

**Ahsanullah University of Science & Technology**

**Department of Computer Science & Engineering**

**Project Report on**

**Shop Management System**

**Course Title: Distributed Database System Lab**

**Course No: CSE 4126**

**Submitted By**

* Name: Saleh Ahmed Shafin

ID: 190104138

* Name: Md. Shahid Hasan

ID: 190104130

**Submitted To**

* Ms. Zarin Tasnim Shejuti

Lecturer

* Ms. Ashna Nawar Ahmed

Lecturer

**Date of Submission: February 15, 2023**

**Introduction**

Shop Management System is a management system project where products can be inserted to different site’s database and searching functionality from different sites is also introduced in this distributed database project.

**Proposed Implementation Method**

There are 3 tables (Shop, Branch, Product) in the server side and 1 table (Product#) in each site.

1. Server/connection.sql file is used for establishing connection with 2 sites.
2. Server/insert\_product.sql file is for inserting product.
   1. At first user will enter branch in which he/she want to store product
   2. Then user will give product details
   3. function insert\_product\_package.get\_branch\_id\_by\_name(BranchName) gives the branch id [PL/SQL control statements(loop) is used here]
   4. If branch is not found then raise custom exception
   5. If branch id is found then insert data into corresponding site’s DB
3. In Site1/Tables/Product1.sql
   1. A sequence and Product1Trigger\_before\_insert trigger is created to auto increment the ProductID when a new product is added
   2. Product1Trigger\_after\_insert trigger is for notifying user about the insertion of new row
4. Server/search\_products.sql file is used to search products according to the name
   1. Procedure search\_product\_by\_name(p\_name in Product.product\_name%type) is responsible for this functionality [PL/SQL control statements and cursors is used here]
   2. Search in both table from both sites and show the data

**Types of Database Fragmentation**

**Global Schema:**

**Shop**(ShopID,shop\_name,shop\_logo,shop\_address,shop\_contact,shop\_email)

**Product**(ProductID,product\_name,product\_description,product\_model,product\_brand,product\_purchase\_rate,product\_sales\_rate,total\_quantity,BranchID)

**Branch**(BranchID, ShopID,branch\_name,branch\_address,branch\_contact,branch\_email)

**Fragmentation Schema:**

Product1: SLBranch=’Dhaka’ Product

Product2: SLBranch=’Rangpur’ Product

**Allocation Schema:** Product1 at site 1.Product2 at site 2.

**Distributed database a good choice**

A shop management system project may need a distributed database for several reasons:

**Scalability:** A distributed database can handle large amounts of data and can easily scale to accommodate the growing needs of the business.

**Performance:** Distributed databases can improve performance by distributing data across multiple nodes, which can reduce the load on a single node and improve the response time of queries.

**High availability:** A distributed database can ensure that data remains available even if a node or server goes down.

**Data consistency:** Distributed databases can ensure data consistency.

**Flexibility:** A distributed database can be deployed on different platforms.

**Security:** Distributed databases can provide better security by distributing the data across multiple nodes, which makes it more difficult for hackers to access all the data in one place.

**Conclusion**

The fragmentation tables maintain completeness property as horizontal fragmentation is performed on Product table.