INVENTORY MANAGEMENT SYSTEM

Database Design Document

Group - 24

Podduturi Shrenik Reddy - B180434CS

Nadimpalli Anand Varma - B180373CS

Yadla Prasanth Babu - B180580CS

Mandu Kireeti Nayak - B180443CS

Version (1.0) Date:27-11-2020

Table of Contents

1 Purpose.. 4

1.1 Document Objectives.. 4

1.2 Intended Audience.. 4

1.3 Acronyms and Abbreviations.. 4

1.4 Key Personnel.. 5

1.5 Data Owners.. 6

2 Assumptions, Constraints and Dependencies.. 6

2.1 Assumptions.. 6

3 System Overview.. 7

3.1 Database Management System Configuration.. 7

3.2 Database Software Utilities.. 7

3.3 Support Software.. 7

4 Architecture.. 8

4.1 Hardware Architecture.. 8

4.2 Software Architecture.. 8

4.3 Datastores.. 9

5 Database-Wide Design Decisions.. 10

5.1 DBMS Platform.. 10

5.2 Security and Availability.. 10

5.3 Distribution.. 10

5.4 Backup and Restore Operations.. 11

5.5 Maintenance.. 11

5.6 Performance and Availability Decisions.. 11

6 Database Administrative Functions.. 12

6.1 Responsibility.. 12

6.2 Database Identification.. 12

6.3 Application / Systems Using the Database.. 13

6.4 Relationship to Other Databases.. 13

6.5 Relational Schema Design... 13

6.6 Entity Mapping.. 14

6.7 Normalisation of Database..
14

6.8 Mapping Rules... 15

6.9 Data Transfer Requirements.. 15

6.10 Storage.. 16

6.11 Backup and Recovery.. 16

7 Detailed Database Design.. 17

7.1 Data Software Objects and Resultant Data Structures... 17

1 Purpose

The Database Design Document maps the logical data model to the target database management system with consideration to the system's performance requirements. The Database Design converts logical or conceptual data constructs to physical storage constructs (e.g., tables, files) of the target Database Management System (DBMS).

1.1 Document Objectives

The Database Design Document has the following objectives:

To describe the design of a database, that is, a collection of related data stored in one or more computerized files that can be accessed by users or computer developers via a DBMS.

To serve as a basis for implementing the database and related software units. It provides the acquirer visibility into the design and provides information necessary for software development.

1.2 Intended Audience

This document is intended for the following audiences:

Technical reviewers, who must evaluate the quality of this document.

Developers including:

Architects, whose overall architecture design must meet the requirements specified in this document.

Designers, whose design must meet the requirements specified in this document

Developers, whose software must implement the requirements specified in this document.

Quality Assurance personnel, whose test cases must validate the requirements specified in this document.

1.3 Acronyms and Abbreviations

Acronym / Abbreviation	Meaning
RDMS	Relational Database Management System
DBA	Database Administrator

1.4 Key Personnel's

Role	Name	Email	PHONE NO
DBA	Podduturi Shrenik Reddy	shrenik_b180434cs@nitc.ac.in	9177012001
DBA	Nadimpalli Anand Varma	anandvarma_b180373cs@nitc.ac.in	9618278852
DBA	Yadla Prasanth Babu	prasanthbabu_b180580cs@nitc.ac.in	9515369726
DBA	Mandu Kireeti Nayak	kireeti_b180443cs@nitc.ac.in	9989889713

1.5 Data Owner

Type of data	Name	Email	Mobile no
Transactional	Nadimpalli Anand Varma	anandvarma_b180373cs@nitc.ac.in	9618278852
Master data	Mandu Kireeti Nayak	kireeti_b180443cs@nitc.ac.in	9989889713
Reporting data	Podduturi Shrenik Reddy	shrenik_b180434cs@nitc.ac.in	9177012001
Metadata	Yadla Prasanth Babu	prasanthbabu_b180580cs@nitc.ac.in	9515369726

2 Assumptions, Constraints and Dependencies

2.1 Assumptions

The product needs the following third-party applications for the development of the project:

● Android Studio (for development of android based applications)

● Netbeans

● Photoshop (for editing layouts, icons, buttons, etc)

3 System Overview

3.1 Database Management System Configuration

There is no designated hardware for the system so it can work on any system with any os of any version on pc.

Some hardware requirements will be:

- Operating system: Windows, Linux
- Hard disk: 40 GB
- RAM: 256MB (minimum)
- Processor: Pentium(R)Dual-core

CPU Limitation: The current version of the server we are using for creating the database will work on windows 7 and above.

3.2 Database Software Utilities

The database management system used is my SQL as the name suggests SQL is used for creating and handling database and all required functions and before performing any related functions or queries Xampp server should be started and we can access the database through Server: 127.0.0.1 via TCP/IP and the related queries should be performed through the server.

3.3 Support Software

The software directly related to the database we are using for storing our database is my SQL which basically uses SQL as a query language to perform all the database related queries whereas **MySQL** is written in C and C++. Its **SQL** parser is written in yacc, but it **uses** a home-brewed lexical analyzer.

My SQL is a web server database and all the details regarding it and apache server is the one through which we are going to handle all the files related to the database.

All the details regarding software including name and versions are provided below:

Database Server:

- Server: 127.0.0.1 via TCP/IP
- Server type: MariaDB
- Server connection: SSL is not being used
- Server version: 10.4.14-MariaDB - mariadb.org binary distribution
- Protocol version: 10
- User: root@localhost

Server charset: UTF-8 Unicode (utf8mb4)

Web Server :

Apache/2.4.46 (Win64) OpenSSL/1.1.1g PHP/7.4.9

Database client version: libmysql - MySQLnd 7.4.9

PHP extension: mysqli

PHP version: 7.4.9

phpMyadmin : Version information: 5.0.2, latest stable version: 5.0.4

4 Architecture

4.1 Hardware Architecture

In hardware architecture we will mostly see how the data is stored in databases and how they are connected.It mostly comprises internal level.So it deals with how the information is stored.

4.2 Software Architecture

Software architecture mostly deals with external level.Software architecture deals with interfaces.

4.2.1 User interfaces

4.2.1.1 Dashboard Interface:

This is the opening interface that the user will see when they open the portal. In this interface, multiple login options will be displayed – customer login, employee login, admin login. From which the user can select which login form, and it redirects to the corresponding login page.

4.2.1.2 Admin Login Interface:

In this interface, when the user selects the admin login it will show two text fields namely username and password along with the login button. From this interface, the authorized login person has access to insert, modify the employee details, and verify the customer details.

4.2.1.3 Employee Login Interface:

In this interface, when the user selects the employee login it will show two text fields namely employeeID and password along with the login button. From this interface, the authorized employee has access to insert the product details and supplier details. Users can verify the customer details and what they selected and ordered.

4.2.1.4 Customer Login Interface:

In this interface, when the user select the customer login it will show two text fields namely username and password along with the login button, if the user is new, there will be a register button to create his account in the portal and he needed to fill some recommended details (like Name, address, mobile number).

4.3 Datastores

Data is stored in the database and in the corresponding tables .For file management, we are storing the content in htdocs in the xampp folder.Php files are used for backend and html and css files are used for front end.

5 Database-Wide Design Decisions

5.1 DBMS Platform

This DBMS software can be implemented with ease on linux based operating systems,windows,and mac os. An open source cross platform web-server should be installed to create a database (ex: xampp)

Which can have baseline requirements to meet to install.

Baseline requirements for the system:

- Windows 2008 server or later
- Mac os X 10.6
- CentOS,Ubuntu,Fedora,Gentoo
- Hard disk space : 40GB, RAM :256 MB
- Processor: Pentium(R)Dual-core CPU

5.2 Security and Availability

This software will,

- Authenticate each user based on the type of user which are admin and student
- When an user performs an action which is not authorised by the user, the system will display an error message if it's found to be unauthorized.

5.3 Distribution

The Master database is the primary configuration database in SQL Server. It contains information on all the databases that exist on the server, including the physical database files and their locations. The Master database file also contains SQL Server’s configuration settings and login account information.

Components in the master database:

- Registrations and Remote Logins
- Local Databases and Database Files
- Login Account
- Server Configuration Settings
- Processes and Locks

A current backup will be kept updating it from time to time as it is critical to any server recovery. Integrity standards are high and the privacy is kept for the user,no unauthorized information will not be disclosed. No business rules are included.

5.4 Backup and Restore Operations

The user information will be kept private for safety and security issues and will not be disclosed to any other third party organizations so that user privacy is intact and information is safe.

The data is backed up on a regular basis so the data will not be lost if database crashes or any other harm which leads to loss of data. Also as a safety measure, the data is stored on a private storage so it can’t be accessed from outside. For detailed information on actions in backup please refer to the 6.11 module below.

5.5 Maintenance

Maintenance includes modifications in the software product after it is delivered.
Automatic logging and error reporting techniques, automatic error message generation.

5.6 Performance and Availability Decisions

The DBMS software developed should be able to work effectively to give out information when needed and also to store the data without any latency to avoid any issue.Several factors which affect the performance is that the system resources should be adequate and meet the baseline requirements for the software to run without any issues.

The availability of the software is adequate for users to use, some features are restricted for the users as there would be a potential threat of data loss and privacy issues.the database can only be edited by admins to alter the data when needed.

6 Database Administrative Functions

6.1 Responsibility

These members are responsible for Database Administration.

Role	Name	Responsibility	E-Mail
Database Administrator	Nadimpalli Anand Varma	To ensure that data is protected, from loss and corruption and easily accessible as needed.	anandvarma_b180373cs@nitc.ac.in
System Administrator	Mandu Kireeti Nayak	Ensuring that the peripheral systems are in proper working condition.	kireeti_b180443cs@nitc.ac.in

6.2 Database Identification

Element	**Element Name**	**Description**
db_name	library	*Library* is the name of the database when it is originally created.
db_path	C:\xampp\mysql\data\library	The full path to where the database is stored on the system.

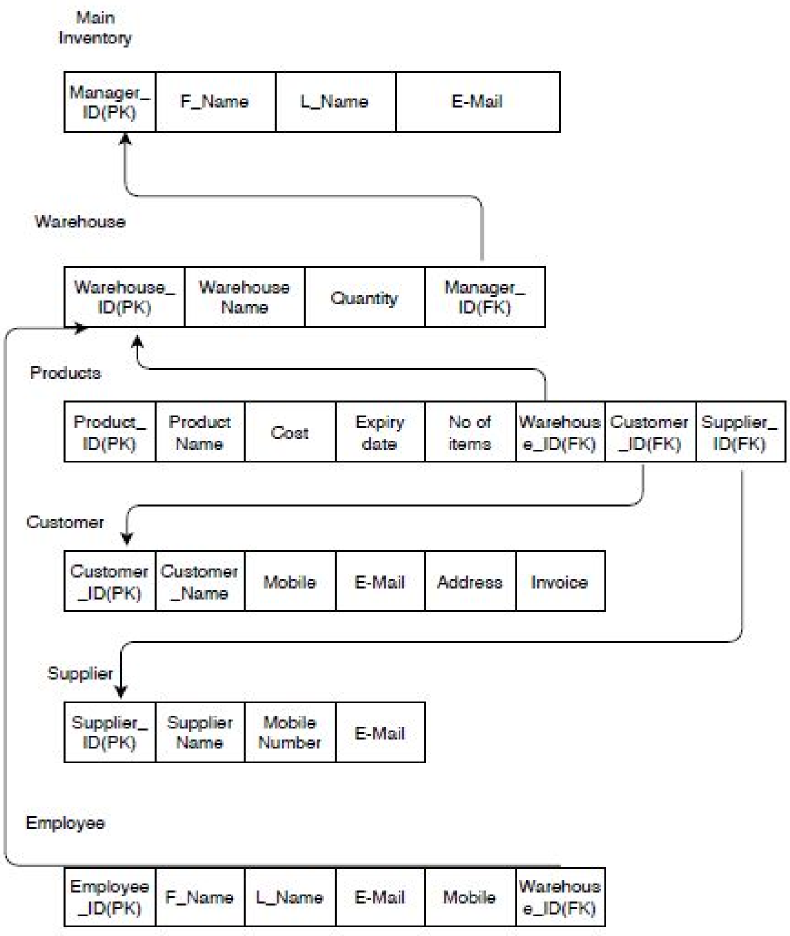
6.3 Application/System Using the Database

The database inventory created and accessed through phpmyadmin is used only by web application inventory management system there is no specific version for it.

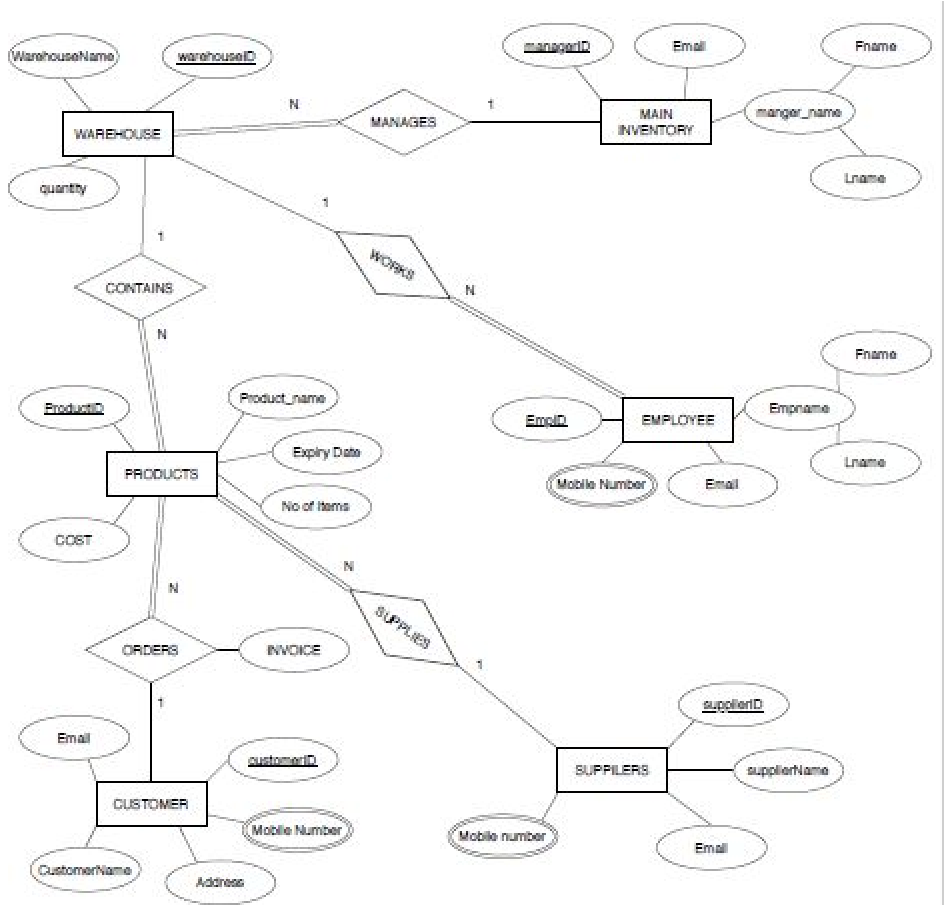
6.4 Relationship to Other Databases

Our application has only required only one database so there can’t be any relationship with another database.

6.5 Relational Schema Design



6.6 Entity Mapping



6.7 Normalisation of Database

1. It is a technique used to remove redundancy in the database and to maintain consistency and integrity.

- It is used to remove certain anomalies in the database in order to maintain consistency.
- Normalisation of above relational schema design:
- Since, Mobile number is a multivalued attribute, if the value in the column of mobile number is more than one then the tuple should be divided into two to follow 1NF.
- All conditions until BCNF are satisfied.

6.8 Mapping Rules

Rules for mapping entries into tables:

Data type of the entry should be same as the data type of the column For example:
a column named id(int) will accept only integer data types

1. The length of any accepted value will be the same as the length defined for it; no additional characters or numbers will be inserted into the table.

Example: in table we defined name (varchar) of length four and suppose entry contains a name “xyzru” only xyzr will be mapped into table and “u” is ignored.

2. on the backend in php or any other language the values to which the names on the front end are inserted should be the names of tables and should not be different.

Example: Name is inserted into username in php then all the values entered for Name will be inserted into user name

6.9 Data Transfer Requirements

1. All the stored data can be transferred by following the below steps:

    1. Start Xampp server.
    2. Open php my admin.
    3. Go to the database for which data need to be transferred.
    4. Select export and export data in sql extension.
    5. Open any other database and import the data.

6.10 Storage

Phpmyadmin\apache server provided by XAmpp is used as a database management system for the application .Database is server type so there is no certain limit for storage of tables in the database.

6.11 Backup and Recovery

For backup in case server fails or pc is damaged a backup shall be created by following steps:

1. Go to the root directory where the xampp folder is located.
2. Go to the mysql folder in the xampp folder.
3. Go to bin folder
4. Go to sql.
5. Copy the path.
6. Open the terminal and go to the directory where xampp is located.
7. Paste the path.
8. Type the command “mysqldump -u root -p portal>backup.sql”
9. Execute the command
10. A backup will be created.
11. Save the backup file.
12. For recovery start xampp server open mysql\php my admin
13. Go to export select the backup .sql file and click export
14. Data will be recovered

7 Detailed Database Design

7.1 Data Software Objects and Resultant Data Structures

This section of the document explains the entities used in the project, their attributes and how they will work together. Basically, this is intended to make the design more easy and understandable for everyone.

Entities:

1. Main inventory.
2. Warehouse.
3. Products.
4. Customer.
5. Supplier.
6. Employee.

Main Inventory:

Managed by a manager and has multiple warehouses associated with it (Manager is Admin of Database).

Attributes:

Name	Data Type	Type
Manager_ID	integer	Primary key attribute
F_Name	string	Non key attribute
L_Name	string	Non key attribute
E-Mail	string	Non key attribute

Warehouse:

Attributes:

Name	Data Type	Type
Warehouse_ID	integer	Primary key attribute
Warehouse Name	string	Non key attribute
Quantity	string	Non key attribute
Manager_ID	integer	Foreign key attribute

Products:

Attributes:

Name	Data Type	Type
Product_ID	integer	Primary key attribute
Product Name	string	Non key attribute
Cost	integer	Non key attribute
Expiry Date	Date	Non key attribute
No of items	integer	Non key attribute
Warehouse_ID	integer	Foreign key attribute
Customer_ID	integer	Foreign key attribute
Supplier_ID	integer	Foreign key attribute

Customer:

Attributes:

Name	Data Type	Type
Customer_ID	integer	Primary key attribute
Customer Name	string	Non key attribute
Mobile	integer	Non key attribute
E-Mail	string	Non key attribute
Address	string	Non key attribute
Invoice	string	Non key attribute

Supplier:

Attributes:

Name	Data Type	Type
Supplier_ID	integer	Primary key attribute
Supplier Name	string	Non key attribute
Mobile Number	integer	Non key attribute
E-Mail	string	Non key attribute

Employee:

Attributes:

Name	Data Type	Type
Employee_ID	integer	Primary key attribute
F_Name	string	Non key attribute
L_Name	string	Non key attribute
E-Mail	string	Non key attribute
Mobile	integer	Non key attribute
Warehouse_ID	integer	Foreign key attribute