Phase 1 Input Agent Final Documentation

Executive Summary

The Input Agent is a critical component of the system responsible for receiving, processing and normalizing multimodal user inputs (text, voice, files) into a unified format that can be consumed by the Planner Agent. It operates as the primary interface between users and the agentic system.

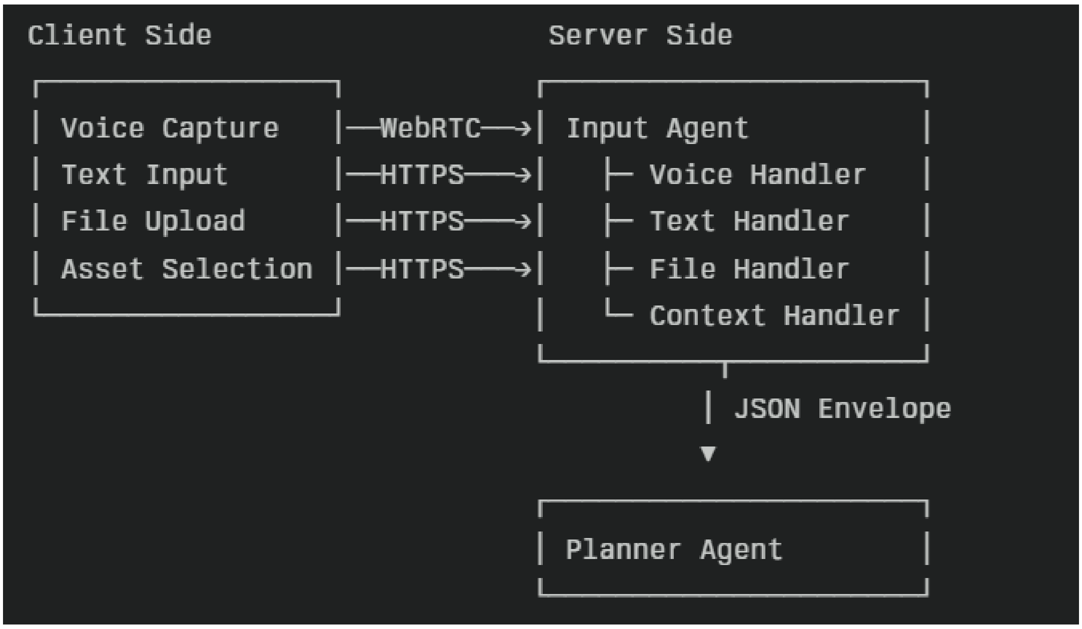
Architecture Overview

Core Components

1. Input Reception Layer
    - Text Input Handler: Processes typed messages from the conversational interface
    - Voice Input Handler: Manages WebRTC audio streams and real-time transcription
    - File Input Handler: Processes document/file uploads and metadata extraction
    - Asset Context Handler: Manages selected asset data integration
2. Processing Pipeline
    - Voice Processing Engine: Handles ASR (Whisper V3), speaker diarization and confidence scoring
    - Envelope Generator: Creates unified JSON structures for all input types
    - Context Aggregator: Combines multimodal inputs with asset context
3. Output Interface
    - Planner Agent API: Delivers processed inputs to the reasoning engine
    - Audit Logger: Records all input activities for compliance and analytics

Technical Specifications

Data Flow Architecture



```
Client Side                    Server Side

┌─────────────────┐            ┌───────────────────────┐
│ Voice Capture   │──WebRTC──→│ Input Agent           │
│ Text Input      │──HTTPS───→│    ├─ Voice Handler   │
│ File Upload     │──HTTPS───→│    ├─ Text Handler    │
│ Asset Selection │──HTTPS───→│    ├─ File Handler    │
└─────────────────┘            │    └─ Context Handler │
                               └───────────┬───────────┘
                                         │ JSON Envelope
                                         ▼
                               ┌───────────────────────┐
                               │ Planner Agent         │
                               └───────────────────────┘
```

Input Data Types & Envelope Structure

Unified Input Envelope (JSON)

json

```
{
  "envelope_id": "uuid",
  "timestamp": "xx-xx-xxxx xx:xx",
  "user_id": "string",
  "session_id": "string",
  "input_type": "text|voice|file|multimodal",
  "content": {
    "text": "string",
    "transcript": {
      "raw": "string",
      "interim": ["array of strings"],
      "final": "string",
      "confidence": 0.95,
      "speaker_info": {
        "speaker_id": "string",
        "diarization": true
      }
    },
    "files": [{
      "id": "uuid",
      "name": "string",
      "type": "mime-type",
      "size": "bytes",
      "url": "string"
    }],
    "asset_context": [{
      "asset_id": "uuid",
      "data_item": "string",
      "metadata": {}
    }]
  },
  "metadata": {
    "voice_mode_active": true,
    "interaction_method": "typing|speaking|upload",
    "client_info": {}
  }
}
```

Implementation Requirements

Client-Side Responsibilities

- Audio Capture: WebRTC-based voice recording with Opus codec
- Real-time Streaming: Voice Activity Detection (VAD) and audio transmission
- UI Management: Voice mode activation, file selection, asset browsing
- Local Processing: Basic audio preprocessing and noise suppression

Server-Side Responsibilities

- ASR Processing: Whisper V3 integration for speech-to-text conversion
- Speaker Diarization: Multi-speaker identification and segmentation
- File Processing: Document parsing and metadata extraction
- Envelope Generation: Unified data structure creation

- Context Integration: Asset data inclusion and validation

Performance Requirements

Latency Targets

- Voice Transcription: < 300ms time-to-first-word
- Text Processing: < 100ms processing time
- File Upload: < 2 seconds for documents up to 10MB
- Envelope Generation: < 50ms packaging time

Scalability Specifications

- Concurrent Users: Support 1000 simultaneous voice sessions (post production)
- Throughput: Process 10,000 text inputs per second
- Audio Quality: 16kHz, 16-bit, mono audio streams
- File Limits: 100MB max file size, 50 files per conversation

Error Handling & Recovery

Voice Processing Errors

- ASR Failures: Fallback to text input mode with user notification
- Audio Stream Issues: Automatic reconnection with buffer management
- Speaker Diarization Errors: Continue processing with single-speaker assumption

File Processing Errors

- Unsupported Formats: Clear error messages with supported format list
- Size Limits: Progressive upload with compression suggestions
- Corruption: Checksum validation with re-upload prompts

Asset Integration Errors

- Connection Failures: Cache last-known state, retry with exponential backoff
- Permission Issues: Clear authorization prompts and fallback options

Security & Compliance

Data Protection

- Voice Data: Store raw transcriptions separately from conversation history
- File Encryption: AES-256 encryption for all uploaded documents
- Access Controls: Purpose-based permissions for enterprise assets
- Audit Trail: Comprehensive logging of all input activities

Privacy Controls

- Data Retention: Configurable retention periods for different input types
- User Controls: Clear opt-out mechanisms for voice processing

Integration Points

External Dependencies

- ASR Service: Whisper V3 or equivalent streaming ASR
- TTS Service: ElevenLabs V3 for voice response generation
- File Storage: Secure cloud storage for uploaded documents
- WebRTC Infrastructure: Real-time communication platform (e.g., LiveKit)

Internal Integrations

- Planner Agent: Primary consumer of processed inputs
- Memory System: Episodic and semantic memory storage

- Asset Directory: Connected asset metadata and capabilities
- Audit System: Comprehensive activity logging

Success Metrics

Quality Metrics

- Transcription Accuracy: > 95% word error rate
- Context Preservation: 100% asset context accuracy
- Envelope Completeness: < 0.1% malformed envelopes

Performance Metrics

- Response Time: Meet all latency targets 95% of time
- Availability: 99.99% uptime for input processing
- Error Rates: < 0.1% processing failures

\

Development Timeline

Phase 1 (2 months)

Core text and file input processing

Voice processing and WebRTC integration

Asset context integration and optimization

Key Milestones

Text input processing complete

Voice transcription pipeline operational

Full multimodal integration and testing complete