```
generator client {
  provider = "prisma-client-js"
}

datasource db {
  provider = "postgresql"
  url = env("DATABASE_URL")
}

model User {
  id String @id @default(cuid())
  name String?
  email String @unique
  emailVerified DateTime?
  image String?
  role Role @default(USER)
  passwordHash String?
  createdAt DateTime @default(now())
  updatedAt DateTime @updatedAt

  profile Profile?
  leaderboard LeaderboardEntry[]
  messages ChatMessage[]
}

model Profile {
  id String @id @default(cuid())
  user User @relation(fields: [userId], references: [id])
  userId String @unique
  bio String?
  country String?
}

model Game {
  id String @id @default(cuid())
  title String
  slug String @unique
  description String?
  coverUrl String?
  category String?
  createdAt DateTime @default(now())
  updatedAt DateTime @updatedAt
  leaderboard LeaderboardEntry[]
  matches Match[]
}

model LeaderboardEntry {
  id String @id @default(cuid())
  user User @relation(fields: [userId], references: [id])
  userId String
  game Game @relation(fields: [gameId], references: [id])
  gameId String
  score Int @default(0)
  createdAt DateTime @default(now())

  @@index([gameId, score])
}

model ChatMessage {
  id String @id @default(cuid())
  room String
  user User @relation(fields: [userId], references: [id])
  userId String
  text String
  createdAt DateTime @default(now())
}

model Match {
  id String @id @default(cuid())
  game Game @relation(fields: [gameId], references: [id])
  gameId String
  hostId String
  status MatchStatus @default(OPEN)
  participants String[] @default([])
  createdAt DateTime @default(now())
  updatedAt DateTime @updatedAt
}

enum Role {
  USER
  ADMIN
}

enum MatchStatus {
  OPEN
  ONGOING
  FINISHED
}
```