**Capstone 1 Project Requirements**

Design and build an application using Spring Boot and JPA backed by a SQL database that meets the following criteria:

Meets all client requirements

Employs TDD in every phase of architecting the project

Provides highly usable documentation

Normalizes all database tables to 3rd normal form

Employs at least 1 join

**Custom Cars**

A client is running a custom car manufacturing firm. They hand build custom cars to order. They work with 5 different suppliers who all provide different parts. Supplier A provides tires, brakes, rims, exhaust, and bumpers. Supplier B provides radios, speakers, miscellaneous electronics, upholstery, and trim. Supplier C provides fiberglass, steel, and molded plastic parts. Supplier D provides engines, differentials, catalytic converters, fuel pumps, and axles. Supplier E provides radiators, fans, belts, compressors, and heater cores.

The client needs to be able to accomplish the following tasks:

Add new type of item to inventory and associate it with a specific vendor.

Add or update vendor information (name, email, phone)

Update type of items in inventory (part name, associated vendor, quantity in stock, price per item)

Add a new individual item to the inventory (each type like brake can have hundreds of individual items each with a unique serial number. If we have 50 brakes in stock, there should be at least 50 entries in table of individual parts with brake as their type and a unique serial number for each)

Delete individual items from inventory (when a new car is built, it should be associated with the requisite parts by serial number and those should be marked as “used” in a status column, they should not be removed from the database. Items will be deleted if they are lost or defective)

Search for low inventory (any item with less than 50 remaining in stock).

Add new orders with a start date and completion date (this should automatically decrement inventory by the specified amount on the start date - this requires scheduling a task to run on a given date. This should also associate individual parts serial numbers with this build and mark those parts as “used” in the individual parts table).

Delete orders (should mark any previously associated parts as unused).

Update quantity in stock of any part.

DATABASE:

CREATE DATABASE custom\_cars\_db

USE custom\_cars\_db

CREATE TABLE vendors(

vendor\_id INTEGER AUTO\_INCREMENT NOT NULL,

vendor\_name VARCHAR(20)NOT NULL,

email VARCHAR(25) NOT NULL,

phone VARCHAR(15) NOT NULL,

PRIMARY KEY (vendor\_id)

);

CREATE TABLE orders(

orders\_id INTEGER AUTO\_INCREMENT NOT NULL,

start\_date DATE NOT NULL,

completion\_date DATE NOT NULL,

part\_serial\_number INTEGER NOT NULL,

new\_car\_number INTEGER NOT NULL,

PRIMARY KEY (orders\_id)

);

CREATE TABLE new\_cars(

new\_cars\_id INTEGER AUTO\_INCREMENT NOT NULL,

part\_serial\_number INTEGER NOT NULL,

PRIMARY KEY (new\_cars\_id)

);

CREATE TABLE individual\_parts(

part\_serial\_number INTEGER AUTO\_INCREMENT NOT NULL,

part\_name VARCHAR(25) NOT NULL,

quantity INTEGER NOT NULL,

vendor\_id INTEGER NOT NULL,

price DOUBLE NOT NULL,

state BOOLEAN NOT NULL,

-- status can only be: “used” or “lost/defective”

PRIMARY KEY (part\_serial\_number)

);

CREATE TABLE suppliers(

supplier\_id INTEGER AUTO\_INCREMENT NOT NULL,

part\_serial\_number INTEGER NOT NULL,

vendor\_id INTEGER NOT NULL,

PRIMARY KEY (supplier\_id)

);