**RDBMS MINI PROJECT**

**(SPRINT 1 &2)**

**WAREHOUSE MGMT. SYSTEM**

**Table of Contents**

|  |  |  |  |
| --- | --- | --- | --- |
| Serial No. | Topic Name | Content | Page No. |
| 1. | Introduction | 1.1 Setup checklist for mini project | 3 |
|  |  | 1.2 Instructions | 3 |
| 2. | Problem statement | 2.1 Objective | 4 |
|  |  | 2.2 Abstract of the project | 4 |
|  |  | 2.3 Functional components of the project | 4 |
|  |  | 2.4 Technology used | 5 |
| 3. | Implementation in RDBMS LOT | 3.1 Guidelines on the functionality to be built | 6 |
| 4. | Evaluation | * 1. Evaluation | 7 |

1. **Introduction**

This document outlines a mini project for the RDBMS LOT. The project is to design the database, reports the queries related to Warehouse Management System. This document contains information about the attributed that will be participating in the system and guidelines about reports.

* 1. **Setup Checklist for Mini Project**

Minimum System Requirements

* Physical Memory (RAM) - 1GB Minimum
* Virtual Memory - Double the amount of RAM
* Disk space - Total 5 GB
* Processor - 550 MHz minimum
* Video Adapter - 256 colors
  1. **Instructions**
* Follow standards while coding
* Create a directory by your name in drive <drive>. In this directory, create a subdirectory MiniProject. Store your Project here.
* You can refer to your course material.
* The total time required to complete this mini project is 8 hrs.
* Maintain the code.

1. **Problem Statement**
   1. **Objective**

Designing the database, developing the queries and basic reports required for Warehouse Management System.

* 1. **Abstract of the Project**

## Our company has several warehouses, each warehouse is designated by a unique 4-letter symbol (by a letter we mean a..z and A..Z). Each warehouse has several bins that are identified uniquely by numbers (unsigned integers), i.e. each warehouse has bins 0, 1, 2, 3, … Each bin has a particular capacity. In our warehouses (more precisely in the bins in

## our warehouses) we store parts. Each part is designated by a unique part number (a 5-symbol sequence of digits and letters). Several parts together can form another part. We call such a part “assembly”. In the warehouses we store only the constituent parts, but we record the assemblies in our database as it were a part. Assemblies cannot be parts of other assemblies. A part can be a constituent part in at most in one assembly parts arrive in batches. Each batch for a particular part has a unique batch number (unsigned integer)

## and arrives on a particular date. Each batch has a size, i.e. the number of items in the batch. All items from the same batch are stored together in the same bin (no batch is stored in more than 1 bin). Each item in a batch has a unique item number (unsigned integer). For example: part A1, batch 27, item 1 or part A1, batch 23, item 1 etc.

## When a batch arrives, its date-in is recorded. A particular manager checks its arrival, and this fact must be recorded in the database. Some parts may be backordered. A part can be backordered only by a manager. The manager, the date of the backorder are recorded, and also the quantity backordered.

## When a backorder shipment arrives, the backorder’s remaining quantity is updated (the number of items arrived is subtracted from the remaining quantity), and if it is less or equal to 0, the backorder is deleted, but must be kept for record. There may be only a single current (active) backorder for any parts. Assemblies cannot be backordered, only their constituent parts.

## When an item is shipped out of the warehouse, its date-out is recorded together with the employee who checked its shipping.

## Employee has a unique employee number (a 6-digit number), phone number(s) (it consists of a 3-digit area code and a 6-digit number an employee can have 0 to many phone numbers), name(s) (it consists of an up=to-10-characters fist name, an up-to-10-characters middle name, and an up-to-20-characters last name, an employee can have 1 to many names), address(s) (it consists of an up-to-6-characters street number, an up-to-20-characters street name, an up-to-20-characters city name, and a 2-character abbreviation of the province, an employee can have 1 to many address). Some of the employees are managers. Every employee who is not a manager works under supervision of a single manager. Managers do not work under other managers.

* 1. **Functional components of the project**

Design the normalized relational database using the following details. You can make appropriate assumptions wherever required. Some of the attributes are given below with the restrictions on data it can contain. Find the required attributes for all the tables and create appropriate constraints on it. (For Ex. Primary key, Foreign key, etc.)

Some of the entities and attributes are as follows:

* **Item\_Master –** Item Number, Part number, Batch number, Date out, Checked out
* **Bin\_Details –** Bin Number, Warehouse Id
* **Warehouse\_Master –** Warehouse Id, Warehouse Name, Address
* **Part\_Master –** Part Number
* **SubPart\_Master –** Part Number , Assembly Number
* **Emp\_Name –** Employee Number, First Name, Last Name, Middle Name
* **Emp\_Address –** Employee Number, Street Number, Street Name, City,

Province

* **Emp\_Master** – Employee Number
* **Worker\_Master** - Employee Number
* **Manager\_Master** - Employee Number
* **Current\_Backorder** – Part Number, Original Quantity, Remaining Quantity, Back Order Date, Employee Number

This Mini project will be done individually. Implement the Software development life cycle for the project and develop code for the respective functionality. Billing will be done using online presentation mode, where customer will choose their plan.

This project shall be done in 3 parts :

* Employee login and registration.
* When a Batch Arrives
* When an item is shipped out of the warehouse

Some of the guidelines/protocols are given below:- Normalize the tables.

* Create additional tables, if necessary.
  1. **Technology Used**

Databases:

Oracle 11G Express Edition

1. **Implementation in RDBMS LOT**
   1. **Guidelines on the functionality to be built :**

* Create a procedure which gets all employee no, employee name for all the workers that work under a specific manager employee no is input to procedure
* Create a procedure which gets list of all current backorders done by the manager employee no is input to procedure
* Create a procedure which for each warehouse bin, give the remaining capacity of the bin. Call the remaining capacity remaining\_capacity.
* Create a procedure which returns Give employee\_no and number of workers managed for all the managers with The smallest number of workers managed.
* Create a procedure which give all the phones and employee\_no for all the managers.
* Create a procedure which is called when a backorder shipment arrives, the backorder’s remaining quantity is updated (the number of items arrived is subtracted from the remaining quantity), and if it is less or equal to 0, the backorder is deleted, but must be kept for record.
* Create a trigger which gets triggered when a particular manager checks its arrival, batch date–in and this fact must be recorded in the database.

**SPRINT 2**

1. Perform performance tuning on the database and the pl/sql code built.
2. Create the entire database in Oracle db.
3. **Evaluation and assessment parameters:**
   1. **Evaluation**

* Evaluation will be done at the end of Oracle training
* Total Marks: 100
* Marks Distribution mentioned below.

This Mini project will be done individually. Implement the Software development life cycle for the project and develop code for the respective functionality. Evaluation will be done using online presentation mode, where participant will present their work.

This project shall be evaluated in two parts:

* Marks distribution is for one part of project evaluation (Marks: 90)
* Project Presentation is another part of project evaluation (Marks: 10)