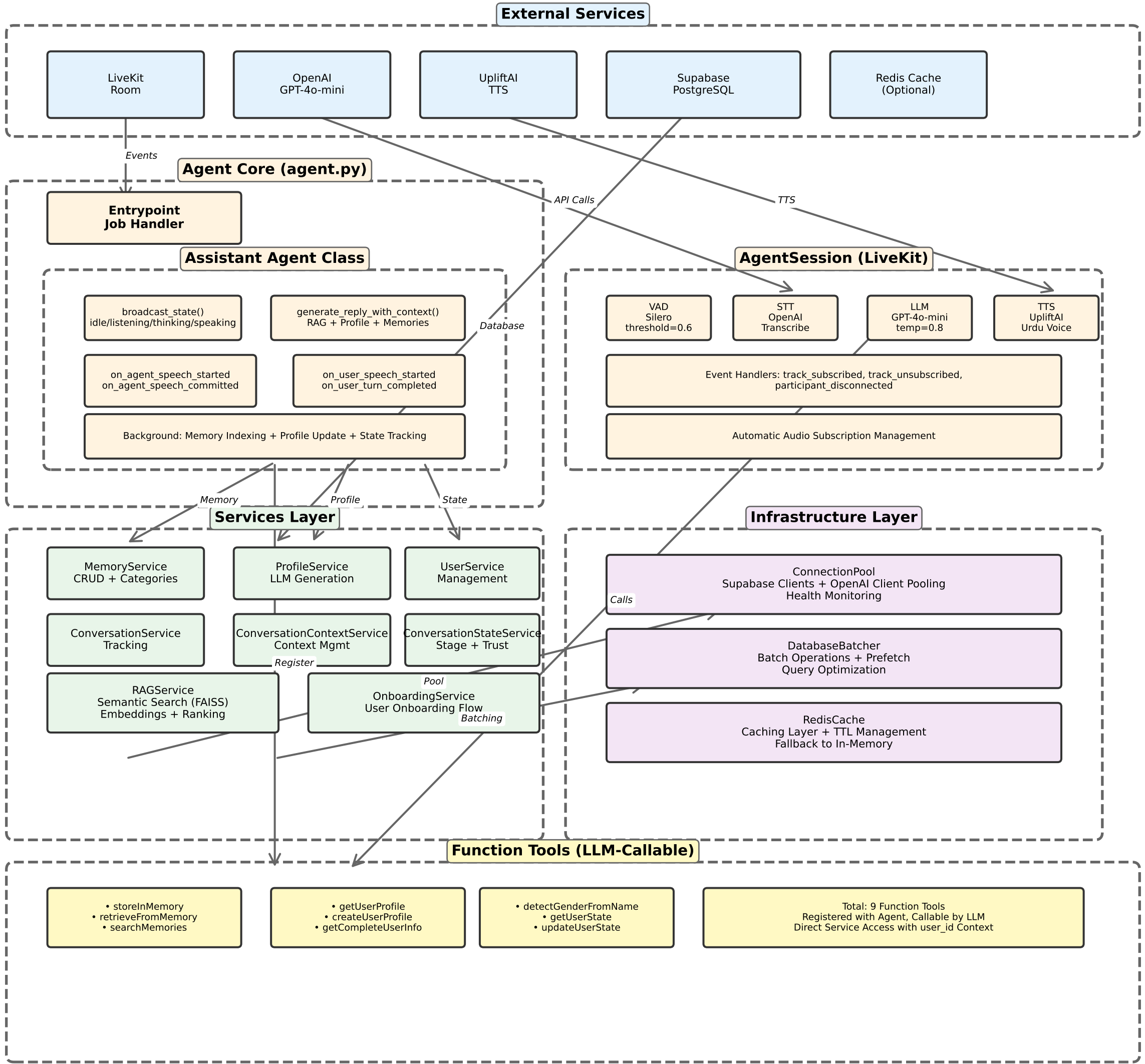


Companion Agent Architecture

LiveKit Agents Framework + Services Layer



Key Data Flows

- 1 User Speech → VAD Detection → STT → LLM + Context (Profile + RAG) → TTS → Audio Output
- 2 LLM Function Calls → Tools → Services → Database (Background, Zero Latency)
- 3 State Broadcasting → listening/thinking/speaking → Frontend UI Updates
- 4 Memory Storage → RAG Indexing → Semantic Search → Context Enrichment
- 5 Event-Driven: track_subscribed, participant_disconnected → Graceful Handling

Architecture Highlights

- Event-Driven Architecture (LiveKit Best Practices)
- Non-Blocking Background Processing (RAG + Profile + State)
- Connection Pooling + Database Batching + Redis Caching
- Optimized VAD (threshold=0.6, min_speech=0.15s)
- Comprehensive Error Handling + Resource Cleanup
- Zero-Latency User Experience (Background Tasks)

Generated for Companion Agent (agent.py) - LiveKit Agents Framework