**🔧 Tech & Architecture**

**1. How does GhostLAN handle LAN matchmaking without the internet?**

**Answer:**  
We use **UDP broadcast** or **mDNS/Zeroconf** to auto-discover peers on the same LAN. Once players are found, we use WebRTC or direct socket communication to establish a local game lobby and voice chat. No online server is involved.

**2. How does the offline anti-cheat system actually work?**

**Answer:**  
We use **rule-based detection** and lightweight telemetry analytics to flag suspicious behavior — like unnatural headshot accuracy, impossible reaction times, or movement through walls. It's designed for local execution, not full machine learning inference, to stay lightweight and fast.

**3. What game engines or titles does GhostLAN support?**

**Answer (honest MVP scope):**  
Right now, we’re working with **custom sandbox setups** and compatible open-source games for proof-of-concept.  
In the future, we plan to build **game-specific plugins or SDKs** for popular multiplayer titles — or provide an overlay layer if native integration isn’t possible.

**4. What happens if one of the devices goes offline mid-game?**

**Answer:**  
Each client logs locally, so even if one drops, it can sync stats or match state once it reconnects. We also plan to introduce an optional **local mini-server** (like a Raspberry Pi or host peer) to improve fault tolerance.

**5. Why did you go with FastAPI and not something like Flask or Node?**

**Answer:**  
FastAPI gives us **great performance**, **async support**, and easy-to-document APIs — all important for building local-first, fast-response systems. It also plays well with Python-based AI modules for cheat detection.

**🌐 Offline & Syncing Model**

**6. How do you handle syncing once internet returns?**

**Answer:**  
We store data in a **local sync queue** using SQLite or JSON. When internet access is restored, a background service pushes data to cloud dashboards, Discord webhooks, or external APIs — optionally with user confirmation or automatic push.

**7. Can two people playing the same game see each other’s stats live?**

**Answer:**  
Yes — if connected on the same LAN, GhostLAN can enable stat sharing or observer mode across peers via local sockets. Otherwise, data stays on-device and syncs post-match.

**🧠 Vision & Differentiation**

**8. What makes GhostLAN different from existing tools like Discord or Faceit?**

**Answer:**  
Those tools **require a stable internet connection** — GhostLAN doesn’t. We’re building for environments where **those tools break entirely**. Our **offline-first design**, **local voice**, **local anti-cheat**, and **LAN-only play** are what set us apart.

**9. Who is your target customer or first real user?**

**Answer:**

* **College fest organizers**
* **LAN cafés in tier 2/3 cities**
* **Mobile eSports communities**  
  They all face poor internet and can benefit from GhostLAN's plug-and-play tournament tools.

**10. What’s your long-term vision for GhostLAN?**

**Answer:**  
We want to build **LAN-first eSports kits** — complete with:

* Voice,
* Anti-cheat,
* Dashboards,
* Plug-and-play Raspberry Pi servers,  
  Then expand into **peer-hosted tournaments** and eventually **decentralized eSports platforms**.

**💰 Feasibility & Monetization**

**11. How will you monetize this?**

**Answer:**  
Three main paths:

1. **B2B licenses** for colleges, gaming cafés, and event organizers.
2. **Premium analytics dashboards** and branded sync overlays.
3. **Modular tournament kits** with offline hardware bundles or server installs.

**12. What are the risks or limitations right now?**

**Answer:**

* Game compatibility: we need to either integrate with or wrap around games manually.
* P2P networking across LANs can be tricky due to firewalls or OS permissions.
* Anti-cheat at edge-level is powerful but can’t yet replace server-grade systems.

But these are solvable with incremental iterations and platform-specific support.