Invoice Database

Version Alpha

10/17/2020

Table of Contents

[1. Introduction 1](#_Toc432497654)

[2. Overview 2](#_Toc432497655)

[3. Assumptions/Constraints/Risks 3](#_Toc432497656)

[3.1 Assumptions 3](#_Toc432497657)

[3.2 Constraints 3](#_Toc432497658)

[3.3 Risks 3](#_Toc432497659)

[4. Database Administration and Monitoring 5](#_Toc432497669)

[4.1 Roles and Responsibilities 5](#_Toc432497670)

[4.2 System Information 5](#_Toc432497671)

[4.2.1 Security and Privacy 5](#_Toc432497674)

[4.3 Performance Monitoring and Database Efficiency 6](#_Toc432497675)

[4.3.1 Data Transfer Requirements 6](#_Toc432497677)

[4.3.2 Data Formats 6](#_Toc432497678)

[4.4 Backup and Recovery 6](#_Toc432497679)

## Introduction

The database that I am proposing is an invoice database that keeps track of invoices, payments, customers names and other important information. Having a database that can track invoices is crucial for any business in order to provide sales and audit information, as well as being able to recall past invoices to have listed items bought. What is important for this database to work is to have several tables that can pull customer information, invoices, and details of each invoice relating to products that are sold.

## Overview

The overview of the database will be a invoice database that has customer information listed within the tables, tables with product information, and a table with invoice information.

The database that is being proposed is basically like an inventory and sales warehouse to store and collect information as needed and to be able to be accessed whenever it is needed. The database will have fields that specify invoice numbers, dates, products sold, the customer, and a total field that sums up the prices of the products, and of course price fields.

The customer information tables will have customer information for the customers who have bought products in the past or are placing orders for products. This database will contain information like first name, last name, address, city, state, customer ID, phone number and email address. This information will be used to contact the customer. The last table that needs to be implemented is products. Information implemented will be product Id, name of product, inventory levels, and price.

## Assumptions/Constraints/Risks

### Assumptions

The server that we will be using will be ran on a Windows server 2019 and should be able to be accessed by users that have the right privileges to make changes to the database. I assume that the initial size of the database should be able to accommodate the average sales of a small business to a medium sized enterprise and configured for higher use if the company wants to adjust the size. The database should accommodate up to 10GB starting out. The database will need to be able to log in orders around the clock and be able to roll back transactions in the case that an order gets canceled. The transactions of the database must be able to be deleted and created. Response times should be as little as possible in order to retrieve and process database objects. Configured to eastern U.S. Time.

### Constraints

The constraints on these tables will have to have unique identifiers using customer Id numbers, Invoice Id, and product Id as well. Other dependencies that needed to be taken into considered are regional laws, privacy, and any auditory laws that has to do with the sale of products. Other constraints that need to be considered will be availability of the database including downtime, maintenance, or other functions that could render the database unusable for some time.

### Risks

The risks associated with this database can be associated with the connections the database has with computers. To mitigate this problem we can use encryption, authentication, strong passwords, and other security measures to prevent man in the middle attacks. Another problem can be how many connections are connected to the database that can cause denial of service attacks, which can be prevented by limiting number of connections for users as well as implementing firewalls to prevent flow of bad traffic.

Another risk is having a hacker elevate privileges to alter the database or delete it, several measures will have to be implemented including authentication, removing users no longer active, and disabling full access to accounts. Data inside the database needs to be unique and not duplicated which can happen at any time.

## Database Administration and Monitoring

### Roles and Responsibilities

* Database Administrator- has access to modify tables, insert or correct fields (UPDATE), delete information. Assist problems that are occurring within the database.
* System Administrator- Maintains the systems for which the database is running on.
* Security Administrator- provides security functionality to the systems and the database. Also responsible for maintaining security to servers, installing firewalls, regulating traffic and connections to the server, disabling unused ports, updating computers firmware, and other security related functions.
* General users- specific named users who can access the database to view only. Any information needed to be updated must be passed to a higher-level administrator.

### System Information

#### Database Management System Configuration

Server will be ran using PostgreSQL version 12.4. Security and Privacy

The users needing to access the database will be implemented with least privilege. Access the physical database will be also only allowed for allowed employees. Access to the database will be protected physically behind metric scanners, fences, access cards, security cameras, guards, and locks require a key.

The privacy of the information on the database must be in compliant with federal and state regulations including industry best practices, financial regulations, and privacy of customer data. The information presented in the database is only accessible by the roles that has proper authentication.

### Performance Monitoring and Database Efficiency

Postgresql has several monitoring tools that can get information on statistics of query times and allocation, data usage, size limits and resource consumption. Having several administrators do nightly/weekly checks in order to keep all time performance high while making sure that the database will continue to run smoothly.

#### Data Transfer Requirements

Any data transferred from Point-Of-Sale systems (POS), must be kept in a secure data line, data on prices has to be in a monetary decimal format ($0.00), transaction dates must be in a date format (1/11/01). Any information transferred over computers to the database will be protected using SSL.

### Backup and Recovery

Instructions: Describe required strategies and scheduling for periodic backups of the data. If certain objects have differing requirements, provide a breakdown by object. Describe the methodology for reestablishment or recreation of the necessary data schema and system support files.

The availability for the database must be consistent, with back up servers and the original database inaccessible to the public. IT support for the database must be available with as little downtime as possible. With the use of multiple servers providing RAID for redundancy and striping, the recovery of any server that goes down will be recovered with as little downtime as possible. Maintaining the servers as well as the data inside the database will be also persistence for logging and auditing purposes.

Backing up the servers should be performed on a nightly basis using incremental back up that adds information added from the last back up, as well as roll back to dates before problems occurred.