Group Members: Brenden Guillen, Jonathan Thornton

Group Name: The Raters Not Graders

Problem Statement

Our project idea is an application similar to "Rate My Professor", but is geared more towards both the classes offered by a university and the professors that teach that class. The purpose is to help students find classes and professors at an institution that they would like to take by looking through courses they could take for their major as well as course difficulty ratings and ratings for professors that teach the course. A secondary goal of the project is to gather a list of classes here at S&T and related information that is searchable and sortable by the university, department, professor that teaches a specific course, class difficulty, comments, ratings, and possibly more. Entities include universities, classes, professors, students/users, textbooks, and departments. The database will also keep track of the cost for each course, average cost per credit for a particular university, textbooks for classes, and the price, rating and any possible pdf links for a textbook. This will be implemented with a command line program.

Conceptual Database Design

Assumptions

1. A course must be taught by a professor.

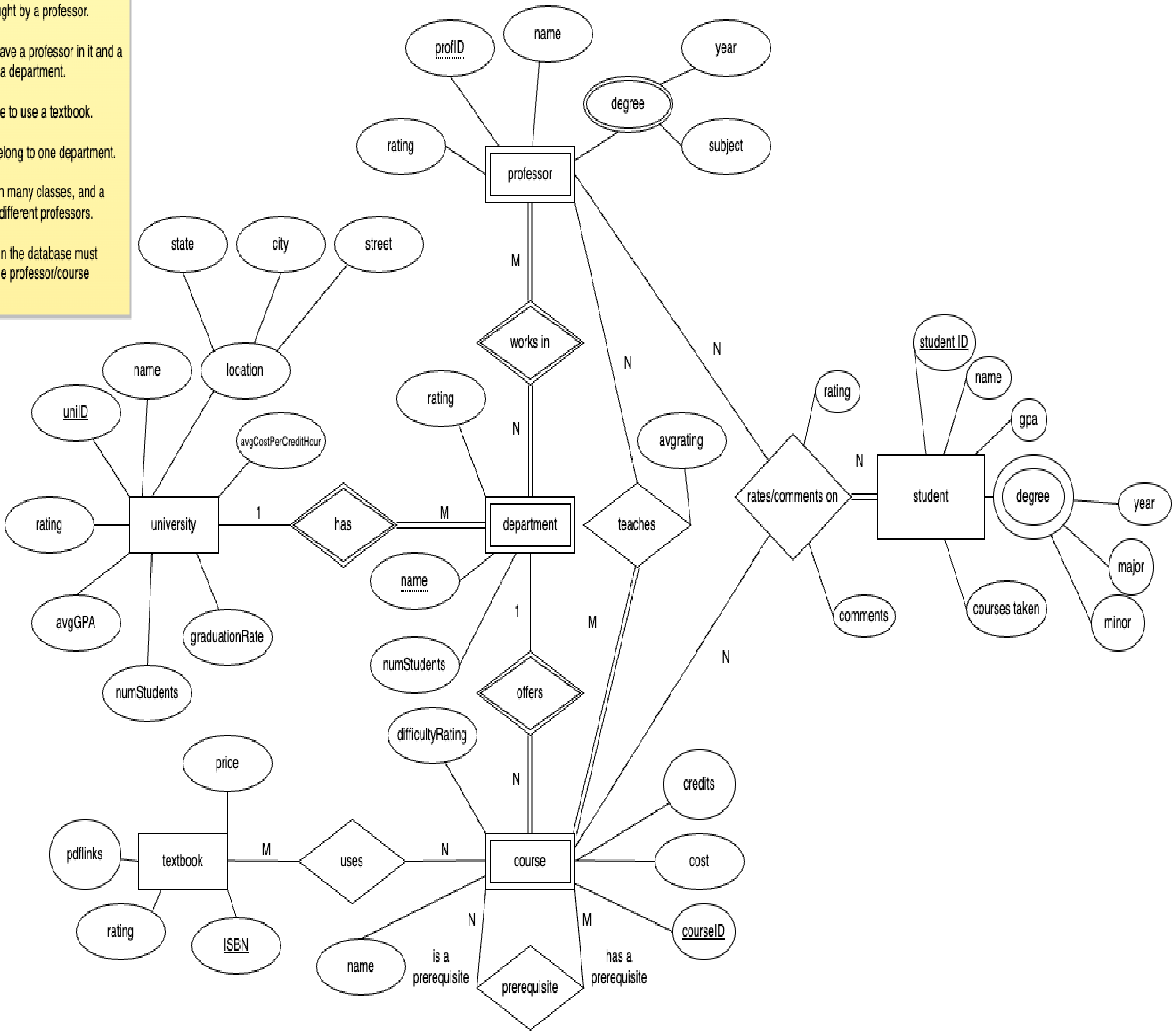
2. A department must have a professor in it and a professor must work in a department.

3. A course doesn't have to use a textbook.

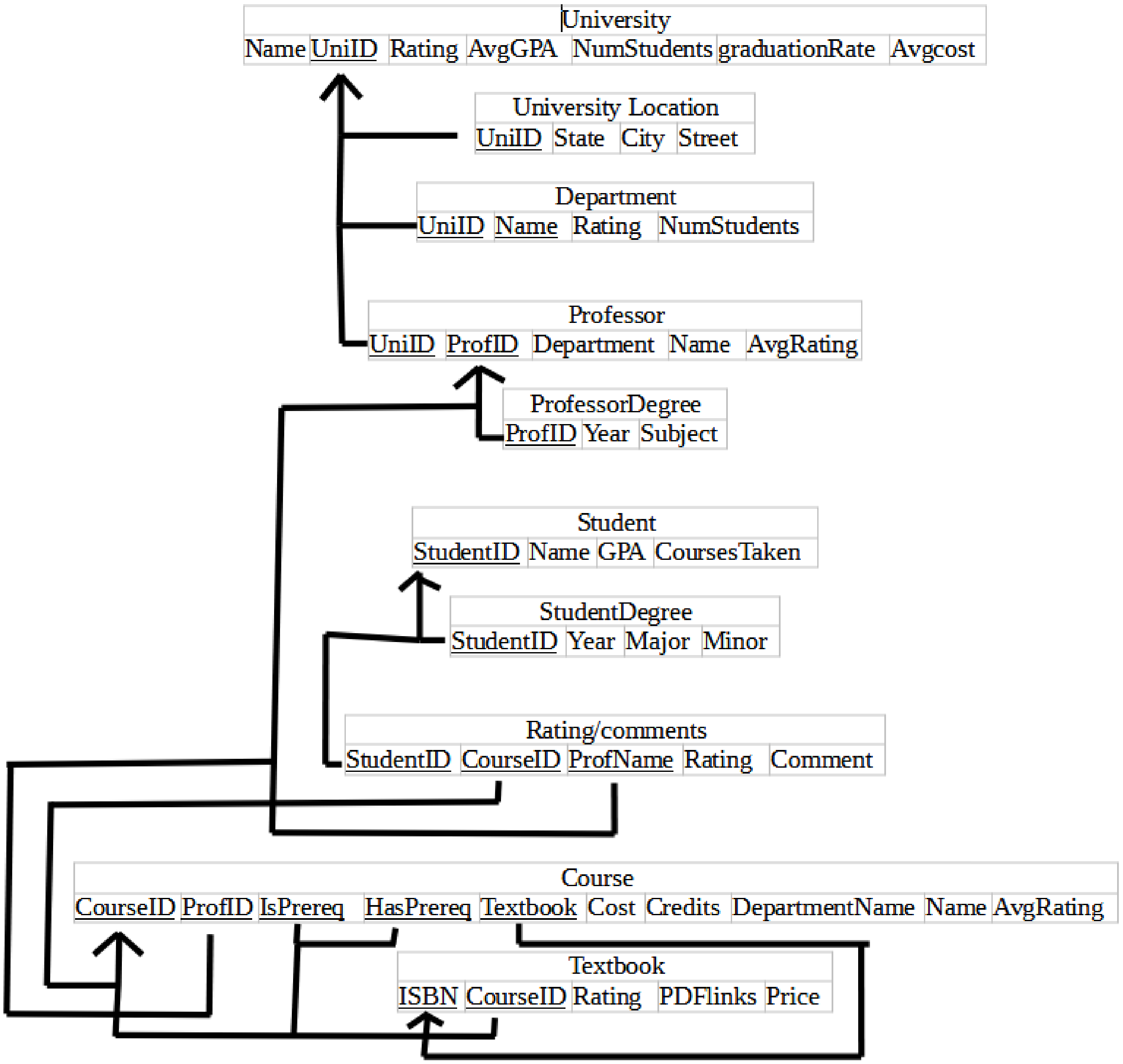
4. A course may only belong to one department.

5. A professor can teach many classes, and a class can be taught by different professors.

6. All students entered in the database must comment on at least one professor/course



Logical Database Design



University						
Name	UniID	Rating	AvgGPA	NumStudents	graduationRate	Avgcost

University Location			
UniID	State	City	Street

Department			
UniID	Name	Rating	NumStudents

Professor				
UniID	ProfID	Department	Name	AvgRating

ProfessorDegree		
ProfID	Year	Subject

Student			
StudentID	Name	GPA	CoursesTaken

StudentDegree			
StudentID	Year	Major	Minor

Rating/comments				
StudentID	CourseID	ProfName	Rating	Comment

Course									
CourseID	ProfID	IsPrereq	HasPrereq	Textbook	Cost	Credits	DepartmentName	Name	AvgRating

Textbook				
ISBN	CourseID	Rating	PDFlinks	Price

Summary Table of Data Types

Table	Attribute	Data Type	Constraints	Notes

University	Name	String	
University	UniID	Integer	Primary Key
University	Rating	Float	
University	AvgGPA	Float	
University	NumStudents	Integer	
University	GraduationRate	Float	
University	AvgCost	Float	
UniversityLocation	UniID	Integer	Foreign Key
UniversityLocation	State	String	NotNull
UniversityLocation	City	String	NotNull
UniversityLocation	Street	String	NotNull
Department	UniID	Integer	Foreign Key
Department	Name	String	NotNull
Department	Rating	Float	
Department	NumStudents	Integer	
Professor	UniID	Integer	Foreign Key

Professor	Department	String		
Professor	ProfID	Integer	Primary Key	
Professor	Name	String	NotNull	
Professor	AvgRating	float		
ProfessorDegree	ProfID	Integer	Foreign Key	
ProfessorDegree	Year	Integer		
ProfessorDegree	Subject	String		
Student	StudentID	Integer	Primary Key	
Student	Name	String	NotNull	
Student	GPA	Float		
Student	CoursesTaken	String		
StudentDegree	StudentID	Integer	Foreign Key	
StudentDegree	Year	Integer		
StudentDegree	Major	String		
StudentDegree	Minor	String		
Rating/commen	StudentID	Integer	Foreign	Needed For this one

ts			Key	
Rating/comments	CourseID	Integer	Foreign Key	Optional, for comments/ratings for the course specifically
Rating/comments	ProfID	Integer	Foreign Key	Optional, for comments/ratings for the professor specifically
Rating/comments	Rating	Float		
Rating/comments	Comment	String		

Textbook	ISBN	Integer	Primary Key	
Textbook	Rating	Float		
Textbook	PDFLinks	String		links to free pdf’s for those that don’t care for paperback
Textbook	Price	Float		
Textbook	CourseID	Integer	Foreign Key	

Course	CourseID	Integer	Primary Key	
Course	Name	String		
Course	Cost	Float		
Course	Credits	Float		Allowed to have half credit classes
Course	DepartmentName	String		
Course	ProfID	String	Foreign Key	
Course	IsPrereq	String	Foreign Key	

Course	HasPrereq	String	Foreign Key
Course	TextbookISBN	Integer	Foreign Key
Course	AvgRating	Float	

Application Program Design

```
add_uni()
    Take in provided uni
    if(Uni is already in the database)
        Don’t add uni to database
    else
        Add uni to database

add_dept()
    Take in provided dept and uni
    if(Dept is already in the uni)
        Don’t add dept to uni
    else
        Add dept to uni

add_prof()
    Take in provided prof, dept, and uni
    if(Prof is already in the dept of the uni)
        Don’t add prof to dept of uni
    else
        Add prof to dept of uni

add_course()
    Take in provided course, dept, and uni
    if(course is already being offered by dept of a uni)
        Don’t add course to dept of uni
    else
        Add course to dept of uni

add_textbook()
    Takes in textbook
    if(Textbook is already in database)
        Don’t add textbook to database
    else
        Add textbook to database

add_student()
    Takes in student
    if(Student is already in database)
        Don’t add student in database
    else
        Add student in database

delete_uni()
    Take in provided uni
    if(Uni is in the database)
        Delete uni, delete all related weak entities

delete_dept()
    Take in provided dept and uni
    if(Dept is in the uni)
        Delete dept, delete dept in other tables

delete_prof()
    Take in provided prof, dept, and uni
    if(Prof is in the dept of the uni)
        Delete prof, delete prof in other tables

delete_course()
    Take in provided course, dept, and uni
    if(Course is being offered by dept of a uni)
```

```
        Delete course, delete course in other tables

delete_textbook()
    Takes in textbook
    if(Textbook is in database)
        Delete textbook, delete in other tables

delete_student()
    Takes in student
    if(Student is in database)
        Delete student, delete in other tables

add_course_teacher()
    Takes in a course and a professor
    if(Prof has already taught the class)
        Don’t add prof as a teacher for the class
    else
        Add prof as a teacher for the class

add_course_textbook()
    Takes in a course and textbook
    if(Textbook is being used by course)
        Don’t add textbook as a course textbook
    else
        Add textbook as a course textbook

add_prereq()
    Takes in two courses
    if(Course1 is already a prereq of Course2)
        Don’t add Course1 as a prereq for Course2
    else
        Add Course1 as a prereq for Course2

rate_uni() //How good the overall uni is
    Takes in a number and uni
```

```
        Changes uni rating

    rate_dept() //How good the overall dept is
        Takes in a number and dept
        Changes dept rating

    rate_course_difficulty() //How difficult the course is
        Takes in a number and course
        Changes course difficulty rating

    rate_textbook() //How good a textbook is
        Takes in a number and textbook
        Changes textbook rating

    rate_course_with_prof() //How well a prof taught a course
        Takes in a prof, course, and rating
        Changes professor rating

    list_unis_by_rating()
        Lists out all unis in order of rating

    list_depts_by_rating()
        Takes in dept name
        Lists out all unis in database in order of dept
rating

    list_profs_by_rating()
        Takes in uni
        Lists out all profs in uni in order of prof rating

    list_courses_by_rating()
        Takes in course
        Lists out all unis in order of course rating

    get_avg_prof_rating()
        Takes in a uni
        Averages the ratings of all professors in the uni

    get_worst_prof()
        Takes in uni
        Finds the lowest prof rating in the uni

    get_best_prof()
        Takes in uni
        Finds the highest prof rating in the uni

    get_easiest_course()
        Takes in uni and dept
        Finds lowest class difficulty rating in the dept

    get_hardest_course()
        Takes in uni and dept
        Finds highest class difficulty rating in the dept

    add_to_plan()
        Takes in course and adds it to course planner
```