myTeams Database System

Joshua Hayes

ID: 922379312

GitHub username: jhayes23

Checkpoint #	Date Submitted
Checkpoint 3	03/14/2023
Checkpoint 2	02/28/2023
Checkpoint 1	02/21/2023

Table of Contents

1 Project Description

1.1 Project Overview

The goal of this relational database system is to solve issues in present day HR software as a service products. With this database system companies will be able to effectively manage current/future employees, departments, invoices, and track productivity metrics inhouse. Companies will be able to store important employee data, department information such as budgets, expenses, and track project metrics that are present in our competition's products. However, some key features this database shall implement are to allow hiring managers to store notes alongside the information of a potential hire. With this key feature, less time will be lost looking for previous notes. Another feature that this product shall implement is to allow employees to see how their currently stored degrees, experience, and credentials match against vacant position requirements in order to improve the chances of promoting within. This database system also supports project management features such as managing priorities, assigning teams, and extending due dates. This database would be useful for companies both large and small that are looking to effectively manage employee data, payroll, invoices, clients, and projects.

1.2 Use Cases

Use Case Title:	Previous interviewee
Actors:	Applicant (Justin), Hiring Manager (Philip)
Description:	This use case begins when Justin is interviewed for a position with Philip's company. Philip takes notes during his interview with Justin. Justin does not end up getting the position due to a more fitting candidate accepting the role. However, Justin applies for another position at Philip's company and Philip was able to retrieve the notes previously saved.

Use Case Title:	Different interviewer
Actors:	Applicant (Wallace), Hiring Manager (Philip), Team Manager (Annie)
Description:	This use case begins with Wallace being interviewed by Philip. Wallace knocks the interview out of the park and is pushed to the next round of interviews, which is with the Team's Lead, Annie. Curiously, Annie checks to see if any notes were saved from the hiring manager. This initiates the private notes case. If the notes are marked as private, then they shall not be able to be read by another person. Otherwise, the notes may be shared with specified individuals.

Use Case Title:	Access
Actors:	Team Manager (Annie)
Description:	This use case begins with the Team Manager Annie interviewing a second-round applicant for a position. Annie goes into the notes section and sees that there are previous notes added for the applicant. This initiates the authentication user case. If Annie is on the list of individuals with access, she will be able to read the notes previously entered. If she is not on the list. She will not be able to access the data.

Use Case Title:	Promoting within
Actors:	Ambitious employee (Maya)
Description:	This use case begins with Maya looking to move up in the company by looking at current vacant positions after earning a certificate. Maya updates her information and then can see additional roles as well as current roles that are more aligned with her current skillset. Maya can also opt in to sharing her

	desire to move up within the company so that her current resume can be shared automatically with managers in other departments looking to fill a position.

Use Case Title:	Split Decision
Actors:	Doubly Qualified Employee (Maya)
Description:	This use case begins with Maya opting in to sharing her desire to move up within the company. The company A has several new positions available and 5 teams in different departments. To improve this situation, candidates that currently work at the company can list up to 3 departments that they are interested in.

2 Database Requirements

2.1 Functional Requirements

Company

1. A company shall have at least one location.
2. A company location shall have many employees.
3. A company location shall have many departments.
4. A company shall have at many clients.
5. A company shall have many locations.
6. A company location shall have one payroll.
7. A location shall belong to one company.
8. A company shall be able to hire for many job positions.

Manager

1. A manager shall have at least one employee.
2. A manager shall be able to manage many projects.
3. A manager shall be able to leave many notes for a candidate.
4. Managers shall be able to access many candidate notes.

Employee

1. An employee shall belong to only one location.
2. An employee shall be able to belong in only one department.
3. An employee shall be able to express interest in many positions.
4. An employee shall have only one job title.
5. An employee shall be assigned to work on many projects.
6. An employee shall have many skills.
7. An employee shall have many certifications.
8. An employee shall have one manager.
9. An employee shall be able to work on many projects.
10. An employee shall have one payroll.
11. An employee shall have only one salary.
12. An employee shall have one name.
13. An employee shall have one company email.
14. An employee shall have one hire date.
15. A position may be assigned to many employees.

Department

1. A department shall be able to have many employees.
2. A department shall have many positions.
3. A department shall be able to have many projects.
4. A project shall be managed by many managers.
5. A project shall have one project name.
6. A department shall belong to many company locations.
7. A project shall belong to one department.
8. A project shall have only one priority.
9. A priority shall have many projects.
10. A job title shall belong to many employees.
11. A department shall have only one budget.

12. A department shall be able to submit many invoices.
13. A department shall be assigned many clients.
14. A client may be assigned to many departments.
15. A client shall be able to track many projects.

Budget

1. A budget shall belong to many departments.
2. A budget shall have many invoices.
3. A client shall be assigned to pay many invoices.
4. An invoice shall be assigned to at least one client.

Payroll

1. A payroll shall have many employees.
2. A payroll shall belong to many company locations.
3. An employee shall have many paystubs.
4. An employee shall have many W-2s.

Hiring

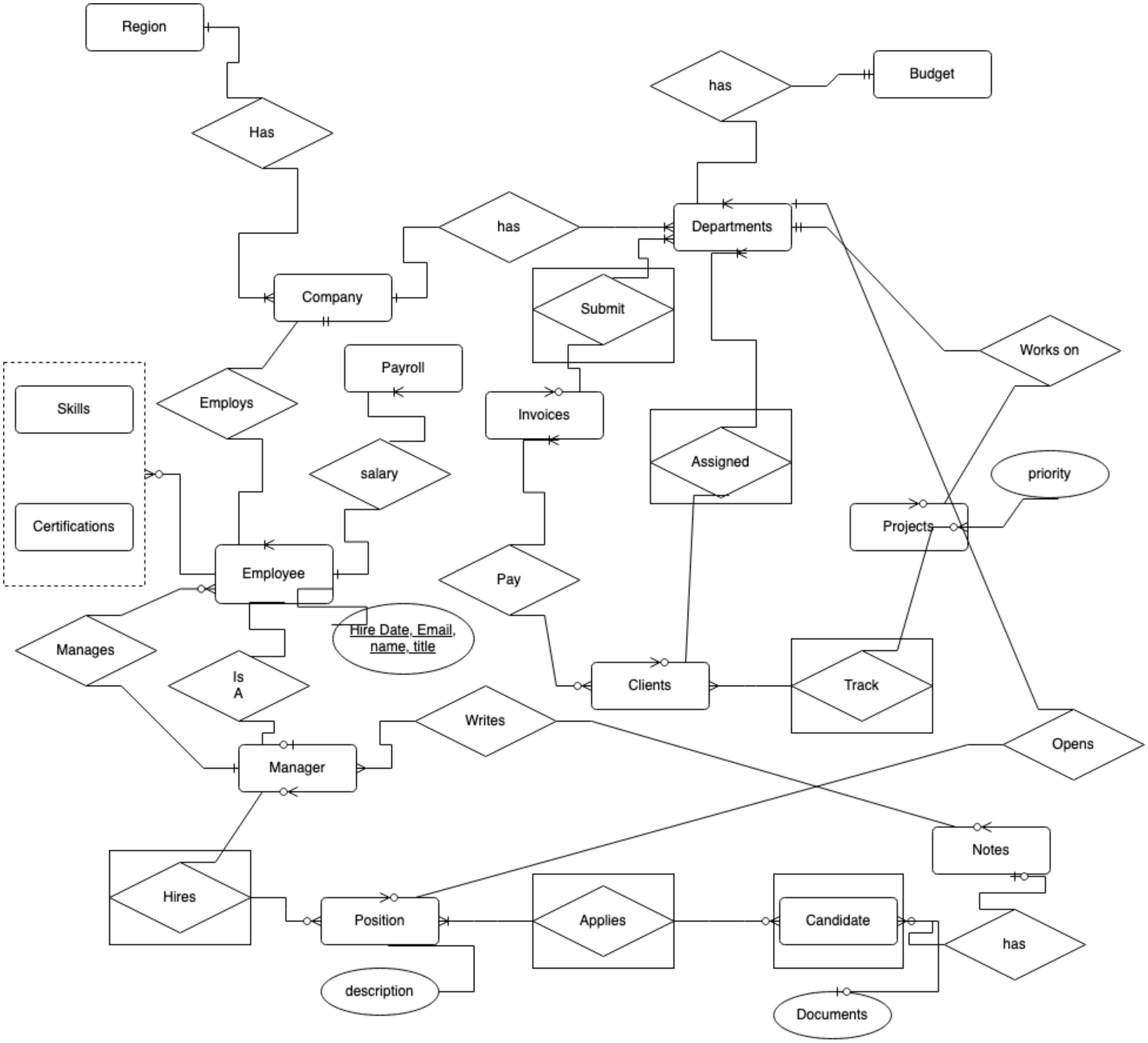
1. A candidate shall be able to apply to at least one position.
2. A candidate shall be interviewed by many managers.
3. A position shall have only one company location.
4. A candidate shall be able to be hired for one position.
5. A position shall be able to have many candidates.
6. A candidate shall be able to upload least one candidate document.
7. A candidate shall have multiple notes.
8. A candidate shall be able to upload many documents.
9. A job shall have one job description.
10. A job description shall have many job skill requirements.

2.2 Nonfunctional Requirements

Security

1. The database shall be automatically backed up every day.
2. All the values inserted into the database shall be consistent with the attribute's datatype and domain.

3 Entity Relationship Diagram (ERD)


Region
Has
has
Budget
has
Departments
Company
Submit
Works on
Payroll
Skills
Employs
Invoices
Assigned
salary
priority
Certifications
Projects
Employee
Pay
Hire Date, Email, name, title
Manages
Is A
Clients
Track
Writes
Opens
Manager
Notes
Hires
Position
Applies
Candidate
has
description
Documents