Database Final Project

Matthew Lee, Rishabh Meswani, Spencer Wilson

Database Description

Introduction

Our group created a database focused on helping those looking to recruit in the job market today from the perspective of a UC Berkeley student. Now more than ever, we are relying on data to help provide us with insight on the current state of employment. UC Berkeley is the perfect place to gather and deploy this database because we have historically gone on to a diverse array of positions in the job market. Being a fairly large student body, the breadth of potential data input is massive. Ideally, the database would consist of graduates from a wide array of majors and disciplines around campus and demonstrate how many jobs and companies there truly are out there. We hope this database can someday be used by an official administrative body of the University such as the ASUC or Career Center to improve the accessibility of such data to the general student population.

We were partly inspired to build this database by the current, more low-technology resources available to the public on the UC Berkeley Career Fair website. There, visitors can click on a link for each major and view a pdf for each of the last few years containing data about students in that program who volunteered to share their data anonymously. Each entry is a row in a table that includes their post-grad plans and where they plan on working, if applicable. For instance, the Molecular and Cell Biology pdf contains many students who were planning on furthering their studies at medical school or graduate programs related to health sciences, whereas computer science and EECS graduates often went-on to work at large tech giants such as Google or Amazon (for simplicity sake, we decided to focus on individuals entering the workforce immediately after graduation). This website, although filled with quality information, is

difficult to access and filled with too many pdfs. I remember when trying to decide on a major, I had opened several tabs for only a single pdf each and most files contained very limited information. We decided to make our database much simpler and accessible for a wide audience. One aspect of the website we greatly appreciated, however, was the addition of an average salary for each major program. We decided that we wanted to use this in our own project and actually used an average salary function and rounding method as well in the final product.

What does our platform do?

Our plans on how to organize the database changed over time. Initially, we were thinking about creating a form that was used potentially by both employers and potential employees. However, over time, we realized we wanted to focus the experience more on the potential employees, especially given how COVID-19 has impacted the job market so negatively as so many soon-to-be grads are scrambling to find employment. We also realized that after doing some research about what students are most interested in about a job, one of the first things they mention is naturally salary. Therefore, we decided to make the average salary by occupation (for UC Berkeley alumni) one of the focal points of our database. We also have an option for them to find openings that recruiters/companies are targeting UC Berkeley students for. Furthermore, in times like these, students want to seek out guidance from alumni mentors who may be working at companies they hope to work at too. For instance, many of our friends are currently reaching out to current employees at nearby companies to coffee chat or receive resume reviews even virtually nowadays in the hopes of gaining some insight on that job lifestyle. Therefore, we wanted to include the option for students to input their companies of interest into a text box and see the names of UC Berkeley alumni who are currently or were previously employed in a certain company. We chose to not include contact information in this iteration of the database because in reality, a publically-available resource would not advertise the emails and/or phone numbers of professionals in the workforce. Rather, we believe a

smarter system would be for students to be able to reach out to the UC Berkeley Career Center and receive permission to view certain requested individuals' contact information. This would both protect the privacy of professionals as well as encourage other alumni of UC Berkeley to input their information into this database without fear.

Database Organization

Our database is organized into 11 tables. They are as follows:

Industries
The Industries entity contains the different sectors of the economy companies operate within. We found which industries are the most common amongst UC Berkeley graduates and assigned an ID value to each.

Companies
The Companies entity contains the names and scale of companies people work for, have worked for, or have applied to. We collected data from public websites and datasets about important numerical attributes about these companies such as size.

Companies_Industries
The Companies_Industries entity serves as a junction between companies and their industries.

Offices
The Offices entity contains the locations and headcount of where the companies' employees work. We collected this information from corporate websites of the most popular companies of employment for UC Berkeley graduates.

Education_Levels
The Education_Levels entity contains the different levels of educational attainment people have achieved. We assigned an ID to each completed highest level of education.

People
The People entity contains the names and skills of everyone who has submitted their information. The data was collected about individuals who represent a diverse array of

skills and levels of education completed and some of whom are employed at popular companies.

People_Education
The People_Education entity serves as a junction between people and the highest level of education they have achieved.

Professions
The Professions entity contains the name, general description, and average salary of professions people hold or have held. This data is collected from online anonymous crowdsourced information such as LinkedIn and Glassdoor.

Positions
The Positions entity contains current and past positions people have held, what their profession was, which office they worked from, the salary they earned, the date they began working, and the end date if a data point represents past employment. The data is collected from publicly-available sources online about common positions held in today's economy.

Openings
The Openings entity contains open positions, the position's profession, which office the position is for, and a description detailing expectations and desired qualifications. The data was collected from popular job posting sites.

Applications
The Applications entity contains the results of people's applications as a particular profession to an office, and the years of experience they had before applying. This data comes from submitted information.

Future Steps

There are a few key changes we would implement if we were to continue developing this database. For instance, we would increase the functionality of the webpage by harnessing the power of joining more tables across our database together. For instance, we would consider looking at which industries are looking to hire more applicants during some months and fewer people during other months. We also would hope to make it possible to easily enter data from the user side into the database so that more and more

individuals could easily submit their information (given certain security, of course) so make our data more robust for others (as imagined, we would have to create some POST and PUT requests for this). If people have developed significantly since submitting an application, adding context to each application would aid in comparison. One of our later decisions was to disconnect applications from openings and link them directly to professions and offices. The thought was applications are historic and openings are current, but adding a boolean attribute for active to openings would solve this issue while making the more natural connection. It would also provide information on the opening's expectation that isn't currently available when looking at applications. We would restore that connection and retain more data than before by keeping the history of openings. We would also create some mechanisms for a future employee to contact a former UC Berkeley alumni directly through the website.

Data Dictionary

Industries

Attribute	**Data Type**	**Constraint**	**Description**
Industry_Id	int	PK	Industry Id; auto incremented
Industry_Name	varchar(255)	Not null	Industry name

Companies

Attribute	**Data Type**	**Constraint**	**Description**
Company_Id	int	PK	Company Id; auto incremented
Company_Name	varchar(255)	Not null	Company name
Company_Size	mediumint		Number of employees

Companies_Industries

Attribute	**Data Type**	**Constraint**	**Description**

Company_Id	int	FK	References an existing Company_Id from Companies
Industry_Id	int	FK	References an existing Industry_Id from Industries

Offices

Attribute	Data Type	Constraint	Description
Office_Id	int	PK	Office Id; auto incremented
Company_Id	int	FK	References an existing Company_Id from Companies
Address	varchar(100)	Not null	Street address
City	varchar(35)	Not null	City name
State	char(2)		Two character state or region code
Zip	char(5)		Zip or postal code
Country	char(3)	Not null	Three digit country code
Size	smallint		Number of employees

Education_Levels

Attribute	Data Type	Constraint	Description
Education_Id	int	PK	Education Id; auto incremented
Education_Name	varchar(255)	Not null	Name of degree or level of attainment

People

Attribute	Data Type	Constraint	Description
Person_Id	int	PK	Person Id; auto incremented
Person_Name	varchar(255)	Not null	Person’s name
Skills	text		List and explanations of a person’s professional skills

People_Education

Attribute	Data Type	Constraint	Description
Person_Id	int	FK	References an existing Person_Id from People
Education_Id	int	FK	References an existing Education_Id from Education_Levels

Professions

Attribute	Data Type	Constraint	Description
Profession_Id	int	PK	Profession Id; auto incremented
Profession_Name	varchar(255)	Not null	Profession Name
Profession_Description	text		Describes the function of a profession
Average_Salary	int		Average yearly professional salary in dollars

Positions

Attribute	Data Type	Constraint	Description
Position_Id	int	PK	Position Id; auto incremented
Profession_Id	int	FK	References an existing Profession_Id from Professions
Office_Id	int	FK	References an existing Office_Id from Offices
Person_Id	int	FK	References an existing Person_Id from People
Salary	int	Not null	Yearly salary in dollars
Start_Date	date	Not null	Date began working; formatted YYYY-MM-DD
End_Date	date		Date stopped working; formatted YYYY-MM-DD; null if currently working

Openings

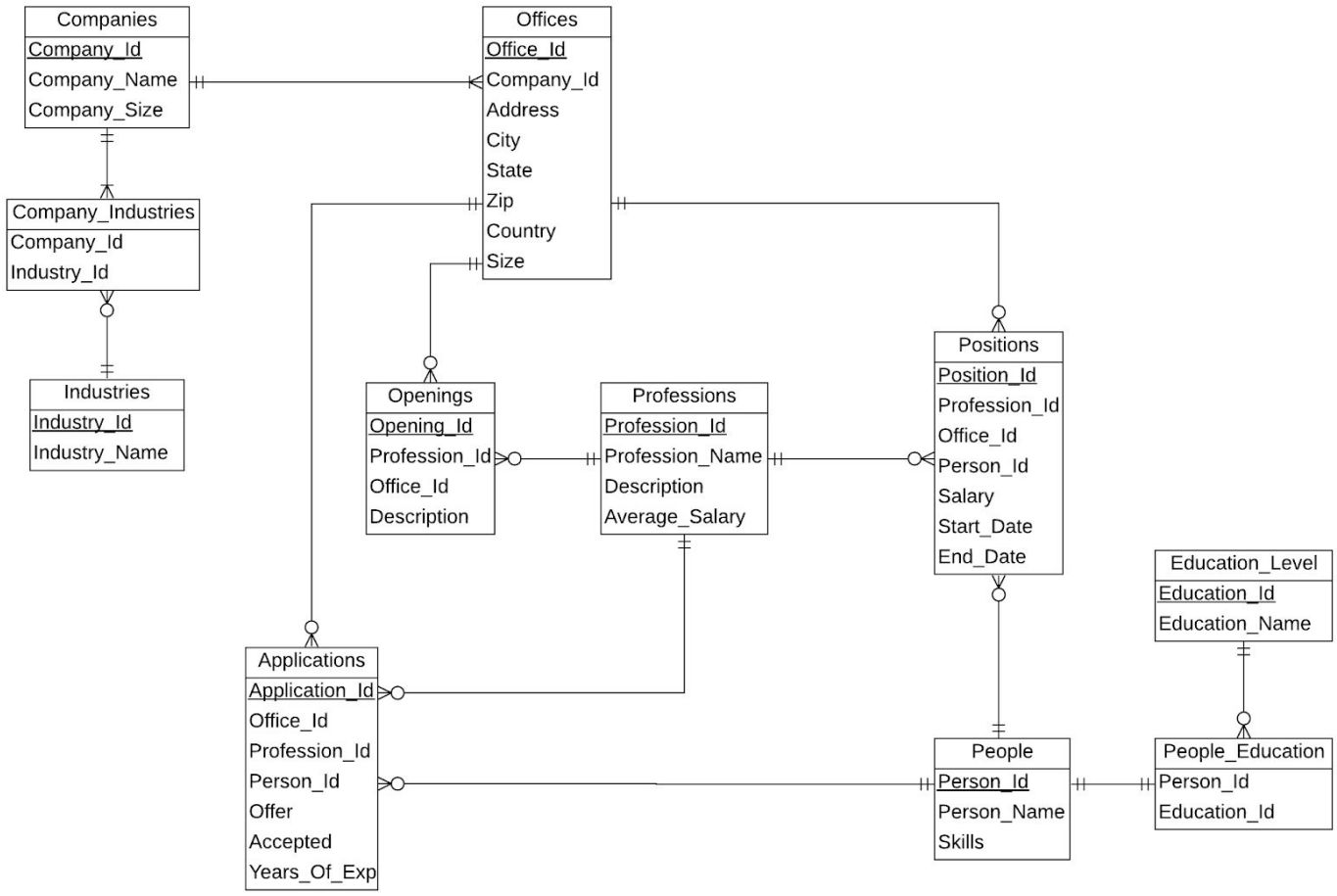
Attributes	Data Type	Constraint	Description
Opening_Id	int	PK	Opening Id; auto incremented
Profession_Id	int	FK	References an existing Profession_Id from Professions
Office_Id	int	FK	References an existing Office_Id from Offices
Opening_Description	text	Not null	Describes what the

			open position entails, what qualifications are expected, and any other relevant information e.g. benefits

Applications

Attributes	**Data Type**	**Constraint**	**Description**
Application_Id	int	PK	Application Id; auto incremented
Office_Id	int	FK	References an existing Office_Id from Offices
Profession_Id	int	FK	References an existing Profession_Id from Professions
Person_Id	int	FK	References an existing Person_Id from People
Offer	binary	Not null	Indicates whether an offer was given
Accepted	binary	Not null	Indicates whether an offer was accepted; always 0 if the application was denied
Years_Of_Experience	smallint	Not null	Years of relevant work experience prior to applying

Relationships Diagram



Sample Queries and Results

1) Query: What openings exist for a certain type of profession? (in this case, Software Engineering is picked from the drop-down menu).

```
SELECT Company_Name, Opening_Description, Profession_Name FROM Professions
INNER JOIN Openings ON Professions.Profession_Id = Openings.Profession_Id
INNER JOIN Offices O on Openings.Office_Id = O.Office_Id INNER JOIN Companies
C on O.Company_Id = C.Company_Id WHERE Profession_Name = 'Software
Engineering'
```

Result:

	Company_Name	Opening_Description	Profession_Name
1	Chick-fil-A	Working on AWS.	Software Engineering
2	Epic	Focusing on AWS Lambda Functions.	Software Engineering
3	Chick-fil-A	Developing in Java.	Software Engineering
4	McDonalds	Developing in Kotlin.	Software Engineering
5	Facebook	Developing in C.	Software Engineering
6	LinkedIn	Developing in C#.	Software Engineering
7	Epic	Creating database tools.	Software Engineering
8	Chick-fil-A	Developing in Python.	Software Engineering
9	McDonalds	Developing Azure tools.	Software Engineering
10	Facebook	Developing back-end infrastructure tools.	Software Engineering
11	LinkedIn	Developing in C++.	Software Engineering
12	LinkedIn	Developing infrastructure platforms.	Software Engineering

2) Query: Out of the UC Berkeley alumni that reported their positions, what is the average salary by profession (in this case, Software Engineering was picked from the drop-down menu)?

```
SELECT Profession_Name, ROUND(AVG(Positions.Salary)) as 'Average Salary' FROM
Positions INNER JOIN Professions P on Positions.Profession_Id =
P.Profession_Id WHERE Profession_Name = 'Software Engineering'
```

Result:

	Profession_Name	`Average Salary`
1	Software Engineering	126000

3) Query: What are the names and professions of all UC Berkeley alumni who work or have worked at a specific company? Apple, for example?

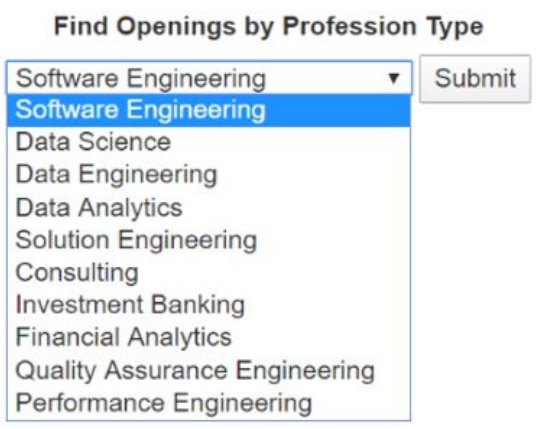
```
SELECT Person_Name, Profession_Name FROM People INNER JOIN Positions P on
People.Person_Id = P.Person_Id INNER JOIN Offices O on P.Office_Id =
O.Office_Id INNER JOIN Companies C on O.Company_Id = C.Company_Id INNER JOIN
Positions on People.Person_Id = Positions.Person_Id INNER JOIN Professions on
Positions.Profession_Id = Professions.Profession_Id WHERE Company_Name =
'Apple'
```

Result:

	Person_Name	Profession_Name
1	Julien Smith	Data Science
2	Quinton Wessells	Data Engineering
3	Megan Gillfillan	Financial Analytics
4	Rose Friedman	Quality Assurance Engineering
5	Charlotte Guerry	Performance Engineering

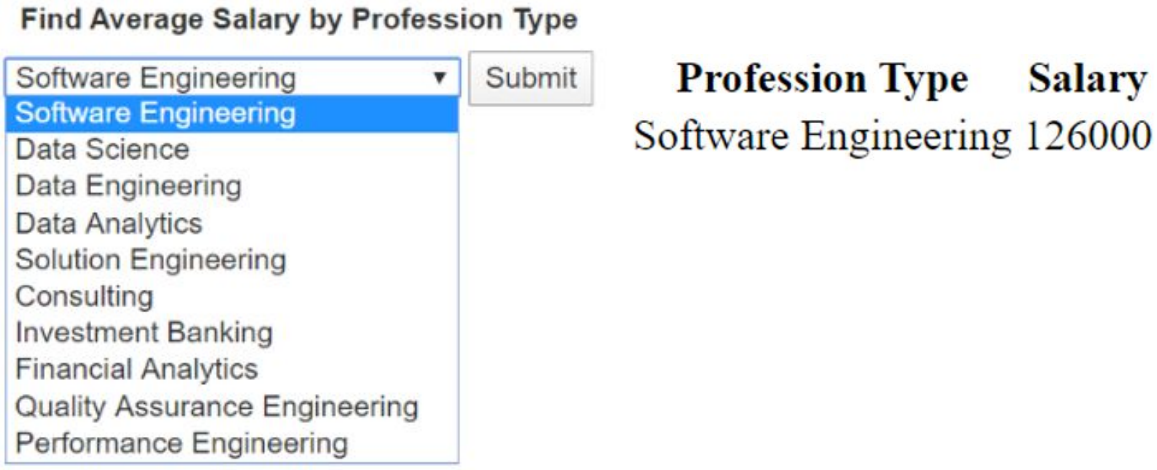
Sample Forms

1) What openings exist for a particular profession?

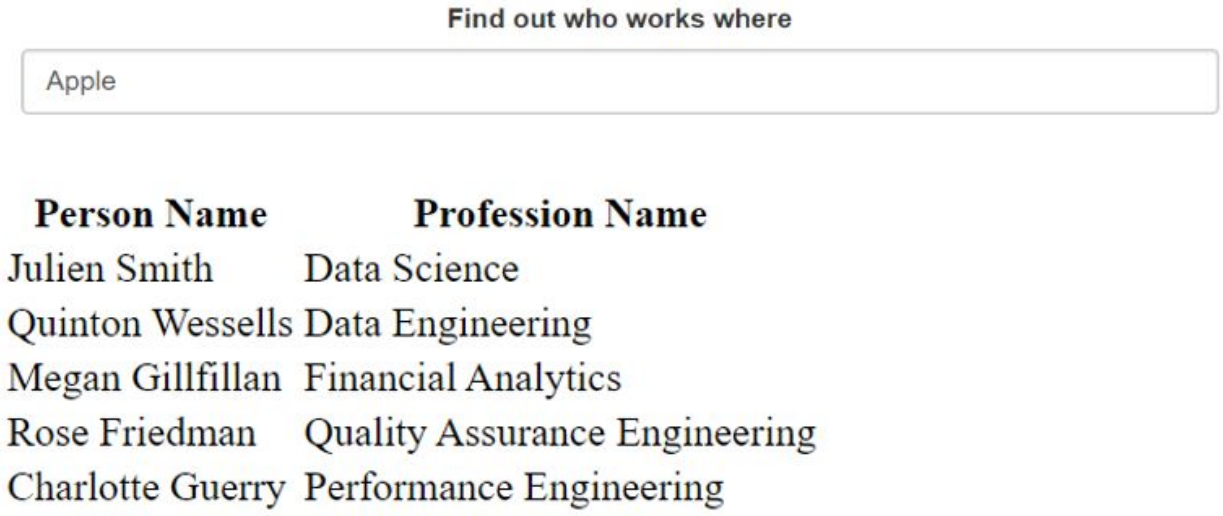


Company Name	Description
Chick-fil-A	Working on AWS.
Epic	Focusing on AWS Lambda Functions.
Chick-fil-A	Developing in Java.
McDonalds	Developing in Kotlin.
Facebook	Developing in C.
LinkedIn	Developing in C#.
Epic	Creating database tools.
Chick-fil-A	Developing in Python.
McDonalds	Developing Azure tools.
Facebook	Developing back-end infrastructure tools.
LinkedIn	Developing in C++.
LinkedIn	Developing infrastructure platforms.

2) What is the average salary of UC Berkeley alumni in a certain profession?



3) What UC Berkeley alumni work at a specific company?



Sample Reports

1) For each education level, how many Berkeley alumni have reported their positions and for these positions, what is the average salary by education level?

```
SELECT People_Education.Education_Id, Education_Name,
COUNT(People_Education.Education_Id) as 'Amount per Education Level',
AVG(Salary) as ‘Average Salary’ FROM People_Education INNER JOIN Positions P
on People_Education.Person_Id = P.Person_Id  INNER JOIN Education_Levels EL on
People_Education.Education_Id = EL.Education_Id   GROUP BY Education_Id;
```

Result:

	Education_Id	Education_Name	`Amount per Education Level`	`AVG(Salary)`
1	1	High School	1	68000.0000
2	2	Some College	7	108357.1429
3	3	Bachelor's	15	105266.6667
4	4	Masters	10	95400.0000
5	5	Doctorate	5	94400.0000

2) What are the top 5 offices with most openings? How many openings do they have, and where are they?

```
SELECT O.Office_Id, Company_Name, COUNT(Openings.Office_Id) as 'Amount of
Openings', City, State, Country FROM Openings INNER JOIN Offices O on
Openings.Office_Id = O.Office_Id INNER JOIN Companies C on O.Company_Id =
C.Company_Id GROUP BY Openings.Office_Id ORDER BY 'Amount of Openings' DESC
LIMIT 5;
```

Result:

	Office_Id	Company_Name	`Amount of Openings`	City	State	Country
1	2	Epic	4	Verona	WI	USA
2	3	Chick-fil-A	4	Atlanta	GA	USA
3	4	McDonalds	6	Chicago	IL	USA
4	7	Facebook	6	Menlo Park	CA	USA
5	8	LinkedIn	7	Sunnyvale	CA	USA