Student: Popa Diana-Maria

**Group: 30234**

Table of Contents

1. Requirements Analysis 3

1.1 Assignment Specification 3

1.2 Functional Requirements 3

1.3 Non-functional Requirements 3

2. Use-Case Model 3

3. System Architectural Design 3

4. UML Sequence Diagrams 3

5. Class Design 3

6. Data Model 3

7. System Testing 3

8. Bibliography 3

1. Requirements Analysis

# Assignment Specification

‘Furniture Management System’ is an application designed for the order managers of a furniture manufacturer. The application s has two types of users (a regular user represented by the order manager and an administrator user) which have to provide a username and a password in order to use the application.

The regular user can perform the following operations:

* Add/update/view order information (customer, shipping address, identification number, delivery date, status.).
* Create/update/delete/view product information (title, description, color, size, price, stock)
* Add products to order and update order value and stock accordingly.

The administrator user can perform the following operations:

* CRUD on employees’ information.
* Generate reports for a particular period containing the activities performed by an employee.

# Functional Requirements

* perform CRUD operation on orders, products, employees
* generate reports

# Non-functional Requirements

* System performance: the system needs to respond quickly to the requests of the user
* System security: the system is secured by requiring a user name and a password to can use the application

2. Use-Case Model

Use case: add product to order

Level: user-goal level

Primary actor: regular user

Main success scenario:

Get information about order and product

The stock of the product is grater then the required quantity

Update order information

Extensions: if the stock is insufficient, get an error message

3. System Architectural Design

**3.1 Architectural Pattern Description**

The system uses the Layered architectural pattern (3-tier architecture pattern). N-tier application architecture is characterized by the functional decomposition of applications, service components. Each tier is completely independent from all other tiers, except for those immediately above and below it.

The first layer is the data source layer. This layer has is in direct contact with the database through SQL queries.

The second layer is the domain logic layer, the layer describing the business logic of the application.

The third layer is one that the user comes in contact with, the graphical user interface.

There are no calls from an inferior layer to a superior one, and usually a superior layer calls operation from the next lower level

**3.2 Diagrams**

Package diagram



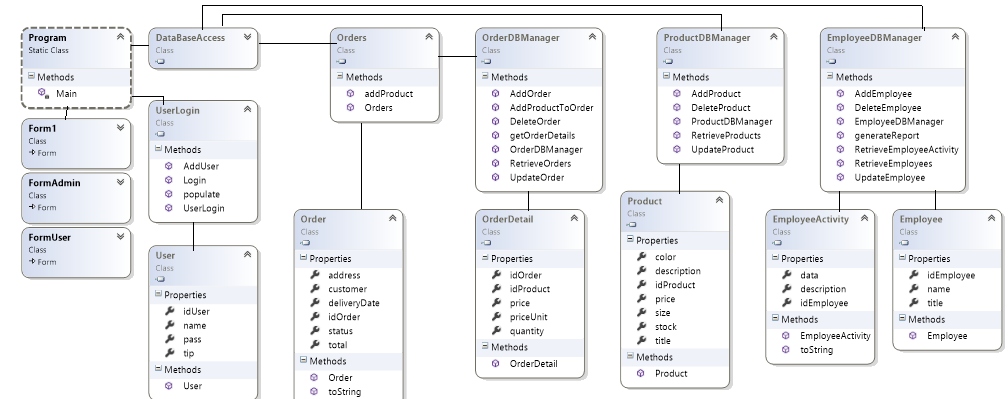
Deployment Diagram



4. UML Sequence Diagrams



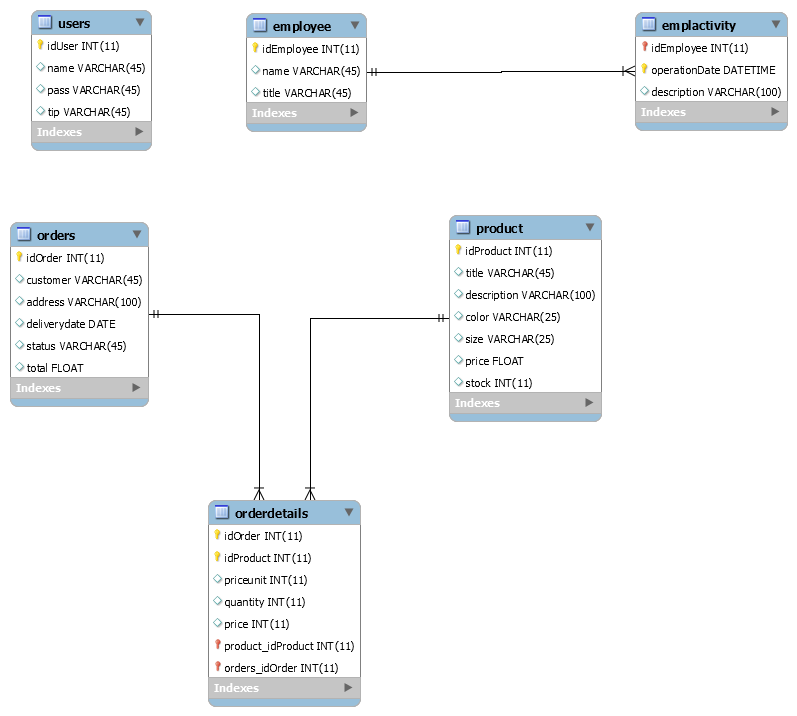
5. Class Design

****

**5.2 UML Class Diagram**

**

6. Data Model

**

7. System Testing

I have check that my code is working as expected by creating and running unit tests. Unit testing breaks down the functionality of the program into many testable behaviors that can test be tested as individual units. Visual Studio Test Explorer provides a flexible and efficient way to run your unit tests and view their results in Visual Studio.

8. Bibliography

<https://msdn.microsoft.com/en-us/library/system.security.cryptography.md5(v=vs.110).aspx>

<http://java.sun.com/docs/books/tutorial/jdbc/basics/index.html>