

|  |
| --- |
| **PROJECT ON**  **Warehouse Management** |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Developed by**  **Members:**   |  |  |  | | --- | --- | --- | | **No.** | **Student Name** | **Student ID** | | 1 | Tu Le Thanh | Student1341374 | | 2 | Loc Tran Tan | Student1352787 | | 3 | Huy Tran Thanh Hoang | Student1346977 | | 4 | Anh Vu Hoang | Student1347130 |  * **Class No.: C2110L** * **Start Date: 01st Jan 2023** * **End Date: 13rd Feb 2023** * **Name of the Coordinator: Phan Huu Tri** * **Date of Submission: 15th Feb 2023** |

|  |
| --- |
| **SYSTEM ANALYSIS** |

|  |
| --- |
| **System Summary:**   1. **Object:**   Develop a “Sale and Inventory Management System (application)” that users (contained: salesman, sales manager, inventory manager and administrator) can manage functions.   1. **Functions:**   Base on user’s role, they are allowed to use varieties of functions.   1. Sales person:  * Login and change password; * See status (“available” or “not”) of particular item of the inventory; * Enter purchased items and produce a bill with some information (id, date, time, item, quantity, price, etc); * Cancel the produced bill; * Cancel any purchased item; * Produce a receipt; * Inform the inventory manager and the sales manager.  1. Sales manager:  * Login and change password; * Check succeed and canceled bills, transactions, returned items of the day; * Check canceled bills; * Check exception reports from sales person and contact with inventory manager to correct those. * Generate sales-trend graphs.  1. Inventory manager:  * Login and change password; * Check inventory’s status and validate; * See the sales person’s reports; * Generate inventory reports and trends.  1. Administrator:  * Login and change password; * Add new users, items, category; * Change item’s price. |

|  |
| --- |
| **CLASS DIAGRAM** |

|  |
| --- |
|  |

|  |
| --- |
| **DATABASE DESIGN** |

|  |
| --- |
| **Database Name: warehouse\_management**  **Number of tables: 8**  **Tables Names:**   1. Category 2. Item 3. User 4. Warehouse 5. WarehouseInInvoiceDetails 6. OutInvoiceDetails 7. WarehouseBackInvoiceDetails 8. Feedback   **Schematic Diagram Of Database:**  Schematic diagram of the database depicting the primary and foreign key  relationships. |

|  |
| --- |
| **TABLE DESIGN** |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Database Name:**  **Category:**   |  |  |  |  | | --- | --- | --- | --- | | Field Name | Data type | Width | Description | | categoryID | varchar | 250 | Unique ID of each category | | CategoryName | varchar | 250 | Name of each category |   **Item**   |  |  |  |  | | --- | --- | --- | --- | | Field Name | Data type | Width | Description | | itemID | varchar | 250 | Unique id of each item | | itemName | varchar | 250 | Name of the item | | categoryID | varchar | 250 | The category id that item belongs to | | status | tinyint | 1 | Status of item in the inventory (“available” or “not”) | | price | double |  | Price of item | | expiredDate | Date |  | Expired date of item | | image | longblod |  | A thumbnail of item | | review | varchar | 250 | Some reviews of item | | unit | varchar | 250 | Unit in which the item is measured (can be “gram”, “litre”, “oz”, “bag”, etc) |   **User**   |  |  |  |  | | --- | --- | --- | --- | | Field Name | Data type | Width | Description | | userID | varchar | 250 | Unique id of each user | | userName | varchar | 250 | Name of user | | role | int | 11 | Role of user (aministrator / sales management / inventory management or sales person) | | joinDate | date |  | The date that user join the system | | leaveDate | date |  | The date that user leave the system | | shift | varchar | 250 | Morning / Afternoon / Evening | | password | varchar | 250 | Password to login of user’s account |   **Warehouse**   |  |  |  |  | | --- | --- | --- | --- | | Field Name | Data type | Width | Description | | id | int | 11 | Unique id of each warehouse | | itemID | varchar | 250 | Link to item id | | itemName | varchar | 250 | Link to name of item | | dateAdded | Date |  | Added date of each warehouse | | status | tinyint | 1 | Status of item in each warehouse (“available” or “not”) | | totalInventory | double |  | Total of item in each warehouse |   **WarehouseInInvoiceDetails**   |  |  |  |  | | --- | --- | --- | --- | | Field Name | Data type | Width | Description | | id | int | 11 | Unique id of each warehouse imported item | | itemID | varchar | 250 | Link to item id | | quantity | int | 11 | Imported quantity of item | | price | double |  | Imported price of item | | total | double |  | Total imported value of item | | dateIn | date |  | The date that item imported into the warehouse | | manufactureID | varchar | 250 | The manufactured id of the item | | unit | varchar | 250 | Unit in which the item is measured (can be “gram”, “litre”, “oz”, “bag”, etc) | | userID | varchar | 250 | Id of the inventory manager | | status | tinyint | 1 | Status of the imported item |   **OutInvoiceDetails**   |  |  |  |  | | --- | --- | --- | --- | | Field Name | Data type | Width | Description | | outInvoiceDetailsID | int | 11 | Unique id of purchased invoice | | itemID | varchar | 250 | Link to item id | | itemName | varchar | 250 | Link to item’s name | | quantity | int | 11 | Purchased quantity of item | | price | double |  | Listing price of item | | Total | double |  | Total purchased value of invoice | | status | tinyint | 1 | Status of the purchase invoice | | payment | varchar | 250 | Kind of payment (online / transfer / credit card / cash) | | created | date |  | The date that purchased invoice produced | | unit | varchar | 250 | Unit in which the item is measured (can be “gram”, “litre”, “oz”, “bag”, etc) |   **WarehouseBackInvoiceDetails**   |  |  |  |  | | --- | --- | --- | --- | | Field Name | Data type | Width | Description | | id | int | 11 | Unique id of each warehouse returned item | | itemID | varchar | 250 | Link to item id | | quantity | int | 11 | quantity of returned item | | feedback | varchar | 250 | Feedback of customer (returning reason) | | date | date |  | The date that returned invoice produced | | userID | varchar | 250 | Id of sales person who produced the returned invoice |   **Feedback**   |  |  |  |  | | --- | --- | --- | --- | | Field Name | Data type | Width | Description | | feedbackID | int | 11 | Unique id of each feedback from user | | feedback | varchar | 300 | Content of feedback | | warehouseItemID | varchar | 300 | The id of warehouse’s item | | Date | date |  | The date when feedback produced | | typeFeedback | varchar | 300 | Type of feedback (positive or nagative) | | userID | varchar | 250 | Id of sales person who produced feedback | |

|  |
| --- |
| **VALIDATIONS PERFORMED** |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Database Name:**  **Category**   |  |  | | --- | --- | | Validation Required | Method Used For Validation | | Id must be unique | Primary key constraint | | Name | NOT NULL |   **Item**   |  |  | | --- | --- | | Validation Required | Method Used For Validation | | Id must be unique | Primary key constraint | | ItemName, Status, price, expiredDate, unit | NOT NULL | | categoryID | Foreign\_key |   **User**   |  |  | | --- | --- | | Validation Required | Method Used For Validation | | Id must be unique | Primary key constraint | | userName, password, role | NOT NULL |   **Warehouse**   |  |  | | --- | --- | | Validation Required | Method Used For Validation | | Id must be unique | Primary key constraint, auto\_increment | | itemID, itemName | Foreign\_key | | dateAdd, status, totalInventory | NOT NULL |   **WarehouseInInvoiceDetails**   |  |  | | --- | --- | | Validation Required | Method Used For Validation | | Id must be unique | Primary key constraint, auto\_increment | | itemID, userID | Foreign\_key | | Quantity, price, total, dateIn, unit, status | NOT NULL | | manufacturedID | Can be NULL |   **OutInvoiceDetails**   |  |  | | --- | --- | | Validation Required | Method Used For Validation | | Id must be unique | Primary key constraint, auto\_increment | | itemID, itemName | Foreign\_key | | Quantity, price, total, created, unit, status, payment | NOT NULL |   **WarehouseBackInvoiceDetails**   |  |  | | --- | --- | | Validation Required | Method Used For Validation | | Id must be unique | Primary key constraint, auto\_increment | | itemID, userID | Foreign\_key | | Quantity, feedback, date | NOT NULL |   **Feedback**   |  |  | | --- | --- | | Validation Required | Method Used For Validation | | Id must be unique | Primary key constraint, auto\_increment | | feedback | NOT NULL | |

|  |
| --- |
| **DATA FLOW DIAGRAM** |

|  |
| --- |
| **Data Flow Diagram representing the control flow of Project Case Study.** |

|  |
| --- |
| **(WEB/WINDOWS) FORMS OUTLINE** |

|  |  |
| --- | --- |
| **Forms Name** | **Description** |
|  | Login form is cleared, the password field is encoded to protect the user’s information, the small logos are designed to identify the label |
|  | The Home interface contains a menubar with accoun function. Users can change their password or logout if they have done works |
|  | The password management function is designed for users to change their password if necessary. A massage window will appear if the function is succeed or failed (in case the two passwords does not match) |
|  | This is the sales person interface with four main function |
|  | The function is for searching item purpose, a table with items’s information will appear when typing keywords, “refresh” button is used to clear the search field |
|  | The function is for checking inventory’ status whether item is available or not |
|  | The function is for producing purchased invoice |
|  | The function is for checking invoice’s information. The sales person can cancel the purchased invoice or return item base on customer’s demand |
|  | The function is for giving feedback or request to the inventory management |
|  | The function is for returning item and linked to the purchased invoice function, the feedback can be the reason that customer deny the item |
|  | This is the sales management interface with four main function |
|  | The function is for managing the invoice details |
|  | The function is for managing the transaction details |
|  | The function is for managing the canceled invoice details |
|  | The function is for managing the returned item details |
|  | This is the inventory management interface with five main function |
|  | The function is for managing in and out invoice of the warehouse |
|  | The function is for checking item\_trend |
|  | The function is for managing the imported item |
|  | The function is for checking storage ability of warehouse base on category |
|  | The function is for checking feedback |
|  | This is the admistrator interface with four main function |
|  | The fucntion is for creating new user. The first password is created by administrator, this will be changed on the first time user login the system |
|  | The function is for creating new item that has not had in the system |
|  | The function is for creating new category that has not had in the system |
|  | The function is for changing item’s price |

|  |
| --- |
| **CONFIGURATION** |

|  |
| --- |
| **Software & Technology Used:**   * **Languages: Java;** * **Interface: java Swing;** * **IDE: Eclipse;** * **Apache web server: Xampp** * **Database: MySQL (tool: phpMyAdmin)** |

|  |
| --- |
| **PROJECT TIMELINES** |

|  |  |  |
| --- | --- | --- |
| **S.No.** | **Task to be performed** | **Time** |
| **1** | Make database with all table related, Import information into the database;  Create login interface | 1 week |
| **2** | Create admin interface, sales person interface, sales management interface, inventory interface with main function | 1 week |
| **3** | Create additional function (feedback, returned item…) | 1 week |
| **4** | Link the information among the function and check the system | 1 week |

|  |
| --- |
| **INDIVIDUAL PROJECT SCHEDULE** |

|  |  |  |
| --- | --- | --- |
| **S.No.** | **Activity Description** | **Responsibility** |
| **1.** | Analyse the requirements of project, design inventory\_management’s interface and related functions (contact with sale\_person, check inventory, report inventory’s items and item\_trend), produce login function, create database | Tu Le Thanh |
| **2.** | Design sales\_person’s interface and related functions (check item status, produce and cancel purchased invoice, return item, make feedback), produce changing password function, create database | Loc Tran Tan |
| **3.** | Design sales\_management’s interface and related functions (check invoices, returned items and transactions, check exceptional report, check sales\_trend), create database | Huy Tran Thanh Hoang |
| **4.** | Design administrator’s interface and related functions (create new user, item, category, change the item’s price), create database | Anh Vu Hoang |