Maven Code

Multimodal Vision Platform
Vision

Version 1.0

Revision History

Date	Version	Description	Author
09/22/25	1.1	Initial Draft	MavenCode Team

Table of Contents

Vision

1. Introduction

The purpose of this document is to collect, analyze, and define the business requirements, such as high-level needs, desired ultimate business outcomes and features of the MavenCode multi-modal platform.. It focuses on the capabilities needed by the stakeholders and the target users, and why these needs exist in the first place. The details of how the MavenCode multi-modal platform fulfills these needs are detailed in the use-case and supplementary specifications

1.1 Background

Currently, many essential processes require extensive time and resources for companies across every industry. Some industries, like agriculture, require constant monitoring and data analysis on vast amounts of information. Along with this, there are many different types of data that must be recorded, interpreted, and acted upon in real-time, such as video, image, text, and audio. The big investment of time, effort, and money, which is necessary for these tasks to get completed, is also reliant on whether or not the hired party works diligently. This uncertainty can make the whole process seem like a risk, which results in an uneasy feeling for everyone involved. For these reasons, companies have continuously searched for alternative solutions, that don't require constant monitoring and input from someone with an extensive technological background. One of the main pieces of technology currently being used to carry-out these advanced processes is through drones, but current implementations often require highly experienced drone-pilots, and the capabilities can be limited even then.

1.2 References

Object Detection & Classification with MCP Server, MavenCode

Semantic Segmentation & Spatial Understanding with MCP Server, MavenCode

Speech Processing & Audio Intelligence with MCP Server, MavenCode

2. Business Requirements

Business Opportunity/Problem Statement

The problem of	*limited vision-language understanding in current systems*
affects	*Maven code & their clients (car companies, real estate companies)*
the impact of which is	*quick analysis , decision making, gathering metrics*
a successful solution would be	*a system that integrates advanced VLMs with reasoning, temporal understanding, and multimodal fusion to deliver accurate descriptions, visual Q&A, domain-specific insights, and accessible services at scale.*

2.1 Business Objectives

BO-1: High-performance multimodal analysis

- Deliver single-image analysis in under 200 ms and handle 100+ concurrent requests/second, ensuring production-grade scalability

BO-2: Accuracy

- Achieve >98% OCR accuracy for printed text, >95% for handwritten text, and >90% accuracy for core vision-language tasks

BO-3: Cost efficiency

- Optimize compute usage to reduce cost per request while maintaining throughput, with quantization (INT8/FP16) and GPU acceleration.
- Track API usage growth across endpoints and integrate semantic caching

BO-4: Customer impact

- Improve accessibility with real-time image narration and document reading assistance, aiming to serve 50+ simultaneous users per session with <2GB RAM each
- Regular feedback and NPS scores

BO-5: Broad Market adoption

- Create models that target various key verticals (e.g, security, agriculture)

2.2 Success Metrics

SM-1**:** OCR Accuracy: >98% for printed text, >95% for handwritten

SM-2: Processing Speed: <3s for single image analysis

SM-3; Document Analysis: <5s for multi-page document processing

SM-4: Video Processing: 10fps for real-time analysis

SM-5: Concurrent Users: 50+ simultaneous vision processing sessions

SM-6: Memory Efficiency: <2GB RAM per concurrent session

SM-7: Availability: Maintain 99.9% uptime SLA, with robust monitoring, failover, and load balancing

2.3 Vision Statement

For	Teams working with real-world audio/video who need speech, vision, and spatial understanding.
Who	Want simple, unified APIs instead of building ML pipelines from scratch.
The (product name)	The MavenDrone is a modular MCP-based AI platform with FastAPI endpoints.
That	Provides STT/TTS, object detection & tracking, semantic/panoptic segmentation, depth, layout, spatial queries, and multimodal reasoning real-time, streaming with switchable model backends.
Unlike	Ad-hoc notebooks or one-off demos that don't scale or integrate with production tools.
Our product	Delivers defined endpoints and SDKs, clear performance targets, and GPU/CUDA-ready Docker deployment, so teams can go from quick demo to reliable production.

For teams that need speech, vision, and spatial understanding from real-world audio/video. Who want simple, unified APIs to transcribe/synthesize speech and to segment scenes with 3D/spatial reasoning. The MavenDrone is an MCP-based platform with FastAPI endpoints for STT/TTS and for semantic/panoptic segmentation, depth, layout, and spatial queries. That delivers real-time processing and lets you switch model backends to balance accuracy and speed.Unlike ad-hoc notebooks that don't scale or integrate, it provides defined endpoints, clear performance targets, and deployment guidance on GPU/CUDA and Docker.

2.4 Business Risks

RI-1: Performance targets not met like STT/TTS latency and/or segmentation FPS/accuracy. (Probability = 0.4; Impact = 5)
RI-2: Privacy/Security/Compliance requirements block certain use cases or telemetry. (Probability = 0.3; Impact = 4)

RI-3: Scalability under load like concurrency, streaming, load balancing, & queueing hurts reliability. (Probability = 0.35; Impact = 4)

RI-4: Model/backend availability or changes causing disrupted workflows. (Probability = 0.25; Impact = 3)

RI-5: GPU/CUDA resources or container orchestration are unavailable or limited. (Probability = 0.3; Impact = 4)

2.5 Business Assumptions and Dependencies

AS-1: The system follows an MCP server pattern with FastAPI and HTTP/WebSocket endpoints.
AS-2: Model backends are switchable to trade speed vs. accuracy.
AS-3: Stated performance goals like speech latency, segmentation FPS/accuracy, spatial metrics are the evaluation bar.
AS-4: Streaming support is available where called out in the services.
DE-1: APIs/Endpoints, plus model list/switch endpoints.
DE-2: Models/Frameworks: Whisper or faster-whisper; Coqui TTS/Bark; SAM 2.0, SegFormer, Mask2Former; depth via MiDaS/DPT; 3D via Open3D/PyTorch3D.
DE-3: LLM Backends: smaller (Llama/Qwen) and larger (GPT-4V/Claude Vision/Llama 3.1/70B) with runtime switching.
DE-4: Infra/Runtime: Docker containers with GPU/CUDA; Redis caching; optional S3; load balancing and queueing for scale.
DE-5: Security/Compliance: authentication and rate limiting, TLS, encrypted storage options, deletion policies, and audit trails.

3. Stakeholder Profiles and User Descriptions

3.1 Stakeholder Profiles

Stakeholder	Major value or benefit from this product	Attitudes	Major features of interest	Constraints	End user or not?
Retail and E-Commerce	Quicker property estimation.	Strong focus on accuracy.	Ability to identify general square feet of property, materials used and outdoor assets.	Drone most likely will not be going inside	No
Insurance adjusting	Being able to assess property damage.	Concern about union relationships and possible downsizing; otherwise receptive	Quicker and safer damage assessment.	Training users in conducting drones	No
Medical Image Analysis	Better and faster tumor analysis. Quicker turn around for intervention.	Strong enthusiasm, but might not use it as much as expected because of social value of eating lunches in cafeteria and restaurants	Simplicity of use; Reliability; Accuracy.	Unknown more research needs to be done	Yes
Companies needing more accessibility with TTS and STT	Allowing users with disabilities such as hearing or vision loss to obtain the information they need and interact with	Empathy. Open minded. Being creative on how these solutions can be implemented.	Minimal changes in current payroll applications	No resources yet committed to make software changes	No
Farmers	AI is able to predict future crop yields and issues.	Understanding of the variability of farming environments which would affect predictions		Might not have staff and capacity to handle order levels; might not have all menus online	No

3.2 User Environment

The company serves users across several industries, so the working environment varies widely. In general, their solutions reduce the number of people needed to complete tasks by automating complex or repetitive work with AI/ML. For example, instead of a team manually reviewing patient records, a healthcare provider can rely on ML pipelines to pre-analyze data, or in finance, analysts can let automated fraud detection models handle high transaction volumes.

Task cycles that once took days or weeks (like setting up infrastructure, analyzing seismic datasets, or building PoCs) are now compressed into hours or days using Google Cloud tools. This allows end users to spend less time on manual data handling and more time making decisions.

Each industry has its own constraints: healthcare requires HIPAA compliance and patient privacy, finance needs low latency and strict regulatory oversight, and oil & gas often involves outdoor or remote work where drones and sensors feed data into ML workflows. Current platforms mix legacy systems with Google Cloud products such as Kubernetes, Dataflow, BigQuery, and Cloud ML, and integration is often needed with existing systems like EHRs, CRMs, trading platforms, or seismic analysis tools.

3.3 Alternatives and Competition

Stakeholders could consider several alternatives instead of using this company's solutions. One option is training and managing their own machine learning models in-house, which offers full control but requires significant investment in infrastructure, talent, and time. Another alternative is relying on general-purpose AI platforms such as ChatGPT, Gemini, or Claude, which can provide powerful off-the-shelf capabilities but may lack the industry-specific customization, integration, and data security that enterprise clients need. Competitor cloud providers also exist and offer comparable machine learning and analytics services, though they may present higher complexity or integration challenges.

4. Scope and Limitations

The Multimodal Vision Platform is designed to deliver a production-grade system that enables intelligent, agentic AI interactions through the integration of computer vision, speech processing, and language understanding. Structured around a modular architecture and built on the Model Context Protocol (MCP), the platform encompasses four primary subsystems: VoiceVision for speech and audio intelligence, VisionLang for vision-language understanding, ObjectTracker for object detection and tracking, and SpatialSense for semantic segmentation and spatial analysis. These components collectively enable capabilities such as real-time speech-to-text and text-to-speech synthesis, image captioning and visual question answering, object classification and multi-object tracking, 3D scene reconstruction, and cross-modal reasoning that fuses visual and auditory inputs into coherent outputs. A shared infrastructure supports containerized microservices, standardized APIs, real-time monitoring, and scalable deployment across environments.

While the platform introduces advanced multimodal functionality, it also operates within several constraints. Real-time performance targets such as low latency and high concurrency may be impacted by hardware limitations or under-resourced deployments. Some domain-specific applications, including medical analysis or support for low-resource languages, may require extensive fine-tuning and data curation beyond the scope of the initial release. Audio processing components, particularly speech recognition, may face accuracy challenges in noisy or variable environments. Given the complexity of integrating multiple AI services, early versions may exhibit inconsistencies or synchronization issues during multimodal interaction. Additionally, although the platform includes basic API security, authentication, and monitoring, enterprise-grade regulatory compliance such as HIPAA or GDPR is not guaranteed at launch. Finally, the initial production deployment will be limited to a closed beta phase in order to validate functionality, gather user feedback, and optimize system performance before a wider release.

The digital twin/simulator can be used for offline testing and validation. The project does NOT provide a live, real-time synchronization between a Gazebo simulator and a physical drone — live double-navigation (simultaneous mirroring of a physical drone and a simulator) is unsupported and unsafe unless custom synchronization is implemented.

Despite these limitations, the platform is structured for iterative improvement, with each development sprint focused on enhancing feature maturity, integration stability, and system scalability. The architecture is intentionally designed to accommodate future expansions while maintaining a strong foundation for robust multimodal AI services.

4.1 Product Perspective

This project builds on the MVP that was made last year. The new version is part of a bigger system at MavenCode that uses AI to understand both visuals (images and videos) and audio (speech). The platform takes in video, images, or audio and can answer questions about them in plain language. It connects to existing tools for labeling and data management, and it’s designed to work easily with other systems through APIs (like REST, gRPC, or WebSockets).

4.2 Major Features / Scope

The Multimodal Vision Platform is a production-grade, intelligent system designed to enable seamless agentic AI interactions through multimodal inputs (vision, speech, and language). The system is organized into four major

functional subsystems and a supporting platform services layer. Each subsystem provides high-value features addressing core use cases for end users, developers, and system integrators.

1. VoiceVision MCP Services (Speech Processing & Audio Intelligence)

FE-1: Real-Time Speech-to-Text (STT)
Convert live or recorded audio input into accurate, structured text in real-time. Supports streaming protocols and multi-language input.

FE-2: Text-to-Speech (TTS) Synthesis
Generate natural, expressive audio output from text prompts using neural TTS engines. Supports multilingual voices and emotional tone rendering.

FE-3: Audio Command Recognition and Parsing
Interpret user-issued spoken commands for system navigation, task execution, or agent instruction. Includes support for intent detection and context disambiguation.

FE-4: Speaker Diarization and Emotion Detection
Identify individual speakers and detect emotional states from audio streams to enable more context-aware interactions.

---

2. VisionLang MCP Intelligence (Vision-Language Understanding)

FE-5: Image Captioning and Understanding
Automatically generate descriptive captions for images, providing accessible visual summaries or aiding content indexing.

FE-6: Visual Question Answering (VQA)
Answer user queries about specific content within an image, such as “What is the person holding?” or “How many animals are there?”

FE-7: Visual Reasoning and Interpretation
Understand complex image scenarios, including spatial relationships, object interactions, and logical inferences (e.g., causality, comparisons).

FE-8: Video Comprehension and Summarization
Analyze videos to detect actions, events, and sequences; generate narrative summaries; and extract frame- or scene-level insights.

FE-9: Multimodal Narrative Generation
Combine visual and audio inputs to generate coherent, story-like outputs for accessibility, educational, or creative applications.

---

3. ObjectTracker MCP Services (Object Detection & Tracking)

FE-10: Real-Time Object Detection
Detect and classify multiple objects in images or live video feeds with bounding boxes and confidence scores.

FE-11: Fine-Grained and Domain-Specific Classification
Classify objects into detailed subcategories, supporting specialized domains such as medical, industrial, or sports applications.

FE-12: Multi-Object Tracking (MOT)
Track multiple moving objects across video frames while preserving unique identities and trajectories.

FE-13: Activity and Behavior Analysis
Identify patterns of behavior based on object interaction (e.g., crowd movement, anomalies, or suspicious activity).

FE-14: Scene Classification and Relationship Mapping
Determine the overall scene context and detect relationships among objects (e.g., "person holding object" or "vehicle near stop sign").

---

4. SpatialSense MCP Platform (Segmentation & Spatial Understanding)

FE-15: Semantic and Instance Segmentation
Provide pixel-level segmentation of images to identify object boundaries and classes, enabling detailed scene understanding.

FE-16: 3D Scene Reconstruction and Depth Estimation
Estimate spatial depth and generate point clouds from 2D images for applications like AR/VR or robotics.

FE-17: Panoptic and Multi-Scale Segmentation
Combine semantic and instance segmentation to produce comprehensive visual parsing of complex scenes.

FE-18: Spatial Navigation and Pathfinding Assistance
Understand physical space layouts to assist with navigation, route prediction, or obstacle avoidance.

FE-19: Temporal Consistency Across Frames
Maintain visual consistency in object segmentation and spatial understanding across image sequences or video streams.

---

5. Platform-Wide Services and Integration Features

FE-20: Model Context Protocol (MCP) Compliance
Ensure all subsystems communicate using the standardized MCP format to enable context-aware, agentic AI interactions.

FE-21: API Gateway and Multimodal Service Access
Expose unified APIs for developers and third-party systems to interact with multimodal services through RESTful endpoints.

FE-22: Cross-Modal Reasoning and Fusion
Enable combined understanding across modalities (e.g., linking speech with visual context or aligning audio with video scenes).

FE-23: Monitoring, Logging, and Performance Dashboards
Provide real-time observability into system health, model performance, and usage analytics for operations teams.

FE-24: User Access Control and Security
Support authentication, API key management, rate limiting, and audit trails to ensure secure and compliant use.

System Partitioning Strategy (Subsystems)

To manage complexity and align with team structure, the system is partitioned into the following subsystems:

- **VoiceVision Services Subsystem** – All speech/audio-related features (FE-1 to FE-5)
- **VisionLang Intelligence Subsystem** – Vision-language understanding and reasoning (FE-6 to FE-10)
- **ObjectTracker Services Subsystem** – Object detection, classification, and tracking (FE-11 to FE-15)
- **SpatialSense Platform Subsystem** – Image segmentation and spatial reasoning (FE-16 to FE-20)
- **Platform Services Layer** – Shared system-level infrastructure and integration services (FE-21 to FE-25)

Each subsystem supports multiple use cases and shares infrastructure components via MCP and containerized microservices architecture.

4.3 Deployment Considerations

- To use the system effectively, some things need to be in place:
- Hardware Needs – Powerful computers with GPUs (like NVIDIA A10 or better), large amounts of memory (64GB+), and fast storage for handling video and audio data.
- Scalability – The system should be able to grow and handle more users or bigger workloads by using container systems (like Kubernetes), multi-GPU setups, and smart batching.
- Integration – The product is “API-first,” meaning it’s built to connect with other systems easily through REST, gRPC, or WebSocket endpoints. SDKs are also available to help developers.
- Security & Privacy – Since the system works with video and audio, it must follow privacy and compliance rules (like GDPR or HIPAA). Data should be encrypted, access should be controlled, and logs should be kept for accountability.
- Deployment Options – It can be set up in the cloud or on a company’s own servers, depending on security and customer needs.

5. Other Product Requirements

- **Standards & Compliance**
  - MCP protocol compliance
  - GDPR/HIPAA where applicable
  - WCAG 2.1 AA accessibility
  - Model explainability & audit logs
- **Hardware & Platform**
  - GPU (NVIDIA A10+, 24GB VRAM)
  - 16+ CPU cores, 64GB+ RAM, SSD storage
  - Cloud (Kubernetes/Docker) + on-premise support
  - PostgreSQL + MongoDB storage
- **Performance & Reliability**
  - Latency <200ms per image
  - ≥100 concurrent requests/sec
  - 99.9% uptime SLA
  - Scalable via Kubernetes + multi-GPU
- **Fault Tolerance & Robustness**
  - Redundant services with failover
  - Graceful degradation on subsystem failure
  - Monitoring via Prometheus + Grafana
- **Environmental & Operational**
  - Cloud and edge deployment options
  - Regional data residency options
  - Privacy-by-design: minimize raw data storage
- **Documentation & Support**
  - User guides + API reference
  - SDKs + sample notebooks
  - Ops playbooks + monitoring dashboards