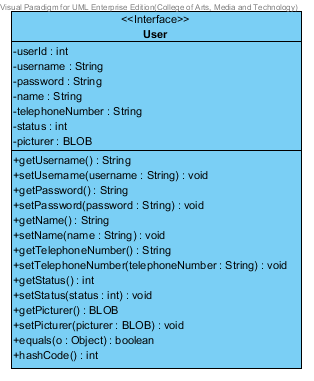
***4. METHOD DESIGN***

**4.1 Entity**

**4.1.1 MD-01: User**

**Class Name: User**

****

**Entities**

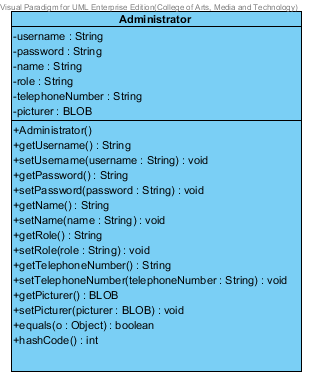
|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **NAME** | **DESCRIPTION** | **REMARK** |
| 1 | userId | This is an id for identify the object of User Class. | **TYPE :** Integer |
| 2 | username | This is a username of user. | **TYPE :** String |
| 3 | password | This is a password of user. | **TYPE :** String |
| 4 | name | This is a Full-name of user. | **TYPE :** String |
| 5 | telephoneNumber | This is a telephone number of user. | **TYPE :** String |
| 6 | status | This is a status to separate the type of User Object. | **TYPE :** Integer |
| 7 | Picture | This is a picture of user. | **TYPE :** BLOB |

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **NAME** | **DESCRIPTION** | **REMARK** |
| 1 | equals() |  | **Return Type :** Boolean  **Parameters :** Object |
| 2 | hashCode() |  | **Return Type :** Integer |

**Method**

**4.1.2 MD-02: Administrator**

**Class Name: Administrator**

****

**Entities**

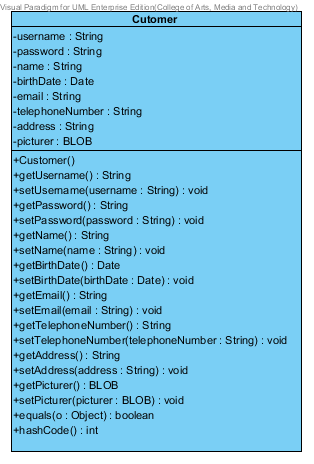
|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **NAME** | **DESCRIPTION** | **REMARK** |
| 1 | username | This is a username of Administrator. | **TYPE :** String |
| 2 | password | This is a password of Administrator. | **TYPE :** String |
| 3 | name | This is a Full-name of Administrator. | **TYPE :** String |
| 4 | role | This is a role that Administrator responsibility. | **TYPE :** String |
| 5 | telephoneNumber | This is a telephone number of Administrator. | **TYPE :** String |
| 6 | Picture | This is a picture of Administrator. | **TYPE :** BLOB |

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **NAME** | **DESCRIPTION** | **REMARK** |
| 1 | equals() |  | **Return Type :** Boolean  **Parameters :** Object |
| 2 | hashCode() |  | **Return Type :** Integer |

**Method**

**4.1.3 MD-03: Customer**

**Class Name: Customer**

****

**Entities**

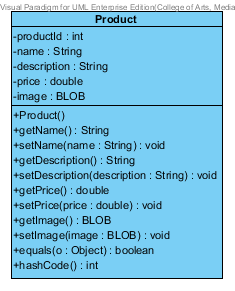
|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **NAME** | **DESCRIPTION** | **REMARK** |
| 1 | username | This is a username of customer. | **TYPE :** String |
| 2 | password | This is a password of customer. | **TYPE :** String |
| 3 | name | This is a Full-name of customer. | **TYPE :** String |
| 4 | birthDate | This is a birth date of customer | **TYPE :** String |
| 5 | email | This is an email of customer. | **TYPE :** String |
| 6 | telephoneNumber | This is a telephone number of customer. | **TYPE :** String |
| 7 | address | This is an address of customer. | **TYPE :** String |
| 8 | Picture | This is a picture of customer. | **TYPE :** BLOB |

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **NAME** | **DESCRIPTION** | **REMARK** |
| 1 | equals() |  | **Return Type :** Boolean  **Parameters :** Object |
| 2 | hashCode() |  | **Return Type :** Integer |

**Method**

**4.1.4 MD-04: Product**

**Class Name: Product**

****

**Entities**

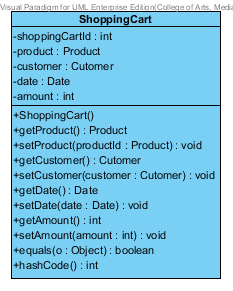
|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **NAME** | **DESCRIPTION** | **REMARK** |
| 1 | productId | This is an id for identify the object of Product Class. | **TYPE :** Integer |
| 2 | Name | This is a name of product. | **TYPE :** String |
| 3 | description | This is a description of product. | **TYPE :** String |
| 4 | Price | This is a price of product | **TYPE :** Double |
| 5 | Image | This is an image of product. | **TYPE :** BLOB |

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **NAME** | **DESCRIPTION** | **REMARK** |
| 1 | equals() |  | **Return Type :** Boolean  **Parameters :** Object |
| 2 | hashCode() |  | **Return Type :** Integer |

**Method**

**4.1.5 MD-05: Shopping Cart**

**Class Name: Shopping Cart**

****

**Entities**

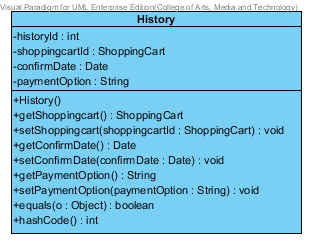
|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **NAME** | **DESCRIPTION** | **REMARK** |
| 1 | shoppingCartId | This is an id for identify the object of shopping Cart Class. | **TYPE :** Integer |
| 2 | product | This is an object to hold product information. | **TYPE :** Product |
| 3 | customer | This is an object to hold customer information. | **TYPE :** Customer |
| 4 | date | This is a date when customer is adding a product into The shopping Cart. | **TYPE :** Date |
| 5 | amount | This is an amount of product which is customer adding into the shopping Cart. | **TYPE :** Integer |

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **NAME** | **DESCRIPTION** | **REMARK** |
| 1 | equals() |  | **Return Type :** Boolean  **Parameters :** Object |
| 2 | hashCode() |  | **Return Type :** Integer |

**Method**

**4.1.6 MD-06: History**

**Class Name: History**

****

**Entities**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **NAME** | **DESCRIPTION** | **REMARK** |
| 1 | historyId | This is an id for identify the object of History Class. | **TYPE :** Integer |
| 2 | shoppingCartId | This is an object to hold shopping Cart information. | **TYPE :** ShoppingCart |
| 3 | confirmDate | This is a date when customer confirm to purchase a product. | **TYPE :** Date |
| 4 | paymentOption | This is a payment Option which consist of transfer, credit card and Paypal. | **TYPE :** String |

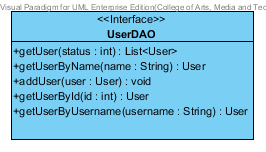
|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **NAME** | **DESCRIPTION** | **REMARK** |
| 1 | equals() |  | **Return Type :** Boolean |
| 2 | hashCode() |  | **Return Type :** Integer |

**Method**

**4.2 MVC Pattern of User Class**

**4.2.1 MD-07: UserDAO**

**Class Name: UserDAO**

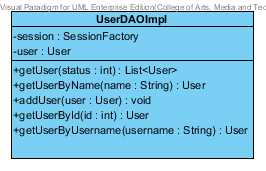
****

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **NAME** | **DESCRIPTION** | **REMARK** |
| 1 | getUser() | This method uses for getting a user. | **Return Type :** List<User>  **Parameters :** status |
| 2 | getUserByName() | This method uses for getting a user by using user’s name from database. | **Return Type :** User  **Parameters :** name |
| 3 | addUser() | This method uses for adding a user into a database. | **Return Type :** void  **Parameters :** user |
| 4 | getUserById() | This method uses for getting a user by using user’s id from database. | **Return Type :** User  **Parameters :** id |
| 5 | getUserByUsername() | This method uses for getting a user by using user’s username from database. | **Return Type :** User  **Parameters :** username |

**Method**

**4.2.2 MD-08: UserDAOImpl**

**Class Name: UserDAOImpl**

****

**Entities**

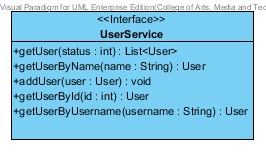
|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **NAME** | **DESCRIPTION** | **REMARK** |
| 1 | session | This is an object to hold sessionFactory. | **TYPE :** SessionFactory |
| 2 | user | This is an object to hold user’s information. | **TYPE :** User |

**Method**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **NAME** | **DESCRIPTION** | **REMARK** |
| 1 | getUser() | This method uses for getting a user. | **Return Type :** List<User>  **Parameters :** status |
| 2 | getUserByName() | This method uses for getting a user by using user’s name from database. | **Return Type :** User  **Parameters :** name |
| 3 | addUser() | This method uses for adding a user into a database. | **Return Type :** void  **Parameters :** user |
| 4 | getUserById() | This method uses for getting a user by using user’s id from database. | **Return Type :** User  **Parameters :** id |
| 5 | getUserByUsername() | This method uses for getting a user by using user’s username from database. | **Return Type :** User  **Parameters :** username |

**4.2.3 MD-09: UserService**

**Class Name: UserService**

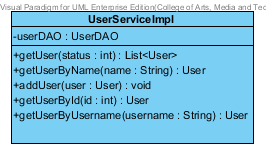
****

**Method**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **NAME** | **DESCRIPTION** | **REMARK** |
| 1 | getUser() | This method uses for getting a user. | **Return Type :** List<User>  **Parameters :** status |
| 2 | getUserByName() | This method uses for getting a user by using user’s name from database. | **Return Type :** User  **Parameters :** name |
| 3 | addUser() | This method uses for adding a user into a database. | **Return Type :** void  **Parameters :** user |
| 4 | getUserById() | This method uses for getting a user by using user’s id from database. | **Return Type :** User  **Parameters :** id |
| 5 | getUserByUsername() | This method uses for getting a user by using user’s username from database. | **Return Type :** User  **Parameters :** username |

**4.2.4 MD-10: UserServiceImpl**

**Class Name: UserServiceImpl**

****

**Entities**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **NAME** | **DESCRIPTION** | **REMARK** |
| 1 | userDAO | This is an object to hold userDAO Class. | **TYPE :** UserDAO |

**Method**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **NAME** | **DESCRIPTION** | **REMARK** |
| 1 | getUser() | This method uses for getting a user. | **Return Type :** List<User>  **Parameters :** status |
| 2 | getUserByName() | This method uses for getting a user by using user’s name from database. | **Return Type :** User  **Parameters :** name |
| 3 | addUser() | This method uses for adding a user into a database. | **Return Type :** void  **Parameters :** user |
| 4 | getUserById() | This method uses for getting a user by using user’s id from database. | **Return Type :** User  **Parameters :** id |
| 5 | getUserByUsername() | This method uses for getting a user by using user’s username from database. | **Return Type :** User  **Parameters :** username |