Matiks: Full-Stack Engineer Intern Assignment

Assignment Title

Scalable Leaderboard System with Search & Tie-Aware Ranking

Note: This internship is in-person (Bangalore) and requires immediate availability. Interns demonstrating strong performance will be eligible for a PPO.
Deadline for submission: 19th Jan EOD

Overview

At **Matiks**, we build systems that require **correctness, scale, and clarity**.
This assignment evaluates your ability to design a **robust leaderboard system** that works correctly under scale and edge cases.

You will implement a **Leaderboard Platform** that supports:

- Large-scale user data
- Accurate ranking with ties
- Efficient leaderboard updates
- Fast user search and rank lookup

Objective

Build a **Leaderboard System** that:

- Manages **10,000+ users (seeded data)** at same time it should have capacity to handle millions of users
- **Calculates correct ranks**
- Simulates **random score updates**
- Allows **searching any user** and retrieving their rank

Technical Stack

Mandatory

- **Frontend:** React Native (Expo)
- **Backend:** Golang

Core Requirements

Ranking Rules (Critical Requirement)

- All the users will have rating based on their skills anywhere between 100 to 5000 (min rating: 100, max rating 5000)
- Rankings are based on rating.
- Users with the same rating must have the same rank

Frontend (React Native)

The app should include:

Leaderboard Screen

- Fetch and Displays the leaderboard on the screen.
- Shows: Rank, Username, Rating

User Search

- Input to search by username
- Displays: Global Rank, Username, Rating

   Think of scenario when thousands of users are playing the game every minute and their rating are updated. Whenever i hit search, i should be able to get live global rank of that user.

Performance Expectations

- Must handle **10k+ users smoothly**
- Search should feel **instant**
- Ranking updates should not block requests
- UI must remain responsive

Deliverables

- GitHub Repository
- A video Link

- Use expo (for mobile and web app) and deploy the web version on vercel/netlify and share the link

Example:-

If i search for a user with text rahul i should get all the users with there username and global rank like

Global Rank	Username	Rating
200	rahul	4600
800	rahul_burman	3900
800	rahul_mathur	3900
9876	rahul_kumar	1234