

```

module.exports = async function handler(req, res) {
  if (req.method !== "POST") {
    return res.status(405).json({ error: "Method not allowed" });
  }

  var context = req.body.context;

  if (!context) {
    return res.status(400).json({ error: "No context provided" });
  }

  var systemPrompt = "You are a senior personalisation strategist with deep exper

  var requestBody = {
    model: "llama3-70b-8192",
    temperature: 0.7,
    max_tokens: 1500,
    messages: [
      { role: "system", content: systemPrompt },
      { role: "user", content: "Audit this business and return JSON:\n\n" + conte
    ]
  };

  try {
    var response = await fetch("https://api.groq.com/openai/v1/chat/completions",
      method: "POST",
      headers: {
        "Content-Type": "application/json",
        "Authorization": "Bearer " + process.env.GROQ_API_KEY
      },
      body: JSON.stringify(requestBody)
    );
  }

  var data = await response.json();

  if (!response.ok) {
    return res.status(500).json({ error: data.error ? data.error.message : "Gro
  }

  var raw = data.choices[0].message.content;
  var clean = raw.replace(/\` `json/g, "").replace(/\` `/g, "").trim();
  var audit = JSON.parse(clean);

  return res.status(200).json(audit);
}

```

```
    } catch (err) {
      console.error(err);
      return res.status(500).json({ error: "Failed to generate audit" });
    }
};
```