SERIOUS BUSINESS

Unremarkable Inventory Management

CST 363 Intro to Database Systems - Project 1 Part 2

Designed & Developed by:

Julie Asato
Matthew Robertson
Raymond Talmage

Part 1

Describe the application

Our MySQL application is an inventory management tool for personal use. It allows users to make an inventory list of items they want to keep. Additionally it keeps track of how many they need as well as how many they possess. It has a suppliers list and catalog to help users in knowing where find previously purchased items and how much they should expect to pay for that item.

Information needs

This application at its core only needs a list of items, how many of each are on hand and how many are needed. We extended the functionality to maintain a list of suppliers and a catalog of items, to aid in future price evaluations and purchasing of items as needed.

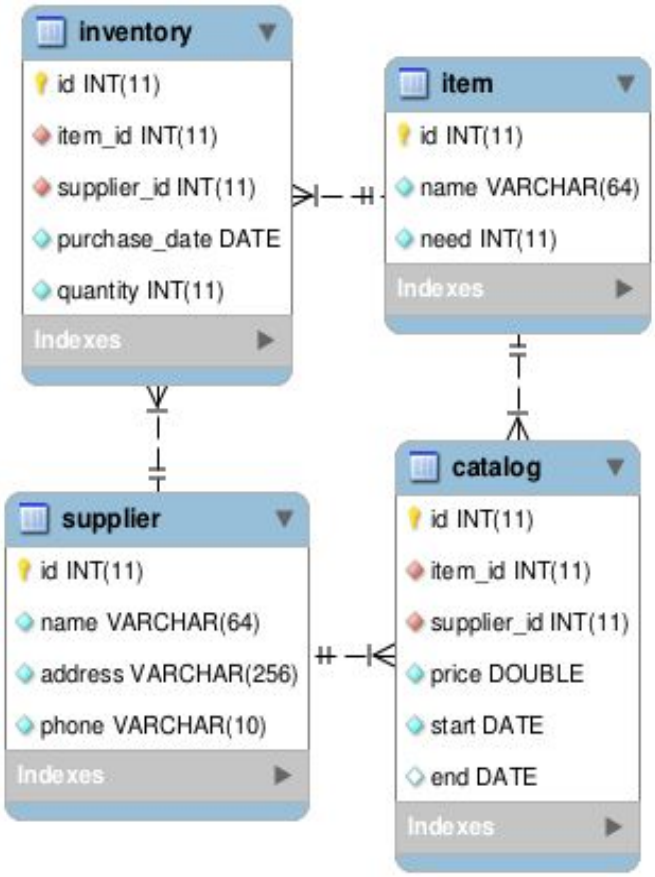
Userbase & User interactions

The intended user base for this application is everyday consumers who want to keep track of their purchases and inventory. A user interacts with the application through the app.html page. There they can browse the inventory, suppliers or catalog. They can also add an item, add a supplier, add an item to the catalog or add an item to the inventory.

Database Structure

The database backing our application contains four tables: inventory, item, supplier and catalog. The item table has two columns: id, name. It is meant to store the items the application is aware of, regardless of where it is sold, how much it costs and how many are on hand. The supplier table has four columns: id, name, address, and phone. This table maintains a list of suppliers and their contact information. As an enhancement, the address could be used in the future along with google maps in the future to calculate travel distances for the user. The catalog table has four columns: item_id, supplier_id, price and discontinued. It keeps track of purchasable items, their prices and availability at particular stores. The inventory table has three columns: item_id, need, have. This table is holds data for the core functionality of the application.

E-R diagram of the database model

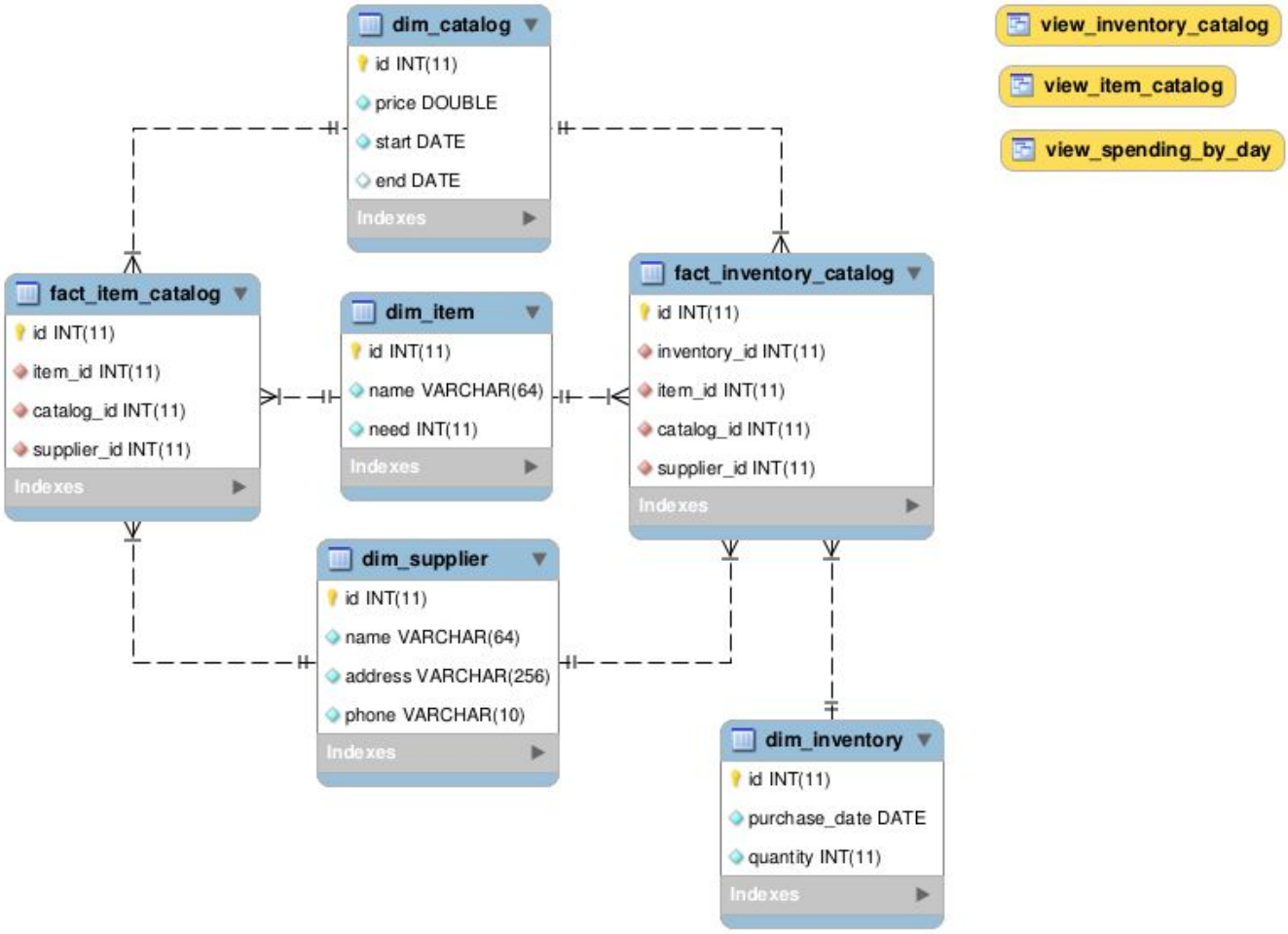


How is the design normalized?

It is normalized in the 3rd normal form model because the data is divided into discernable parts of a whole and there is no redundant data. The design uses foreign keys from the item and supplier tables to reduce data redundancy. With a normalized structure, we minimize our disk usage. Also, each of our tables represents a distinct entity or concept and their data is only shared by foreign key.

Part 2

Star Schema design of the warehouse database



Instructor Feedback

The login table was mistakenly left in the folder after being turned in. There was no intention of including a login page for the project.