**Database Design Of HRMS System**

Tables to be created for the project:

List of tables:

| **Name** | **Description** |
| --- | --- |
| Department | Stores each departments unique id,reference for assigning employees to departments |
| positions | Defines available job positions,with title and descriptions |
| employees | Hold personal and employment details for each staff member |
| attendance | Track daily clock-in clock-out and attendance status per employee per date,ensure only one record per employee per day |
| Leave request | Record employee time-off requests,including type,date range and approval status,includes timestamps and reviewer info. |
| Pay roll | Store monthly payroll data per employee,including salary components and net amounts. |
| Salary structure | Defines compensation breakdown per position— basic pay,allowance, deductions—with effective date range.helps calculate payroll |
| Job postings | Capture open job listings,linked to position and department , along with status and posting dates.tracks who posted it |
| candidates | Store basic candidate information and application date for job postings. Used as a source for applications |
| applications | Maps candidates to job postings with application status and timestamp .track hiring process flow |
| projects | Defines projects with names,durations and status stages.serves as master list for assignments |
| Employee projects | Mapps employees to projects with assignment date,role,hour allocated, and active status.junction table linking people to projects |

*Department:*

**Purpose and usage:**Stores each department (e.g., Sales, HR). Used for assigning

employees and grouping other data**.**

| **Column** | **Type** |  | **Constraints** |
| --- | --- | --- | --- |
| dept\_id (PK) | INT | Auto-Increment | PK on dept\_id |
| name | VARCHAR | Not null | UNIQUE on name |

*Positions*

**Purpose and usage:** Defines job roles within the company (e.g., Manager,

Engineer).

| **Column** | **Type** | **Notes** | **Constraints** |
| --- | --- | --- | --- |
| position\_id(PK) | INT | Auto\_increment | PK on position\_id |
| title | VARCHAR | Not null |  |
| description | TEXT |  |  |

*Employees:*

**Purpose and usage:**Stores employee personal and organizational data, roles,

reporting structure, and access control.

**Column**

| **Column** | **Type** | **Type** | **Constraints** |
| --- | --- | --- | --- |
| emp\_id(PK) | INT | Auto increment | PK on emp\_id |
| emp\_code | VARCHAR |  |  |
| First name | VARCHAR | Not null |  |
| Last name | VARCHAR |  |  |
| Email | VARCHAR | UNIQUE | UNIQUE on email |
| Phone | VARCHAR | UNIQUE |  |
| DOB | DATE |  |  |
| address | VARCHAR |  |  |
| gender | ENUM | eg:male,female,other |  |
| marital\_status | ENUM | eg:married,unmarried,devorced |  |
| Emergency number | VARCHAR |  |  |
| Hire date | DATE |  |  |
| Dept id(FK) | INT |  | FK on dept\_id |
| Position id(FK) | INT |  | FK on position\_id |
| Manager id(FK) | INT |  | FK on manager\_id |
| role | ENUM | eg:manager,developer,tester | Limited to predefined roles |

*Attendance:*

**Purpose and usage:**Records daily attendance of employees, including time stamps

and leave statuses.

| **Column** | **Type** | **Notes** | **constraints** |
| --- | --- | --- | --- |
| attendance\_id (PK) | INT | Auto-increment | PK on attendance\_id |
| emp\_id (FK) | INT | Not null | FK on emp\_id |
| Date | DATE | Not null | Part of composite unique constraints |
| Clock in | TIME |  |  |
| Clock out | TIME |  |  |
| Status | ENUM | Eg:present,absent,on leave,half day,holy day |  |

*Leave request:*

**Purpose and usage:**Manages leave applications, their statuses, who reviewed and

when.

| **Column** | **type** | **Notes** | **Constraints** |
| --- | --- | --- | --- |
| leave\_id (PK) | INT | Auto-increment | PK on leave\_id |
| emp\_id (FK) | INT | Not null | FK on emp\_id |
| Leave type | VARCHAR | Not null |  |
| Start date | DATE | Not null |  |
| End date | DATE | Not null |  |
| Status | ENUM | eg:approved,rejected,pending |  |
| Requested on | TIMESTAMP | Not null |  |
| Reviewed by(FK) | INT |  | FK on reviewed\_by |
| Reviewed on | TIMESTAMP |  |  |

*Pay roll:*

**Purpose and usage:**Contains payroll details per cycle: salary computation,

deductions, and pay status.

| **Column** | **Type** | **Notes** | **Constraints** |
| --- | --- | --- | --- |
| payroll\_id (PK) | INT | Auto-increment | PK on payroll\_id |
| emp\_id (FK) | INT | Not null | FK on emp\_id |
| Cycle start | DATE | Not null |  |
| Cycle end | DATE | Not null |  |
| Basic salary | DECIMAL | Not null |  |
| Gross salary | DECIMAL | Not null |  |
| Deductions | DECIMAL | Not null |  |
| Net salary | DECIMAL | Not null |  |
| Status | ENUM | eg:draft,approved,paid |  |
| Created by(FK) | INT | Not null | FK on created\_by |
| Created on | TIMESTAMP | Not null |  |

*Salary structure:*

**Purpose and usage:**Defines pay scale per position, with components and validity

periods for historical or future changes.

| **Column** | **Type** | **Notes** | **Constraints** |
| --- | --- | --- | --- |
| struct\_id (PK) | INT | Auto-increment | PK on struct\_id |
| position\_id (FK) | INT | Not null | FK on position\_id |
| Basic | DECIMAL |  |  |
| Hra | DECIMAL |  |  |
| Da | DECIMAL |  |  |
| Deductions | JSON |  |  |
| Allowance | JSON |  |  |
| Effective from | DATE | Not null |  |
| Effective to | DATE | Not null |  |

*Job postings:*

**Purpose and usage:**Tracks hiring: open jobs, their status, who posted them, and

relevant dates.

| **Column** | **Type** | **Notes** | **Constraints** |
| --- | --- | --- | --- |
| job\_id (PK) | INT | Auto-increment | PK on job\_id |
| position\_id (FK) | INT | Not null | FK on position\_id |
| Dept\_id (FK) | INT | Not null | FK on dept\_id |
| Posted by(FK) | INT | Not null | FK on posted\_by |
| Title | VARCHAR | Not null |  |
| Description | TEXT |  |  |
| Status | ENUM | eg:open,closed,filed |  |
| Posted on | DATE | Not null |  |
| Closed on | DATE |  |  |

*Candidates:*

**Purpose and usage:**Stores applicant information for recruitment; contact details

and resume references.

| **Column** | **Type** | **Notes** | **Constraints** |
| --- | --- | --- | --- |
| Candidate id (PK) | INT | Auto-increment | PK on candidate\_id |
| Name | VARCHAR |  |  |
| Email | VARCHAR |  |  |
| Phone | VARCHAR |  |  |
| Resume link | VARCHAR |  |  |
| Applied on | DATE |  |  |

*Applications:*

**Purpose and usage:**Surveys candidate progress for job postings:

from application to hire or rejection.

| **Column** | **Type** | **Notes** | **Constraints** |
| --- | --- | --- | --- |
| App id(PK) | INT | Auto-increment | PK on app\_id |
| Job id(FK) | INT | Not null | FK on job\_id |
| Candidate id(FK) | INT | Not null | FK on candidate\_id |
| Status | ENUM | eg:applied,shortlisted,interviewed,hired,rejected |  |
| Updated on | TIMESTAMP | Not null |  |

*Projects:*

**Purpose and usage:**Stores each project’s unique ID, name, timeline, and

status—serving as the central registry of company initiatives.

| **Column** | **Type** | **Notes** | **Constraints** |
| --- | --- | --- | --- |
| Project\_id (PK) | INT | Auto-increment | PK on project\_id |
| name | VARCHAR(50) | Not null ,UNIQUE |  |
| description | TEXT |  |  |
| start\_date | DATE |  |  |
| end\_date | DATE |  |  |
| status | ENUM | Eg:planned,active,completed,on hold, cancelled |  |

*Employee projects:*

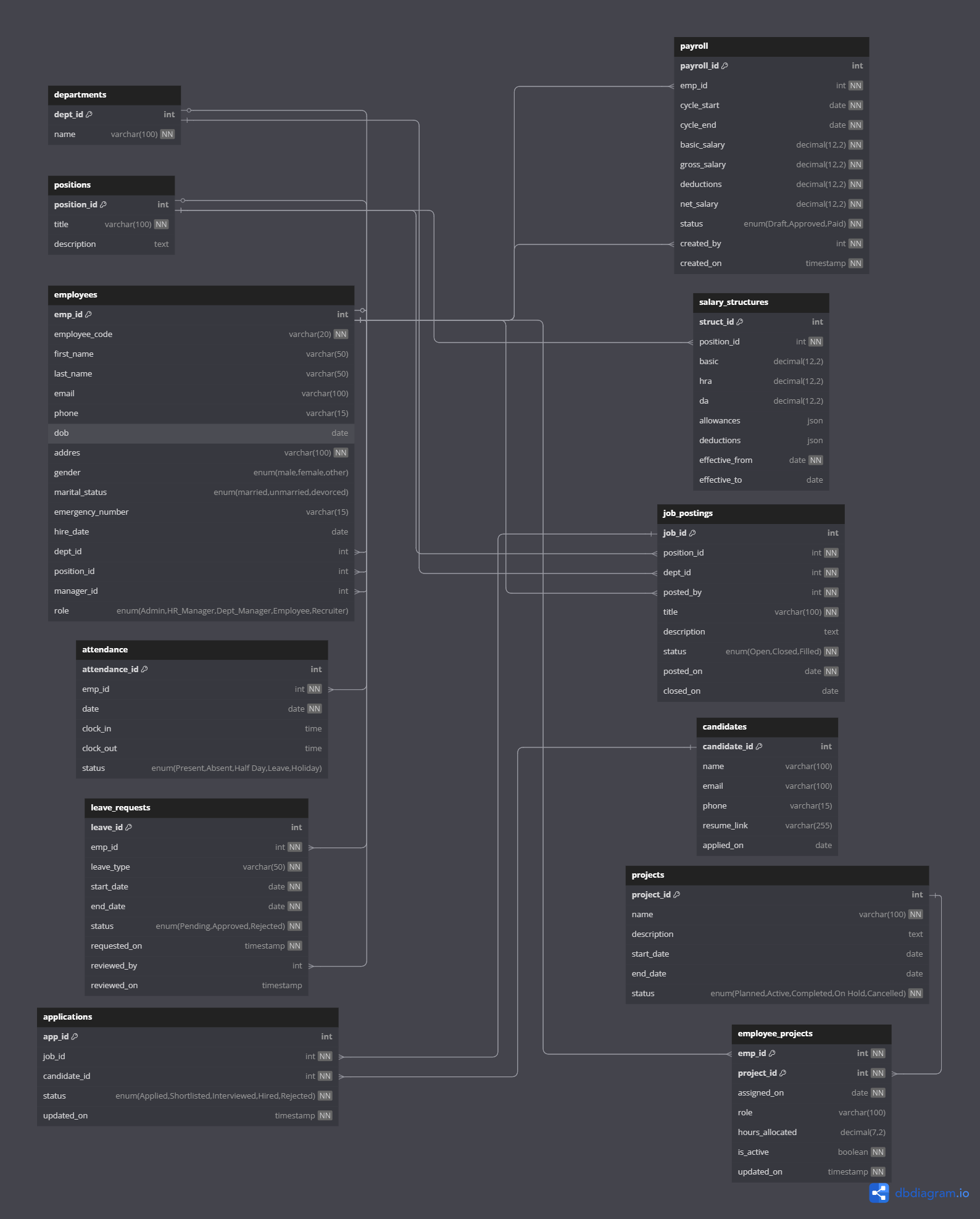
**Purpose and usage:**A junction (associative) table linking employees and

projects. It records who works on what, when, in what role, and for how many

hours—enabling many‐to‐many associations with rich assignment details.

| **Column** | **Type** | **Notes** | **Constraints** |
| --- | --- | --- | --- |
| emp\_id(FK) | INT | Not null | FK on emp\_id |
| project\_id(FK) | INT | Not null | FK on project\_id |
| assigned\_on | DATE | Not null |  |
| role | VARCHAR(100) |  |  |
| hour\_allocated | DECIMAL |  |  |
| is\_active | BOOLEAN | Not null |  |
| updated\_on | TIMESTAMP | Not null |  |
| **PK on (emp\_id,project\_id)** |  |  | Composite PK for unique assignment per employee–project pair |

|  |
| --- |
|  |

***ER*** *diagram:* **