CPSC 6620: Database Management Systems

Final Project Part 1

David Welsh

Lucid Chart ERD

Before we go *farther*, I want to apologize for the mess of the ERD you may be seeing. As of writing and turning this in, this was my first time working with Lucid Chart and it feels like that came through. Though I plan to tinker around with this some more on Monday, as I'm trying to get this turned in on Sunday evening to not accidentally forget it. With that out of the way, let's get to the nature of this ERD.

The Entity Relationship Diagram models a yoga school system that tracks students, leaders, classes, attendance, and skill levels. At its foundation is the Student Entity, which stores details such as each student's ID, name, date of birth, and date they joined. As all of the leaders are also students, but not all students are leaders, the leader schema is modeled after the student schema and is connected in an optional way. This prevents redundancy in the data, and allows us to assign leader specific criteria, such as the payment Boolean, to the student leaders.

The classes are represented by the Class Entity, which contains the class's unique identifier, associated level, and weekly chronological information, as well as the leader assigned to it. As each leader can teach many classes, but each class only has one leader, they exist in a one-to-many relationship. Now, to accommodate for the fact that each class can take place over multiple times/dates, a class session entity records individual class meetings, including the session date. This enables the system to track attendance and leadership roles at each session level.

Now, because each leader can assist in a session and one leader serves as the head, the session leader bridge entity was created, as it resolves a many to many relationships between leaders and class sessions. Similarly, the session attendance entity tracks which students attend which sessions, forming a many to many relationship between students and class sessions. This setup allows for a detailed attendance history and flexibility, such as sessions with no attendees, or new students who've yet to attend any classes.

Lastly, the level entity captures the structure of yoga progression, including level name and description, while the level requirement entity lists the specific requirements for each level. To track student progress, the student level bridge entity connects students and levels, recording

the date the level was achieved. This design supports historical tracking of advancement from beginner to master and ensures that each student begins at the beginner level upon joining.

In conclusion, the design is fully normalized, modular, and capable of capturing every important aspect of class participation, leadership, and student development within the yoga school.