

E-Supermarket

Mini-Supermarket Management System

**Documentation**

| **C2008LM** | |
| --- | --- |
| **Group Member** | Nguyễn Bá Tuấn Anh  Phạm Vũ Tân  Lương Thanh Văn  Vũ Duy Long |
| **Instructor** | Đinh Ngọc Lân |

Hanoi, 03/2021

[**Problem Definition**](#_ey96krplcorj) **3**

[Problem Abstraction](#_3znysh7) 3

[The Current System](#_2et92p0) 3

[The Proposed System](#_tyjcwt) 3

[Boundaries of the System](#_3dy6vkm) 4

[Hardware and Software Requirements](#_1t3h5sf) 4

[Minimum Requirements](#_s7dh0f8cbyrf) 4

[Recommended Requirements](#_rlxa97nofntb) 4

**User's** [**Requirements Specification**](#_17dp8vu) **5**

[Users of the System](#_3rdcrjn) 5

[Cashier](#_5a0nruffc2jz) 5

[Warehouse Staff](#_pjvv9gdsacrg) 5

[Administrator](#_egxjxcq5b1jg) 6

[System functions](#_4imv4hhhrze3) 7

[**Architecture & Design of Project**](#_r97whg2i9eek) **21**

[Architecture](#_35nkun2) 21

[Data Flow Diagram](#_1ksv4uv) 21

[Flow Chart](#_44sinio) 22

[Login](#_covvsn599aa2) 22

[Search data](#_np9rlqb7g0t) 22

[Create data](#_vv38o8ds6j9i) 23

[Update data](#_2e1h86vwvevo) 24

[Delete data](#_trwlfgp1xq17) 25

[Entity Relationship Diagram](#_2jxsxqh) 27

[Database Structure](#_z337ya) 27

[Table Design](#_30v61yfrrfbh) 27

[Table categories: product’s categories.](#_2dem3at1q0zh) 27

[Table suppliers: Contains all suppliers.](#_aigg67nuk8fk) 27

[Table products: Contain all the products.](#_38n7betbsgun) 28

[Table brands : product’s brands](#_2hbta62atsrv) 28

[Table discounts:Table of discount detail](#_3szx762vg7ti) 29

[Table discounts\_product:Percent discount for each product](#_hkfncxdmf9s2) 29

[Table orders: Table of Orders](#_iemq6fz6xih3) 29

[Table order\_items: Table of Listing items on an Order](#_7glv09o68g1a) 30

[Table import\_products: Table of imported products](#_kko1tp2vdamv) 30

[Table employees: List of employees](#_hfsu06r29uyh) 31

[Table employee\_roles: Table of Roles](#_xf1axeka0l99) 32

[Table roles: Table of roles detail](#_nlt76p3a0ans) 32

[Table customers: Table of Loyal customers](#_dae37od93j3q) 32

[Database Diagram](#_o5ej0w1sqibw) 33

[Screen Design](#_m576hyrqmy9a) 33

[**Task Sheet**](#_wo9vrcoiymqj) **33**

[**Checklists**](#_1ci93xb) **34**

[Check List of Validation](#_3whwml4) 34

[Submission Checklist](#_2bn6wsx) 34

## Problem Definition

### Problem Abstraction

Stores are a sales model that has existed for centuries, and in recent decades, Supermarkets have become a very popular centralized sales model in many countries around the world. With a wide variety of goods and an extremely large number of products, it is a massive system that needs strict and accurate management of goods, orders and employees.

### The Current System

With the traditional sales method, the entire process of importing, selling, managing employees or statistics is manual and mostly recorded on paper. Leads to mistakes, asynchronous, inaccurate and very difficult to look up.

### The Proposed System

E-supermarket is a software solution to help manage tasks in supermarkets and grocery stores in the most optimal way. Apply advanced functions in all sales segments such as: Import - export goods, payment, invoice printing, warehouse management, employee management, order management, etc. Managers have the most intuitive and flexible view of the business situation in the store and can update information quickly and accurately.

* Increase sales efficiency compared to traditional ways
  + Optimize product searching, sales, and print invoices quickly
  + Automatic billing software, no need to spend time calculating or calculating as before
  + Save time and operating costs.
  + Search and update product information based on a search engine.
* Product and inventory management
  + Product and inventory management
  + Category management, grouping of goods by product category, brand, supplier, … Allowing to declare and manage unlimited items
  + Updating goods information instantly.
  + Automatically update the quantity of goods every time there is a transaction, avoiding data discrepancies and confusion between items
  + Through the software, it is possible to conduct inventory at any time.
  + Report the amount of inventory, sold-out goods to help store owners quickly import and export as well as take measures to push goods, avoiding inventory imbalance.
* Transparent and clarify revenue
  + Provide reports, summarize revenue statistics from time to time.
  + Provide detailed report analysis to help business owners make informed decisions.
* Loyalty customers management system
  + Save customer contact information
  + Statistics of orders and items that customers have purchased
  + Statistics of how many customers have spent
* Employee manager
  + Decentralize employees to divide work according to appropriate positions.
  + Supermarket / grocery owners can also monitor, monitor and know the working efficiency of each person.

### Boundaries of the System

* + - * Cashiers must be trained for order processing skills
      * Inventory staff must be trained for import product skills
      * The software is implemented for the Console interface, so there are a few limitations on display and interaction.

### Hardware and Software Requirements

#### Minimum Requirements

| Hardware | - Intel® Core™ i3 or higher  - 2 GB of RAM or higher  - SSD 120 GB |
| --- | --- |
| Software | - OS: Windows 10, Mac OS X, Linux  - Java Virtual Machine  - Java SE 8  - NetBeans 8.2  - DBMS: MySQL Server 5.7 |

#### Recommended Requirements

| Hardware | - Intel® Core™ i5 or higher  - 4 GB of RAM or higher  - SSD 240 GB |
| --- | --- |
| Software | - OS: Windows 10, Mac OS X, Linux  - Java Virtual Machine  - Java SE 8  - Apache NetBeans 12 LTS  - DBMS: MySQL Server 8.0 |

## Users Requirements Specification

### Users of the System

| **No** | **Users** | **Permission** |
| --- | --- | --- |
| 1 | Employees | ● Cashier: Manage Orders, Customers  ● Warehouse Staff: Manage Warehouse, Imported Product |
| 2 | Administrator | Full control of the System |

#### Cashier

* + - * Login
      * Logout
      * Change password
      * Product
        + Search a Product
        + View product details
        + Listing Product
        + View listing products
        + Search listing all products
        + View listing products details
      * Search Orders
        + Search order by date
        + Search order by ID
        + Search Customers
      * Show revenue of all order by date by Cashier
      * Export orders information to file Excel

#### Warehouse Staff

* + - * Login
      * Logout
      * Change password
      * Product
        + Product list
        + Search a Product by Name
        + Search product details
        + CRUD info product
      * Manage Import
        + Import product
        + Search import product history
      * Supplier
        + Supplier list
        + Search supplier by Name
        + Search supplier details
        + CRUD suppliers
      * Category
        + Search category list
        + Search category by Name
        + Search category details
        + CRUD categories
      * Brand
        + Search brand list
        + Search of brand by Name
        + CRUD brands
      * Order Search
        + Search orders by Date
        + Search orders by Customer Name
        + Search orders list
        + Search orders details
      * Statistical
        + Search product list out of stock
        + Search list of best-selling products of the month
        + Look for sales

#### Administrator

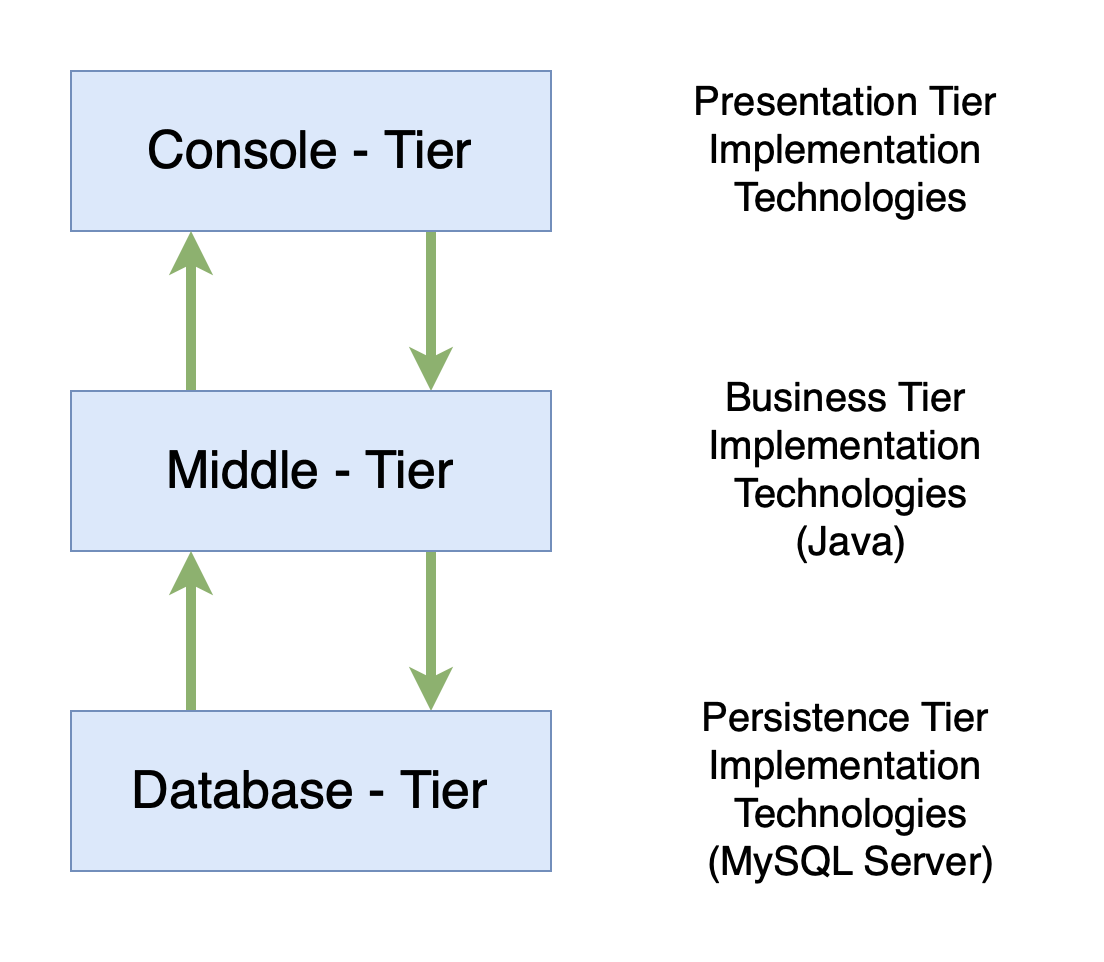
* + - * Login
      * Logout
      * Account
      * Change password
      * Update personal information
      * Manage Employees
        + Search Employees list
        + CRUD employee
        + Search of Employee by Name
        + Search Employee details
      * Manage Category
        + Search category list
        + Search category by Name
        + Search category details
        + CRUD categories
      * Discount
        + Search discount list
        + CRUD discount program
        + Search discount details
      * Supplier
        + Supplier list
        + Search supplier by Name
        + Search supplier details
        + CRUD suppliers
      * Product
        + Product list
        + Search product details
        + Search a Product by Name
        + Search product by category
      * Order Management
        + Search orders by Date
        + Search orders by customer’s Name
        + Search orders list
        + Search orders details
      * Statistical
        + Search product list out of stock
        + Search list of best-selling products of the month
      * Turnover
        + Today
        + Yesterday
        + in the last 7 days
        + in the last 30 days
        + in this month
        + in the past month
        + Customs

### System functions

| **No.** | **Function name** | **Users1** | **Detail** | |
| --- | --- | --- | --- | --- |
| 1 | Login | A  C  I | Description | Login to the system. |
| Input | Username and password. |
| Process | Check validation and permission in the appropriate table. |
| Output | • If username and password are true: display appropriate screen;  • If username or password is false: display error message. |
| 2 | Log out | A  C  I | Description | Log out of the system. |
| Input | Select Logout function at Menu. |
| Process | Delete the login session on the system |
| Output | Go to the Login menu after clearing data |
| 3 | Change information | A | Description | Change profile information, system login person |
| Input | New information |
| Process | Check validation of input data:  • If input data is valid, → Update new data into appropriate table;  • If input data is invalid → Don’t update new information into appropriate table |
| Output | • If change profile information successful: record in appropriate table has been updated and display successful message;  • If not: record has not been updated and display error message. |
| 4 | List of employee | A | Description | View all Employees in the employees table (including: search function) |
| Input | Enter a keyword if Search is selected |
| Process | Get all the employees in the table or search by approximate name if user selects Search |
| Output | • If there is data: display a list of employees;  • Otherwise: display no data message. |
| 5 | Add new employee | A | Description | Add new Employee’s information in table employees |
| Input | Employee information: name, address, phone, department, username, password |
| Process | Check validation of input data:  • If input data is valid → Insert data into table employees;  • If input data is invalid → Don’t insert information into table employees |
| Output | • If create new Employee successful: new record has been inserted into table employees and display successful message;  • If not: record has not been inserted and display error message. |
| 6 | Edit Employee Information | A | Description | Update new information for Employee’s |
| Input | New information of Employee: name, address, phone, department, password |
| Process | Check validation of input data:  • If input data is valid, → Update new data into table employees;  • If input data is invalid → Don’t update new information into table employees |
| Output | • If edit Employee information successful: record in table employees has been updated and display successful message;  • If not: record has not been updated and display error message. |
| 7 | Delete Employee | A | Description | Delete Employee out of table employees |
| Input | ID of Employee in table employees |
| Process | • If Employee has no constraint in another table, → Record in the table has been deleted;  • If Employee has constrained in another table → don’t delete |
| Output | • If delete successful: record has been deleted and display successful message;  • If not: record has not been deleted and display error message. |
| 8 | View Information Employee | A | Description | View all Employee’s information in table Employees |
| Input | ID of Employee in table employees |
| Process | • Check if employee ID exists in employees table → Get all information of Employee; |
| Output | • If found: Show all employee information;  • If not: display message not found. |
| 9 | List of Category | A  C  I | Description | View all Categories in the categories table (including: search function) |
| Input | Enter a keyword if Search is selected |
| Process | Get all the categories in the table or search by approximate name if user selects Search |
| Output | • If there is data: display a list of categories;  • Otherwise: display no data message. |
| 10 | Add new Category | I | Description | Add new Category’s information in table categories |
| Input | Category information: name |
| Process | Check validation of input data:  • If input data is valid → Insert data into table categories;  • If input data is invalid → Don’t insert information into table categories |
| Output | • If create new Category successful: new record has been inserted into table categories and display successful message;  • If not: record has not been inserted and display error message. |
| 11 | Edit Category Information | I | Description | Update new information for Category’s |
| Input | New information of Category: name |
| Process | Check validation of input data:  • If input data is valid, → Update new data into table categories;  • If input data is invalid → Don’t update new information into table categories |
| Output | • If edit Category information successful: record in table categories has been updated and display successful message;  • If not: record has not been updated and display error message. |
| 12 | Delete Category | I | Description | Delete Category out of table categories |
| Input | ID of Category in table categories |
| Process | • If Category has no constraint in another table, → Record in the table has been deleted;  • If Category has constrained in another table → don’t delete |
| Output | • If delete successful: record has been deleted and display successful message;  • If not: record has not been deleted and display error message. |
| 13 | View Information Category | A  C  I | Description | View all Category’s information in table categories |
| Input | ID of Category in table categories |
| Process | • Check if Category ID exists in categories table → Get all information of Category; |
| Output | • If found: Show all category information;  • If not: display message not found. |
| 14 | List of Discount | A | Description | View all Discounts in the discounts table (including: search function) |
| Input | Enter a keyword if Search is selected |
| Process | Get all the Discounts in the table or search by approximate name if user selects Search |
| Output | • If there is data: display a list of discounts;  • Otherwise: display no data message. |
| 15 | Add new Discount | A | Description | Add new Discount’s information in table discounts |
| Input | Discount information: name |
| Process | Check validation of input data:  • If input data is valid → Insert data into table discounts;  • If input data is invalid → Don’t insert information into table discounts |
| Output | • If create new Discount successful: new record has been inserted into table discounts and display successful message;  • If not: record has not been inserted and display error message. |
| 16 | Edit Discount Information | A | Description | Update new information for Discount |
| Input | New information of Discount: name |
| Process | Check validation of input data:  • If input data is valid, → Update new data into table discounts;  • If input data is invalid → Don’t update new information into table discounts |
| Output | • If edit Discount information successful: record in table discounts has been updated and display successful message;  • If not: record has not been updated and display error message. |
| 17 | Delete Discount | A | Description | Delete Discount out of table discounts |
| Input | ID of Discount in table discounts |
| Process | • If Discount has no constraint in another table, → Record in the table has been deleted;  • If Discount has constrained in another table → don’t delete |
| Output | • If delete successful: record has been deleted and display successful message;  • If not: record has not been deleted and display error message. |
| 18 | View Information Discount | A | Description | View all Discount information in table discounts |
| Input | ID of Discount in table discounts |
| Process | • Check if Discount ID exists in discounts table → Get all information of Discount; |
| Output | • If found: Show all Discount information;  • If not: display message not found. |
| 19 | List of Supplier | A | Description | View all Suppliers in the Suppliers table (including: search function) |
| Input | Enter a keyword if Search is selected |
| Process | Get all the Suppliers in the table or search by approximate name if user selects Search |
| Output | • If there is data: display a list of Suppliers;  • Otherwise: display no data message. |
| 20 | Add new Supplier | A  I | Description | Add new Supplier information in table Suppliers |
| Input | Supplier information: name |
| Process | Check validation of input data:  • If input data is valid → Insert data into table Suppliers;  • If input data is invalid → Don’t insert information into table Suppliers |
| Output | • If create new Supplier successful: new record has been inserted into table Suppliers and display successful message;  • If not: record has not been inserted and display error message. |
| 21 | Edit Supplier Information | A  I | Description | Update new information for Supplier |
| Input | New information of Supplier: name |
| Process | Check validation of input data:  • If input data is valid, → Update new data into table Suppliers;  • If input data is invalid → Don’t update new information into table Suppliers |
| Output | • If edit Supplier information successful: record in table Suppliers has been updated and display successful message;  • If not: record has not been updated and display error message. |
| 22 | Delete Supplier | A  I | Description | Delete Supplier out of table Suppliers |
| Input | ID of Supplier in table Suppliers |
| Process | • If Supplier has no constraint in another table, → Record in the table has been deleted;  • If Supplier has constrained in another table → don’t delete |
| Output | • If delete successful: record has been deleted and display successful message;  • If not: record has not been deleted and display error message. |
| 23 | View Information Supplier | A  I | Description | View all Supplier information in table Suppliers |
| Input | ID of Supplier in table Suppliers |
| Process | • Check if Supplier ID exists in discounts table → Get all information of Supplier; |
| Output | • If found: Show all Supplier information;  • If not: display message not found. |
| 24 | List of Product | A  C  I | Description | View all Products in the Products table (including: search function) |
| Input | Enter a keyword if Search is selected |
| Process | Get all the Products in the table or search by approximate name if user selects Search |
| Output | • If there is data: display a list of Products;  • Otherwise: display no data message. |
| 25 | Add new Product | I | Description | Add new Product information in table Products |
| Input | Product information: name |
| Process | Check validation of input data:  • If input data is valid → Insert data into table Product;  • If input data is invalid → Don’t insert information into table Products |
| Output | • If create new Product successful: new record has been inserted into table Products and display successful message;  • If not: record has not been inserted and display error message. |
| 26 | Edit Product Information | I | Description | Update new information for Product |
| Input | New information of Product: name |
| Process | Check validation of input data:  • If input data is valid, → Update new data into table Products;  • If input data is invalid → Don’t update new information into table Products |
| Output | • If edit Product information successful: record in table Products has been updated and display successful message;  • If not: record has not been updated and display error message. |
| 27 | Delete Product | I | Description | Delete Product out of table Products |
| Input | ID of Product in table Products |
| Process | • If Product has no constraint in another table, → Record in table has been deleted;  • If Product has constrained in another table → don’t delete |
| Output | • If delete successful: record has been deleted and display successful message;  • If not: record has not been deleted and display error message. |
| 28 | View Information Product | A  C  I | Description | View all Product information in table Products |
| Input | ID of Product in table Products |
| Process | • Check if Product ID exists in discounts table → Get all information of Product; |
| Output | • If found: Show all Product information;  • If not: display message not found. |
| 29 | List of Order | A  I | Description | View all Orders in the Orders table |
| Input |  |
| Process | Get all the Orders in the table |
| Output | • If there is data: display a list of Orders;  • Otherwise: display no data message. |
| 30 | Search Order by Customer | A  I | Description | Search for Orders by Customer in the Orders table |
| Input | Customer information: name |
| Process | Get all Orders in table by Customer name |
| Output | • If there is data: display a list of Orders;  • Otherwise: display no data message. |
| 31 | Search Order by date range | A  I | Description | Search for Orders by date range from day to day in the Orders table |
| Input | From date and to date |
| Process | Get all the Orders in the table from day to day |
| Output | • If there is data: display a list of Orders;  • Otherwise: display no data message. |
| 32 | Add new Order | I | Description | Add new Order information in table Orders |
| Input | Order information:   * Product ID * Product quantity   If there are old customers, will enter more customer information |
| Process | Check validation of input data:  • If input data is valid, → Insert data into table Order;  • If input data is invalid → Don’t insert information into table Orders |
| Output | • If create new Order successful: new record has been inserted into table Orders and display successful message;  • If not: record has not been inserted and display error message. |
| 33 | List of orders for today | C | Description | View all Today's Orders in the Orders table |
| Input |  |
| Process | Get all the Orders in the table for today |
| Output | • If there is data: display a list of Orders;  • Otherwise: display no data message. |
| 34 | View Information Order | A  C  I | Description | View all Order information in table Orders |
| Input | ID of Order in table Orders |
| Process | • Check if Order ID exists in discounts table → Get all information of Order; |
| Output | • If found: Show all Order information;  • If not: display message not found. |
| 35 | Statistics of products out of stock | A  C  I | Description | View all out of stock products in the Products table |
| Input |  |
| Process | • Find all products that are out of stock (with quantity = 0) |
| Output | • If found: Show all Product information;  • If not, display message no data. |
| 36 | Statistics of best-selling products | A  C  I | Description | View all the best-selling products of the month at the Order Details table |
| Input |  |
| Process | • Find all products with the most purchases for the month |
| Output | • If found: Show all Product information;  • If not, display message no data. |
| 37 | Revenue statistics | A  C  I | Description | View store sales statistics |
| Input | If viewing custom, enter date to date |
| Process | Statistics of the total amount of the number of products sold according to the following criteria:   1. Today 2. Yesterday 3. The last 7 days 4. The last 30 days 5. This month 6. Last month 7. Or enter a custom interval |
| Output | Display sales by each criterion |
| 38 | Revenue of today's orders generated by the cashier | C | Description | View revenue all orders for today of currently Cashier logged in |
| Input | Enter date to date |
| Process | Get all the Orders in the table for today with Cashier’s ID |
| Output | • If there is data: display a list of Orders;  • Otherwise: display no data message. |
| 39 | Search Customer | C | Description | View all Customers in the Customers table (including: search function) |
| Input | Enter a keyword: Customer name |
| Process | Search by approximate name |
| Output | • If there is data: display a list of Customers;  • Otherwise: display no data message. |
| 40 | List of Brand | I | Description | View all Brands in the Brands table (including: search function) |
| Input | Enter a keyword if Search is selected |
| Process | Get all the Brands in the table or search by approximate name if user selects Search |
| Output | • If there is data: display a list of Brands;  • Otherwise: display no data message. |
| 41 | Add new Brand | A  I | Description | Add new Brand information in table Brands |
| Input | Brands information: name |
| Process | Check validation of input data:  • If input data is valid → Insert data into table Brands;  • If input data is invalid → Don’t insert information into table Brands |
| Output | • If create new Brand successful: new record has been inserted into table Brands and display successful message;  • If not: record has not been inserted and display error message. |
| 42 | Edit Brand Information | A  I | Description | Update new information for Brand |
| Input | New information of Brand: name |
| Process | Check validation of input data:  • If input data is valid, → Update new data into table Brands;  • If input data is invalid → Don’t update new information into table Brands |
| Output | • If edit Brand information successful: record in table Brands has been updated and display successful message;  • If not: record has not been updated and display error message. |
| 43 | Delete Brand | A  I | Description | Delete Brand out of table Brands |
| Input | ID of Brand in table Brands |
| Process | • If Brand has no constraint in another table, → Record in the table has been deleted;  • If Brand has constrained in another table → don’t delete |
| Output | • If delete successful: record has been deleted and display successful message;  • If not: record has not been deleted and display error message. |
| 44 | View Information Brand | A  I | Description | View all Brand information in table Brands |
| Input | ID of Brand in table Brands |
| Process | • Check if Brand ID exists in discounts table → Get all information of Brand; |
| Output | • If found: Show all Brand information;  • If not: display message not found. |

## Architecture & Design of the Project

### Architecture

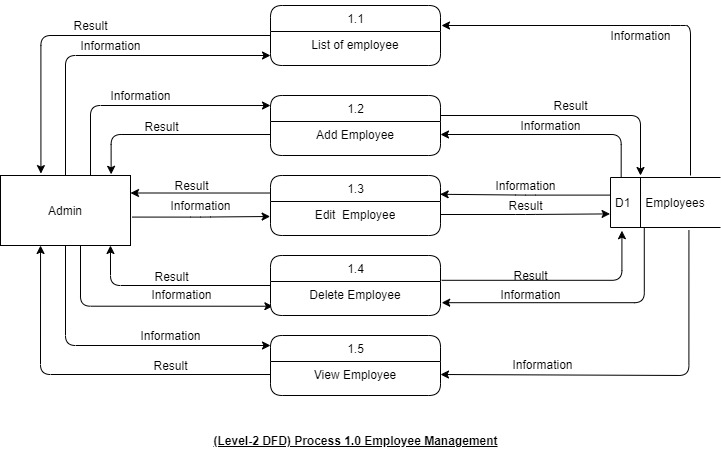
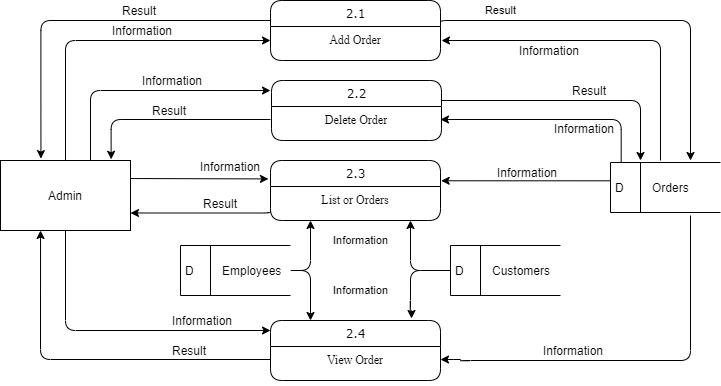


### Data Flow Diagram

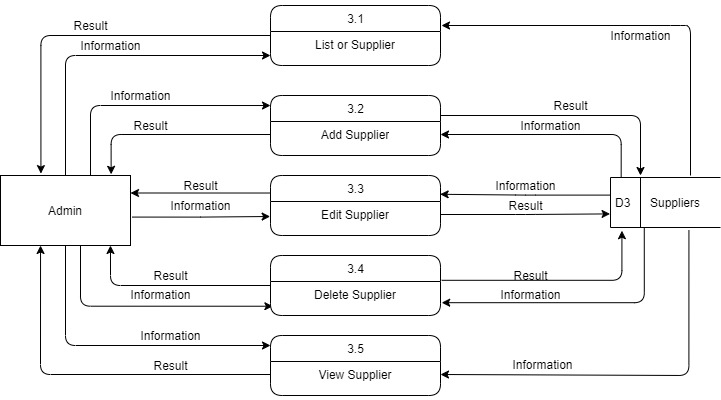
#### DFD level 0: Context Diagram

#### DFD level 1: All process

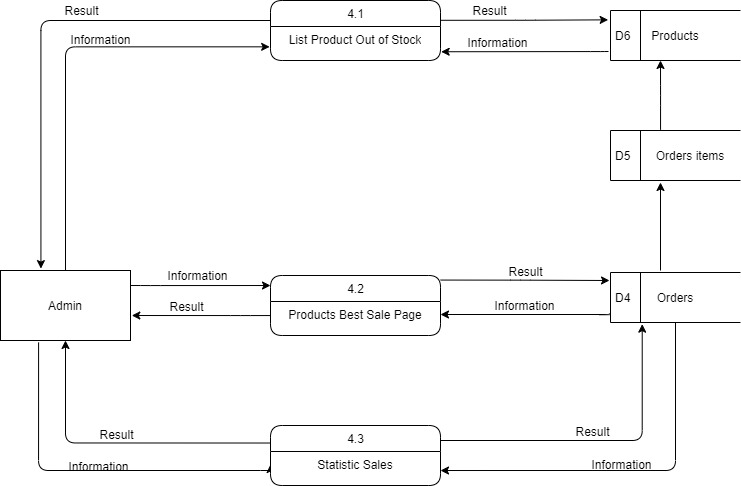
#### DFD level 2: Admin

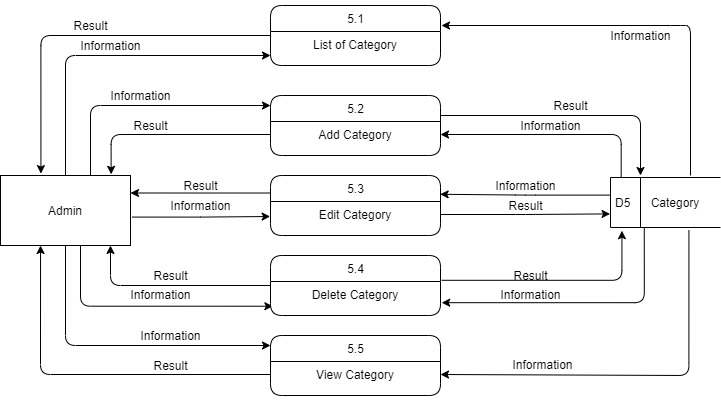
**(Level-2 DFD ADMIN) Process 2.0 Admin Order Management**



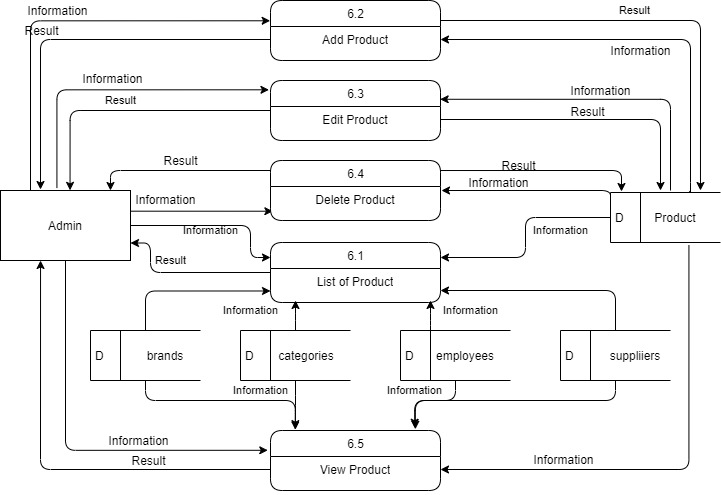
**(Level-2 DFD ADMIN) Process 3.0 Admin Supplier Management**

****

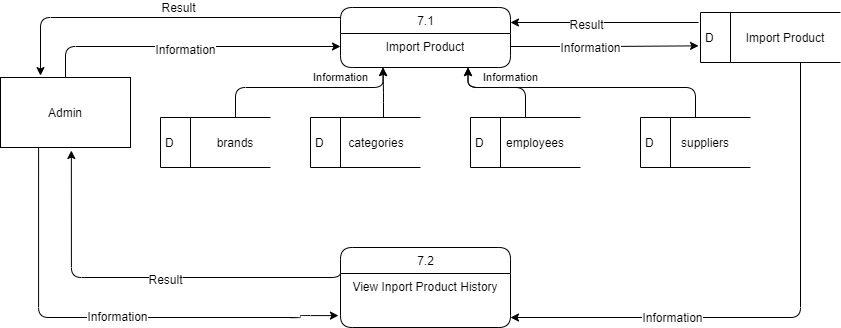
**(Level-2 DFD ADMIN) Process 4.0 Admin Statistics Management**



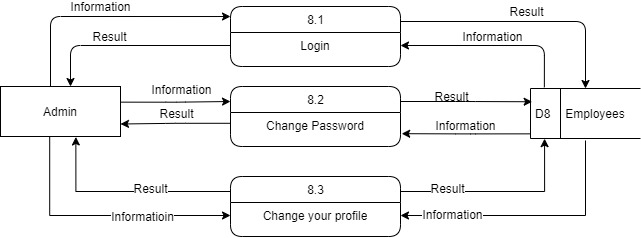
**(Level-2 DFD ADMIN) Process 5.0 Admin Category Management**



**(Level-2 DFD ADMIN) Process 6.0 Admin Product Management**

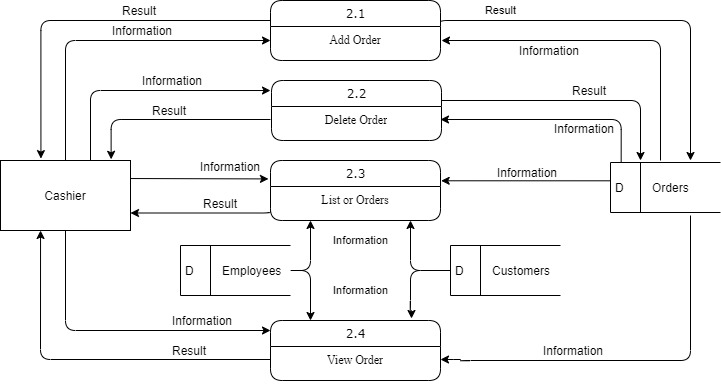


**(Level-2 DFD ADMIN) Process 7.0 Admin Warehouse Management**

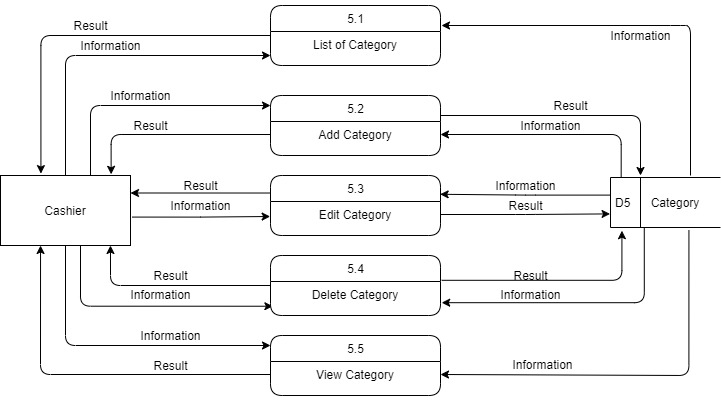


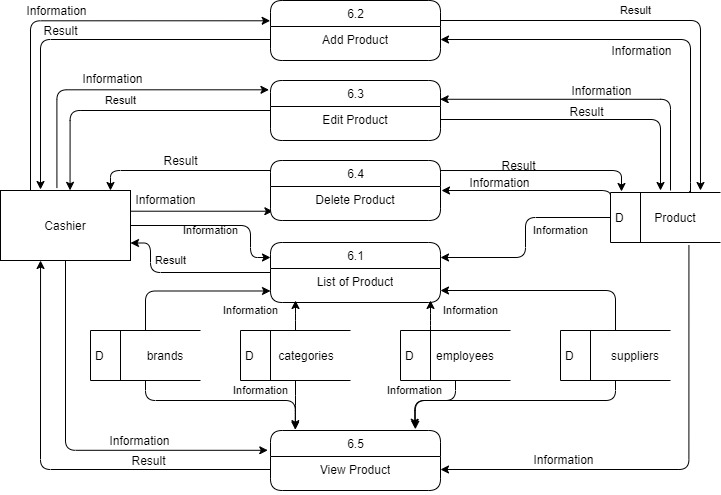
**(Level-2 DFD ADMIN) Process 8.0 Admin Login and Change Password**

#### DFD level 2: Cashier

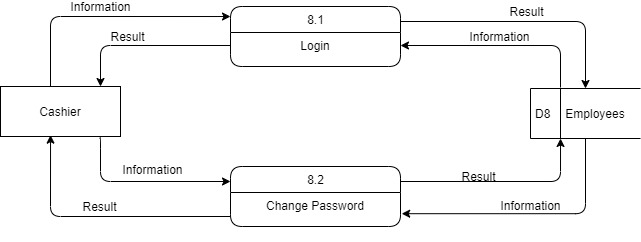


**(Level-2 DFD CASHIER) Process 2.0 Cashier Orders Management**

**(Level-2 DFD CASHIER) Process 5.0 Cashier Category Management**

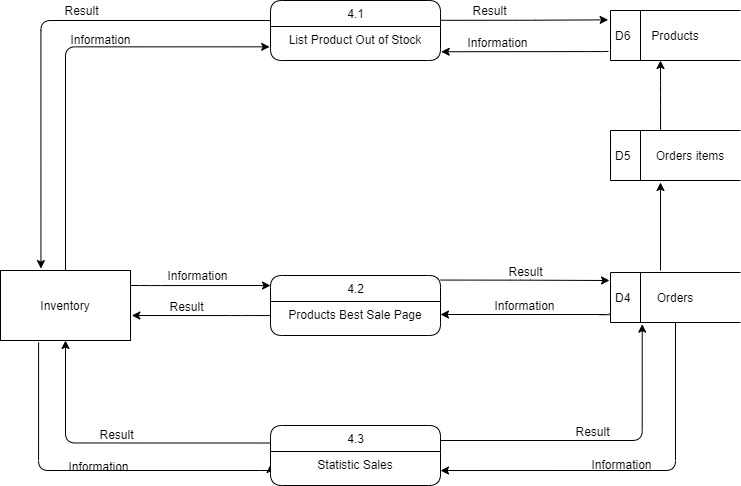


**(Level-2 DFD CASHIER) Process 6.0 Cashier Product Management**

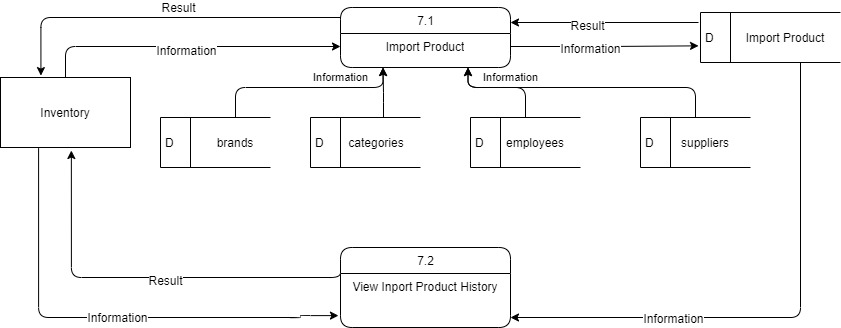


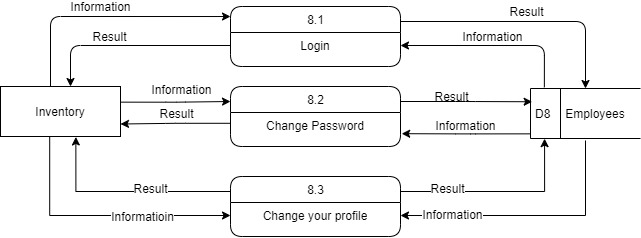
**(Level-2 DFD CASHIER) Process 8.0 Cashier Login and Change Password**

#### DFD level 2: Inventory



**(Level-2 DFD INVENTORY) Process 4.0 Inventory Statistics Management**

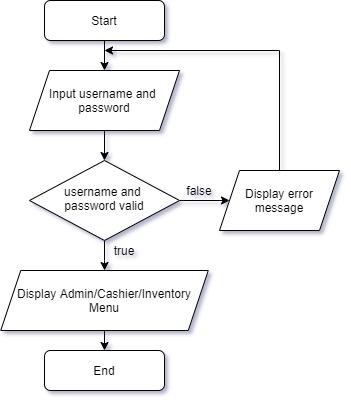
**(Level-2 DFD INVENTORY) Process 7.0 Inventory Warehouse Management**



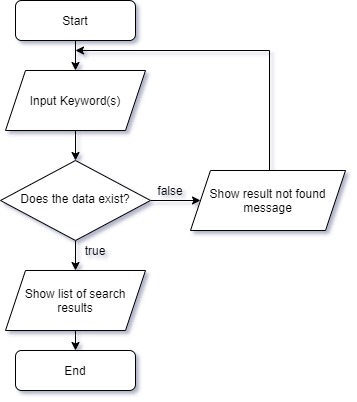
**(Level-2 DFD INVENTORY) Process 8.0 Inventory Login and Change Password**

### Flow Chart

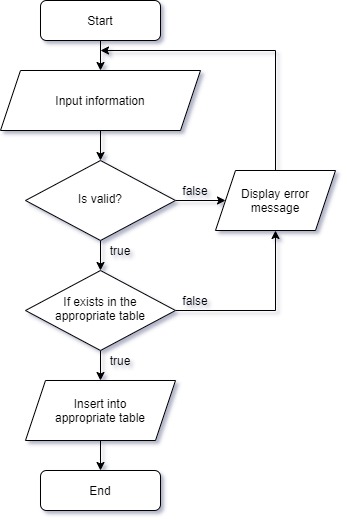
#### Login

******

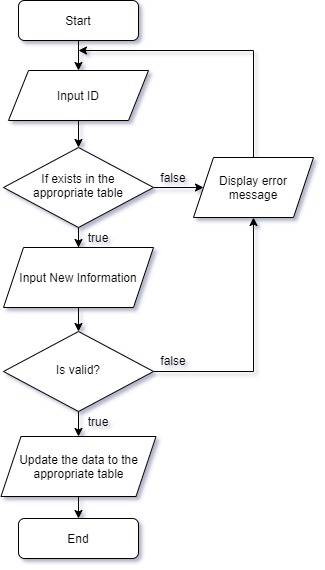
#### Search data

******

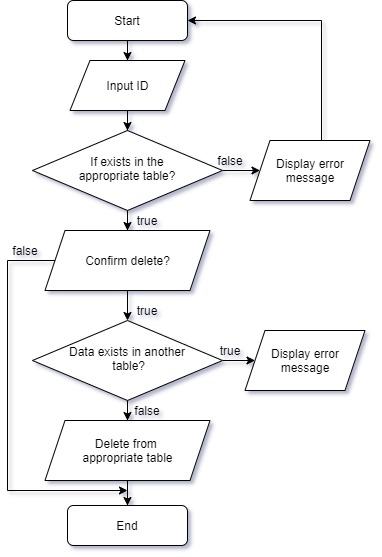
#### Create data

******

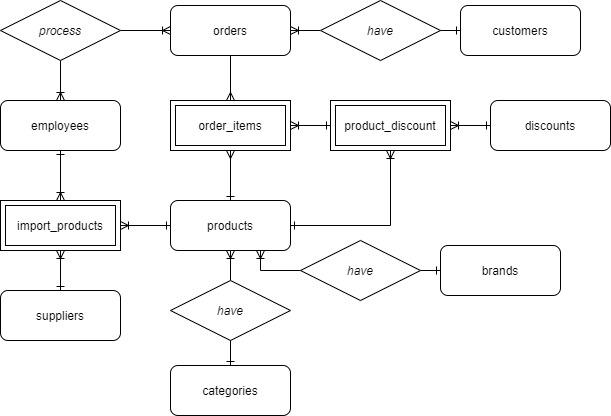
#### Update data

******

#### Delete data

******

### Entity Relationship Diagram



### Database Structure

#### Table Design

##### Table categories: product’s categories.

| **Key**2 | **Field name** | **Datatype** | **Description** |
| --- | --- | --- | --- |
| **PK UN**  AI | category\_id | int | category’s ID |
|  | category\_name | varchar(50) | category’s Name |

##### Table suppliers: Contains all suppliers.

| **Key** | **Field name** | **Datatype** | **Description** |
| --- | --- | --- | --- |
| **PK UN**  AI | supplier\_id | int | supplier’s ID |
|  | supplier\_name | varchar(100) | supplier’s Name |
|  | supplier\_add | varchar(255) | supplier’s Address |

##### Table products: Contain all the products.

| **Key** | **Field name** | **Datatype** | **Description** |
| --- | --- | --- | --- |
| **PK UN**  AI | product\_id | int | product’s ID |
| **UN FK** | brand\_id | int | brand’s ID |
| **UN FK** | category\_id | int | category’s ID |
| **UN FK** | supplier\_id | int | supplier’s ID |
| **UN FK** | employee\_id | int | employee’s ID (warehouse staff) |
|  | product\_name | varchar(100) | product’s Name |
|  | product\_price | double | product’s Price |
|  | product\_stock | int | product’s Stock |
|  | created\_date | datetime | Created date |
|  | updated\_date | datetime | Updated date |

##### Table brands : product’s brands

| **Key** | **Field Name** | **Data Type** | **Description** |
| --- | --- | --- | --- |
| **PK UN**  AI | brand\_id | int | brand’s ID |
|  | brand\_name | varchar(50) | brand’s Name |
|  | brand\_add | varchar(150) | brand’s Address |

##### Table discounts:Table of discount detail

| **Key** | **Field Name** | **Data Type** | **Description** |
| --- | --- | --- | --- |
| **PK UN**  AI | discount\_id | int | discount’s ID |
|  | discount\_name | varchar(100) | discount’s Name |

##### Table discount\_product:Percent discount for each product

| **Key** | **Field Name** | **Data Type** | **Description** |
| --- | --- | --- | --- |
| **PK UN**  AI | discount\_product\_id | int | product’s discount ID |
| **UN FK** | discount\_id | int | discount’s ID |
| **UN FK** | product\_id | int | product’s ID |
|  | discount | float | discount (percent) |
|  | start\_date | datetime | Start date |
|  | end\_date | datetime | Close date |

##### Table orders: Table of Orders

| **Key** | **Field Name** | **Data Type** | **Description** |
| --- | --- | --- | --- |
| **PK UN**  AI | order\_id | int | Order’s ID |
| **UN FK** | customer\_id | int | customer’s ID |
| **UN FK** | employee\_id | int | employee’s ID |
|  | order\_date | datetime | Create order time |
|  | amount | double | oder’s Amount |

##### Table order\_items: Table of Listing items on an Order

| **Key** | **Field Name** | **Data Type** | **Description** |
| --- | --- | --- | --- |
| **PK UN**  AI | order\_item\_id | int | order’s item ID |
| **UN FK** | order\_id | int | order’s ID |
| **UN FK** | product\_id | int | product’s ID |
|  | product\_name | varchar(50) | product’s Name |
|  | product\_quantity | int | quantity |
|  | product\_price | double | product’s Price |
|  | discount\_product\_id | int | product’s discount ID |
|  | discount\_price | double | price with discount |

##### Table import\_products: Table of imported products

| **Key** | **Field Name** | **Data Type** | **Description** |
| --- | --- | --- | --- |
| **PK UN**  AI | import\_product\_id | int | imported product’s ID |
| **UN FK** | supplier\_id | int | supplier’s ID |
| **UN FK** | product\_id | int | product’s ID |
| **UN FK** | employee\_id | int | employee’s ID (warehouse staff) |
|  | product\_quantity | int | product’s quantity |
|  | product\_price | double | product’s price |
|  | date\_added | datetime | date added |

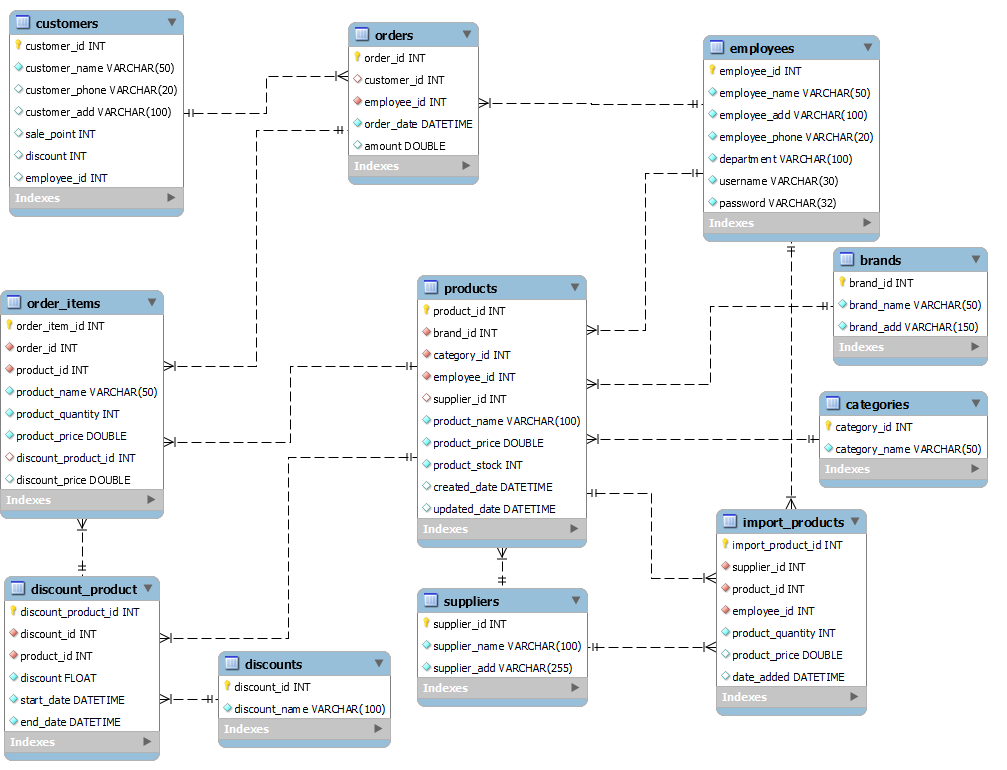
##### Table employees: List of employees

| **Key** | **Field Name** | **Data Type** | **Description** |
| --- | --- | --- | --- |
| **PK UN**  AI | employee\_id | int | employee’s ID |
|  | employee\_name | varchar(50) | employee’s Name |
|  | employee\_add | varchar(100) | employee’s Address |
|  | employee\_phone | varchar(20) | employee’s Phone Number |
|  | department | varchar(100) | department |
|  | username | varchar(30) | Username Login |
|  | password | varchar(32) | Password |

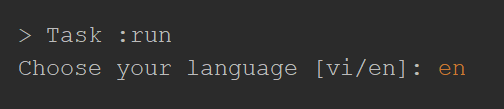
##### Table customers: Table of Loyal customers

| **Key** | **Field Name** | **Data Type** | **Description** |
| --- | --- | --- | --- |
| **PK UN**  AI | customer­\_id | int | customer’s ID |
|  | customer\_name | varchar(50) | customer’s Name |
|  | customer\_phone | varchar(20) | customer’s Phone Number |
|  | customer\_add | varchar(100) | customer’s Address |
|  | sale\_point | int | accumulated Points |
|  | discount | int | Percent discount |
|  | employee\_id | int | employee’s ID (Cashier) |

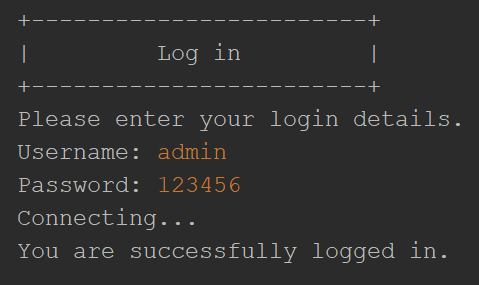
### Database Diagram

****

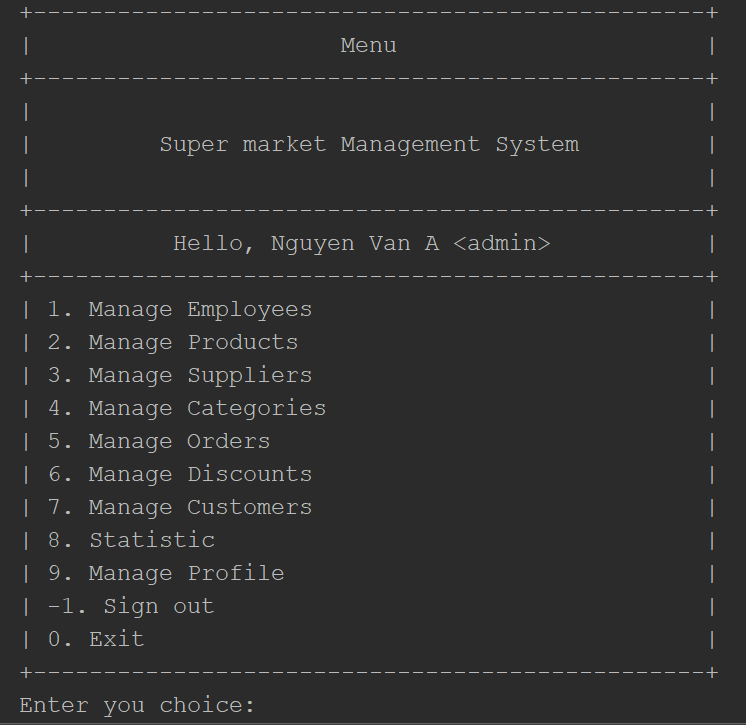
### Screen Design



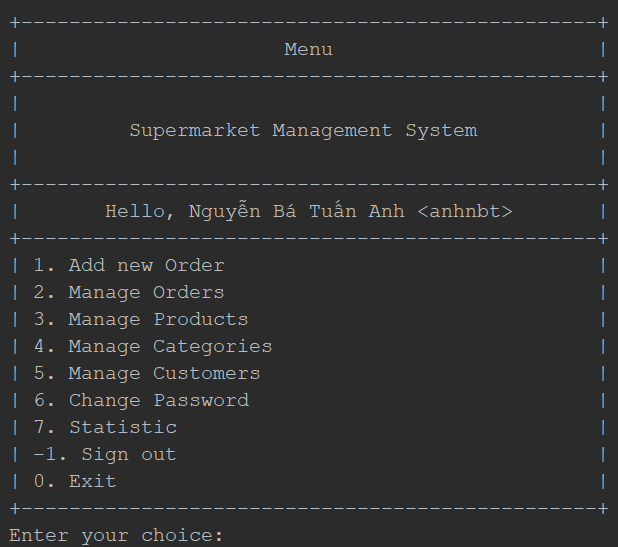
Choose your language



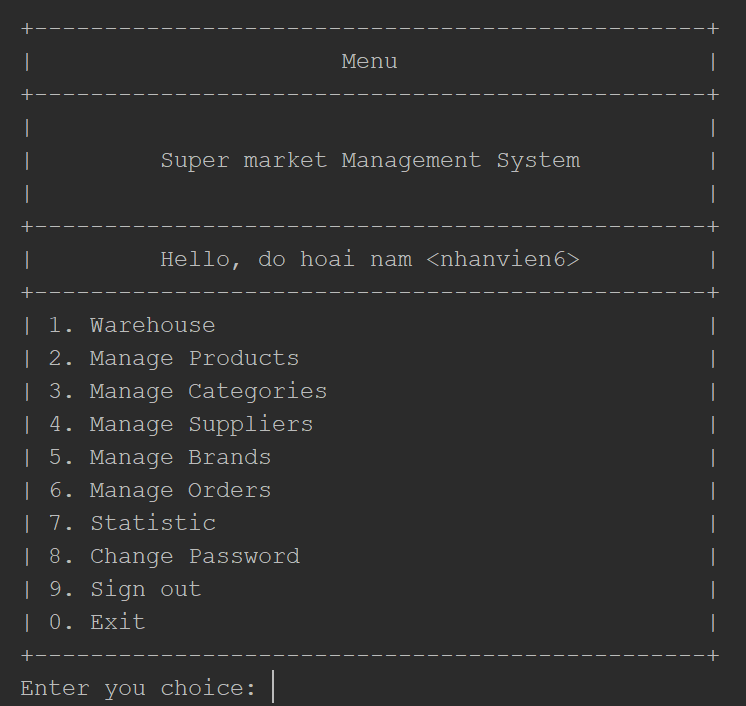
Login



Admin Main Menu



Cashier Main Menu



Inventory Main Menu

## Task Sheet

| **STT** | **Task** | **Member Name** | **Time Start - Finish** |
| --- | --- | --- | --- |
| 1 | Document | Nguyễn Bá Tuấn Anh  Phạm Vũ Tân  Lương Thanh Văn  Vũ Duy Long | 19/07/2021 - 22/09/2021 |
| 2 | Development | Nguyễn Bá Tuấn Anh  Phạm Vũ Tân  Lương Thanh Văn  Vũ Duy Long | 19/07/2021 - 22/09/2021 |
| 3 | Test | Nguyễn Bá Tuấn Anh  Phạm Vũ Tân  Lương Thanh Văn  Vũ Duy Long | 19/07/2021 - 22/09/2021 |

## Checklists

### Check List of Validation

| **No** | **Option** | **Validated** |
| --- | --- | --- |
| 1 | Do all the options present in the application display the correct result? | Y |
| 2 | Does the application’s functionality resolve the user problem, and satisfy their needs? | Y |
| 3 | Has the hardware and software been correctly chosen? | Y |
| 4 | Ensure data in constraint | Y |

### Submission Checklist

| **No** | **Option** | **Yes** | **No** | **NA** | **Comments** |
| --- | --- | --- | --- | --- | --- |
| 1 | Are the users able to log in to the application after validation is performed on the username and password? | Y |  |  |  |
| 2 | Can employees modify their details after logging in? | Y |  |  |  |
| 3 | Are all the screen contents devoid of spelling mistakes? | Y |  |  |  |
| 4 | Is the application user-friendly? | Y |  |  |  |

**--- THE END ---**