Court Scout — Case Study

Portfolio:https://www.arjunportfolio.xyz **Product:** https://courtscout.vercel.app

---

SECTION 1 — The Problem: Coordination Friction Prevents Play

Booking a tennis court in London should be simple.
In practice, it often isn't.

My experience — which ultimately motivated this project — was spending significant time jumping between multiple council websites, each with unintuitive UX, inconsistent availability displays, and different booking flows, just to find a single open slot.

Even after checking several sites, I would often miss available courts because:

- availability refreshed unpredictably
- slots opened and closed without notification
- interfaces made it hard to scan options quickly

This friction wasn't limited to booking itself.
It made **planning** the hardest part of playing.

The Real Problem Isn't Booking — It's Coordination

At first glance, this looks like a booking UX issue.
But through repeated attempts and observation, the deeper problem became clear:

> **Coordinating time, location, and availability across fragmented systems creates enough uncertainty that games often don't happen at all.**

This coordination burden shows up in several ways:

- Players repeatedly refresh different websites hoping a slot opens
- Groups struggle to find a court that is reasonably convenient for everyone
- Plans fall apart when availability changes unexpectedly
- The effort required to book outweighs the motivation to play

The result is a paradox:
Courts exist, players exist — but sessions still fail to materialize.

Fragmentation Amplifies Cognitive Load

Unlike many consumer booking experiences, tennis courts in London are managed by:

- different councils
- different operators
- different booking platforms

There is no single source of truth for availability.

For users, this means:

- mentally tracking multiple systems
- remembering which sites to check
- rechecking availability manually
- managing uncertainty without feedback

Over time, this cognitive overhead discourages spontaneous play and makes recurring games harder to sustain.

Why This Matters

This problem has real downstream effects:

- Fewer games get played
- Casual players disengage
- Coordination falls onto the most motivated person in the group
- Planning becomes a barrier instead of a facilitator

What should be a lightweight, enjoyable activity becomes an administrative task.

This project explores how product design can reduce coordination friction — not by changing how courts are run, but by changing how players experience availability, timing, and planning.

SECTION 2 — Who This Affects: Behavioral Segmentation

The coordination challenges described in Section 1 do not affect all tennis players in the same way.
Rather than segmenting users by age, skill level, or frequency of play, I focused on **behavioral patterns** — how people actually plan and coordinate games.

This revealed distinct segments with different pain intensities and expectations.

Segment 1 — Planned-with-Friends Players

These players:

- Play regularly with a fixed group of friends
- Often aim for recurring weekly or bi-weekly sessions
- Value convenience and predictability over novelty

Primary pain: Finding a court that is:

- available at the same time
- reasonably accessible for all players

Even when everyone agrees on a time, the lack of aggregated availability means:

- one person repeatedly checks multiple sites
- availability changes break plans
- coordination becomes a recurring burden

Behavioral insight:
This group experiences *coordination fatigue* — not because booking is impossible, but because repeating the process every week is mentally taxing.

Segment 2 — Semi-Social Players (Friends + Strangers)

These players:

- Use platforms like Spin, RacketPal, or WhatsApp groups

- Mix known friends with new or semi-known players
- Are more flexible on location, but less tolerant of uncertainty

Primary pain:
Unreliable planning.

Because players may not know each other well:

- trust in the booking process matters more
- last-minute changes create friction
- unclear availability leads to drop-offs

Behavioral insight:
For this segment, certainty is more valuable than optionality.
A reliable signal that a court is available is often enough to unlock commitment.

Segment 3 — Event-Driven Players

These players:

- Organize one-off games, meetups, or small tournaments
- Care about logistics more than routine
- Are often coordinating larger groups

Primary pain:
Logistical complexity.

They must account for:

- time windows
- multiple locations
- player availability
- court availability simultaneously

Behavioral insight:
This segment is highly sensitive to coordination failure.
If availability is unclear, events are postponed or canceled entirely.

Cross-Segment Pattern

Despite differences in motivation and frequency, all segments share a common constraint:

> **They are forced to mentally coordinate across fragmented systems with no reliable feedback loop.**

Across segments:

- discovery is not the problem
- intent already exists
- coordination and timing are the bottlenecks

This insight guided product decisions toward:

- reducing uncertainty
- surfacing availability passively
- minimizing the need for manual checking

SECTION 3 — Why This Problem Is Hard: Constraints & Tradeoffs

At a surface level, the problem of booking a tennis court looks solvable through better UI or faster search.
In reality, several structural constraints make this problem difficult to address in a clean, centralized way.

Understanding these constraints was critical to shaping both the MVP scope and the product strategy.

1. Fragmented Supply with No Shared Infrastructure

Tennis courts in London are managed by:

- different local councils
- private operators
- community organizations

Each operates:

- its own website
- its own booking flow
- its own availability logic
- its own update cadence

There is no standardized API, schema, or integration layer.

Product implication:
A “single source of truth” does not exist by default. Any attempt to present unified availability must reconcile inconsistent data and refresh cycles.

2. Inconsistent and Unreliable Availability Signals

Availability information varies widely across sites:

- some update in near real-time
- others lag by minutes or hours

- some expose partial availability
- others hide slots behind login flows

From a user's perspective:

- a visible slot may already be gone
- a missing slot may open moments later
- refreshing becomes habitual but unreliable

Product implication:
Trust becomes a core design concern.
If users don't believe availability signals are accurate, the product loses value immediately.

3. Legal, Ethical, and Platform Constraints

Unlike first-party booking platforms, an aggregator must operate carefully:

- respect website terms of service
- avoid actions that resemble automated booking
- ensure data is read-only and non-invasive
- prevent misuse or unfair access

Product implication:
The product cannot simply "optimize" for power users at the expense of sustainability.
Design decisions must balance usefulness with responsible behavior.

4. Coordination Involves Multiple Uncertain Variables

Booking a court is not a single-variable optimization problem.

Players must coordinate:

- time availability

- geographic convenience
- number of players
- court availability
- last-minute changes

Even small uncertainty in one variable cascades into failure.

Product implication:
The product must reduce *uncertainty*, not just increase *options*.
Surfacing more choices without clarity increases cognitive load rather than reducing it.

5. Trust and Reliability Matter More Than Feature Depth

In consumer coordination tools:

- one failed experience erodes confidence
- missed alerts feel worse than no alerts
- unreliable data discourages repeat use

Product implication:
Reliability and clarity must come before advanced features.
A simpler but trustworthy signal is more valuable than a complex but fragile one.

6. Adoption Depends on Minimal Behavior Change

Players already coordinate through:

- WhatsApp
- Spin
- existing routines

Any solution that requires:

- new habits

- repeated manual setup
- constant attention faces high drop-off.

Product implication:
Value must be delivered passively:

- aggregated views instead of manual checks
- alerts instead of constant monitoring
- minimal setup to see benefit

Summary: Why This Is a Product Problem, Not Just a Technical One

These constraints reveal that the core challenge is not building a scraper or a booking engine.

It is designing a system that:

- works across fragmented infrastructure
- earns user trust
- reduces uncertainty
- supports real-world coordination
- delivers value without demanding behavior change

These realities guided the product toward a narrow but high-impact MVP focused on **availability aggregation and alerts**, rather than full automation or heavy workflow changes.

SECTION 4 — Status Quo & Competitive Landscape

Before defining a solution, I looked closely at how players currently solve this problem — both manually and through existing tools.

Rather than evaluating competitors purely on features, I focused on **how well each approach supports real-world coordination**.

The Status Quo: Manual Coordination

For most players, booking a court today involves some combination of:

- Manually checking multiple council or operator websites
- Refreshing pages repeatedly to catch newly opened slots
- Sharing screenshots or links in WhatsApp groups
- Relying on one person in the group to “handle booking”

This approach has several drawbacks:

- High time investment for a low success rate
- No notifications when availability changes
- Poor visibility into alternative options
- Fragile coordination if plans shift

Key insight:
Manual checking technically works, but it externalizes the cost of coordination onto the most motivated person in the group.

Existing Platforms: Partial Solutions

Several products touch parts of the tennis ecosystem, but none fully address the coordination problem.

1. Social Tennis Platforms (e.g., Spin, RacketPal)

These platforms are effective at:

- finding players
- organizing matches
- facilitating introductions

However, they typically:

- rely on external court booking systems
- do not aggregate real-time court availability
- leave booking and logistics to users

Gap:
Discovery is solved, but **planning and execution are not**.

2. Individual Court Booking Websites

Council and operator websites provide:

- official booking access
- up-to-date availability for a single location

But they:

- vary widely in UX quality
- lack comparison across locations
- require repeated manual checking
- offer no coordination or alerting features

Gap:
Each site works in isolation, forcing users to mentally stitch information together.

3. Niche Aggregation Attempts

Some community-built or niche tools attempt to:

- scrape availability
- present it in a single view

However, these often suffer from:

- poor UX
- limited reliability
- lack of alerts
- unclear trustworthiness

Gap**:**
The problem is acknowledged, but execution often fails at usability and confidence.

Why Existing Solutions Fall Short

Across all approaches, a consistent pattern emerges:

- Availability is fragmented
- Signals are unreliable or delayed
- Coordination remains manual
- Trust is fragile

Most solutions optimize for **information access**, not **coordination outcomes**.

> Users don't just want to *see* slots — they want confidence that planning around those slots will actually work.

The Opportunity

This landscape suggested a clear opportunity:

- Not to replace booking platforms
- Not to become a social network
- Not to automate everything

But to act as a **coordination layer** that:

- aggregates availability across sources
- reduces uncertainty through alerts
- helps groups converge on a viable plan

This insight shaped the product direction toward **aggregation + notification**, rather than deeper automation or platform replacement.

SECTION 5 — Value Hypothesis

The research and analysis in the previous sections pointed to a consistent conclusion:

The primary barrier to playing tennis was **not discovery, pricing, or willingness to play** — it was **coordination under uncertainty**.

Players already knew:

- where courts were located
- who they wanted to play with
- roughly when they were available

What they lacked was a **reliable way to know when a viable option actually existed** — without constantly checking multiple systems.

Core Value Hypothesis

> **If players can see aggregated court availability and receive timely alerts when slots open, they will spend less time coordinating and significantly increase the likelihood that games actually happen.**

It assumes that:

- reducing uncertainty is more valuable than increasing options
- passive signals (alerts) outperform active monitoring (manual checking)
- trust and reliability matter more than automation depth

What “Value” Means in This Context

For this product, value is not measured by:

- number of searches
- time spent browsing
- feature usage

Value is measured by:

- fewer failed plans
- faster convergence on a bookable slot
- reduced coordination effort
- increased confidence that planning will succeed

In other words, **successful play is the primary outcome**.

Why Aggregation Alone Is Not Enough

Simply aggregating availability addresses only part of the problem.

Without alerts:

- users must still monitor changes
- cognitive load remains high
- coordination still breaks when timing shifts

The hypothesis therefore depends on **both**:

1. a unified view of availability

2. a mechanism to notify users when conditions change

Together, these form a feedback loop that replaces repeated manual checking with **trustworthy signals**.

MVP-Level Assumptions Embedded in the Hypothesis

To validate this hypothesis, the MVP implicitly tests several assumptions:

- Players are willing to set alerts instead of constantly checking
- Alerts are perceived as reliable enough to act on
- Aggregated availability is “good enough” even if not perfectly real-time
- Reducing coordination friction leads to higher follow-through

These assumptions guided scope decisions and success metrics, rather than being treated as abstract risks.

Why This Hypothesis Was Worth Testing First

Among many possible directions — full booking automation, social coordination tools, or scheduling optimizers — this hypothesis was chosen because it:

- targets the most acute pain
- requires minimal behavior change
- avoids legal and ethical risks
- can be validated with a narrow MVP
- creates immediate user value without deep integration

This made it the fastest and safest way to learn whether the coordination problem could be meaningfully reduced.

SECTION 6 — Solution Exploration

With a clear value hypothesis in place, I explored multiple solution directions to reduce coordination friction and improve booking outcomes.

The goal at this stage was not to design the most comprehensive system, but to identify which approach best balanced:

- user value
- reliability
- adoption friction
- feasibility under real-world constraints

Option A — Manual Coordination Improvements (Guides, Templates, Checklists)

Description
Lightweight guidance to help users coordinate more efficiently:

- recommended booking strategies
- shared checklists
- suggested court shortlists

Pros

- No technical complexity
- No data reliability risk
- Zero legal or platform concerns

Cons

- Still requires manual checking

- Does not reduce uncertainty
- Places burden back on users
- Low leverage over the core problem

Assessment
Improves awareness but does not meaningfully change outcomes.
Rejected due to low impact on coordination friction.

Option B — Social Coordination Layer (Chats, Polls, Group Planning)

Description
A coordination-first product focused on:

- group chats
- availability polling
- shared planning workflows

Pros

- Addresses social coordination directly
- Familiar interaction patterns
- Potential for network effects

Cons

- Does not solve fragmented availability
- Adds another coordination surface
- Requires behavior change
- High complexity for early validation

Assessment
Solves symptoms, not the root cause.
Rejected due to high friction and indirect value.

Option C — Full Booking Automation

Description
Automatically monitor availability and book courts on behalf of users when conditions are met.

Pros

- Maximum convenience
- Clear user value if reliable
- Strong differentiation

Cons

- Legal and ToS risks
- High trust requirements
- Fragile reliability
- Complex error handling
- Difficult to launch responsibly

Assessment
High risk and over-scoped for an MVP.
Rejected despite strong theoretical appeal.

Option D — Aggregated Availability + Alerts (Chosen MVP)

Description
A coordination layer that:

- aggregates court availability across sources
- presents it in a unified, scannable view
- notifies users when slots open

Pros

- Directly reduces uncertainty
- Low behavior change
- Passive value delivery (alerts)
- Legally safer than automation
- Feasible to build and iterate quickly
- Builds trust through transparency

Cons

- Availability may not be perfectly real-time
- Requires careful handling of reliability
- Does not guarantee successful booking

Assessment
Best balance of value, feasibility, and adoption
Directly validates the core value hypothesis
Creates a foundation for future extensions

Why Aggregation + Alerts Won

This option was chosen because it:

- targets the core coordination pain
- minimizes user effort
- avoids overpromising automation
- respects platform constraints

SECTION 7 — MVP Scope (What I Built)

With the solution direction established, the MVP was intentionally scoped to validate the core value hypothesis:

> **Reducing uncertainty through aggregated availability and alerts would meaningfully improve coordination and increase successful bookings — without requiring behavior change.**

The MVP focused on delivering this value end-to-end, while avoiding features that would introduce unnecessary risk, complexity, or premature scale assumptions.

MVP Capability 1 — Aggregated Court Availability

The MVP surfaces a unified view of tennis court availability across multiple independent court operators.

What it does

- Periodically collects availability data from fragmented court websites
- Normalizes availability into a consistent internal format
- Displays upcoming open slots in a single, scannable interface

Why this mattered
Without aggregation, users are forced to mentally reconcile multiple systems.
This capability directly reduces cognitive load by presenting availability in one place.

PM reasoning

- Solves the most time-consuming part of coordination
- Provides immediate value even without alerts
- Creates a foundation for trust if accuracy is maintained

MVP Capability 2 — Map-Based Discovery & Location Awareness

The frontend presents court availability on a map, allowing users to visually assess proximity and convenience.

What it does

- Shows courts relative to a user's location
- Makes geographic tradeoffs visible at a glance
- Supports faster elimination of impractical options

Why this mattered
Coordination isn't just about *whether* a court is available — it's about *whether it works for everyone involved*.

PM reasoning

- Reduces back-and-forth discussions
- Helps groups converge faster on viable options
- Supports both regular and ad-hoc planning behavior

MVP Capability 3 — Availability Alerts (Email)

Instead of requiring users to repeatedly check availability, the MVP supports passive notifications.

What it does

- Allows users to sign up for alerts
- Notifies users when relevant slots become available
- Shifts effort from monitoring to decision-making

Why this mattered
Alerts replace habitual refreshing with a reliable signal — the single most impactful behavior change identified during exploration.

PM reasoning

- Delivers value even when users are not actively using the product
- Reduces frustration from missed opportunities
- Encourages repeat engagement through usefulness, not habit pressure

MVP Capability 4 — Lightweight Signup & Early Access

The MVP includes basic email capture to:

- enable alerts
- support controlled rollout
- measure early engagement

Why this mattered
This allowed validation of:

- willingness to opt in
- perceived value of alerts
- early demand without marketing investment

What the MVP Explicitly Did *Not* Include

Several features were intentionally excluded from the MVP:

- Automated booking on behalf of users
- Payments or monetization
- Social features or chat
- Group scheduling tools
- Multi-sport support

- Mobile apps

They were deprioritized in favor of learning quickly whether the core coordination problem could be meaningfully reduced.

SECTION 8 — Success Metrics

The goal of the Court Scout MVP was not to maximize usage or time spent in the product.
Success was defined by **whether coordination friction decreased and games actually happened**.

Metrics were therefore chosen to measure **outcomes**, not activity.

Primary Success Metric — Time to Successful Booking

Definition:
The time elapsed between a user starting to look for a court and successfully identifying a bookable slot.

Why this mattered:
In the status quo, users often spend long periods checking multiple websites without success.
Reducing this time directly reflects reduced cognitive load and improved coordination.

What success looks like:

- Faster convergence on viable options
- Less repeated checking
- Increased confidence in planning

Supporting Metric 1 — Alert to Booking Conversion

Definition:
The percentage of users who successfully book a court after receiving an availability alert.

Why this mattered:
This metric tests the core value hypothesis:

> Alerts reduce uncertainty and prompt action.

A high conversion rate indicates:

- alerts are timely
- availability signals are trusted
- notifications align with user intent

Supporting Metric 2 — Repeat Usage Across Weeks

Definition:
Whether users return to use CourtScout across multiple planning cycles.

Why this mattered:
Tennis planning is recurring.
Repeat usage indicates that the product meaningfully reduces friction, not just novelty-driven interest.

Supporting Metric 3 — Drop-off from Search to Alert Setup

Definition:
The percentage of users who view availability but do not set alerts.

Why this mattered:
This metric helps diagnose:

- whether aggregation alone is sufficient
- whether alert setup is intuitive
- whether users understand the product's value quickly

Guardrail Metrics — Reliability & Trust

Because trust is central to coordination, several guardrail metrics were defined:

- Accuracy of availability data
- Alert latency (time between slot opening and notification)
- Frequency of false positives or missed slots

Why these mattered:
In coordination products, unreliable signals damage trust faster than missing features.

What Metrics Were *Not* Used

Several common metrics were intentionally excluded:

- Daily active users
- Time spent in product
- Feature usage counts

Reason:
These metrics do not directly correlate with successful play or reduced coordination friction.

How Metrics Informed Next Decisions

These metrics were designed to inform:

- whether aggregation meaningfully reduces effort
- whether alerts are worth expanding
- where friction still exists in the flow
- whether to invest in deeper coordination features

The goal was to learn quickly, not to optimize prematurely.

SECTION 9 — Rollout Strategy

Given the coordination-heavy nature of the problem and the importance of trust and reliability, Court Scout was intentionally designed for a **controlled, staged rollout** rather than a broad public launch.

The goal of rollout was not rapid growth, but **high-signal learning**.

Phase 1 — Personal Use & Reliability Validation

The initial rollout focused on personal use and close observation.

Objectives

- Validate data accuracy and freshness
- Identify failure modes in availability signals
- Measure how often alerts aligned with real booking opportunities

Why this mattered
For coordination products, even a small number of false positives or missed alerts can erode trust quickly.
Before expanding usage, the system needed to be reliable enough to support real decisions.

Phase 2 — Limited Circulation to Known Players

The next planned phase was limited sharing with:

- friends
- regular playing partners
- small, known groups

Objectives

- Observe real-world coordination behavior
- Test alert usefulness across different planning styles
- Identify where users hesitate or drop off

Why this mattered
Early users were likely to provide qualitative feedback and tolerate rough edges, making them ideal for validating core assumptions without reputational risk.

Phase 3 — Targeted Community Distribution

Once reliability and value were confirmed, the product could be shared more broadly through:

- local tennis groups
- Spin or WhatsApp communities
- informal word-of-mouth channels

Objectives

- Validate repeat usage beyond close contacts
- Assess organic demand
- Measure whether the product solves coordination at slightly larger scale

SECTION 10 — Learnings & Reflection

The most meaningful insights came from observing how people plan — and why plans often fail.

1. Coordination Problems Are About Uncertainty, Not Choice

My initial instinct was that players needed *more options*: more courts, more views, more flexibility.

In practice, the opposite was true.

The biggest blocker to action was not lack of choice, but **lack of certainty**. When availability felt unreliable, players hesitated, deferred decisions, or abandoned plans entirely.

Key takeaway:
Reducing uncertainty creates more value than increasing optionality.This insight shaped decisions throughout the MVP, especially the emphasis on alerts and reliability over advanced filters or automation.

2. Trust Is the Core Currency of Coordination Products

In coordination-heavy workflows, trust is fragile.

A single missed alert or inaccurate availability signal carries more weight than several successful interactions. Users quickly form a mental model of whether a system is "safe" to rely on.

Key takeaway:
Reliability is not a technical requirement — it is a user experience requirement.

This reframed how I thought about rollout, scope, and guardrail metrics.

3. Passive Value Beats Active Engagement

Initially, it was tempting to design for frequent interaction: dashboards, refresh flows, richer browsing experiences.

However, the highest-value moments occurred when users **didn't have to open the product at all**.

Alerts replaced habitual checking.
Signals replaced monitoring.

Key takeaway:
The best coordination tools remove work rather than create engagement loops.

This reinforced the importance of designing for outcomes, not usage metrics.

4. Fragmentation Shifts the Burden to Users by Default

Fragmented systems don't fail visibly — they fail quietly by pushing coordination work onto users.

In this case:

- players became manual aggregators
- the most motivated person carried the burden
- friction accumulated invisibly

Key takeaway:
When infrastructure is fragmented, the product opportunity often lies in *absorbing coordination cost*, not replacing systems.

This reframing helped avoid over-scoping the solution toward automation or platform replacement.