COMM 3220 - 03

ConnectedU

A Peer-to-Peer Tutoring Service

DEC 2019 // PREPARED BY JULIA FRISCH, LENA LIN, JOSEPH LU, SAM MARZYIE, MATTHEW MERRICK

Table of Contents

Project Description

The purpose of our team's database system will be for a hypothetical entrepreneurship venture in the form of an application, called ConnectedU ("Connect-E-D-U"), which matches tutors and tutees, schedules their sessions, and manages payments all in one place. The primary goal of the database is to aggregate data from thousands of individuals, sort through them, and find matches in terms of availability, classes, and pricing information for tutoring services so that negotiation is removed from the process. As opposed to having a human management team, the app would be entirely responsible for pairing individuals. As a result, our information management system must be highly sophisticated and effective. It is worth noting that the initial design of this database application is targeted toward university students only, therefore, participants would have to be an active university student in the United States (both tutors and tutees) in order to be eligible to utilize our application.

The application would not only remove additional fees through third party vendors but also created more flexibility of tutoring time slot selections and efficient communication between both parties. The idea of the design is to remove consumer pain points in the current market and create a more seamless service that will be successful similar to the business model implemented by Uber. In order to execute this exciting technology platform, database modeling is necessary. Therefore, our team must first identify several market problems to guide our research for a resourceful database platform. This later inspired us to create ConnectedU and thus a solution to the following current market problems:

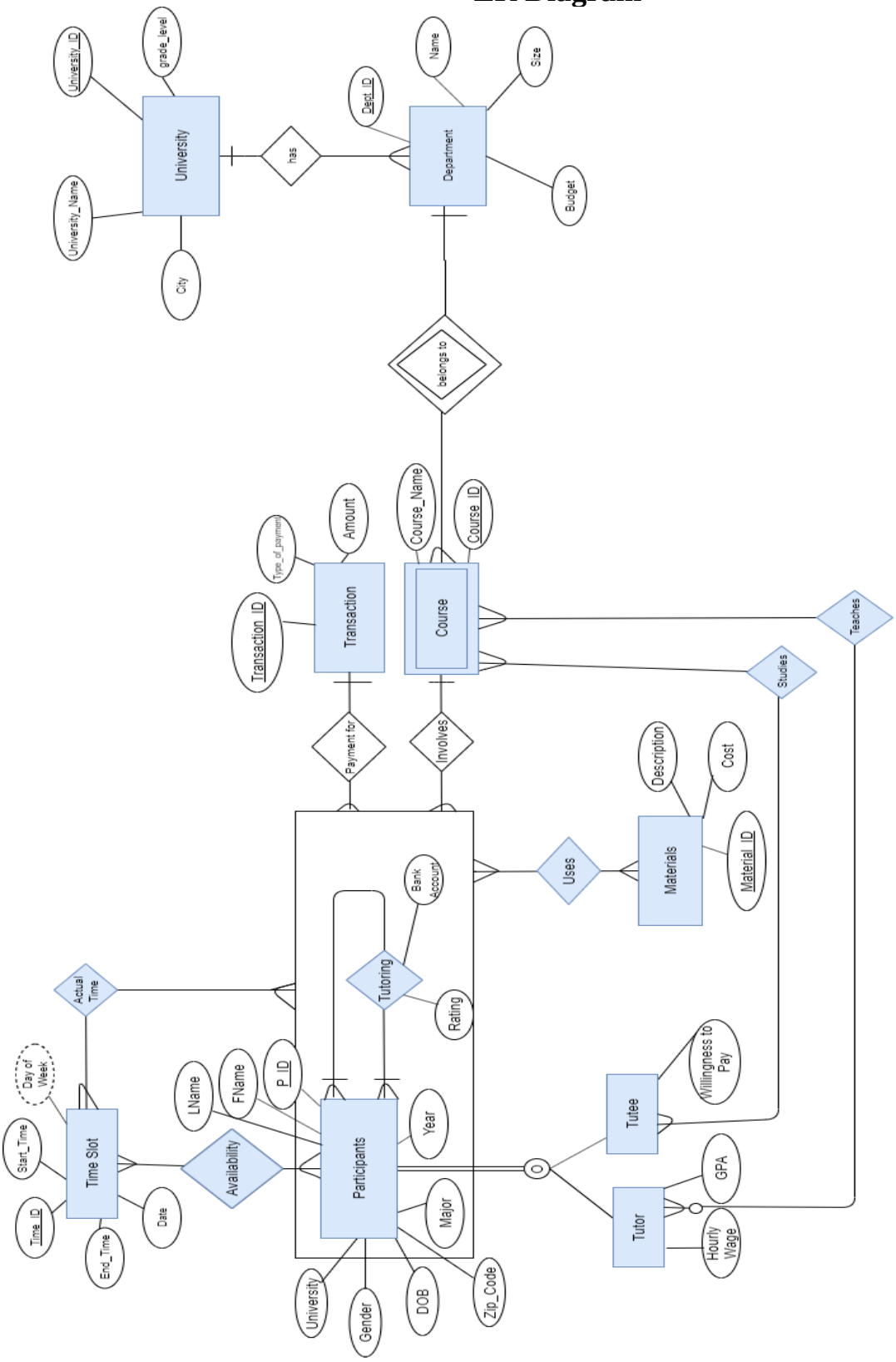
1. Tutors don't have a platform for them to post and provide services for extra cash inflows.

2. Students cannot find tutors easily and in a timely manner (specially to review concepts for upcoming exams and quizzes).
3. The payment system is limited or non-existent between tutors and students.
4. Third-party coordinators are not always up-to-date with connecting tutors and students and fulfilling requests.

As mentioned above, the information problem lies with matching students' demand for tutoring with the appropriate availability of other students that are willing to tutor. In addition to matching time availability, ConnectedU will provide many other benefits to all parties involved. These include setting an affordable price range, verifying tutor's ability or skill-sets required to teach a course and estimating the longevity of the session. ConnectedU is a platform that will bring two parties together using the information offered up by the tutor and the information sought after by the tutee. By coordinating schedules and price ranges, this system will make it easier for students to find tutors in the subjects they need, at the times they are available, for a price they can afford.

Overall, this application will serve a student need that many universities are currently unable to meet: streamline the process for students to connect with tutors. The adoption of ConnectedU will not only benefit tutors and tutees, but it will also improve the overall academic climate at each respective university. It will foster a collaborative environment where students value each other's expertise and guidance, and struggling students feel comfortable reaching out for help with difficult courses. We believe that every student has an arsenal of talents that he or she should be able to share with others. Thus, allowing students to monetize their knowledge while gaining access to their peers' knowledge is a worthwhile venture and a value add for all universities around the country.

ER Diagram


University
University_ID
grade_level
University_Name
City
has
Department
Dept_ID
Name
Size
Budget
belongs to
Course
Course_Name
Course_ID
Transaction
Transaction_ID
Amount
Type_of_payment
Payment for
Involves
Teaches
Studies
Uses
Materials
Description
Cost
Material_ID
Tutoring
Bank Account
Rating
Participants
P_ID
LName
FName
Year
Major
University
Gender
DOB
Zip_Code
Availability
Time Slot
Time_ID
Day of Week
Start_Time
End_Time
Date
Actual Time
Tutee
Willingness to Pay
Tutor
GPA
Hourly Wage
O

Transformed Relational Model

After creating the ER diagram, we transformed each entity and relationship into a relational model. This helped us enter our data in our database in SQL. Based on our ER diagram, our team have evaluated 15 different relational tables needed for our database in the following:

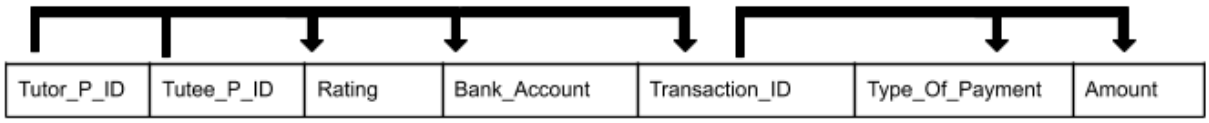
1. Participant (P_ID, University, Gender, DOB, Zip_Code, Major, Year, LName,)
2. Tutoring (Tutee_P_ID, Tutor_P_ID, Rating, Bank_Account, Course_ID, Transaction_ID, Dept_ID)
3. Time Slot (Time_ID, Start_Time, End_Time, Date)
4. Tutee (P_ID, Willingngess_to_Pay)
5. Tutor (P_ID), Hourly_Wage, GPA)
6. Materials (Material_ID, Cost, Description)
7. Uses (Tutor_P_ID, Tutee_P_ID, Material_ID)
8. Transaction (Transaction_ID, Type_of_payment, Amount)
9. Course (Course_ID, Dept_ID, Course_Name)
10. University (University_ID, Grade_Level, City, University_Name)
11. Department (Dept_ID, Name, Size, Budget, University_ID)
12. Availability (P_ID, Time_ID)
13. Studies (P_ID, Course_ID, Dept_ID)
14. Teaches (P_ID, Course_ID, Dept_ID)
15. Actual_Time (Time_ID, Tutee_P_ID, Tutor_P_ID)

Examples of BCNF relations

After creating the relational model, we had to ensure that all tables were normalized to BCNF.

Below are three examples of BCNF relations:

Example 1: Participant and Transaction Relations



a. **Normalize to 1NF**

(Tutor_P_ID, Tutee_P_ID, Transaction_ID, Rating, Bank_Account, Type_Of_Payment, Amount)

b. **Normalize to 2NF** - No partial dependency

R1(Tutor_P_ID, Tutee_P_ID,Transaction_ID,Rating, Bank_Account)

R2(Transaction_ID, Type_of_Payment, Amount)

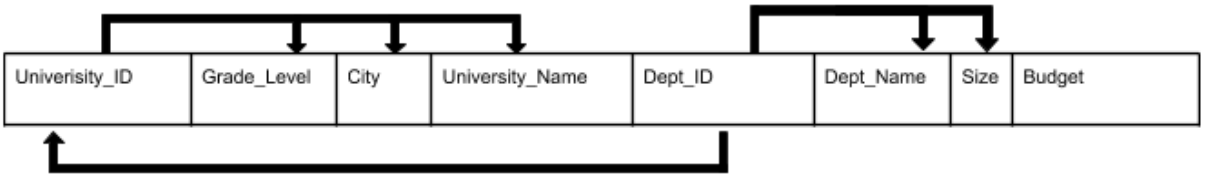
c. **Normalize to 3NF**- No transitive Dependency

No transitive Dependency, therefore 3NF satisfied

d. **Normalize to BCNF**- All determinants are a primary key

Therefore, the 3NF relation is in BCNF.

Example 2: University and Department Relations



a. **Normalize to 1NF**

(University_ID, Dept_ID, Grade_Level, City, University_Name, Dept_Name, Size, Budget)

b. **Normalize to 2NF-** no partial dependency

R1(University_ID, Grade_Level, City, University_Name)

R2(Dep_ID, Name, Size, Budget, University_ID)

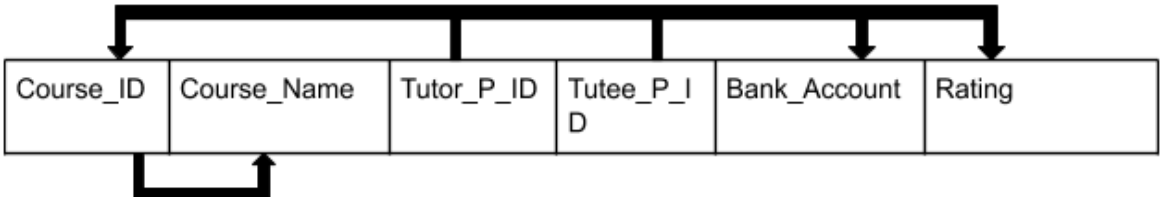
c. **Normalize to 3NF** - No transitive Dependency

Same as 2NF

d. **Normalize to BCNF**- All determinants are a primary key

Therefore, the 3NF relation is in BCNF

Example 3: Course and Tutoring Table



a. **Normalize to 1NF**

(Course_ID, Tutor_P_ID, Tutee_P_ID, Course_Name, Bank_Account, Rating)

b. **Normalize to 2NF**

R1 (Course_ID, Course_Name)

R2 (Tutor_P_ID, Tutee_P_ID, Course_ID, Bank_Account, Rating)

c. **Normalize to 3NF**

Same as 2NF

d. **Normalize to BCNF**

Every Determinant is a candidate key. Therefore

10 SQL Query Statements

Our team has developed 10 different relevant and complex SQL statements in the following:

1. **For each tutor, list their names (first and last) and their corresponding tutees' name.**

```
Select distinct A.Fname, A.Lname, B.Fname, B.Lname
From Tutoring, Participants A, Participants B
Where Tutee_P_ID=A.P_ID
and Tutor_P_ID=B.P_ID
```

Tutor_Fname	Tutor_Lname	Tutee_Fname	Tutee_Lname
Joseph	Lu	Lena	Lin
Joseph	Lu	Matthew	Merrick
Julia	Frisch	Lena	Lin
Julia	Frisch	Matthew	Merrick
Kyle	Guy	Sam	Marzyie
Kyle	Guy	Suprateek	Sarker
Lena	Lin	Suprateek	Sarker
Matthew	Merrick	Moh	Ahmed
Matthew	Merrick	Sam	Marzyie
Sam	Marzyie	John	Smith

2. **List the names of the tutors who tutors more than 2 tutees AND have a rating of 5.**

```
Select Fname, Lname
From Tutoring, Participants
Where Tutor_P_ID=Participants.P_ID and
Rating=5
Group by Fname, Lname
Having count (tutor_p_id)>2
```

Fname	Lname
Julia	Frisch

3. **How many tutoring sessions have 5 star ratings with tutor's GPA greater than 3.7?**

```
Select count (rating)
From tutoring, tutor
Where rating =5 and
Tutor_P_ID =tutor.P_ID and
GPA >3.7
```

Column1
5

4. Tutors with at least one 5 star rating are given a 10% raise on their price. Change the database accordingly. Next, show the contents of the table.

```
Update Tutor
Set Hourly_Wage=1.1*Hourly_Wage
From tutoring
Where tutoring.tutor_P_ID= tutor.P_ID and
rating=5

select*
From tutor
```

	P_ID	GPA	Hourly_Wage
▶	1	4.9	10
	2	3.9	12
	3	3.8	16
	4	4	20
	5	3.5	11
	6	3.8	13

5. List all the participants (Fname, Lname) who are born after 11-20-1997 and their school year.

```
Select Fname, Lname, Year
From Participants
Where DOB > '11-20-1997'
```

	Fname	Lname	Year
▶	Sam	Marzyie	4th
	Joseph	Lu	4th
	Kyle	Guy	Grad
	Julia	Frisch	4th
	Suprateek	Sarker	Grad
	John	Smith	2nd

6. Delete all the participants who are 20 years old and who's willing to pay is less than $20. Show the content of the table.

```
Delete from Participants
Where datediff(yy, DOB, getdate())=20
and
P_ID in (
        Select P_ID
        From tutee
        Where willingness_to_pay <= 20)

Select *
From Participants
```

	Fname	Lname	P_ID	University_ID	Gender	DOB	Zip_code	Major	Year
▶	Matthew	Merrick	1	55201	M	5/3/1997 12:00:00 AM	75225	Finance	4th
	Lena	Lin	2	55201	F	7/17/1997 12:00:00 AM	12538	Information technology	4th
	Sam	Marzyie	3	55201	M	1/16/1998 12:00:00 AM	18447	Marketing	4th
	Joseph	Lu	4	55201	M	3/8/1998 12:00:00 AM	97226	Database Management	4th
	Kyle	Guy	5	22303	M	2/19/1998 12:00:00 AM	57922	Neuroscience	Grad
	Julia	Frisch	6	55201	F	6/4/1998 12:00:00 AM	11950	Marketing	4th
	Moh	Ahmed	7	11111	M	3/19/1997 12:00:00 AM	69393	Public Policy	3rd
	Suprateek	Sarker	8	12989	M	4/17/1998 12:00:00 AM	22903	Computer Science	Grad

7. List the names of all participants who are available on Monday, 1/13/2020 as well as their time availability.

```
Select Fname, Lname, start_time, end_time
From Participants, availability, time_slot
Where participants.P_ID=Availability.P_ID and
availability.time_ID=Time_slot.Time_ID
And date = '1/13/2020'
Group by Fname, Lname, start_time, end_time
```

	Fname	Lname	start_time	end_time
▶	John	Smith	23:00:00	00:00:00
	Lena	Lin	17:00:00	18:00:00
	Lena	Lin	23:00:00	00:00:00
	Moh	Ahmed	23:00:00	00:00:00
	Sam	Marzyie	21:00:00	22:00:00
	Suprateek	Sarker	19:00:00	20:00:00
	Suprateek	Sarker	23:00:00	00:00:00

8. Which of the tutors who are fourth years (Fname, Lname) that are available on any weekday after 5?

```
Select Fname, Lname
From tutor, Participants
Where tutor.P_ID=participants.P_ID
And year = '4th'
Intersect
Select Fname, Lname
From Participants, Availability, Time_slot
Where Participants.P_ID=Availability.P_ID
And Availability.Time_ID=Time_slot.Time_ID
And start_time >= '17:00:00'
```

	Fname	Lname
▶	Joseph	Lu
	Julia	Frisch
	Lena	Lin
	Matthew	Merrick
	Sam	Marzyie

9. List the names of the tutors who used materials during the tutoring sessions, the name of the materials, as well as the cost of both materials and their rating for that session.

```
Select Fname, Lname, Description, Cost, Rating
From Participants, Uses, Material, Tutoring
Where P_ID=Uses.Tutor_P_ID and
Uses.Material_ID=Material.Material_ID
And P_ID =Tutoring.Tutor_P_ID
```

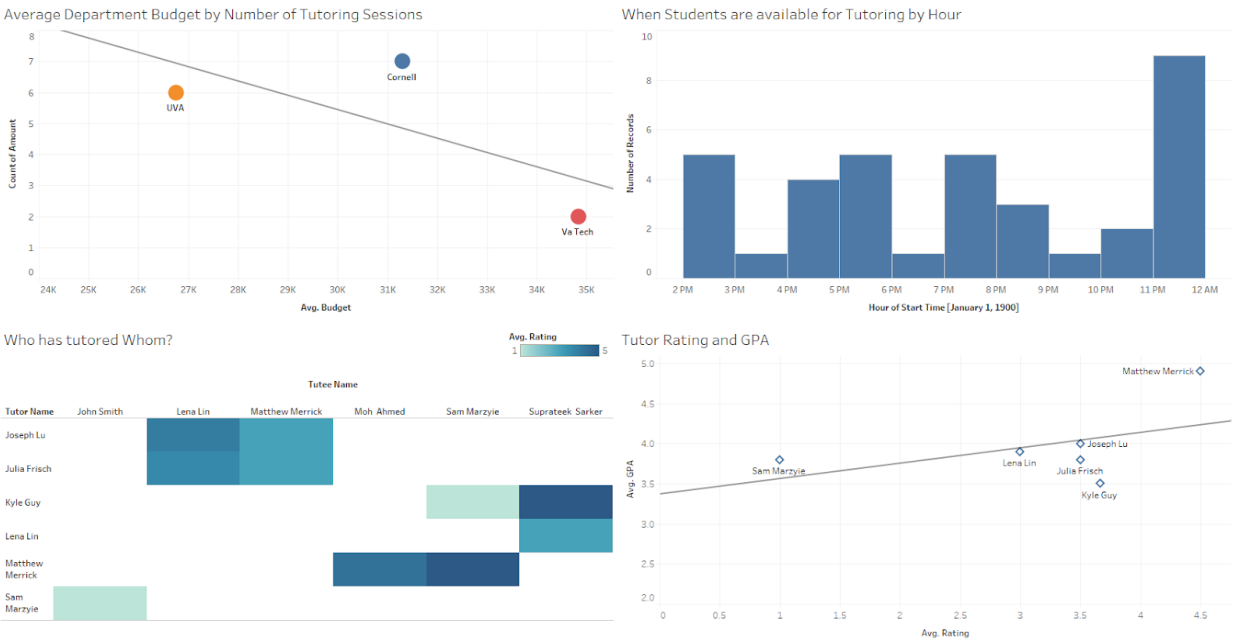
Fname	Lname	Description	Cost	Rating
Matthew	Merrick	Textbook	50	3
Lena	Lin	Course Pack	40	4
Sam	Marzyie	Ebook	25	5
Joseph	Lu	Chegg	60	2
Kyle	Guy	Textbook	15	4
Julia	Frisch	Course Pack	30	5
Matthew	Merrick	Textbook	50	3
Matthew	Merrick	Textbook	50	3
Matthew	Merrick	Textbook	50	3
Joseph	Lu	Chegg	60	2
Kyle	Guy	Textbook	15	5
Kyle	Guy	Textbook	15	5
Julia	Frisch	Course Pack	30	5
Julia	Frisch	Course Pack	30	5
Julia	Frisch	Course Pack	30	5

10. List each tutor name along with the number of sessions they have completed.

```
Select Fname, Lname, count (transaction_ID)
From participants, tutoring
Where Participants.P_ID=tutoring.Tutor_P_ID
Group By Fname, Lname
```

Fname	Lname	num_session
Joseph	Lu	2
Julia	Frisch	4
Kyle	Guy	3
Lena	Lin	1
Matthew	Merrick	4
Sam	Marzyie	1

Tableau Dashboard



Here, we present a Tableau dashboard that provides information that would be of use to a school administrator who is interested in finding out about how their school and its tutors are performing. Specifically, we have chosen one at UVA who wants to compare how UVA is doing compared to other schools, and dive deeper into each tutor at UVA. First, in the top left, we see a comparison of 3 different schools on budget and number of tutoring sessions. As we can see, there seems to be a general trend of fewer tutoring sessions with higher budgets, which makes sense if we assume that better endowed departments tend to have better teachers, which leads to lower need for tutors.

Next, on the top right, we see the most common times for tutoring, with 11-12 being most popular. This information might spur the administrator to create more programs for late night academic help, or ask professors to be available at times currently unfilled by tutors such as 3-4pm, when the supply of tutors is lowest.

Finally, on the bottom, we see an overview of what tutoring is going on around grounds, linking each tutor to tutee in case the administrator wants to review individual tutors or tutees, and a useful scatterplot which shows the positive relationship between tutor GPA and rating, meaning the administrator might want to limit tutors to a certain GPA and above, or create a team to dive deeper into what makes a good tutor.

Reflections and Conclusions

Over the course of the semester, our team has successfully developed an idea for an application and designed the backend of the information system. Through ConnectEDU, we will serve an unmet need for students by creating a platform to match tutors with tutees, plan tutoring sessions, and manage payments. This will provide peer-to-peer tutoring throughout the country and will reduce cost while increasing efficiency for students that need assistance outside of class.

Throughout this hands-on process, the team gained a greater appreciation for the complexities of database design. It was rewarding to complete the process from start to finish, beginning with app ideation and ending with a comprehensive Tableau dashboard. The team became much more comfortable with ER diagrams, SQL queries, and Tableau throughout the process. Especially because we not only conceived the idea, but manually entered all relevant information that would be included in the information system, this project allowed us to apply each of the compartmentalized class concepts into one fluid assignment so that we could understand the process from A to Z.

Our team has worked well together throughout the project by collaborating on tasks including the ER diagram and the creation of the relational tables, then assigning other tasks such as query writing, the Tableau dashboard, and report writing fairly. This allowed the group to effectively manage our time while producing a successful project. Even though we delegated tasks, every group member effectively communicated the steps that he/she completed such that everyone was on the same page. After completing this project, we are all confident that in preparing on a real-world application of database design in the future.

Exhibits: SQL Initial Data Results

University

University_ID	grade_level	University_Name	City
55201	Undergraduate	UVA	Charlottesville
22303	Grad	UPenn	Philadelphia
11111	Undergraduate	Va Tech	Blacksburg
12989	Grad	Cornell	Ithaca
45441	Undergraduate	VCU	Richmond

Courses

Course_ID	Dept_ID	Course_Name
1	678	Calculus I
2	456	Intro to Computer Science
3	345	Architectural Design
4	123	Corporate Finance
5	345	Architectural History
6	123	Intermediate Accounting I
7	123	Advanced Accounting
8	123	Federal Taxation
9	456	Intro to Engineering
10	678	Calculus II
11	678	Linear Algebra
12	678	Discrete Math
13	456	Intro to Java
14	345	History of Architecture in Eastern Europe
15	124	Intermediate Investments
16	333	Intro to Python
17	123	Intermediate Accounting II
18	123	Managerial Accounting
19	123	Financial Accounting

Tutor

P_ID	GPA	Hourly Wage
1	4.9	10
2	3.9	12
3	3.8	15
4	4	20
5	3.5	10
6	3.8	12

Department

University_ID	Dept_ID	Name	Size	Budget
55201	123	Commerce	561	26752.84
11111	456	Engineering	405	34833.06
55201	345	Architecture	453	24695.48
12989	678	Mathematics	250	31295.43
45441	124	Business	330	30607.65
55201	333	Engineering	500	40977.62

Transaction

Transaction_ID	Type_of_payment	Amount
1	Credit	40
2	Debit	75
3	Credit	50
4	Debit	25
5	Credit	40
6	Credit	30
7	Debit	35
8	Credit	32
9	Debit	0
10	Credit	50
11	Credit	20
12	Debit	25
13	Credit	40
14	Debit	30
15	Credit	35

Participants

Fname	Lname	P_ID	University_ID	Gender	DOB	Zip_code	Major	Year
Matthew	Merrick	1	55201	M	5/3/1997 12:00:00 AM	75225	Finance	4th
Lena	Lin	2	55201	F	7/17/1997 12:00:00 AM	12538	Information technology	4th
Sam	Marzyie	3	55201	M	1/16/1998 12:00:00 AM	18447	Marketing	4th
Joseph	Lu	4	55201	M	3/8/1998 12:00:00 AM	97226	Database Management	4th
Kyle	Guy	5	22303	M	2/19/1998 12:00:00 AM	57922	Neuroscience	Grad
Julia	Frisch	6	55201	F	6/4/1998 12:00:00 AM	11950	Marketing	4th
Moh	Ahmed	7	11111	M	3/19/1997 12:00:00 AM	69393	Public Policy	3rd
Suprateek	Sarker	8	12989	M	4/17/1998 12:00:00 AM	22903	Computer Science	Grad
John	Smith	9	45441	M	1/1/1999 12:00:00 AM	12345	Architecture	2nd

Availability

P_ID	Time_ID
1	24
1	26
1	40
1	44
2	4
2	10
2	20
2	39
3	8
3	11
3	33
3	43
4	20
4	21
4	26
4	39
5	11
5	22
5	27
5	36
6	26
6	34
6	45
6	50
7	10
7	30
7	31
7	44
8	6
8	10
8	21
8	43
9	10
9	17
9	37
9	43

Studies

Tutee_P_ID	Course_ID	Dept_ID
1	1	678
2	6	123
3	19	123
4	6	123
5	12	678
6	15	124
7	18	123
8	7	123
9	18	123
5	10	678
2	9	456
4	7	123
6	14	345
7	5	345
8	6	123
9	12	678
1	10	678
2	3	345

Material

Material_ID	Description	Cost
123	Textbook	50
321	Course Pack	40
224	Ebook	25
445	Chegg	60
667	Textbook	15
494	Course Pack	30
933	Connect	45
100	Course Pack	10
221	Ebook	55

Teaches

Tutor_P_ID	Course_ID	Dept_ID
1	6	1
2	7	2
3	8	3
4	9	4
5	10	5
6	11	6
1	6	7
1	6	8
1	17	9
4	9	10
5	10	11
5	10	12
6	11	13
6	11	14
6	11	15

Tutee

P_ID	Willingness_to_Pay
1	15
2	20
3	25
7	15
8	50
9	20

Uses

tutor_P_ID	Tutee_P_ID	Material_ID
1	7	123
2	8	321
3	9	224
4	2	445
5	3	667
6	1	494
7	2	933
8	3	100
9	4	221

Actual Time

Time_ID	P_ID	P_ID.1
1	1	7
2	2	8
3	3	9
4	4	2
5	5	3
6	6	1

Tutoring

Tutor_P_ID	Tutee_P_ID	Bank_Account	Rating	Course_ID	Transaction_ID
1	7	11223344	3	6	1
2	8	55667788	4	7	2
3	9	77889900	5	8	3
4	2	90909090	2	9	4
5	3	12121212	4	10	5
6	1	12897894	5	11	6
1	7	11223344	3	6	7
1	7	11223344	3	6	8
1	3	11223344	3	17	9
4	1	90909090	2	9	10
5	8	12121212	5	10	11
5	8	12121212	5	10	12
6	2	12897894	5	11	13
6	2	12897894	5	11	14
6	2	12897894	5	11	15

Time Slot

Time_ID	Start_Time	End_Time	Date
1	14:00:00	15:00:00	1/13/2020 12:00:00 AM
2	15:00:00	16:00:00	1/13/2020 12:00:00 AM
3	16:00:00	17:00:00	1/13/2020 12:00:00 AM
4	17:00:00	18:00:00	1/13/2020 12:00:00 AM
5	18:00:00	19:00:00	1/13/2020 12:00:00 AM
6	19:00:00	20:00:00	1/13/2020 12:00:00 AM
7	20:00:00	21:00:00	1/13/2020 12:00:00 AM
8	21:00:00	22:00:00	1/13/2020 12:00:00 AM
9	22:00:00	23:00:00	1/13/2020 12:00:00 AM
10	23:00:00	00:00:00	1/13/2020 12:00:00 AM
11	14:00:00	15:00:00	1/14/2020 12:00:00 AM
12	15:00:00	16:00:00	1/14/2020 12:00:00 AM
13	16:00:00	17:00:00	1/14/2020 12:00:00 AM
14	17:00:00	18:00:00	1/14/2020 12:00:00 AM
15	18:00:00	19:00:00	1/14/2020 12:00:00 AM
16	19:00:00	20:00:00	1/14/2020 12:00:00 AM
17	20:00:00	21:00:00	1/14/2020 12:00:00 AM
18	21:00:00	22:00:00	1/14/2020 12:00:00 AM
19	22:00:00	23:00:00	1/14/2020 12:00:00 AM
20	23:00:00	00:00:00	1/14/2020 12:00:00 AM
21	14:00:00	15:00:00	1/15/2020 12:00:00 AM
22	15:00:00	16:00:00	1/15/2020 12:00:00 AM
23	16:00:00	17:00:00	1/15/2020 12:00:00 AM
24	17:00:00	18:00:00	1/15/2020 12:00:00 AM
25	18:00:00	19:00:00	1/15/2020 12:00:00 AM
26	19:00:00	20:00:00	1/15/2020 12:00:00 AM
27	20:00:00	21:00:00	1/15/2020 12:00:00 AM
28	21:00:00	22:00:00	1/15/2020 12:00:00 AM
29	22:00:00	23:00:00	1/15/2020 12:00:00 AM
30	23:00:00	00:00:00	1/15/2020 12:00:00 AM
31	14:00:00	15:00:00	1/16/2020 12:00:00 AM
32	15:00:00	16:00:00	1/16/2020 12:00:00 AM
33	16:00:00	17:00:00	1/16/2020 12:00:00 AM
34	17:00:00	18:00:00	1/16/2020 12:00:00 AM
35	18:00:00	19:00:00	1/16/2020 12:00:00 AM
36	19:00:00	20:00:00	1/16/2020 12:00:00 AM
37	20:00:00	21:00:00	1/16/2020 12:00:00 AM
38	21:00:00	22:00:00	1/16/2020 12:00:00 AM
39	22:00:00	23:00:00	1/16/2020 12:00:00 AM
40	23:00:00	00:00:00	1/16/2020 12:00:00 AM
41	14:00:00	15:00:00	1/17/2020 12:00:00 AM
42	15:00:00	16:00:00	1/17/2020 12:00:00 AM
43	16:00:00	17:00:00	1/17/2020 12:00:00 AM
44	17:00:00	18:00:00	1/17/2020 12:00:00 AM
45	18:00:00	19:00:00	1/17/2020 12:00:00 AM
46	19:00:00	20:00:00	1/17/2020 12:00:00 AM
47	20:00:00	21:00:00	1/17/2020 12:00:00 AM
48	21:00:00	22:00:00	1/17/2020 12:00:00 AM