

Documentation For Java POS(Point of Sales) Project

Design and Developed By
Dipanjan Ghosh

Objective: The objective of building a point of sale (POS) system is to provide a comprehensive solution that allows businesses to manage their sales transactions, inventory, and customer data in a streamlined and efficient way. The system should be designed to simplify the checkout process, reduce errors, and provide real-time data to business owners so that they can make informed decisions about their operations.

- Simplifying the checkout process: The system should be designed to make it easy for cashiers to ring up sales quickly and accurately. This can be achieved through features such as barcode scanning, touch screen interfaces, and automatic pricing calculations.
- Tracking inventory levels: A good POS system should allow businesses to track their inventory levels in real-time. This can help them avoid stockouts and ensure that they always have the products their customers want.
- Managing customer data: The system should be able to store and manage customer data, such as contact information, purchase history, and loyalty program details. This can help businesses to build stronger relationships with their customers and provide a more personalised shopping experience.
- Generating reports: The system should be able to generate reports that provide insights into sales trends, inventory levels, and customer behaviour. This information can help business owners make informed decisions about their operations and identify areas for improvement.
- Integrating with other systems: The system should be designed to integrate with other business systems, such as accounting software, eCommerce platforms, and marketing tools. This can help businesses streamline their operations and reduce manual data entry.

Overall, the objective of building a POS system is to provide businesses with a powerful tool for managing their sales, inventory, and customer data in a way that is efficient, accurate, and scalable.

Technology User in the project:

For Gui :

- Java Swing , JFrame

For Backend:

- Core java
- JDBC connection

For Database:

- Mysql

IDE used:

- Apache NetBeans 16

Demo Of My Design:

Gui:

Item	Quantity	Amount

Sub Total:

Tax:

Total:

Payment Method:

Cash:

Change:

Buttons: Pay, Reset, Print, Remove, EXIT

Functionalities:

Item	Quantity	Amount
Cup Cake	1	4.99
Orange Juice	1	19.99
Ice Tea	1	19.99
Black Coffee	1	25.99

The barcode will change on change the total

By clicking this button the item will add on the table

Sub Total: INR 70.96

Tax: INR 2.77

Total: INR 73.73

Payment Method:

Cash:

Change:

Buttons: Pay, Reset, Print, Remove, EXIT

In the previous image we are adding the product which the user want to order.

The subtotal will add the price of the products which the user wants to order , the tax will be calculated internally in the backend and the total is calculated by adding the subtotal and tax.

The interface includes a numeric keypad on the left with buttons for digits 0-9, a decimal point, and a 'C' (clear) button. In the center, there is a table for the current order items:

Item	Quantity	Amount
Cup Cake	1	4.99
Orange Juice	1	19.99
Ice Tea	1	19.99
Black Coffee	1	25.99

Below the table is a barcode. To the right of the table is a grid of 18 food and drink images. At the bottom right of the grid, there are two red text annotations: "after we click the pay button the payment is done and data is stored on database" and "reset button is reset the page , remove button can remove item from the table which the user want to remove".

The bottom section contains a summary of the order:

Sub Total	INR 70.96	Payment Method	Cash
Tax	INR 2.77	Cash	75
Total	INR 73.73	Change	INR 1.27

On the right side of the bottom section are buttons for 'Pay', 'Reset', 'Print', 'Remove', and an 'EXIT' button.

In the previous image we choose the payment method. Here the user wants to pay with the cash and he/she pays rs 75 and the change will be calculated by subtracting cash paid from the total. After clicking the pay btn the data will be stored in the database as order and order_items. After paying we can remove the item using the remove button. That case we have to select the item from the item table and click the remove button. Reset button will reset the whole window and we can add an item

The interface is identical to the previous screenshot, but the 'Payment Method' dropdown menu is now set to 'Cash'. The 'Cash' input field shows the value '75'. The 'Pay' button is highlighted with a red border. The 'Print' button is also highlighted with a red border. The 'EXIT' button is visible at the bottom right.

At the bottom right of the product grid, there are two red text annotations: "print is used to print the data of the product, total price, tax" and "Exit will exit from the window."

After payment is done we can print the bill using the print button. Exit button will exit from the window giving an alert message "Do you want to exit? yes/no." If we click the yes button then only we can exit from the window.

Use Cases: A point of sale (POS) system is used to process transactions between a customer and a business.

Here are some common places where POS systems are used:

- Retail stores: POS systems are commonly used in retail stores to process transactions, manage inventory, and track sales data.
- Restaurants: POS systems are used in restaurants to take orders, process payments, manage table layouts, and track inventory.
- Hospitality industry: POS systems are used in hotels, resorts, and other hospitality businesses to process room charges, manage reservations, and track inventory.
- Healthcare: POS systems are used in healthcare settings to process payments for medical services, manage patient data, and track inventory.
- Entertainment venues: POS systems are used in entertainment venues, such as movie theatres and concert halls, to sell tickets and process payments for concessions.

Overall, POS systems are used in a wide range of industries to streamline transactions, manage inventory, and collect data that can help businesses make informed decisions.

Conclusion: Implementing a POS system can bring numerous benefits to a business, including increased efficiency, improved accuracy, and enhanced data collection capabilities.

The system can streamline transactions, automate inventory management, and provide valuable insights into sales trends and customer behaviour.

However, it's important to carefully plan and execute a POS system project to ensure its success. This includes selecting the right system for the business needs, properly training employees on how to use the system, and regularly maintaining and updating the system to prevent downtime or data loss.

Overall, a well-executed POS system project can bring significant benefits to a business, but it requires careful planning, implementation, and ongoing maintenance to be successful.