

CSE 5330/7330 Fall 2018
Project Definition
Do your own work. Reference any material used.

This project is to be completed by each student individually. In addition, the project is to be completed in 3+1 phases. At the end of each phase you are to submit that portion of the project for grading. At that time, you will receive information needed to complete the next phase of the project.

Any DBMS may be used for the project as long as a SQL interface is used. As the last phase of the project will require that Dr. Moore be able to test some unseen queries against your database, he must have access to your database. This can be done by supplying a database dump suitable for importing, or scheduling a time for hands-on testing.

You are to design and implement a database to keep track of building inspections required by the department of Faulty or Damaged Buildings (FODB). Your user contact (and person to approach with any questions) is Dr. Moore.

FODB coordinates building inspections requested by builders on a monthly basis. Each building inspection is either passed or not. Inspections have a type code (3 characters, e.g. PLU, FRM, ELE, etc.) and possibly sequencing requirements. Some inspections cannot be performed before other inspections, e.g. final plumbing inspection cannot be performed until the framing inspection is passed. Each inspection has a numeric score, with 75 or more out of 100 being sufficient for a pass status. Each inspection data contains the date of inspection, inspector identification, inspection score, and textual information about the inspection. The textual information can be updated later, but the score can never be changed. FODB maintains a pool of inspectors. Each inspector has a unique 5 digit employee ID, name, and date they were hired. They can conduct any type of inspection but can only perform at most 5 inspections per month. Any failed inspection can be repeated until passed. Particular information maintained about builders includes: Name (30 byte character string), address (40 byte character string), license# (5 digit number). A builder's license# is unique. A builder and location must exist prior to requesting an inspection. A request for an inspection may be assigned to any available inspector assuming the prerequisite inspections have a pass status.

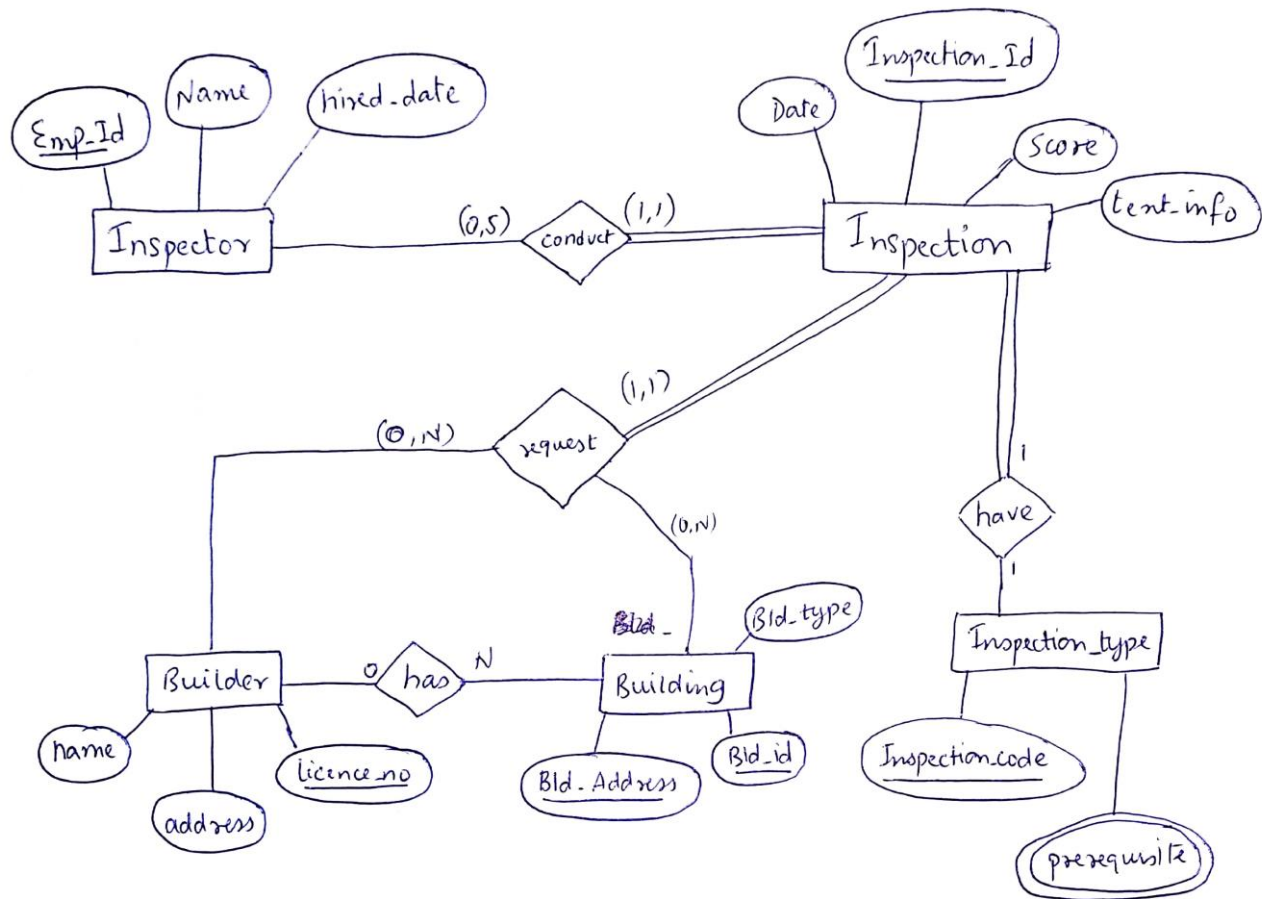
The actual applications to be run against the database have not yet been determined by the user; however you have been asked to start the development of the database to get it ready. Your assignment is divided into phases. At the end of each phase, you will be provided with any missing information needed to complete the next phase

Project Phases

1. ER Diagram and Initial Relational Design (25 pts; Due: 10/24):

- a. Construct an ER Diagram with attributes, being precise in your notation, including cardinality constraints. The ER diagram you create must support all requirements stated above. If you add any restrictions or information not stated above, please specify. Indicate any design requirements that are not included in the ER diagram.

Ans: Below is the ER Diagram:



- b. Produce an initial Relational schema. Given your ER diagram, provide an initial description of the tables you plan to create, identifying keys and foreign keys.

Ans:

Entities: Inspector, Inspection, Inspection_Type, Builder, Bld_Location.

Relational schema:

Inspector: Empid as Primary Key.

Inspection: Inspection_Id as Primary Key, Emp_Id as Foreign Key, Inspection_code as Foreign Key, Bld_Address as Foreign Key.

Inspection_Type: Inspection_code as Primary Key

Builder: Licence_no as Primary Key

Building: Bld_id and Bld_Address as Primary Key, Licence_no as Foreign Key.

Note: I have added the attribute Bld_type in Building table which says about the type of building like is it Commercial or Residential.

Inspector	
Emp-Id	→ Primary key
Name	
hired-date	
Inspection	
Date	
Inspection-Id	→ (Primary key)
Score	
text-info	
Emp-Id	→ Foreign key
Inspection_Type	
Inspection-code	→ Foreign key
Bld-Address	→ Foreign key
Inspection_Type	
Inspection-code	→ Primary Key
Prerequisite	
Builder	
name	
address	
licence_no	→ Primary key
Building	
Bld-Address	→ Primary key
Bld-Id	→ Primary key
Bld-type	
licence_no	→ Foreign key

c. Submit for grading your ER diagram and relational schema.

2. Database Implementation (25 pts; Due: 11/14):

- Using SQL DDL statements, create the relations as designed in phase 2. You must include any needed constraints or triggers to ensure design requirements are met.
- Populate the relations using data provided by your user.
- Submit for grading proof of creation of the relations, data, and triggers. This could be output of SHOW commands and SELECT * statements or a text file containing all commands to create and populate your database. Be sure to indicate your choice

of DBMS. Provide an updated ER diagram (if needed) highlighting the changes made during implementation.

3. Applications Development (25 pts; Due: 12/03)

- a. You will be provided a list of application requirements. The requirements will differ between CSE5330 and CSE7330.
- b. Using the relations populated in Phase 2, you are to create SQL code to implement a set of queries against the database.
- c. Submit for grading proof of execution of these queries.

4. Final Testing (25 pts; Due: 12/03):

- a. When you have submitted Phase 3 for grading, Dr. Moore will perform his own testing of the database.
- b. Access to your database (or sufficient detail to easily recreate it) will be necessary.
- c. The testing will differ between CSE5330 and CSE7330.

NOTES:

1. Requirements are subject to change at any time at the discretion of the user.
2. If you change details of an earlier phase implementation, please provide detail of this with your submission during the next phase.
3. You will receive a written grade on each phase while you are working on the next phase.