**MIS 320 Team Project**

MIS 320 Final Project

Section C1

Drew Lamb, Elmer McElmeel, Benjamin Ng, Alec Jensen

**Table of Contents**

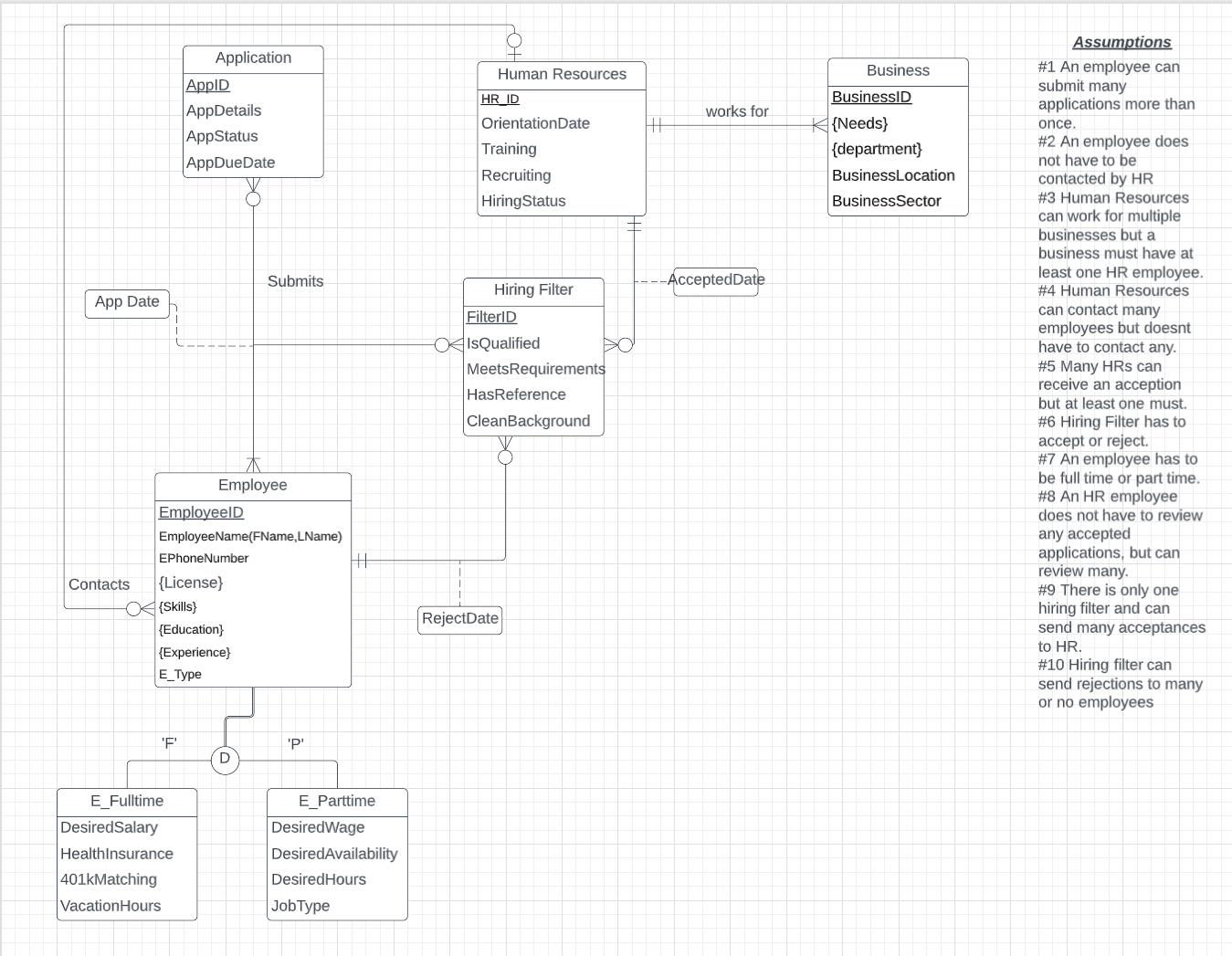
| Introduction: | 3 |
| --- | --- |
| EER Diagram: | 4 |
| Project Assumptions: | 4 - 5 |
| Logical Database Design: | 5 - 6 |
| Table Scripts: | 6 - 13 |
| Creation Scripts: | 6 - 9 |
| Insertion Scripts: | 9 - 13 |
| Table Explanation: | 13 |
| Queries: | 13 - 17 |
| Queries: | 13 - 16 |
| Query Explanation: | 16 -17 |
| Summary: | 17 |

**Introduction**

Our project was created to solve the problem with high job turnover rates in industries across the world. Businesses trying to find qualified candidates is becoming harder and harder, therefore we set out to create a new management system that would allow you to view candidates quickly. It is important to solve this problem, as businesses will continue to struggle with worker shortage, and qualified workers may continue to struggle to find credible job offers. Our database will look to bridge the gap between qualified candidates and finding the jobs that best fit their skill sets in companies desperate for workers. Creating a database that can filter through candidates and find the skills that the companies or departments require will greatly improve the efficiency of their hiring process.

Our database will utilize a hiring filter than will interact with the applicants skills, education, experience and possible licenses. This would eventually be able to process whether a candidate was the correct fit for positions within the company utilizing our database. Our database will also be utilized to store previous candidates, who may not have been qualified for current positions, but may be interested in future positions within the company. This will allow companies to focus solely on the interview aspect of the hiring process, and avoid looking for employees that meet specific criteria while digging through thousands of resumes.

**EER Diagram**



**Project Assumptions**

1. An employee can submit many applications more than once.
2. An employee does not have to be contacted by HR
3. Human Resources can work for multiple businesses but a business must have at least one HR employee.
4. Human Resources can contact many employees but doesn't have to contact any.
5. Many HRs can receive an acception but at least one must.
6. Hiring Filter has to accept or reject.
7. An employee has to be full time or part time.
8. An HR employee does not have to review any accepted applications, but can review many.
9. There is only one hiring filter and can send many acceptances to HR.
10. Hiring filter can send rejections to many or no employees

These assumptions were created to mimic how we believed a system like this could be implemented in a business. Firstly we identified the main components of a business during the hiring process such as: applications of employees, human resources, the business, and the employee. Next we added a hiring filter, which would allow us to filter the qualified applicants from the unqualified. We then created assumptions of how these entities would interact with each other in such a database.

**Logical Database Design**

**Employee:** EmployeeID, EmployeeFName, EmployeeLName, EPhoneNumber, E\_type, **HR\_ID** [HR\_ID is FK to HumanResources]

**License:** **EmployeeID**, License [EmployeeID is FK to Employee]

**Skills:** **EmployeeID**, Skills [EmployeeID is FK to Employee]

**Education:** **EmployeeID**, Education [EmployeeID is FK to Employee]

**Experience:** **EmployeeID**, Experience [EmployeeID is FK to Employee]

**Application:** AppID, AppDetails, AppStatus, AppDueDate

**HumanResources:** HR\_ID, OrientationDate, Training, Recruiting, HiringStatus

**Business:** BusinessID, BusinessName, BusinessLocation, BusinessSector, **HR\_ID** [ HR\_ID is FK to HumanResources]

**Needs:** **BusinessID**, Needs [BusinessID is FK to Business]

**Department: BusinessID**, Departments [BusinessID is FK to Business]

**HiringFilter:** FilterID, IsQualified, MeetsRequirements, HasReference, CleanBackground, **HR\_ID**, **EmployeeID**, AcceptDate, RejectDate [HR\_ID is FK to HumanResources, EmployeeID is FK to Employee]

**E\_Parttime:** **PEmployeeID**, DesiredWage, DesiredAvailability, DesiredHours, JobType [PEmployeeID is FK to Employee]

**E\_Fulltime:** **FEmployeeID**, DesiredSalary, HealthInsurance, 401kMatching, VacationHours [FEmployeeID is FK to Employee]

**Submission:** (1) **AppID, EmployeeID, FilterID**, AppDate

(2) SubmissionID, **AppID, EmployeeID, FilterID**, AppDate

[AppID is FK to Application, EmployeeID is FK to Employee, FilterID is FK to HiringFilter]

**Table Creation Scripts**

Create table Application(

AppID int NOT NULL,

AppDetails varchar(50) NOT NULL,

AppStatus varchar(10) NOT NULL,

AppDueDate timestamp(6),

EmployeeID int references Employee,

PRIMARY KEY (AppID));

Create table Employee(

EmployeeID int NOT NULL,

EFName varchar(40) NOT NULL,

ELName varchar(40) NOT NULL,

EPhoneNum bigint NOT NULL,

E\_Type char(2) NOT NULL,

HR\_ID int references HumanResources,

PRIMARY KEY (EmployeeID));

Create table HumanResources(

HR\_ID int NOT NULL,

OrientationDate timestamp(6),

Training varchar(10) NOT NULL,

Recruiting varchar(3) NOT NULL,

HiringStatus varchar(10),

PRIMARY KEY (HR\_ID));

Create table License(

License int NOT NULL,

EmployeeID int references Employee,

PRIMARY KEY (License));

Create table Business(

BusinessID int NOT NULL,

BusinessName varchar(50) NOT NULL,

BusinessLocation varchar(40) NOT NULL,

BusinessSector varchar(10) NOT NULL,

HR\_ID int references HumanResources,

PRIMARY KEY (BusinessID));

Create table Skills(

Skills varchar(20) NOT NULL,

EmployeeID int references Employee,

PRIMARY KEY (EmployeeID));

Create table Education(

Education varchar(10) NOT NULL,

EmployeeID int references Employee,

PRIMARY KEY (EmployeeID));

Create table Experience(

Experience varchar(10) NOT NULL,

EmployeeID int references Employee,

PRIMARY KEY (EmployeeID));

Create table Needs(

Needs varchar(40) NOT NULL,

EmployeeID int references Employee,

BusinessID int references Business,

PRIMARY KEY (BusinessID));

Create table Department(

Departments varchar(20) NOT NULL,

DepartmentID int NOT NULL,

BusinessID int references Business,

PRIMARY KEY (DepartmentID));

Create table HiringFilter(

FilterID int NOT NULL,

IsQualified varchar(10),

MeetsRequirements varchar(10),

HasReference varchar(10),

CleanBackground varchar(10),

AcceptDate timestamp(6),

RejectDate timestamp(6),

HR\_ID int references HumanResources,

EmployeeID int references Employee,

PRIMARY KEY (FilterID));

Create table E\_Parttime(

DesiredWage int(6) NOT NULL,

DesiredAvailability varchar(10) NOT NULL,

DesiredHours varchar(3) NOT NULL,

JobType varchar(10) NOT NULL,

EmployeeID int references Employee,

PRIMARY KEY (EmployeeID));

Create table E\_Fulltime(

DesiredSalary int(6) NOT NULL,

HealthInsurance varchar(10) NOT NULL,

401kMatching varchar(3) NOT NULL,

VacationHours varchar(3) NOT NULL,

EmployeeID int references Employee,

PRIMARY KEY (EmployeeID));

Create table Submission(

SubmissionID int NOT NULL,

AppDate timestamp(6),

AppID int references Application,

EmployeeID int references Employee,

FilterID int references HiringFilter,

PRIMARY KEY (SubmissionID));

**Table Insertion Scripts**

**Employee**

Insert into Employee values

(1, 'Joe', 'Diffie', 7751850001, 'FT', 1);

Insert into Employee values

(2, 'John', 'Smith', 8864473215, 'PT', 1);

Insert into Employee values

(3, 'Amy', 'Johnson', 7789935568, 'FT', 1);

Insert into Employee values

(4, 'Lee', 'Jones', 7759643346, 'PT', 1);

Insert into Employee values

(5, 'Amber', 'Lee', 4468936645, 'FT', 1);

**Application**

Insert into Application values

(1, 'LinkedIn', 'Approved', '2022-07-21',1);

Insert into Application values

(2, 'Website', 'Pending', '2022-05-13',2);

Insert into Application values

(3, 'Indeed', 'Denied', '2022-09-28',3);

Insert into Application values

(4, 'LinkedIn', 'Denied', '2022-02-03',4);

Insert into Application values

(5, 'Website', 'Pending', '2022-08-01',5);

**Human Resources**

Insert into HumanResources values

(1, '2022-05-12', 'Yes','Yes','Complete');

Insert into HumanResources values

(2, '2022-04-29', 'Yes','No','Interview');

Insert into HumanResources values

(3, '2022-12-17', 'No','Yes','Interview');

Insert into HumanResources values

(4, '2022-09-09', 'Yes','No','Complete');

Insert into HumanResources values

(5, '2022-10-16', 'No','No','Rejected');

**License**

Insert into License values

(1, 3);

Insert into License values

(2, 1);

Insert into License values

(3, 2);

Insert into License values

(4, 1);

Insert into License values

(5, 4);

**Skills**

Insert into Skills values

('Communication',1);

Insert into Skills values

('Leadership',2);

Insert into Skills values

('Teamwork',3);

Insert into Skills values

('Adaptability',4);

Insert into Skills values

('Critical Thinking',5);

**Education**

Insert into Education values

('Bachelors', 1);

Insert into Education values

('Associates',2);

Insert into Education values

('Bachelors',3);

Insert into Education values

('HighSchool',4);

Insert into Education values

('Masters',5);

**Experience**

Insert into Experience values

('MarketingM', 1);

Insert into Experience values

('SalesAssoc', 2);

Insert into Experience values

('ITDirector', 3);

Insert into Experience values

('Reception', 4);

Insert into Experience values

('CTO', 5);

**Business**

Insert into Business values

(1, 'Johns Finance', 'Ames', 'Financial', 1);

Insert into Business values

(2, 'Highway Tech', 'Seattle', 'IT', 2);

Insert into Business values

(3, 'Prime RealEstate', 'Ames', 'RealEstate', 3);

Insert into Business values

(4, 'Great Health', 'Juneau', 'HealthCare', 4);

Insert into Business values

(5, 'Yellowstone Retail', 'New York', 'Retail',5);

**Needs**

Insert into Needs values

('Sales Associate',1,1);

Insert into Needs values

('Marketing Representative',2,2);

Insert into Needs values

('Finance Specialist',3,3);

Insert into Needs values

('Assistant Manager of Operations',4,4);

Insert into Needs values

('Data Analyst',5,5);

**Part Time**

Insert into E\_Parttime values

('15', 'part', '20', 'Janitor', 2);

Insert into E\_Parttime values

('22', 'full', '40', 'IT Tech', 4);

**Full Time**

Insert into E\_Fulltime values

('80000', 'Yes', 'No', '168', 1);

Insert into E\_Fulltime values

('120000', 'Yes', 'Yes', '240', 3);

Insert into E\_Fulltime values

('150000', 'Yes', 'Yes', '240', 5);

**Department**

Insert into Department values

('Sales',1, 1);

Insert into Department values

('Marketing',2, 2);

Insert into Department values

('Operations',3,3);

Insert into Department values

('Finance', 4,4);

Insert into Department values

('Management',5,2);

Insert into Department values

('DiscoMan', 6,2);

Insert into Department values

('SignSpinner', 7,2);

Insert into Department values

('Cashier', 8,8);

**Hiring Filter**

Insert into HiringFilter values

(1, 'yes', 'no', 'yes', 'yes', '2022-06-05', NULL, 1, 1);

Insert into HiringFilter values

(2, 'yes', 'yes', 'no', 'yes', NULL, NULL, 2, 2);

Insert into HiringFilter values

(3, 'no', 'no', 'yes', 'yes', NULL, '2022-08-26', 3, 3);

Insert into HiringFilter values

(4, 'yes', 'yes', 'yes', 'no', NULL, '2022-02-01', 4, 4);

Insert into HiringFilter values

(5, 'yes', 'yes', 'yes', 'yes', NULL, NULL, 5, 5);

**Table Explanation**

These tables were created to show values that we thought would be useful in the queries we chose to run. Although this is mostly made up data that we have created, it can be useful to see how our queries would pull information for companies looking to use this database. The tables take on many different attributes to represent the many different functionalities that our database could serve to a business.

**Queries**

When creating queries we wanted to find information that an employer might use while working with our database, while still utilizing the greater uses of SQL that we learned later in our course. Although some of these queries may seem more basic than others, they still show the functionality of our database and how a business may utilize our functionality to search for qualified employees.

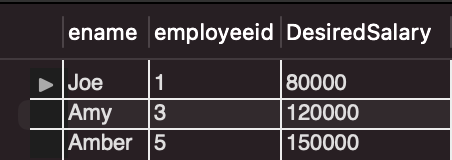
**1. What are the full time employees currently applying for a job, and what is their desired salary?**

**Query:**

Select employee.Efname as ename, employee.employeeid, e\_fulltime.DesiredSalary

From employee, e\_fulltime

Where employee.employeeid = e\_fulltime.employeeid;



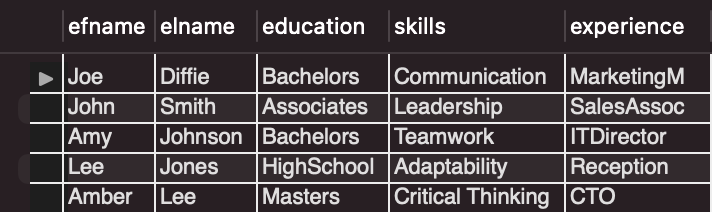
**2. What are the employees first names, last names, education level, skills, and experiences?**

**Query:**

select employee.efname, employee.elname, education.education, skills.skills, experience.experience

from employee, education, skills, experience

where employee.employeeid = education.employeeid and employee.employeeid = skills.employeeid and experience.employeeid = employee.employeeid;



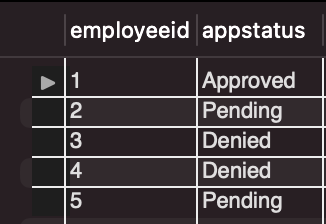
**3. What are the current statuses of hiring of different employees?**

**Query:**

Select employee.employeeid, application.appstatus

From employee, application

Where employee.employeeid = application.employeeid;

****

**4. What departments of the company are currently located in Seattle?**

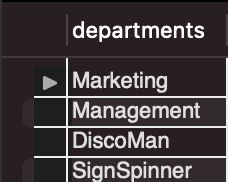
**Query:**

Select department.departments

From department, business

Where business.businessid = department.businessid

And business.businesslocation = 'seattle';



**5. What are the current parttime employees searching for jobs, and the businesses available in Ames?**

**Query:**

Select business.businessname as name

From business

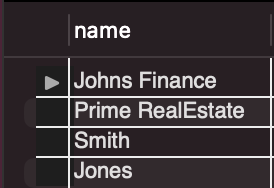
Where business.businesslocation = 'ames'

Union

Select employee.elname as name

From employee, e\_parttime

Where employee.employeeid = e\_parttime.employeeid;



**6. What the current desired salaries of full time applicants and their experience levels in descending order?**

**Query:**

Select e\_fulltime.desiredsalary as salary, experience.experience

From e\_fulltime, experience, employee

Where e\_fulltime.employeeid = employee.employeeid

And employee.employeeid = experience.employeeid

order by e\_fulltime.desiredsalary desc;



**Query Explanation**

**1. What are the full time employees currently applying for a job, and what is their desired salary?**

This query would be useful in businesses wanting to visualize current employees within the applicants that are looking for full time work at a certain level of salary. Our filter would then be able to filter out those that don’t meet the companies current budget, and possibly look deeper into other attributes of the employee which will be touched on in further queries

**2. What are the employees first names, last names, education level, skills, and experiences?**

This query would be useful in viewing all employees currently among your applicants and what each of them could possibly bring to the company in terms of their education levels, skills and experiences

**3. What are the current statuses of hiring of different employees?**

This query is useful for what our database was intended to do. Looking out how the hiring filter could approve or deny applicants based on their qualifications for jobs available in the company they are applying for.

**4. What departments of the company are currently located in Seattle?**

This query could be used to see what departments currently reside in our database for a specific company and allow them to see if they need to add more and get rid of certain departments.

**5. What are the current parttime employees searching for jobs, and the businesses available in Ames?**

This query was mainly focused on the idea of utilizing a union with our sql to create a query from our tables. We chose two tables that share the same city with the business and applicants located in that city. It could be slightly improved by not utilizing the same name for both business and employee and rather having them in rows instead.

**6. What the current desired salaries of full time applicants and their experience levels in descending order?**

This query would be useful for a business that is looking at the highest desired salary among applicants as those tend to be the more experienced. Therefore in this query we listed their experience level as well. This could show a company the higher level applicants that could be possibly added to their companies.

**Summary**

Although our database isn’t completely finished to our liking, it currently has the characteristics that would make up our database going forward. Our next step would be to apply the hiring filter into the process of hiring employees and utilizing HR to deny or accept employees based on the hiring filters recommendations. This as said previously, would greatly help the efficiency of a company during the hiring process, as to speed up the time it takes to filter through resumes. This database would be able to notify applicants of rejections, without the use of additional employees of a company. We believe that further expansion of this database would lead to something very valuable for many companies to utilize.