# Product Specification — Food Chooser Application

## 1. Overview

The Food Chooser application helps users decide what to eat for dinner, reducing decision fatigue and making the experience engaging and enjoyable through personalization and gamification.  
  
The product will evolve from a simple meal-tracking and recommendation engine to an AI-powered assistant that can automatically order food and integrate with food delivery services.

## 2. Goals & Objectives

* Primary Goal: Help users decide what to eat for dinner by leveraging historical data, user preferences, contextual conditions (e.g., weather), and budget.
* Secondary Goals:

1. Gamify the food selection process to increase engagement.
2. Reduce repetitive choices while allowing for practical trends.
3. Enable seamless integrations with food delivery services in the future.
4. Create a potential revenue stream through partnerships with services like Uber Eats, DoorDash, or Yelp.

## 3. Product Evolution Roadmap

### Phase 1 — MVP (Manual Entry & Basic AI Model)

* Core Features:

1. Meal History Tracking: Users manually log dinner choices and associated cost. Optionally add notes on day’s activities or mood. Application automatically enriches entries with contextual data (e.g., weather, day of week).
2. Budget & Configurations: User sets budget range (min/max). App considers budget in recommendations.
3. Basic Recommendation Engine: Assigns weights to foods based on past satisfaction ratings (1–5 stars), recency, and seasonal/weather patterns. Overrides increase future recommendation likelihood.
4. Gamified Selection: Users receive a “mystery egg” (Bronze, Silver, Gold, Diamond) corresponding to a budget tier. The egg reveals a food recommendation.
5. Spending Tracking: Aggregate and display total food spending over time.

### Phase 2 — Automated Ordering & LLM Integration

1. Delivery Service Integration: Connect with Uber Eats or DoorDash APIs. Option for AI to place orders automatically for a surprise experience.
2. LLM-Based Recommendation: Feed full user history and contextual data to a large language model for natural-language recommendations.
3. Contextual Constraints: Incorporate dietary restrictions, preparation time, or delivery speed preferences.

### Phase 3 — Partnership & Monetization

1. Licensing recommendation engine to food platforms (e.g., Yelp, Uber Eats).
2. “Mystery Deal” promotions: occasional surprise recommendations tied to discounts to increase daily active users.
3. Premium subscription for advanced customization and priority deals.

## 4. Recommendation Model (MVP)

Initial Model Inputs:  
- Historical food choices.  
- Satisfaction ratings.  
- Recency scores.  
- Weather data (temp, precipitation, season).  
- Budget range.  
  
Logic:  
1. Score each food using a weighted formula: Score = (Rating Weight) + (Recency Penalty) + (Weather Match Bonus) + (Budget Fit Score)  
2. Exclude foods eaten in the past 1 day.  
3. Adjust weights dynamically based on user overrides and trends.  
  
Output: Ranked list of possible dinner options → Gamified reveal.

## 5. Gamification Design

Mystery Egg Tiers:  
- Bronze: Budget-friendly option.  
- Silver: Mid-range option.  
- Gold: Premium choice.  
- Diamond: Top-tier splurge.  
  
Mechanics:  
- Egg opens to reveal restaurant + dish recommendation.  
- Occasional surprise upgrades to higher tiers to delight users.

## 6. Success Metrics

* Engagement Metrics: DAU, number of recommendations generated per week, override frequency.
* User Satisfaction Metrics: Average post-meal rating, reduction in decision time for dinner.
* Business Metrics (future phases): Partnership deals signed, incremental revenue from mystery deal conversions.

## 7. Risks & Mitigations

|  |  |  |
| --- | --- | --- |
| Risk | Impact | Mitigation |
| Cold start problem (no history) | Low recommendation quality | Use starter food list + quick preference survey |
| Budget misalignment | Poor recommendations | Require budget input in onboarding |
| User fatigue with gamification | Reduced engagement | Offer manual list view as alternative |
| API changes with delivery partners | Feature downtime | Modular integration layer for flexibility |

## 8. Next Steps

1. Define data schema for meal logging.
2. Implement MVP UI for logging and viewing history.
3. Develop basic scoring model.
4. Design gamification UI (egg animations, reveal flow).
5. Conduct limited beta with manual data entry.