

FSD_05 – Readiness Validator

Purpose:

This module performs a **final readiness check** before allowing an agent to go live. It verifies:

- All **critical discovery tags** are covered with high confidence
- Any **dynamic qualification questions** have been answered
- Uploaded content is **sufficient**, recent, and well-segmented
- Conflicts or contradictions are resolved or flagged
- The agent has enough data to respond **intelligently and safely**

Think of it as **Audira's internal compliance and QA layer** before greenlighting the agent.

FSD_05 – Section Breakdown:

Section	Description
1. Scope	What it validates and what happens if validation fails
2. Input Sources	Tag map, answers, document segments, metadata
3. Validation Rules	Coverage %, priority tags, segment recency, answer confidence
4. Scoring Engine	Readiness score calculation (pass/fail with reasons)
5. Output Schema	Report object used by dashboard, launch logic, or agent
6. Fail Cases & Routing	What happens if validator fails
7. Admin Overrides	Manual paths to force launch with justification
8. Future Enhancements	Self-healing logic, industry templates, auto-retraining triggers

Section 1: Scope

Purpose:

This module acts as a **final checkpoint** before any agent (or agent configuration) is launched. It ensures the AI agent has received enough input, clarity, and coverage to perform confidently, ethically, and meaningfully for the SMB it represents.

What the Validator Confirms:

- **Critical discovery tags** have been fully processed and are present with high confidence.

- **All AI-generated qualification questions** (post-upload or dynamic) have been answered or flagged.
 - **Input documents** are:
 - Properly segmented
 - Within the acceptable recency window (e.g., last 12 months)
 - Not overly sparse (e.g., <10% of expected segments)
 - **User-provided answers** do not conflict with document content (e.g., stated industry ≠ detected sector).
 - **Agent has enough meaningful data** to activate each assigned capability module (e.g., financial assistant, operations optimizer).
 - **No unresolved contradictions** or high-risk ambiguities exist across the onboarding set.
-

What Happens if Validation Fails:

If any of the above conditions are not met, the validator will:

- Prevent the agent from going live.
 - Generate a **readiness report** with:
 - **Reason codes** (e.g., MISSING_TAG:pricing_strategy)
 - **Recommended next actions** (e.g., “Re-answer dynamic Q4”)
 - **Suggested fallback modules** (e.g., partial activation of certain features).
 - Flag the case in the **Admin Review Dashboard** (cross-linked to FSD_09).
-

Why It Matters (Strategic Goals):

- Prevents premature launches that could damage trust or result in poor responses.
 - Enables **automated agent QA at scale** across thousands of SMBs.
 - Powers **adaptive training triggers** based on failure modes.
 - Allows **Audira partners to track readiness metrics** across industries and cohorts.
-

◆ Section 2: Input Sources

The Readiness Validator consumes structured outputs from earlier modules (FSD_01 → FSD_04) to assess whether the agent is ready to go live.

Primary Inputs:

Source	Description	Source FSD
Discovery Tag Map	Dictionary of all tags detected from answers and documents (including confidence scores, sources, and category groupings).	FSD_02

Segmented Document Blocks	All user-uploaded content after segmentation, labeling, and type classification (e.g., tables, paragraphs, invoices).	FSD_01 + FSD_02
Answer Matrix	User responses to fixed onboarding questions, dynamic questions, and AI-suggested clarifications. Includes timestamps and confidence metrics.	FSD_01 + FSD_04
Reference Link Map	Cross-references between document segments, tags, and answers, allowing semantic linkage across mixed input formats.	FSD_03
Qualification Completion Log	Flags whether each dynamic qualification question (post-tagging) has been answered, skipped, or ignored.	FSD_04
Agent Intent Profile	Summary of which modules or services the user expects the agent to perform (e.g., scheduling, finance, sales).	Audira Core Config
Metadata & Upload History	Includes timestamps, file types, upload frequency, doc source info (e.g., scanned vs. typed), and coverage stats.	FSD_01

Why This Matters:

These inputs form a **composite profile** of what the user said, uploaded, and implied — allowing the Readiness Validator to holistically assess whether the agent is **informed**, **conflict-free**, and **fit for purpose**.

◆ Section 3: Validation Rules

This section defines the **core logic** the validator uses to determine whether an agent has enough qualified, coherent, and up-to-date information to be approved for launch.

Coverage Rules:

Rule	Description	Threshold / Criteria
Minimum Tag Coverage	Required % of high-priority discovery tags must be present.	$\geq 85\%$ of required tags from agent intent
Critical Tag Match	Tags labeled as mandatory (e.g., <code>business_model</code> , <code>target_customer</code>) must exist.	All must be detected or answered
Tag Confidence Threshold	Tags must have confidence \geq defined level.	≥ 0.75 (on 0–1 scale)
Qualification Answer Rate	At least 90% of dynamic qualification questions must be answered.	$\geq 90\%$
Document Recency	At least one document must be uploaded within the last X months.	≤ 12 months old
Segment Density	Total segmented content should meet a volume threshold.	≥ 20 segments or $\geq 1,500$ tokens total

Answer-Segment Link	Each core answer must be linked to at least one document segment or inferred tag.	Must resolve to a source
Conflict-Free State	No open conflicts between answers and document data.	e.g., business type ≠ detected sector

Conflict Examples:

Conflict Type	Description	Example
Sector Mismatch	Claimed industry ≠ detected content type	Answer: “Retail” vs. doc says “SaaS startup”
Contradictory Answers	Different answers provided across sessions	“We use Stripe” vs. “We only accept wire”
Tag Overlap Conflict	Two tags with mutual exclusivity detected	“Subscription-based” + “One-time license”

Validation Scoring:

Each validation rule contributes to an **overall readiness score**, weighted by importance:

Component	Weight (%)
Critical Tag Presence	25%
Tag Confidence + Coverage	20%
Qualification Completion	15%
Document Freshness	10%
Segment Depth	10%
Conflict-Free Analysis	15%
Answer-to-Doc Linkage	5%

Passing Threshold: Readiness Score $\geq 80\%$ with no failed **critical** validations.

Section 4: Scoring Engine

This section explains how the validator **computes a readiness score**, determines **pass/fail status**, and outputs **debuggable trace data** for admin review or auto-healing logic.

Scoring Formula:

The final score is calculated using a **weighted aggregation** of validation rule results:

```
readiness_score = (
    tag_coverage_score * 0.20 +
    critical_tag_score * 0.25 +
    tag_confidence_score * 0.10 +
```

```

qualification_answer_score * 0.15 +
document_recency_score * 0.10 +
segment_depth_score * 0.10 +
conflict_score * 0.15 +
answer_to_segment_linkage_score * 0.05
)

```

Score Normalization (0–100 Scale):

Each sub-score is scaled as follows:

Sub-Score	Logic
<code>tag_coverage_score</code>	% of required tags detected (max 100)
<code>critical_tag_score</code>	Binary pass/fail → 100 or 0
<code>tag_confidence_score</code>	Avg confidence × 100
<code>qualification_answer_score</code>	% answered of dynamic questions
<code>document_recency_score</code>	100 if recent doc present, else 0
<code>segment_depth_score</code>	100 if segment volume ≥ threshold, else scaled
<code>conflict_score</code>	100 if no conflict, else penalty based on severity
<code>answer_to_segment_linkage_score</code>	% of core answers with doc/segment link

Pass/Fail Logic:

- **PASS:**
 - Score ≥ 80
 - AND no failed **critical validations** (e.g., `critical_tag_score = 0`)
- **FAIL – Retry Recommended:**
 - Score between 60–79
 - Minor gaps, conflicts, or incomplete qualifications
 - Auto-generated retry instructions provided
- **FAIL – Block Launch:**
 - Score < 60
 - Or any **critical validation failed**

Output Format:

The scoring engine returns a full `ReadinessReport` object:

```
{
  "agent_id": "AGENT_00231",
  "readiness_score": 82,
  "status": "PASS",
  "failed_areas": [],
  "suggested_fixes": [],
  "timestamp": "2025-06-11T00:45:00Z"
}
```

If failed:

```
{  
  "status": "FAIL",  
  "readiness_score": 52,  
  "failed_areas": [  
    "critical_tag_missing:pricing_strategy",  
    "segment_depth_insufficient"  
,  
  "suggested_fixes": [  
    "Re-answer onboarding Q7 (pricing)",  
    "Upload a business overview document"  
,  
  ]  
}
```

◆ Section 5: Output Schema

This section defines the structured data format emitted by the Readiness Validator, used by downstream systems such as:

- The **Admin Review Dashboard** (FSD_09)
- The **Launch Controller** (Audira platform core)
- The **Feedback Router** (dynamic agent tuning)
- The **Agent Memory Construction** logic (FSD_07)

Output Object: ReadinessReport

```
{  
  "agent_id": "AGENT_000231",  
  "business_name": "Nova Retail Solutions",  
  "readiness_score": 82,  
  "status": "PASS",  
  "timestamp": "2025-06-11T00:45:00Z",  
  "validated_by": "READINESS_VALIDATOR_V1",  
  "component_scores": {  
    "tag_coverage_score": 90,  
    "critical_tag_score": 100,  
    "tag_confidence_score": 82,  
    "qualification_answer_score": 100,  
    "document_recency_score": 100,  
    "segment_depth_score": 70,  
    "conflict_score": 95,  
    "answer_to_segment_linkage_score": 60  
,  
  "failed_areas": [],  
  "suggested_fixes": [],  
  "segment_recap": {  
    "recent_docs_found": true,  
    "total_segments": 22,  
    "segment_types": {  
      "paragraph": 12,  
      "table": 3,  
    }  
  }  
}
```

```

        "image": 2,
        "form": 5
    }
},
"unresolved_conflicts": [],
"qualified_tags": [
    "business_model:subscription",
    "industry:retail",
    "payment_provider:stripe"
],
"missing_critical_tags": [],
"next_steps": {
    "can_launch": true,
    "requires_admin_review": false,
    "auto_retry_suggested": false
}
}
}

```

Integration Targets:

Consumer Module	Usage
Admin Review Dashboard (FSD_09)	Displays detailed report for human QA or override
Launch Logic Controller	Uses <code>status</code> and <code>can_launch</code> to approve/block launch
Dynamic Qualification Engine (FSD_04)	May auto-generate new questions if gaps are found
Agent Update Module (FSD_10)	Stores report for future retraining or delta updates
Scoring Aggregator (Audira Metrics)	Tracks trends across cohorts, industries, or partners

Section 6: Fail Cases & Routing

This section defines **what happens when validation fails** and how the system intelligently routes each case for resolution, retry, or manual override.

Fail Case Categories:

Code	Description	Example
<code>MISSING_CRITICAL_TAG</code>	One or more mandatory tags not present	<code>MISSING_CRITICAL_TAG:pricing_strategy</code>
<code>LOW_QUALIFICATION_SCORE</code>	Too many unanswered dynamic questions	User skipped 4/10 post-upload questions
<code>DOCUMENT_GAPS</code>	No recent docs or insufficient segments	Only 1 doc uploaded from 2019

CONFLICT_DETECTED	Detected contradiction in answers/tags	Business model ≠ product pricing approach
INSUFFICIENT_LINKAGE	User answers not linked to doc content	No traceable source for legal structure
LOW_READINESS_SCORE	Composite score < threshold (e.g., < 60)	Total score = 54 with 3 weak areas

Routing Logic:

Each fail case triggers an automated routing path:

Scenario	Action	Destination
Minor failure (score 60–79)	Auto-recommend retry fixes	Back to User
Major failure (<60)	Block launch + admin alert	Admin Review Dashboard (FSD_09)
Conflict detected	Trigger AI contradiction resolver	Conflict Parser Module (future)
Unanswered questions	Regenerate with fallback logic	Dynamic Qualification Engine (FSD_04)
System error	Capture traceback for dev logs	Internal Audit Queue

Suggested Fixes:

The validator outputs **custom next-step guidance** for each failure case:

```
{
  "failed_areas": [
    "MISSING_CRITICAL_TAG:distribution_model",
    "DOCUMENT_GAPS:recency"
  ],
  "suggested_fixes": [
    "Answer onboarding question Q3 (Distribution Model)",
    "Upload a recent overview doc (< 12 months old)"
  ]
}
```

Admin Routing Flags:

Flag	Meaning
<code>requires_admin_review: true</code>	Sent to dashboard for human QA
<code>auto_retry_suggested: true</code>	User can attempt to resolve automatically

`launch_blocked: true`

Agent activation prevented until override or fix

❖ Section 7: Admin Overrides

This section defines how **authorized Audira admins or certified partners** can manually override a failed readiness state — with full traceability, justification logging, and downstream implications.

❖ *Override Triggers:*

Manual override may be permitted in these scenarios:

Scenario	Allow Override?	Notes
Known low-signal industry	<input checked="" type="checkbox"/>	E.g., sectors with sparse documentation (e.g., verbal coaching)
Temporary data source gaps	<input checked="" type="checkbox"/>	If user has acknowledged and committed to submit later
Conflict explained by user	<input checked="" type="checkbox"/>	Admin accepts user justification for apparent mismatch
Auto-retry failed due to edge case	<input checked="" type="checkbox"/>	E.g., multi-brand business models not handled well by validator
System error or misclassification	<input checked="" type="checkbox"/>	Debug logs support override
Missing critical tag with admin note	Only with escalation	Requires internal approval or audit trail

✓ *Override Workflow:*

1. Admin clicks “Override & Launch” from Review Dashboard (FSD_09)
2. Prompted to enter:
 - o Justification Note (required)
 - o Confidence Level (Low / Medium / High)
 - o Agent Capability Warnings (optional toggle per module)
3. System logs:
 - o Admin ID + Timestamp
 - o Pre override report snapshot
 - o Post override launch state
4. ReadinessReport.status is updated to FORCED_PASS

✍ *Example Audit Log Entry:*

```
{  
  "agent_id": "AGENT_00231",
```

```

"override_by": "admin_kareem",
"timestamp": "2025-06-11T01:12:00Z",
"original_score": 68,
"final_status": "FORCED_PASS",
"justification": "Agent is for niche B2B vertical with no public docs. Verified via call.",
"confidence_level": "High"
}

```

Risk Flagging:

If override is used, agent is flagged in backend systems:

- **Visible to partner dashboards**
 - **Auto-tagged for performance monitoring**
 - May trigger soft caps on what the agent can promise (e.g., disable financial prediction features)
-

Section 8: Future Enhancements

This section outlines planned or optional upgrades that can expand the validator's capabilities over time, making it more intelligent, scalable, and self-improving.

1. Self-Healing Logic

Automatically attempts to resolve gaps using AI-powered inference or clarification loops.

Feature	Description
Gap Filler	Suggests answers to missing tags/questions by scanning uploaded documents
Auto-Clarify	Initiates a mini-follow-up with the user via chat if a critical tag is missing
Tag Inferencer	Uses LLMs to fill low-confidence tags if confidence is borderline (e.g., 0.65–0.74)

2. Industry-Specific Validator Templates

Allow the readiness validator to load predefined logic per industry or business archetype.

Industry	Example Logic
E-commerce	Enforce pricing, shipping, return policy, and product catalog structure
Professional Services	Focus on service scope, compliance docs, and client acquisition channels
Healthcare	Require licenses, compliance formats, consent protocols

These templates can also feed into downstream scoring tolerances or required tag lists.

3. Agent Retraining Trigger

When readiness fails or changes over time (due to new uploads), this enhancement allows:

- **Auto-revalidation** after key documents are uploaded
 - Triggering a **rebuild of the prompt chain** or memory (FSD_06 / FSD_07)
 - Notifying admins of repeated failures for a single agent
-

4. LLM-Enhanced Conflict Resolver (v2)

Integrates with Audira's in-house contradiction detector to provide **explanations and suggestions**, not just flags.

Feature	Description
Explain Conflict	"The user selected 'sole proprietor' but uploaded incorporation docs."
Ask for Clarification	Automatically message user to explain or fix contradiction
Resolve with Confidence	If explanation is clear, the system may resolve and retry launch check

5. Validator Performance Dashboard

Track validator usage, pass/fail rates, false positives, override frequency, and improvement opportunities by segment, geography, or partner.
