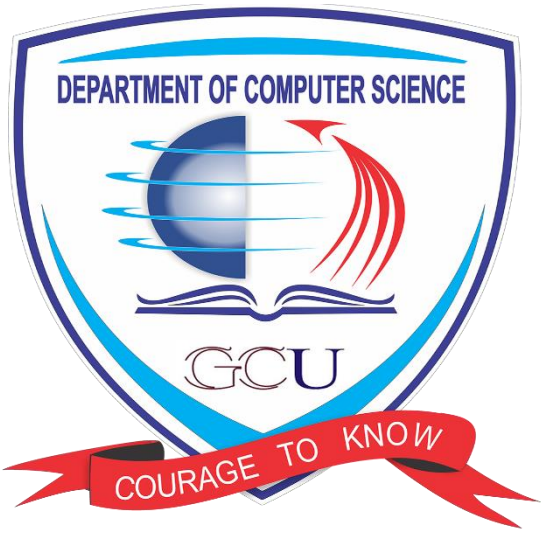


DEPARTMENT OF COMPUTER SCIENCE, GOVERNMENT COLLEGE UNIVERSITY, LAHORE

PROJECT

APPLIANCE MANUFACTURING FACTORY DATABASE SYSTEM

SUBMITTED TO

PROF. NADEEM ZAFFAR

SUBMITTED BY

M. JAWAD AMIN	**0175-BSCS-2020**
ALI HASSAN	**0163-BSCS-2020**
REHMAN DAR	**0100-BSCS-2020**

SECTION & SEMESTER

SECTION (A) **SEMESTER (IV)**

Table Of Content:

INTRODUCTION:

A **factory** or a **manufacturing plant** or a **production plant** is an industrial site, often a complex consisting of several buildings filled with machinery, where workers manufacture items or operate machines which process each item into another. They are a critical part of modern economic production, with most of the world's goods being created or processed within factories. Most modern factories have large warehouses or warehouse-like facilities that contain heavy equipment used for assembly line production. Large factories tend to be located with access to multiple modes of transportation, with some having rail, highway and water loading and unloading facilities.

An **Appliance Factory** also works in the same manner and manufactures home appliances. Manufacturing of a product involves a lot of phases through which it must pass in a factory. When we talk about a factory it means that different types of products are being manufacture there. Manufacturing of these products involves assembly of many parts, manufacturing its subparts and many other things. In a factory it is very difficult and tedious to keep track of the data of every phase with records on paper. So, their data should have to be store in a place from where it can be access easily and there should be no errors. For this purpose, we are designing a database for an appliance factory which would be a general database design and can be use in any type of factory which manufactures multiple products.

DATABASE DEVELOPMENT LIFE CYCLE (DBLC):

DBLC stands for Database Development Life Cycle. Within the information system, the most successful databases are subjected to frequent evaluation and revision within a framework known as DBLC.

PHASES OF DBLC:

There are six phases of Database Development Life Cycle (DBLC) which should be followed in the same sequence to design a database. Names of these phases are as follows:

1. Database Initial Study
2. Database Design
3. Database Implementation and Loading
4. Database Testing and Evaluation
5. Database Operations
6. Maintenance and Evolution

DATABSE INITIAL STUDY:

Database Initial Study consists of the following steps which are as follows:

- Analyze Company Solution
- Define Problems and Constraints
- Define Objectives
- Define Scope and Boundaries

- **ANALYZE COMPANY SOLUTION:**

In this step a database designer discovers what the company's operational components. He also checks how they function, and how they interact with each other. We are designing a Database for an Appliance Manufacturing Factory. The database we are designing is not the complete database of an Appliance Manufacturing Factory, basically it is a database of the manufacturing module of the factory. The operational components in this module which we have identified are workers, warehouses, products, raw material, assembly lines and department for each product. These all components interact with each other. These components depend on each other for their working. For example, workers operate the assembly line, products and raw material store in the warehouses, each department has some warehouses and assembly line for the product productions. They all interact with each other in an efficient way which enables the manufacturing module work perfectly.

- **DEFINE PROBLEMS AND CONSTRAINTS:**

In this step designer continue to careful probe to generate additional information that will help define problems within the larger framework of company operation. The problem which we have found is that a factory owner wants to keep the record of what is going inside his factory. Basic purpose of this record keeping is that he wants to make different types of decisions on the bases of this dataset and wants to check whether the factory is making profit or not. For example, decision regarding increasing the production of products in the factory or improving the quality of the products. On the other hand, constraints regarding this problem are limiting the use of resources, competitive analysis, improving the utilization of resources, and removing the disruptions.

- **DEFINE OBJECTIVES:**

The main objective of our database is to keep the record of what is going inside the factory. Some of the other objectives are as follows:

1. **HIGH QULITY PRODUCTS:**

One of the most important objectives of this database enable the manufacturing process is to produce a high-quality product. If you own a business that relies on a manufacturer to produce your goods, you will lose faith in that manufacturer if the product is not made to the highest standard.

2. **PRODUCE MAXIMUM GOODS:**

Another objective of our database is enabling the factory to produce as many goods as possible in the least expensive way possible. Manufacturers can only achieve that objective by relying upon heavy equipment, much of which is automated. Although this equipment is an asset when it comes to boosting production, it also poses risks to workers. For example, heating elements or disconnected joints can burn employees at a factory.

3. **BOOSTING EFFICIENCY:**

Another objective of our database is to boost the efficiency of the manufacturing process. Boosting efficiency while controlling costs is another objective of the manufacturing process. In manufacturing, efficiency is interconnected with costs. In general, the greater the efficiency, the lower the costs. For example, if a manufacturer misses a production deadline, the company might face a contract penalty, or it might lose a bonus. In the same way, if workers are taking too long to finish a project, the company may have to pay workers overtime or hire additional temporary staff to get back on schedule.

- **DEFINE SCOPE AND BOUNDARIES:**

In a database design we also must define the scope of our database. Basically, scope of a database helps us to define the required data structures, datatypes, number of entities and physical size of the database. The scope of our database is just to keep the record of the manufacturing module of appliance factory and tracking whether the manufacturing process is working efficiently or not and checking the usage of raw material and performances of the workers. Boundaries are the external limits to the database system beyond which it will not work perfectly. Some boundaries are hardware and software boundaries which bounds our database.

DATABASE DESIGN:

This phase is all about designing the structure of the database means that how would be actual appearance of the database logical design or physical design. This phase consists of the following steps:

- Create Conceptual Design
- DBMS Software Selection
- Create Logical Design
- Create Physical Design

- **CREATE CONCEPTUAL DESIGN:**

In this step we design the ERDs for our database. This step involves the designing of the **Unnormalized** ERDs and then we must **Normalized** these ERDs. In this step we also do the distributed database design and data model verification. This step consists of the following sub steps which are as follows:

1. Scenario
2. Noun Verb Analysis
3. Unnormalized ERD
4. Normalized ERD

1. **SCENERIO:**

A Worker works for one and only one Department but a Department can have one or more than one Workers. Each Worker has its ID, Salary, SSN, First Name, Last Name, Phone and Position. One Worker can or cannot manages one or more than

one Worker. One Worker possesses one and only one Worker Detail and one Worker Detail can only be given to one Worker. Worker Detail has Blood Group, DOB, Address, CNIC, Email and Gender. A Worker can work in at most one Assembly Line and one Assembly Line can be Operated by one or more Workers. An Assembly Line has its Id, Name, Units Produces Per Day and Function. A Warehouse Stores one or more than one Products and Product can be in one and only one Warehouse. A Product has its Name, Price, Quantity, Expiry Date, and Id. Warehouse has its Name, Size, and its Id. Warehouse can store one or more than one Raw Material, but Raw Material can be store in only one Warehouse. Raw Material has its Name, Quantity, Price, Expiry Date, and its Id. One Department contains one or more Warehouse, but one Warehouse can only be in one Department. One Assembly Line produces one or more Products, but one product can be produced by only one Assembly Line. One Worker can or cannot deals with one Warehouse, but one Warehouse can have one or more Worker. One Assembly Line uses one or many Raw Materials and one Raw Material can be used in one or many Assembly Lines. Location has Postal Code, Country, Province, Street Address, City, and Id. Department has Name and Id. A Warehouse possesses only one Location, and a Location is possessed by only one Warehouse. A Department possesses only one Location, and a Location is possessed by only one Department. A Department is managed by one or more Administrators, and one Administrator has only one Department. An Administrator possesses one Admin Detail, and one Admin Detail is possessed by only one Administrator. An Administrator has Id, Phone, Salary, Position, First Name, and Last Name. Admin Detail has CNIC, Email, Gender, DOB, Blood Group, and Address. One Admin can manage maximum many admins and one admin is managed by maximum one Admin. Administration can control many Workers and a Worker can be control by one Admin.

2. NOUN VERB ANALYSIS:

NOUNS	ADJECTIVES	VERBS
PRODUCT_TBL	Expiry Date, Price, Name, Id, Quantity	Produce by Assembly, Stores in Warehouse,
WAREHOUSE_TBL	Name, Size, Id	Stores Products, Stores Raw Material, Manage by Workers, Contains by Department, Possesses Location
RAWMATERIAL_TBL	Expiry Date, Id, Quantity, Name, Price	Stores in Warehouse, Use by Assembly line
WORKER_TBL	SSN, Salary, Id, Full Name (First Name, Last Name), Phone Position	Manages Warehouse, Supervises Workers, Works in Assembly in Line, Works for Department, Possesses Detail, Control by Administration
DEPARTMENT_TBL	Name, Id	Possesses Location, Contains Warehouses, Have Assembly Line, Manages by Administration, Contains Workers

LOCATION_TBL	Postal Code, Country, Province, Street Address, Id, City	Possessed by Department, Possessed by Warehouse
ASSEMBLYLINE_TBL	Name, Id, Function, Units Per Day	Operates by Workers, Belongs to Department, Uses Raw Material, Produces Products
WORKERDETAIL_TBL	CNIC, DOB, Blood Group, Gender, Address	Possessed by Workers
ADMINISTRATION_TBL	SSN, Salary, Id, Full Name (First Name, Last Name), Phone Position	Manages Admins, Manages Departments, Control Workers
ADMINDETAIL_TBL	CNIC, DOB, Blood Group, Gender, Address	Possessed by Administration

3. ER-DIAGRAMS:

ERD stands for Entity Relationship Diagram. Basically, it is the graphical representation of Entities and their relationships with other Entities. There are two types of Entity Relationship Diagrams which are un-normalized and normalized Entity Relationship Diagrams. Entity Relationship Diagrams of our scenario will be provided in handwritten format.

- **SOFTWARE SELECTION:**

In this process we select a Database Management System S Software (DBMS) depending on the structure of our Entity Relationship Diagrams, efficiency, and user-friendly interface of the DBMS. A good selected software helps the user to use it well and impacts positive on the performance.

- **LOGICAL DESIGN:**

A logical design consists of the relational model the entities which describes structure of the entities regarding their attributes and their relationships with other entities. Relational models of our entities are as follows:

PRODUCT_TBL

Names	Datatype	Length	Nullable
P_ID	Numeric	4	Not Null
P_Name	Varchar	50	Not Null
P_Type	Varchar	50	Not Null
P_Price	Numeric	10	Not Null
P_Quantity	Numeric	5	Not Null
P_Expiry	date	-	Not Null
WH_ID	Numeric	4	Not Null
AL_ID	Numeric	4	Not Null

RAWMATERIAL_TBL

Names	Datatype	Length	Nullable
RM_Id	Numeric	5	Not Null
RM_Name	Varchar	50	Not Null
RM_Price	Numeric	6	Not Null
RM_Quantity	Numeric	6	Not Null
RM_Expiry	date	-	Not Null
WH_Id	Numeric	4	Not Null

WORKER_TBL			
Names	**Datatype**	**Length**	**Nullable**
W_Id	Numeric	4	Not Null
W_First_name	Varchar	50	Not Null
W_Last_Name	Varchar	50	Not Null
W_Position	Varchar	50	Not Null
W_Salary	Numeric	8	Not Null
W_Phone	Numeric	15	Not Null
W_SSN	Numeric	15	Not Null
W_Supervisor	Numeric	4	-
WH_ID	Numeric	4	-
AL_ID	Numeric	4	-
WD_ID	Numeric	4	Not Null
D_ID	Numeric	4	-

WORKERDETAIL_TBL			
Names	**Datatype**	**Length**	**Nullable**
W-CNIC	Varchar	15	Not Null
W-Email	Varchar	50	-
W-Gender	Varchar	10	Not Null
W-Address	Varchar	50	Not Null
W-DOB	date	-	Not Null
W-Blood_Group	Varchar	10	-
W_ID	Numeric	4	Not Null

ASSEMBLYLINE_TBL			
Names	**Data Type**	**Length**	**Nullable**
AL_Id	Numeric	4	Not Null
AL_Name	Varchar	50	Not Null
AL_Function	Varchar	100	Not Null
AL_Units_Per_Day	Numeric	4	Not Null
D_Id	Numeric	4	Not Null

ASSEMBLY_USE_RAW_TBL			
Names	**Datatype**	**Length**	**Nullable**
AL_ID	Numeric	4	Not Null
RM_ID	Numeric	5	Not Null
Raw_Per_Unit	Numeric	4	Not Null

ADMINISTRATION_TBL			
Names	**Datatype**	**Length**	**Nullable**
ADM_Id	Numeric	4	Not Null
ADM_First_Name	Varchar	50	Not Null
ADM_Last_Name	Varchar	50	Not Null
ADM_Position	Varchar	50	Not Null
ADM_Salary	Numeric	8	-
ADM_Phone	Numeric	15	Not Null
ADM_SSN	Numeric	15	-
ADM_Manager	Numeric	4	-
D_ID	Numeric	4	-

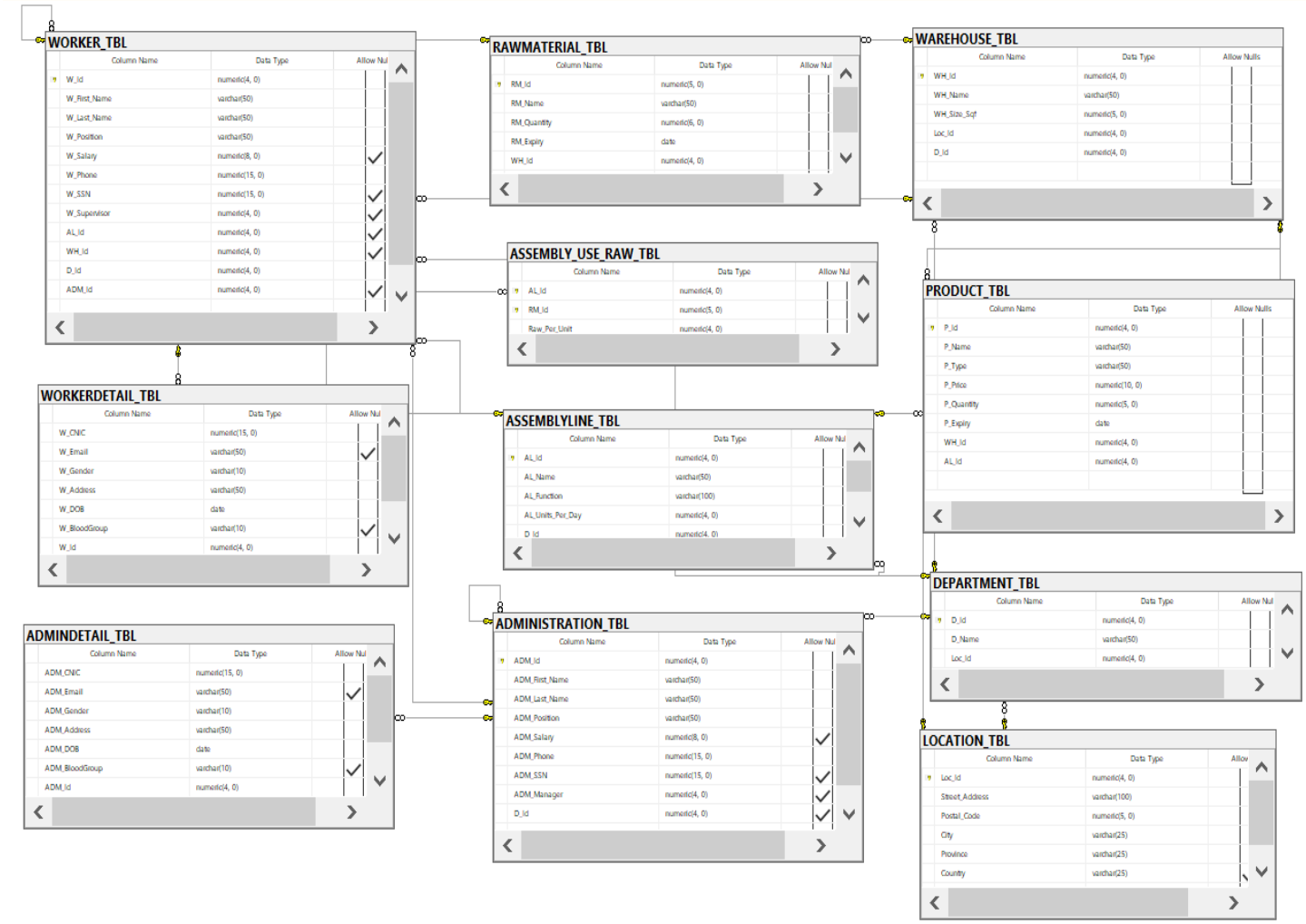
DEPARTMENT_TBL			
Names	**Datatype**	**Length**	**Nullable**
D_ID	Numeric	5	Not Null
D_Name	Varchar	50	Not Null
Loc_Id	Numeric	4	Not Null

ADMINDETAIL_TBL			
Names	**Datatype**	**Length**	**Nullable**
ADM_CNIC	Numeric	15	Not Null
ADM_Email	Varchar	50	Not Null
ADM_Gender	Varchar	10	Not Null
ADM_Address	Varchar	50	Not Null
ADM_DOB	date	-	Not Null
ADM_BloodGroup	Varchar	10	Not Null
ADM_ID	Numeric	4	Not Null

WAREHOUSE_TBL			
Names	**Data Type**	**Length**	**Nullable**
WH_Id	Numeric	4	Not Null
WH_Name	Varchar	50	Not Null
WH_Size_Sqf	Numeric	5	Not Null
Loc_Id	Varchar	4	Not Null
D_Id	Numeric	4	Not Null

- **PHYSICAL DESIGN:**

  In this step we talk about the relational model of the related entities which are created by the DBMS. The relational model for our ERD is as follows:



IMPLEMENTATION & LOADING:

This phase of database involves the following phases which are as follows:

- Installing DBMS
- Creating the Databases
- Load or Convert Data

- **INSTALLING DBMS:**

  In this phase we install the DBMS to implements our database. For this purpose, we are using the Microsoft SQL Server which is the most widely used DBMS. One more thing, while you are installing the DBMS you should have to

understand all requirements of your database such as storage, all types of constraints you want to apply and how to use the DBMS.

- **CREATING DATABASE:**

  This is the phase in which our actual implementation of the database starts. In this phase we create tables means entities of our database along with their attributes means with their columns.

- **LOADING & CONVERTING DATA:**

  In this phase we insert data into our created database according to the structure of the database which has been created. Exact data into the exact table into exact columns and rows.