# Stage 1 Project Requirements –Data Requirements and ERD

## 50 Total Points. Submit an e-copy to canvas or submit a hard copy before class.

**Please review Prj0 requirements for Project Size requirements, project originality requirements, and unsuitable projects. Note that even though I approved your project, what I had in mind when I read your proposal may be different from what you had in mind. So make sure your project satisfies requirements stated in Prj0 with respect to size, originality, and suitability.**

# Project Size Requirement

The ideal size of the project is 6-8 tables.

**Step 1):** (15 Points) Discuss information requirements for the database application, addressing the following points.

* (0.5 point) Discuss background of the company and the industry they are in.
* (0.5 point) Discuss problems with the current way of doing business or with the current system, problems with obtaining timely and accurate vital information. Your database may not be able to resolve all problems identified here. Specify the problems your database is trying to solve (i.e.; scope of your solution).
* (4 points) Provide detailed descriptions of business operations that will need or generate data for which you are designing the database. The write up here serves as the natural language description of data requirements. This is not the place to describe your tables or what the attributes are for the tables. This is where you describe business processes and business rules.
* (8 Points) Use the four sentence templates discussed in class to describe detailed data requirements that mostly match the data requirements descriptions in the last bullet. The sentences should start with “Database needs to store.” For cardinality of each relationship, start the two sentences with “each.” See the third bullet below. The descriptions of data requirements should be such that another person (say, your instructor) can draw an ERD based on your descriptions. You may want to use bulleted statements to clearly state data requirements. Provide an example instance of each entity as well as for attributes of relationships. Here are some examples:
  + (For entities, attributes, and primary keys). Database needs to store information about each Order. For each order, database needs to store OrderId, OrderDate, and Total dollar amount. OrderId is unique to each Order. Order example: Order1234 ordered on 3/1/2015, with a dollar amount of $540.
  + (For relationships) Database needs to store what products are being ordered in each order.
  + (For relationship cardinalities): Each order can contain multiple products. Each product can appear in multiple orders. For example, Order1234 ordered SKU78 and SKU72; Order34 ordered SKU72 and SKU83.
  + (For relationship attributes): Database needs to store quantity and price of each product ordered in each order. (Or, for each product ordered in each order, database needs to store quantity and price.) For example, Order 1234 ordered Product SKU78 at $78/unit for 5 units. Order1234 also ordered SKU72 at $60/unit for 9 units.

Your write-up may benefit from an iterative process whereby you compare your ERD with your write-up and revise the write-up if you find that certain information is not contained in the write-up to allow another person to derive attributes, primary keys, or cardinalities, or if you find that your ERD does not match your write up. Give examples of data elements if those elements are specific to the application environment, because other people (e.g., your instructor) may not know what they are (see the next requirement).

* (2 points) Provide sample data for all data requirements as part of the submission. If the application environment uses any paper forms to collect data, fill out a set of forms with fictitious information as sample data and attach those forms as part of the submission. If the application environment does not use paper forms, provide screen captures or otherwise provide sample data (made up data are fine). If your project is made up, provide made up forms or data.

**Step 2):** (20 Points) Draw an E-R diagram to document data requirements of your application according to your descriptions provided in Step 1), including the mapping ratios for relationships. The ERD you draw here should match your descriptions in Step 1, Bullet 4. Clearly identify attributes of entities and relationships. Each entity must have a primary key and necessary attributes.

* To save on diagramming effort, you may want to list all attributes of an entity inside the box for that entity.
* For each entity, underline or otherwise indicate the primary key.
* If writing out the attributes inside the box crowds the diagram, list the attributes for each entity separately, beneath the diagram if there is room, or on a separate page.
* If the meaning of an attribute is not self-explanatory, add explanations.

**You may hand-draw your ERD, or you may use any drawing tools.**

**There are many free online drawing tools available, for example,** [**http://www.draw.io**](http://www.draw.io) **(use a template under Software Design). ERDplus.com is another good tool. If you’re not using Visio, Take a screenshot of your ERD and insert the screenshot as an image in your Word Document. If you’re using Visio, insert the ERD as an editable image. The solution file for ERD exercises posted in Canvas were drawn with Visio. You can copy one and modify it for your own ERD. It would save you some time this way.**

**Step 3) (15 Points)** Transform the E-R model into a relational data model. Report the following 6 steps. Follow the example seen here.

1. Convert every entity into a table. Use underline(s) to identify the primary key of a table. See Example a) on the next page.
2. Convert all 1-n and 1-1 relationships. Clearly indicate attributes added as a result of converting a 1-1 or a 1-n relationships. *Use italic fonts for this purpose.* See Example b) on the next page.
3. Convert all m-n relationships. See Example b) on the next Page.
4. List the final conversion result. See Example c) on the next page. Use underlines to indicate primary keys. Identify foreign keys in each table.
5. After you have listed the tables based on the ERD, determine which normal form each table is in. If a table does not satisfy BCNF, either normalize it or clearly state the reasons why you choose to keep the table in a lower normal form. See Example d) on the next page.
6. List the final normalized tables with primary keys and foreign keys identified. See Example e) on the next page.

# Example: Project ERD is below and the conversion process follows.



**a): Convert all entities and get:**

Vendor: (vendorId, vname, address, phone)

Order: (order#, Orderdate)

Product: (Stock#, price, Desc)

Payment: (PaymentNo, Pdate, Amount)

**b): Convert all relationships**

* **Vendor-places-Order Relationship:** This is a 1-n relationship with an attribute. Add the primary key of "1" entity (Vendor) to the many side (Order) as a foreign key. Relationship attribute is added too. Get:

Order: (order#, Orderdate, *VendorId, OrderSource*)

**Foreign key**: VendorId

* **Order-Contains-Products Relationship:** This is anm-n relationship with attributes, so make a new table, OrderDetails:

OrderDetails: (Order#, Stock#, Quantity, Price)

**Foreign keys**: Order#, Stock#

* **Payment-pays for-Order:** This is a 1-1 relationship. Since each payment must have a corresponding order but each order may not have a corresponding payment yet, to advoid null values, add order# (“1” side) to Payment (pretend “many” side). Get:

Payment: (PaymentNo, PDate, Amount, *Order#)*

**Foreign key**: Order#

**c): Final Conversion Result:**

Vendor: (vendorId, vname, address, phone)

Order: (order#, Orderdate, *VendorId, OrderSource*) **Foreign key: VendorId**

Product: (Stock#, price, Desc)

Payment: (PaymentNo, Pdate, Amount, *Order#)*. **Foreign key: Order#**

OrderDetails: (Order#, Stock#, Quantity, Price). **Foreign keys: Order#, Stock#**

**d): Normal Form Determinations:**

All Tables are in BCNF.

**e): Final Result after normalization**

Vendor: (vendorId, vname, address, phone)

Order: (order#, Orderdate, *VendorId, OrderSource*) **Foreign key: VendorId**

Product: (Stock#, price, Desc)

Payment: (PaymentNo, PDate, Amount, *Order#)*. **Foreign key: Order#**

OrderDetails: (Order#, Stock#, Quantity, Price). **Foreign keys: Order#, Stock#**

# Important

Drawing ERD and converting it to a list of tables is an iterative process. After drawing the ERD, it’s beneficial to convert that ERD into a list of tables and then investigate whether the final tables make sense. This often will give you feedback on your ERD.