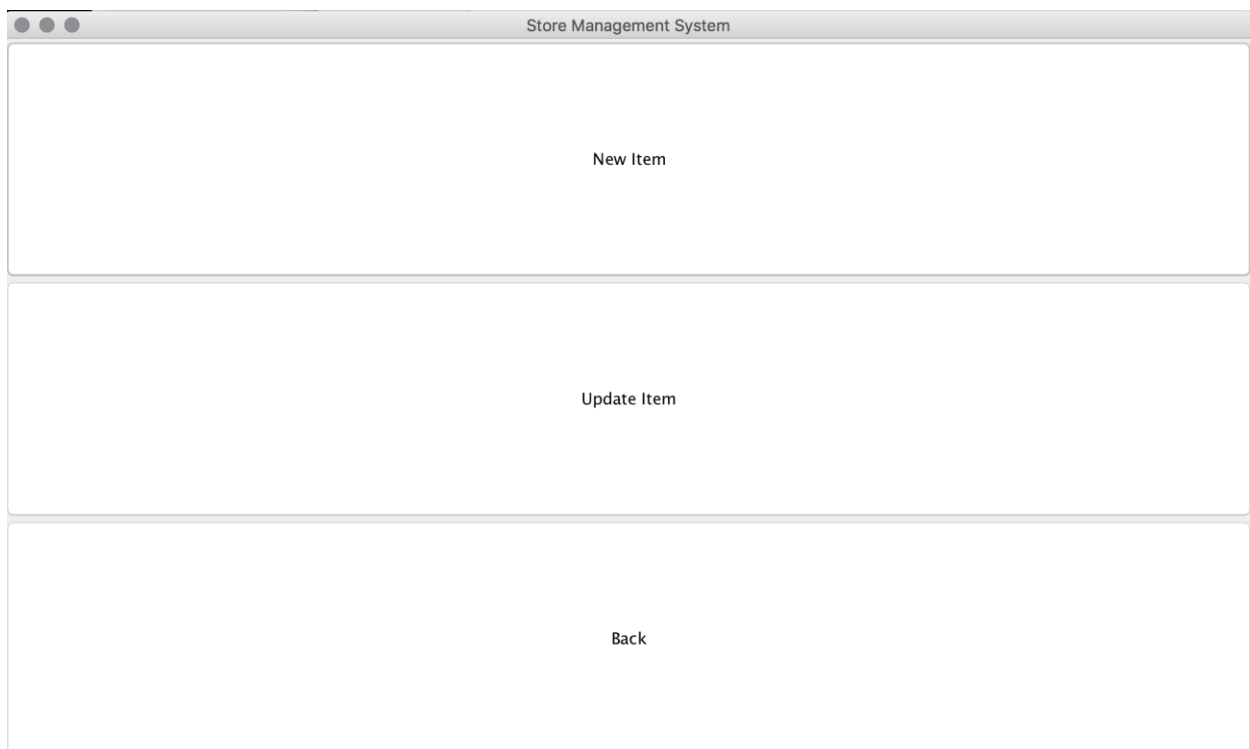
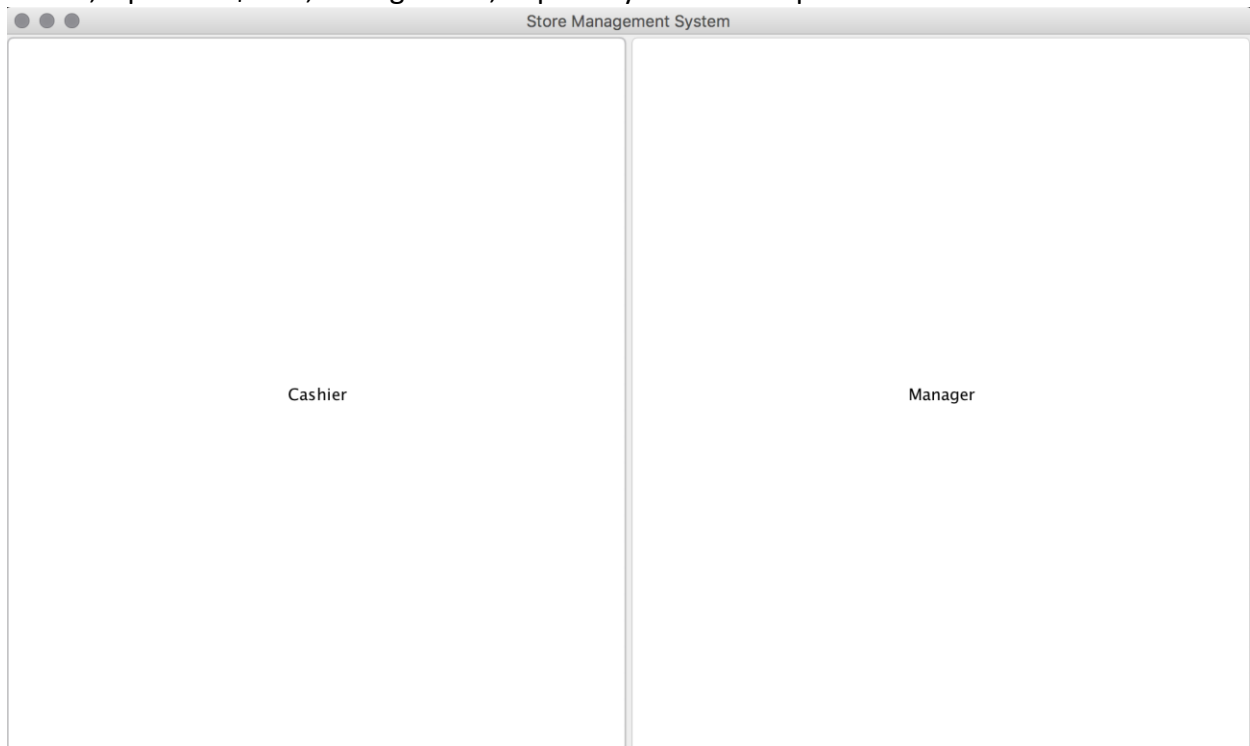



Scenario 1 we add a new item to the store management system. We add bread, with an item ID of 102, a price of \$5.00, a weight of 0, a quantity of 10 and a producer if of 200.



Store Management System

create new Item



ID#

Name

Price

Weight

Quantity

Producer

Cancel


OK

Back

Store Management System

New Item

create new Item



ID#

102

Name

bread

Price

5.00

Weight

0

Quantity

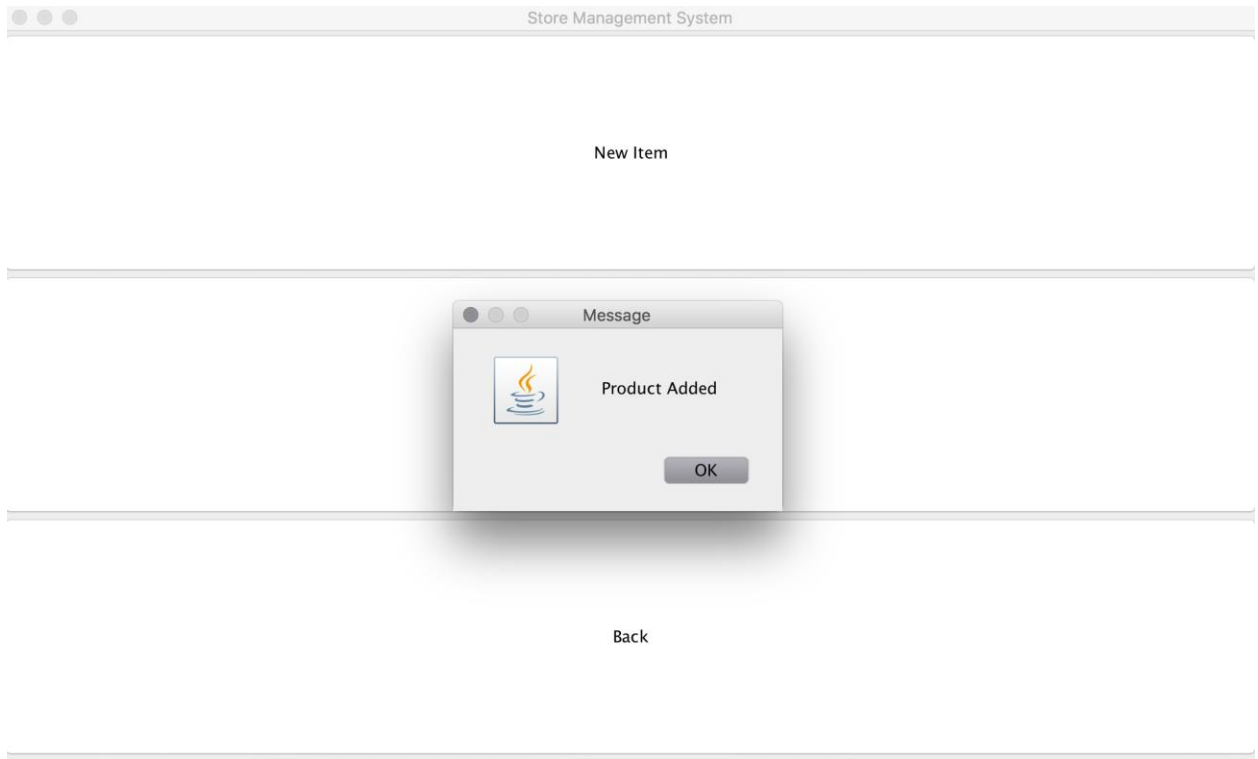
10

Producer

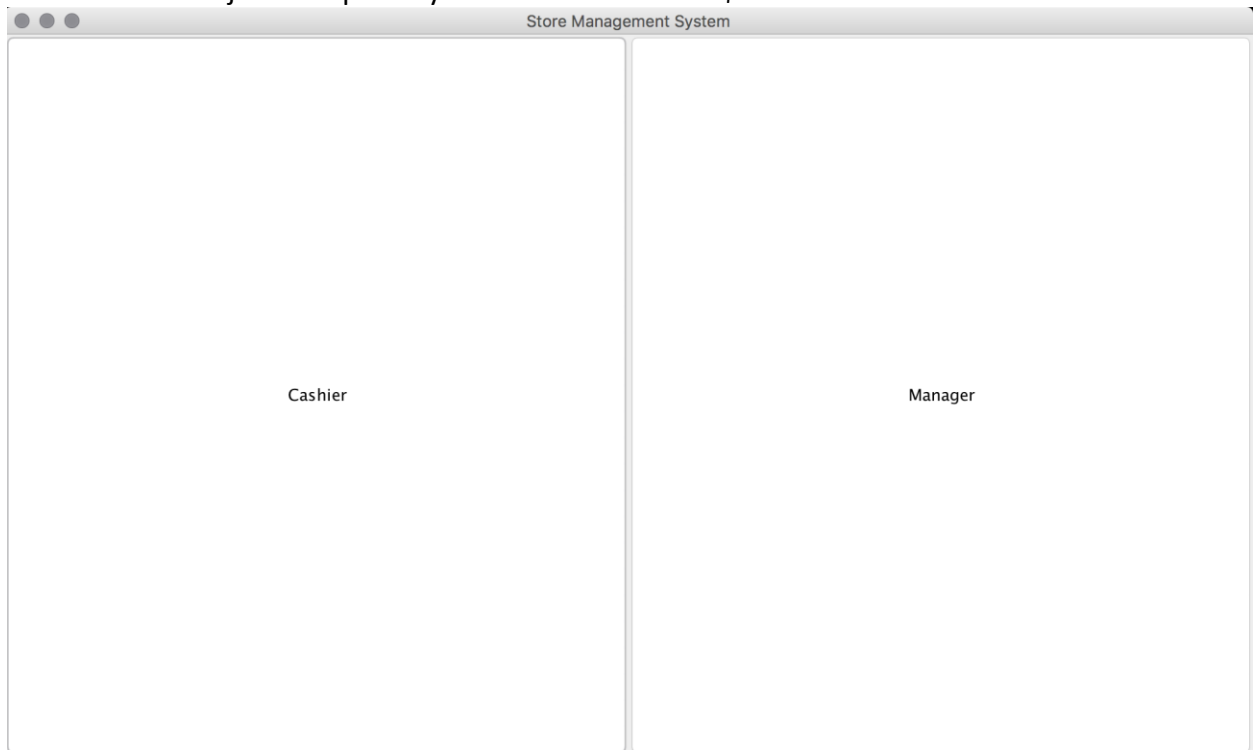
200

Cancel

OK



Scenario 2 we adjust the quantity of the above item to \$6.50.



●●●Store Management System

New Item

Update Item

Back

●●●Store Management System

New Item

Update Item



ID#


CancelOK

Back

Store Management System

New Item

Update Item



ID#

102

Cancel

OK

Back


● ● ●

Store Management System

New Item

● ● ●

Update Item



Name

bread

Price

5.0

Weight

0.0

Quantity

10

Producer

200

Cancel

OK

Back


● ● ●

Store Management System

New Item

● ● ●

Update Item



Name

bread

Price

6.5

Weight

0.0

Quantity

10

Producer

200

Cancel

OK

Back

Scenario 3 we add 2 bread and a coke to the cart then pay cash and finish the transaction.

```
graph TD; subgraph "Store Management System"; Cashier[Cashier]; Manager[Manager]; end; AddItem[Add Item]; Remove[Remove]; ApplyDiscount[Apply Discount]; Override[Override]; Payment[Payment]; Back[Back]; Cashier --> AddItem; Cashier --> Remove; Cashier --> ApplyDiscount; Cashier --> Override; Cashier --> Payment; Cashier --> Back; Manager --> AddItem; Manager --> Remove; Manager --> ApplyDiscount; Manager --> Override; Manager --> Payment; Manager --> Back;
```

The diagram illustrates the functional requirements for a Store Management System, divided into two main user roles: Cashier and Manager.

**Cashier Role:**

- The Cashier can perform all core system functions: Add Item, Remove, Apply Discount, Override, Payment, and Back.

**Manager Role:**

- The Manager can also perform all core system functions: Add Item, Remove, Apply Discount, Override, Payment, and Back.

**System Components:**

- Entities:** Cashier, Manager
- Use Cases:** Add Item, Remove, Apply Discount, Override, Payment, Back

Store Management System


ID#	Name	Quantity	Weight	Price
-----	------	----------	--------	-------

Add Item

Remove

Apply Discount

Add Item



ID#

quantity

Cancel OK

Back

\$0.0

Store Management System

ID#	Name	Quantity	Weight	Price
-----	------	----------	--------	-------


\$0.0

Add Item

Remove

Apply Discount

Add Item



ID#  
102

quantity  
2

Cancel OK

Back



Store Management System

ID#	Name	Quantity	Weight	Price
102	bread	2.0	0.0	6.5

Add Item

Remove

Apply Discount

Override

Payment

Back

\$13.0

Store Management System

ID#	Name	Quantity	Weight	Price
102	bread	2.0	0.0	6.5

Add Item

Remove


Apply Discount

Override

Payment

Back

Add Item



ID#  
101

quantity  
1

CancelOK

Store Management System

ID#	Name	Quantity	Weight	Price
102	bread	2.0	0.0	6.5
101	coke	1.0	0.0	3.0

Cash

Credit/Debit

Gift Card

Finish

Back

\$16.0

Store Management System

Add Item

Remove

Apply Discount

Override

Payment

Back

\$16.0

Store Management System

ID#	Name	Quantity	Weight	Price
102	bread	2.0	0.0	6.5
101	coke	1.0	0.0	3.0

Cash


Credit/Debit

Gift Card

Finish

Back

Cash Payment



Enter Cash Amount

16.

Cancel

OK

\$16.0

Store Management System

ID#	Name	Quantity	Weight	Price
102	bread	2.0	0.0	6.5
101	coke	1.0	0.0	3.0

Cash

Credit/Debit

Gift Card

Finish

Back

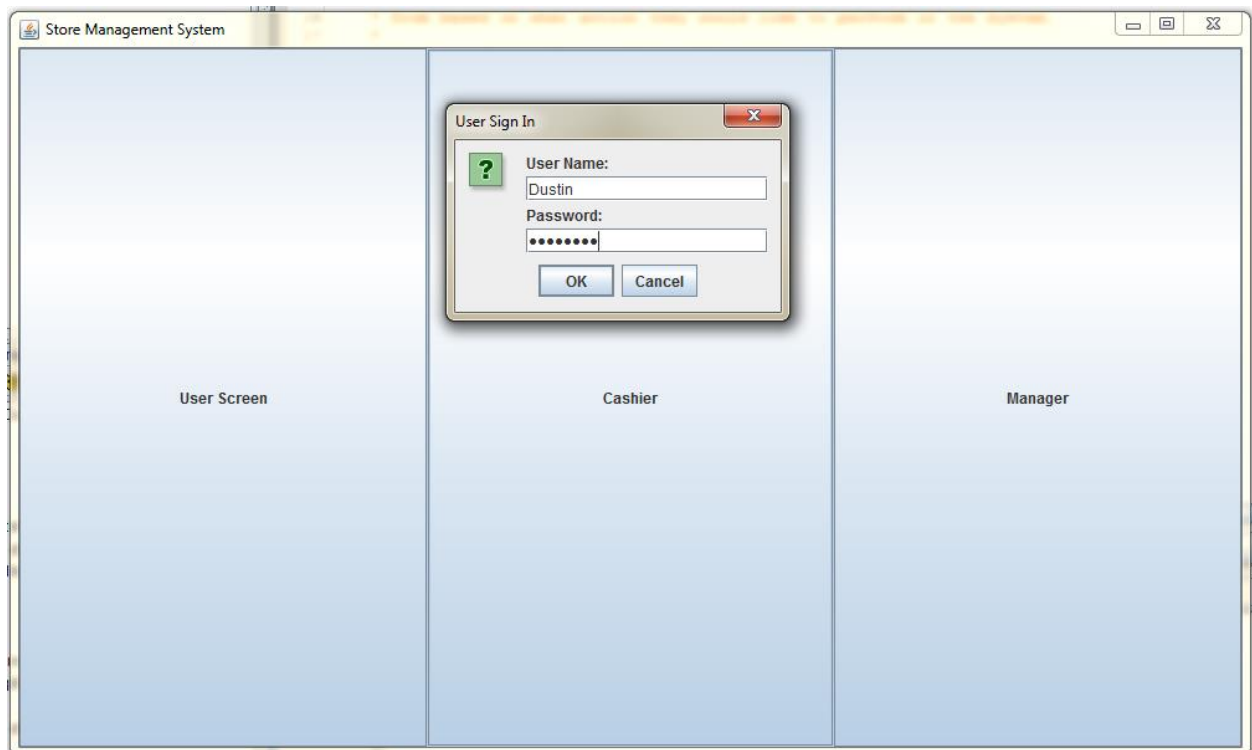
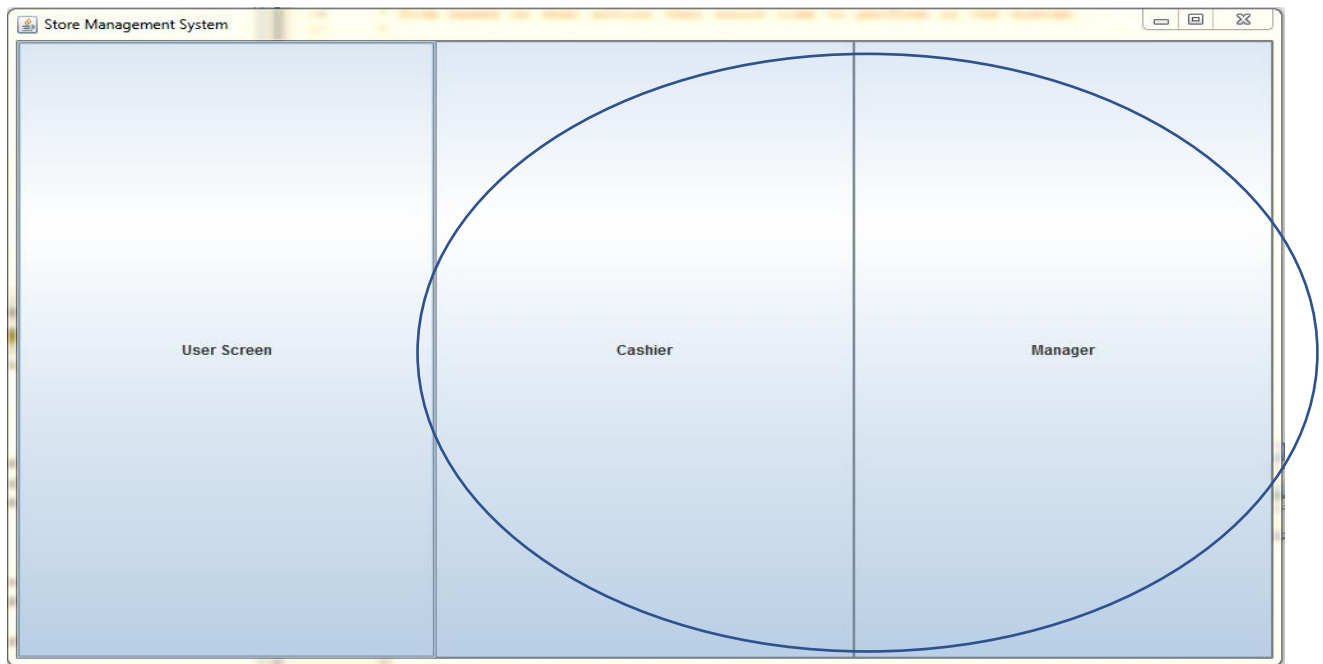
\$0.0

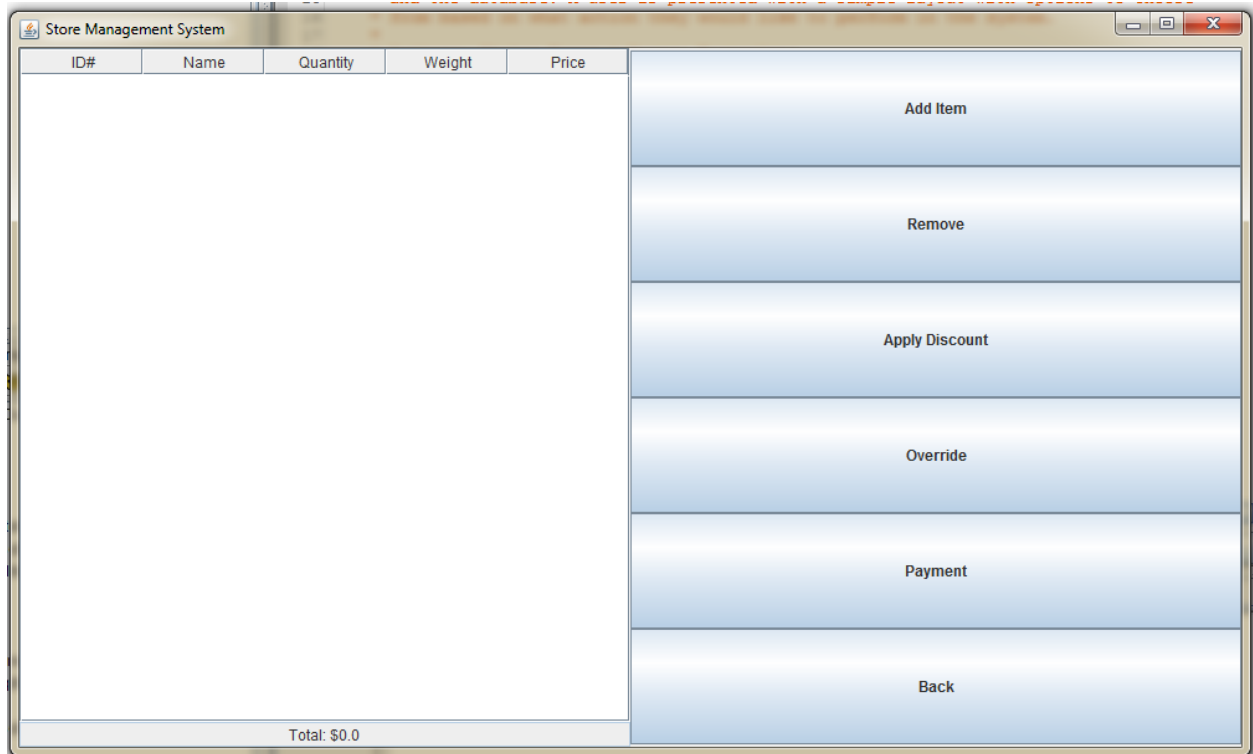
ID#	Name	Quantity	Weight	Price
	Cash			
	Credit/Debit			
	Gift Card			
	Finish			
	Back			

\$0.0

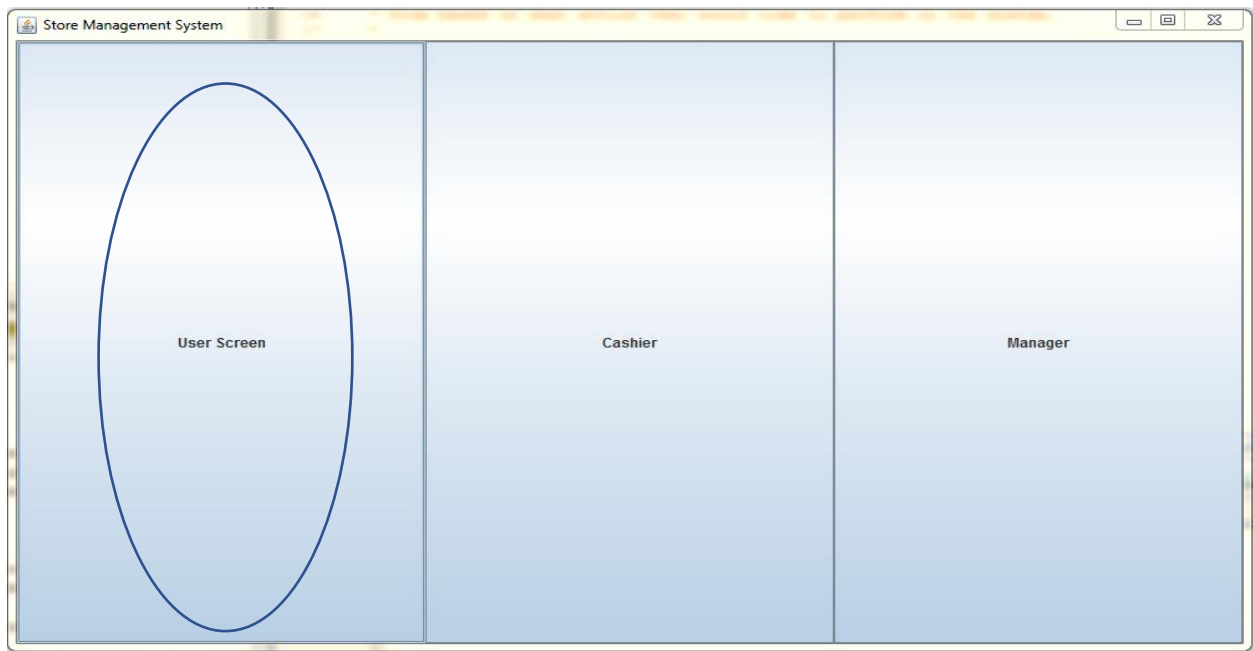
-----Screenshots Demonstrating Iteration 2 Use Cases-----

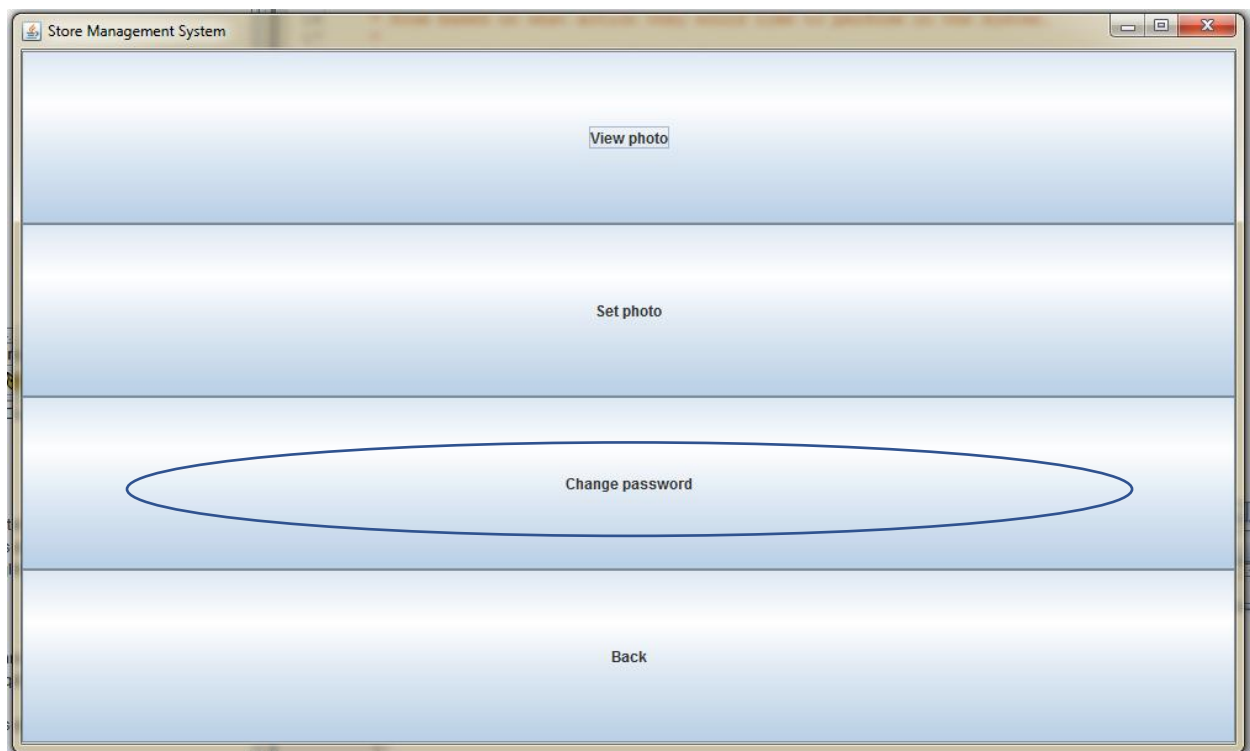
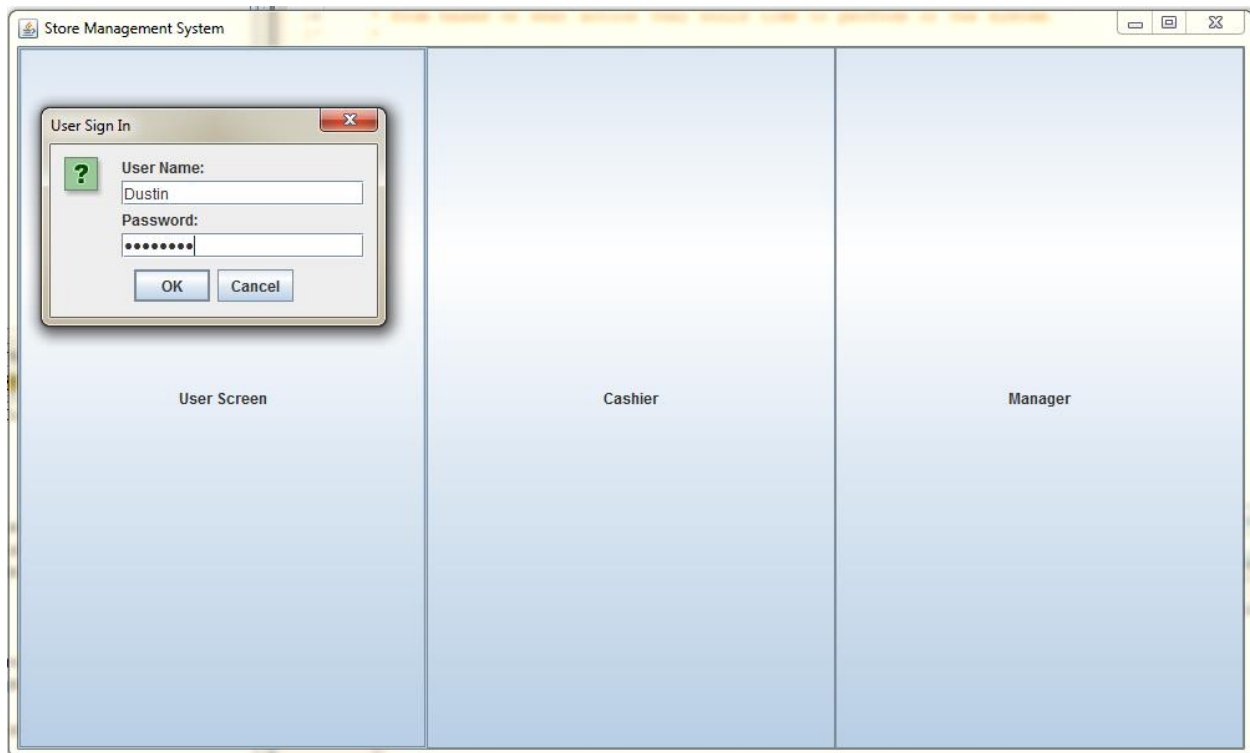
1. A cashier or manager logging in to the system.

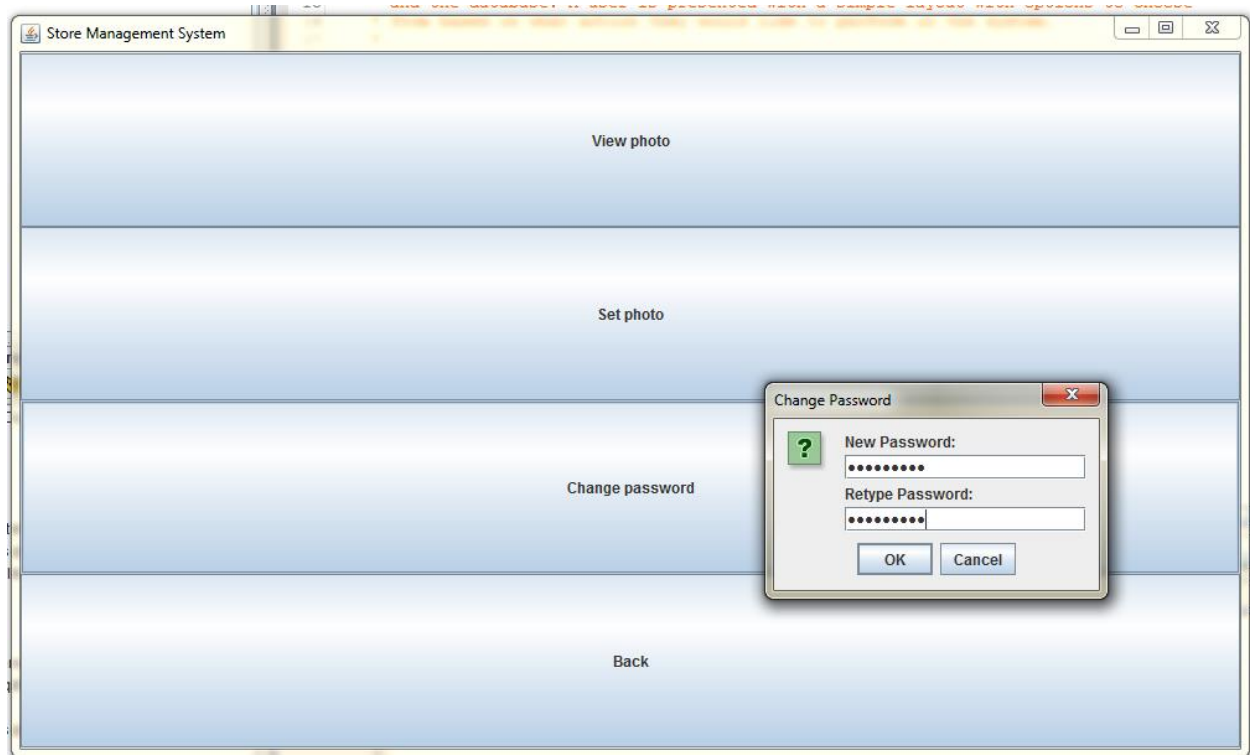




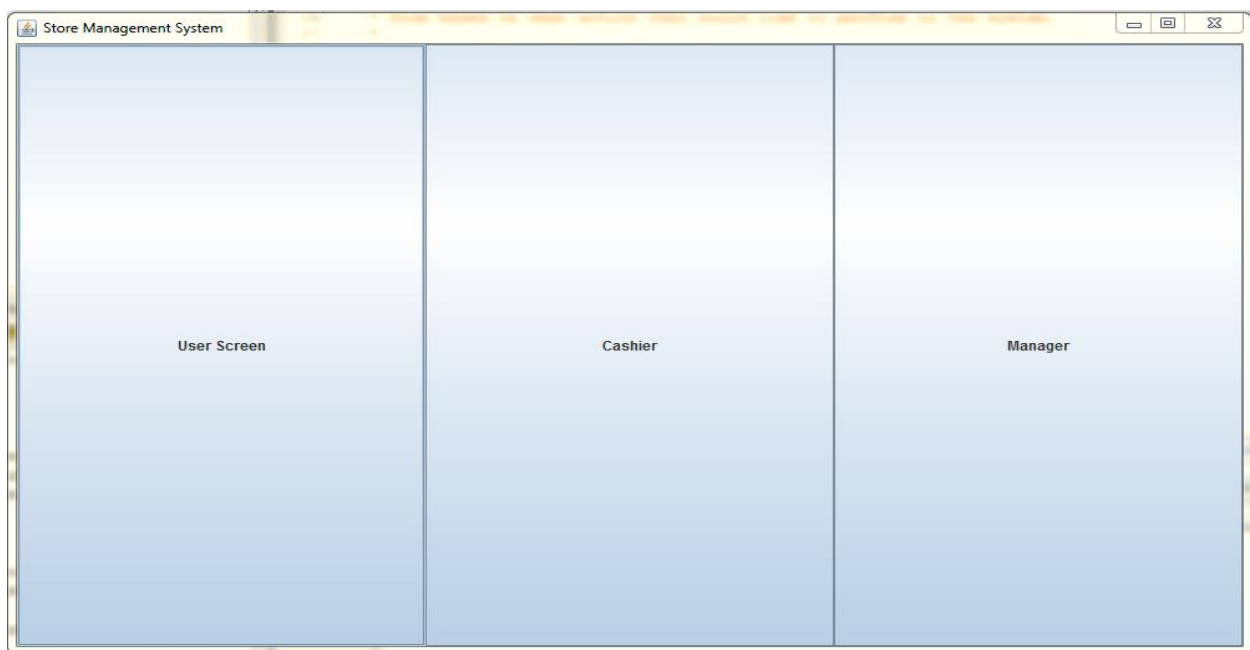
2. A user changes their login password.



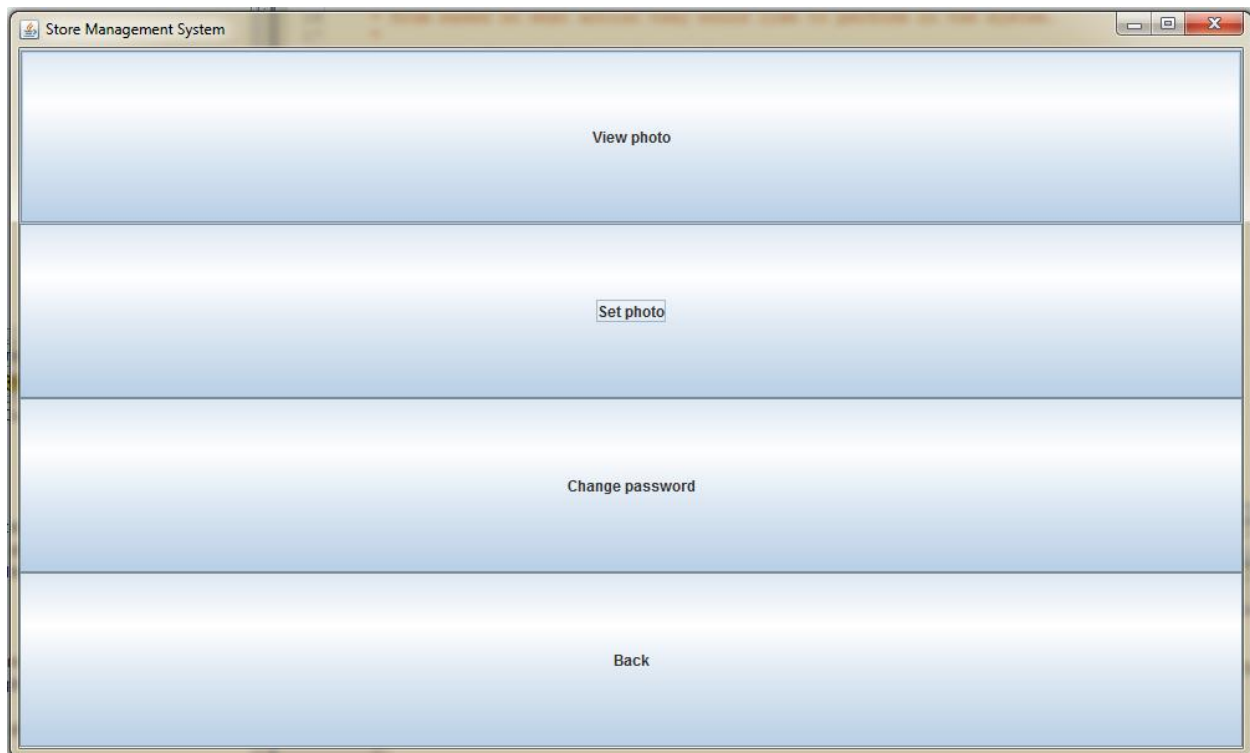
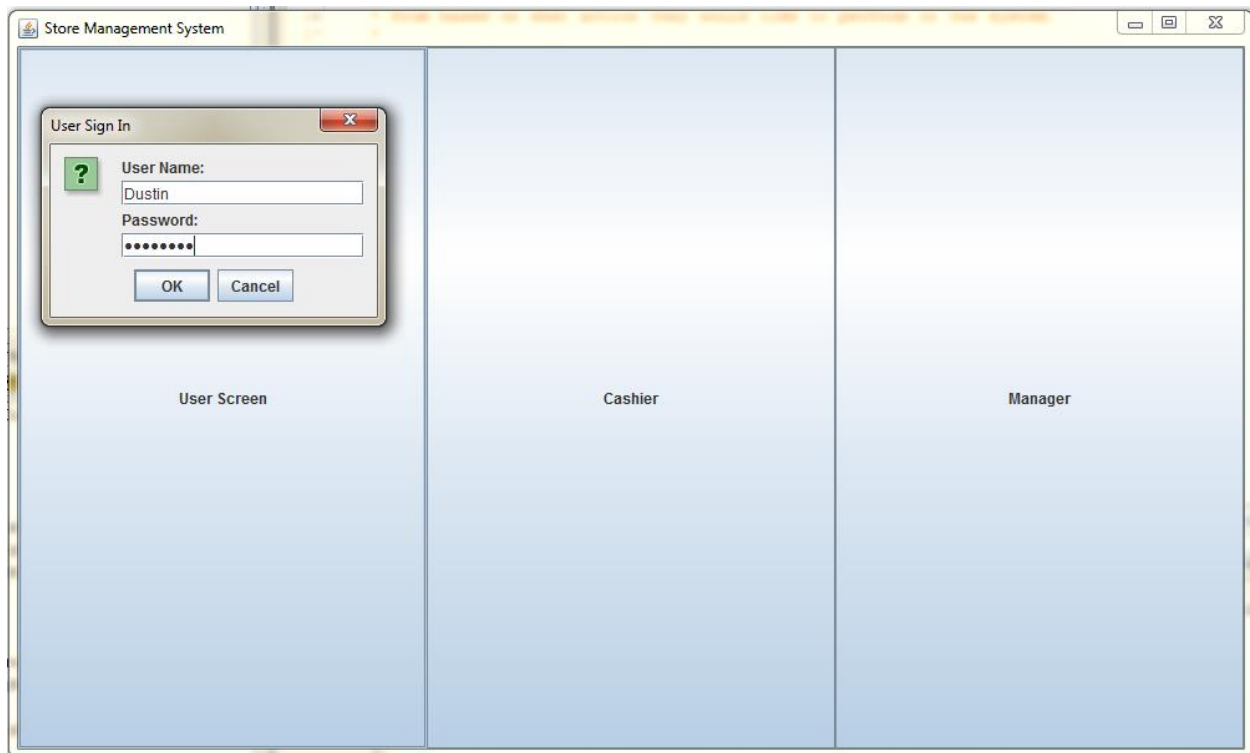


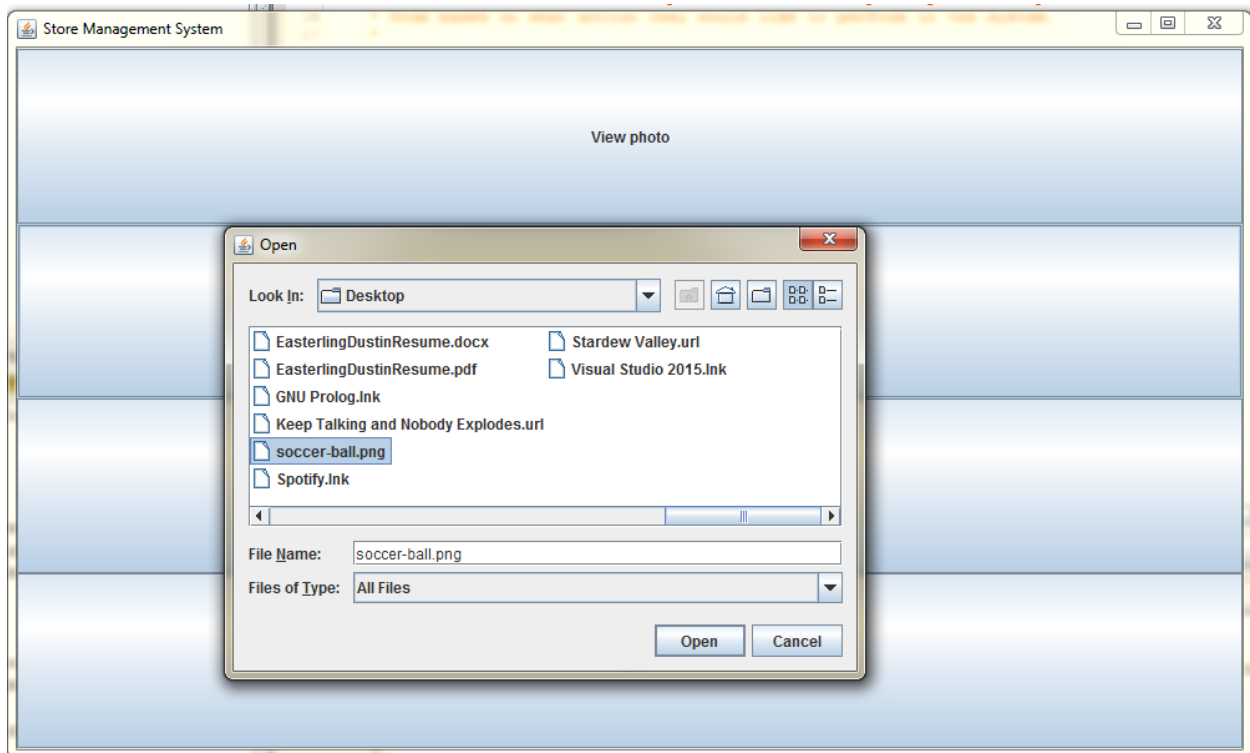


3. A user changes their profile picture.

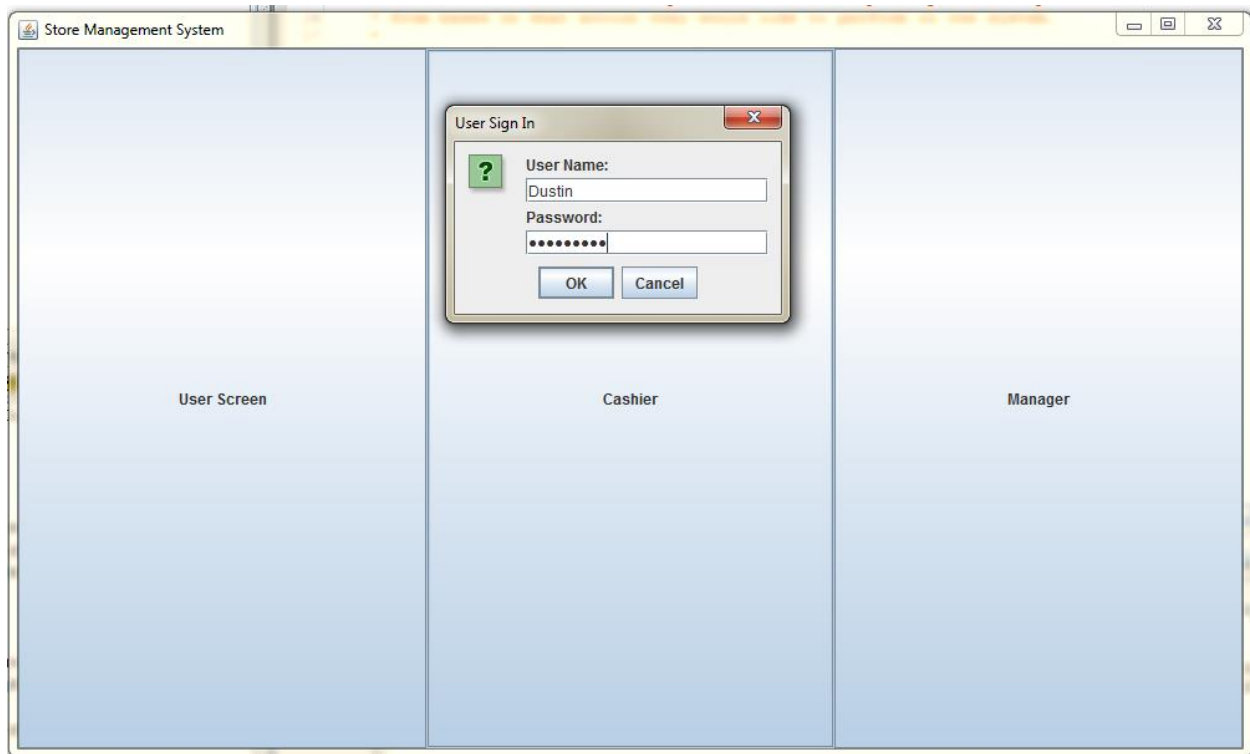








4. A cashier checks out an order.



Store Management System

ID#	Name	Quantity	Weight	Price
-----	------	----------	--------	-------

Total: \$0.0

Add Item

Remove

Apply Discount

Override

Payment

Back

Store Management System

ID#	Name	Quantity	Weight	Price
-----	------	----------	--------	-------

Total: \$0.0

Add Item

Remove

Apply Discount

Override

Payment

Back

Add Item

? ID#: 100

Quantity: 2

OK Cancel

Store Management System

ID#	Name	Quantity	Weight	Price
100	apple	2.0	0.0	10.0
101	coke	3.0	0.0	3.0
109	carrots	3.0	0.0	2.5
105	poptarts	7.0	0.0	4.5
104	walnuts	2.0	0.0	7.0

Total: \$82.0

Add Item

Remove

Apply Discount

Override

Payment

Back

Store Management System

ID#	Name	Quantity	Weight	Price
100	apple	2.0	0.0	10.0
101	coke	3.0	0.0	3.0
109	carrots	3.0	0.0	2.5
105	poptarts	7.0	0.0	4.5
104	walnuts	2.0	0.0	7.0

Total: \$82.0

Cash

Credit/Debit

Gift Card

Finish

Back

Cash Payment

?

Enter Cash Amount:

82.00

OK

Cancel

Store Management System

ID#	Name	Quantity	Weight	Price
100	apple	2.0	0.0	10.0
101	coke	3.0	0.0	3.0
109	carrots	3.0	0.0	2.5
105	poptarts	7.0	0.0	4.5
104	walnuts	2.0	0.0	7.0

Total: \$0.0

Cash

Credit/Debit

Gift Card

Finish

Back

Store Management System

ID#	Name	Quantity	Weight	Price
-----	------	----------	--------	-------

Total: \$0.0

Cash

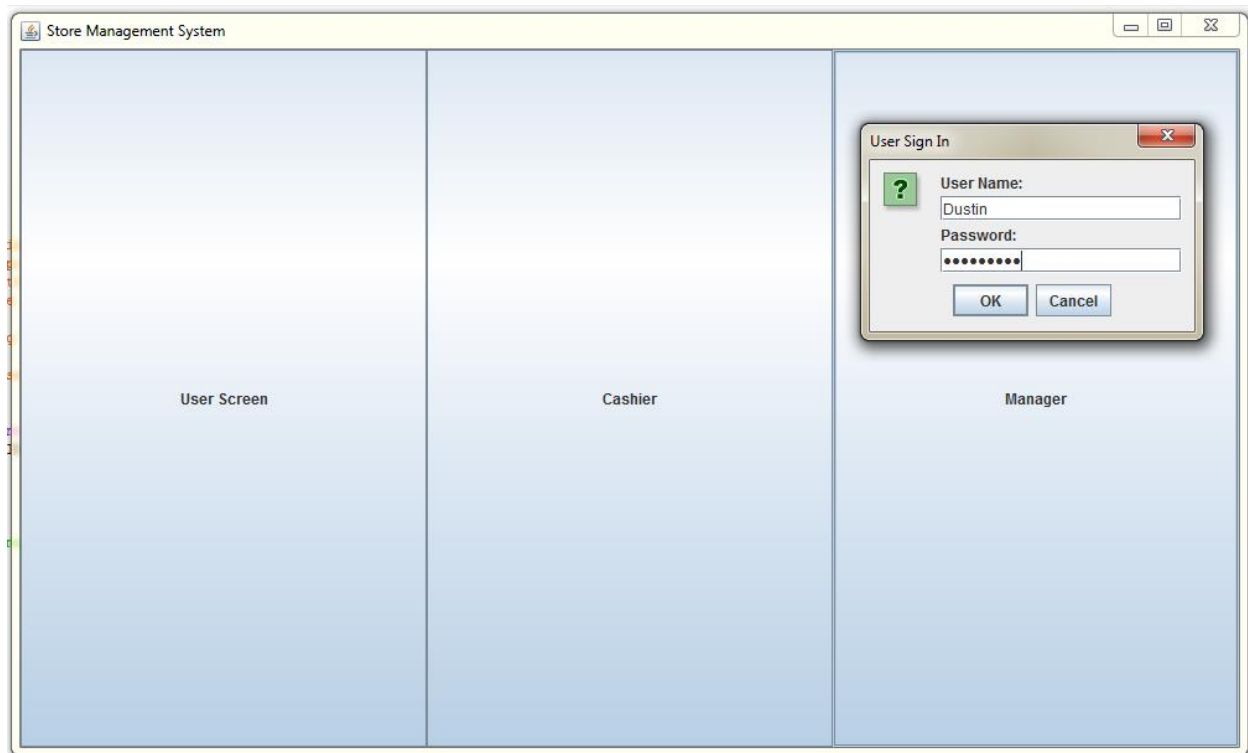
Credit/Debit

Gift Card

Finish

Back

5. A manager adds a new product into the database.



Store Management System

Add User

View Report

New Item

Update Item

Back

Create New Item

ID#  
110

Name  
Plums

Price  
1.50

Weight  
0

Quantity  
100

Producer  
400

OK Cancel

Store Management System

Add User

View Report

New Item

Update Item

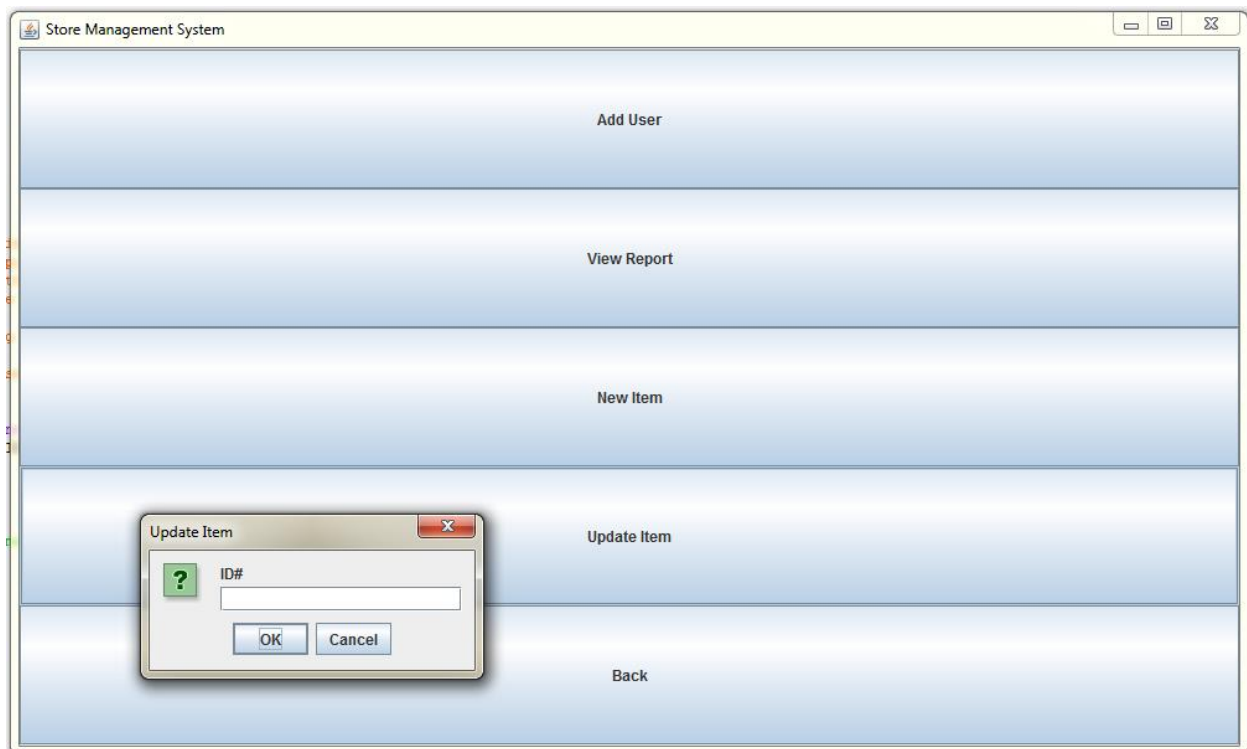
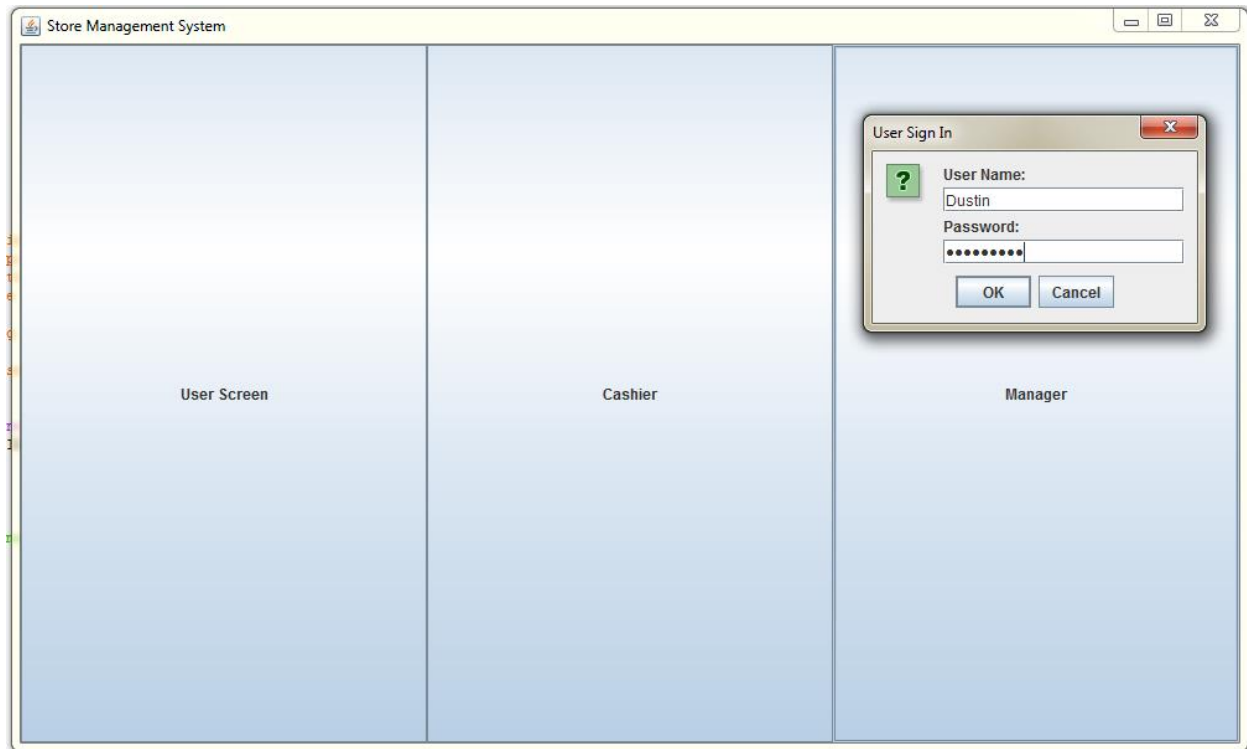
Back

Message

Product Added

OK

6. A manager updates a current product in the database.





Store Management System

ID#	Name	Quantity	Weight	Price
102	milk	5.0	0.0	4.0
106	cheezits	8.0	0.0	4.0

Total: \$52.0

Add Item

Remove

Apply Discount

Override

Payment

Back

Store Management System

ID#	Name	Quantity	Weight	Price
102	milk	5.0	0.0	4.0
106	cheezits	8.0	0.0	4.0

Total: \$52.0

Cash

Credit/Debit

Gift Card

Finish

Back

Store Management System

ID#	Name	Quantity	Weight	Price
102	milk	5.0	0.0	4.0
106	cheezits	8.0	0.0	4.0

Cash

Credit/Debit

Gift Card

Finish

Back

Total: \$52.0

Cash Payment

Enter Cash Amount

52.0

OK Cancel

Store Management System

ID#	Name	Quantity	Weight	Price
102	milk	5.0	0.0	4.0
106	cheezits	8.0	0.0	4.0

Cash

Credit/Debit

Gift Card

Finish

Back

Total: \$0.0

Receipt!

Modeling 3700 Store

Order Number: 21

Cashier: Dustin

Customer: James

Date: 2017.12.14.15.37.25

102 milk 5.0 \$4.0

106 cheezits 8.0 \$4.0

Total Tax: \$4.68

Total: \$56.68

Thanks for Shopping With Us!!!

OK

The screenshot shows a Java Swing window titled "Store Management System". The window has a standard Mac OS X title bar with minimize, maximize, and close buttons. Inside the window, there's a large white rectangular area on the left side, which appears to be a placeholder for a list or grid of items. To the right of this area is a vertical stack of five light blue rectangular buttons with rounded corners. From top to bottom, the buttons are labeled "Cash", "Credit/Debit", "Gift Card", "Finish", and "Back". At the very bottom of the window, centered below the main white area, is the text "Total: \$0.0". The background of the desktop behind the window features a colorful abstract pattern.