**Statement of Work (SOW)**

**Justin Hughes**  
DBST651, *Fall 2017*

**Overview**:

I will create a database for my company to enable complete product inventory management and order tracking for our customer base. This database will allow for product tracking all the way through the life cycle from ordering to current inventory.

**Purpose and Objective**:

The database will allow the storage and tracking of all product inventory for an individual machine, as well as provide ordering capabilities and tracking based on those inventory levels. This will enable the customer and our staff to look at currently inventory (including the product, lot number, and quantity). The database will also allow the generation of orders based on the inventory levels, store the history of all orders placed and the contents of those orders, and the product database containing all potential products and their vendor ID’s.

**Scope**:

There will be 5 different tables created in total for this database. Those tables include CUSTOMER, ORDERS, PO\_CONTENTS, INVENTORY, and PRODUCTS. These tables will be created for use in a relational database model with multiple relationships between the various tables.

**Diagram Tool**:

ER Assistant

**Database**:

I will be using an Oracle 12c Database using SQL Developer

**Hardware and Software**:

UMUC Virtual Desktop Connection with Intel® Xeon. Operating system is Windows 7 Enterprise Edition.

**DDL and DML**:

Oracle SQL (Structured Query Language) will be used for the DDL and DML. For the DDL scripts, I will use SQL to transcribe the entities modeled in the ER diagram (ERD) into database table creation scripts. For the DML scripts, SQL will be used to insert or remove entries into the various database tables.

**Business Requirements**

The purpose of this document is to outline the entities required for the database to monitor customer inventory and ordering, as well as the relationships between those entities. These entities, their attributes, and the mentioned relationships will also be modeled in a supplemental ERD file.

**Entities**

* Entity Name: CUSTOMER
  + Description: Houses all information about an individual customer
  + Primary Key: CUSTOMER\_ID
  + Foreign Key: None
  + Parent Entity: None
* Entity Name: ORDERS
  + Description: Houses information for all orders placed for that customer
  + Primary Key: ORDER\_ID
  + Foreign Key: CUSTOMER\_ID (from CUSTOMER)
  + Parent Entity: CUSTOMER
* Entity Name: PO\_CONTENTS
  + Description: Houses the contents of an individual purchase order (PO)
  + Primary Key: VENDOR\_ITEM
  + Foreign Key: ORDER\_ID (From ORDER)
  + Foreign Key 2: NDC\_NUM (From PRODUCTS)
  + Parent Entities: ORDER and PRODUCTS
* Entity Name: PRODUCTS
  + Description: Houses information on all potential products that can be ordered and used by our customers
  + Primary Key: PRODUCT\_ID
  + Foreign Key: None
  + Parent Entity: None
* Entity Name: INVENTORY
  + Description: Houses information on current product inventory for an individual customer
  + Primary Key:
    - Note: Unique INV\_ID will be generated for each new combination of Product ID, Lot Number, and Expiration Date
  + Foreign Key: PRODUCT\_ID (From PRODUCTS)
  + Foreign Key 2: CUSTOMER\_ID (From CUSTOMER)
  + Parent Entities: CUSTOMER and PRODUCTS

**Relationships**

* 1:M between CUSTOMER and ORDER
  + Business Rules: Any customer can have zero, one or many orders; an order will only belong to one customer
* 1:M between ORDER and PO\_CONTENTS
  + Business rules: An order must have one or many content items; A content item will only belong to one order
* 1:M between PRODUCTS and PO\_CONTENTS
  + Business Rules: A Product ID has to match up with one or many NDC numbers in an order’s contents; an NDC number on an order’s contents will only match up to one Product ID
* 1:M between PRODUCTS and INVENTORY
  + Business Rules: A Product ID can have zero, one, or many Inventory ID’s associated with it; an Inventory ID will match up to only one Product ID