

Project Tracker

George Mistkawi

URL for Website

<http://web.engr.oregonstate.edu/~mistkawg/CS%20340%20-%20Website/>

Project Outline

My website is designed to track and display projects that I am working on. It will help organize school and personal projects, as well as make them publicly viewable for companies and peers to see what kind of projects I have worked on over my time at oregon state.

Database Outline

The entities in my database are:

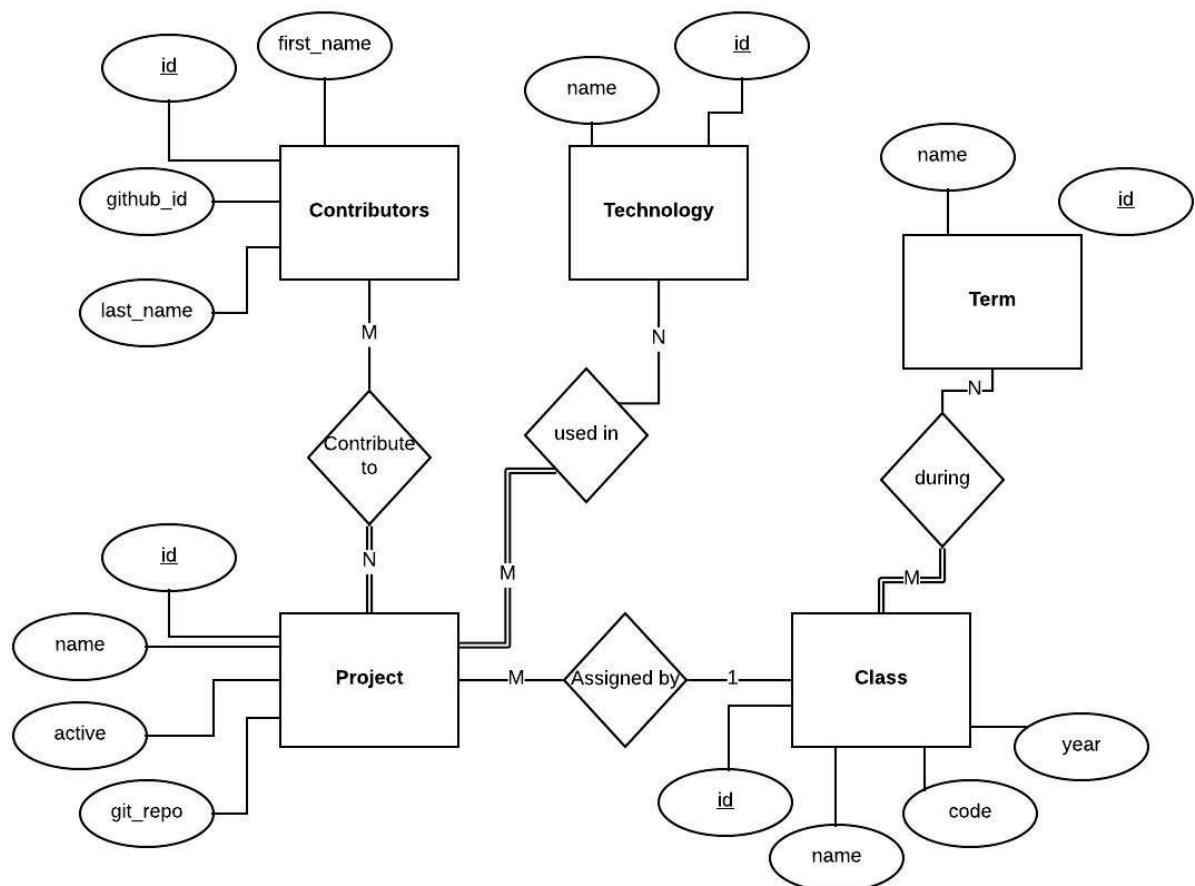
- **Project:** a software project that is being worked on
 - **id:** auto-incrementing primary key
 - **name:** string with no default and cannot be blank
 - **active:** boolean value to show if the project is active, defaults to true
 - **git_repo:** a string the is the web address of the github repository for the project, no default
 - **fk_class_id:** this is an integer foreign key to the id member of the class table, may be blank
- **Class:** the class that the project is being done for
 - **id:** auto-incrementing primary key
 - **name:** name of the class, string with no default and cannot be blank
 - **code:** the letter and number combination for a class, string with no default and cannot be blank
 - **fk_term_id:** this is an integer foreign key to the id member of the term table, no default and cannot be blank
 - **year:** this is a smallint that is the year the class was taken in, no default and cannot be blank'
- **Term:** the terms that are available
 - **id:** auto-incrementing primary key
 - **name:** the name of the term (fall, winter, spring, summer)
- **Contributor:** anyone who worked on a project
 - **id:** auto-incrementing primary key
 - **first_name:** string with no default and cannot be blank
 - **last_name:** string with no default and cannot be blank
 - **github_id:** the contributors github username, string with no default

- **Technology:** technology or language that was used in a project
 - **id:** auto-incrementing primary key
 - **name:** string with no default and cannot be blank

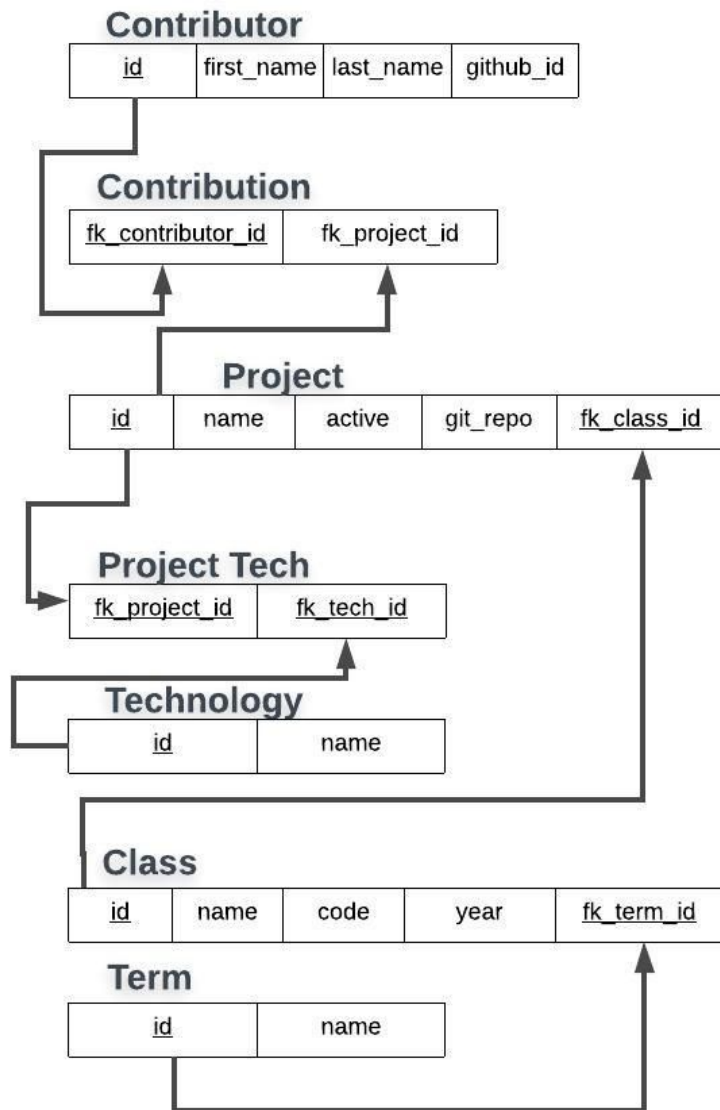
The relationships in my database are:

- **Contributors work on projects:** This is a many-to-many relationship, as many students may have worked or be working on many projects at the same time.
- **Projects are assigned to classes:** This is a many-to-one relationship. A project may only be worked on for a single class, or no class, but a class can have many projects.
- **Technologies are used in projects:** This is a many-to-many relationship, as many technologies may be used in a project, and many projects may use a certain technology.
- **Classes are taken during terms:** This is a many-to-one relationship. A class may only be assigned to one term, but a term may have many classes.

ER Diagram



Schema



Feedback

I decided to redo my project completely and did not turn in the review for 2, and 4-6.

Step 3 Peer Review

ERD 1. Are the attributes for each entity in the ERD same as that described in the database outline? For currency entity, I'd change the "tag" to symbol, or the other way around, to make them match. 2. Is the participation of entities in the relationships same as that described in the outline? Yes 3. Is the cardinality of entities in the relationships same as that described in the outline? Yes 4. Is there something that could be changed/improved in the E R Diagram and/or the overall database design? The exchange rate relationship with a market does not show the relationship with currency. I'm not sure how you could do this in an ER Diagram, but your Schema shows it in a concise way. Schema 5. Are the relationship tables present where required and correctly defined, when compared with the database outline? Yes 6. Are foreign keys present where required and correctly defined, when compared with the database outline? Yes 7. Do the entity attributes match those described in the outline? Yes. The Wallet relationship table caught me by surprise, until I realized that was the relationship between user and currency. The relationship table is accurately named. 8. Is there something that could be changed/improved in the Schema and/or the overall database design? I found it clear to follow, but others may ask for not as many overlapping lines. DDQ file 1. Is the SQL file syntactically correct? This can be easily verified by importing/copying it in phpmyadmin. (Do not forget to take backup of your own database before you do this!) Syntactically, the file is correct. It imported quickly into phpMyAdmin. When you do Insert Into statements, you can let the database do the work when you have auto incrementing integers. You would just need to call out the specific fields that you are inserting into first, and then the values. For example: INSERT INTO user VALUES (1, 'JOE', '1234'), (2, 'BOB', '4321'), (3, 'George', '1111'); Would be: INSERT INTO user (name, password) VALUES ('JOE', '1234'), ('BOB', '4321'), ('George', '1111'); 2. Are the data types appropriate considering the description of the attribute in the database outline? Yes 3. Are the foreign keys correctly defined when compared to the Schema? Yes 4. Are relationship tables present when compared to the ERD/Schema? Yes, it matches them perfectly.

Actions based on Feedback

I added another entity to the erd and outline, this entity is Rates because I believe this was better left as its own entity instead of trying to force it into a relationship definition. The schema already had the relationship and entity defined properly, now all 3 pieces match. I simplified my insert statements in my sql file as explained by my reviewer.