1. User Table

Backend Columns:

- **user_id (PK, Auto-generated)**

- **role (FK → Role Table)**

- **email (Unique)**

- **password**

- **is_active (Boolean)**

- **is_verified (Boolean)**

- **create_date**

- **update_date**

Functionalities:

Admin:

- **View all users (joined with Employee table)**

    o **Show: role, email, first_name, last_name, designation, department, reporting_person**

- **Delete user (and related employee record)**

- **Edit: reporting_person, designation, department**

- **Assign role to unverified users**

---

2. Role Table

Backend Columns:

- **id (PK)**

- **name (e.g., SuperAdmin, Admin, Employee)**

---

3. Employee Table

Backend Columns:

- **employee_id (PK, Auto-generated)**

- **user_id (FK → User Table)**

- **first_name**

- **last_name**

- **gender**

- **dob**

- **location**

- **designation**

- **department**

- **reporting_person (Employee ID)**

- **skills**

- **education**

Functionalities:

Admin:

- **Create employee record after user is verified**

- **View all employee details (joined with User Table)**

Employee:

- **Update personal info: first_name, last_name, gender, dob, location, skills, education**

---

4. Skills Table

Backend Columns:

- **id (PK)**

- **employee_id (FK)**

- **name**

Functionalities:

- **Admin & Employee: View skills**

- **Employee: Add, Edit, Delete own skills**

---

5. Education Table

Backend Columns:

- **id (PK)**

- **employee_id (FK)**

- **name**

Functionalities:

- **Admin & Employee: View education**

- **Employee: Add, Edit, Delete own education**

---

6. Events Table

Frontend Fields:

- **title, eventType, enableDate, expireDate, file**

Backend Columns:

- **id (PK)**

- **title**

- **event_type**

- **enable_date (LocalDate)**

- **expire_date (LocalDate)**

- **created_by (fetched from current employee using user_id)**

- **status (Pending / Approved / Rejected)**

- **file_name**

- **file_path**

Functionalities:

Admin:

- **View all events**

- **Approve / Reject / Ignore (Cannot delete or update content)**

Employee:

- **Create event**

- **View own events**

- **Edit, Delete own events**

---

7. Leave Table

Frontend Fields:

- **leave_type, leave_from, leave_to, duration, cover_person, reason, file**

Backend Columns:

- **id (PK)**

- **username (fetched using user_id)**

- **leave_type**

- **leave_from**

- **leave_to**

- **duration**

- **cover_person (selected from same department)**

- **report_to (auto-filled from Employee table)**

- **reason**

- **leave_status**

- **file_name**

- **file_path**

Functionalities:

Admin:

- **View all leaves**

- **Approve / Reject / Update status**

- **Cannot delete**

Employee:

- **Apply for leave**

- **View, Edit, Delete own leave**

---

8. Refer Candidate Table

Frontend Fields:

- **vacancy, applicant_name, applicant_email, message, resume**

Backend Columns:

- **id (PK)**

- **vacancy**

- **applicant_name**

- **applicant_email**

- **message**

- **resume_file_name**

- **resume_file_path**

- **referred_by (fetched using user_id)**

- **status (Read / Unread)**

Functionalities:

Admin:

- **View all referrals**

- **Update status (Read/Unread)**

Employee:

- **Create, View, Edit, Delete own referrals**

---

9. Vacancy Table

Backend Columns:

- **id (PK)**

- **name (Job title)**

Functionalities:

Admin:

- **Add, Update, Delete**

Employee:

- **View vacancies**

---

10. Claim Table

Frontend Fields:

- **claim_type, description, file**

Backend Columns:

- **id (PK)**

- **claim_type**

- **description**

- **file_name**

- **file_path**

- **status (Pending / Approved / Rejected)**

- **requested_by (fetched using user_id)**

Functionalities:

Admin:

- **View all claims**

- **Approve / Reject / Update status**

- **Cannot delete**

Employee:

- **View, Create, Edit, Delete own claims**

---

11. Department Table

Backend Columns:

- **department_id (PK)**

- **name**

Departments are predefined and used as dropdowns in the frontend

---

12. Team Table

Backend Columns:

- **team_id (PK)**

- **name**

- **department_id (FK → Department)**

Functionalities:

Admin:

- **Create teams for departments**

- **One department can have many teams**

- **Each employee can belong to only one team**

Employee:

- **View own team**

---

13. Team_Members Table (Mapping Table)

Backend Columns:

- **id (PK)**

- **team_id (FK)**

- **employee_id (FK)**

Functionalities:

- **Admin & Employee: View team members**

---

14. Project Table

Backend Columns:

- **project_id (PK)**
- **name**
- **description**
- **team_id (FK)**
- **start_date**
- **end_date**

Functionalities:

Admin:

- **Create, Assign team, Update, Delete**

Employee:

- **View all projects**
- **View their own project (based on assigned team)**

---

15. Timesheet Table

Frontend Fields:

- **date, working_hours, activities**

Backend Columns:

- **timesheet_id (PK)**
- **employee_id (FK)**
- **project_id (auto-fetched using team assignment)**

- **date**

- **working_hours**

- **activities**

Functionalities:

Admin:

- **View, Approve, Reject timesheets**

Employee:

- **Submit timesheet with:**

  o **Auto-filled project**

  o **Selected date**

  o **Working hours**

  o **Description of activities**

---

16. Letter Generation (No Database Table)

Frontend Form:

- **Fields vary by letter type**

Backend:

- **Accepts form data**

- **Generates letter and sends back to frontend (e.g., PDF)**

---

17. Predefined Lists (Frontend)

These are constant values used in dropdowns and validations:

- **Departments**

- **Leave Types**

- **Claim Types**

- **Status Values (Pending, Approved, Rejected, Read, Unread, etc.)**