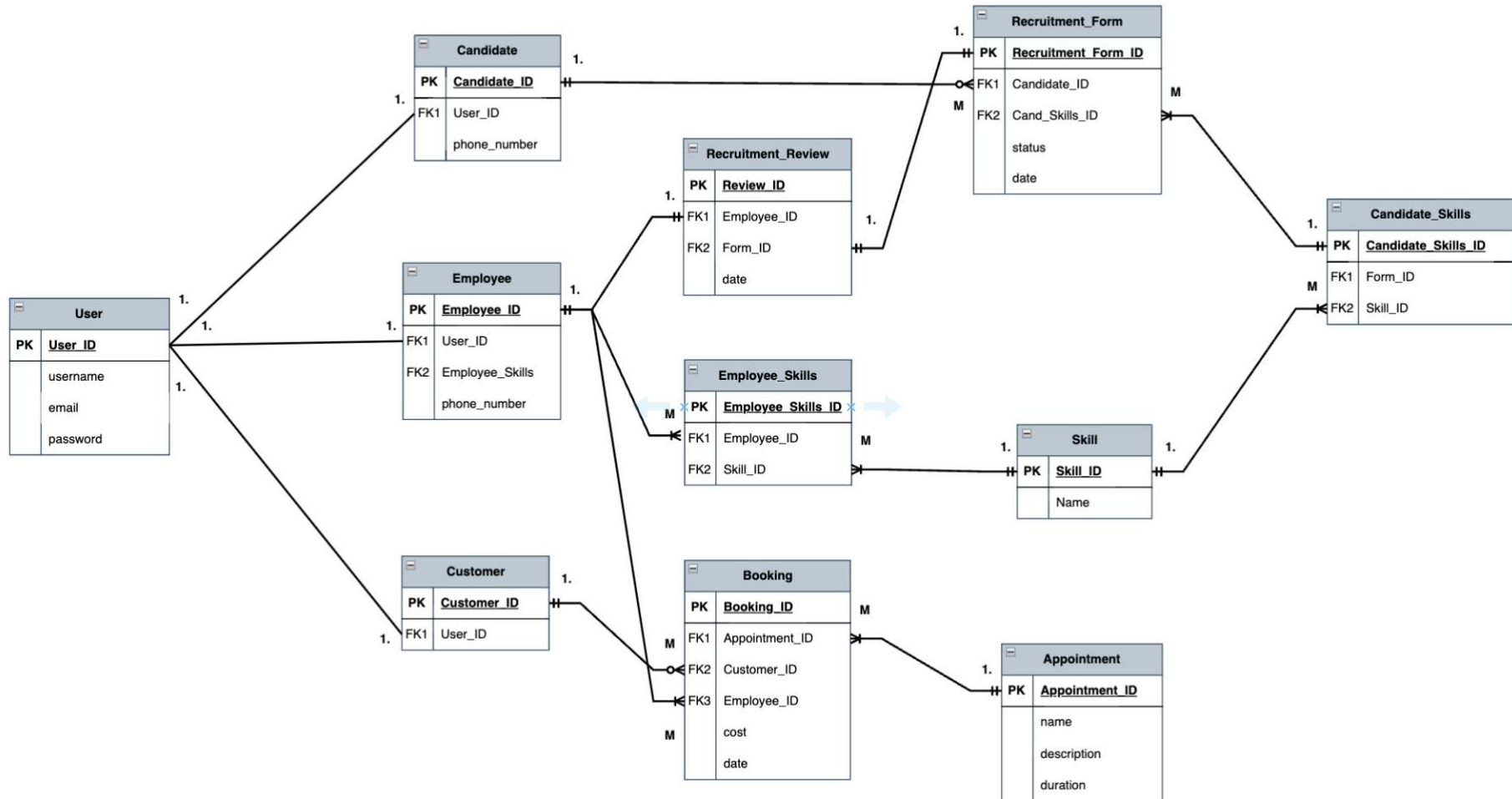Diagram:

Explanation:

User, Customer, Candidate and Employee:

- **User** is the base class for the entities **Customer**, **Employee**, and **Candidate**.
  Each has a **one-to-one** relationship with **User**, as a single user account can only represent one role.
- **User** contains the following attributes:
    - username
    - email_address
    - password

- **Customer**, **Employee**, and **Candidate** each contain a **foreign key to User**.
    - **Customer**: No additional attributes.
    - **Employee**: Also includes a phone_number and is linked to **Employee_Skills**.
    - **Candidate**: Includes a phone_number.

Booking and Appointments

- **Customer** has a **one-to-many** relationship with **Booking**, as a customer can make multiple bookings with different employees for various appointments.
- **Employee** also has a **one-to-many** relationship with **Booking**, since they can be assigned to multiple bookings.
- **Appointment** has a **many-to-one** relationship with **Booking**, as many bookings can be for the same type of appointment.

- **Booking** contains the following attributes:

- o Foreign key to **Customer**
- o Foreign key to **Employee**
- o Foreign key to **Appointment**
- o date
- o cost
- o status

- **Appointment** contains:
  - o name
  - o description
  - o duration

Recruitment Process

- **Candidate** has a **one-to-many** relationship with **Recruitment_Form,** as they can submit multiple forms.
- **Recruitment_Form** has a **many-to-one** relationship with **Candidate_Skills**, as multiple forms can share the same set of skills.
- **Recruitment_Form** contains:
  - o Foreign key to **Candidate**
  - o Foreign key to **Candidate_Skills** (multiple entries; at least 3 recommended)
  - o status
  - o date

- **Candidate_Skills** serves as a join table for **Recruitment_Form** and **Skill**, and contains:
  - o Foreign key to **Recruitment_Form**
  - o Foreign key to **Skill**

Employee_Skills and Recruitment_Review

- **Skill** has a **one-to-many** relationship with both **Employee_Skills** and **Candidate_Skills**, as one skill can be associated with multiple employees and candidates.
- **Employee** has a **one-to-many** relationship with **Employee_Skills**, as an employee can have multiple skills.
- **Employee_Skills** contains:
    - Foreign key to **Employee**
    - Foreign key to **Skill**

- There is a **one-to-one** relationship between **Employee** and the accepted **Recruitment_Form**, represented via a **Recruitment_Review** entity.
  This ensures only one recruitment form is accepted per candidate, resulting in employment.