Project Step 4 Draft

Group 25: Pablo Garza and Gino Febles

Link: http://classwork.engr.oregonstate.edu:8111

a) – Project Overview, Database Outline, ERD, Schema Diagram, & Sample Data:

The Good Place Community Library Book Reservation System

Overview

The Good Place library is located in a small town in the USA. The library serves an approximately number of three thousand members and maintains a collection of twenty thousand books across its three branches. Recently, there has been an uptick in customer complaints about books not being available in any of their locations. Thus, the library decided to enlist the services of Bookworms Consultants Inc to design a book database and reservation system to improve their operations and customer satisfaction. With this system, members will be able to reserve books. Each reservation holds the book for only one calendar day, after which it becomes available for other members unless re-reserved.

Database Outline

- **Members:** records details of the library members.
  - memberID: int, auto_increment, unique, not NULL, PK
  - firstName: varchar(255), not NULL
  - lastName: varchar(255), not NULL
  - phoneNumber: varchar(255), not NULL
  - memberEmail: varchar(255), not NULL
  - Relationship: a 1:M relationship between Members and Reservations is implemented with memberID as a FK inside of Reservations.

- **BooksReservations:** records details of the members' reservations.
  - reservationID: int, auto_increment, unique, not NULL, PK
  - memberID: int, auto_increment, unique, not NULL, FK from Members table.
  - bookID: int, auto_increment, unique, not NULL, FK from Books table.
  - libraryID: int, auto_increment, unique, not NULL, FK from Library Locations table.
  - date: timestamp, not NULL.

  - M:1 relationship between BookReservations and Members is implemented with memberID as a FK inside of BookReservations.
  - M:1 relationship between BookReservations and Books is implemented with bookID as a FK inside of BookReservations.
  - M:1 relationship between BookReservations and LibraryLocations is implemented with locationID as a FK inside of BookReservations.

- **Books:** records details of the books in the library.
  - bookID: int, auto_increment, unique, not NULL, PK
  - bookName: varchar(255), not NULL
  - authorId: int, auto_increment, unique, not NULL, FK
  - Relationship: a M:1 relationship between Books and Authors is implemented with authorID as a FK inside of Books.
  - Relationship: M:M between Books and LibraryLocations will be implemented through an intersection table BookLocations.

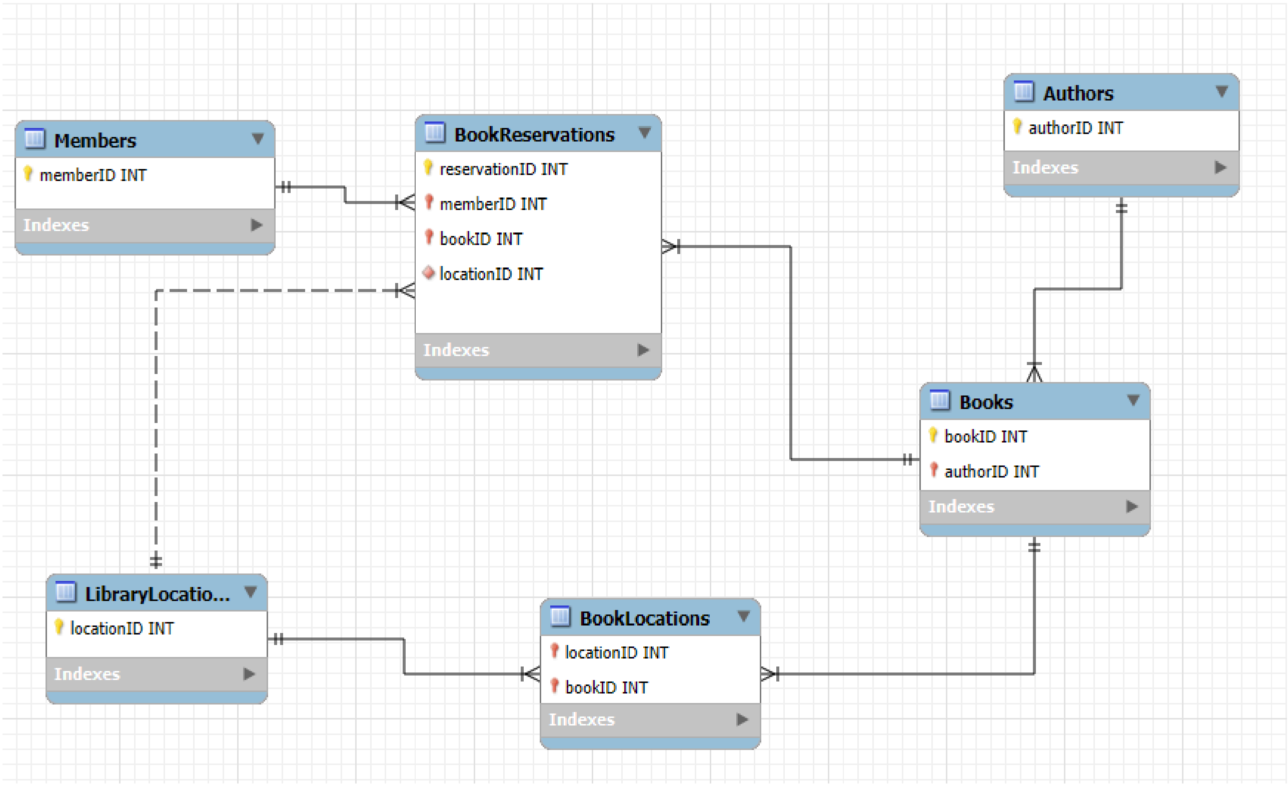
- **Authors:** records details of the book authors.
  - authorID: int, auto_increment, unique, not NULL, PK
  - firstName: varchar(255), not NULL
  - lastName: varchar(255), not NULL
  - Relationship: a 1:M relationship between Authors and Books is implemented with authorID as a FK inside of Books.

- **LibraryLocations:** records details of the library's locations.
  - locationID: int, auto_increment, unique, not NULL, PK
  - locationName: varchar(255), not NULL
  - locationAddress: varchar(255), not NULL
  - locationEmail: varchar(255), not NULL
  - locationPhone: varchar(255), not NULL
  - Relationship: a M:M relationship between LibraryLocations and Books will be implemented through an intersection table BookLocations.

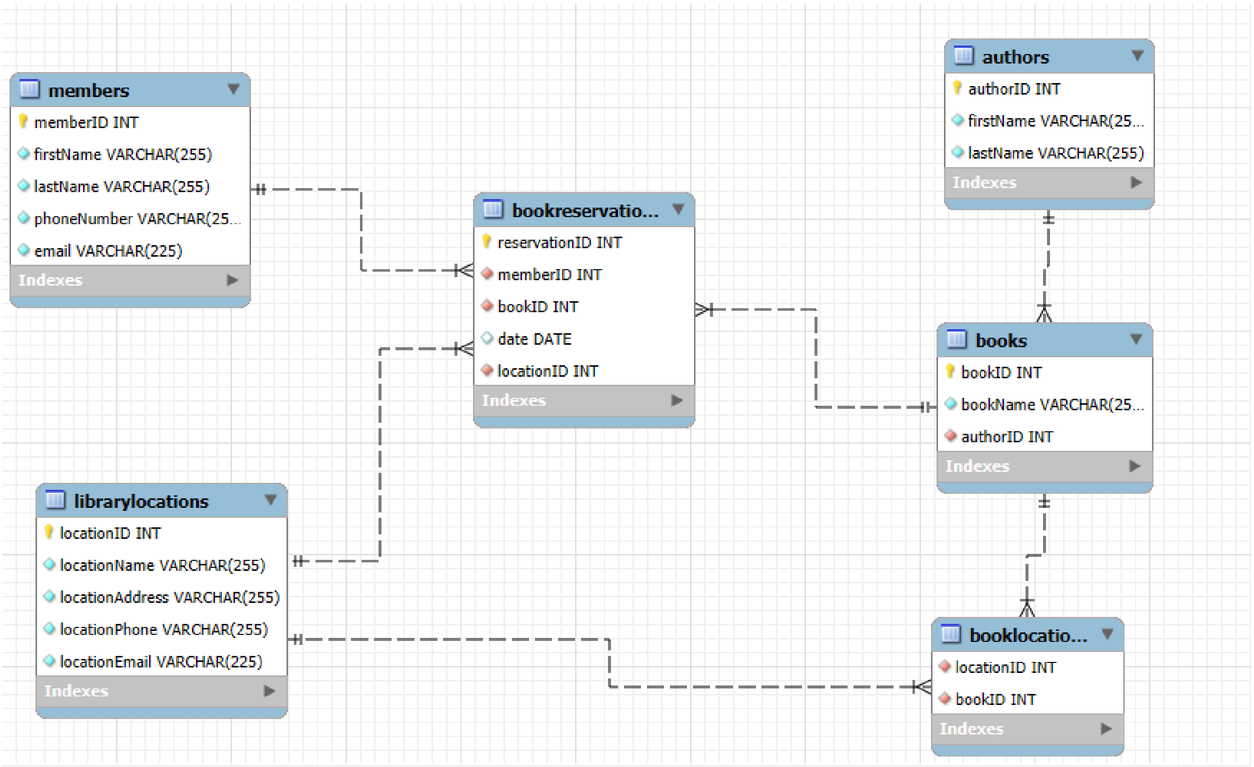
- **BookLocations:** intersection table between Books and LibraryLocations.
  - locationID: int, auto_increment, unique, not NULL, FK
  - bookID: int, auto_increment, unique, not NULL, FK
  - Relationship: a M:1 relationship between BookLocations and Books is implemented with bookID as a FK inside BookLocations.

- Relationship; a M:1 relationship between BookLocations and LibraryLocations is implement with locationID as a FK inside BookLocations.

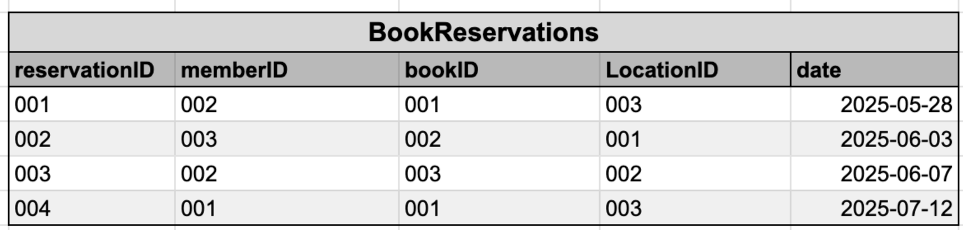
c) Entity-Relationship Diagram:



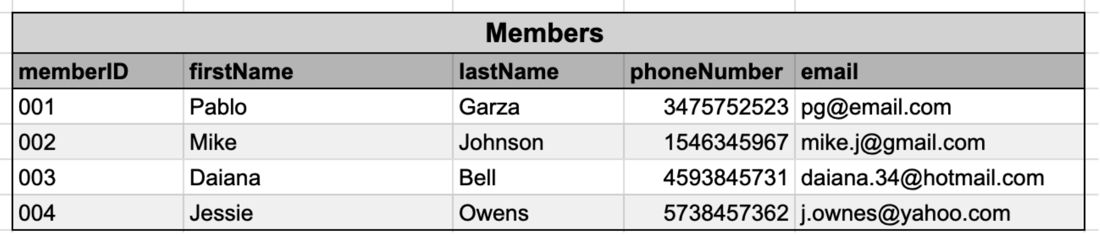
d) Schema:



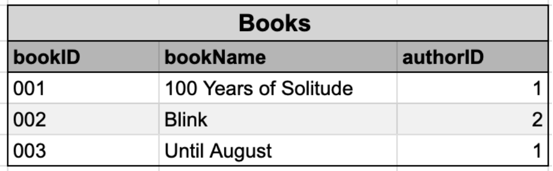
e) Example Data



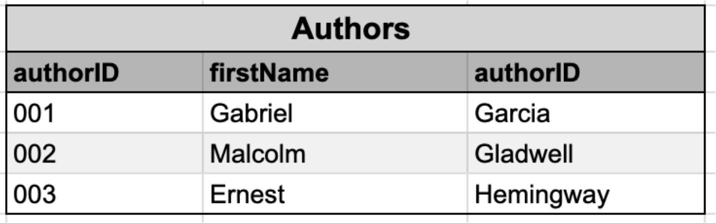
BookReservations				
reservationID	memberID	bookID	LocationID	date
001	002	001	003	2025-05-28
002	003	002	001	2025-06-03
003	002	003	002	2025-06-07
004	001	001	003	2025-07-12



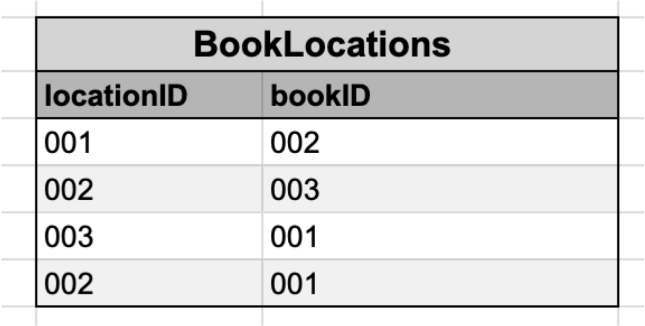
Members				
memberID	firstName	lastName	phoneNumber	email
001	Pablo	Garza	3475752523	pg@email.com
002	Mike	Johnson	1546345967	mike.j@gmail.com
003	Daiana	Bell	4593845731	daiana.34@hotmail.com
004	Jessie	Owens	5738457362	j.ownes@yahoo.com



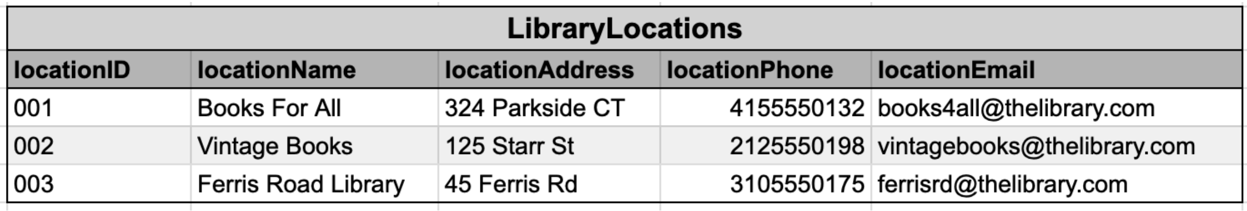
Books		
bookID	bookName	authorID
001	100 Years of Solitude	1
002	Blink	2
003	Until August	1



Authors		
authorID	firstName	authorID
001	Gabriel	Garcia
002	Malcolm	Gladwell
003	Ernest	Hemingway



BookLocations	
locationID	bookID
001	002
002	003
003	001
002	001



LibraryLocations				
locationID	locationName	locationAddress	locationPhone	locationEmail
001	Books For All	324 Parkside CT	4155550132	books4all@thelibrary.com
002	Vintage Books	125 Starr St	2125550198	vintagebooks@thelibrary.com
003	Ferris Road Library	45 Ferris Rd	3105550175	ferrisrd@thelibrary.com

Citations

- All work is original
- Example data builtwith google sheets
- Er diagram and Schema built with MySQL Workbench

b) – Fixes based on Feedback from Previous Steps:

Upgrades to the Draft version Step 4 Draft

- Added one record to book locations table to illustrate M:N relationship

Feedback - Step 3

Peer review 1, Michael Lu

Does the UI utilize a SELECT for every table in the schema? **In other words, data from each table in the schema should be displayed on the UI. Note: it is generally not acceptable for just a single query to join all tables and displays them.**

Yes, each page clearly displays data from its corresponding table

***Does the UI implement an INSERT form for at least one table in the schema?* In other words, there should be UI input fields that correspond to at least one table.**

Yes, the "Add" buttons across the pages suggest that insert forms are implemented for at least books, authors, members, and locations.

***Does the UI have at least one DELETE for any one entity?* In other words, is there a form/link/button that will allow the deletion of a row in at least one table?**

Yes, each table view has a "Delete" button, indicating that rows from those entities can be deleted.

***Does the UI have at least one DELETE that will remove things from a M:M relationship?* In other words, if an order is deleted from the Orders table, it should also delete the corresponding rows from the OrderDetails table, BUT it should not delete any Products or Customers.**

Yes, the "Confirmed Reservations" section displays data from a many-to-many relationship between members and books. The "Delete" option here most likely removes records from the intersection table without deleting the books or members themselves.

***Is there at least one UPDATE form in the UI for any one entity?* In other words, in the case of Products, can productName, listPrice, qtyOnHand, e.g. be updated for a single ProductID record?**

Yes, the "Edit" button available across the different tables suggests that the user can update existing records.

***Is there at least one UPDATE form in the UI to modify an M:M relationship?* In other words, does the UPDATE allow the user to select a different foreign key value to update the intersection table with?**

Yes, the reservations table, which reflects a many-to-many relationship, includes an "Edit" button, likely allowing updates to reservation details such as dates or associated books.

Do you have any other suggestions for the team to help with their HTML UI? For example, using AS aliases to replace obscure column names such as fname with First Name.

I think your column names are simple and clear. One thing I would suggest is adding a brief description or welcome message to the homepage can immediately inform users about the website’s purpose and how to navigate it. For example, a short paragraph like *“Welcome to The Good Place Library! Here you can explore our book collections, manage your reservations, and find information about our library locations...etc* would make the homepage feel more engaging and informative.

Peer review 2, Drake Vickers

be displayed on the UI. Note: it is generally not acceptable for just a single query to join all tables and displays them.

Yes, the UI does have a SELECT for all tables within the schema.

Does the UI implement an INSERT form for at least one table in the schema? In other words, there should be UI input fields that correspond to at least one table.

Yes , the UI is built for a CREATE form that corresponds to at least one table.
Does the UI have at least one DELETE for any one entity? In other words, is there a form/link/button that will allow the deletion of a row in at least one table?
No the UI does not have DELETE buttons for all entities as it appears the members seem permanent.
Does the UI have at least one DELETE that will remove things from a M:M relationship? In other words, if an order is deleted from the Orders table, it should also delete the corresponding rows from the OrderDetails table, BUT it should not delete any Products or Customers.
Yes the UI has at least one DELETE that corresponds to a M:M relationship.
Is there at least one UPDATE form in the UI for any one entity? In other words, in the case of Products, can productName, listPrice, qtyOnHand, e.g. be updated for a single ProductID record?
No there is not at least one UPDATE forms in the UI for every entity, members and book locations have no edit button.
Is there at least one UPDATE form in the UI to modify an M:M relationship? In other words, does the UPDATE allow the user to select a different foreign key value to update the intersection table with?
yes there is at least one UPDATE form in the UI to modify an M:M relationship.
Do you have any other suggestions for the team to help with their HTML UI? For example, using AS aliases to replace obscure column names such as fname with First Name.
When selecting authorID, memberID, or locationID, show drop-downs with names instead of raw IDs (e.g., full author names or location names).
For reservations, show book titles and member names by joining relevant tables for better readability.
work all original

Peer review 3, Tamithe Beyer

- ***Does the UI utilize a SELECT for every table in the schema?* In other words, data from each table in the schema should be displayed on the UI. Note: it is generally not acceptable for just a single query to join all tables and displays them.** Yes, the UI utilized a SELECT for every table in the schema. Data from each table in the schema is displayed on each page of the UI.
- ***Does the UI implement an INSERT form for at least one table in the schema?* In other words, there should be UI input fields that correspond to at least one table.** The UI implements an INSERT form for multiple tables in the schema.
- ***Does the UI have at least one DELETE for any one entity?* In other words, is there a form/link/button that will allow the deletion of a row in at least one**

table? Yes, the UI has a DELETE button for the Books, Authors, Library Locations, and Book Reservations tables.

- ***Does the UI have at least one DELETE that will remove things from a M:M relationship?*** **In other words, if an order is deleted from the Orders table, it should also delete the corresponding rows from the OrderDetails table, BUT it should not delete any Products or Customers.** Yes, the UI has at least one DELETE that will remove things from a M:M relationship.
- ***Is there at least one UPDATE form in the UI for any one entity?*** **In other words, in the case of Products, can productName, listPrice, qtyOnHand, e.g. be updated for a single ProductID record?** Yes, the UI has at least one UPDATE form in the UI for any one entity.
- ***Is there at least one UPDATE form in the UI to modify an M:M relationship?*** **In other words, does the UPDATE allow the user to select a different foreign key value to update the intersection table with?** Yes, there is at least one UPDATE form in the UI to modify an M:M relationship as well.
- ***Do you have any other suggestions for the team to help with their HTML UI? For example, using AS aliases to replace obscure column names such as fname with First Name.*** I really do not see anything that needs to be modified with your HTML UI. Your column names are clear and understandable. You have easy to navigate links at the top of each page to get to all the tables in your database. It appears to be a well executed, thought out, solid HTML UI.

- **As a reviewer, clearly describe to what extent your feedback to the team was original (e.g. "all my work") or non-original (e.g. used AI tools per Code Citation Tips).** All feedback is original.

Peer review 4, Alexander Stephen

Does the UI utilize a SELECT for every table in the schema? In other words, data from each table in the schema should be displayed on the UI.

- *Yes all tables from the schema are displayed in the website UI.*

Does the UI implement an INSERT form for at least one table in the schema? In other words, there should be UI input fields that correspond to at least one table.

- *There is an added function to four of the tables but, I don't think they are in the right spot.*

Does the UI have at least one DELETE for any one entity? In other words, is there a form/link/button that will allow the deletion of a row in at least one table?

- *Yes, four of the tables have a delete form.*

Does the UI have at least one DELETE that will remove things from a M:M relationship? In other words, if an order is deleted from the Orders table, it should also delete the corresponding rows from the OrderDetails table,

- *Yes, one of the tables has a delete form a M:*N relationship

Is there at least one UPDATE form in the UI for any one entity? In other words, in the case of Products, can productName, listPrice, qtyOnHand,

- *Yes, there is one table with an update form.*

Is there at least one UPDATE form in the UI to modify an M:M relationship? In other words, does the UPDATE allow the user to select a different foreign key value to update the intersection table with?

- *Yes, there is one table with an update form to update a M:*N relationship.

Do you have any other suggestions for the team to help with their HTML UI? For example, using AS aliases to replace obscure column names such as fname with First Name.

- I would add some information to your homepage but other then that everything looks great.

As a reviewer, clearly describe to what extent your feedback to the team was original (e.g. "all my work") or non-original (e.g. used AI tools per Code Citation Tips).

- All work is original.

Actions based on feedback - Step 3

- Based on Alexander's, and Michael's suggestions, we've decided to add a brief description to the home page.

Upgrades to the draft versions - Step 3

- The HTML file of the home page has a brief description now.
- We added a Foreign Key to the book reservations table. This way, it is clear what copy of a book is being reserved.

Feedback - Step 2

Peer review 1, TA

Test data and report looks good! All entities look to be properly normalized. Great job!

Peer review 2, Ji Mun

- Does the schema present a physical model that follows the database outline and the ER logical diagram exactly?

Yes schema presents a physical model that follows the database outline and the ER logical diagram exactly.

- Is there consistency in a) naming between overview, outline, ER and schema entity/attributes b) entities plural, attributes singular c) use of capitalization for naming?
    - The naming of the attributes, and the tables match in the outline, ER and Schema.
    - Entities are plural, and attributes are singular. Attributes use camel case.
    - Convention: Entities are capitalized, attributes are not.
- Is the schema easy to read (e.g. diagram is clear and readable with relationship lines not crossed)?

Yes, Schema and ERD are both easy to read, easy to follow the lines and they do not cross.

- Are intersection tables properly formed (e.g. two FKs and facilitate a M:N relationship)?

M:N Tables: LibraryLocations and Books

There is an intersection table to accommodate the LibraryLocations and Books: BookLocations. It has to foreign keys from both tables.

- Does the sample data suggest any non-normalized issues, e.g. partial dependencies or transitive dependencies?

All tables are normalized to 3NF.

- Is the SQL file syntactically correct? This can be easily verified by using PhPMyAdmin and your CS 340 database (do not forget to take backup of your own database before you do this!)

The SQL file is syntactically correct.

- In the SQL, are the data types appropriate considering the description of the attribute in the database outline?

Yes all the data types for the attributes are appropriate considering the description of the database outline.

- In the SQL, are the primary and foreign keys correctly defined when compared to the Schema? Are appropriate CASCADE operations declared?

Yes in the SQL all the primary and foreign keys are correctly defined when compared to the Schema. CASCADE operations are correctly declared, syntactically correct.

- In the SQL, are relationship tables present when compared to the ERD/Schema?

The relationship between tables are defined mainly by foreign keys, and this is represented in SQL. I do wonder why the firstName and lastName have a constraint UNIQUE, because in the real world, two people with same names may exist.

- In the SQL, is all example data shown in the PDF INSERTED?

Yes, although they are not direct screenshots queried from the database, maybe they used a tool I am unfamiliar with.

- Is the SQL well-structured and commented (e.g. hand authored) or not (e.g. exported from MySQL)?

SQL is well structured, and well commented. They are hand authored and well organized.

Peer review 3, Patricia (Trish) Stackpole

- Does the schema present a physical model that follows the database outline and the ER logical diagram exactly?
    - Overall it's well done, consistent, and easy to follow.
- Is there consistency in a) naming between overview, outline, ER and schema entity/attributes b) entities plural, attributes singular c) use of capitalization for naming?
    - A) yes naming is consistent between outline, ER, and Schema
    - B) Yes, Entities (Books, LibraryLocations) are plural and upper case while attributes (bookID, firstName) were camelCase and singular.
    - C) Yes consistent capitalization for naming was used, Entity names were uppercase while attribute names were camelCase.
    - The naming conventions made sense, were easy to follow, and were consistent between the description, ERD, Schema, and SQL
- Is the schema easy to read (e.g. diagram is clear and readable with relationship lines not crossed)?
    - Your ERD and Schema are neat and easy to read. I appreciated that you were able to fit both the schema and ERD on one page, it made it so much easier to review.
- Are intersection tables properly formed (e.g. two FKs and facilitate a M:N relationship)?
    - Intersection Tables are formatted correctly in the schema, ERD, and SQL. However I would recommend updating the database outline to describe if the PKs in the BookReservations and BookLocations tables are compound primary keys or made with unique auto incremented numbers.
- Does the sample data suggest any non-normalized issues, e.g. partial dependencies or transitive dependencies?
    - I did not find any issues such as partial dependencies or transitive dependencies

- Is the SQL file syntactically correct? This can be easily verified by using PhPMyAdmin and your CS 340 database (do not forget to take backup of your own database before you do this!)
    - There were no SQL issues I could find. I was able to add to my DB without issues and run sample queries. It was clear this could be very useful!
- In the SQL, are the data types appropriate considering the description of the attribute in the database outline?
    - Yes the datatypes are appropriate and match the PDF
- In the SQL and match the outline In the SQL are the primary and foreign keys correctly defined when compared to the Schema? Are appropriate CASCADE operations declared?
    - PKs and FKs match the ERD and Schema. See above questions, there were a few places in the database outline that needed to be updated with this information
    - Cascade operations were declared appropriately
- In the SQL, are relationship tables present when compared to the ERD/Schema?
    - Yes, ERD, Schema, and SQL Tables all match and the relationship tables are present.
- In the SQL, is all example data shown in the PDF INSERTED?
    - Yes, all data in the PDF was inserted into tables in the SQL
    - There are some spelling inconsistencies between your sample data in the PDF and in the SQL (for example “Ernest Hemingway” vs “Ernst Hemingway”).
- Is the SQL well-structured and commented (e.g. hand authored) or not (e.g. exported from MySQL)?
    - SQL is well structured and organized. More comments could be helpful in the table creation section, for example, what type of table, the purpose of the table etc.

Some other feedback I had that didn’t quite fit into the above questions: Look over your database overview again, there are a few small grammatical errors. I would recommend also updating a few things in the outline and overview for clarity. The “Date” attribute description in the BooksReservations intersection table, it’s not clear if that’s the date the book was reserved, the day it is due, or the day it was checked out. In the overview “able to reserve books for one day” could be clarified, there is some slight ambiguity if that’s the length the book can be checked out for, or if that’s the length the reservation will be held for.

Actions based on feedback - Step 2

- Made revision to overview.

Upgrades to the draft versions - Step 2

Added clarification to how the book reservation works in the overview.

Feedback - Step 1

Peer review 1, TA

Make sure you add the intersection table to your ER diagram between Books and LibraryLocations. Other than that, looks good!

Derek Lilienthal , Apr 19 at 7:19pm

Regraded - omitting intersection tables is not point deductible

Derek Lilienthal , Apr 21 at 7:26am

Peer Review 2, William Reilly

There are some minor grammar/spacing issues in your overview but it's otherwise concise and addresses a problem as well as a solution. I think your ERD has a good start but it is rather incomplete as it doesn't have the properties of each entity, so that will need to be corrected.

Suggestions:

I think it would be a good idea to attach more specific numbers e.g. there has been a 35% uptick in customer complaints, and/or The Good Place Library serves 10000 customers annually. Maybe a statistic on how much material the Good Place has in their holdings. Keep in mind the numbers should be appropriate to the solution you're proposing.

Your phone numbers should be stored as strings - ints can/are used for calculation and you wouldn't want an operation of that sort happening to your customer's phone numbers. Unwinding that could be very complicated.

Double check the cardinalities on your ERD diagram as well. One author can write many books, and one book can have many authors.

You have a good start here, it just needs some more filling out.

Peer Review 3, Kristin Pinilla

Hi Pablo and Gino! I think you two did a nice job on your project draft.

Your overview states that the customer complaints about books not being available in any of The Good Place library's three branches is a problem to be solved by a website with DB backend. The hope is that this will improve The Good Place library's operations and customer satisfaction. I think that, in order to better illustrate the scope and scale of your proposed DB solution, it could be helpful to include facts such as how many books each branch can hold and how many members utilize the library system.

Your outline describes four entities, Members, Books, Authors, and LibraryLocations, so satisfies the rubric requirement of having at least four entities. I think all of your entities do represent a single idea to be stored as a list, and that they sufficiently list attribute data types and constraints. It seems like they also follow proper naming conventions, using plurals for entity names. It's possible that you two have already done this, but I didn't see it in your draft, so thought I'd mention that it might be good to consider the sizes that you need for each attribute that uses VARCHAR.

Regarding your attributes, it looks like there is consistency in naming conventions – your attributes are singular and follow camelCase. You may have a reason for doing so, but for the phoneNumber attribute in your Members entity and the locationPhone attribute in your LibraryLocations entity, you might want to consider storing as VARCHAR rather than INT for the purpose of keeping the format of a phone number that you wouldn't get with integers.

The 1:M relationship that you've identified between Members and Reservations, makes sense to me, as does your M:M relationship between LibraryLocations and Books. It might be worth considering if Authors and Books should have a M:M relationship, since can an author be considered an author if they haven't written any books? And isn't it possible that a book can have more than one authors?

I think your project is going to end up being really useful and great!

The above review is all my work.

Actions based on feedback - Step 1

- We changed the data type of phoneNumber and locationPhone.
- We included 2 more facts in our overview to provide further context given that both peer reviews suggested that more numbers could be useful.
- We decided to keep the relationship between Authors and Books 1:M so we can focus on implementing one single M:M relationship.

- As for William’s note “our ERD has a good start but it is rather incomplete as it doesn't have the properties of each entity, so that will need to be corrected. ”. Although, this was not required in the initial draft. We updated our ERD model.
- We are clarifying the size of our VARCHAR types.
- We added the intersection table to our diagram.

Upgrades to the draft versions - Step 1

No upgrades to the draft version. All changes made were based on peer review feedback.