# AI Agent Design for Schengen Tourist Visa Workflow

This document translates your requirements into a production-ready set of AI agents, responsibilities, interfaces, rules, and example prompts. It shows exactly **where** to add each agent inside your application and how they interoperate.

## 0) End-to-End Flow (Where Agents Fit)

|  |
| --- |
| 1 digraph G {  2 rankdir=LR;  3 subgraph cluster\_intake {label="Applicant Intake"; style=dashed; color=lightgrey;  4 A1["Applicant Portal / API Upload"];  5 A2["Pre-Normalization Service\n(OCR, virus-scan, unicode, EXIF)"];  6 }  7 subgraph cluster\_verify {label="Verification"; style=dashed; color=lightgrey;  8 V1["Tourist Visa Verification Agent\n(Interactive)"];  9 }  10 subgraph cluster\_comms {label="Partner Communications"; style=dashed; color=lightgrey;  11 C1["Schengen Partner Communications Agent"];  12 }  13 subgraph cluster\_decide {label="Decision Support"; style=dashed; color=lightgrey;  14 D1["Decision Agent\n(Recommendation only)"];  15 }  16 subgraph cluster\_coord {label="Operations"; style=dashed; color=lightgrey;  17 R1["Coordinator Agent\n(Assignment & SLA)"];  18 H1["Human Officer Console\n(HITL)"];  19 }  20  21 A1 -> A2 -> V1 -> D1 -> H1;  22 V1 -> C1 [label="needs partner intel?"];  23 C1 -> D1 [label="PartnerIntel"];  24 D1 -> R1 [label="risk & priority"];  25 V1 -> R1 [label="blocking issues -> priority bump"];  26 H1 -> R1 [label="reassign/queue mgmt"];  27 }  28 |

**Placement summary**

* **Pre‑Normalization** sits right after upload to clean/sanitize raw inputs.
* **Verification Agent** runs next; interactive UI shows issues to applicant for quick fixes.
* **Comms Agent** is invoked on demand (e.g., to verify bookings, check alerts).
* **Decision Agent** compiles evidence and proposes a recommendation for officer review.
* **Coordinator Agent** runs continuously to assign and rebalance cases.

## 1) Agent Charters & IO Contracts

### 1.1 Tourist Visa Verification Agent (Interactive)

**Purpose:** Ensure application is complete, machine-safe (legacy systems), and internally consistent.

**Input:** ApplicationBundle (JSON) + uploaded files registry.

**Checks**

* **Presence**: all required fields to apply for a visa.
* **Special characters**: strip or flag characters unsupported by legacy police/partner systems; preserve originals in \*\_raw fields.
* **Formats**: dates ISO-8601, emails RFC-valid, phone E.164, country codes ISO-3166, currency ISO-4217.
* **Document readiness**: file type, non-zero bytes, no password, readable PDF, basic OCR signal.
* **Consistency**: arrival/departure order; hotel coverage matches duration; return flight present; insurance covers entire stay; name consistency across documents.

**Output:** VerificationReport

* readyForDecision: boolean
* missingFields[], formatIssues[], riskyChars[], docFindings[], consistencyFindings[]
* blockingIssues[] (stops decision) and autoFixes[] (safe normalizations)
* normalizedPayload (optional) for downstream use

### 1.2 Schengen Partner Communications Agent

**Purpose:** Draft, send, and track structured, auditable communications with Schengen partners.

**Input:** Case context, templates library, partner directory/endpoints, VerificationReport highlights.

**Responsibilities**

* Compose standardized requests (e.g., hotel verification, ticket validation, alerts follow-up).
* Apply redaction policy and legal basis tags (GDPR purpose limitation).
* Maintain comms\_log (timestamps, content hashes, attachment ids).
* Summarize inbound responses into structured PartnerIntel.

**Output**

* PartnerMessage (draft + metadata), PartnerResponseSummary, PartnerIntel objects.

### 1.3 Decision Agent (Recommendation Only)

**Purpose:** Produce a transparent recommendation (Grant / Refuse / Request More Info) with a scorecard and rationale.

**Signals**

* **Funds sufficiency**: bank statements vs duration & accommodation; sponsor coverage if declared.
* **Police report**: presence & status.
* **Travel evidence**: return/onward tickets, date alignment.
* **Accommodation**: coverage across stay, name consistency.
* **Schengen intel**: prior violations/alerts.
* **Authenticity & consistency**: MRZ, file integrity, cross-doc name/date matching.

**Output:** DecisionProposal

* recommendedOutcome, scorecard (per factor), riskFlags[], blockingIssues[], explanations[], and policy\_refs[]

### 1.4 Coordinator Agent (Bonus)

**Purpose:** Assign cases to officers/queues using risk, SLA, language, specialization, and workload.

**Input:** Case metadata, DecisionAgent signals, Verification blocking issues, team roster & WIP.

**Output:** AssignmentPlan with priority, assignee/queue, due\_time, reasons[].

**Policies**

* High-risk or MANUAL\_REVIEW → senior adjudicator queue.
* Language/region matching; conflict-of-interest avoidance.
* Load balancing with max WIP per officer and SLA weighting.

## 2) Minimal JSON Schemas

|  |
| --- |
| 1 // ApplicationBundle  2 {  3 "applicationId": "uuid",  4 "submittedAt": "2025-10-27T11:53:00Z",  5 "applicant": {  6 "fullName": "string",  7 "dob": "YYYY-MM-DD",  8 "nationality": "ISO-3166-1 alpha-2",  9 "passportNumber": "string",  10 "passportExpiry": "YYYY-MM-DD",  11 "contact": {"email": "string", "phone": "+31123456789"}  12 },  13 "travel": {  14 "arrivalDate": "YYYY-MM-DD",  15 "departureDate": "YYYY-MM-DD",  16 "purpose": "tourism"  17 },  18 "accommodation": [{"from": "YYYY-MM-DD", "to": "YYYY-MM-DD", "name": "string"}],  19 "financials": {  20 "currency": "EUR",  21 "bankStatements": [{"fileId": "uuid", "periodFrom": "YYYY-MM", "periodTo": "YYYY-MM"}]  22 },  23 "documents": {  24 "policeReport": {"fileId": "uuid", "issueDate": "YYYY-MM-DD"},  25 "tickets": [{"fileId": "uuid", "type": "return", "pnr": "string"}],  26 "insurance": {"fileId": "uuid", "coverageAmount": 30000}  27 },  28 "attachments": [{"fileId": "uuid", "fileName": "string", "mime": "application/pdf"}]  29 }  30 |

|  |
| --- |
| 1 // VerificationReport  2 {  3 "applicationId": "uuid",  4 "readyForDecision": false,  5 "missingFields": [{"path": "applicant.passportNumber", "reason": "Required"}],  6 "formatIssues": [{"path": "travel.arrivalDate", "expected": "YYYY-MM-DD"}],  7 "riskyChars": [{"path": "applicant.fullName", "value": "J\u200bohn", "action": "strip\_zero\_width"}],  8 "docFindings": [{"fileId": "uuid", "issue": "passwordProtected"}],  9 "consistencyFindings": [{"msg": "Hotel coverage 3/5 nights"}],  10 "blockingIssues": [{"code": "PASSPORT\_EXPIRES\_TOO\_SOON"}],  11 "autoFixes": [{"path": "applicant.fullName", "fixedValue": "John", "note": "Removed zero-width char"}]  12 }  13 |

|  |
| --- |
| 1 // DecisionProposal  2 {  3 "applicationId": "uuid",  4 "recommendedOutcome": "RequestMoreInfo",  5 "scorecard": {  6 "financialSufficiency": 0.8,  7 "travelEvidence": 0.95,  8 "policeClearance": 1.0,  9 "accommodationCoverage": 0.7,  10 "schengenIntel": 0.9,  11 "docAuthenticity": 0.85  12 },  13 "riskFlags": [{"code": "RECENT\_LARGE\_INFLOW", "severity": "medium"}],  14 "blockingIssues": [],  15 "explanations": [  16 "Funds meet 80% of expected threshold; hotel coverage missing for 1 night; return confirmed; police report clear."  17 ],  18 "policy\_refs": ["POL-FUNDS-1.3", "POL-TRAVEL-2.0"]  19 }  20 |

|  |
| --- |
| 1 // AssignmentPlan  2 {  3 "applicationId": "uuid",  4 "priority": "HIGH",  5 "assignee": {"userId": "officer\_023", "name": "A. Janssen"},  6 "due\_time": "2025-10-28T16:00:00+01:00",  7 "reasons": ["Manual review recommended", "Travel date < 7 days", "Hotel coverage gap"]  8 }  9 |

## 3) Rule Sets (ready to codify)

### 3.1 Verification Rules

* **Required fields**: fullName, dob, passportNumber, passportExpiry, nationality, arrivalDate, departureDate, tickets[return], insurance.fileId, bankStatements[]\*, \*accommodation[].
* **Dates**: ISO-8601; arrivalDate < departureDate; passport valid ≥ 3 months beyond departure.
* **Special characters**: reject control chars U+0000–U+001F (except \n, \t when permitted); strip zero-width; normalize Unicode **NFKD**; for machine fields keep ASCII fallback while preserving original in \*\_raw.
* **Consistency**: hotel covers ≥ 90% of nights (configurable); tickets’ names & dates match passport and itinerary; insurance coverage spans full stay with minimum medical coverage (e.g., €30,000—configurable).
* **Documents**: accept PDF/JPEG/PNG; size ≤ policy; not password-protected; DPI ≥ 200 for scans; quick OCR check for text density.

### 3.2 Decision Logic (hybrid: rules + scoring)

**Hard blocks**

* Missing passport/return ticket/insurance.
* Passport validity below threshold.

**Scoring (weights example)**

* Financial sufficiency **35%**
* Police & partner intel **25%**
* Travel evidence **15%**
* Accommodation consistency **10%**
* Document authenticity **10%**
* History/compliance **5%**

**Outcome**

* Any hard block → RequestMoreInfo or Refuse (per policy).
* Score ≥ 0.85 → Grant; 0.65–0.84 → RequestMoreInfo; < 0.65 → Refuse.

## 4) Prompts (starter)

**Verification Agent (system)**

|  |
| --- |
| 1 You validate visa applications for completeness, formatting, and cross-field consistency.  2 Return ONLY JSON per VerificationReport schema. Do not invent values. Flag special characters and mixed-script risks.  3 |

**Comms Agent (system)**

|  |
| --- |
| 1 You draft formal, standardized communications to Schengen partners using approved templates and only case facts.  2 Redact extraneous PII and include legal basis tags. Output a PartnerMessage JSON plus a human-readable draft.  3 |

**Decision Agent (system)**

|  |
| --- |
| 1 You produce a DecisionProposal using the defined hard rules and weights. Provide per-factor rationale and cite data points used. You do not make the final legal decision.  2 Return DecisionProposal JSON only.  3 |

**Coordinator Agent (system)**

|  |
| --- |
| 1 Assign cases to queues/officers by priority, language, specialization, and workload. Return AssignmentPlan JSON with reasons and SLA.  2 |

## 5) Implementation Notes

* **Pre-Normalization**: do Unicode NFKD; transliteration for machine fields; virus scan; PDF password detection; light OCR for MRZ and tickets.
* **RAG for Comms**: index partner templates, internal policy, and previous approved comms. Hard-block sending without citations.
* **Explainability**: persist feature values, prompts, and model responses in an **immutable audit log** with hashes.
* **PII & GDPR**: field-level encryption for passport/bank data; purpose tags on data access; retention timers; no raw PII in model training.
* **HITL**: required for outbound partner comms and for all final decisions.

## 6) Minimal Pseudocode (wireframe)

|  |
| --- |
| 1 # verify(bundle) -> VerificationReport  2 # decide(bundle, verification, partner\_intel) -> DecisionProposal  3 # coordinate(case, decision, roster) -> AssignmentPlan  4 |

## 7) Next Steps / Backlog

1. **Confirm thresholds**: daily funds requirement, insurance minimum, passport validity rule.
2. **Provide sample ApplicationBundle(s)** to test Verification Agent.
3. **Collect partner templates** and delivery channels (API/email/portal) for the Comms Agent.
4. **Define team roster & SLA** for Coordinator routing.
5. Build a **sandbox** with synthetic cases to measure precision/recall of verification and agreement rate of decisions.

## 8) What I Need from You

* Your current data model (field names you already store).
* Policy documents or the key thresholds you want encoded.
* Preferred stack (Python/FastAPI, Node/Nest, .NET) so I can provide concrete code scaffolds.
* Any on‑prem / data residency constraints.

Once you share one anonymized sample application, I’ll run through:

* a generated VerificationReport,
* a DecisionProposal, and
* a routing AssignmentPlan tailored to your roster.