**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. The relationship between Users and Reviews is one-many, because a user can write multiple reviews for different courses/professors, but each review can only be written by one user.

Reviews

This relation stores the reviews from users. Its attributes are user-rated Workload (hours per week), Amount Learned (1-10), Difficulty (1-10), Lecture Quality (1-10), Organization (1-10), and custom text. We represent this relation as a weak entity because a review is dependent on 3 other data points existing: the user who writes the review, the course that the review is about, and the professor that the review is about. The relationships between Professors/Courses and Reviews is also one-many using similar logic as above (many reviews can exist for any given professor/course, but each review can only be about one professor/course).

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 rating based on the attributes of the Professors Reviews table and the “Times Rated Excellent” column of this table.

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 rating based on the attributes of the Course Reviews table and the “GPA” column of this table. The relationship between this table and the GPA table is one-one, since each course has a specific GPA (and vice versa).

GPA

This relation keeps track of the GPA of courses at UIUC. Its attributes relate to GPA data: a SectionID, Course Name, Professor and numerical GPA.

**Functional Dependencies**

Username, ProfessorID, CourseNumber→ Workload, AmountLearned, Difficulty, Lecture Quality Management, ReviewText (Reviews)

Course Number → Course Name, Department, Credit Hours, Rating (Courses)

Username → Password, FirstName, LastName, IsAdmin (Users)

SectionID → CourseName, Professor, GPA (GPA)

ProfessorID → FirstName, LastName, Times Rated Excellent, Rating (Professors Table)

**Relational Schema**

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

Table-Reviews(Username:VARCHAR(255) [FK to Users.Username], CourseNumber:VARCHAR(255) [FK to Courses.CourseNumber], ProfessorID:VARCHAR(255) [FK to Professors.ProfessorID], Workload:TINYINT, AmountLearned:TINYINT, Difficulty:TINYINT, LectureQuality: TINYINT, Management:TINYINT, ReviewText:VARCHAR(255), [PK: Username, CourseNumber, ProfessorID])

Table-Courses(CourseNumber:VARCHAR(255) [PK], CourseName:VARCHAR(255), Department:VARCHAR(255), CreditHours:TINYINT, Rating:DECIMAL)

Table-GPA(SectionID:VARCHAR(255) [PK]: CourseName:VARCHAR(255), Professor:VARCHAR(255), GPAData: DECIMAL)

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