

E-R Diagram + Requirements Definition Document (30 points)

Note: This assignment should be completed as a team.

Background: In this assignment, you will complete the Entity-Relationship (E-R) diagram and Requirements Definition Document for your semester project. Use information gathered from interviews, emails, project description, use cases, etc. Remember, logical E-R diagrams are *system-independent*, meaning that you could later design and/or implement the systems you are proposing (and their underlying databases) using any number of different software packages (including any number of different database management systems).

Overall formatting and submission instructions: Submit two files: one file for Part 1, and one file for Part 2. Submit via TITANium by the listed due date.

Part 1: E-R Diagram

Instructions: Based on information you have received throughout the semester, you know that the system your client wants will need to store data about various things. Create an E-R diagram depicting how the database underlying the system could store the required information. Start by considering what are the “things” (or people or places) that you need to store information about and create entities representing them. Generic examples (NOT specific to your class project, though you may or may not have one or more of the following) of entities could be “inventory items”, “vendors”, “invoices”, “reservations”, etc. Then think about what characteristics or attributes you want to store about each of these things. Then put the appropriate relationships between them. For this assignment, you do not have to create intersection entities – you can leave M:N relationships if you have any.

If you need help or guidance specific to the types of information the system for your project will store (whether the instructor-based case or an outside real project), please feel free to contact me.

Part 2: Requirements Definition Document

Instructions: Now that you have used several tools and techniques for gathering system requirements (interview, followup discussions, project descriptions, use case analysis, DFDs and ERDs and the interaction with the project sponsor to create those, etc.), it is time to compile the requirements into a formal list—the Requirements Definition Document.

Use the examples in the textbook (Ch 3) as templates. Organize your functional requirements into different logical categories (NOT information vs. process, but in another meaningful way specific to the project you are working on and its major pieces—again, see textbook for examples). By this point in the project, the functional requirements should be fairly comprehensive as you should have a good idea from interaction with the sponsor what the main functions of the system will be. Organize your nonfunctional requirements into the four categories listed in the textbook—operational, performance, security, and political/cultural.

You should list at least one requirement in each category, though some categories will have many more than that. (As a side note, the nonfunctional requirements list might be small now, and will grow as we talk more about nonfunctional requirements during the design phase.) I would expect at least 10-20 listed system requirements (functional and nonfunctional together) in total, though there might be a wide range depending on the project you are working on.

Grading Rubric:

Part 1 – Data model (E-R diagram)	16 pts
Correct entities (i.e., no major entities missing based on previous project information)	4 pts
Meaningful attributes listed for each entity	4 pts
Correct relationships between entities	2 pts
Correct maximum cardinalities on relationships (some relationships may have multiple correct answers in this regard – others will not)	2 pts
Correct minimum cardinalities on relationships (some relationships may have multiple correct answers in this regard – others will not)	2 pts
Meaningful identifiers (i.e., primary keys)	2 pts
Part 2 – Requirements Definition Document	10 pts
Several functional requirements listed	3 pts
Functional requirements are organized into meaningful categories	2 pts
Several nonfunctional requirements listed	2 pts
Nonfunctional requirements are divided into the correct categories	1 pt
At least one nonfunctional requirement in each of the four types	2 pts
General formatting	4 pts
Diagrams are worded clearly with no grammatical errors	2 pts
Diagrams are formatted in a way that is easy to read and understand	2 pts
Total	30 pts