

VeriFact: Final Demo Presentation

Real-Time Fact Verification for Online Meetings

Team T2503

May 2026

CS491 Senior Design Project · Bilkent University

Background: Problem & Solution

The Problem

- **Misinformation:** Claims in live meetings are hard to verify.
- **Manual Effort:** Fact-checking is slow and disruptive.
- **Trust Gap:** Unverified statements reduce productivity.

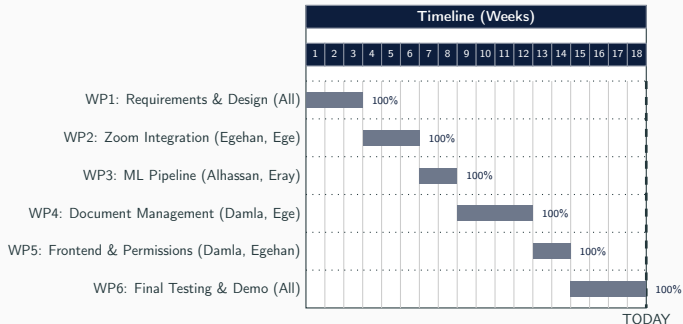
The VeriFact Solution

- **Real-Time Verification:** Claims verified in < 3s.
- **Voice Agent:** Instant RAG-based Q&A via "Hey Agent".
- **Privacy First:** Can be run locally, with only S3 and Azure STT as external dependencies.

Pathway to Realization

- **Initial Design:** Envisioned a larger pipeline with different models for claim detection, claim stance classification and reasoning of the stance. Planned to use local STT.
- **Evolution:** Simplified the architecture around one LLM (Qwen 3) and prompted it for structured responses. Switched to Azure STT for better accuracy and performance.
- **The Result:** A fully containerized, production-ready system that integrates seamlessly with Zoom's SDK.

Accomplishments & Contributions

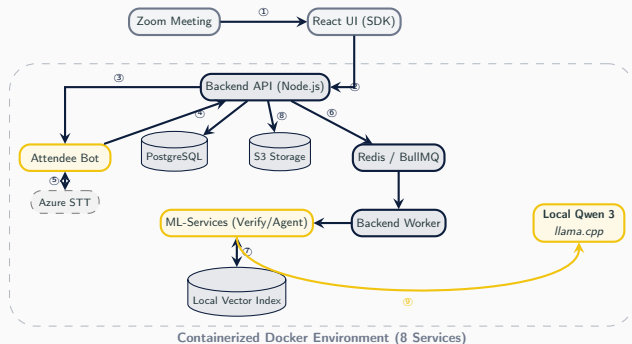


Team Contributions

Backend/Zoom: Eray, Alhassan **ML/RAG:** Ege, Egehan **Frontend/Docs:** Damla

Figure 1: Finalized Project Timeline and Work Package Completion

Architecture: Services & Data Flow



- ① User Interactions / SDK Events
- ② WebSockets (Real-time Updates)
- ③ Bot Invitation Control
- ④ Webhooks (Transcription Chunks)
- ⑤ Voice-Triggered Agent Query

- ⑥ Async ML Task Queuing
- ⑦ Local Vector Search (Server-side)
- ⑧ S3 File Store (Public Upload/Download)
- ⑨ Local LLM Inference (Agentic Chat)

Figure 2: System Component Interactions and Distributed Data Flow

Retrospective & Lessons Learned

What Went Well

- **Orchestration:** 8 containers managed flawlessly via Docker.
- **Latency:** Achieved $< 3s$ E2E verification time.
- **Permission system:** Implemented the permission system within organizations for different use cases.
- **Privacy:** 100% local inference successful, so that organization data is safe.

Challenges & Lessons

- **Challenge:** STT performance was the biggest hurdle.
- **Solution:** Solved via Azure STT, after many CUDA and Whisper STT attempts failed.
- **Modularity:** Performance optimization in one module should be extensively tested to ensure no breaking changes occur in other modules.

Demo Flow: VeriFact in Action

1. **Activation:** Bot joins the Zoom meeting.
2. **Real-Time Detection:** System identifies factual claims.
3. **RAG Verification:** Claims verified against private documents.
4. **Hey Agent:** Voice-triggered interactive Q&A.
5. **Persistence:** Fact-check history saved for later review.

Thank You!

Questions?