Synthetic Users AI/GenAI Algorithm Design Document & Data Flow Diagram

Document ID: CRA-6.1.1-ADDD-001

Version: 1.0

Effective Date: March 25, 2026

Last Updated: March 25, 2026

Owner: CTO — Artur Ventura

Approved By: CEO — Kwame Ferreira

Classification: Internal – Confidential

CRA Control: CRA 6.1.1

---

1. Purpose

This document describes the algorithm design, system architecture, and data flow of Synthetic Users' AI/GenAI platform. It explains how prompts are constructed, how AI-generated responses are processed, how the system distinguishes between user-provided data and AI-generated content, and how customer and JPMC data is protected throughout the AI inference pipeline.

This document is produced in response to JPMC Security Controls Assessment (SCA) control CRA 6.1.1.

---

2. System Architecture Overview

The Synthetic Users platform is built on a four-layer architecture:

Layer	Components	Purpose
Presentation Layer	Web application (Next.js), REST/GraphQL API	Customer-facing interface; study configuration and output delivery
Orchestration Layer	LLM Shuffle, Persona Engine, Research Agents	AI feature coordination, model routing, persona construction
AI Inference Layer	Third-party LLM APIs (OpenAI, Anthropic, Google Gemini, Mistral)	Language model inference; response generation
Data Layer	PostgreSQL (Render), AWS S3, Vector Store (RAG index)	Persistent storage; document retrieval; session data

No AI model weights are hosted by Synthetic Users. All LLM inference is performed via API calls to third-party providers.

---

3. Core AI Components

3.1 LLM Shuffle — Multi-Provider Orchestration

LLM Shuffle is Synthetic Users' proprietary model routing layer. Rather than relying on a single language model, LLM Shuffle distributes inference requests across multiple frontier LLM providers. This architecture provides:

- **Bias reduction** — different models exhibit different biases; using multiple models reduces systematic skew
- **Output diversity** — varied model responses produce richer synthetic participant sets
- **Resilience** — automatic failover if a provider is unavailable or rate-limited
- **Vendor independence** — no lock-in to a single AI provider

Model selection for each inference call is determined by study configuration, participant persona type, and real-time provider availability.

3.2 Persona Engine — OCEAN Personality Architecture

The Persona Engine constructs psychologically grounded synthetic participant profiles using the **OCEAN model** (Openness, Conscientiousness, Extraversion, Agreeableness, Neuroticism) — the leading empirically validated framework in personality psychology.

Each synthetic participant is assigned:

- OCEAN trait scores (derived from demographic and psychographic inputs)
- Role, background, and professional context
- Behavioral anchors specific to the study domain
- Consistent response tendencies that persist across a full simulated interview

Persona definitions are injected into the LLM system prompt as structured context, ensuring consistent character fidelity throughout the session.

3.3 RAG Pipeline — Retrieval-Augmented Generation

For studies requiring grounding in specific domain knowledge or customer-provided context, the platform uses a Retrieval-Augmented Generation (RAG) pipeline:

- **Ingestion:** Customer-provided documents are ingested, chunked, and embedded into a tenant-scoped vector index
- **Retrieval:** At inference time, semantically relevant chunks are retrieved and injected into the prompt context window
- **Isolation:** Each tenant's vector index is strictly isolated — no cross-tenant retrieval is possible
- **No persistence of JPMC data:** JPMC-specific context is provided in the prompt context window only and is not persisted to the shared RAG index

3.4 Research Agents

Research Agents are AI-assisted workflow components that help customers design effective studies. They assist with question formulation, audience definition, and study structure — but all final decisions remain with the human researcher. Agent outputs are clearly labelled as AI suggestions.

---

4. Prompt Construction

Prompt construction follows a deterministic, layered structure:

```
[System Prompt]
  └─ Platform instructions (role, safety constraints, output format)
  └─ Persona definition (OCEAN profile, background, behavioral anchors)
  └─ Study-specific instructions (domain, interview type, question set)

[Context Window]
  └─ RAG-retrieved chunks (if applicable, scoped to tenant)
  └─ Conversation history (current session only)

[User Turn]
  └─ Current interview question (researcher-authored)
```

Key safeguards in prompt construction:

- System prompt contents are never exposed in model outputs
- User-supplied text is injected as data, not as system instructions, to mitigate prompt injection
- Maximum token limits are enforced per prompt component
- No JPMC data is included in system prompts shared across tenants

5. AI Data Flow

5.1 Data Flow — Standard Inference Request

```
Customer (browser)
    → REST API (authenticated, TLS 1.2+)
    → Orchestration Layer (LLM Shuffle + Persona Engine)
    → Prompt Constructor
```

```
    → LLM Provider API (TLS 1.2+, no data retained by provider per DPA)
    → Response Post-Processor (output sanitisation, format validation)
    → API Response
    → Customer (browser)
```

No intermediate storage of prompt or response content occurs outside of the active session. Session data is stored in the customer's tenant-scoped database partition.

5.2 Data Flow — RAG-Enabled Request

```
Customer document upload
    → Document Ingestion Worker (sandboxed)
    → Chunking + Embedding (provider API call)
    → Tenant-Scoped Vector Index (isolated per customer)

At inference time:
    → Query Embedding
    → Tenant Vector Index Retrieval (scoped, no cross-tenant access)
    → Retrieved chunks injected into prompt context window
    → Standard Inference Flow (as above)
```

5.3 What Is and Is Not Sent to LLM Providers

Data Category	Sent to LLM Provider?	Notes
Persona definition (OCEAN parameters, role)	Yes	Injected as system prompt context
Study questions (researcher-authored)	Yes	Part of user turn
RAG-retrieved document chunks	Yes	Injected as context window content
Customer PII	No	Never included in prompts

JPMC data	Only if explicitly scoped to JPMC study	Covered by provider DPA
Platform API keys or credentials	No	Never included in prompts
Other tenants' data	No	Strict tenant isolation enforced

6. AI-Generated vs. User-Provided Data — Distinction

The platform maintains a clear architectural separation between user-provided data and AI-generated content:

Data Type	Source	Storage	Labelling
Study configuration (questions, audience)	Human researcher	Customer database partition	Stored as `type: researcher_input`
Persona parameters	Human researcher + system defaults	Customer database partition	Stored as `type: persona_config`
AI-generated interview responses	LLM inference	Customer database partition	Stored as `type: ai_generated`, flagged in UI
Research reports and knowledge graphs	AI synthesis of AI responses	Customer database partition	Prominently labelled "AI-Generated Synthesis"
Customer-uploaded documents (RAG)	Human / customer	Tenant vector index	Stored as `type: customer_document`

All AI-generated outputs presented in the user interface are visually labelled as AI-generated content. The platform does not present synthetic outputs as human

responses.

7. LLM Selection Logic

Model selection within LLM Shuffle is governed by the following priority order:

1. **Study-level override** — if the researcher has selected a specific model for the study
2. **Persona-type routing** — certain persona types perform better with specific model families (configured internally)
3. **Provider availability** — real-time health check; unavailable providers are skipped
4. **Load balancing** — requests are distributed across available providers to avoid rate limit saturation
5. **Fallback** — if all preferred providers are unavailable, a designated fallback provider is used

8. Output Safeguards

All LLM responses pass through a post-processing layer before being returned to the customer:

- **Format validation** — response structure is validated against expected schema
- **Content filtering** — responses are screened for content policy violations
- **PII scanning** — outputs are scanned for unexpected PII patterns (e.g., real names, contact details)
- **Injection detection** — outputs are checked for patterns indicating prompt injection success
- **Length enforcement** — responses outside expected length bounds are flagged for review

9. AI/GenAI Data Protection & Subprocessors

All LLM providers used by Synthetic Users are engaged under Data Processing Agreements (DPAs) that prohibit:

- Training or fine-tuning on API request data
- Retention of API request and response content beyond the request lifecycle
- Secondary use of customer data for any purpose

For the full list of AI/GenAI subprocessors, see Subprocessors & Data Flow.

10. GenAI Lifecycle

Lifecycle Stage	Synthetic Users Practice
Model selection	Security review + DPA validation before production enablement
Integration	SAST + code review; no hardcoded credentials
Testing	Adversarial prompt testing; data isolation testing; output safety review
Production monitoring	Output anomaly detection; latency/error rate monitoring
Model updates	Lightweight re-validation; CTO sign-off for default model switch
Retirement	Migration testing; provider data deletion confirmation

For full AI/GenAI lifecycle requirements, see the SDLC AI/GenAI Addendum.

11. Related Documents

- SDLC AI/GenAI Addendum
- Responsible AI & Risk Management Overview
- AI Safety & Bias Mitigation Controls
- Subprocessors & Data Flow
- Third-Party Risk Management Policy
- Information Governance & Records Management Standard

Synthetic Users, Inc. — 4223 Glencoe Ave, Suite C215-523, Marina del Rey CA 90292