

Project, Part 1

CS 4347 Fall 2023

Due September 21st, 2023, 11:59pm

1 Instructions:

This assignment is a Cohort Assignment. Students will form a team, called a cohort, to complete this assignment. While everyone may work together and share information, each member of the cohort must submit their own copy of the project.

2 Objective:

To begin the modelling aspect of constructing a database for the Database Class. The goal of the project is to build the database, not the database application. To that end, certain phases must be completed. This phase consists of

- Cohort and proposal, 20 points
- Software discovery, 10 points
- An entity-relationship diagram, 35 points
- A dictionary of keys and attributes of the entity-relationship diagram. 35 points

3 Cohort and Proposal: (20 points)

For the term project, each cohort must create a Database. Each cohort must write up an English description of what they plan to create. The cohort's name will be used as a prefix for submitting documents, programs, and archive files.

3.1 The Cohort

3.1.1 How big should a team be?

The team should be between three to five people. Teams bigger than this tends to fall apart. Yes, you can attempt a team of 1 person, but it is not recommended.

3.1.2 How will the team function?

Once your cohort is formed, cohorts are expected to stay connected with each other. Github, Facebook, Discord, Groupme, Slack, and whatever the next cool software will be. In addition, Dr. Becker strongly recommends face-to-face meetings. Class time will be made available for cohort meetings.

3.1.3 Naming the Cohort and Writing the Description

Each Cohort should have a good, clear cohort name. I want a name you choose to be suitable to be put into the Blackboard gradebook. Blackboard has a 32-character limit, and yes, the cohort's name must be readable, useable characters. Such a name can be silly or serious, but it needs to be appropriate (syllabus), recognizable, and understandable.

3.2 Proposal

3.2.1 Basic Description:

Given these descriptions, write a proposal on how the cohort will build a database. What **assumptions** does your cohort make? This proposal must be written up in English in a style for non-computer people.

Example

Toys are big business today. Yes, Toys. Anything from a doll to a limited-edition maquette becomes big business.

People

Seller-someone buying or selling a particular toy.

Online Shop-An e-commerce website for selling toys.

Convention Market-a public gathering where toys are sold to eager fans and collectors.

Storefront-A regular storefront, with a cash register and cashier

Collector-someone who has a large collection and is downsizing or someone with a desire to own a particular toy.

Relative-parent, aunt, uncle, or sibling that might be buying the toy as a present.

Friend-a non-relative buying the toy as a present.

Types of Toys

Dolls and Action Figures-everything from infants to superheroes to fashion dolls

Vehicles-cars, planes, submarines, starships!

Playsets-playmats, buildings, secret caves, shops, and kitchens

Building Kits- building blocks to plastic model kits

Dioramas-shelf nooks, full-scale model train sets, crystal grottos

Educational-fossils, volcanos, medical instruments

Methods of Payment

Cash-if the seller has a cash register.

Check-still desired to avoid paying the with credit card fee.

Credit-3% and they will secure the transaction.

Online Service-various online banking tools

Cryptocurrency-various "coins" that might be accepted.

Consider Security

Accounts-shop accounts and monetary accounts

Passwords-a database of passwords

Roles – who has the right to see the transactions.

Captchas-Is the buyer human?

Verifications-Does the buyer or seller have a cellphone to confirm their identity?

3.2.2 Additional

Any cohort can create their own database on a different topic, but it must be as complicated as the example above.

3.3 For this section

Turn in a PDF with the following sections:

- The name of the cohort
- The reason for the name of the cohort.
- A listing of the members of the cohort
- The project description, even if you are just using the given example. Describe what you are going to build in English.
- Assumptions: ideas and observations about Entities and Relationships
 - This should not be blank! It could be as short as half a page or many pages long depending on how creative your cohort is.
 - Assumptions pop up in discussion. If you have a discussion in the cohort, and someone says, “Yes, but what about...” you should record it here.
 - Typical Assumptions:
 - What makes a regular entity?
 - What makes a weak entity?
 - Which entity owns the weak entity?
 - Do any of the entities fit as a subclass of another entity?
 - What about cardinality? Many-to-many, 1-to-many, 1-to-1?
 - Are the subclasses disjoint or overlapping?
 - What should the primary key be? Is there a surrogate key instead?
-

4 Software Discovery (10 points)

For this project, an entity-relationship diagram must be constructed. Undoubtedly, a dozen systems exist for drawing such diagrams. Whether it be pencil and paper, a simple paint program, to a professional tool, to online freeware: Draw IO, Visio, MySQL workbench and other software packages exist as either open source or as educational software. What will your cohort use? Where is it from? Be sure to include a screen capture. Yes, even if the design tool you chose is pencil and paper.

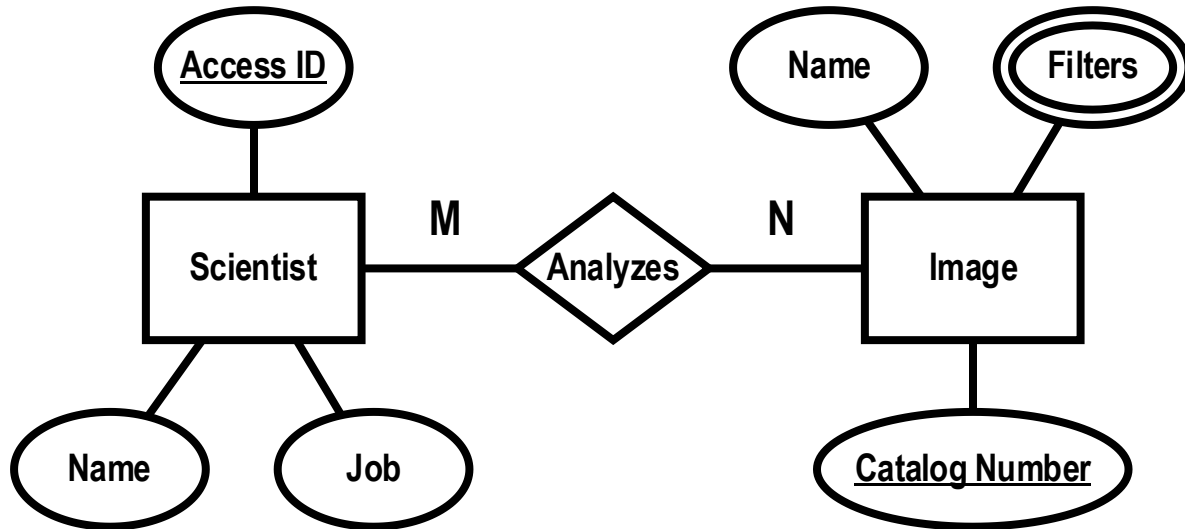
4.1 For this section:

- What software are you using? Cite the software.
- Include a picture of what you are using.
- Give a short description why you are using this software, even if the reason is “Its cheap and free.”

5 Entity-Relationship Diagram (35 points)

Please be aware there are many, many database tools in existence that auto-generate diagrams. You will be expected to use the symbols and notations from our textbook and examples for your project. If you use auto-generating software, you will be expected to re-draw the diagrams with the correct symbols and notation.

A small fragment of such a diagram would be:



For this section:

- You will have an entity-relationship diagram based on your proposal.
- The diagram must match what you have described.
- It should be “complex enough.”
 - How big should the diagram be?
 - If you add all the entities, subclasses, and relations together and wind up between 20 and 30, your project is large enough.
- The Diagram **MUST** have
 - Every strong entity must have a primary key attribute.
 - Every weak entity must have an owning entity and may include a surrogate key.
 - Every entity must have useful information attributes.
 - Every relationship must have cardinality.
 - A weak/owning relationship must exist in this diagram.
 - At least one class lattice must exist in this diagram.

6 Entity-Relationship Dictionary (35 points)

A Dictionary is a list of all the pieces of the diagram, with their definitions. Every entity, relationship, attribute, and key and other properties in the diagram must have an entry. The entries must be detailed to match the diagram. If the entity has an attribute in the diagram, then the entity should have an attribute in the table. The entities and relationships must be sorted into alphabetical order in the Dictionary. There will be two dictionaries. One for entities and any of their subclasses. The other dictionary will be for the relationships.

Example:

ENTITY DICTIONARY:

Entity: Image

Description:

An image produced from the data of the James Webb Space Telescope

Attributes:

- Catalog Number, Int
- Name, VARCHAR(80)
- Filters, VarChar(80), a set of filters applied when the original data was taken.
- Primary Key: Catalog Number

Entity: Scientist

Description:

One of several types of scientists that are studying the results of the James Webb Space Telescope

Attributes:

- AccessID, INT
- Name, VARCHAR(80)
- Job, VARCHAR(80)
- Primary Key: AccessID

RELATION DICTIONARY

Relationship: Analyzes

Description:

A scientist can analyze an image. An image can be analyzed by a scientist.

Entities: Scientist, Image

Cardinality: M:N (Many-to-Many)

Participation: Partial-to-Partial

For this section:

- The Entity Dictionary
 - The dictionary will be in alphabetical order by name of Entity.
 - Classes and Subclasses will be treated as Entities.
 - Every Entity on the diagram must be in the dictionary.
 - Every Attribute of every entity must be in the dictionary.
 - Every Strong Entity must have a defined primary key.
 - Every Weak Entity must be owned by a strong entity.
 - Weak Entities may have a surrogate key.
- The Relation Dictionary
 - The dictionary will be in alphabetical order by name of the Relation.
 - Every Relation in the diagram must be in the dictionary.
 - Any Attributes of a Relation must be in the dictionary.
 - The Entities of the Relation must be specified.
 - The Cardinality of the Relation must be specified.

7 Naming Conventions for Turn In

Turn in each of the four sections with appropriate names. Each file should be a Portable Document File (.PDF)

These files should be turned in as separate files (Which can be done during one session on Blackboard)

Note: The word *Cohort* in the table below should be replaced by the name of your cohort.

Proposal	<i>Cohort</i> .Proposal.pdf
Software Discovery	<i>Cohort</i> .SoftwareDiscovery.pdf
Entity-Relationship Diagram	<i>Cohort</i> .ERDiagram.pdf
Entity-Relationship Dictionary	<i>Cohort</i> .ERDictionary.pdf

7.1 Additional

- The assignment may be turned in multiple times before the deadline.
- This year, cohorts will not be part of the Blackboard group turn-in option.
- Each individual student will have to turn in their own copy.
- The assignment should not be a zip file, the files should be independent for online grading.
- Only the last submission, as a set of all four files, only will be counted.