

Ceezaa MVP v1 – Profile PRD

TL;DR

Ceezaa's Profile feature serves as a private identity hub, allowing users to manage account details, view and curate saved places and playlists, and see a dynamic taste profile that evolves with use. This foundation not only centralizes core user data but also personalizes the experience, giving every user deeper engagement and insight into their unique preferences. It targets anyone seeking a personalized, evolving guide and reflection of their tastes within Ceezaa.

Goals

Business Goals

- Achieve 60%+ onboarding completion for new users via Profile within 2 weeks of launch.
- Increase weekly active engagement with saved lists/taste profiles by 30% over baseline.
- Reduce account-related support requests by 25% via intuitive self-service profile management.
- Capture opt-in preferences for notifications and communications to enable future retention campaigns.

User Goals

- Provide a centralized, easy-to-access space for controlling account details, privacy, and saved content.
- Empower users to curate and revisit favorite places and custom playlists for quick planning.
- Offer a personalized, visual summary of evolving tastes and patterns, building a sense of ownership.
- Allow convenient management of notifications, privacy settings, and sign out.

Non-Goals

- Public user profiles or social sharing from Profile (out of scope for MVP).
 - Direct user-to-user messaging or community activity feeds.
 - Deep analytics or downloadable data export functionality.
-

User Stories

Personas:

- Explorer (Everyday user)
- Curator (Avid planner with many saved places)
- Recommender (Power user, might share taste info externally)
- Support Admin (Internal system role)

Explorer

- As an Explorer, I want to see and update my account info, so that my profile remains accurate.
- As an Explorer, I want to view all my saved places and playlists in one tap, so that I can quickly find and revisit favorites.
- As an Explorer, I want a summary of my taste profile, so that I can understand how my preferences are evolving.

Curator

- As a Curator, I want to organize, add, or remove places from my lists, so that my recommendations are always relevant.
- As a Curator, I want to set privacy preferences, so I control what's kept private.
- As a Curator, I want efficient access to profile settings, so that maintenance is simple.

Recommender

- As a Recommender, I want a personalized taste dashboard, so I can track and reflect on my experiences.
- As a Recommender, I want options to manage notification types, so I only get alerts that matter to me.

Support Admin

- As a Support Admin, I want users to be able to edit/delete their profile data, so I receive fewer manual support requests.

Functional Requirements

- **Profile Home (Priority: High)**
 - Summary card with avatar, user name, and quick links to major areas.
 - Taste Profile: visual summary of dominant tastes, recent activity, and highlights.
 - Recent Saves: showcase most recently saved places/playlists.
- **Account Information (Priority: High)**
 - View and edit name, email, password, and avatar.
 - Email verification status.
 - Change password and delete account workflow.
- **Saved Places & Playlists (Priority: High)**

- List all saved places and playlists with thumbnails and metadata.
 - Ability to add, remove, or reorder items.
 - Curate public/private status (if planned for future versions).
 - **Dynamic Taste Profile (Priority: Medium)**
 - Generate and visually present user's taste patterns (e.g., cuisine types, favorite neighborhoods).
 - Update dynamically as users engage.
 - Simple badges or achievements for milestones.
 - **Notifications & Permissions (Priority: Medium)**
 - Manage notification preferences (e.g., weekly highlights, special offers).
 - Opt-in/out toggles.
 - Permissions management for location or account access.
 - **Privacy Settings (Priority: Medium)**
 - Overview and toggle for data sharing.
 - Access to privacy policy and terms.
 - Simple view/export of stored data (future enhancement).
 - **Logout (Priority: High)**
 - One-tap log out with confirmation modal.
-

User Experience

Entry Point & First-Time User Experience

- Users access Profile via dedicated tab or top-right menu icon.
- First-time users get a "Welcome" pop-up that outlines Profile's personal benefits and core sections.
- Optional guided walkthrough for key features (taste profile, saved places).

Core Experience

- **Step 1:** User lands on Profile Home.
 - Minimal friction: familiar UX layout, prominent avatar/name, dashboard feel.
 - Callouts for incomplete parts (e.g., verify email).
- **Step 2:** User explores taste profile.
 - Simple data visualization shows top tastes, recent trends.
 - Option to dive deeper for custom insights (if available).
- **Step 3:** User accesses Saved Places & Playlists.
 - Filter, sort, and manage individual lists.
 - Edits reflected in real time.
- **Step 4:** User edits account information.
 - Inline edit fields, clear feedback on success/error.
 - Password resets and delete account confirm with required reauth.

- **Step 5:** User manages notifications/permissions.
 - Toggle switches, clear descriptions for each option.
- **Step 6:** User reviews privacy settings or logs out.
 - Simple, accessible links and confirmation steps.

Advanced Features & Edge Cases

- Display appropriate UI for users with no saved items/taste activity (“Start saving places!”).
- Graceful error handling for connectivity or data sync issues.
- Power users: bulk edit saved items, see streaks/milestones in taste profile.

UI/UX Highlights

- Accessible, high-contrast design suitable for all users.
- Responsive mobile and desktop layouts.
- Clear, actionable call-to-actions and feedback messages.
- Support for dark and light mode.
- All actions support “undo” where relevant.

Narrative

Clara is a food enthusiast exploring new neighborhoods through Ceezaa. Over time, she's saved dozens of places and curated playlists for "Coffee Crawls," "Ramen Nights," and more. But keeping track of her journey had always been scattered—until she taps the Profile tab.

Here, Clara sees a dynamic snapshot of her evolving tastes: her most visited cuisines, favorite streets, and a badge for "Adventurous Eater." Her saved places are neatly organized, making planning a Friday night out with friends frictionless. She updates her email, tweaks notification settings to receive only city-wide new openings, and feels reassured by the clear privacy controls.

Now, every time Clara opens Ceezaa, Profile helps her reflect, plan, and celebrate her discoveries. It becomes not just an admin tool, but a personal reflection space—deepening her loyalty and excitement for what she'll find next, while freeing support admins from redundant requests. Ceezaa pivots from being just another recommendation app to a true companion on Clara's food adventure.

Success Metrics

Metric	Measurement
Profile tab daily/weekly active usage	% of MAUs accessing Profile

Saved item engagement rate	% of users editing/saving per week
Taste profile view rate	% of users viewing taste summary
Account update self-service rate	% of users updating account w/o help
Notification opt-in rate	% of users customizing notifications
Profile drop-off rate	# of users abandoning onboarding UX

User-Centric Metrics

- % of users accessing Profile at least weekly
- Average saved items/playlists per user
- User satisfaction score (post-interaction survey)

Business Metrics

- Decrease in account-related support tickets
- Increase in retention and repeat engagement metrics
- Notification opt-in rates

Technical Metrics

- Profile load time (<1.2s target)
- Uptime (>99.7%)
- Profile data sync error rates (<0.5%)

Tracking Plan

- Profile tab click
- Edit account info
- Save/remove places or playlists
- Taste profile view
- Notification/permission toggle
- Successful logout

Technical Considerations

Technical Needs

- Secure user data model for account info, saved items, and taste analytics.
- Endpoints for CRUD operations: account, saved items, profile data.
- Front-end Profile dashboard: responsive and modular UI.
- Back-end services for taste analytics (batch/dynamic updates).

Integration Points

- Authentication system for login and session management.
- Existing data stores for places and playlists.
- Notification/permissions system integration.

Data Storage & Privacy

- All user data encrypted at rest and in transit.
- Adherence to GDPR/CCPA for right to access/delete.
- Clear separation of private (profile-only) and public (future use) data.

Scalability & Performance

- Initial support for 10,000+ concurrent users.
- Taste profile and saved item lookups sub-500ms.
- Cache commonly-viewed items for speed.

Potential Challenges

- Handling complex merge/update logic for saved items.
- Ensuring instant UX feedback on slow connections.
- Providing robust fallbacks for failed data fetches.
- Clear communication for sensitive actions (delete, email update).

Milestones & Sequencing

Project Estimate

- **Medium:** 2–4 weeks

Team Size & Composition

- **Small Team:** 2–3 people
 - 1 Product/Designer (owns spec, flows, wireframes)
 - 1 Full-stack Engineer (front-end/back-end/API)
 - Optional: 1 part-time QA/support

Suggested Phases

1. Foundation & Data Modeling (Week 1)

- Key Deliverables:
 - Data models for account/profile/saved items (Engineer)
 - High-fidelity wireframes (Product/Designer)
- Dependencies:

- Access to auth/database systems

2. Core Feature Build (Weeks 2-3)

- Key Deliverables:
 - CRUD for account, saved items, taste profile (Engineer)
 - Responsive Profile dashboard & onboarding UX (Engineer/Product)
- Dependencies:
 - Completion of foundational data models

3. Polishing & Testing (Week 4)

- Key Deliverables:
 - UX refinements, accessibility, dark mode (Product/Engineer)
 - Manual/automated tests, bugfixes, QA pass (Engineer, QA/Support)
- Dependencies:
 - Core feature completion

4. Launch & Monitoring (End of Week 4)

- Key Deliverables:
 - Production deployment, success metric monitoring setup (Engineer)
 - Pre/post-launch support checklist (Product/Support)
 - Dependencies:
 - Completion of all previous phases
-