ShopperArmy Survey API Integration

Product Requirements Document

Project: External Survey Inventory System
Version: 1.0
Date: December 30, 2024
Owner: Andrew Abony

Executive Summary

ShopperArmy currently creates missions manually through an admin dashboard. We want to expand our survey inventory by integrating with external survey API providers (starting with Morning Consult). This requires building infrastructure to pull survey inventory from third-party APIs, store it, and display it for review—with a future phase to auto-deploy surveys as missions.

Problem Statement

Today: A team member logs into ShopperArmy admin → manually creates a mission → defines targeting, copy, reward → publishes it.

Opportunity: Survey API providers like Morning Consult have thousands of surveys available at any time, with defined targeting and pay rates. We can tap into this inventory to dramatically increase earning opportunities for our users—without manually creating each mission.

Challenge: These APIs work differently than our current system. They provide a live, constantly-changing feed of available surveys. Quotas fill and close. Pay rates and targeting are pre-defined by the survey buyer. We need infrastructure to ingest, track, and eventually deploy this inventory.

Solution Overview

Build a **Survey Inventory System** that:

1. Connects to external survey APIs (starting with Morning Consult)
2. Pulls available survey inventory on a scheduled basis
3. Stores survey data in our database

4. Displays inventory in an admin dashboard with full qualification details
5. (Phase 2) Auto-creates missions from eligible surveys

Phase 1: Survey Inventory Dashboard

Objective

Create visibility into what external surveys are available at any time, with all targeting and economic details, before we commit to auto-deploying them.

Functional Requirements

1.1 API Connection Layer

Purpose: Connect to Morning Consult (and future providers) and pull survey data.

Requirement	Details
API Authentication	Store and manage API credentials securely (Bearer tokens, API keys)
Provider Abstraction	Build a provider interface so we can add Lucid, Cint, etc. later without rewriting core logic
Rate Limiting	Respect API rate limits (Morning Consult: 100 requests/min)
Error Handling	Log failures, retry with backoff, alert on persistent failures

Morning Consult Specifics:

- Base URL (Sandbox): `https://sample-api-sandbox.morningconsult.com/v1`
- Base URL (Production): `https://sample-api.morningconsult.com/v1`
- Auth: Bearer token in header
- Primary endpoint: `GET /supplier/bids` returns available surveys with nested quota details

1.2 Data Storage

Purpose: Store survey inventory locally for querying and historical tracking.

Core Tables:

```
survey_providers
- id
- name (e.g., "morning_consult")
- api_base_url
- credentials (encrypted)
- is_active
- created_at, updated_at

external_surveys
- id
- provider_id (FK)
- external_bid_id (provider's ID)
- name
- country
- topic
- survey_url_base
- length_of_interview_seconds
- published_at
- expires_at
- incidence_rate
- is_active
- raw_json (store full API response)
- created_at, updated_at
- last_synced_at

survey_quotas
- id
- external_survey_id (FK)
- external_quota_id
- cpi_cents (what we earn)
- completes_required
- completes_current
- is_open
- raw_json
- created_at, updated_at

quota_qualifications
- id
- survey_quota_id (FK)
- qualification_type (e.g., "age", "gender", "household_language", "cbsa2020")
- qualification_values (JSON array of accepted values)
- created_at
```

```
qualification_legend
- id
- provider_id (FK)
- qualification_type
- response_id
- response_text (human-readable label)
- created_at
```

1.3 Sync Job (Cron)

Purpose: Keep local inventory fresh.

Setting	Value
Frequency	Every 60 minutes
Scope	Pull all active bids, update existing records, mark closed surveys
Logging	Log each sync: surveys added, updated, closed
Alerts	Notify if sync fails 3+ times consecutively

Sync Logic:

1. Call `GET /supplier/bids` with pagination
2. For each bid: upsert survey record, upsert quota records, upsert qualifications
3. Mark surveys as inactive if no longer returned by API
4. Update `last_synced_at` timestamp

1.4 Admin Dashboard UI

Purpose: Let team view available survey inventory.

Main View: Survey Inventory Table

Column	Description
Provider	Morning Consult, Lucid, etc.
Survey ID	External bid ID
Topic	General, Finance, Tech, etc.
Country	US, UK, etc.
Length	Survey duration in minutes
CPI Range	$1.00 - $4.00 (min to max across quotas)
Total Slots	Sum of completes_required across quotas
Filled	Sum of completes_current
Fill %	Progress toward full
Quotas	Count of quota segments
Status	Open / Closed
Expires	End date
Last Sync	When we last pulled data

Filters:

- Provider
- Country
- Status (Open/Closed)
- CPI range (min)
- Length range
- Date range (published/expires)

Detail View (click into a survey):

Show all quotas with decoded qualifications:

```
Survey: Ad-Hoc Survey (9c094c11-b640-4424-a09f-d3c786a3de80)
Provider: Morning Consult
Length: 24.6 minutes
Expires: Jan 25, 2026

QUOTAS:
┌──────────┬────────┬─────────┬───────────────┬──────────┬──────────┬───────
│ Quota    │ Gender │ Age     │ Language      │ CPI      │ Needed │ Filled │
├──────────┼────────┼─────────┼───────────────┼──────────┼──────────┼───────
│ quota-1 │ Male   │ 18-34  │ English only  │ $3.50   │ 250   │ 12    │
│ quota-2 │ Male   │ 18-34  │ Bilingual     │ $4.00   │ 250   │ 8     │
│ quota-3 │ Male   │ 35-44  │ English only  │ $3.50   │ 250   │ 15    │
│ ...    │ ...  │ ...  │ ...         │ ...    │ ...  │ ...  │
└──────────┴────────┴─────────┴───────────────┴──────────┴──────────┴───────
```

Qualification Legend View:

Separate page showing the mapping of codes → human labels for each provider:

```
MORNING CONSULT - QUALIFICATION LEGEND

Gender:
  1 = Male
  2 = Female

Household Language:
  1 = English only
  2 = Spanish only
  3 = Both equally
  4 = Mostly English, some Spanish
  5 = Mostly Spanish, some English

Age:
  [ID equals actual age, 18-99]

CBSA (Metro Areas):
  12060 = Atlanta-Sandy Springs-Alpharetta, GA
  33100 = Miami-Fort Lauderdale-Pompano Beach, FL
  ...
```

Non-Functional Requirements

Requirement	Target
Sync reliability	99%+ successful syncs
Dashboard load time	< 2 seconds
Data freshness	< 90 minutes old
Multi-provider ready	Architecture supports adding providers without schema changes

---

Phase 2: Auto-Deploy Missions (Future Scope)

Objective

Automatically create ShopperArmy missions from external survey inventory.

High-Level Requirements (to be detailed later)

1. **Eligibility Rules Engine**
    - Define which surveys to auto-deploy (min CPI, max length, allowed topics)
    - Exclude surveys that don't meet quality thresholds
2. **Mission Creation**
    - Map external survey to ShopperArmy mission format
    - Set user reward (e.g., 50% of CPI)
    - Generate mission copy from survey metadata
    - Set targeting based on quota qualifications
3. **User Matching**
    - Match ShopperArmy user profiles to quota qualifications
    - Pre-filter which surveys each user sees
    - Call provider's eligibility endpoint before showing survey
4. **Session Tracking**
    - Generate signed survey URLs for each user
    - Track: started, completed, screened out, quality terminated

    - Handle callbacks/webhooks from provider

5. **Reconciliation & Payment**
    - Match completions to earnings
    - Credit users for completed surveys
    - Reconcile with provider invoices
6. **Ed25519 URL Signing**
    - Implement cryptographic signing required by Morning Consult
    - Secure key storage and rotation

API Reference Summary

Morning Consult - Key Endpoints

Endpoint	Method	Purpose
`/supplier/bids`	GET	List available surveys with quotas
`/supplier/eligibility`	POST	Check if panelist qualifies
`/supplier/sessions`	GET	List/track sessions
`/lookup/countries/{country}/qualifications`	GET	Get qualification code mappings

Key Data Concepts

Term	Definition
Bid	A survey opportunity
Quota	A targeting segment within a bid (e.g., "Males 18-34")
CPI	Cost Per Interview - what provider pays us per complete (in cents)
LOI	Length of Interview - survey duration in seconds
Qualification	Demographic attribute (age, gender, language, geography)

Term	Definition
Session	A user's attempt at a survey

Success Metrics

Phase 1

- Dashboard live and displaying real inventory
- Sync job running reliably every hour
- Team can view available surveys and quota details
- Legend decodes all qualification codes to readable labels

Phase 2 (Future)

- X surveys auto-deployed per day
- Y% of users matched to relevant surveys
- Z completions per week
- Revenue per complete tracking

Open Questions

1. **Revenue split:** What percentage of CPI do we pass to users? (e.g., $4.00 CPI → user gets $2.00?)
2. **Quality thresholds:** Minimum CPI or maximum LOI to auto-deploy? Morning Consult surveys are 20-40 min at $3-4 CPI.
3. **Provider priority:** After Morning Consult, which providers to integrate next? (Lucid, Cint, Theorem Reach, etc.)
4. **User profile enrichment:** Do we need to collect additional demographics from users to match more quotas? (Currently may not have household_language)

Appendix

A. Sample API Response Structure

```json
{
  "bid_id": "9c094c11-b640-4424-a09f-d3c786a3de80",
  "name": "Ad-Hoc Survey",
  "country_id": "us",
  "topic": "general",
  "length_of_interview": 1476,
  "published_at": "2025-12-26T00:00:00Z",
  "end_date": "2026-01-25",
  "quotas": [
    {
      "quota_id": "quota-1",
      "cpi": 350,
      "num_respondents": 250,
      "completes": 12,
      "qualifications": [
        {"id": "age", "response_ids": ["18","19"..."34"]},
        {"id": "gender", "response_ids": ["1"]},
        {"id": "household_language", "response_ids": ["1"]}
      ]
    }
  ]
}
```

B. Files Available

- `morning_consult_inventory.csv` - Sample inventory snapshot
- `morning_consult_summary.csv` - Summary statistics
- `morning_consult_qualifications.txt` - Qualification mapping reference
- `Sample_API_Docs.pdf` - Full Morning Consult API documentation

Timeline Estimate

Phase	Scope	Estimate
Phase 1	API connection + DB + Admin UI	2-3 weeks

Phase	Scope	Estimate
Phase 2	Auto-deploy + User matching + Tracking	4-6 weeks

Document prepared for Linear project intake.