

Ceezaa – PRD: Discover (Plan Your Life by Vibe)

TL;DR

Ceezaa Discover empowers users to plan experiences and activities based on moods, preferences, and group vibes—not just logistics. By surfacing options that fit the way users want to feel, Discover makes planning downtime or group activities effortless and engaging. The feature targets individuals and groups seeking spontaneous yet personalized plans that match their emotional energy.

Goals

Business Goals

- Achieve a 20% increase in user engagement and daily active users within three months of launch.
- Drive a 15% uplift in group session initiations and collaborations using DISCOVER.
- Increase in-app recommendations viewed by 30% quarter-over-quarter.
- Support retention by enabling 60% of users to save or revisit past vibe-based discovery sessions.

User Goals

- Discover activities and experiences that match current moods, boosting satisfaction and perceived relevance.
- Effortlessly plan with friends or solo by browsing, filtering, and saving options based on group or personal vibes.
- Recall and reuse previously successful plans with a memorable and accessible discovery history.
- Reduce planning friction by streamlining search, filtering, and group voting into a single, cohesive flow.

Non-Goals

- DISCOVER will not replace advanced calendar or detailed scheduling suites.
 - Transaction, payment, or ticket procurement integrations are not included in MVP.
 - Avoid deep personalization based on fine-grained personal analytics—focus on moods and group vibes, not exhaustive personal data mining.
-

User Stories

Persona 1: Solo User (Alex)

- Select activities by mood to quickly match personal feelings.
- Save favorite vibes and activities for easy future access.
- Adjust mood filters to avoid repetitive suggestions.

Persona 2: Group Organizer (Jamie)

- Invite friends to join vibe-based sessions for collaborative planning.
- Refine shortlisted activities based on live group votes.
- Reuse or remix successful past outings.

Persona 3: Returning User (Morgan)

- Review discovery history and planned vibes for easy repetition.
- Compare current and past moods for planning insight.

Functional Requirements

APIs and Endpoints

1. Session Management

Endpoint	Method	Description	Auth	Stub/Real
/api/sessions	POST	Create new session (solo or group)	Yes	Real
/api/sessions/{id}	GET	Retrieve session status and details	Yes	Real
/api/sessions/{id}/join	POST	Join an existing group session by code/link	Yes	Real
/api/sessions/{id}	PATCH	Update session parameters (moods, filters)	Yes	Real
/api/sessions/{id}/close	POST	Finalize session, archive history, handle cleanup	Yes	Real

Session Object Schema

2. Mood Selection & Aggregation

Endpoint	Method	Description	Auth	Stub/Real
/api/sessions/{id}/moods	POST	Update participant's selected mood and intensity	Yes	Real
/api/sessions/{id}/moods	GET	Get all current group moods	Yes	Real

Mood Object Example

3. Activity Discovery & Feed

Endpoint	Method	Description	Auth	Stub/Real
/api/feed/discover	POST	Fetch curated activities based on moods & filters	Yes	Stub for MVP
/api/activities/{id}	GET	Get details for a discovery result	Yes	Real

Activity Object Schema

- Allowed Third-Party APIs for Activity Curation:** For MVP, allow integration with public event/activity APIs such as Yelp Fusion, Eventbrite Public API, and any Ceezaa-owned curated internal database.
- Stubbed data:** Activity feed search is initially stubbed—use seeded sample activities and static integrations for MVP. Formal API connectors to third-party providers must be modular for future scaling.

4. Group Voting

Endpoint	Method	Description	Auth	Stub/Real
/api/sessions/{id}/vote	POST	Submit or update real-time votes on activities	Yes	Real
/api/sessions/{id}/votes	GET	Fetch current voting results	Yes	Real

Vote Object Schema

5. Saving & Bookmarking Activities

Endpoint	Method	Description	Auth	Stub/Real
/api/bookmarks	POST	Save activity to user or group memory	Yes	Real
/api/bookmarks	GET	List user's or group's saved items	Yes	Real

Bookmark Object Schema

6. Session History

Endpoint	Method	Description	Auth	Stub/Real
/api/sessions/history	GET	List all sessions for user	Yes	Real

Edge Cases & Error Handling

- No Feed Results Found**
 - API returns 204 No Content.
 - Fallback:** Display guidance message ("Try widening your vibe or filters."); suggest relaxing filters or trying a different mood. API suggests most recent popular activities as backup.

- **User Drops/Network Loss in Group Session**
 - `/sessions/{id}/status` endpoint monitors live connections.
 - **Fallback:** User is marked as temporarily inactive; session state is saved. If group falls below 2 active, pause voting and show "Waiting for more participants."
 - **Vote Submission After Activity Removed**
 - On POST `/vote`, check that `activityId` exists in current session activities.
 - **Fallback:** Return 409 Conflict if activity is stale; UI will prompt user to refresh feed.
 - **Race Conditions (Simultaneous Edits)**
 - Use ETag or revision field on session PATCH.
 - If PATCH rejected, client must fetch latest session and reapply changes.
 - **Invalid Custom Mood or Filter Data**
 - Mood names sanitized on input (no special characters, max 24 chars).
 - Invalid ranges (e.g., max < min) trigger 400 Bad Request.
 - **Session Expired or Already Closed**
 - All session-modifying API requests validate session is active.
 - 410 Gone if session no longer valid; UI prompts user to start new session.
-

Fallback Behaviors

- On API or network failure, client enters a "retry in background" state, showing cached feed/cards with last known good results.
 - For real-time voting, if socket fails, fallback to periodic poll (5s) endpoint sync.
 - If third-party activity feed integration rate-limits or errors, fallback to Ceezaa's static activity "starter" set.
-

Performance KPIs

KPI	Target for MVP
Feed refresh latency	≤ 1.5s for 95% of requests
Group vote sync propagation	≤ 600ms end-to-end
Session create/join response time	≤ 1s
Max concurrent active sessions	5,000+
Session state consistency	99.7%
API error rate (4xx/5xx, user flows)	< 2% per week
Uptime/availability (US/EU regions)	99.5%+

Integration & Dependencies

System/Service	Integration Required	MVP Status
User Auth	Yes	Real
Activity Content Feed	Yes	Stubbed with real API adapter interface for MVP; real integration in v2
Real-time Group Sync	Yes	Real (WebSockets/SignalR)
Notifications/Invites	Yes	Stubbed (simple email/link only)
Analytics/Tracking	Yes	Real
Third-party APIs	Optional	Mocked or limited real (e.g., Yelp with demo key)

User Experience

Entry Point & Onboarding

- Users access DISCOVER via the home screen or any shared session invite (deep link).
- Onboarding includes “Plan by Vibe” walkthrough and brief intro to mood selection, filtering, and group features.
- Optional personality or mood prompt is used to pre-populate first feed.

Core FLOW

1. **Mood Grid (Step 1):**
 - User selects from vibrant, accessible mood tiles (Chill, Adventurous, Social, Reflective, etc.).
 - Intensity/sliders are used for fine-tuning.
 - Custom mood creation available (input sanitized and limited).
2. **Refinement & Filtering (Step 2):**
 - Real-time preview of feed; filters validated on entry.
 - If filters are too restrictive, immediate visual warnings and suggestions.
3. **Discovery Feed (Step 3):**
 - Activities displayed as swipeable feed cards.
 - Details, source, cost, group suitability, images, and external links included.
 - Option to favorite, expand for info, or reject.

4. Group Session (Step 4):

- Sessions visible to participants; each user's selected moods/intensities displayed as avatars or tags.
- Real-time voting and reactions animate as group interacts.
- Automatic aggregation to create a dynamic shortlist.

5. Plan Creation & History (Step 5):

- On final selection, plan is saved to both user and group history.
- Modal for sharing, adding notes, or saving to bookmarks.
- History accessible from home/profile for quick revisit or "remix."

Advanced Features & Edge Cases

- "Remix" replaces rejected activities while respecting current mood/filters.
- Users can join group session late; their preferences update the feed dynamically.
- If minimum group size isn't met (e.g., everyone leaves), session is auto-paused with clear on-screen instructions.
- Limit custom moods in MVP to 3 per user to avoid clutter.

UI/UX Highlights

- AA-standard accessibility: tested color contrast for all interactive elements.
- Responsive design: optimized for mobile-first, usable on tablets.
- Micro-interactions/feedback for voting, mood changes, and state transitions.
- Clear empty/dead-end states ("No results? Let's remix your vibe or adjust filters!").

Narrative

It's Saturday night. Alex wants to make plans but doesn't have the energy to scroll endless lists or browse generic search results. He opens Ceezaa DISCOVER and is greeted by vibrant mood tiles—"Chill," "Adventurous," "Arts & Culture"—inviting him to choose how he wants to feel tonight. After picking "Chill" and "Low Energy," options immediately populate: outdoor movies, cozy cafés, mellow board game nights.

Inspired, Alex taps "Start Group Session" and invites his friends with a shareable link. As friends join, each sets their own vibe—some are "Social," others are "Relaxed." DISCOVER blends everyone's preferences, surfacing options like a local jazz bar or a scenic picnic spot. Group voting is effortless, playful, and visible in real time, eliminating group chat indecision.

Once a choice is made, the plan is saved and instantly shareable. Later, the group can revisit their history, remix the vibe for next time, or simply repeat what worked. Instead of hours lost in logistics, Ceezaa's DISCOVER makes spontaneous, authentic experiences just a couple of taps away—deepening user connection and fueling ongoing engagement for the business.

Success Metrics

Metric	Measurement Method
DAU/WAU growth (DISCOVER users)	Analytics dashboard, active user logs
Number of group sessions created	Session creation events
Average session duration	Start/end logs per discovery session
Feature utilization rate (mood filters)	Filter application events, per-user cohort
Repeat discovery session rate	Count of users reusing/saving past sessions
Group vote aggregation success	% of sessions reaching consensus

User-Centric Metrics

- Percentage of users who create or join at least one group session.
- Net Promoter Score for discovery experience.
- Save/bookmark rate per user per week.

Business Metrics

- Uplift in retention of users who use DISCOVER vs. those who do not.
- Increase in activities recommended per user/month.
- Viral coefficient measured via session invite shares.

Technical Metrics

- 99.5% uptime/reliability for real-time collaboration.
- Sub-2-second response time for mood/feed updates.
- Error rate on session creation and group join flows < 2%.

Tracking Plan

- Mood selection events
 - Filter application/change events
 - Session creation/start
 - Session join/invite
 - Activity save/bookmark
 - Group voting actions
 - Plan confirmation
 - Revisit/remix of prior sessions
-

Technical Considerations

API/Service Architecture

- All endpoints authenticated via JWT/session tokens.
- REST API for all core operations; WebSockets/SignalR/pub-sub for real-time group voting and session sync.
- Activity feed initially powered by in-memory stubbed sample set; back-end modularized for future third-party integrations.
- All data models versioned and documented in OpenAPI spec.

Data Models

- **Session:** Ties together participants, moods, filters, and discovered activities.
- **Mood:** Predefined or user-generated (with input validation/limits).
- **Activity:** Sourced from stub/internal DB or whitelisted third-party APIs.
- **Vote:** Each activity in session tracks tally and user votes.
- **Bookmark:** History and re-use marker (user or group scoped).

Integration/Dependency Summary

Component	Real for MVP	Stub	Notes
User authentication	Yes		OAuth/openid, existing Ceezaa auth flow
Activity/feed discovery		Yes	Feed stubbed with curated entries, real API adapter present
Real-time group sync	Yes		WebSocket or SignalR, fallback to polling
Notifications/invites		Yes	Email/link sharing only; push in later release
Analytics/tracking	Yes		Segment, Mixpanel, or similar

Error Handling & Stability

- All API responses use consistent JSON error structure: `{ "error": "description", "code": "status" }`
- Graceful UI feedback for common issues (connectivity loss, API failures, stale sessions).
- Consistent fallbacks for failed third-party integrations.
- Unit and integration tests stub external dependencies; coverage target 80%+ for all endpoints.

Performance & Scale

- Profile latency for mood->feed query under 1000ms (using stub set).
 - Real-time voting events must achieve end-to-end lag \leq 600ms.
 - Stress-test session concurrency to 5,000 active; horizontally scalable service deployment.
-

Milestones & Sequencing

Project Estimate

- Medium scope: 2–4 weeks for minimum viable product.

Team Size & Composition

- Product/UX: 1 Lead
- Engineering: 1 Full-Stack Developer
- UI/QA: 0.5 FTE (part-time shared resource)

Suggested Phases

1. Discovery & Prototyping (Week 1)

- **Deliverables:** UX spec, prototype, mood grid flows. Engineer spikes core APIs and mocks group sync.
- **Dependencies:** Activity stub data, user endpoint access.

2. Core Feature Build (Weeks 2–3)

- **Deliverables:** Functional front-end components; mood/grid filters; group backend; activity feed; real-time voting; session/bookmark APIs.
- **Dependencies:** Auth hooks, real-time group sync in test, analytics initialized.

3. Testing & Group Collaboration (Week 4)

- **Deliverables:** Usability QA, group voting/edge-case test runs, onboarding/empty states finalized.
- **Dependencies:** Test pool; load/stability monitoring set up.

4. Launch & Measurement (Following Week)

- **Deliverables:** Deploy MVP to production, enable real analytics, event tracking. Feedback checklist for revision.
 - **Dependencies:** Deployment credentials, legal/privacy compliance reviewed.
-