Here's a structured breakdown for the updated database choice and design:

Database Technology Justification

Chosen Database: PostgreSQL (Relational)

A relational database like PostgreSQL is better suited for the structure and requirements of this Lead Management System. Here's why:

- **Data Consistency**: Leads, inquiries, and scores benefit from structured relationships and consistency, which a relational model supports well.
- **Complex Queries**: We'll need complex queries to calculate lead scores, filter, and sort leads based on engagement. PostgreSQL offers robust querying capabilities that make it ideal for these operations.
- **Transaction Support**: Managing inquiries and scoring involves multiple changes to the database. PostgreSQL's ACID compliance ensures data integrity during these transactions.
- **Scalability for Stretch Features**: PostgreSQL can handle large amounts of data, such as detailed analytics on leads, without sacrificing query speed or reliability. This is crucial as we add new stretch features for tracking interactions and setting automated follow-ups.

Updated Database Structure

1. Tables for MVP

1.1 Users Table

- **Description**: Stores user account information.
- **Columns**:
    - `user_id` (Primary Key, UUID): Unique identifier for each user.
    - `username` (VARCHAR): User's name.
    - `email` (VARCHAR): User's email address.
    - `password_hash` (TEXT): Hashed password.
    - `created_at` (TIMESTAMP): Account creation date.

1.2 Leads Table

- **Description**: Manages individual leads, tracking their score and interactions.
- **Columns**:
  - `lead_id` (Primary Key, UUID): Unique identifier for each lead.
  - `user_id` (Foreign Key to Users): Links the lead to a specific user.
  - `name` (VARCHAR): Lead's name.
  - `contact_info` (VARCHAR): Contact details.
  - `score` (INTEGER): Engagement score of the lead.
  - `created_at` (TIMESTAMP): Date the lead was created.

1.3 Inquiries Table

- **Description**: Stores inquiries made by leads, which contribute to the lead score.
- **Columns**:
  - `inquiry_id` (Primary Key, UUID): Unique identifier for each inquiry.
  - `lead_id` (Foreign Key to Leads): Links the inquiry to a lead.
  - `message` (TEXT): Inquiry content.
  - `created_at` (TIMESTAMP): Date of the inquiry.

2. Additional Tables for Stretch Feature

2.1 LeadAnalytics Table

- **Description**: Tracks detailed lead interactions for analytics purposes, such as page views or tool usage.
- **Columns**:
  - `analytic_id` (Primary Key, UUID): Unique identifier for each analytic event.
  - `lead_id` (Foreign Key to Leads): Links the analytic event to a specific lead.
  - `event_type` (VARCHAR): Type of event, e.g., "page_view," "tool_interaction."
  - `event_value` (VARCHAR): Detail of the event, e.g., page URL or tool used.
  - `event_date` (TIMESTAMP): Date and time of the event.

2.2 FollowUpReminders Table

- **Description**: Stores follow-up reminders for leads based on engagement and custom criteria.
- **Columns**:
    - `reminder_id` (Primary Key, UUID): Unique identifier for each reminder.
    - `lead_id` (Foreign Key to Leads): Lead associated with the reminder.
    - `reminder_date` (TIMESTAMP): Scheduled follow-up date.
    - `reminder_message` (TEXT): Custom message or notes for the reminder.
    - `created_at` (TIMESTAMP): Date the reminder was set.

Usage of Data Structures

- **Users Table**: This table controls user access to the platform.
- **Leads Table**: Central table for managing all leads, with `score` tracking their engagement.
- **Inquiries Table**: Each inquiry submitted by a lead is recorded here and contributes to the `score` in the Leads table.
- **LeadAnalytics Table** (Stretch Feature): Stores detailed information on lead activities, allowing realtors to assess engagement depth.
- **FollowUpReminders Table** (Stretch Feature): Allows users to set reminders based on engagement criteria, supporting more proactive lead management.

Interaction Flow

1. **Log in / Sign up**: Users authenticate via the Users table.
2. **Dashboard**: User retrieves leads and their scores, interacting with the Leads table.
3. **Add Inquiry**: Each inquiry a lead makes is recorded in the Inquiries table and updates the lead's score.
4. **Lead Analytics (Stretch Feature)**: Users view detailed activity in LeadAnalytics for deeper insight.
5. **Follow-Up Reminders (Stretch Feature)**: Users set reminders in the FollowUpReminders table to maintain contact with high-potential leads.

This relational database design allows for smooth integration of core and stretch features, giving the application flexibility for future development and maintaining data integrity and organization.