Entities

Users

This relation keeps track of the users of our site. Here, you can find expected user data: Username/email (primary key), Login password, First/last name, and whether or not they're an admin. We made this a separate relation because a user can write multiple reviews for different courses/professors (one-many). Putting user data as an attribute of either the course/professor review relations would create duplicate data.

Course Reviews

This relation keeps track of the course reviews from users. Its attributes are user-rated workload (hours per week), difficulty (1-10), amount learned (1-10), and custom text. A custom ReviewID key is created, the concatenation of the course number and the username. Two foreign keys, username (from Users) and course number (from Courses) highlight the weak entity nature of this table (if either the course or user is deleted, their corresponding reviews should be as well). This relation is independent from both Courses and Users because there can be multiple reviews per user AND per course, so having Course Reviews as a separate table eliminates redundancy.

Courses

This relation keeps track of UIUC courses. Its attributes are course information: Course Number, Course Name, Department, Credit Hours, and an average GPA. It also has custom ranking based on the attributes of the Course Reviews table and the "GPA" column of this table.

Professor Reviews

This relation keeps track of the professor reviews from users. Its attributes are user-rated Lecture Quality (hours per week), Approachability (1-10), Test Difficulty (1-10), and custom text. A custom ReviewID key is created, the concatenation of the professor's NetID and the username. Two foreign keys, username (from Users) and NetID number (from Professors) highlight the weak entity nature of this table (if either the course or user is deleted, their corresponding reviews should be as well). This relation is independent from both Professors and Users because there can be multiple reviews per user AND per course, so having Professor Reviews as a separate table eliminates redundancy.

Professors

This relation keeps track of UIUC courses. Its attributes are professor information: NetID, First Name, Last Name and Times Rated Excellent. It also has custom ranking based on the attributes of the Professors Reviews table and the “Times Rated Excellent” column of this table.

Normalization:
Our database is in 3NF, because we have removed attributes with multiple values, made sure all attributes depend on a primary key for their table, and eliminated transitive dependencies.

Listed below are the Functional Dependencies:
ReviewID, Username, CourseID → CourseNumber, Workload, AmountLearned, Difficulty, ReviewText (Course Review Table)

Course Number → GPA, Course Name, Department, Credit Hours, Ranking
Username → Password, FirstName, LastName, IsAdmin (Courses Table)

CourseNumber, ProfessorID -> GPAData (GPA Table)

NetID → FirstName, LastName, Times Rated Excellent, Rating (Users Table)

ReviewID, Username, ProfessorID → LectureDifficulty, Approachability, TestDifficulty, ReviewText (Professor Reviews Table)

Relational Schema
Table-Users(Username:VARCHAR(255) [PK], Password:VARCHAR(255), FirstName:VARCHAR(255), LastName:VARCHAR(255),IsAdmin:BIT)

Table-CourseReviews(ReviewID:VARCHAR(255) [PK], Username:VARCHAR(255) [FK to Users.Username], CourseNumber:VARCHAR(255) [FK to Courses.CourseNumber], Workload:TINYINT, AmountLearned:TINYINT, Difficulty:TINYINT, ReviewText:VARCHAR(255))

Table-Courses(CourseNumber:VARCHAR(255) [PK], GPA:DECIMAL [FK to GPA.CourseNumber], Department:VARCHAR(255), CreditHours:TINYINT, Ranking:DECIMAL, CourseName:VARCHAR(255))

Table-GPA(CourseNumber:VARCHAR(255) [FK to Courses.CourseNumber], ProfessorID:VARCHAR(255) [FK to Professors.NetID], GPAData: DECIMAL)

Table-ProfessorReviews(ReviewID:VARCHAR(255) [PK], Username:VARCHAR(255) [FK to Users.Username], ProfessorId:VARCHAR(255) [FK to Professors.NetID], LectureQuality:TINYINT, Approachability:TINYINT, TestDifficulty:TINYINT, ReviewText:VARCHAR(255))

Table-Professors(NetID:VARCHAR(255) [PK], FirstName:VARCHAR(255), LastName:VARCHAR(255), CreditHours:TINYINT, TimesRatedExcellent:TINYINT, Rating:DECIMAL)