



National University Of Computer and Emerging Sciences



OpenBravo Enhancements

Muhammad Faraz (07-0119)

Ihsan-ul-Haq Pervaiz (07-0118)

Saleh Shah (06-0077)

Sultan Ahmed (06-0186)

Supervisor

Dr. Fakhar-ul-Islam Lodhi

B.S. Computer Science

Final Year Project: December 2010

Department of Computer Science

FAST-NU, Lahore, Pakistan

Intellectual Property Right Declaration

This is to declare that the work under the title OpenBravo Enhancements

carried out in partial fulfillment of the requirements of:

BS FYP MS Research Survey MS Thesis Course Project

is the sole property of the National University of Computer and Emerging Science, and is protected under the intellectual property right laws and conventions. It can only be considered / used for purposes like extension for further enhancement, product development, adoption for commercial/organizational usage, etc., with the permission of the University.

This above statement applies to all students and faculty.

Date: 31st December, 2010

Student 1

Name:

Signature: _____

Student 2

Name:

Signature: _____

Student 3

Name:

Signature: _____

Student 4

Name:

Signature: _____

Supervisor (Faculty)

Name: Dr. Fakhar-ul-Islam Lodhi

Signature: _____

Anti-Plagiarism Declaration

This is to declare that the above publication produced under the title:

Openbravo Enhancements

is the sole contribution of the author(s) and no part hereof has been reproduced on **as it is** basis (cut and paste) which can be considered as **Plagiarism**. All referenced parts have been used to argue the idea and have been cited properly. I/We will be responsible and liable for any consequence if violation of this declaration is determined.

Date: 31st December, 2010

Student 1

Name:

Signature: _____

Student 2

Name:

Signature: _____

Student 3

Name:

Signature: _____

Student 4

Name:

Signature: _____

Table of Contents

1. Introduction.....	1
2. Goals and objectives.....	2
3. Scope of the project.....	2
4. Development Lifecycle and Project Summary.....	4
5. Functional Requirements.....	5
6. Actors.....	9
7. Use-Cases.....	10
8. Graphical User Interfaces.....	24
9. Requirements Traceability Matrix.....	31
10. Class Design.....	32
11. Data Design.....	34
12. Implementation.....	35
13. Conclusion and Future Work.....	37
14. References.....	39
15. Glossary.....	40

Abstract

Openbravo is an opensource suite for small and medium-sized businesses comprising of ERP and POS applications. This report deliberates upon the various aspects of the Openbravo Enhancements Project intended to add some useful extensions to existing applications in order to make it more useful for businesses.

Point of Sale (POS) Applications are meant to automate the sales terminal related business processes. Within the realm of Openbravo POS system, this project adds provision for Customer Loyalty Management. The purpose of Loyalty Management System is to increase sales in a Store and attract customers by offering them different deals and discounts and keep track of all the customers who avail those offers.

Furthermore, stores will give customers loyalty cards and when customer shops from specified stores then few points are added to his account according to the purchase, when these points reach certain level then different deals are offered to the customer depending upon the number of points in his account, when customer wants to avail deal he will shop from the store and after deducting some of his points he will be given discounts on that purchase and hence customer will avail deal.

Another aspect of project is to add better support for production planning of sales orders by allowing a greater control to end-users of which sales orders to consider for the production within Openbravo ERP application. This will help stream-line a lot of sales and production related processes for a manufacturing business.

1. Introduction

This document contains complete documentation for Openbravo Enhancements Project aiming at the addition of Customer Loyalty Management module to the existing Openbravo Point-Of-Sale (POS) software and improvement of Sales Orders Production Planning process in Openbravo ERP.

Customer Loyalty Management aims to provide an opportunity to the store management to attract valuable customers and increase sales.

It can do so by using one or more of the functionalities that we have provided. They are as follows:

1. Add / Update / Delete various shopping deals. Following deals shall be supported:
 - a. Buy M, GetN
 - b. Discount based upon products
 - c. Discount based upon shopping
 - d. Discount based upon time
2. Award points to customers on their shopping pattern
3. Utilize customer loyalty points
4. Avail deals

Openbravo ERP enhancements include the following features:

1. A new filter for selection of specific orders to plan for a production cycle
2. Generation of proper production plan for a given or multiple orders

This document is divided into following sections:

1. Introduction
2. Goals and objectives
3. Scope of the project
4. Development Lifecycle and Project Summary
5. Functional Requirements
6. Non-Functional Requirements
7. Actors

8. Use-Cases
9. Graphical User Interfaces
10. Requirements Traceability Matrix
11. Assumptions
12. Class Design
13. Data Design
14. Implementation
15. Conclusion
16. References
17. Glossary
18. Appendix

2. Goals and objectives

The following goals are envisioned for this project:

1. To extend Openbravo POS:
 - (a) to offer various kind of deals on various products
 - (b) To keep a record of valuable customers
 - (c) Allow valuable customers to avail deals
2. To extend Openbravo ERP to improve existing production planning by incorporating order-based planning

3. Scope of the project

The following functionality has been considered as a scope for this implementation:

1. Openbravo POS:
 - (a) Add / Update / Delete various shopping deals. Following deals shall be supported:
 - i. Buy M, Get N
 - ii. Discount based upon products
 - iii. Discount based upon shopping
 - iv. Discount based upon time
 - (b) Award points to customers on their shopping pattern
 - (c) Utilize customer loyalty points

(d) Avail deals

2. Order-based Production Planning in Openbravo ERP

4. Development Lifecycle and Project Summary

Incremental Lifecycle was used and the work was done in two major increments.

An outline of the work done under the increments is as follows:

Increment 1

- Code Understanding
- Use Case Development
- Database schema
- Add, Remove and Update Buy M, Get N deal.
- Add, Remove and Update discount based on product.
- Support for MySQL, Oracle, PostgreSQL, Derby.

Increment 2

- Add, Remove and Update discount based on time
- Add, Remove and Update discount based on shopping
- Award points to customers on purchase of items
- Utilize customer loyalty points
- To Avail Deals

5. Functional Requirements

Openbravo POS

1. Add Deal

- a. User shall be able to add a new deal of Buy M Get N type
- b. User shall be able to add a new deal of discount based on product type.
- c. User shall be able to add a new deal of discount based on shopping type.
- d. User shall be able to add a new deal of discount based on time type.
- e. User shall be able to add a new deal based upon customer loyalty points.

2. Delete Deal

- a. User shall be able to delete an existing deal of Buy M Get N type
- b. User shall be able to delete an existing deal of discount based on product type.
- c. User shall be able to delete an existing deal of discount based on shopping type.
- d. User shall be able to delete an existing deal of discount based on time type.
- e. User shall be able to delete an existing deal based upon customer loyalty points.

3. Update Deal

- a. User shall be able to update attributes of an existing deal of Buy M Get N type
- b. User shall be able to update attributes of an existing deal of discount based on product type.
- c. User shall be able to update attributes of an existing deal of discount based on shopping type.
- d. User shall be able to update attributes of an existing deal of discount based on time type.
- e. User shall be able to update attributes of an existing deal based upon customer loyalty points.

4. Close Deal

User shall be able to close/deactivate deals of the following types:

- a. Buy M Get N
- b. Discount based on product
- c. Discount based on shopping
- d. Discount based on time

5. Award Points to Customers

- a. System should avoid customer loyalty points to customers on if their shopping amount exceeds a certain level.

6. Utilize Points

- a. User shall be able to award free products specified by Points Deals to customers if the points required to obtain it are less than or equal to customer's points.

7. Avail Deals

- a. System should be able to apply specific deals if criteria associated with each deal are met.
- b. For further elicitation of these requirements, consider the following examples:
- c. Buy M Get N
- d. If user Buys M amount of Product X, he shall Get N amount of Product Y
- e. Product Based Discount
- f. If user Buys Product X worth \$100 and there is 25% Discount on Product, then he shall have to Pay \$75.
- g. Shopping Based Discount
- h. If user's Total Shopping exceeds \$1000 and there is 10% Discount on shopping that exceeds \$1000, then user shall have to Pay \$900.
- i. Time Based Discount
- j. If user's Total Shopping for \$1000, if the purchase is within "1st Jan 2010" and "1st Feb 2010" and there is 5% Discount on Total Price, than user shall have to Pay \$950.

Openbravo ERP

8. Order filter and Selection of Multiple Orders in MRP Manufacturing Plan

- a. System shall provide support for selecting orders to plan in the MRP Manufacturing Plan. The following constraints/provisions shall apply:
 - i. Only those orders with Status marked as “Completed” shall be shown.
 - ii. User shall be able to select multiple orders
 - iii. The following information relating to each sales order shall be shown during the selection process in order to facilitate selection:
 1. Customer Name: Name of the customer (represented as Business Partner) who placed order
 2. Delivery Date: The promised delivery date for complete order
 3. Order Number: The document number representing the sales order
 4. Order Amount: The total amount of order
 5. Description: Description of sales order
 - iv. User shall be able to filter sales orders based upon the following attributes:
 1. Order Number: A string representing the Document Number of Sales Order. If user specifies the Order Number then only the order with the given Order Number after applying other filtering criteria shall be shown in the result
 2. Customer: A customer (Business Partner) who placed the order. Customer shall be selectable from a list of customers recorded in the system. If user selects a customer then all the orders placed by the selected customer after applying other filtering criteria shall be shown
 3. From Date: A date representing the starting date of the period. If user provides a From Date, then only those orders having Delivery Date greater than or equal to the specified From Date, after applying other filtering criteria shall be shown.

4. To Date: A date representing the ending date of the period. If user provides a To Date, then only those orders having Delivery Date lesser than or equal to the specified To Date, after applying other filtering criteria shall be shown.

9. Changes in Generation of MRP Manufacturing Plan

- a. System shall accommodate the selected sales orders (as a result of section Error: Reference source not found) in the generation of Manufacturing Plan. The changes to be incorporated shall be:

- i. If user selects any sales order(s), system shall generate Manufacturing Plan for only selected orders, after applying other filtering criteria in the current Manufacturing Plan window. For example:

if user selected Sales Orders “1000000”, “1000001” and “1000002”, the Manufacturing Plan shall include production of only those products (and their intermediaries) that are included in these orders if user selected Sales Orders “1000000”, “1000001” and “1000002” and also selected Product “Bicycle”, and Bicycle is only part of Sales Order “1000000” then only Product Bicycle (and its intermediaries) shall be included in the manufacturing plan according to the specifications listed in Sales Order “1000000”. The rest of the sales orders shall be excluded from the manufacturing plan in this case.

- ii. If user does not select any sales order, system shall generate all Manufacturing Plan for all orders that satisfy other filtering criteria in the current Manufacturing Plan window.

6. Actors

OpenBravo POS has the following actors:

1. Administrator
2. Employee

Openbravo ERP has the following actors:

1. Production Planner

For the detail interaction of the actors with the system, kindly refer to Article 9 (Requirements Traceability Matrix).

7. Use-Cases

7.1 Use-Case Name

Add Deal: Buy M Get N

7.1.1 Brief Description

User shall be able to add new deals of type Buy M Get N.

7.1.2 Pre-condition(s)

Admin should be logged into the system and be on the “Deals” page.

7.1.3 Main Flow

1. User clicks on the “Buy M Get N” button.
2. System takes him to the Buy M Get N page.
3. User clicks on the “New” button.
4. System creates a new form.
5. User enters following deal details:
 - a. Deal Name
 - b. Start Date
 - c. End Date
 - d. Active Status
 - e. The product to be bought
 - f. Quantity of the product to be bought (Units)
 - g. The product to be give free
 - h. Quantity of the product to be given free (Units)
6. User clicks on the “Save” button.
7. System saves the deal.

7.1.4 Alternative Flows

5c. if the end date is less than the start date, date will not be entered.

User must re-enter the end date.

5e. if the product entered is already involved in another deal, system gives an error.

User must enter another product.

7. if the user has left a field empty, the system will give him a “Record Not Saved” error.

7.1.5 Post-condition(s)

System displays the new deal in the list of deals on the left side of the same page.

7.2 Use-Case Name

Add Deal: Discount on Products

7.2.1 Brief Description

User shall be able to add new deals of type Discount. The discount will be associated with each product.

7.2.2 Pre-condition(s)

Admin should be logged into the system and be on the “Deals” page.

7.2.3 Main Flow

1. User clicks on the “Product based Discount” button.
2. System takes him to the Discount Products page.
4. User clicks on the “New” button.
5. System creates a new form.
6. User enters following deal details:
 - a. Deal Name
 - b. Start Date
 - c. End Date
 - d. Active Status
 - e. The product to be bought
 - f. Percentage discount on the product
7. User clicks on the “Save” button.
8. System saves the deal.

7.2.4 Alternative Flows

5c. if the end date is less than the start date, date will not be entered.

User must re-enter the end date.

5e. if the product entered is already involved in another deal, system gives an error.

User must enter another product.

7. if the user has left a field empty, the system will give him a “Record Not Saved” error.

7.2.5 Post-condition(s)

System displays the new deal in the list of deals on the left side of the same page.

7.3 Use-Case Name

Add Deal: Discount Based on Time

7.3.1 Brief Description

User shall be able to add new deals of Discount type. The deal will give discount on the total shopping amount if the shopping is within a specified time range.

7.3.2 Pre-condition(s)

Admin should be logged into the system and be on the “Deals” page.

7.3.3 Main Flow

1. User clicks on the “Time based Discount” button.
2. System takes him to the Discount Products page.
3. User clicks on the “New” button.
4. System creates a new form.
5. User enters following deal details:
 - a. Deal Name
 - b. Start Date
 - c. End Date
 - d. Active Status
 - e. Percentage off on the total shopping amount
6. User clicks on the “Save” button.
7. System saves the deal.

7.3.4 Alternative Flows

5c. if the end date is less than the start date, date will not be entered.

User must re-enter the end date.

7. if the user has left a field empty, the system will give him a “Record Not Saved” error.

7.3.5 Post-condition(s)

System displays the new deal in the list of deals on the left side of the same page.

7.4 Use-Case Name

Add Deal: Discount Based on Shopping

7.4.1 Brief Description

User shall be able to add new deals of Discount type. The deal will give discount on the total shopping amount if it exceeds a certain limit.

7.4.2 Pre-condition(s)

Admin should be logged into the system and be on the “Deals” page.

7.4.3 Main Flow

1. User clicks on the “Shopping based Discount” button.
2. System takes him to the Discount Products page.
3. User clicks on the “New” button.
4. System creates a new form.
5. User enters following deal details:
 - a. Deal Name
 - b. Start Date
 - c. End Date
 - d. Active Status
 - e. Minimum price limit, which if exceeded would make the deal applicable.
 - f. Percentage off on the total shopping amount.
6. User clicks on the “Save” button.
7. System saves the deal.

7.4.4 Alternative Flows

5c. if the end date is less than the start date, date will not be entered.

User must re-enter the end date.

7. if the user has left a field empty, the system will give him a “Record Not Saved” error.

8.4.5 Post-condition(s)

System displays the new deal in the list of deals on the left side of the same page.

7.5 Use-Case Name

Add Deal: Points Based

7.5.1 Brief Description

User shall be able to add new deals based on Customer loyalty points.

7.5.2 Pre-condition(s)

Admin should be logged into the system and be on the “Deals” page.

7.5.3 Main Flow

1. User clicks on the “Points based Deals” button.
2. System takes him to the “Points based Deals” page.
3. User clicks on the “New” button.
4. System creates a new form.
5. User enters following deal details:
 - a. Deal Name
 - b. Start Date
 - c. End Date
 - d. Active Status
 - e. The product to be awarded
 - f. Minimum points required to avail this deal (Required Points)
6. User clicks on the “Save” button.
7. System saves the deal.

7.5.4 Alternative Flows

5c. if the end date is less than the start date, date will not be entered.

User must re-enter the end date.

7. if the user has left a field empty, the system will give him a “Record Not Saved” error.

7.5.5 Post-condition(s)

System displays the new deal in the list of deals on the left side of the same page.

7.6 Use-Case Name

Delete Deal

7.6.1 Brief Description

User shall be able to delete a previously added deal.

7.6.2 Pre-condition(s)

Admin should be logged into the system and be on the “Deals” page.

7.6.3 Main Flow

1. User clicks on one of the following:
 - a. “Buy M Get N” button.
 - b. “Product based Discount” button.
 - c. “Time based Discount” button.
 - d. “Shopping based Discount” button.
2. System takes him to the page corresponding to the button.
3. User selects a deal from the available deals listed in leftmost column.
4. System displays the deal details.
5. User clicks on the “Delete” button.
6. User clicks on the “Save” button.
7. System saves the deal.

8.6.4 Alternative Flows

None

8.6.5 Post-condition(s)

System displays removes the deal from the list of deals on the left side of the same page.

7.7 Use-Case Name

Update/Modify Deal

7.7.1 Brief Description

User shall be able to update/modify the previously added deal.

7.7.2 Pre-condition(s)

Admin should be logged into the system and be on the “Deals” page.

7.7.3 Main Flow

1. User clicks on one of the following:
 - a. “Buy M Get N” button.
 - b. “Product based Discount” button.
 - c. “Time based Discount” button.
 - d. “Shopping based Discount” button.
2. System takes him to the page corresponding to the button.
3. User selects a deal from the available deals listed in leftmost column.
4. System displays the deal details.
5. User modifies any or all of the fields.
6. User clicks on the “Save” button.
7. System saves the deal.

7.7.4 Alternative Flows

5. if the end date is less than the start date, date will not be entered.
User must re-enter the end date.
5. if the product entered is already involved in another deal, system gives an error.
User must enter another product.
5. if the user has left a field empty, the system will give him a “Record Not Saved” error.

7.7.5 Post-condition(s)

System displays the updated deal in the list of deals on the left side of the same page.

7.8 Use-Case Name

Avail Deal: Buy M Get N

7.8.1 Brief Description

User shall be able to avail a previously added deal of Buy M Get N type. The deal will award N quantity of free products to a customer on the purchase of M quantity of another product.

7.8.2 Pre-condition(s)

User should be logged into the system and be on the “Sales” page.

7.8.3 Main Flow

1. User selects a product.
2. System checks if the product bought is under Buy M Get N deals.
 - a. If yes, System adds N quantity of the product specified by the deal.

7.8.4 Alternative Flows

2. If the deal is not active, then free product will not be awarded.

7.8.5 Post-condition(s)

System displays the newly added product(s) in the current ticket.

7.9 Use-Case Name

Avail Deal: Product based discount

7.9.1 Brief Description

User shall be able to avail a previously added deal of Discount type. The deal will discount a certain price from the added product.

7.9.2 Pre-condition(s)

User should be logged into the system and be on the “Sales” page.

7.9.3 Main Flow

1. User selects a product.
3. System checks if the product bought is under Product based Discount deals.
 - b. If yes, System deducts the discount percentage from product’s price.

7.9.4 Alternative Flows

2. If the deal is not active, then free product will not be awarded.

7.9.5 Post-condition(s)

System displays the newly added product with the discounted price in the current ticket.

7.10 Use-Case Name

Avail Deal: Shopping based Discount

7.10.1 Brief Description

User shall be able to avail a previously added deal of Shopping based Discount type. The deal will deduct a certain discount percentage from the total shopping price.

7.10.2 Pre-condition(s)

User should be logged into the system and be on the “Sales” page.

7.10.3 Main Flow

1. User clicks on “=” button.
2. System checks if the total shopping price is greater than the amount specified by any Shopping based Discount deal.
 - a. If yes, System deducts the discount percentage specified by the deal, from the total shopping price.

7.10.4 Alternative Flows

2. If the deal is not active, then free product will not be awarded.

7.10.5 Post-condition(s)

None.

7.11 Use-Case Name

Avail Deal: Time based Discount

7.11.1 Brief Description

User shall be able to avail a previously added deal of Time based Discount type. The deal will deduct a certain discount percentage from the total shopping price if the shopping is within a specified time range.

7.11.2 Pre-condition(s)

User should be logged into the system and be on the “Sales” page.

7.11.3 Main Flow

1. User clicks on “ = ” button.
2. System checks if the current date is within the range of dates specified by any Time based Discount deal.
 - a. If yes, System deducts the discount percentage specified by the deal, from the total shopping price.

7.11.4 Alternative Flows

2. If the deal is not active, then free product will not be awarded.

7.11.5 Post-condition(s)

None

7.12 Use-Case Name

Avail Deal: Points Based

7.12.1 Brief Description

User shall be able to avail a previously added Points based Deal. The deal will award a certain product if the user has sufficient points.

7.12.2 Pre-condition(s)

User should be logged into the system and be on the “Sales” page.

7.12.3 Main Flow

1. User clicks on “ Deals ” from the “Main” menu on the right.
2. System displays all points based deals.
3. User selects a customer.
4. System displays the customer’s points in the “Points” text field.
5. User selects a deal from the list.
6. System deducts the points required for the product from customer’s current points.

7.12.4 Alternative Flows

None

7.12.5 Post-condition(s)

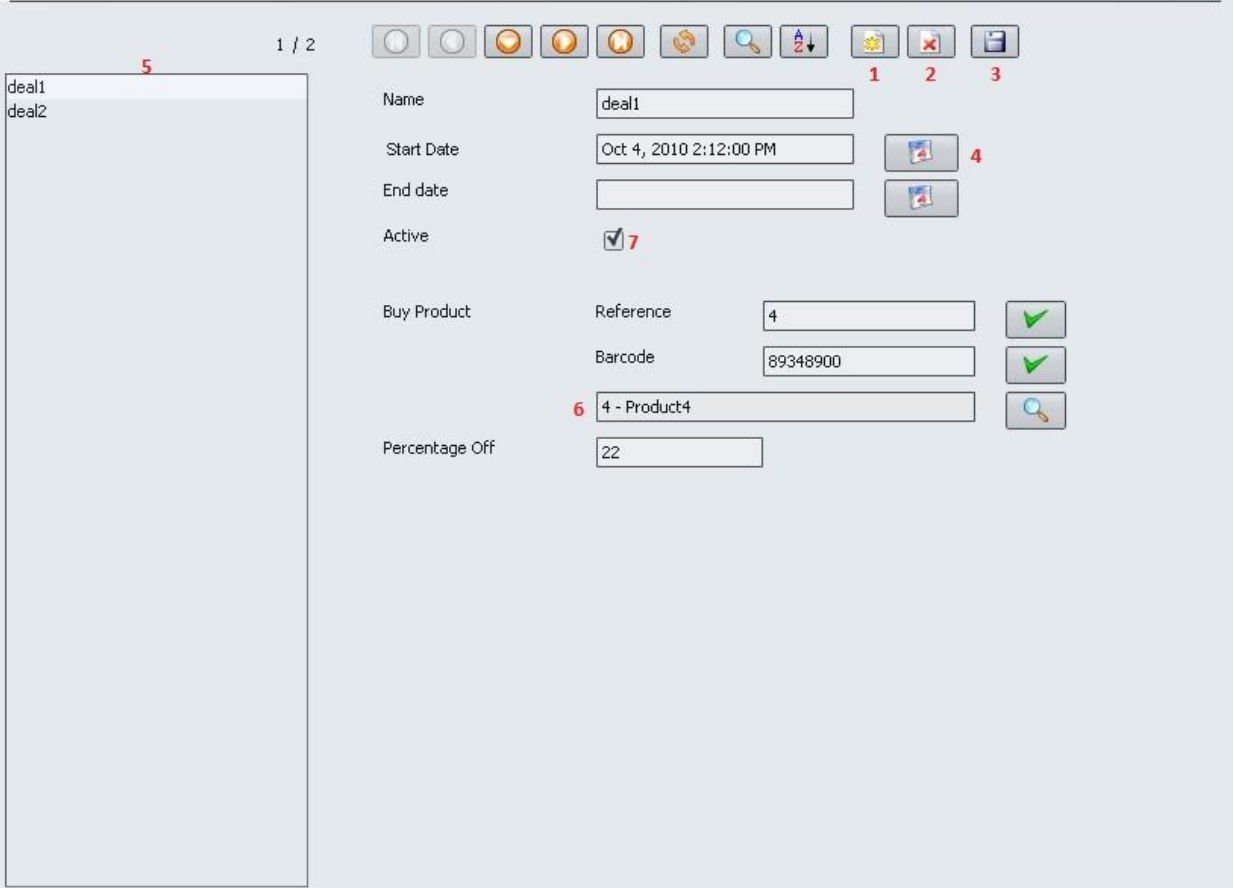
None

8. Graphical User Interfaces

01: Buy M Get N	
Interface Id.	001
Use case Reference	Use Case 8.1 , Use Case 8.6, Use Case 8.7
Snapshot	
<div style="border: 1px solid black; padding: 5px;"> <p>Buy M Get N</p> <div style="display: flex; justify-content: space-between; align-items: center;"> <div style="border: 1px solid gray; padding: 5px; width: 20%;"> <p style="color: red; font-weight: bold;">5</p> <p>1 / 2</p> <p>first deal</p> <p>second deal</p> </div> <div style="width: 80%;"> <div style="display: flex; justify-content: space-between; align-items: center; margin-bottom: 10px;"> <div> <p>Name <input type="text" value="first deal"/></p> <p>Start Date <input type="text" value="Oct 4, 2010 2:09:00 PM"/> 4</p> <p>End date <input type="text" value="Oct 30, 2010 2:09:00 PM"/> </p> <p>Active <input checked="" type="checkbox"/> 8</p> </div> <div style="margin-left: 20px;"> <p>Buy Product</p> <p>Reference <input type="text" value="1"/> </p> <p>Barcode <input type="text" value="1234566"/> </p> <p>6 <input type="text" value="1 - Product1"/> </p> <p>Units <input type="text" value="1"/></p> </div> </div> <div style="margin-top: 10px;"> <p>Free Product</p> <p>Reference <input type="text" value="1"/> </p> <p>Barcode <input type="text" value="1234566"/> </p> <p>7 <input type="text" value="1 - Product1"/> </p> <p>Units <input type="text" value="1"/></p> </div> </div> </div> </div>	

Description

1. Add new deal
2. Delete selected deal
3. Save Changes
4. Open date-picker
5. List displaying all added deals
- 6,7. Selected product's reference number concatenated with product name
8. Shows whether the deal is currently active or not

02: Product Based Discount	
Interface Id.	002
Use case Reference	Use Case 8.2 , Use Case 8.6, Use Case 8.7
Snapshot	
Product based Discount	
	
Description	
<ol style="list-style-type: none"> 1. Add new deal 2. Delete selected deal 3. Save Changes 4. Open date-picker 5. List displaying all added deals 6. Selected product's reference number concatenated with product name 7. Shows whether the deal is currently active or not 	

03: Time Based Discount










Interface Id.	003
Use case Reference	Use Case 8.3 , Use Case 8.6, Use Case 8.7

Snapshot


Time based Discount


1deal

1 / 1

Name

Start Date  4

End date 

Active 7

Percent Off 6

1 2 3

Description

1. Add new deal
2. Delete selected deal
3. Save Changes
4. Open date-picker
5. List displaying all added deals
6. The specified percentage will be deducted from the total shopping amount
7. Shows whether the deal is currently active or not

04: Shopping Based Discount	
Interface Id.	004
Use case Reference	Use Case 8.4 , Use Case 8.6, Use Case 8.7
Snapshot	

Shopping based Discount

two deal 5

one deal

1 / 2

Name	two deal	
Start Date	Oct 16, 2010 2:14:00 PM	4
End date		
Active	<input type="checkbox"/>	6
Price Limit	\$4,000.00	7
Percent Off	55	8

Description

1. Add new deal
2. Delete selected deal
3. Save Changes
4. Open date-picker
5. List displaying all added deals
6. Shows whether the deal is currently active or not
7. The minimum amount above which this deal would be applicable
8. The specified percentage will be deducted from the total shopping amount

05: Points Based Deal	
Interface Id.	005
Use case Reference	<i>Use Case 8.5 , Use Case 8.6, Use Case 8.7</i>
Snapshot	

Points Based Deals

5
1 / 4

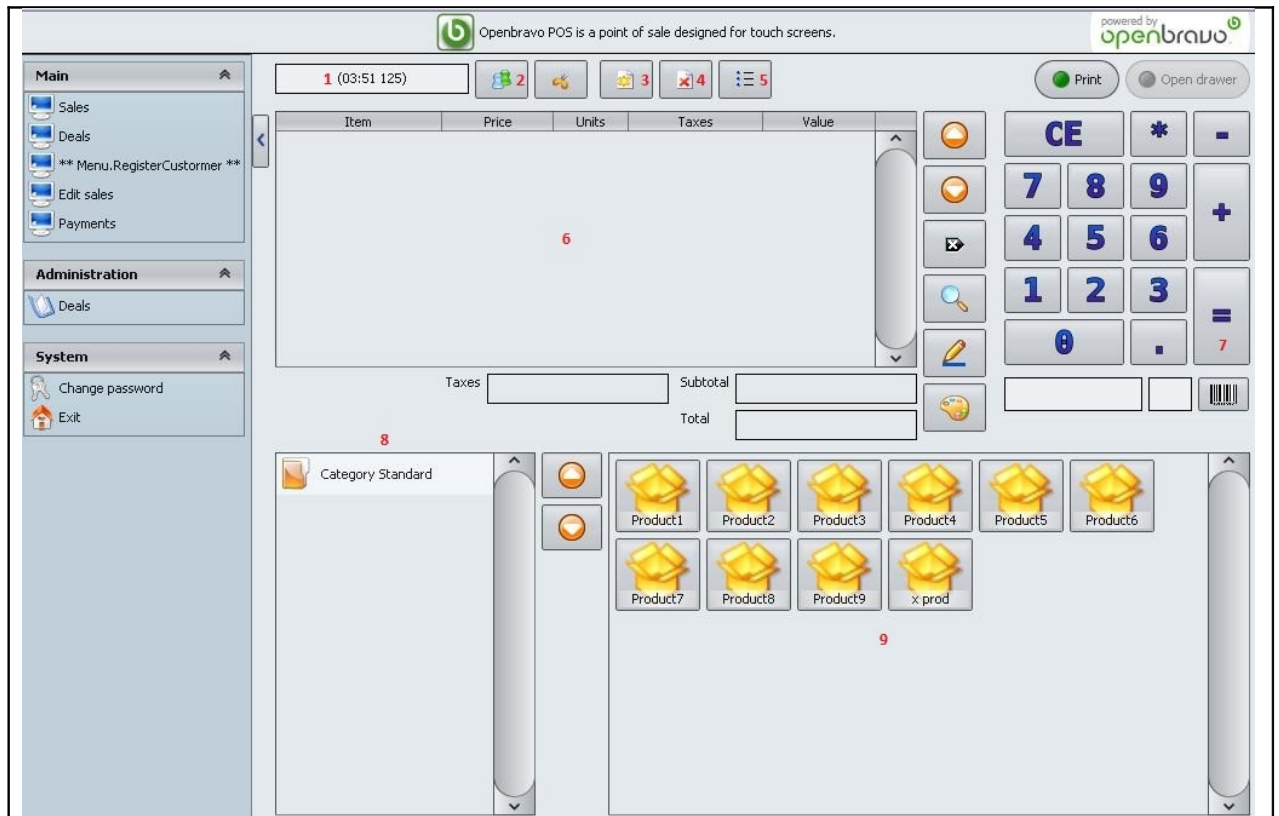
1
2
3

<div style="border: 1px solid gray; height: 150px; padding: 2px;"> <p style="margin: 0;">pd3</p> <p style="margin: 0;">pd2</p> <p style="margin: 0;">pd1</p> <p style="margin: 0;">pd4</p> </div>	<p>Name <input style="width: 150px;" type="text" value="pd3"/></p> <p>Start Date <input style="width: 150px;" type="text" value="Nov 29, 2010 1:02:00 PM"/> 4</p> <p>End date <input style="width: 150px;" type="text" value="Nov 30, 2010 1:02:00 PM"/> </p> <p>Active <input checked="" type="checkbox"/> 6</p> <p>Buy Product</p> <p>Reference <input style="width: 100px;" type="text" value="3"/> </p> <p>Barcode <input style="width: 100px;" type="text" value="34355345354"/> </p> <p>7 <input style="width: 150px;" type="text" value="3 - Product3"/> </p> <p>Required Points 8 <input style="width: 100px;" type="text" value="4"/></p>
---	--

Description

1. Add new deal
2. Delete selected deal
3. Save Changes
4. Open date-picker
5. List displaying all added deals
6. Shows whether the deal is currently active or not
7. Selected product's reference number concatenated with product name
8. The minimum points above which this deal can be availed

06: Sales Panel	
Interface Id.	006
Use case Reference	<i>Use Case 8.8 , Use Case 8.9, Use Case 8.10, Use Case 8.11</i>
Snapshot	

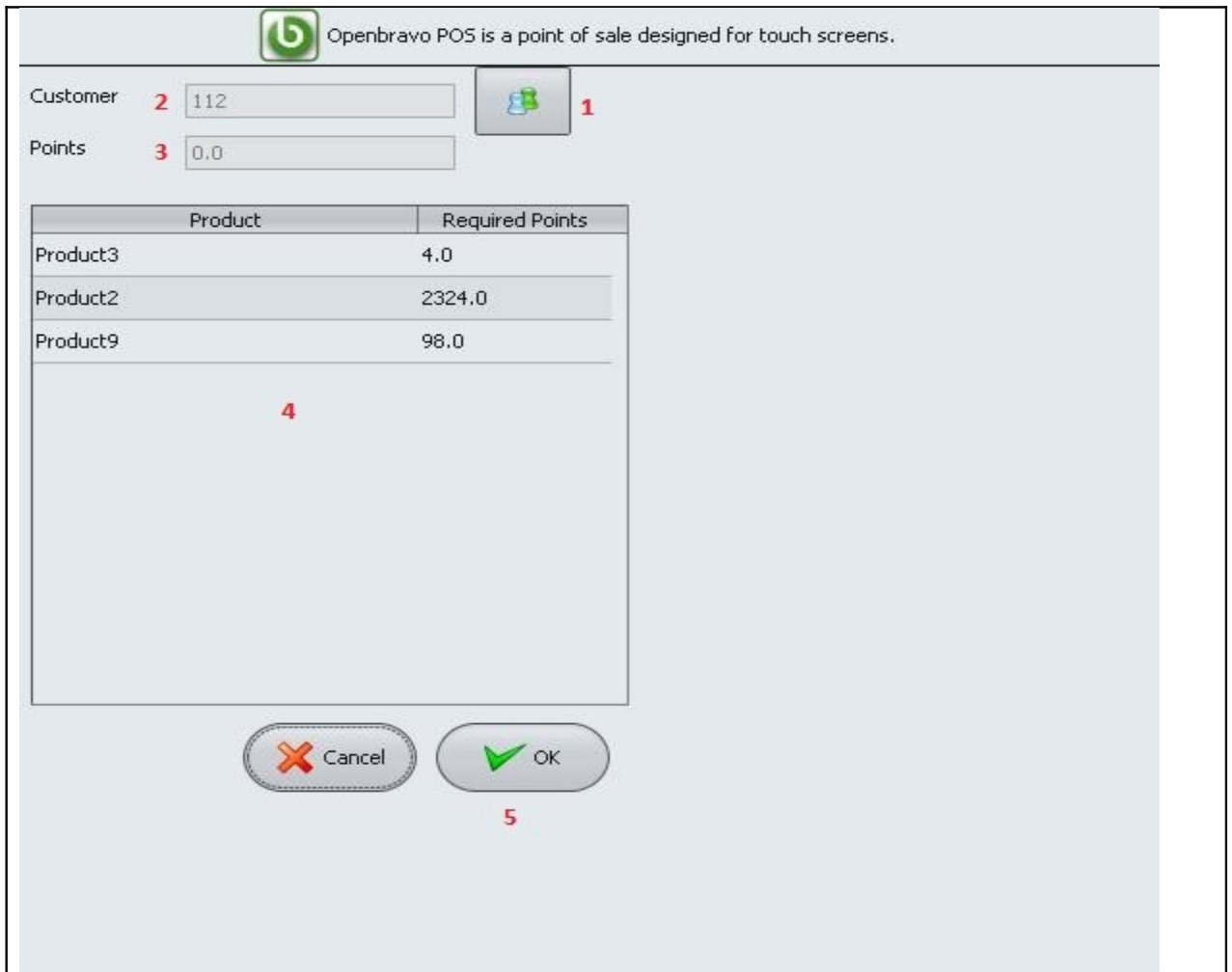


Description

1. This field shows the time at which the ticket was created and the associated customer
2. Add a customer
3. Create new ticket
4. Delete current ticket
5. View currently active tickets
6. Each row in this panel shows a product that has been entered into the ticket
7. This button finalizes a ticket and opens the payment window in a pop up
8. The categories of products to be selected
9. A graphical representation of products existing in the system. Clicking a product will enter it into the receipt.

07: Avail Points Based Deal

Interface Id.	007
Use case Reference	Use Case 8.12
Snapshot	



Description

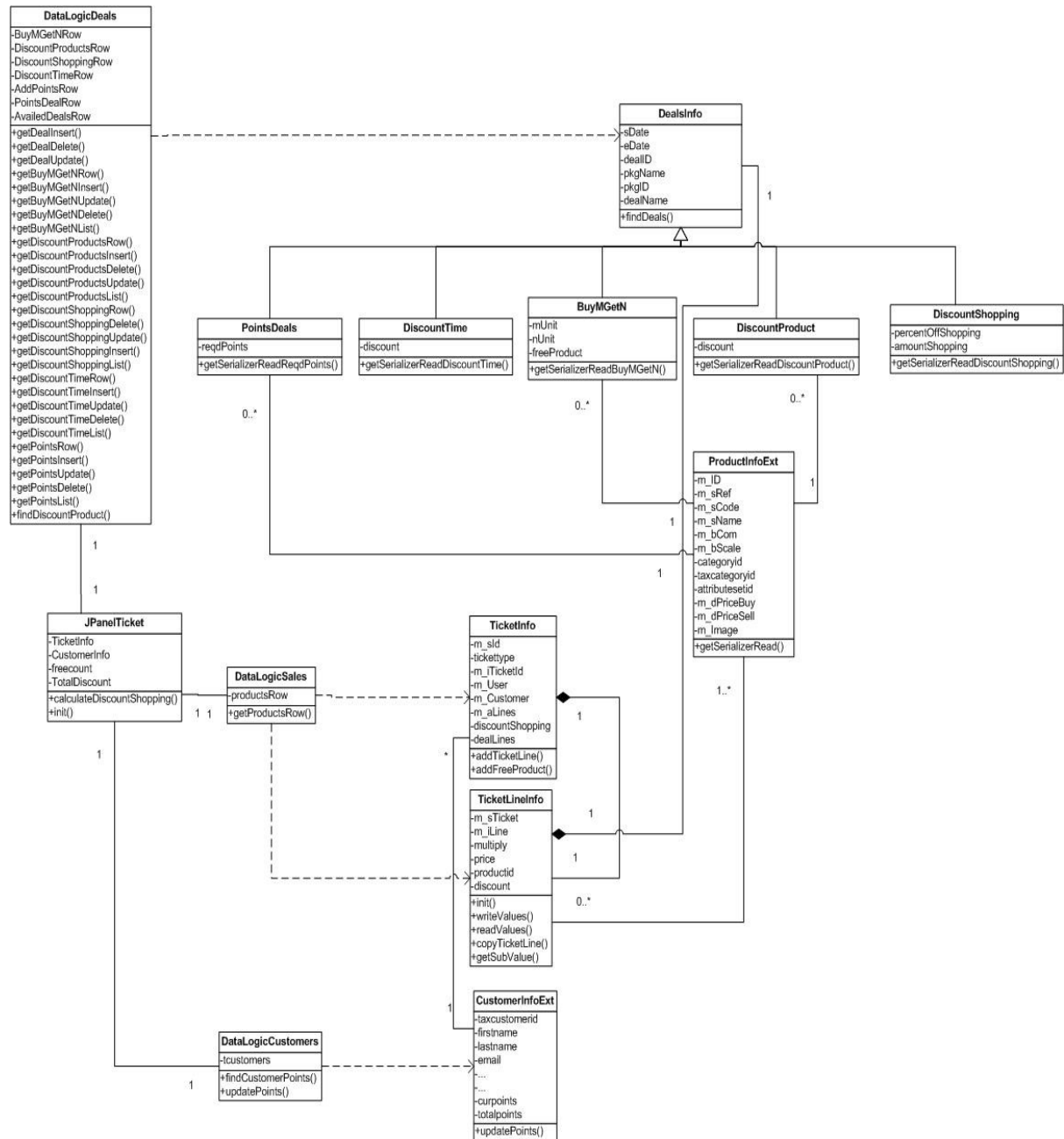
1. Select a customer
2. Customer Id
3. Available loyalty points for the selected customer
4. All available points based deals with the points required for each deal
5. If a deal is selected, OK button will deduct the deal's points from customers current points

9. Requirements Traceability Matrix

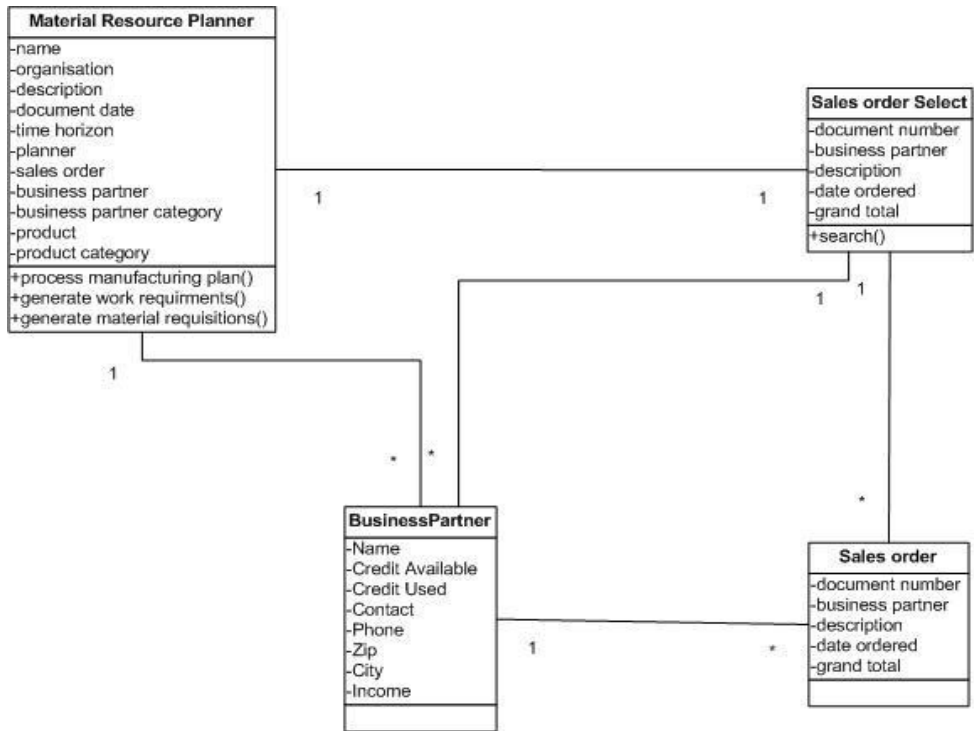
Use Case	Actor	Functional Requirements	Non-Functional Requirements	GUI
8.1	Admin	1a	-	1
8.2	Admin	1b	-	2
8.3	Admin	1d	-	3
8.4	Admin	1c	-	4
8.5	Admin	1e	-	5
8.6	Admin	2(a,b,c,d,e)	-	1,2,3,4,5
8.7	Admin	3(a,b,c,d,e)	-	1,2,3,4,5
8.8	Employee, Admin	7a	-	6
8.9	Employee, Admin	7b	-	6
8.10	Employee, Admin	7c	-	6
8.11	Employee, Admin	7d	-	6
8.12	Employee, Admin	6a	-	7

10. Class Design

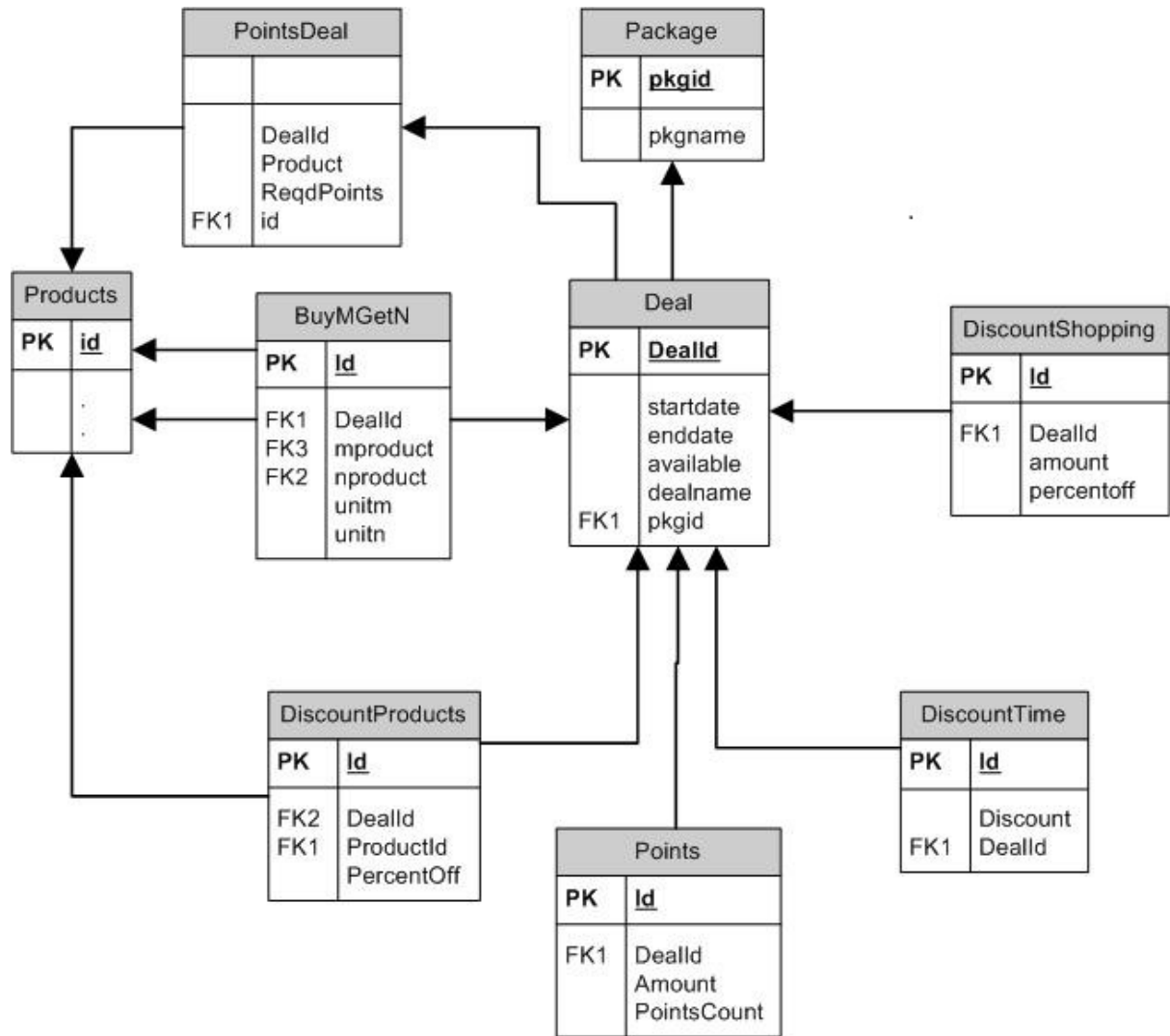
Openbravo POS



Openbravo ERP

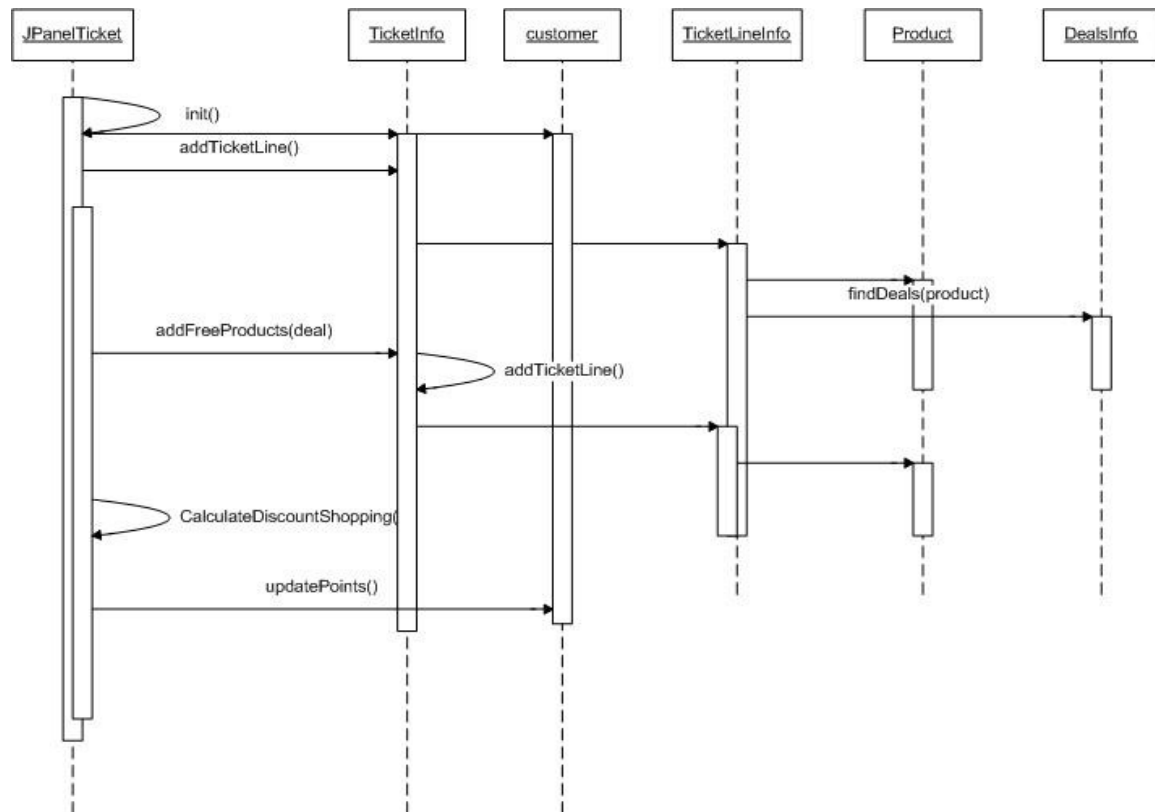


11. Data Design

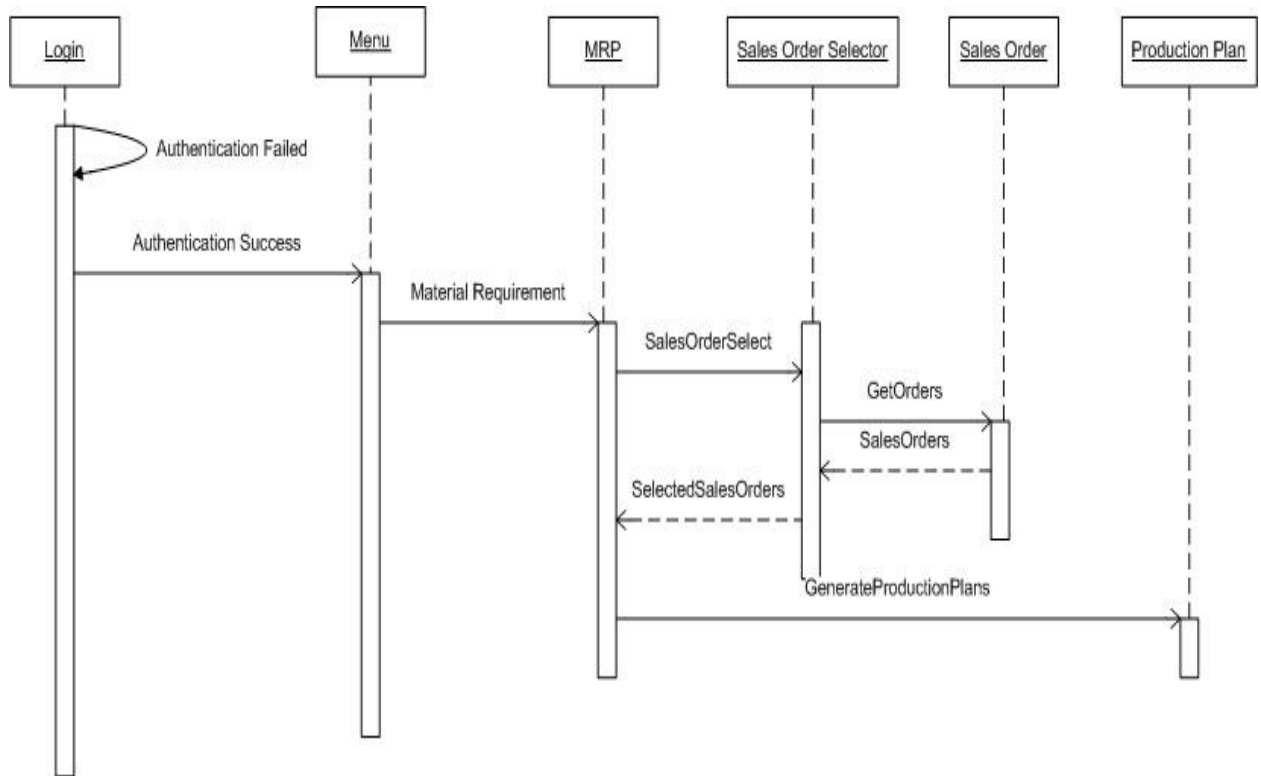


12. Implementation

Openbravo POS



Openbravo ERP



13. Conclusion and Future Work

Working on Openbravo Enhancements project was a unique and interesting experience.

Although there was nothing new in the tools and languages used (NetBeans IDE, Eclipse, Java and SQL), the integration of a new module in the existing complex code of Openbravo POS and Openbravo ERP proved to be quite a challenge.

An important lesson learnt here is to tackle one problem at a time. As often the ample discussion of possible problems led to more talking and less work. We overcame that obstacle by exploring and understanding only the code that was linked directly to the problem at hand instead of going through complex hierarchies etc.

Future Work

For anyone looking into further enhancements, implementing the following in Openbravo POS would be the best option:

Reports

Products bought by most valuable customers:

This report will provide a summary of all those products which are bought by the most valuable customers whereas most valuable customers are those customers who have highest points earned so far.(These points will also include points that customers have availed.)

Products bought by less valuable customers

This report provides the list of products which are bought by less valuable customers where as less valuable customers are those customers having less number of points and purchases.

Customer with highest loyalty points

This report provides the list of customers who have the highest loyalty points in their account.

Customer Sales before and after Issuance of Loyalty Card:

This report will provide a summary of sales for customers before and after the issuance of loyalty card. It will brief on monthly purchases that customers used to do before they were issued a card and after that.

Most valuable Deal:

This report will describe which deals are proven to be the most valuable. Which deals are more famous in customers and which they want to avail most of the time.

Most Valuable Customer:

This report will brief on customers that have been availing more deals and earning more points and hence been more valuable to the organization.

14. References

<http://wiki.openbravo.com>

<http://forge.openbravo.com>

15. Glossary

Admin: Administrator

FK: Foreign Key

PK: Primary Key

POS: Point Of Sale