```
import { NextApiRequest } from "next";
import { NextApiResponseServerIO } from "@/types/next";
import { prisma } from "@/lib/prisma";
export default function handler(reg: NextApiReguest, res: NextApiResponseServerIO) {
 if (!res.socket.server.io) {
  console.log("Starting Socket.IO server...");
  const { Server } = require("socket.io");
  const io = new Server(res.socket.server, {
   path: "/api/socketio"
  });
  res.socket.server.io = io;
  io.on("connection", (socket) => {
   console.log("User connected", socket.id);
   socket.on("chatMessage", async ({ userId, room, text }) => {
    if (!text.trim()) return;
    const msg = await prisma.chatMessage.create({
     data: { userId, room, text }
    });
    io.to(room).emit("chatMessage", msg);
   });
   socket.on("joinRoom", (room) => {
    socket.join(room);
   });
   socket.on("disconnect", () => {
    console.log("User disconnected", socket.id);
   });
  });
 }
 res.end();
```