Real-time software for shared phone experiences

StellerSight Private Limited

Product, engineering and validation evidence - 9 October 2026

What the company is building

StellerSight develops software products and technology services. This evidence note focuses on Aura, an independent phone-based crowd display, and the adaptive synchronization work in SyncPlayer. It describes implemented behavior and recorded checks; it does not assert worldwide novelty, patent rights, paying-customer traction or crowd-scale performance.

Aura: coordinated light using participating phones

A host creates an event and shares an invitation. Guests join anonymously and their screens render coordinated effects locally against a shared server timeline. Organized events use checked row/seat assignments. Free-crowd events use shared effects, with experimental camera mapping for stationary groups. The native Android app also supports nearby discovery and optional BLE proximity experiments.

SyncPlayer: adaptive playback synchronization

The implementation separates clock estimation, correction policy and transport. It uses drift estimates, network-quality signals and graduated rate/seek corrections to coordinate playback. This provides technical context for the company's real-time engineering work; exact visible or audible synchronization depends on devices, media players and network conditions.

Application positioning

The evidence supports an ordinary startup recognition application based on product development and improvement of real-time interaction workflows. The software can be reused across separate events, with potential for product commercialization and skilled engineering, testing and deployment work. Venue capacity and business outcomes require further validation.

Company identity

Indian private limited company. Incorporated 17 April 2026. CIN U62099MR2026PTC475675. Registered-office location: Panvel, Raigad, Maharashtra. The legal identity is consistent across the incorporation certificate, approved e-MOA, MSME records and live NSWS profile. This product note does not replace those official documents.

How Aura works

1. Host and invitation

Authenticated hosts create persistent events. Guest invitations identify the event separately from host account credentials.

2. Time and local rendering

Phones estimate server time using a four-timestamp exchange. Ordinary commands are scheduled at least 1.2 seconds ahead; each phone renders from the shared timeline.

3. Participation and recovery

The server validates participant actions and recipient-specific spark catches. Versioned event state and remembered assignments support rejoining.

4. Explicit placement

Organized events check duplicate seats. Free-crowd Mosaic requires measured LightMap placement; arrival order is not treated as a physical location.

5. Optional research mode

Spatial Lab collects consented compact BLE/motion summaries, supports measured ground truth and exports experiment data. It does not infer precise floor-plan coordinates.

Implemented differentiation

The integration combines timing calibration, host control, persistent event recovery, checked seating and server-confirmed audience interactions. This improves the workflow beyond independent color-screen operation. It is an engineering and product-development claim, not a completed prior-art or market-exclusivity finding.

Operational boundary

The current pilot uses one API process. The organized seating allocation is at most 512, and the namespace connection cap is 600. These are configured limits, not demonstrated simultaneous physical-device capacity. Large venues require network planning, a representative-device rehearsal, measured screen-to-screen skew and battery checks.

Recorded development and checks

18 Sep 2026 - Spark interaction release

17 interaction/timing/render checks; 69 existing checks; isolated 24-client API/socket scenario with 24 distinct catches. The live browser report records nine checks with three anonymous guest browser contexts and no JavaScript errors.

24 Sep 2026 - Independent Aura event workflow

Persistent events, anonymous joining and standalone studio. Recorded synthetic 20-client protocol rehearsal is software behavior evidence, not a 20-phone field trial.

4 Oct 2026 - Product guide review

The guide documents calibrated timing, experimental optical mapping, host control, reconnect behavior and the distinction between synthetic checks and physical-device measurements.

7 Oct 2026 - Organized / free-crowd release 2.3.0

Release note records 37 backend, nine native and 16 clock/render/optical/evaluator checks, builds and native analysis. Deployed checks included duplicate-seat rejection, rejoin and free-crowd Mosaic guard. Creation screens were checked on one USB phone.

What these checks establish

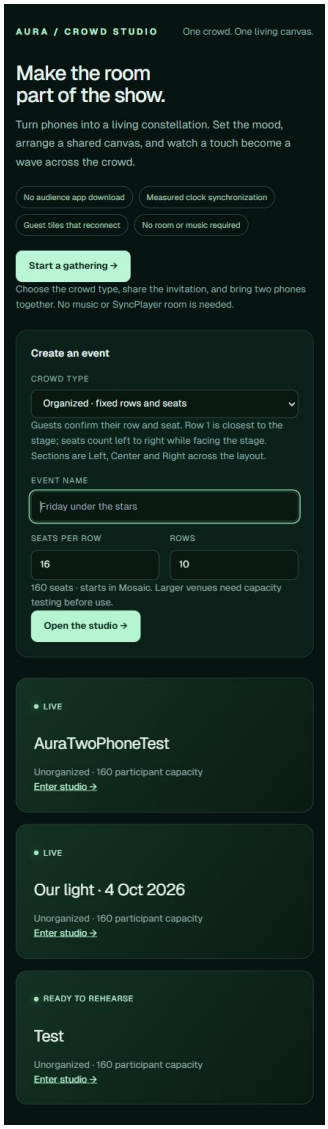
Recorded checks support claims about implemented workflows, concurrency rules, persisted participation, blackout rendering and release behavior. This note reviews existing evidence; it does not imply every test was rerun for the recognition application.

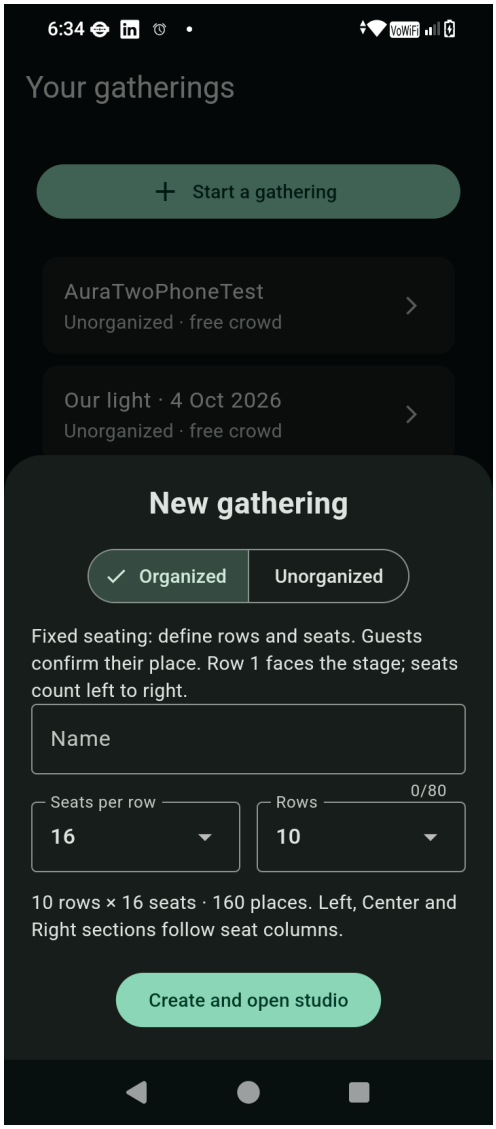
What still needs measurement

Mixed-device physical screen-to-screen timing, radio reliability, optical scanning across a real group, battery cost and venue-scale capacity remain open. BLE signal-strength evidence is not a measured distance or secure proof of presence. The latest release note records outstanding dependency-audit findings; the pilot is not described as fully security-audited.

Implemented Android event screens

Aura 2.3.0 screens captured on 7 October 2026. These show implemented controls, not audience-size or performance evidence.





Organized seating and free-crowd workflows

Organized events use rows and seats with one rectangular block. Guests confirm their organizer-assigned seats, and duplicate assignments are rejected. Free-crowd events start with shared effects. Camera mapping is a controlled experiment for visible stationary phones; continuous moving-crowd mapping and arbitrary uploaded image/video mosaics remain future work.

Evidence map and development path

Product and release scope

AURA-EVENTS.md; AURA-LIVE.md; AURA-SPARK-RELEASE.md; AURA-SPATIAL-LAB.md; Aura Guide to Shared Phone Light Experiences.docx.

Implementation

backend/src/aura/aura-live.service.ts; aura-proximity.service.ts; aura-spatial-lab.service.ts; backend/src/sync/sync-clock.ts, sync-policy.ts and sync-transport.ts; frontend/app/aura/page.tsx and native Android client.

Machine-readable recorded checks

scripts/aura-spark-integration-results.json (24 simulated participants, 24 distinct catches); scripts/aura-spark-live-results.json (nine live browser checks, three guest browser contexts, no JS errors).

Legal identity

Certificate-Of-Incorporation.pdf; approved e-MOA and e-AOA; msme.pdf and msme-a.pdf. Official documents are retained unchanged and supplied separately where required.

Public product access

https://stellersight.com/aura - product explainer. https://sync.stellersight.com/aura - host application; sign-in and an event invitation are needed for relevant flows.

Potential to scale

Reusable event configuration, invitations and locally rendered programs provide a route to repeatable deployment across organizers. Growth would require paid-product validation, representative venue tests, support procedures and a shared socket/state design for multiple API instances. Potential skilled roles include product engineering, mobile development, QA and event integration; no employment or revenue forecast is asserted.

IP and ownership scope

An engineering patent dossier is present in the records, dated before incorporation, but it is a draft for counsel review. No patent application number or filing receipt is established by that dossier. This note claims neither granted/filed IP nor exclusive ownership of every component. Third-party packages and open-source components remain subject to their licences.

Endpoint Lab: practical API learning

Private working prototype | Evidence reviewed 9 October 2026

Endpoint Lab helps learners build, test and repair sandbox applications through actual requests and responses. Its account-scoped API worlds, custom API builder and mission engine bring instruction, execution, failure diagnosis and evidence-based grading into a single workflow.

Implemented engineering

The inspected source binds sandbox state and stored request evidence to authenticated owners. Mission grading examines method, path, response status and ordered recovery evidence. Lab Studio provides versioned blueprints, request histories, deterministic replay, isolated branches, controlled fault setup, expiring attempt-scoped keys and instructor assignments. Workflow reports are distinguished from server-verified mission completion.

Learning and integration surfaces

The project documents lessons and assessments, CRUD sandboxes, OpenAPI import/export, generated small clients, bounded request workflows, teaching OAuth/PKCE, WebSocket echo and delivery-recovery exercises. Controlled teaching credentials do not replace platform sign-in. Optional AI coaching is off and is not required for the learning or grading workflows.

Why this supports the application

The integration improves practical software training by connecting executable API behavior with verified objectives and recoverable exercises. It fits the approved MOA provisions for software/SaaS development, technology education and training, and R&D. It complements Aura and SyncPlayer as another documented software initiative; it is not evidence of a patent, customer adoption or educational efficacy.

Endpoint Lab: validation and boundaries

Recorded checks, reviewed rather than rerun

STUDIO_RELEASE.md records nine functional suites with 1,004 checks, including ownership/roles/concurrency, HTTP authentication/CSRF, missions, contracts, studio workflows and recovery. It records a local restore of 13 application tables and a bounded local HTTP smoke test. These are recorded local development results, not production capacity or independent certification.

Release-stage boundary

The referenced development chat and pilot playbook describe an owner-private deployment. Independently authenticated hosted learner/instructor sessions, alert delivery, backup scheduling, hosted staging recovery and learner pilots remain launch dependencies. No paying customers, completed learning-outcome trial, external identity provider, real VPN infrastructure or live commercial billing are asserted.

Commercial direction

Potential commercialization uses learner or team subscriptions, instructor workspaces and training/support engagements. Prices, paid demand and measurable learning outcomes require validation. The private learning site remains private; this note makes no change to its audience or access controls.

Evidence sources

Referenced Codex chat: Build interactive API learning lab, thread 01a0f5e2-ab0a-79e1-be94-cd836163540b. Local project: D:/api/endpoint-academy. Files reviewed: README.md, STUDIO_RELEASE.md, PILOT_PLAYBOOK.md, package.json, lib/mission-engine.ts and lib/lab-studio-api.ts.

Disclosure boundary

This public supplement describes architecture and stage. It excludes account data, credentials, private request payloads, learner records and internal access configuration.