MINISTRY OF EDUCATION AND RESEARCH

TECHNICAL UNIVERSITY
OF CLUJ-NAPOCA, ROMANIA

FUNDAMENTAL PROGRAMMING TECHNIQUES

ASSIGNMENT 3

ORDER MANAGEMENT

1. Requirements

Consider an application **OrderManagement** for processing client orders for a warehouse. Relational databases are used to store the products, the clients and the orders. Furthermore, the application should be structured in packages using a layered architecture presented in the support presentation (**ASSIGNMENT_3_SUPPORT_PRESENTATION**.pdf) and should use (minimally) the following classes:

- **Model classes** - represent the data models of the application
- **Business Logic classes -** contain the application logic
- **Presentation classes –** GUI related classes
- **Data access classes -** classes that contain the access to the database

Note: Other classes and packages can be added to implement the full functionality of the application.

2. Deliverables

- **Solution description document** (minimum 2000 words, Times New Roman, 10pt, Single Spacing) organized according to the structure specified in the **Laboratory Description** document.
- **Source files, JavaDoc files, SQL dump file** (containing the SQL statements for: creating the database and the tables, and populating the tables with the corresponding data) – will be uploaded on the personal ***gitlab*** account created according to the instructions in the **Lab Resources** document, and following the steps:
    - Create a repository on ***gitlab*** named according to the following template *PT2021_Group_FirstName_LastName_Assignment_2* – the repository should be placed in the group named according to the template below: *PT2021_Group_FirstName_LastName*
    - Push the source code and the documentation (**push the code not an archive with the code**)
- Make sure that you give access to your group, to the PT lab assistants. On your Group page, go to: Members → Invite Member → and offer Maintainer rights for the user: utcn.dsrl@gmail.com.

3. Evaluation

The assignment will be graded as follows:

Requirement	Grading
Minimum to pass • Object-oriented programming design, classes with maximum 300 lines, methods with maximum 30 lines, Java naming conventions • Use *javadoc* for documenting classes and generate the corresponding JavaDoc files. • Use relational databases for storing the data for the application, minimum three tables: Client, Product and Order. • Create a graphical user interface including:	5 points

○ A window for client operations: add new client, edit client, delete client, view all clients in a table (JTable) ○ A window for product operations: add new product, edit product, delete product, view all product in a table (JTable) ○ A window for creating product orders - the user will be able to select an existing product, select an existing client, and insert a desired quantity for the product to create a valid order. In case there are not enough products, an under stock message will be displayed. After the order is finalized, the product stock is decremented. • Use reflection techniques to create a method that receives a list of objects and generates the header of the table by extracting through reflection the object properties and then populates the table with the values of the elements from the list. • Good quality documentation	
Layered Architecture (the application will contain at least four packages: **dataAccessLayer**, **businessLayer**, **model** and **presentation**)	2 points
Create a bill for each order as a text file or .pdf file	1 point
Use **reflection techniques** to create a generic class that contains the methods for accessing the DB: create object, edit object, delete object and find object. The queries for accessing the DB for a specific object that corresponds to a table will be generated dynamically through reflection.	2 points

4. Bibliography

- **Connect to MySql from a Java application**
  - https://www.baeldung.com/java-jdbc
  - http://www.mkyong.com/jdbc/how-to-connect-to-mysql-with-jdbc-driver-java/
- **Layered architectures**
  - https://dzone.com/articles/layers-standard-enterprise
- **Reflection in Java**
  - http://tutorials.jenkov.com/java-reflection/index.html
- **Creating PDF files in Java**
  - https://www.baeldung.com/java-pdf-creation
- **JAVADOC**
  - https://www.baeldung.com/javadoc
- **SQL dump file generation**
  - https://dev.mysql.com/doc/workbench/en/wb-admin-export-import-management.html