Rapport Technique — Architecture Multi-APK IA Locale et Serveur

Objectif : Développer une architecture unifiée multi-APK avec traitement local et serveur, garantissant performance, sécurité et confidentialité.

1. Objectif Global

Créer une architecture unifiée capable de servir plusieurs applications mobiles (APK) via un modèle d’inférence texte multilingue unique, tout en préservant l’isolation, la performance et la confidentialité utilisateur.

2. Choix du Modèle Central (LLM)

Modèle : OrcaMini-3B GGUF q4 (ou StableLM 3B quantifié) - Exécution sur CPU avec quantization - Moins de 8 Go RAM nécessaires - Lazy load et déchargement automatique

3. Modules Spécialisés

ASR : whisper.cpp (modèles small/base) — CPU, streaming segmenté.
Vision/Pose : MediaPipe / MoveNet / OpenCV — 10–15 FPS, CPU optimisé.

4. Architecture Logique

[APK clients] --(Tor .onion / API key)--> [Tor hidden service -> NGINX] --> [FastAPI Router] - Authentification, classification d’intent, handlers LLM/ASR/Vision, vector store par app.

5. Isolation et Sécurité

- API key + X-App-ID unique par app - Prompt système spécifique à chaque app - Vector store séparé (Qdrant/FAISS) - Intent filter pour bloquer hors-sujet

6. Gestion Ressources

- Quantization 4-bit GGUF - Max 1–2 requêtes simultanées - Streaming ASR, pose à 10–15 FPS - Déchargement automatique des modèles

7. Intent Classifier

Deux couches : règles lexicales + tiny classifier (DistilBERT 100–300M quantifié). Seuil < 0.6 → envoi serveur ; sinon réponse locale.

8. Vector Store / RAG

FAISS ou Qdrant local — collection par App-ID. Isolation documentaire totale et gestion HDD locale.

9. Intégration Tor

Deux hidden services : 1. API publique (.onion) 2. Admin privé (SSH/VPN) Torrc minimal avec ports redirigés vers FastAPI et SSH.

10. Traitement On-Device

ASR, intent classifier, pose/vision et cache exécutés localement. Serveur sollicité uniquement pour requêtes complexes ou faible confiance.

11. Modèles Recommandés

ASR : Whisper.cpp (tiny/base) Intent classifier : TF-Lite ou DistilBERT quantifié LLM local : Llama.cpp 1–2B Vision : MediaPipe / MobileNetV2

12. Politique de Fallback

- ASR confidence < 0.7 → serveur - Intent < 0.6 → serveur - Out-of-scope → redirection locale

13. Flux Opérationnel

Capture → ASR local → Intent check → local ou serveur via Tor → cache résultat. Pipeline optimisé pour latence et économie de ressources.

14. Confidentialité

Transmission uniquement de features (angles, embeddings). Pas d’audio/image brute. Chiffrement par clé éphémère.

15. Monitoring & Tuning

Collecte locale des métriques (ASR, intent, batterie). Synchronisation batchée pour ajuster seuils.

16. Check-List Implémentation

 Télécharger OrcaMini-3B GGUF q4 Installer whisper.cpp Mettre en place FastAPI + NGINX + Tor API keys par App-ID Intent filter + vector store isolé

17. Conclusion

Architecture hybride on-device/cloud minimale assurant performance, sécurité, confidentialité et scalabilité IA sur CPU (≤12 Go RAM).