Ahsanullah University of Science and Technology

Department of Computer Science and Engineering

PROJECT REPORT ON

INVENTORY MANAGEMENT

COURSE NO: CSE 4126

COURSE TITLE : DISTRIBUTED DATABASE SYSTEMS LAB

DATE OF SUBMISSION : 30-08-2022

Submitted To:

G. M. Shahariar
Lecturer, Department of CSE, AUST.

Sanzana Karim Lora
Lecturer, Department of CSE, AUST.

Submitted By:

180204037 -Salsabeel Noor Azmi
GROUP : A2

ABOUT PROJECT :

“Inventory Management” is a project about the management of inventory of a clothing store. The inventory management is a distributed database system that is distributed over 2 branches (Sites -01 and 02) and a Server (Server -01). The whole project is implemented on oracle 10g Database and written on the language PL/SQL Here, we have a well defined database consisting of *multiple relations , *multiple attributes, *multiple diverse tuples ,*fragmentation, *lvl-03 Distribution transparency, *autonomous functionality etc.

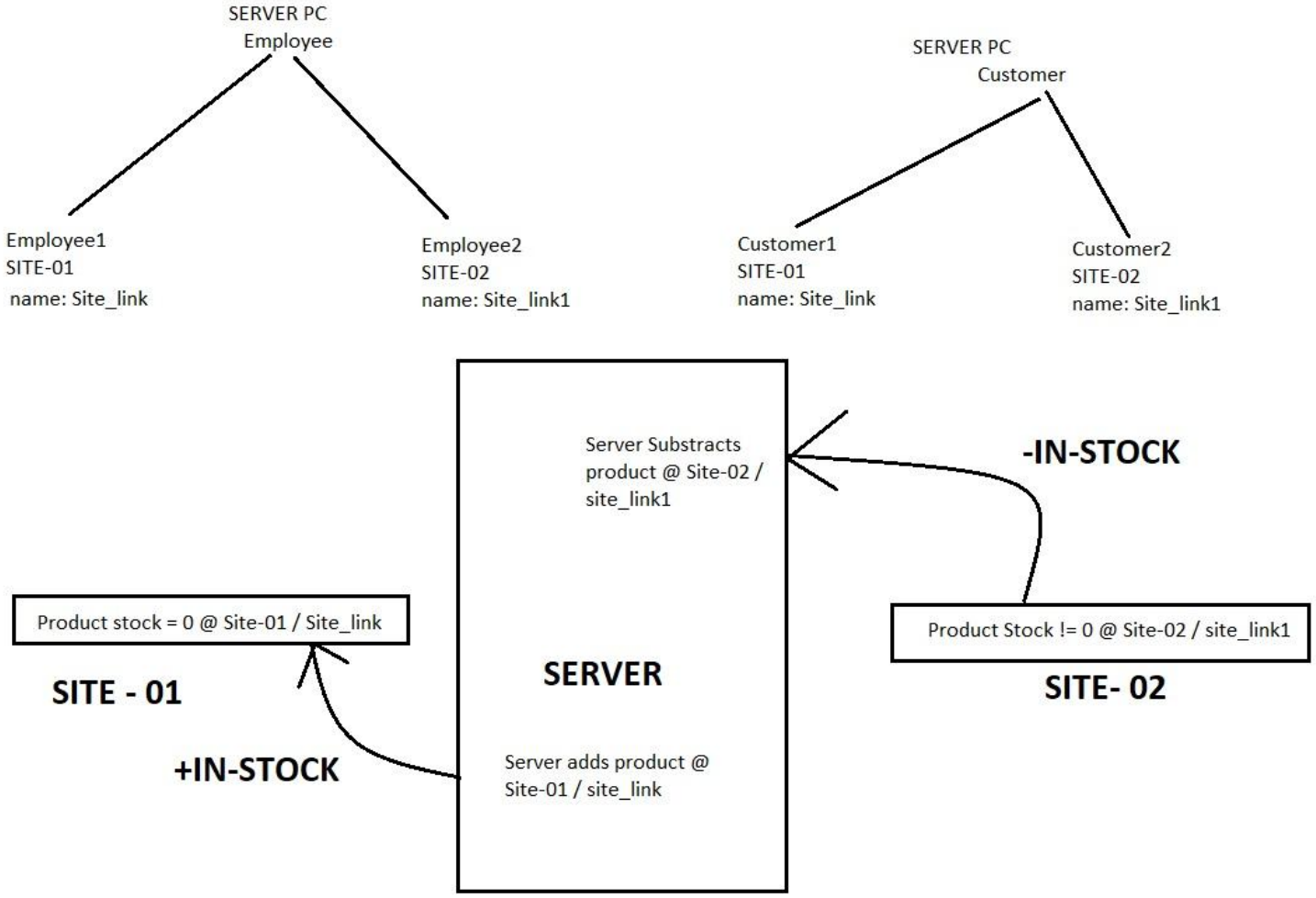
WORKFLOW:

1. DataBase.sql Executed → 2. DataInsertion.sql (Data Inserted) →

3. Proc-Func Executed → 4. Query Executed → 5. Product-transfer executed

6. membership.sql Executed → 7. Stock Distribution executed

CORE FUNCTIONALITY:



STRUCTURE OF THE DISTRIBUTED DATABASE:

The Inventory Management Distributed Database contains the following Files which makes the basic structure of the project:

- DataBase.sql :-
  - Contains 4 Relations as follows : Employee, Product , Customer, OrderDetail
  - Each relation contains Multiple Attributes
  - The special Feature of the relations is that Each primary key (ex:- EmployeeId) automatically starts its starting number and continues chronologically as the insertions are made accordingly. This is obtained by the Help of ***TRIGGER***.
- DataInsertion.sql :-
  - Contains properly differentiated sql codes for insertion of data to the respective relations.
  - Contains unique Data each tuple.
- Proc-Func.sql :-
  - Here in this single file contains all procedures and functions
  - All are implemented properly
  - All the functions are Executed in the query.sql file
  - Inside the procedures and functions contains queries that has IF-ELSE-CASE-USER_INPUT-CURSOR and other required features.
- product-transfer.sql :-
  - Here this sql file contains an autonomous feature of the project.
  - This file is about transferring product from one site to another and increasing and decreasing the stock number completely autonomously from the server.
  - It contains package-procedures-exceptions.
- membership.sql :-
  - This file contains the ability to provide membership to a customer who has spent a certain amount of money.
  - This also updates the membership in all possible branches and it is executed from the server.
- query.sql :-
  - This file contains all the required queries.
  - This file also executes procedures and functions from proc-func.sql
- stockDistribution.sql :-
  - The distribution of products from server to sites

.

FEATURES IMPLEMENTED:

The features implemented :

- IF-ELSE
- SWITCH CASE
- CURSOR (FOR LOOP)
- FUNCTION
- PROCEDURE
- USER INPUT
- PACKAGE
- EXCEPTION
- TRIGGER
- UNION
- PROMPT

SOFTWARE REQUIREMENTS:

Softwares and languages used:

- ☐ Oracle 10g, Notepad++
- ☐ Language : PL/SQL
- ☐ Execution File: sqlplus.exe

MY CONTRIBUTION:

- DataBase.sql :-
  - Employee Table + TRIGGER
  - Product Table + TRIGGER
  - Customer Table + TRIGGER
  - OrderDetail Table + TRIGGER
- DataInsertion.sql :-
  - Employee
  - Product
  - Customer
  - OrderDetail

- Proc-Func.sql :-
    - Procedure
        - IF -ELSE

- product-transfer.sql :-
    - Cursor
    - Autonomous Logic
- membership.sql :-
    - Procedure
        - Cursor
        - Union
- query.sql :-
    - User Input
    - Prompt
    - User Input
    - If-else
- stockDistribution.sql :-
    - User Input
    - Distribution logic
    - Procedure
        - If-Else
        - MOD

CONCLUSION:

So the inventory management project's Distributed Database system is able to manage the inventory of our clothing system performing a wide range of queries and it also consists of autonomous systems. This is extremely useful for autonomy and controlling branches from a headquarter.