**Modelo funcional — Ranking de platos (App web)**

**Descripción breve:** Plataforma web para mostrar platos/comidas de empresas de comida rápida locales en forma de tarjetas únicas en la pantalla principal. Cada tarjeta representa un *plato canónico* (p. ej. "Hamburguesa clásica"). Al hacer click abre un modal con el *ranking* de las casas que venden ese plato (puesto #1, #2, #3...), donde la casa en #1 es la que tiene más puntos/votos. El usuario puede votar por el plato que probó y así sumar puntos a la casa donde lo comió. También podrá ver el número de teléfono de la casa (o revelarlo con un click) y otras acciones (compartir, marcar favorito, denunciar).

**Objetivos del documento**

1. Definir las funcionalidades necesarias para el desarrollo (MVP y alcance extendido).
2. Describir el comportamiento esperado de cada funcionalidad (casos de uso y validaciones).
3. Proponer un modelo de datos relacional y endpoints básicos para implementar la lógica de ranking.

**Público objetivo**

* Vecinos y turistas que quieren encontrar y puntuar comidas locales.
* Propietarios/encargados de casas de comida que quieren aparecer en el ranking y recibir contactos.
* Administrador del sitio (moderación y gestión).

**MVP recomendado (mínimo viable)**

1. Pantalla principal que muestre **platos únicos** (sin repetir) con: imagen, nombre del plato y la empresa en #1.
2. Modal de ranking por plato con listado ordenado por puntos (mostrar teléfono de #1 y botón «Ver número»).
3. Votación: usuario autenticado o anónimo puede votar (restricción anti-abuso básica: 1 voto por plato por IP/usuario por día).
4. CRUD básico de locales y platos (desde admin).
5. Búsqueda simple por plato o por local.
6. Dashboard admin para moderar locales, denuncias y ver métricas básicas.

**1) Esquema de funcionalidades a implementar**

**Nota:** las funcionalidades están agrupadas por prioridad: *MVP*, *Adicionales deseables*, *Monetización / Admin avanzado / Seguridad*.

**Core (MVP)**

* **Pantalla principal (Home)**
  + Mostrar tarjetas por *plato canónico* (sin duplicados). Cada tarjeta muestra: imagen principal, nombre del plato y la empresa actualmente #1 (nombre corto) y un botón "Ver número".
  + Paginación o carga infinita.
* **Modal de ranking por plato**
  + Orden descendente: #1, #2, #3... con número de teléfono, puntaje y botón para llamar/mostrar número.
  + Mostrar foto del local (si hay), horario, dirección corta.
* **Sistema de votación**
  + Votar por el plato y asociar ese voto a la casa donde se probó.
  + Reglas básicas: 1 voto por usuario por plato cada X tiempo (MVP: 1 por día), posibilidad de revertir voto.
* **Autenticación básica**
  + Registro/login por email + contraseña. Opcional: social login (Google/Facebook).
  + Recuperación de contraseña.
* **Gestión de establecimientos y platos (Admin/Moderador)**
  + Crear/editar/activar/desactivar locales y platos canónicos.
  + Revisar y aprobar platos agregados por locales (si permitimos autogestión).
* **Búsqueda / filtro**
  + Buscar platos por nombre, filtrar por categoría (hamburguesas, papas, pizzas, fideos, etc.).
* **Reportes/Denuncias**
  + Usuarios pueden denunciar un plato o una empresa (info errónea, teléfono falso, contenido inapropiado).

**Adicionales deseables**

* **Perfiles de usuario** (historial de votos, favoritos, badges).
* **Reclamación/Verificación de locales** (que un dueño reclame su ficha mediante verificación telefónica o código SMS).
* **Favoritos / Listas** (guardar platos o locales).
* **Comentarios / mini-reseñas** (opcional; moderación necesaria).
* **Compartir en redes sociales** (previsualización con OG tags).
* **Geolocalización y filtro por distancia** (mostrar locales cercanos al usuario).
* **Imagenes por plato**: galería, posibilidad que el local suba fotos de su versión del plato.

**Monetización / Admin avanzado / Seguridad**

* **Promociones / Destacados**: locales pueden pagar por aparecer como "destacado" en un plato (siempre con etiqueta "Patrocinado").
* **Dashboard avanzado**: métricas por local/plato (visitas, llamadas generadas, conversiones, clicks "ver número").
* **Prevención de fraude**: bloqueo por IP/agregación de dispositivos, límites diarios, detección de patrones (múltiples votos desde el mismo dispositivo), reCAPTCHA en votaciones anónimas.
* **Integración con pasarela de pagos** (si se monetiza con promociones o suscripciones).

**2) Explicación detallada de las funcionalidades**

A continuación se muestra el comportamiento requerido, validaciones y ejemplos de flujo para cada funcionalidad crítica.

**A. Pantalla principal (Home)**

**Qué hace:** muestra tarjetas de *platos canónicos* sin repetir. Cada tarjeta contiene: imagen del plato, nombre del plato, nombre del local que ocupa #1, puntaje visible (opcional) y botón "Ver número".

**Comportamiento / Reglas:**

* No repetir el mismo plato canónico. Si existen versiones de "Hamburguesa clásica" de distintos locales, se muestran como una sola tarjeta que representa el plato; al abrir, el modal muestra el ranking con esos locales.
* La tarjeta muestra al local en #1 (según ranking calculado) y una foto representativa (preferiblemente la foto subida por el local o una foto crowdsourced con mayor votación).
* Al hacer click en cualquier parte de la tarjeta o en "Ver ranking" se abre el modal.

**UI:** componente DishCard con props: { dishId, dishName, imageUrl, topLocal }.

**B. Modal de ranking por plato**

**Qué hace:** muestra lista ordenada de establecimientos que ofrecen ese plato. El primero aparece destacado con información de contacto (teléfono), y se muestran los puestos siguientes con su puntaje.

**Comportamiento / Reglas:**

* El primer registro (#1) se destaca (fondo/etiqueta) y su número aparece con un botón "Ver número".
* Para proteger datos y permitir medir leads, el teléfono puede mostrarse sólo tras un click (registrar evento phone\_view). Para evitar scraping masivo, añadir límites por IP/usuario para revelar teléfonos.
* Mostrar total de votos y puntos por cada casa. Mostrar fecha de última votación relevante (para dar contexto de actualidad).

**Interacciones:**

* Botón "Votar por este plato (en esta casa)" en cada fila; al votar, sumar puntos a la casa correspondiente.
* Botón "Llamar" (link tel:) si el navegador/plataforma lo permite.

**C. Sistema de votación y reglas de puntuación**

**Qué hace:** permite a un usuario dar puntos a la casa en la que probó un plato.

**Modelo simple (MVP):**

* Cada voto vale +1 punto.
* Restricción: *1 voto por usuario por plato por día* (o por IP si el usuario no está registrado). Esto reduce manipulación inmediata.
* Guardar el histórico de votos con user\_id (opcional si anónimo), dish\_id, establishment\_id, points, created\_at, meta (ip, user-agent).

**Variantes avanzadas:**

* Sistema de pesos: votos de usuarios verificados (teléfono verificado) valen más.
* Decaimiento temporal: puntos antiguos se van depreciando con el tiempo (para reflejar popularidad reciente).

**Transacción atómica:** al crear un voto:

1. Validar que el usuario puede votar (no excede límite).
2. Insertar fila en votes.
3. Actualizar contador/tabla de ranking (tabla cacheada) o disparar job para recalcular.

**Anti-fraude:**

* Limitar por IP, cookie/device id.
* Bloquear patrones: múltiples votos desde la misma IP hacia el mismo establecimiento en corto tiempo.
* Uso de reCAPTCHA para votantes anónimos.

**D. Autenticación / Perfiles de usuario**

**Qué hace:** permite a usuarios registrarse, identificarse y gestionar su historial.

**Requerimientos mínimos:**

* Registro con email + contraseña y verificación de correo.
* Login con email + contraseña.
* Recuperación de contraseña por email.
* Opcional: login social (Google).

**Comportamiento adicional:**

* Usuarios pueden ver su historial de votos, favoritos y denunciar contenidos.
* Roles: user, business\_owner (con posibilidad de reclamar ficha), admin.

**E. Gestión de establecimientos y platos (Admin y dueños)**

**Admin:**

* Crear/editar/eliminar/ocultar establecimientos y platos canónicos.
* Revisar solicitudes de alta de locales o edición solicitadas por dueños.

**Dueños (opcional):**

* Reclamar ficha del local: proceso con verificación por SMS o llamada con código.
* Subir fotos, horarios, y datos de contacto.
* Ver métricas básicas (visitas a la ficha, clics en "ver número").

**F. Búsqueda y filtros**

**Qué hace:** buscar platos por keywords, filtrar por categoría, distancia y orden (popularidad, más votados, recientes).

**Reglas:**

* Auto-complete en búsqueda por nombre de plato y por local.
* Filtro por categoría.
* Ordenamientos: por score (ranking global del plato), recientes (platos con mayor actividad en los últimos X días), relevancia.

**G. Reportes / Moderación**

**Qué hace:** permitir a usuarios reportar datos incorrectos (teléfono, ubicación, contenido ofensivo) y a admins revisarlos.

**Flujo:**

* Usuario marca "Denunciar" -> describir motivo -> crear ticket en sistema de moderación.
* Admin recibe tickets y puede: corregir datos, desactivar ficha, contactar dueño.

**H. Seguridad y Anti-Fraude**

* reCAPTCHA para acciones sensibles (voto anónimo, alta de local).
* Rate limiting por IP en endpoints de voto y en revelar teléfonos.
* Verificación por teléfono para dueños y para usuarios que quieran aumentar peso de sus votos.
* Logs y alertas automáticas ante ráfagas de votación inusuales.

**I. Analítica y métricas**

* Métricas por plato y por local: visitas a tarjeta, aperturas de modal, clicks en "ver número", llamadas iniciadas (si se logran), votos por día.
* Dashboard admin con gráficos (top platos, top locales, tendencias semanales).