

Voice-Based Email & Messaging Assistant

Hands-Free, Secure, Context-Aware Communication



Intern Name
Armaan Samir Jena



Mentor Name
Santhiya Krishnasamy



Organization
Infosys Springboard

Project Overview

Voice-Based Email & Messaging Assistant

- **Project Title:** Voice-Based Email & Messaging Assistant
- **Core Goal:** Enable secure, voice-driven email and messaging operations with contextual awareness.
- **Target Users:** Individuals seeking hands-free, accessible communication (especially mobility-impaired users).
- **Key Challenge:** Balancing automation, privacy, and platform restrictions.
- **Final Architecture:** Hybrid Model integrating Gmail API, Telegram Bot, and assistive WhatsApp Web.

Problem Statement

Challenges in Voice-Based Communication



Typing Limitations

Typing-based communication is time-consuming and not suitable for hands-free use cases.



Accessibility Issues

Users with mobility or vision impairments face barriers in using traditional email/messaging apps.



Voice Assistant Gaps

Existing assistants like Alexa or Google Assistant cannot perform secure, context-aware email or chat automation.



Accidental Actions

Voice misinterpretation can lead to incorrect or unintended message sending, posing security and privacy risks.

Why Deep Background Analysis Was Required

Understanding Platform Constraints



Platform Diversity

Each messaging and email platform behaves differently, requiring deep research on APIs and automation capabilities.



Policy & Privacy Restrictions

Strict policies prevent background access or automation in mobile and desktop environments without explicit consent.



Automation Risks

Unrestricted automation can trigger spam or policy violations if not carefully sandboxed and user-confirmed.



Expectation

Mentor required a policy-safe, technically feasible, demonstrable prototype architecture.

Scenario 1: Mobile Apps (Android / iOS)

Feasibility and Policy Review



WhatsApp Mobile

- Programmatic access: X
- Inbox reading: X
- Verdict: Not feasible



Gmail Mobile

- API inside app: X
- Background automation: X
- Verdict: Not feasible



Telegram Mobile

- Personal chat access: X
- Verdict: Not feasible



Combined Conclusion

- Automation: X
- Policy safety: X
- Recommendation: Reject mobile automation route.

Scenario 2: Desktop Apps

Feasibility and Limitations



WhatsApp Desktop

- API access: X
- UI automation reliability: !
- Verdict: Not stable or policy-safe



Gmail Desktop

- Limited programmatic control: X
- Background scripts restricted
- Verdict: Not feasible



Telegram Desktop

- Automation through GUI possible but unstable
- Policy conflicts likely
- Verdict: Reject



Combined Conclusion

- Automation: X
- Policy safety: !
- Recommendation: Avoid desktop-level automation due to inconsistency and risk.

Scenario 3: Web Apps (Browser Only)

Evaluating Browser-Based Automation

WhatsApp Web

- Visible content access:
- Drafting ability:
- Background automation:

Gmail Web

- Visible inbox:
- Drafting emails:
- Auto-send:

Telegram Web

- Chat access:
- Drafting possible:
- Background execution:

Combined Conclusion

- Assistive features supported but no full automation. Feasible for controlled, front-end interaction only.

Scenario 4: Browser Extensions

Extending Web App Capabilities

WhatsApp Web + Extension

- DOM access:
- Drafting:
- Auto-send limitation:

Gmail Web + Extension

- DOM access:
- Drafting:
- Auto-send:

Telegram Web + Extension

- DOM control:
- Drafting:
- Automation limited to foreground

Combined Verdict

- Medium automation possible | Safe with browser sandboxing | Suitable as a secondary automation layer

Scenario 5: Own Platform Using APIs

API-Driven Secure Automation

Gmail API

- Full inbox access: 
- Full automation: 
- Secure via OAuth

Telegram API

- Bot-only chats: 
- Partial automation: 

WhatsApp API

- Business-only: 
- Costly & limited access

Combined Verdict

- Clean architecture, secure data handling, partial platform coverage – best suited for integration in hybrid model.

Final Feasibility Comparison

Comparing All Approaches

- **Mobile Apps:** Feasible: | Policy Safe: | Automation Level: Low
- **Desktop Apps:** Feasible: | Policy Safe: | Automation Level: Low
- **Web Apps:** Feasible: | Policy Safe: | Automation Level: Medium
- **Browser Extensions:** Feasible: | Policy Safe: | Automation Level: Medium
- **APIs / Hybrid Model:** Feasible: | Policy Safe: | Automation Level: High – **Final Recommended Architecture**

Final Architecture: Hybrid Model

Optimized for Feasibility and Policy Safety

Gmail (Primary)

Full automation | Voice read/send | OAuth-based authentication ensures secure access.

Telegram (Secondary)

Bot-based messaging | Voice replies supported | Secure via Bot API.

WhatsApp (Assistive)

Message summarization and draft replies | No auto-send for compliance.

Overall Integration

Combines direct APIs and assistive web automation under one unified voice interface.

Why Hybrid Model Is Correct

Technical and Strategic Validation

- **Technical Feasibility:**  Proven API and controlled automation paths ensure reliability.
- **Policy Compliance:**  Aligns with platform terms by using official APIs and limited assistive automation.
- **Mentor Milestone Fit:**  Matches expected project deliverables and evaluation requirements.
- **Demonstrable Automation:**  Real-time voice-driven actions across email and messaging.
- **Scalable Design:**  Modular architecture allows future service integrations.

Milestone Plan Overview

Timeline of Project Execution



Week 1

Research & Architecture – Completed foundational design and feasibility study.



Week 2

Milestone 1 Presentation – Delivered architecture and feasibility analysis.



Weeks 3–4

Milestone 2 Delivery – Full integration, testing, and voice interaction demonstration.

Milestone 1 Deliverables

Initial Implementation Outcomes

- **Voice Pipeline:** ASR (Automatic Speech Recognition) + TTS (Text-to-Speech) integrated for command execution.
- **User Authentication:** OAuth-based login system ensuring secure access to Gmail APIs.
- **Architecture Diagram:** System flow diagram prepared to show hybrid integration model.
- **Feasibility Analysis:** Comparative study of platforms for automation safety and compliance.
- **Live Demo Readiness:** Prototype environment established for mentor review.

Milestone 2 Deliverables (Preview)

Upcoming Integration and Demonstration



Gmail API Integration

Implement full email read/send capabilities via voice commands.



Voice-Based Email Operations

Enable hands-free email reading, drafting, and sending through speech interaction.



Two-Phase Confirmation

Add verification layer to prevent unintended email/message actions.



Context-Aware Actions

Interpret previous conversation and infer intent for message drafting.

Future Scope

Potential Enhancements Beyond Internship



Browser Extension

Develop a native extension for seamless, policy-safe automation in browsers.



Advanced Voice Biometrics

Integrate speaker verification for personalized and secure interactions.



Multi-Language Support

Add multilingual ASR/TTS capabilities to enhance accessibility.



Enterprise Workflows

Expand to enterprise-grade collaboration systems (e.g., Slack, Teams).

Final Conclusion

Project Summary and Readiness



Final Architecture

Hybrid Model integrating Gmail, Telegram, and WhatsApp under a unified voice interface.



Core Reason

Maximizes automation while maintaining full policy compliance and data security.



Mentor Alignment

100% aligned with expected milestones and evaluation criteria.



Evaluation Readiness

Prototype stable, secure, and ready for final demonstration.

Thank You

Questions & Discussion



Project

Voice-Based Email & Messaging
Assistant – A Secure,
Context-Aware, Hybrid Voice
Automation System



Internship Platform
Infosys Springboard



Next Steps

Open for Q&A, feedback, and
mentor discussion.