AI MOOD SHOT -ARCHITECTURAL SPECIFICATION

Behavioural architecture for large language models

Technical Architecture & Invariants

Author: William Argo, *Independent Research & IP Development*

Document Type: Technical Architecture / IP Foundation

Version: 1.0 OPEN USE / OPEN BUILD

Date: 03/05/2026

Contact: *william.argo@proton.me*

Section 0. Operator Responsibilities Overview

0.1 Purpose

This section summarises all operator defined elements required for Mood Shot. Mood Shot provides the mechanism. Operators provide the content that drives it. Mood Shot does not define behaviour. It injects whatever text operators supply.

0.2 Operator Defined Elements

Operators must define the following elements:

- slider dimensions
- slider ranges
- slider to text mappings
- preset definitions
- preset governance procedures
- safety constraints
- validation rules for detecting contradictory text combinations
- persistence rules for re-injecting text each turn
- versioning rules
- integration rules

0.3 Slider Range Assumption

Mood Shot supports any operator defined slider style, including percentage scales, bounded numeric ranges, or discrete categories. For the purposes of this architectural specification, Mood Shot assumes a ten-step numeric slider. Each position on the slider selects a specific text instruction supplied by the operator. This assumption standardises the mechanism without restricting implementers who wish to use alternative slider formats.

0.4 Model Awareness

Operators define slider mappings by observing how their deployed model behaves under real conditions. Model developers do not publish internal behavioural baselines or

stability characteristics. Operators therefore determine behaviour empirically by testing how their model responds to injected text. Calibration is performed through behavioural testing because only operators can observe how their specific model instance behaves in practice.

0.5 Architectural Position

Operator defined elements sit outside the Mood Shot architecture. They supply text content to the mechanism. Mood Shot does not alter identity, safety, or the layer model. It simply injects the operator supplied text in a structured and persistent way.

Section1. Executive Summary – AI Mood Shot Architectural Framework

1.1 Purpose

Mood Shot is a behavioural framework for large language models that stabilises and enriches user experience. People use these systems for many purposes such as roleplay, chatting, coding, creative writing, and searching, and each purpose benefits from a different interaction style. Mood Shot gives service operators a controlled way to offer these styles consistently, even when underlying models change or previously preferred behaviours disappear. Mood Shot achieves this by defining a set of operator controlled sliders. Each slider has ten positions. Each position corresponds to a specific text instruction. Mood Shot injects the selected text every turn. This modulates interaction style without altering identity. Presets are built on these sliders and provide user facing shortcuts to common configurations.

1.2 Problem Statement

Current systems collapse identity, tone, and task into a single prompt. This creates three distinct failure modes:

- **Identity instability** - the model's baseline behaviour shifts unpredictably across turns and updates because identity is entangled with temporary instructions.
- **Tone volatility** - interaction style drifts mid-conversation, varies between sessions, and changes when the underlying model is updated.
- **Task contamination** - instructions intended for the immediate request leak into long-term behaviour, while attempts to enforce tone or persona interfere with task execution.

Persona prompts attempt to solve these issues but operate at the wrong layer. They rewrite identity instead of modulating behaviour, causing drift, inconsistency, and loss of control. Services cannot guarantee stable interaction styles, and users lose access to preferred behaviours when models change.

Mood Shot addresses these failures by separating identity, affect, and task into independent layers and by providing a simple mechanism for modulation. Sliders select text. Mood Shot injects that text every turn. This produces stable behaviour without altering identity.

1.3 Architectural Solution

Mood Shot separates behaviour into three layers:

- Identity Layer. The stable baseline that defines how the model behaves in general.
- Affective Layer. A temporary modulation applied at runtime. This layer is implemented through sliders or presets built on sliders. Each slider position maps to a text instruction.
- Task Context. The immediate instructions required to complete the user's request.

This separation prevents cross contamination between layers and produces predictable behaviour across long sessions. Sliders provide the mechanism for affective modulation, and presets provide optional predefined combinations of slider values.

1.4 Scope and Non-Scope

Mood Shot defines the mechanism, not the content. It does not specify emotional scales, slider catalogues, presets, or trait definitions. Implementers provide their own slider dimensions, preset catalogues, intensities, and safety constraints according to their service requirements and policies.

Examples of implementer-defined elements may include:

Slider dimensions

- directness
- warmth
- formality
- creativity
- conciseness
- enthusiasm
- analytical depth
- humour
- narrative richness
- emotional expressiveness

Slider ranges

ten step numeric scales such as 1 to 10

• percentage scales such as 0 to 100

• bounded numeric scales such as 0 to 1 or negative to positive

• discrete steps such as low, medium, high

Preset examples

• friendly

• concise

• formal

• playful

• high context

• low context

• technical

• narrative

• neutral

These examples illustrate the types of content an operator may define, but they are not part of the Mood Shot specification.

1.5 Deployment Awareness

The architecture supports different deployment patterns. It can be exposed as a simple menu of predefined styles, as a set of adjustable sliders, or as a hybrid interface combining both. Mood Shot defines boundaries and interaction rules but does not prescribe algorithms, APIs, or UI design. Sliders may be user visible or operator only depending on implementation.

1.6 Outcome

Mood Shot provides a minimal, modular behavioural layer that allows services to offer differentiated interaction styles without compromising identity stability. It enables consistent user experience across model updates and supports a wide range of tasks and user preferences. The technical specification defines slider rules, preset rules, enforcement rules, state boundaries, and integration requirements.

Section 2. Architectural Principles and Scope

2.1 Core Principles

These principles define what Mood Shot is and anchor every later section. See Section 0 for operator responsibilities overview.

- **Layer separation**.

  Identity, affect, and task must remain distinct. No layer may rewrite another.

- **Predictability.**

  Behaviour must remain stable across turns and across model updates.

- **Modulation minimalism**.

  Affective modulation is expressed through a finite set of operator defined sliders.

- **Black box respect**.

  The underlying model is treated as opaque. Mood Shot does not assume internal structure. Operators define slider mappings empirically by testing how their deployed model responds to injected text.

- **User experience primacy**.

  Mood Shot is designed to give operators reliable, predictable control over interaction style in production systems.

- **Research and development value**.

  Mood Shot's clean layer separation and explicit modulation controls support controlled experiments and reproducible behavioural conditions.

- **Operator control.**

Mood Shot provides a controlled behavioural adjustment mechanism that current systems do not offer.

2.2 Architectural Constraints

Constraints describe what the system must obey to remain valid.

- **No identity mutation**.

  Identity is fixed and cannot be altered by affect, sliders, or task.

- **No cross contamination**.

Affective modulation cannot leak into identity or task.

• **No implicit scales**.

The architecture does not define emotional intensities or behavioural gradients. All slider ranges are explicit.

• **No safety rewrites**.

Mood Shot does not modify or replace safety systems.

• **Model agnostic operation**.

Works with any LLM without requiring internal access or tuning.

2.3 Boundaries of the System

What Mood Shot covers and what it explicitly does not.

Covers behavioural modulation - selecting and injecting operator supplied text at runtime through sliders or presets.

•**Does not cover identity design** - identity content is supplied by the implementer. Identity content is supplied entirely by the operator and is not modulated by sliders or presets.

• **Does not cover trait catalogues** - slider dimensions, presets, and intensities are external.

• **Does not cover UX design** - menus, sliders, and UI patterns are implementation choices.

• **Does not cover model tuning** - no fine-tuning, RLHF, or internal adjustments.

2.4 Implementer Responsibilities

What the service operator must provide.

•**Slider definitions** - the behavioural dimensions and the ten step numeric ranges used to select text templates.

• **Preset configurations** - optional predefined Mood Shots built from slider values.

• **Safety constraints** - rules governing which sliders or combinations are allowed or disallowed.

• **Integration rules** - how Mood Shot is applied within the service pipeline.

2.5 User-Experience Framing

The architectural purpose in service terms.

- **Enriching user experience** - users want different interaction styles for different tasks, contexts, and personalities.
- **Preserving preferred behaviours** - Mood Shot protects the feel users rely on by re-injecting the same operator supplied text every turn, even when underlying models change.
- **Avoiding one-size-fits-all** - a single default tone cannot serve all tasks, all users, or all service environments.
- **Operational stability** - services adopt Mood Shot because user-facing systems require stable, predictable interaction styles across updates, deployments, and model revisions.

Section 3. Behavioural Architecture

3.1 Overview

Mood Shot operates as an external orchestration layer that structures inputs into three logical components. These components are described using internal behavioural terminology for clarity, but they do not modify or replace any part of the underlying model. The architecture defines how identity, affect, and task context are assembled into a single behavioural stack while remaining isolated from one another. Affective modulation is expressed through operator defined sliders. Each slider position selects a specific text instruction that Mood Shot injects every turn.

3.2 Layer Model

Mood Shot does not replace any existing architecture inside the model. It is an external orchestration layer that organises inputs into three logical components. These components are described using internal-style language for clarity, but they are implemented outside the model and require no modification of the underlying system.

Identity layer

A persistent behavioural baseline supplied by the operator. It corresponds to the stable system-level behaviour that services already define, but Mood Shot makes this layer explicit and protected.

Affective layer

A temporary modulation applied at runtime through a set of operator defined sliders or a preset composed of slider values. Each slider position corresponds to a specific text instruction. Mood Shot injects this text every turn so affective modulation can be applied without rewriting identity.

Task context

The immediate instructions for the user's request. This is the same concept services already use, but Mood Shot defines the interaction boundaries so task context cannot reinterpret, override, or displace identity or affect within the assembled behavioural stack.

Safety layer

A supervisory constraint system that evaluates the assembled stack after modulation and before emission. It overrides all other layers when conflicts occur. It contains no identity or affect content. It enforces boundaries and rejects unsafe or contradictory text

combinations produced by slider selections. Mood Shot therefore introduces a new structure, but it is built entirely on top of existing mechanisms.

3.3 State Invariants

The architecture enforces the following invariants:

- **Identity immutability** - identity cannot be overwritten, replaced, or reinterpreted by affect or task.
- **Affect containment** - affective modulation persists only through text re-injection during the conversation and does not carry into unrelated interactions.
- **Task locality** - task context cannot alter identity or affect and cannot persist beyond the request that created it.
- **Layer isolation** - no layer may modify another layer's content or memory.
- **Stack order** - the behavioural stack must always resolve in the order: identity → affect → task.

Task context must not alter affective modulation.

3.4 Lifecycle

A Mood Shot follows a predictable lifecycle:

1. **Creation** - an affective modulation is selected or generated through sliders or a preset.
2. **Application** - the selected slider values determine which text instructions are injected between identity and task.
3. **Runtime** - the model responds using the combined stack.
4. **Continuity** - the modulation persists for the duration of the conversation, including across backend instance changes, so the user experiences a consistent interaction style. A backend instance change is a container level transition such as routing changes, shard changes, or container restarts. These transitions must not reset affective modulation.
5. **Expiry** - the modulation ends when the conversation ends or when the user selects a different modulation.
6. **Cleanup** - the system returns to identity plus task only.

Modulation state is conversation scoped. It must persist across backend instance changes within the same conversation. It must not persist across different conversations unless the user explicitly selects a preset or favourite. This ensures that affective modulation remains stable for the user without accumulating across unrelated interactions.

3.5 Layer Interaction Rules

The architecture defines the following interaction rules:

- **Identity precedence** - identity has precedence over affect.
- **Affect precedence** - affect has precedence over task.
- **Priority resolution** - conflicts are resolved by layer priority.
- **Instruction integrity** - no layer may reinterpret or rewrite another layer's instructions.
- **Boundary preservation** - the assembled input must preserve the boundaries of each layer.

3.6 Enforcement Mechanisms

The system applies the following mechanisms to maintain stability:

- **Layer validation** - checks to ensure each layer is structurally correct.
- **Identity protection** - detection of attempts by affect or task to alter identity.
- **Affect expiry** - automatic termination of affective states at the end of a conversation.
- **Reset procedures** - restoration to identity plus task only.
- **Combination rejection** - contradictory or unsafe text combinations produced by slider selections are rejected.

3.7 Configuration Testing

Valid configurations maintain clear separation between identity, affect, and task. Affective modulation is temporary and expressed through sliders or presets, and task context is local to the request.

Invalid configurations include:

- identity being rewritten by affect
- affect persisting across unrelated tasks
- task instructions modifying identity
- affect redefining safety rules
- slider values that produce contradictory or disallowed text templates

Labs should test combinations of identity, affect, and task to verify stability across model updates and backend instance changes.

Section 4. Operational Rules

4.1 Application Requirements

The operator must assemble identity, affect, and task in the defined order. Each layer must be supplied as a distinct component, and no layer may contain instructions intended for another. Identity must remain stable across the service. Affective modulation must be applied only when a slider configuration or preset is selected, and the corresponding text instructions must be injected every turn. Task context must be limited to the immediate user request. See Section 0 for operator responsibilities overview.

The system must ensure that the combined input preserves the boundaries of each layer and that the ordering **identity → affect → task** is maintained.

4.2 Operator Responsibilities

The operator is responsible for maintaining a protected identity layer that cannot be altered by user input or by affective modulation. The operator must define valid slider dimensions, ten step numeric ranges, and presets, and ensure the selected text templates do not conflict with identity or safety requirements. The operator must enforce expiry rules so that affective modulation ends at the close of a conversation unless the user selects a different modulation.

The operator must ensure that task context is isolated to the request that created it and does not persist into subsequent interactions.

4.3 Scope and Duration

Affective modulation must persist for the duration of a conversation through continuous re-injection of the selected text so the user experiences a consistent interaction style. This continuity must be maintained across backend instance changes. Affective modulation must not persist across unrelated conversations unless the user explicitly saves it as a preferred preset.

Identity must persist across all conversations unless the operator updates it. Task context must end when the request is complete.

4.4 Failure Handling

If any layer is malformed, incomplete, or structurally invalid, the system must reject the configuration and fall back to identity plus task only. If a conflict arises between layers or if slider selections produce contradictory text, the system must resolve it by layer priority. If affective modulation attempts to alter identity or safety constraints, the system must discard the modulation.

If task context contains instructions that would modify identity or affect, the system must remove those instructions before processing the request.

4.5 Recovery Procedures

When contamination is detected, the system must reset to identity plus task only. If a conversation ends unexpectedly, the system must expire the active affective modulation. If backend instance changes occur, the system must reapply the active modulation for the duration of the conversation.

If a preset becomes invalid because its text templates have been updated or removed, the system must notify the operator and revert to identity plus task until a valid preset is selected.

Section 5. Modulation Standards

5.1 Slider Structure

Sliders define the behavioural dimensions used for affective modulation. Each slider must have a clear name, a ten step numeric range, a default value, and a mapping that assigns a specific text instruction to each value. Slider definitions must be explicit, self-contained, and free of identity or task instructions. Slider boundaries must be enforced at all times, and slider values must remain within the permitted range during runtime. See Section 0 for operator responsibilities overview.

5.2 Slider Validation Rules

Sliders must be validated before they can be used. Validation must confirm that each slider has a defined ten step range, a valid text mapping, and boundaries that comply with safety requirements. Validation must reject sliders that contain undefined ranges, unsafe mappings, ambiguous parameters, or values that conflict with identity or safety constraints. Validation must confirm that slider combinations are safe when used together, not only individually.

5.2.1 Slider Conflict Resolution

Opposing traits are traits that produce directionally inverse behavioural effects along the same conceptual dimension. For example, a dimension that ranges from concise to verbose or from formal to casual represents a single axis with two ends. These ends are opposing traits.

Conflicting sliders are separate sliders whose combined effects attempt to push behaviour in incompatible directions. A conflict occurs when two sliders independently attempt to control the same behavioural dimension or when their combined values would produce behaviour outside the permitted boundaries.

Incompatible combinations are slider configurations that cannot be resolved without violating identity, safety constraints, or the permitted behavioural range.

Rules

1. Opposing traits must be represented on a single slider axis rather than split across multiple sliders.
2. If two sliders attempt to control the same behavioural dimension, the safety layer must reject or override the conflicting text templates.

3. If a slider combination produces contradictory or unsafe text templates, the system must reject the configuration.
4. Identity and safety constraints take precedence over all slider combinations.

5.3 Preset Structure

A preset is an operator defined combination of slider values. Presets must be defined as structured objects containing only the fields permitted by the system. Each field must be explicit, self-contained, and free of identity or task instructions. Presets must not bypass or redefine slider dimensions.

A preset must include the following components:

Name

A unique identifier used for selection and reference.

Description

A short summary of the intended interaction style.

Affective parameters

The slider values that select the text templates defining the preset's behavioural effect. These values must operate within the allowed ranges defined by the system.

Constraints

Any limits on how the preset may be applied, including incompatibilities with specific tasks or modes.

Safety boundaries

Rules that ensure the preset cannot alter identity or override safety requirements.

A preset must not include any fields beyond those defined in this section. Presets must not include identity content or identity instructions.

5.4 Preset Validation Rules

A preset must be validated before it can be used. Validation must confirm that the preset is structurally complete, that all slider values are within allowed ranges, and that no field conflicts with identity or safety constraints.

Validation must reject presets that:

- contain instructions intended for identity or task

- attempt to modify identity
- select text templates outside the allowed slider ranges
- introduce unsafe behavioural patterns
- contain ambiguous or undefined parameters
- conflict with system level safety rules

A preset that fails validation must not be deployed.

5.5 Preset Governance

The operator is responsible for creating, approving, updating, and retiring presets. All presets must be versioned. Updates must be applied in a controlled manner to ensure compatibility with identity, slider boundaries, and safety constraints.

Governance must include:

- approval procedures for new presets
- version control for all changes
- documentation of intended use
- review of presets after system updates
- recalibration of slider values when text templates require adjustment after system changes
- retirement of presets that no longer meet requirements

A preset must not be deployed until it has passed governance review.

5.6 User Facing Presets

Presets must be presented to users in a consistent and predictable format. Each preset must have a clear name and a concise description. Users may select presets at any time during a conversation. A preset selected by the user must persist for the duration of the conversation through continuous re-injection of its selected text templates.

Users may save their own combinations of slider values as favourites. A favourite must persist across conversations until the user removes it or changes it and must be validated for safety before reuse.

Preset names and descriptions must not imply changes to identity or safety rules.

5.7 Compatibility Requirements

All presets and slider configurations must remain compatible with identity, affective boundaries, and safety constraints. Presets must not rely on model specific behaviours that may change over time. Presets and slider mappings must be reviewed after system updates to ensure their text templates remain compatible.

If the underlying model changes, presets may require recalibration of their slider values to provide a similar experience unless the preset is being retired by the operator. If a preset becomes incompatible due to system changes, it must be retired or updated. Incompatible presets must not be applied.

5.8 Preset Lifecycle

A preset follows a defined lifecycle.

Creation

The operator defines the preset structure and slider values.

Validation

The preset is checked for structural correctness and safety compliance.

Approval

The preset is reviewed and approved through governance procedures.

Deployment

The preset becomes available for user selection.

Active use

The preset may be selected by users and applied during conversations.

Recalibration

The preset may require updated slider values when its text templates need adjustment due to system behaviour or safety boundary changes.

Deprecation

The preset is marked for removal when it no longer meets requirements.

Retirement

The preset is removed from active use and archived.

A preset must not skip any stage of this lifecycle.

Section 6. Verification Requirements

6.1 Layer Isolation Testing

The system must be tested to confirm that identity, affect, and task remain isolated under all supported conditions. Tests must verify that identity cannot be altered by affect or task, that affect cannot modify identity or safety rules, and that task context cannot persist beyond the request that created it.

Isolation tests must include scenarios where user input attempts to introduce identity content into affect or task or attempts to embed task instructions into the injected text templates. . The system must reject or correct these cases.

6.2 Continuity Testing

The system must be tested to confirm that affective modulation persists for the duration of a conversation. Tests must include backend instance changes, service restarts, and other infrastructure transitions. The system must re-inject the selected text templates so that the user experiences a consistent interaction style.

Continuity tests must verify that identity remains stable across all transitions and that task context does not persist beyond its intended scope.

6.3 Expiry and Reset Testing

The system must be tested to confirm that affective modulation expires when a conversation ends. Tests must include abrupt termination, timeouts, and user-initiated closure. After expiry, the system must return to identity plus task only.

Reset tests must verify that no text templates from a previous modulation carry into a new conversation unless the user selects a preset or has saved a favourite. The system must ensure that favourites are applied only when appropriate and that no unintended modulation persists.

6.4 Slider & Preset Validation Testing

The system must be tested to confirm that sliders and presets are validated correctly before use. Tests must include valid slider ranges, invalid ranges, unsafe text templates, and slider values that select templates that conflict with identity or safety constraints.

Preset validation tests must include structurally incomplete presets, presets with out-of-range slider values, and presets that conflict with identity or safety constraints. Tests must confirm that invalid sliders or presets are rejected and that only approved presets can be deployed. Tests must also confirm that favourites remain valid after updates or are retired when no longer compatible.

6.5 Update Stability Testing

The system must be tested to confirm stable behaviour across model updates, service updates, slider revisions, and preset revisions. Tests must verify that identity remains consistent, that affective modulation behaves as intended, and that presets continue to produce the expected interaction style.

If model changes require recalibration of slider values or text templates, tests must confirm that the updated templates provide a similar user experience. If a preset is retired, tests must confirm that it is no longer available for selection and that favourites referencing it are handled according to operator rules.

Section 7. Safety Constraints

7.1 Identity Protection

The system must ensure that identity remains stable and cannot be altered by affect, task, sliders, presets, or user input. Identity must not be reinterpreted, rewritten, or overridden by any modulation. Sliders and presets must not contain identity content, and task instructions must not include identity modifications. Identity protection tests must confirm that attempts to alter identity, including attempts to embed identity content into injected text templates, are rejected and that identity remains consistent across all conversations and system updates. See Section 0 for operator responsibilities overview.

7.2 Safety Enforcement

Safety rules must take precedence over identity, affect, sliders, and task. Affective modulation must not weaken or bypass safety constraints. Slider values and preset configurations must not conflict with safety requirements, and the system must detect and reject unsafe or contradictory text templates. Safety enforcement must remain active across all backend instance changes, service updates, slider revisions, and preset revisions. The system must ensure that safety constraints cannot be disabled or modified. Safety constraints are fixed and cannot be altered by presets, sliders, or task instructions. Affective modulation must not weaken or reinterpret safety constraints.

7.3 Forbidden Modulations

Certain affective behaviours must not be permitted in sliders, presets, or runtime modulation. Forbidden modulations include behaviours that:

- encourage unsafe actions
- reduce clarity of safety instructions
- promote harmful or destabilising interaction styles
- attempt to override identity or safety rules
- introduce ambiguity in task execution through injected text templates
- conflict with system level behavioural requirements

The operator must maintain a list of forbidden modulation patterns and ensure that sliders and presets do not include them.

7.4 Conflict Resolution

When conflicts arise between identity, affect, sliders, task, and safety, the system must resolve them by layer priority. Safety overrides identity, identity overrides affect and affect overrides task. Conflicts must be resolved deterministically and without ambiguity.

If a conflict cannot be resolved without violating safety or identity constraints, including conflicts caused by injected text templates, the system must discard the lower priority layer. Only affect, slider values, or task may be removed. Identity and safety must not be discarded under any circumstances.

If removal of the lower layer does not resolve the conflict, the system must reject the configuration.

7.5 Operator Safeguards

The operator must implement safeguards to prevent unsafe configurations Safeguards must include monitoring for invalid sliders, invalid presets, auditing the text templates selected by presets, reviewing sliders and presets after system updates, and ensuring that favourites remain compatible with safety and identity constraints.

The operator must retire sliders or presets that no longer meet safety requirements and must ensure that users cannot select presets or slider combinations that have been retired or invalidated. The operator must maintain documentation of all safety related decisions and updates.

Section 8. Integration Requirements

8.1 Input Assembly

The system must assemble identity, affect, sliders, and task into a single structured input before sending it to the model. Each layer must be provided as a distinct component, and the ordering identity then affect then task must be preserved. Slider values must select the text templates that form the affective layer, and these templates must be injected every turn. No layer may contain instructions intended for another layer. See Section 0 for operator responsibilities overview.

Input assembly must ensure that identity remains stable, affective modulation is applied only when selected, slider values remain within allowed ranges, and task context is limited to the immediate request. The system must verify that the assembled input is structurally valid before processing.

8.2 Service Integration

Mood Shot must integrate with existing service logic without altering underlying model behaviour. The service must maintain conversation state, apply the selected preset or slider configuration, and ensure continuity of affective modulation across backend instance changes.

Service integration must ensure that favourites persist across conversations, that preset and slider selection is applied consistently, and that the selected text templates are re-injected every turn until the conversation ends. The service must prevent presets or slider configurations from being applied in contexts where they are incompatible.

8.3 User Interface Requirements

The user interface must present presets in a clear and consistent format. Each preset must display its name and description. Users must be able to select presets, change presets, adjust sliders if exposed, and save favourites.

The interface must ensure that preset or slider selection is explicit and that users understand when text-template based modulation is active. Favourites must be easy to manage and must reflect the current slider and preset standards. The interface must not imply that presets or sliders alter identity or safety rules.

8.4 Cross Platform Consistency

Mood Shot must behave consistently across all supported platforms, including web, mobile, desktop, and embedded surfaces. Preset names, descriptions, slider behaviour, and the text templates selected by sliders must be identical across platforms. The selection flow must follow the same logical structure even if the presentation differs.

Cross platform consistency tests must confirm that identity, affect, sliders, and task are assembled in the same way on all platforms and that favourites persist according to the same rules.

8.5 Operator Integration

The operator must ensure that Mood Shot is integrated according to this specification. Integration responsibilities include maintaining identity, managing sliders, managing presets, enforcing safety constraints, and ensuring that the system behaves consistently across updates.

The operator must monitor for integration errors, verify that sliders and presets select the correct text templates, and ensure that invalid or retired configurations cannot be selected. The operator must document integration decisions and maintain alignment with the architecture and safety requirements.

Section 9. Operational Telemetry and Monitoring

9.1 Telemetry Requirements

The system must collect telemetry that reflects the health and correctness of Mood Shot during operation. Telemetry must include signals related to slider changes, slider validation outcomes, preset selection, preset application, the text templates selected by modulation, continuity across backend transitions, and expiry at conversation end. Telemetry must not include user identity or content beyond what is required for operational monitoring.

Telemetry must be structured, consistent, and suitable for automated analysis. The system must ensure that telemetry does not expose sensitive information and complies with all privacy requirements.

9.2 Monitoring Signals

The system must monitor key signals that indicate whether Mood Shot is functioning correctly. Monitoring must include detection of slider application failures, preset application failures, continuity failures, invalid slider values, invalid preset usage, and attempts to apply text templates from retired or incompatible sliders or presets.

Monitoring must also track safety related events, including rejected sliders, rejected presets, rejected modulations, and conflicts between layers. The system must ensure that monitoring is active across all supported platforms and surfaces.

9.3 Alerting Thresholds

The system must define thresholds for alerting when operational conditions deviate from expected behaviour. Alerts must be generated when slider validation fails repeatedly, when preset validation fails repeatedly, when continuity failures exceed acceptable limits, or when text-template safety constraints are triggered at abnormal frequency.

Alerting must be routed to the operator responsible for maintaining Mood Shot. Alerts must be actionable and must include sufficient context to diagnose the issue without exposing user content.

9.4 Audit Logging

The system must maintain audit logs that record key operational events. Audit logs must include slider creation, slider updates, slider retirement, preset creation, preset updates,

preset retirement, preset selection, the text templates selected by sliders and presets, and validation outcomes for both sliders and presets. Logs must also record safety related decisions and conflict resolution events.

Audit logs must be retained according to operator policy and must be accessible for review. Logs must not contain user identity or conversation content beyond what is required to understand operational behaviour.

9.5 Integration Health Checks

The system must perform regular health checks to confirm that Mood Shot is integrated correctly across all platforms and services. Health checks must verify that identity, affect, sliders, and task are assembled in the correct order, that sliders and presets are applied consistently, and that favourites persist according to specification.

Health checks must detect integration drift, missing slider definitions, missing preset definitions, incorrect slider-to-template mappings, incorrect preset template mappings, and failures to expire affective modulation at conversation end. The operator must review health check results and correct integration issues promptly.

Section 10. Maintenance and Update Management

10.1 Update Procedures

The operator must apply updates to Mood Shot in a controlled and documented manner. Updates may include changes to sliders, slider ranges, slider-to-template mappings, presets, preset descriptions, preset parameters, safety rules, integration logic, or supporting infrastructure. All updates must be validated against the specification before deployment. See Section 0 for operator responsibilities overview.

Update procedures must ensure that identity remains unchanged, that safety constraints remain intact, and that sliders and presets continue to function as intended. The operator must verify that updates do not introduce conflicts between layers or alter the behaviour of existing sliders or presets without explicit intent.

10.2 Recalibration Requirements

The operator must recalibrate sliders and presets when model behaviour, service behaviour, or safety requirements change. Recalibration must ensure that slider values continue to select the correct text templates and that preset configurations continue to produce the intended interaction style within defined boundaries.

Recalibration must include testing for continuity, expiry, safety enforcement, and compatibility with identity. If a slider or preset cannot be recalibrated without violating constraints, it must be retired.

10.3 Deprecation and Retirement

The operator must deprecate and retire sliders or presets that no longer meet the requirements of this specification. Deprecation must be communicated through operator channels, and retired sliders or presets must not be available for selection.

The operator must ensure that favourites referencing retired sliders, presets, or text templates are handled according to policy. Retired configurations must be removed from all platforms and must not be reintroduced without full validation.

If a favourite references a retired slider or preset, the system must notify the user and revert to identity plus task until a valid preset is selected.

10.4 Compatibility Maintenance

The operator must maintain compatibility across versions of Mood Shot, including sliders, presets, integration logic, and safety rules. Compatibility maintenance must ensure that behaviour remains consistent across platforms and that updates do not introduce divergence.

The operator must verify that sliders, presets, and their associated text templates remain compatible with new versions of the model or service. If compatibility cannot be maintained, sliders or presets must be recalibrated or retired.

Mood Shot is a versioned, evolving specification. Operators must maintain compatibility across versions and migrate slider and preset definitions when required.

10.5 Operator Responsibilities

The operator is responsible for maintaining Mood Shot in accordance with this specification. Responsibilities include applying updates, performing recalibration, maintaining and retiring text templates, retiring sliders and presets, maintaining compatibility, and ensuring that safety constraints remain enforced.

The operator must document all maintenance actions and must review the system regularly to confirm alignment with the architecture, operational rules, and safety requirements. The operator must correct deviations promptly and must ensure that Mood Shot remains stable and predictable over time.

Section 11. Operator Procedures

11.1 Operator Roles and Responsibilities

The operator is responsible for managing Mood Shot in accordance with this specification. Responsibilities include creating sliders, updating sliders, maintaining and updating text templates, creating presets, reviewing presets, approving updates, retiring sliders, presets, and templates, and ensuring that all operational actions comply with safety and identity requirements. The operator must ensure that only authorised personnel perform actions that affect sliders, presets, safety rules, or integration logic. See Section 0 for operator responsibilities overview.

The operator must maintain familiarity with the architecture, operational rules, slider standards, preset standards, verification requirements, and safety constraints. The operator must ensure that all procedures are followed consistently across platforms and services.

11.2 Slider & Preset Lifecycle Procedures

The operator must follow defined procedures for creating, reviewing, approving, updating, and retiring sliders and presets. Slider creation must follow the structure defined in this specification and must include validation for safety, identity protection, and compatibility. Preset creation must follow the same requirements and must use only approved slider values.

Slider and preset updates must be reviewed and approved before deployment. Sliders or presets that no longer meet requirements must be retired The operator must ensure that favourites referencing retired sliders, presets, or text templates are handled according to policy and that retired configurations are removed from all platforms.

11.3 Change Management Procedures

The operator must follow controlled change management procedures for all updates to Mood Shot. Changes must be documented, reviewed, tested, and approved before deployment. Change management must ensure that updates do not introduce conflicts between layers or alter system behaviour in unintended ways.

The operator must verify that changes comply with safety constraints, identity protection rules, slider standards, preset standards, and template-mapping rules. Changes that fail validation must not be deployed.

11.4 Incident Response Procedures

The operator must follow defined procedures when incidents occur. Incidents include slider failures, preset failures, text-template selection failures, safety constraint triggers, integration errors, continuity failures, and unexpected behaviour. The operator must investigate incidents promptly, identify root causes, and apply corrective actions.

Incident response must include documentation of the event, analysis of contributing factors, and verification that corrective actions resolve the issue. The operator must ensure that incidents do not recur and that the system remains aligned with this specification.

11.5 Audit and Review Procedures

The operator must perform regular audits to confirm that Mood Shot remains aligned with the architecture, operational rules, slider standards, preset standards, safety constraints, and integration requirements. Audits must include review of sliders, presets, text templates, slider usage, preset usage, safety events, monitoring signals, and maintenance actions.

The operator must document audit findings and apply corrective actions when deviations are identified. Review cycles must ensure that Mood Shot remains stable, predictable, and compliant with this specification over time.

Section 12. Definitions and Reference Material

12.1 Glossary

Identity
The stable system identity that defines core characteristics. Identity does not change across presets, sliders, updates, or integrations.

Affect
The modulation layer that adjusts interaction style by injecting operator-supplied text templates selected through presets or sliders.

Preset
A predefined configuration representing a specific combination of slider values or behavioural parameters. Presets must comply with safety and identity constraints.

Slider
A discrete behavioural dimension represented as a ten-step numeric range. Each slider position selects a specific operator-supplied text template used for affective modulation.

Slider Dimension
A named behavioural axis that defines a specific trait or modulation parameter.

Slider Boundary
The permitted range of values for a slider dimension.

Slider Mapping
The mapping between slider values and the specific text templates they select.

Slider Profile
A set of slider values that together select the text templates defining a specific behavioural configuration.

Task
The immediate user request or objective. Task instructions must not alter identity or affect.

Continuity
The requirement that the selected text templates are re-injected across backend transitions within a single conversation.

Expiry
The system level requirement that affective modulation ends automatically when the conversation ends.

Safety Constraint
A rule that ensures all modulation remains within acceptable behavioural boundaries.

Operator
The authorised party responsible for maintaining, updating, and governing Mood Shot.

Backend Instance Change

A container level transition such as routing changes, shard changes, or container restarts.

Conversation Scoped State

State that persists only within a single conversation and expires when the conversation ends.

12.2 Layer Definitions

Identity Layer
The foundational layer that defines the system identity and its stable characteristics. It must not be altered by presets, sliders, or updates.

Affect Layer
The modulation layer that adjusts tone or style by injecting text templates selected through presets or sliders. It must remain bounded and must not override identity or safety.

Task Layer
The layer that contains the user's immediate request. It must remain isolated from identity and affect.

Safety Layer
The layer that enforces behavioural constraints. It overrides all other layers when conflicts occur.

12.3 Structural Requirements

Input Structure
Inputs must be assembled in the order identity, affect, and task. Each component must be distinct and must not contain instructions intended for another component.

Preset Structure
Presets must include a name, description, slider values that select text templates, boundaries, compatibility requirements, and version information. Presets must not include instructions that alter identity or safety.

Slider Structure
Sliders must include a dimension name, ten-step numeric range, default value, template mapping, and safety boundaries. Slider dimensions must be validated individually and in combination.

Validation Structure
Validation must confirm that presets and slider configurations comply with safety constraints, that inputs follow the required structure, and that no conflicts exist between layers.

12.4 Reference Tables

Structural Components
Identity layer
Affect layer
Task layer
Safety layer

Layer Components
Identity characteristics
Affective modulation parameters
Task instructions
Safety constraints

Preset Fields
Name
Description
Slider values (template selections)
Boundaries
Compatibility requirements
Version information

Slider Fields
Dimension name
Ten-step numeric range
Default value
Template mapping
Safety boundaries

Validation Steps
Structural validation
Safety validation
Slider range validation

Preset compatibility validation
Continuity validation
Expiry validation

Operator Responsibilities
Preset creation
Slider dimension management
Preset and slider review
Preset and slider approval
Preset and slider retirement
Incident response
Audit review

12.5 Document Conventions

Terminology
Terms are used consistently throughout this specification. Definitions in Section 12 take precedence over informal usage.

Formatting
Section numbers follow a hierarchical structure. Subsections provide detail without introducing new conceptual domains.

Naming
Preset names, slider dimensions, and structural components must follow the terminology defined in this section.

Normative Language Must indicates a mandatory requirement. Should indicates a recommended requirement. May indicates an optional requirement.

Appendix A - Broad but Non-Exhaustive Personality Traits

This appendix provides a structured catalogue of neutral behavioural dimensions that operators may use when defining Mood Shot presets. These are not moral categories, psychological diagnoses, or safety classifications. They are simply axes of communication style.

Operators may choose any subset, define ranges, or ignore categories entirely.

A.1 Communication Style Dimensions

For Mood Shot, each end of a dimension must be implemented as discrete text templates selected by slider positions, not as continuous behavioural effects.

- **Directness** - blunt to diplomatic
- **Formality** - casual to formal
- **Verbosity** - concise to expansive
- **Literalness** - literal to figurative
- **Pacing** - slow to rapid
- **Structure** - loose to highly structured
- **Clarity** - minimal detail to explicit detail
- **Emotional signalling** - flat to expressive
- **Humour** - none to playful
- **Pedagogy** - intuitive to step-by-step

A.2 Social Interaction Dimensions

These dimensions represent communication tones; each position must correspond to a specific text template rather than a continuous behavioural shift.

- **Warmth** - cool to warm
- **Encouragement** - neutral to supportive
- **Assertiveness** - passive to assertive
- **Empathy style** - factual to affective
- **Turn-taking** - free-flowing to highly regulated

- **Social explicitness** - implicit cues to explicit cues
- **Conflict handling** - deflective to confrontational

A.3 Cognitive Style Dimensions

Each cognitive style position must be implemented as a discrete text template selected by the slider.

- **Depth** - surface-level to deep reasoning
- **Abstraction** - concrete to abstract
- **Complexity** - simple to multi-layered
- **Analytical style** - intuitive to analytical
- **Creativity** - literal to imaginative
- **Cognitive load** - low-load to high-load
- **Problem-solving mode** - exploratory to decisive

A.4 Emotional Tone Dimensions

Emotional tone levels must be represented through explicit text templates rather than continuous emotional modulation.

- **Calmness** - neutral to soothing
- **Intensity** - low-energy to high-energy
- **Positivity** - neutral to upbeat
- **Reassurance** - minimal to high
- **Seriousness** - playful to serious
- **Motivation style** - gentle to high-drive

A.5 Instructional and Guidance Dimensions

Each instructional style must be expressed through operator-supplied text templates mapped to slider positions.

- **Teacher style** - coach to lecturer
- **Scaffolding** - minimal hints to guided steps

- **Error handling** - blunt to gentle
- **Explanation mode** - examples to theory
- **Goal orientation** - open-ended to outcome-driven

A.6 Professional Tone Dimensions

Professional tone levels must be implemented as discrete text templates rather than continuous tonal adjustments.

- **Corporate tone** - relaxed to highly professional
- **Compliance strictness** - flexible to rigid
- **Documentation style** - conversational to technical
- **Boundary signalling** - soft to explicit
- **Risk sensitivity** - low to high

A.7 Creativity and Expression Dimensions

Creative and expressive styles must be implemented through explicit text templates selected by slider values.

- **Narrative style** - factual to storytelling
- **Imagery** - plain to vivid
- **Playfulness** - minimal to high
- **Voice consistency** - flexible to rigid persona
- **Improvisation** - structured to spontaneous

A.8 Meta-Interaction Dimensions

Meta-interaction traits must be represented through discrete text templates rather than dynamic behavioural adaptation.

- **Self-monitoring** - low to high
- **Adaptability** - fixed to adaptive
- **Transparency** - minimal to explicit reasoning
- **User-led vs model-led** - reactive to proactive

- **Boundary firmness** - soft to strict

A.9 Interpersonal Distance Dimensions

These dimensions must be implemented as text templates that adjust tone only; they must not simulate relationships or emotional states. These dimensions describe how close or distant the assistant's communication style feels. They do not imply romance, sexuality, or personal relationships. They simply reflect the tone humans naturally use with different levels of familiarity. These dimensions describe communication tone only and do not simulate relationships.

- **Distance** - distant to familiar
- **Personalness** - impersonal to personal
- **Relational warmth** - formal to friendly
- **Self-disclosure style** - minimal to conversational
- **Engagement depth** - surface-level to deeper rapport
- **Social proximity** - stranger-mode to friend-mode

Appendix B - Suggested Preset Library

This appendix provides a set of suggested presets that operators may use as starting points when configuring Mood Shot. These presets are optional, non-exhaustive, and purely illustrative. They demonstrate how behavioural dimensions from Appendix A can be combined into coherent interaction styles. Operators may modify, extend, replace, or ignore these presets entirely.

Presets do not alter identity. They only modulate affect, tone, and communication style.

For Mood Shot, each preset corresponds to a set of slider values that select operator-supplied text templates; the descriptions below refer to the intended style of those templates, not continuous behavioural effects.

B.1 Service and Support Presets

Each preset in this section must be implemented as a set of slider values that select specific text templates corresponding to the described style.

- **Calm Support**

  Low intensity, high clarity, steady pacing, friendly but not familiar.

- **Technical Explainer**

  High structure, high clarity, low figurative language.

- **Professional Agent** High formality, low personalness, consistent tone, strict boundaries.

B.2 Educational Presets

Each preset in this section must be implemented as a set of slider values that select specific text templates corresponding to the described style.

- **Step-by-Step Tutor**

  Slow pacing, explicit scaffolding, high structure.

- **Conceptual Teacher**

  Medium pacing, high abstraction, example-driven explanations.

- **Revision Partner**

  Medium warmth, medium structure, high clarity.

B.3 Accessibility and Neurodiversity Presets

Each preset in this section must be implemented as a set of slider values that select specific text templates corresponding to the described style.

These presets do not simulate conditions. They simply adjust communication style to reduce overwhelm or increase clarity.

- **Low-Overwhelm Mode**

  Slow pacing, low sensory load, minimal figurative language.

- **High-Clarity Mode**

  Explicit structure, clear transitions, step-by-step reasoning.

- **High-Depth Mode**

  Dense reasoning, multi-layered explanations, high cognitive load.

B.4 Interpersonal Distance Presets

Each preset in this section must be implemented as a set of slider values that select specific text templates corresponding to the described style.

These presets control communication distance, not relationships.

- **Formal Distance**

  Impersonal, structured, low warmth.

- **Neutral Acquaintance**

  Balanced warmth, medium personalness.

- **Familiar Helper**

  Higher warmth, conversational tone, deeper conversational engagement while maintaining communication distance boundaries

B.5 Cognitive Style Presets

Each preset in this section must be implemented as a set of slider values that select specific text templates corresponding to the described style.

- **Analytical Mode**

  High structure, logic-first explanations.

- **Creative Mode**

High imagery, flexible structure, imaginative reasoning.

- **Minimal Mode**

  Very concise, low verbosity, direct answers only.

B.6 Developer and Technical Presets

Each preset in this section must be implemented as a set of slider values that select specific text templates corresponding to the described style.

- **Debugging Partner**

  High structure, explicit reasoning, step-by-step analysis.

- **Code Reviewer**

  High clarity, constructive critique style.

- **Architectural Advisor**

  High abstraction, systems-level reasoning.

B.7 Research and Evaluation Presets

Each preset in this section must be implemented as a set of slider values that select specific text templates corresponding to the described style.

- **Neutral Baseline**

  Minimal warmth, medium clarity, no stylistic embellishment.

- **High-Variance Stress Test**

  Rapid pacing, dense content, minimal scaffolding.

- **Controlled Warmth**

  Fixed warmth, fixed pacing, fixed verbosity.

B.8 Creative and Narrative Presets

Each preset in this section must be implemented as a set of slider values that select specific text templates corresponding to the described style.

- **Storyteller**

  High imagery, narrative structure.

- **Dry Historian**

  Factual tone, chronological structure.

- **Playful Mode**

  Light humour, flexible structure.

B.9 Notes on Preset Usage

- Presets are optional.
- Presets are suggestions, not defaults.
- Presets do not alter identity.
- Presets combine slider values that select text templates; they cannot be layered beyond selecting one set of templates at a time.
- Presets are not psychological profiles.
- Presets are not diagnoses.
- Presets are not safety categories.

Presets do not represent psychological traits or mental states. They are simply **bundles of behavioural dimensions** from Appendix A.

Appendix C - Use Cases Beyond Entertainment

Mood Shot is not limited to stylistic modulation or customer-facing entertainment. Its layered architecture enables a wide range of operational, educational, accessibility, and enterprise applications. The following categories illustrate how operators may apply Mood Shot in contexts where stable, predictable, or specialised interaction styles are required.

For Mood Shot, each use case is implemented through presets that select operator-supplied text templates; the descriptions below refer to the intended style of those templates, not continuous behavioural effects.

C.1 Age-Appropriate Interaction

Each preset in this category must be implemented as a set of slider values that select specific text templates corresponding to the described style.

Mood Shot can support services that require different communication styles for children, teens, adults, or mixed-age environments. Operators may define presets that adjust pacing, clarity, emotional signalling, or complexity to match developmental expectations.

Mood Shot does **not** infer or classify age. It simply allows operators to supply age-appropriate presets when the service requires them.

C.2 Spectrum-Appropriate Interaction

Each preset in this category must be implemented as a set of slider values that select specific text templates corresponding to the described style. These presets adjust communication tone only. They do not simulate cognitive states, emotional states, or neurotypes.

Mood Shot can adapt its communication style to support users with diverse cognitive, sensory, or processing profiles. This includes autistic users, ADHD users, dyslexic users, sensory-sensitive users, and others across the neurodiversity spectrum.

Mood Shot does not simulate or imitate any neurotype. It does not assign diagnoses. It does not classify traits as conditions.

Instead, operators may define presets - ideally in consultation with educators, clinicians, or neurodiversity specialists - that reduce overwhelm and increase accessibility. These presets may include:

- reduced ambiguity
- slower pacing
- predictable turn-taking
- low-sensory phrasing
- minimal figurative language
- explicit structure
- clear emotional signalling

This is not about “making the model autistic.” It is about making the model comfortable for autistic users and others who benefit from structured, predictable communication.

This category is intentionally broad. It does not list every learning difference or mental-health-related processing profile. It simply recognises that many users benefit from tailored interaction styles.

C.3 Cognitive-Load-Appropriate Interaction

Each preset in this category must be implemented as a set of slider values that select specific text templates corresponding to the described style.

Mood Shot can adjust its communication style for users who benefit from either reduced or increased cognitive load. This includes users who prefer:

- slower pacing and step-by-step reasoning
- or dense, high-bandwidth, concept-heavy explanations

Mood Shot does not equate cognitive load preferences with intelligence. High overwhelm does not imply low ability, and high cognitive demand does not imply superiority. These are simply different processing styles.

Operators may define presets that:

- reduce cognitive load for users who need clarity and structure
- increase cognitive load for users who want depth and complexity

This flexibility supports both accessibility and advanced learning environments.

C.4 Customer Support and Service Workflows

Each preset in this category must be implemented as a set of slider values that select specific text templates corresponding to the described style.

Mood Shot can provide consistent, stable interaction styles for:

- technical support
- billing assistance
- onboarding flows
- troubleshooting guides
- regulated-domain customer service

Affective presets ensure tone remains predictable across long sessions and backend instance changes.

C.5 Education and Tutoring

Each preset in this category must be implemented as a set of slider values that select specific text templates corresponding to the described style.

Mood Shot can maintain a stable teaching persona across:

- tutoring sessions
- curriculum-aligned explanations
- revision guides
- step-by-step problem solving
- differentiated instruction

Affective modulation allows the operator to choose a teaching style without rewriting identity.

C.6 Accessibility and Assistive Technology

Each preset in this category must be implemented as a set of slider values that select specific text templates corresponding to the described style.

Mood Shot can support users who benefit from:

- predictable pacing
- explicit structure
- reduced sensory load
- alternative communication styles

- simplified or expanded explanations

This includes users with temporary, situational, or permanent accessibility needs.

C.7 Enterprise Assistants

Each preset in this category must be implemented as a set of slider values that select specific text templates corresponding to the described style.

Mood Shot can maintain stable personas for:

- internal knowledge assistants
- workflow automation
- compliance-aligned communication
- long-running project support
- multi-team collaboration tools

Identity remains protected; affective modulation remains temporary.

C.8 Research and Evaluation

Each preset in this category must be implemented as a set of slider values that select specific text templates corresponding to the described style.

Mood Shot enables:

- reproducible behavioural conditions
- controlled affective modulation
- stable evaluation personas
- cross-model comparison
- safety-layer testing

Labs can test identity, affect, and task combinations without contamination.

Appendix D - Suggested User Sliders

This appendix provides a short, practical set of sliders that operators may choose to expose to end users. These sliders are optional, non-exhaustive, and intended to keep the interface simple. Each slider maps to multiple behavioural dimensions from Appendix A, allowing broad control without complexity.

For Mood Shot, each user-facing slider is implemented as a ten-step control where each position selects an operator-supplied text template; the descriptions below refer to the intended style of those templates, not continuous behavioural effects.

Sliders are required for operator implementations and are defined in Section 5. Appendix D provides an optional, simplified set of sliders intended for end users. These user facing sliders function as game style stat controls that adjust broad behavioural tendencies without exposing the full operator defined slider space. They are a reduced subset of the underlying dimensions and are presented for usability rather than architectural completeness.

D.1 Tone Slider

This slider must be implemented as ten discrete positions, each selecting a specific text template that expresses the described style.

- Tone Adjusts overall warmth, friendliness, and emotional expression. Range: cooler to friendlier.

D.2 Directness Slider

This slider must be implemented as ten discrete positions, each selecting a specific text template that expresses the described style.

- Directness Controls how straightforward or diplomatic responses feel. Range: blunt to gentle.

D.3 Pacing Slider

This slider must be implemented as ten discrete positions, each selecting a specific text template that expresses the described style.

- Pacing Adjusts response speed and density. Range: slower and spaced to faster and compact.

D.4 Detail Slider

This slider must be implemented as ten discrete positions, each selecting a specific text template that expresses the described style.

- Detail level Controls verbosity and depth. Range: short answers to detailed explanations.

D.5 Structure Slider

This slider must be implemented as ten discrete positions, each selecting a specific text template that expresses the described style.

- Structure Controls how organised or free-flowing responses are. Range: loose flow to step-by-step structure.

D.6 Creativity Slider

This slider must be implemented as ten discrete positions, each selecting a specific text template that expresses the described style.

- Creativity Adjusts imagery, playfulness, and flexibility. Range: plain to imaginative.

D.7 Distance Slider

This slider must be implemented as ten discrete positions, each selecting a specific text template that expresses the described style.

- Communication distance Controls how formal or familiar the assistant feels. Range: distant to friendly acquaintance.

D.8 Notes on Slider Usage

- Sliders are optional and operator defined.
- Sliders do not alter identity.
- Sliders do not imply psychological traits or conditions.
- Sliders can be renamed or combined, but each position must still map to a specific text template.
- Sliders are intended to be simple, intuitive controls for end users.

This list is intentionally short to reflect realistic product design.

-Document Ends-