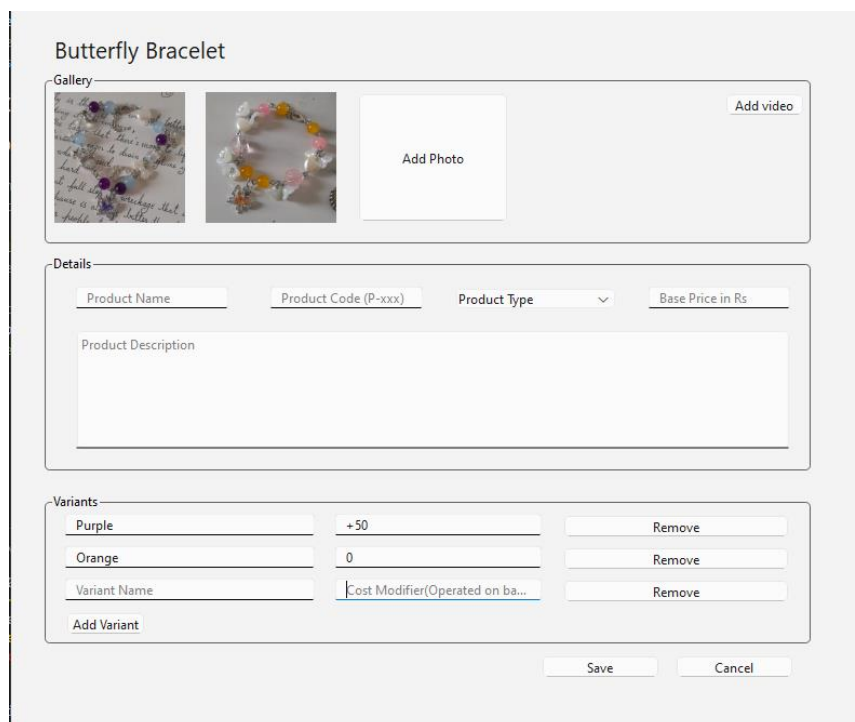


Fig 6: The catalog screen allows the user to look at and search for products, to look at details or to update if they are in stock, or to edit a product's details. There are filters to aid in search (Tentative) and a search bar. Once the edit product button is pressed user is taken to the add/edit product screen.

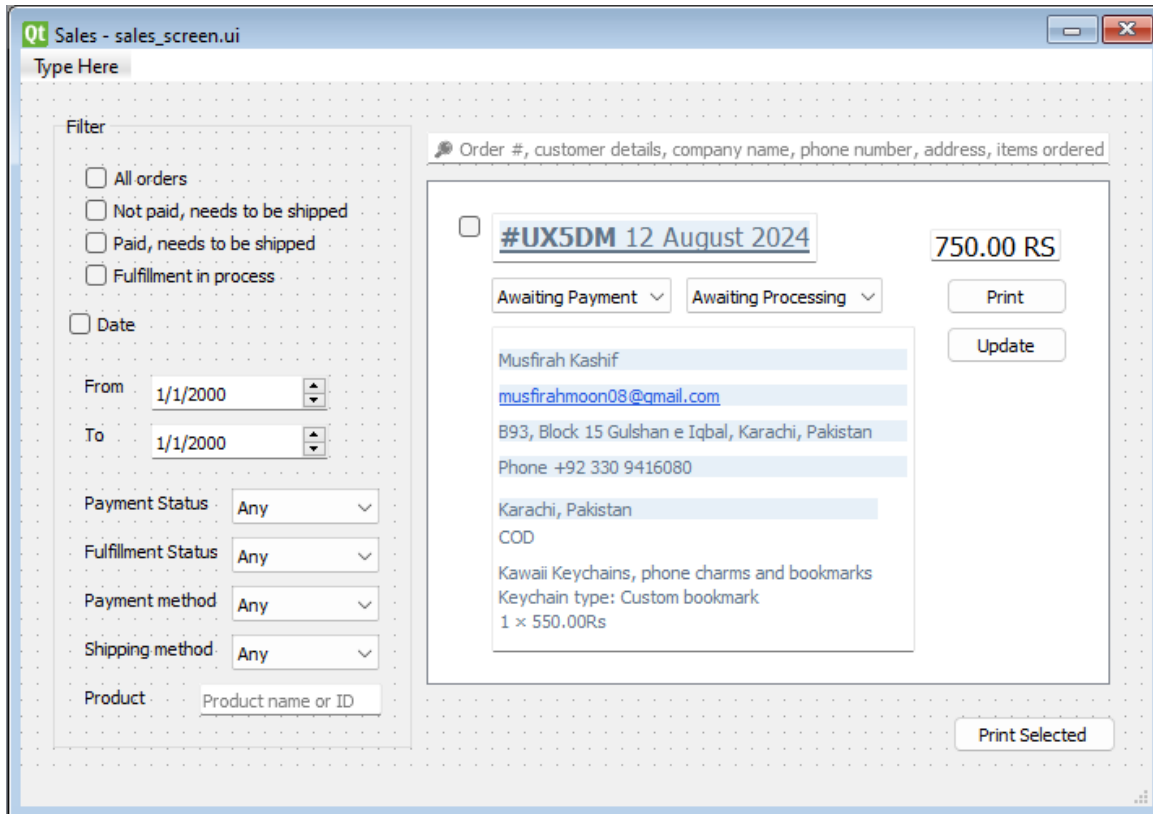


The screenshot shows the 'Butterfly Bracelet' product add/edit screen. It is divided into three main sections:

- Gallery**: Contains two photo thumbnails, an 'Add Photo' button, and an 'Add video' button.
- Details**: Contains input fields for 'Product Name', 'Product Code (P-xxx)', 'Product Type' (dropdown), and 'Base Price in Rs'. Below these is a large text area for 'Product Description'.
- Variants**: Contains a table for managing variants. The table has columns for 'Variant Name', 'Cost Modifier(Operated on ba...', and 'Remove'. Two variants are listed: 'Purple' with a cost modifier of '+50' and 'Orange' with a cost modifier of '0'. An 'Add Variant' button is at the bottom left of the table.

At the bottom of the screen, there are 'Save' and 'Cancel' buttons.

Fig 7: The product add/edit screen allows the user to add or edit the details of the product. The user can set the name, price, add variants, add photos and videos of the product. Once the details are set, the save button is pressed and the product details are processed and added to the catalog.



The screenshot shows a Qt application window titled "Sales - sales_screen.ui". The interface is divided into two main sections. On the left is a "Filter" panel with various checkboxes and dropdown menus. On the right is a detailed view of a selected order, identified by the order number "#UX5DM 12 August 2024" and a total value of "750.00 RS".

Filter Panel:

- ☐ All orders
- ☐ Not paid, needs to be shipped
- ☐ Paid, needs to be shipped
- ☐ Fulfillment in process
- ☐ Date
 - From: 1/1/2000
 - To: 1/1/2000
- Payment Status: Any
- Fulfillment Status: Any
- Payment method: Any
- Shipping method: Any
- Product: Product name or ID

Order Details Panel:

- Order #: #UX5DM 12 August 2024
- Total: 750.00 RS
- Status: Awaiting Payment (dropdown), Awaiting Processing (dropdown)
- Buttons: Print, Update
- Customer Name: Musfirah Kashif
- Email: musfirahmoon08@gmail.com
- Address: B93, Block 15 Gulshan e Iqbal, Karachi, Pakistan
- Phone: +92 330 9416080
- Location: Karachi, Pakistan
- Payment Method: COD
- Items: Kawaii Keychains, phone charms and bookmarks
- Keychain type: Custom bookmark
- Quantity/Price: 1 x 550.00Rs
- Button: Print Selected

Fig 8: The Sales Screen enables users to efficiently manage and review order information. Users can filter orders based on multiple criteria, providing a customized view of the sales data. Upon selecting a specific order, detailed information is presented, including customer data, purchased items, and payment details. This screen also offers functionality to update order statuses and generate invoices, streamlining the sales management process.