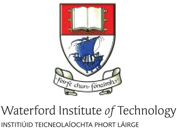
## UML Modelling Assignment

**Topic:**

* A text description in English of your chosen topic.
* The UML Class diagram.
* Data dictionary definitions for each class, attribute and association.

Submitted To: Brendan Jackman

Submitted By: Dmitrij Jakovlev 20044623, Software Engineering 2014, third semester.



## Book Web Shop

## GOAL:

## Description of business process

## UML modelling of business process (diagrams)

## Class diagrams

## Data dictionary definition

## Conclusion

## INTRODUCTION:

## Aim of that assignment is learning of UML method to create and show visual modelling using a Visual Paradigm to create book web shop.

## Tasks:

## Describe process of chosen topic

## Build up using UML language in diagram classes

## Build up class diagram

## Description of business process in plain text.

## Using Internet Book Shop customer log in on site where is situated assortment of production with additional information or detailed information. Every book has ID in catalogue, author, price, year of publishing, name of publisher. Also buyer has opportunities about type of payment, shipping, contact information and details of administration.

## Customer to make an order firstly should make authorization and only after he will be able to log in on his personal page. After that customer looking for book which he wants to buy. He will able to use search engine and looking for by author, publisher and name of book. Customer adds the book in to the cart for the next process to buy her or few books. When he did collect books in cart he has option to remove or change books, or make an order.

## For the purchase of books buyer indicates the delivery address, card number to provide discounts (if it has a discount card of the store), specifies the method of payment and sends the order to the manager of sales.

## The payment may have few forms:

## Through e-Wallet like WMZ, WME, WMG, WMX etc.

## Through bank

## By cash when delivered ordered books by courier

After process of checking payment manager prepared an invoice which keep that kind of information:

* Date of order
* Price
* Number of order
* Address
* Login of customer
* FName, LName of customer
* Type of payment

After manager send invoice to the warehouse for delivery to the customer where staff checks the books on the warehouse and pack the order. Next step include process to send order to courier and creating an invoice where:

* Date of delivery of goods
* Name of courier company
* Quantity of books
* Name of books

In common, process includes itself:

* Acceptance of the goods
* Dispatch or allocation of goods, sale
* Revaluation of goods
* Write-off goods

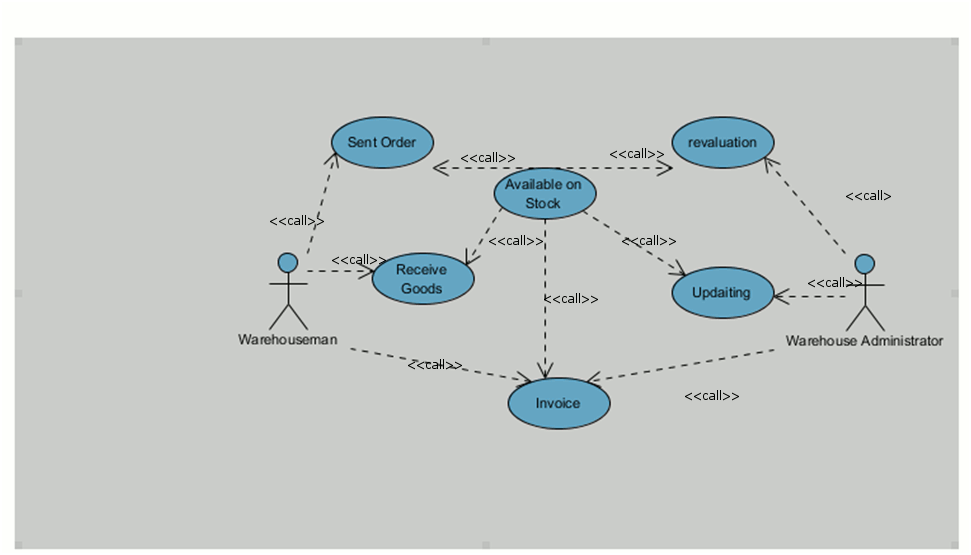
All invoices are sent from the warehouse manager of sales.

Sales Manager receives invoices from a warehouse and updates the catalogue of books in the store's website to book on the internet storefront matched books in stock.

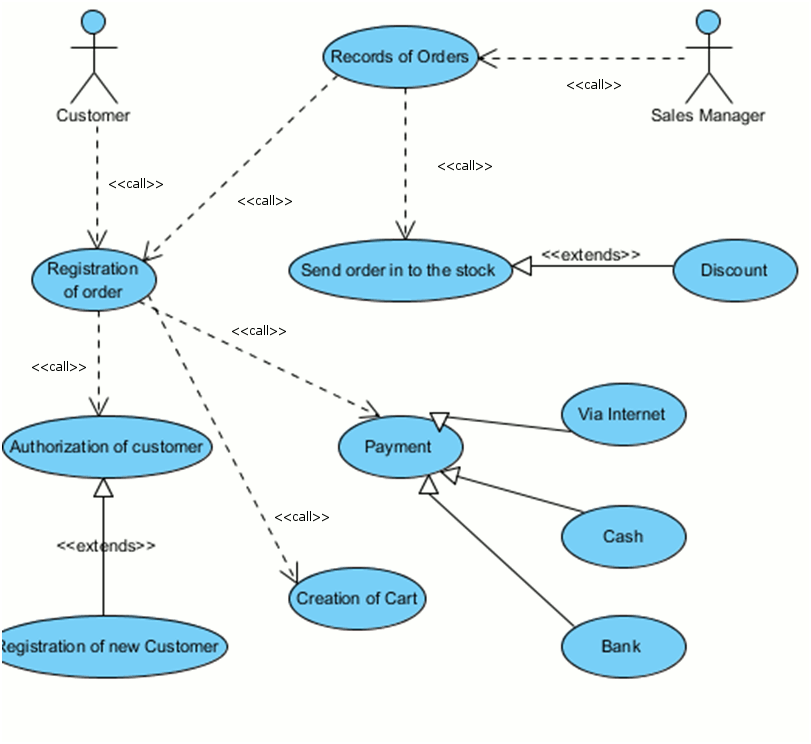
After sales manager:

* Adds information about the books
* Update information about books
* Removes the book from the catalogue

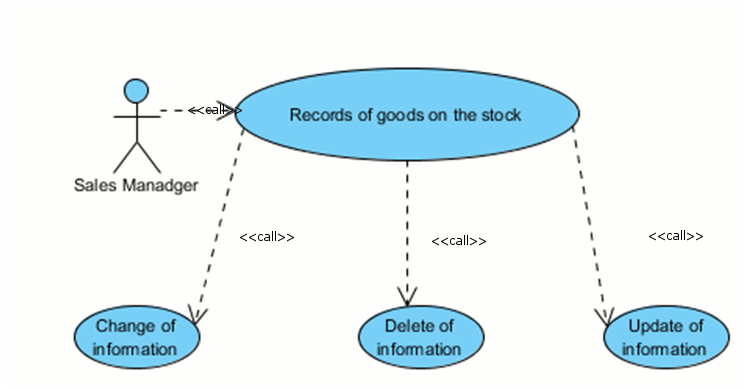
**UML case diagram of full processing using Visual Paradigm.**

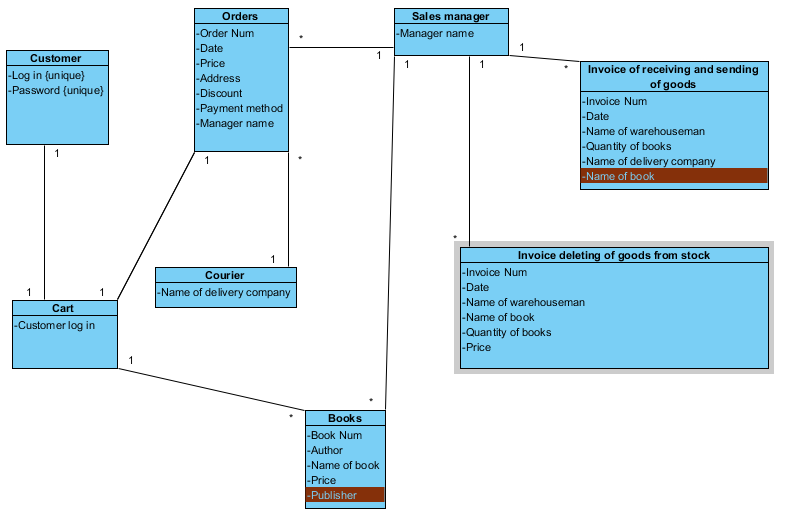


**UML, records of orders on the stock.**

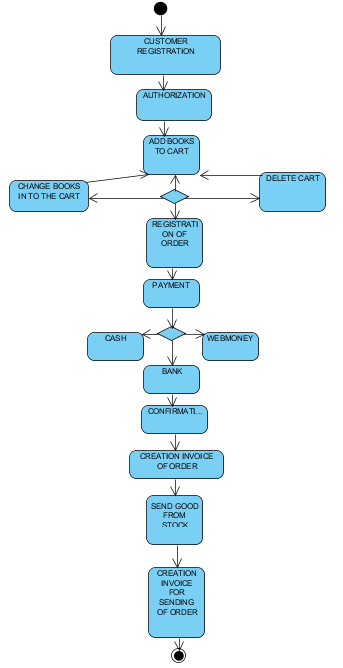


**UML of records of goods on site store**

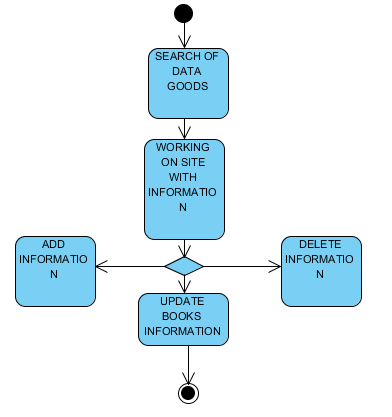


**Class diagram of invoices.**

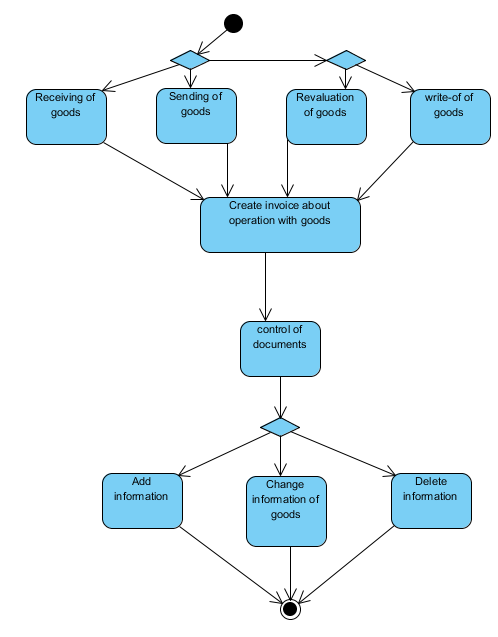
**Diagram of activity. (Records of orders)**

****

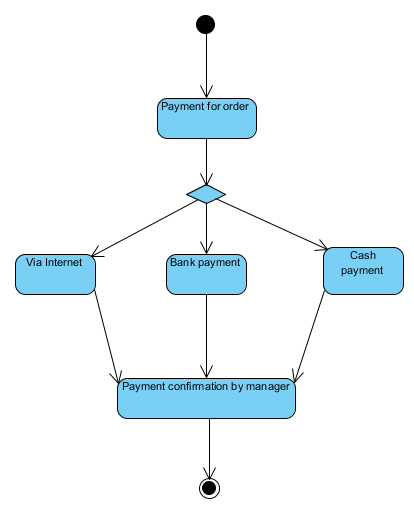
**Diagram of activity. (Records of orders on the stock).**

****

**Diagram of activity. (Records of orders on the site).**



**Diagram of activity. (Payment for order)**



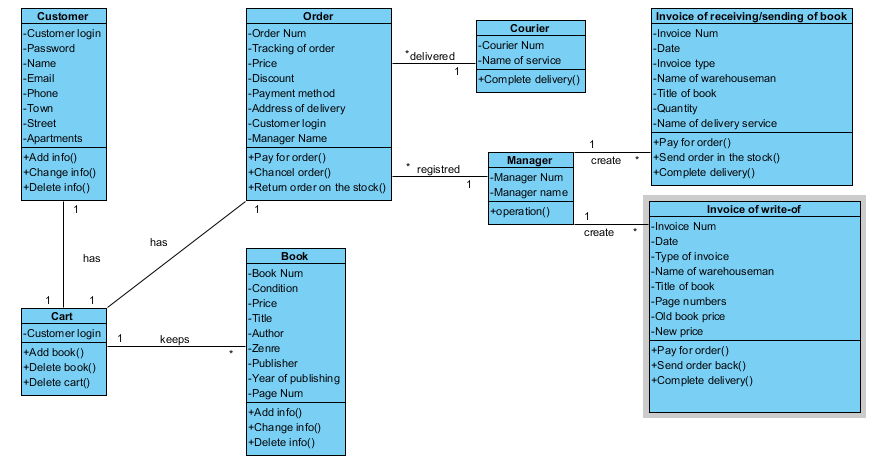
## State diagram (UML), availability of book.

## 5 pic.png

## Data dictionary definition.

**Actors of system and their actions.**

* Customer – potential buyer
* Sales Manager – working with orders and books
* Storekeeper – receive and send the books from warehouse
* Warehouse administrator – checking and writing off books, working with documents and sending to sales manager for updating system.
* Courier – delivery an order associated with manager



## Now create table for each class which has attributes types of data:

## Customer:

|  |  |  |  |
| --- | --- | --- | --- |
| Data | Description | Type | Constraint |
| Customer login | Uniquely indentifies a CustomerID | Varhar(16) PK | Not Null or Empty, Check, Unique |
| Password | Needed to login on web page | Varchar(15) | Cannot be null or empty, must contains any characters but not at least 6 |
| Name | fName and lName of customer | Text(25), Alternate Key | Not Null or Empty |
| Email | Customer email address | Text | Must be of type assigned to the purchasing company... |
| Telephone | For contacts | Varchar(10) | Must be no more than 10 digits. Check |
| Town | Address for delivery an order | Char, Alternate Key | Not Null or Empty |
| Street | Address for delivery an order | Text, Alternate Key | Not Null or Empty |
| House | Address for delivery an order | Text Alternate Key | Not Null or Empty |
| Flat | Address for delivery an order | Text, Alternate Key | Not Null or Empty |

## Cart:

|  |  |  |  |
| --- | --- | --- | --- |
| Data | Description | Type | Constraint |
| Customer login | Uniquely indentifies a CustomerID | Varchar(16)PK | Not Null or Empty, Unique, Check |

## Order:

|  |  |  |  |
| --- | --- | --- | --- |
| Data | Description | Type | Constraint |
| Order Num | Uniquely indentifies a OrderNum | Varchar(25)PK | Not Null or Empty, Auto increment, Check, Unique |
| Tracking of Order | Customer able to track the status of his order via Internet using orderNum. | Varchar | Null |
| Price | The price of order | Numeric | Not Null or Empty and price>0, positive |
| Discount | Discount for books if customer bought before or | Numeric | Numeric, Check(price>discounted\_price) |
| Payment Method | The customer able to choose any payment which convenient for him. | Text | Not Null or Empty |
| Address of delivery:Town, street, House, Flat | Customer address for delivery of order(s). That address should know delivery service. | Alternate Key | Not Null or Empty |
| Customer login | Uniquely indentifies a CustomerID | Varchar(16) | Not Null or Empty |
| Manager Name | To contact with manager | Text(25)(FK) | ManagerNum, UNIQUE(Num) |
| CourierNum | Unique number of courier service | Varchar(20)FK | Not Null or Empty, Check |

## Book:

|  |  |  |  |
| --- | --- | --- | --- |
| Data | Description | Type | Constraint |
| BookNum | Each book has ID to easy search by that parameter. | Varchar(20)PK | Not Null or Empty, autoincrement(each time bookNum ++ automatically) |
| Condition | May be records about condition of book, secondhand or new | Text | Null |
| Price | The price of order | Numeric | Not Null or Empty and price>0, positive |
| Title | The title of book | Varchar(100) | Not Null or Empty |
| Author | Name and Surname of writer | Char(50) | Not Null or Empty |
| Genre | Type of genre(25) | Int | Not Null or Empty |
| Publisher | Who publish that book, keep information about name of company, address etc. | Char(20) | Null |
| Year of publishing | When that book was printing and first time appear in the sale market | Numeric | Null |
| Page Numbers | Shows how many pages in the book | Numeric | Null |
| Customer login |  | Varhar(16) FK | Not Null or Empty |

## Manager:

|  |  |  |  |
| --- | --- | --- | --- |
| Data | Description | Type | Constraint |
| ManagerNum | Unique ID of manager | Int(20)PK | Not Null or Empty, Auto increment, Check, Unique |
| Manager Name | Name of manager | Char | Not Null or Empty |

## Courier:

|  |  |  |  |
| --- | --- | --- | --- |
| Data | Description | Type | Constraint |
| CourierNum | Unique number of courier service | Varchar(20)PK | Not Null or Empty, Auto increment, Check, Unique |
| Name of Service | Name of delivery company | Text | Not Null or Empty |

## Invoice of receiving/sending of book:

|  |  |  |  |
| --- | --- | --- | --- |
| Data | Description | Type | Constraint |
| InvoiceNum | Inditification of invoice in database | Int(20)PK | Auto\_Increment, Not Null or Empty, Check |
| Date | When invoice was send and receive | Date, format dd/mm/yy | Not Null or Empty, Check |
| Invoice Type | Shipping Invoice | Text | Not Null or Empty |
| Name of warehouseman | Name of stuff who  realise goods from warehouse | Text(25) | Not Null or Empty |
| Title of book | Title of ordered books | Varchar(100) | Not Null or Empty |
| Name of delivery service | Name of company who make delivery | Text | Not Null or Empty |
| ManagerNum | ID of manager | Int(20)FK | Not Null or Empty, Auto increment, Check, Unique |

## Invoice of write-off:

|  |  |  |  |
| --- | --- | --- | --- |
| Data | Description | Type | Constraint |
| InvoiceNum | Inditification of invoice in database | Int(20)PK | Auto\_Increment, Not Null or Empty, Check |
| Date | When invoice was send and receive | Date, format dd/mm/yy | Not Null or Empty, Check |
| Invoice Type | Write-off ivoice for supplier about changing the price for goods | Text | Not Null or Empty, Check |
| Name of warehouseman | Name of stuff who  realise goods from warehouse | Text(25) | Not Null or Empty |
| Title | The title of book | Varchar(100) | Not Null or Empty |
| Page Numbers | Shows how many pages in the book | Numeric | Null |
| Old book price | Previous price of book on the stock | Numeric | Not Null or Empty and price>0, positive |
| New book price | When no more, seller negotiates with supplier probably about a new less price. | Numeric | Not Null or Empty and new price<old price, Check, positive |
| ManagerNum | ID of manager | Int(20)FK | Not Null or Empty, Auto increment, Check, Unique |

## CONCLUSION

## I`m used system of book webshop where customer comes to the web store with range of many books with information about:

## Book number

## Author

## Title of book

## Publisher

## Year of publication

## Price

## The site also posted additional information about payment methods, delivery of purchase and contact information with sales manager. To make a purchase client logs onto the site - introduces a unique user name and password to enter in to the private webpage. If the customer does not have a login, he goes through the registration process on the website to get access.

## After that the customer may view range of books, flipping through the pages of the catalogue. He can also use the search by:

## Author

## The title of the book

## Publisher

## Customer defers certain book, putting it in the cart (a specific set of books for future purchase). If customer wishes to delete or change the chosen books he will able to do this by options how:

## Change the set of books remove the numbers of books

## Or remove all inside the cart

## For the purchase books customer should indicates the delivery address, card number to provide discounts (if he has a discount card of this store), specifies the method of payment and sends the order to the manager of sales.

## Payment for orders can be made:

## Via the Internet wallet, said online shop

## Through the bank (agreed with the manager)

## By cash on delivery order the book (via a delivery service or courier)

## After verification of payment manager prepares an invoice order, which includes:

## The date of the order

## The purchase price with the discount

## Number of books of the order

## Delivery address

## Customer login

## Full name of the sales manager

## Method of payment

## Further sales manager sends an invoice to the warehouse for subsequent delivery to the customer.

## In stock keeps books for sale. Warehouseman receives from the manager invoice customer orders and generates a set of books for delivery to customers.

## Ordered books are released from the warehouse storekeeper organization or courier delivery store.

## In general, on the stock performed next operations :

## Receiving of the goods

## Sending of the goods

## Revaluation of goods

## Write-off goods

## All invoices from the stock send to the manager of sales.

## Sales manager receives invoices from the stock and updates the catalogue of books in the store's web site. Further manager of sales:

## Adds information about the books

## Changes information about books

## Removes the book from the catalogue