

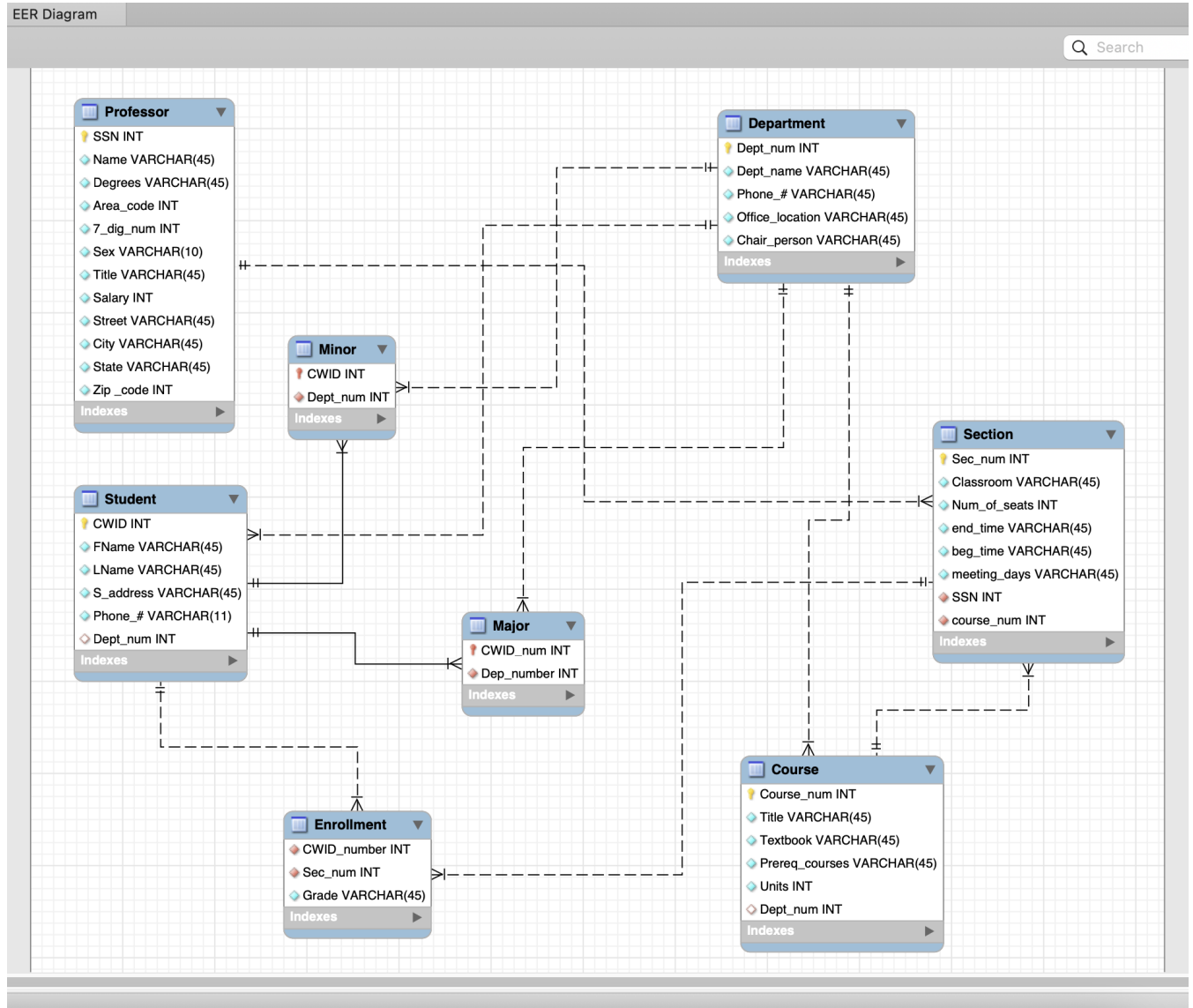
# CPSC 332 Database Project

Aidan Sunahara 1086  
Parker Martel 5934  
Sebastian Rodriguez 0070

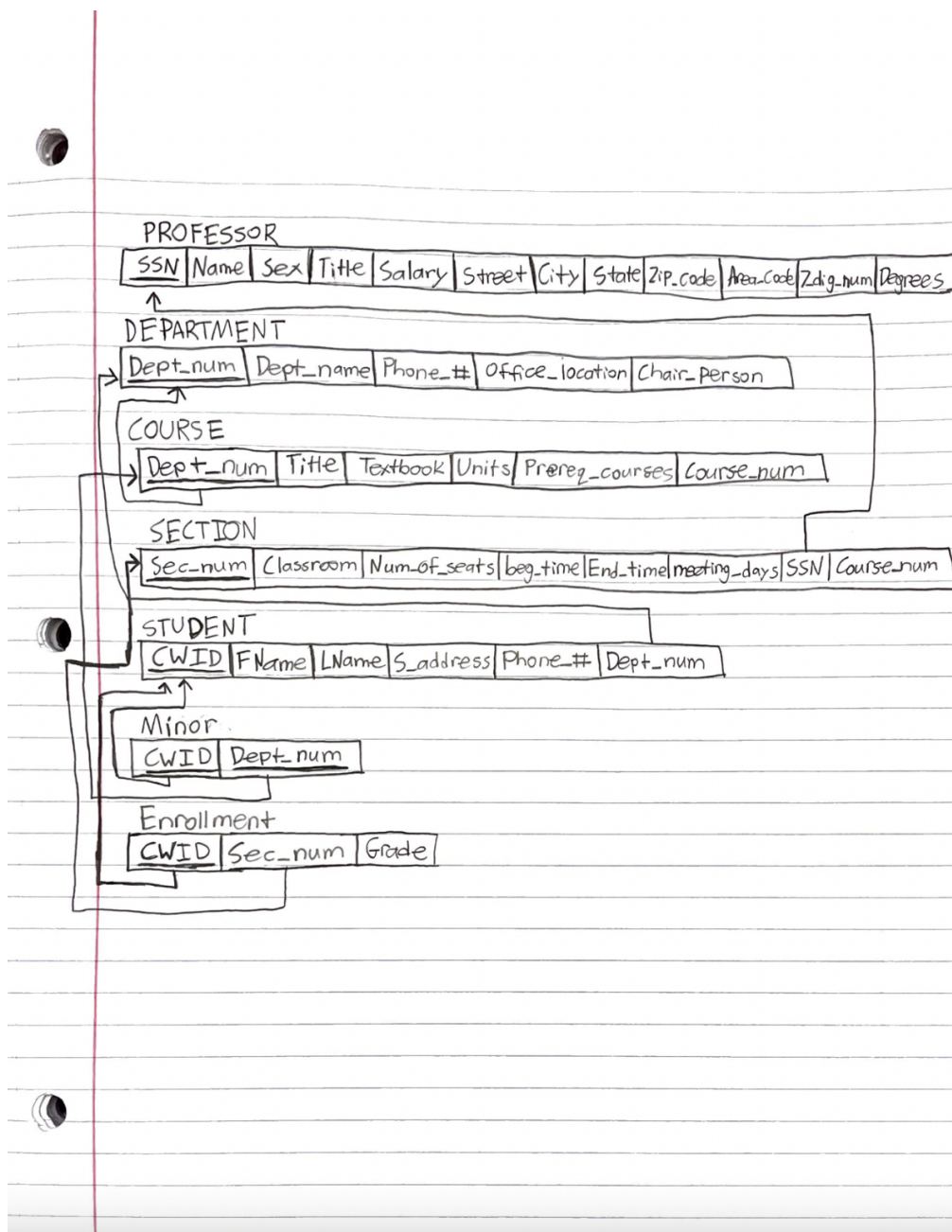
Account - cs332t22

Models:

## ER model



# Relational model



Queries:

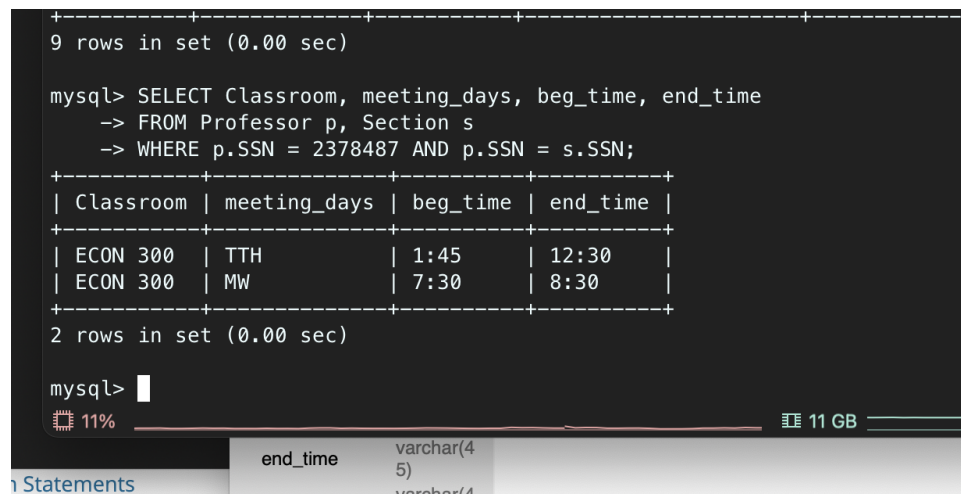
## Professor Query 1

```
SELECT Classroom, meeting_days, beg_time, end_time
FROM Professor p, Section s
WHERE p.SSN = 2378487 AND p.SSN = s.SSN
```

```
9 rows in set (0.00 sec)

mysql> SELECT Classroom, meeting_days, beg_time, end_time
-> FROM Professor p, Section s
-> WHERE p.SSN = 2378487 AND p.SSN = s.SSN;
+-----+-----+-----+-----+
| Classroom | meeting_days | beg_time | end_time |
+-----+-----+-----+-----+
| ECON 300 | TTH          | 1:45     | 12:30    |
| ECON 300 | MW           | 7:30     | 8:30     |
+-----+-----+-----+-----+
2 rows in set (0.00 sec)

mysql> 
```



Classroom	meeting_days	beg_time	end_time
ECON 300	TTH	1:45	12:30
ECON 300	MW	7:30	8:30

1 Statements

end\_time varchar(45)

11% 11 GB

## Professor Query 2

```
SELECT Grade, COUNT(Grade)

FROM Course c, Section s, Enrollment e
WHERE c.Course_num = s.course_num
AND s.Sec_num = e.Sec_num
GROUP BY Grade
```

```
mysql> SELECT Grade, COUNT(Grade)
->
-> FROM Course c, Section s, Enrollment e
-> WHERE c.Course_num = s.course_num
-> AND s.Sec_num = e.Sec_num
-> GROUP BY Grade
-> ;

+-----+-----+
| Grade | COUNT(Grade) |
+-----+-----+
| A     | 4             |
| A+    | 1             |
| A-    | 2             |
| B+    | 3             |
| B-    | 2             |
| C     | 2             |
| C+    | 1             |
| F     | 1             |
+-----+-----+
8 rows in set (0.00 sec)

mysql> 
```

## Student Query 1

```
SELECT S.Sec_num, S.Classroom, S.meeting_days, S.beg_time, Count(E.CWID_number)
FROM Section S , Enrollment E
WHERE E.Sec_num = S.Sec_num AND S.course_num=5893
GROUP BY S.Sec_num
```

```
mysql> select S.Sec_num, S.Classroom, S.meeting_days, S.beg_time, Count(E.CWID_number)
-> from Section S , Enrollment E
-> where E.Sec_num = S.Sec_num
-> and S.course_num=5893
-> GROUP BY S.Sec_num;
```

Sec_num	Classroom	meeting_days	beg_time	Count(E.CWID_number)
69	CS101	MW	2	3

```
1 row in set (0.00 sec)
```

```
mysql> █
```

## Student Query 2

```
SELECT C.Title, E.Grade
FROM Enrollment E, Section S, Course C
WHERE E.Sec_num= S.Sec_num
AND C.Course_num = S.course_num
AND CWID_number = 99362748
```

```
mysql> SELECT C.Title, E.Grade
-> FROM Enrollment E, Section S, Course C
-> WHERE E.Sec_num= S.Sec_num
-> AND C.Course_num = S.course_num
-> AND CWID_number = 99362748
-> ;
```

Title	Grade
CPSC 351	B-
CPSC 332	A-
CPSC 351	A
CPSC 362	A+
CPSC 353	C

```
5 rows in set (0.00 sec)
```

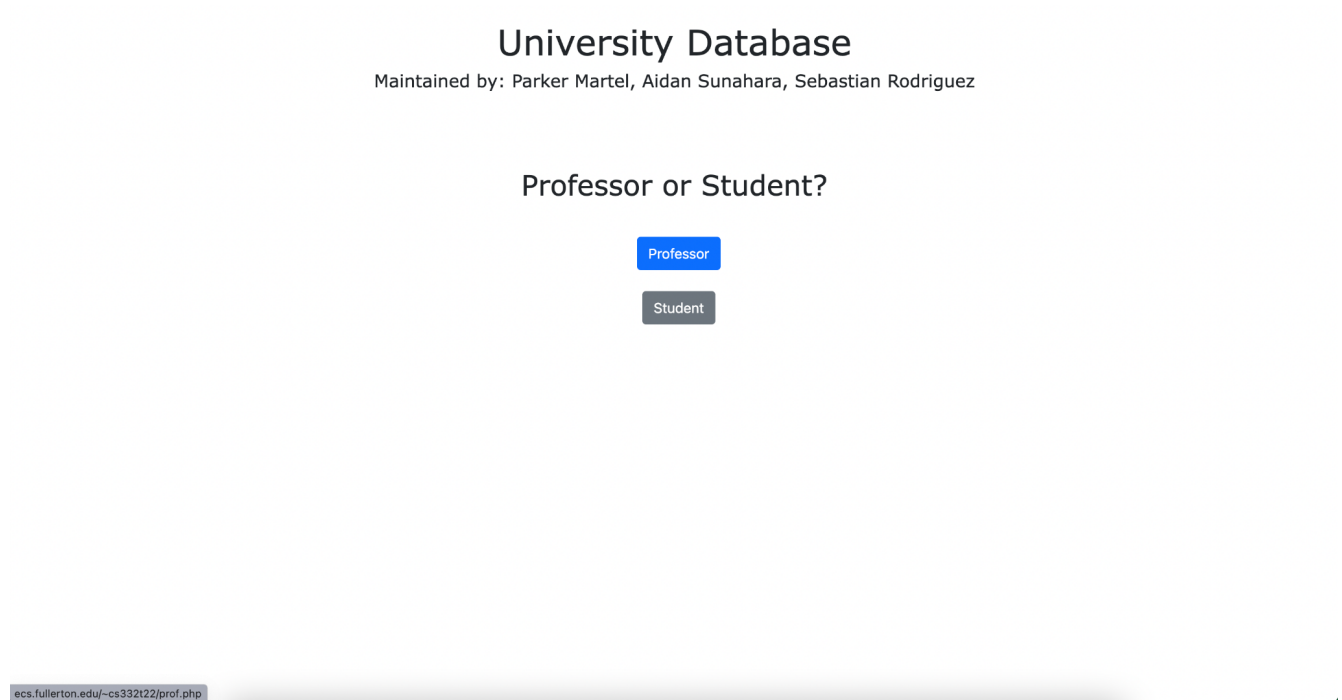
```
mysql> 
```

8%

1

## Interfaces - Pages:

### Homepage:



The screenshot shows the homepage of the 'University Database'. At the top, the title 'University Database' is centered, followed by the text 'Maintained by: Parker Martel, Aidan Sunahara, Sebastian Rodriguez'. Below this is a section titled 'Professor or Student?' with two buttons: 'Professor' (blue) and 'Student' (grey). At the bottom left, there is a small text link: 'ecs.fullerton.edu/~cs332t22/prof.php'.

University Database

Maintained by: Parker Martel, Aidan Sunahara, Sebastian Rodriguez

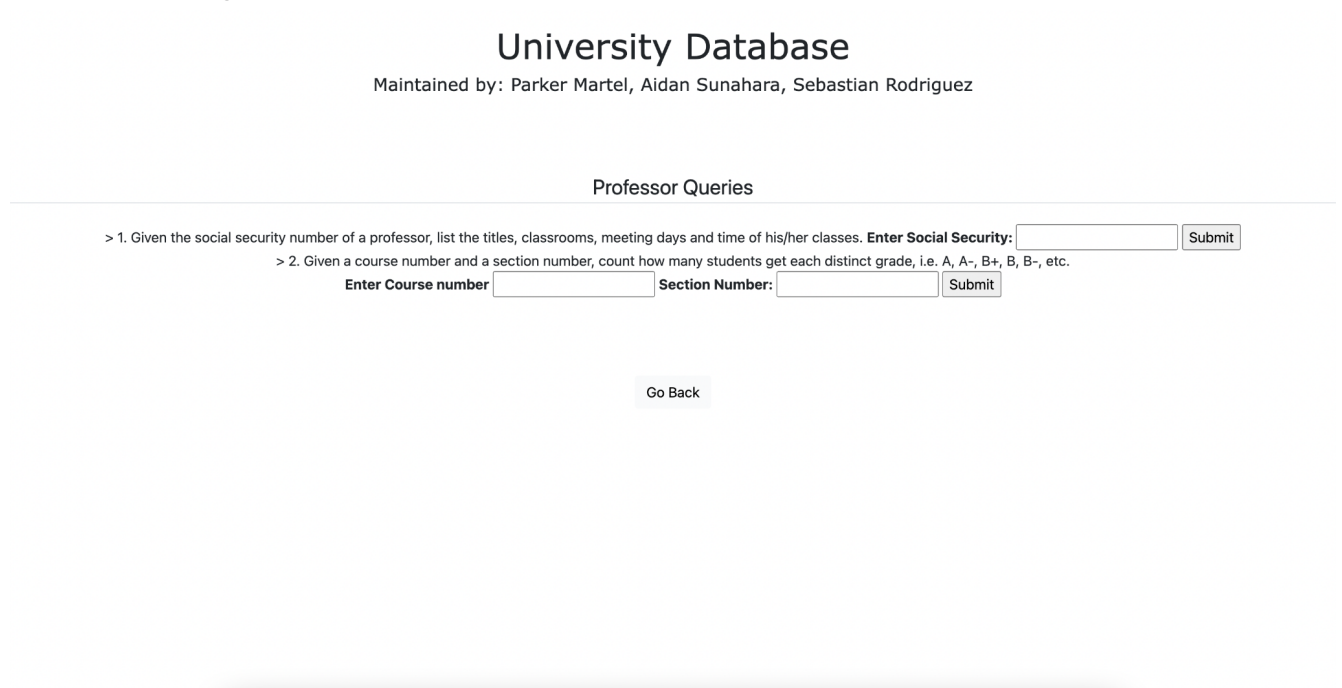
Professor or Student?

Professor

Student

[ecs.fullerton.edu/~cs332t22/prof.php](http://ecs.fullerton.edu/~cs332t22/prof.php)

### Professor page:



The screenshot shows the 'Professor Queries' page of the 'University Database'. The title 'University Database' and the maintainers' names are at the top. Below is a section titled 'Professor Queries'. It contains two numbered instructions: 1. 'Given the social security number of a professor, list the titles, classrooms, meeting days and time of his/her classes. Enter Social Security:' followed by a text input field and a 'Submit' button. 2. 'Given a course number and a section number, count how many students get each distinct grade, i.e. A, A-, B+, B, B-, etc.' followed by 'Enter Course number' and a text input field, 'Section Number:' and a text input field, and a 'Submit' button. At the bottom center is a 'Go Back' button.

University Database

Maintained by: Parker Martel, Aidan Sunahara, Sebastian Rodriguez

Professor Queries

> 1. Given the social security number of a professor, list the titles, classrooms, meeting days and time of his/her classes. Enter Social Security:

> 2. Given a course number and a section number, count how many students get each distinct grade, i.e. A, A-, B+, B, B-, etc.

Enter Course number  Section Number:



## Student page:

### University Database

Maintained by: Parker Martel, Aidan Sunahara, Sebastian Rodriguez

#### Student Queries

- > 1. Given the campus wide ID of a student, list all courses the student took and the grades. **Enter the student CWID:**
- > 2. Given a course number list the sections of the course, including the classrooms, the meeting days and time, and the number of students enrolled in each section. **Enter the course number:**

Interfaces - Sample Runs:

## Student Query 1:

Input:

list all courses the student took and the grades. **Enter the student C**  
ding the classrooms, the meeting days and time, and the number of s

Output:



Section number: 69  
Classroom: CS101  
meeting days: MW  
beginning time: 2  
amount of students: 3

## Student Query 2:

Input:

eries

ie grades. **Enter the student CWID:**

s and time, and the number of students enrolled in each section. **Enter the course number:**

Output:



Title: Econ 101

Grade: C+

Title: Econ 201

Grade: B+

## Professor Query 1:

Input:

his/her classes. **Enter Social Security:** 

get each distinct grade, i.e. A, A-, B+, B, B-, etc.

:

Output:



Classroom: ECON 300

meeting days: TTH

beg time: 1:45

end time: 12:30

Classroom: ECON 300

meeting days: MW

beg time: 7:30

end time: 8:30

## Professor Query 2:

### Input:

my number of a professor, not the class, classroom, meeting days and time of the professor. Enter course number:

> 2. Given a course number and a section number, count how many students get each distinct grade, i.e. A, A-, B+, B, B-, etc.

Enter Course number

Section Number:

### Output:



Grade: B+

Amount of that grade: 1

Grade: C+

Amount of that grade: 1