# AttachMe Database

## 1️⃣ Database Name:

`attachme_db`

---

## 2️⃣ Core Tables & Their Purpose

### 1. Users Table (Authentication & Role Management)

**Stores login credentials and general user details.**

- `id` (Primary Key, Auto Increment)
- `full_name` (VARCHAR, 255) – User's full name
- `email` (VARCHAR, 255, UNIQUE) – Used for login
- `password_hash` (VARCHAR, 255) – Encrypted password
- `role` (ENUM: `student`, `company`, `admin`) – Defines access level
- `created_at` (TIMESTAMP) – Date of registration
- `updated_at` (TIMESTAMP) – Last profile update

🛡️ **Security Considerations:**
✔ Passwords will be **hashed** before storing.
✔ Unique constraint on `email` prevents duplicate accounts.

🔗 **Relationships:**

- One-to-One with **Students**, **Companies**, and **Admins** tables.

---

### 2. Students Table (Student-Specific Data)

**Stores additional student-related details.**

- `id` (Primary Key, Auto Increment)
- `user_id` (Foreign Key, references Users)

- `registration_number` (VARCHAR, 50, UNIQUE)
- `course` (VARCHAR, 255)
- `year_of_study` (INT)
- `level` (ENUM: `Certificate`, `Diploma`, `Degree`, `Masters`)
- `resume_link` (VARCHAR, 255) – Uploaded CV
- `profile_picture` (VARCHAR, 255, NULLABLE)

🔗 **Relationships:**

- One-to-One with **Users** (each student is linked to a user account).
- One-to-Many with **Applications** (students submit multiple applications).

---

## 3. Companies Table (Company-Specific Data)

**Stores company profile details.**

- `id` (Primary Key, Auto Increment)
- `user_id` (Foreign Key, references Users)
- `company_name` (VARCHAR, 255)
- `industry` (VARCHAR, 255)
- `location` (VARCHAR, 255) – Selected from the **47 Kenyan Counties**
- `contact_email` (VARCHAR, 255)
- `contact_phone` (VARCHAR, 20, NULLABLE)
- `created_at` (TIMESTAMP)
- `updated_at` (TIMESTAMP)

🔗 **Relationships:**

- One-to-One with **Users**.
- One-to-Many with **Opportunities** (a company can post multiple attachment openings).

---

## 4. Opportunities Table (Attachment Postings)

**Contains attachment/internship listings from companies.**

- `id` (Primary Key, Auto Increment)
- `company_id` (Foreign Key, references Companies)
- `title` (VARCHAR, 255)
- `description` (TEXT)

- `requirements` (TEXT)
- `available_slots` (INT)
- `application_deadline` (DATE)
- `status` (ENUM: `open`, `closed`)
- `created_at` (TIMESTAMP)
- `updated_at` (TIMESTAMP)

🔗 **Relationships:**

- One-to-Many with **Applications** (Students apply to multiple opportunities).

---

## 5. Applications Table (Tracking Student Applications)

**Manages student applications for attachment opportunities.**

- `id` (Primary Key, Auto Increment)
- `student_id` (Foreign Key, references Students)
- `opportunity_id` (Foreign Key, references Opportunities)
- `status` (ENUM: `pending`, `accepted`, `rejected`)
- `submitted_at` (TIMESTAMP)
- `reviewed_at` (TIMESTAMP, NULLABLE)
- `feedback` (TEXT, NULLABLE)

🔗 **Relationships:**



- One-to-Many with **Students** (A student applies to many opportunities).
- One-to-Many with **Opportunities** (An opportunity receives multiple applications).

---

## 6. Messages Table (Student-Company Communication)

**Handles messaging between students and companies.**

- `id` (Primary Key, Auto Increment)
- `sender_id` (Foreign Key, references Users)
- `receiver_id` (Foreign Key, references Users)
- `message` (TEXT)
- `status` (ENUM: `sent`, `delivered`, `read`)
- `sent_at` (TIMESTAMP)

🔗 **Relationships:**

- Many-to-Many between **Users** (Messages exchanged between students and company representatives).

---

## 7. Notifications Table (User Alerts)

**Manages user notifications (e.g., application updates).**

- `id` (Primary Key, Auto Increment)
- `user_id` (Foreign Key, references Users)
- `message` (TEXT)
- `is_read` (BOOLEAN DEFAULT FALSE)
- `created_at` (TIMESTAMP)

🔗 **Relationships:**

- One-to-Many with **Users** (A user can have multiple notifications).

---

## 8. Admins Table (Admin-Specific Roles)

**Contains system administrators with extra privileges.**



- `id` (Primary Key, Auto Increment)
- `user_id` (Foreign Key, references Users)
- `admin_role` (VARCHAR, 255) – Super Admin, Content Moderator
- `created_at` (TIMESTAMP)

🔗 **Relationships:**

- One-to-One with **Users** (Each admin must have a user account).

---

## 9. Analytics Table (Platform Insights)

**Stores statistical data for analytics and reporting.**

- `id` (Primary Key, Auto Increment)
- `metric` (VARCHAR, 255) – Metric being tracked

- `value` (INT) – The value of the metric
- `recorded_at` (TIMESTAMP)

🔗 **Relationships:**

- Standalone table for platform-wide analytics.

---

## 3️⃣ Security & Performance Considerations

✅ **Indexes:**

- Indexes will be created on **email**, **registration_number**, and **company_name** for fast lookups.

✅ **Foreign Keys & Constraints:**

- **ON DELETE CASCADE** for related tables (e.g., if a user is deleted, their applications are removed).

✅ **Scalability:**

- **Partitioning** large tables (e.g., logs, messages) for faster queries.

✅ **Data Integrity:**

- Ensuring **normalized database structure (3NF)** to avoid redundancy.



---

## 4️⃣ Relationships Overview

🔹 **One-to-One:**

- `Users` → `Students`
- `Users` → `Companies`
- `Users` → `Admins`

🔹 **One-to-Many:**

- `Companies` → `Opportunities`
- `Students` → `Applications`
- `Opportunities` → `Applications`

- `Users` → `Messages`
- `Users` → `Notifications`

🔹 **Many-to-Many (via Messaging Table):**

- `Users` ↔ `Messages` (Students & Companies communicate)