**CSE 3241 Project Checkpoint 01**

**Entities and Relationships**

Names (team members): Date:

**1 - In a NEATLY TYPED document each team needs to submit (Upload it to Carmen) the following document by the Checkpoint 1 due date, providing the following:**

1. Based on the requirements given in the project overview, list the entities to be modeled in this database. For each entity, provide a list of associated attributes (entities usually have multiple attributes).

WAREHOUSE

EQUIP\_CAP, ADDR, CITY, MANAGER, PHONE, DRONE\_CAP

EQUIPMENT

DESC, MODEL\_NUM, SERIAL\_NUM, YEAR, WARR\_EXP, MANUFACTURER, ARRIVAL\_DATE, TYPE, INV\_ID, WEIGHT, SIZE

DRONE

DESC, MODEL\_NUM, SERIAL\_NUM, YEAR, WARR\_EXP, MANUFACTURER, FLEET\_ID, WEIGHT\_CAP, VOLUME\_CAP, DIST\_AUT, MAX\_SPEED, STATUS

MEMBER

FNAME, LNAME, USER\_ID, ADDR, PHONE, EMAIL, START\_DATE, WH\_DIST

1. Start thinking and visualizing the database system (software application) that your team will implement. You need to analyze the use cases described in the Project Description document (see project assignment on Carmen) and provide a high level design of the application user interface to provide to the users. No coding at this point is necessary.
2. Assuming a text-based user interface, design and provide the different menus and submenus needed and show how the user would navigate through the different options.

We are going to implement a text-based navigation menu that allows the users/admins to navigate and perform the actions they need. For example, on the “home page”, both the admin and user view will have a list of submenus they can enter via a command to perform a specific task.

1. Define a basic functionality for your application, e.g. add, modify, or search any of the important data items in the data base. Use a text-based user interface (graphical UX is optional, if you’re not familiar opt for a text-based menus on the text console). Note: you could use Word, Powerpoint, etc to define menus/submenus)

A basic example of a user task would be viewing their order history. The user would complete this task by executing the necessary commands to enter this submenu and to perform the action.

1. No need to write any code yet. No need to provide any code at this point.

Recommendation if you haven’t written any code in the last semester(s): refresh your Java skills and try to implement the menu/submenus and the basic navigation on the following weeks.

1. Based on the requirements given in the project overview, what are the various relationships between entities? (For example, “USER entities rent EQUIPMENT entities”, “ORDER entities contain EQUIPMENT entities”, etc.)

MEMBER places EQUIP\_ORDER

EMPLOYEE places INV\_ORDER

DRONE delivers EQUIPMENT

WAREHOUSE stores INVENTORY

MEMBER writes REVIEW

1. Propose at least two additional entities that it would be useful for this database to model beyond the scope of the project requirements. Provide a list of possible attributes for the additional entities and possible relationships they may have with each other and the rest of the entities in the database. Give a brief, one sentence rationale for why adding these entities would be interesting/useful to the stakeholders for this database project.

EMPLOYEE

SSN

SALARY

Adding an employee table would allow the company to track human resources as it grows.

ORDER

DATE

DESC

NUM\_ITEMS

VALUE

EST\_DOA

ORDER\_NUM

Adding a superclass to track both inventory orders and customer orders would allow the preservation of order history for both inventory types.

1. Give at least four examples of some informal queries/reports that it might be useful for users to be provided by the database system, e.g. a query that provides all the drones that deliver equipment yesterday or a report of all customer with pending deliveries. Include one example for each of the additional entities you proposed in question 4 above. Note: A query is a question for specific information asked/put to the database.

List all of the employees that worked that week.

List all of the available equipment for a specific project type.

How many deliveries were completed on time.

How many customers ordered more than three pieces of equipment.

1. Determine at least three other informal update operations and describe what entities would need to have attributes altered and how they would need to be changed given your above descriptions. Include one example for each of the additional entities you proposed in question 4 above.
2. Adding a new piece of equipment
   1. Entities that need changed:
      1. WAREHOUSE: Would need to change what type of equipment is stored in each warehouse
      2. EQUIPMENT: Would need to change equipment types as there
3. Adding a new employee
   1. Entities that need changed:
      1. EMPLOYEE: Would need to add new tuple
4. Firing and employee
   1. Entities that need changed:
      1. EMPLOYEE: Would need to remove the employee from database or otherwise deactivate
      2. WAREHOUSE: Would need to update workers in warehouse if employee fired from it
5. Provide an ER/EER diagram for your database. Make sure you include ALL entities and relationships in question 1 above ***INCLUDING the entities for question 4 also***, and remember that ***EVERY*** entity in your model needs to connect to another entity in the model via some kind of relationship (No entity should be isolated). Consider the direction of each relationship, do not forget about specifying the type of relationship (1 to 1, 1 to N, N to 1, or N to N).

ATTACHED

1. Map your ER/EER model to a relational schema. Indicate all primary keys (underline) and foreign keys (you may use line/arrows to referenced attribute(s)). Note: if you find that your relational schema seems large or complex, you may want to iterate and go back to previous step and refine the ER/EER modeling and then re-map the changes into the relational schema.

ATTACHED