**Admin Panel Fields**

1. Menu Management

1.1 Menu Items

* Item ID: Unique identifier for each menu item (auto-generated).
* Name: Name of the dish (e.g., "Margherita Pizza").
* Description: Detailed description of the item (text, supports markdown for formatting).
* Price: Base price of the item (decimal, supports multiple currencies).
* Variant Prices: Prices for item variants (e.g., small: $10, large: $15) (array of objects: {variant\_name, price}).
* Image: URL or file path to the item’s image (supports multiple images).
* Video: URL to a short video of the dish (optional).
* Category ID: Reference to the parent category (e.g., "Pizzas").
* Subcategory ID: Reference to the parent subcategory (e.g., "Vegetarian Pizzas").
* Tags: Array of tags (e.g., ["spicy", "gluten-free", "vegan"]).
* Nutritional Information:
  + Calories (integer, e.g., 500 kcal).
  + Allergens (array, e.g., ["gluten", "dairy"]).
  + Dietary Flags (array, e.g., ["vegan", "keto"]).
* Customizations: List of customizable options (array of objects: {name, options, additional\_cost}).
  + Example: {name: "Toppings", options: ["Extra Cheese", "Mushrooms"], additional\_cost: [2.00, 1.50]}.
* Inventory Status: Boolean indicating if the item is in stock (true/false).
* Stock Quantity: Number of available items or linked ingredients (integer, optional).
* Preparation Time: Estimated prep time in minutes (integer, e.g., 15).
* Availability:
  + Start Date/Time (datetime, for seasonal items).
  + End Date/Time (datetime, for seasonal items).
  + Days Available (array, e.g., ["Monday", "Tuesday"]).
* Language Translations:
  + Name (object, e.g., {en: "Margherita Pizza", es: "Pizza Margherita"}).
  + Description (object, e.g., {en: "Classic pizza...", es: "Pizza clásica..."}).
* SEO Metadata:
  + Meta Title (string).
  + Meta Description (string).
* Is Active: Boolean to enable/disable the item (true/false).

1.2 Menu Categories

* Category ID: Unique identifier (auto-generated).
* Name: Category name (e.g., "Appetizers").
* Description: Brief description (text).
* Image: URL or file path to category image.
* Parent Category ID: Reference to parent category for nested categories (optional).
* Priority: Integer to determine display order (e.g., 1 for top priority).
* Language Translations:
  + Name (object, e.g., {en: "Appetizers", fr: "Entrées"}).
  + Description (object).
* Discount: Percentage or fixed discount applied to the category (optional, e.g., 10%).
* Is Active: Boolean to enable/disable the category.

1.3 Menu Subcategories

* Subcategory ID: Unique identifier (auto-generated).
* Name: Subcategory name (e.g., "Vegetarian").
* Description: Brief description (text).
* Image: URL or file path to subcategory image.
* Parent Category ID: Reference to the parent category.
* Priority: Integer for display order.
* Language Translations:
  + Name (object).
  + Description (object).
* Discount: Percentage or fixed discount (optional).
* Is Active: Boolean to enable/disable the subcategory.

2. Restaurant/Cafe Management

2.1 Restaurant/Cafe Profile

* Restaurant ID: Unique identifier (auto-generated).
* Name: Restaurant name (e.g., "Tasty Bistro").
* Description: Detailed description of the restaurant (text, supports markdown).
* Address:
  + Street Address (string).
  + City (string).
  + State/Province (string).
  + Postal Code (string).
  + Country (string).
  + Latitude (decimal, for geolocation).
  + Longitude (decimal, for geolocation).
* Contact Information:
  + Phone Number (string, supports multiple).
  + Email (string).
  + Website URL (string, optional).
* Logo: URL or file path to the restaurant’s logo.
* Cover Image: URL or file path to a cover image or banner.
* Social Media Links:
  + Instagram URL (string, optional).
  + TikTok URL (string, optional).
  + Facebook URL (string, optional).
  + Twitter/X URL (string, optional).
* Virtual Tour URL: Link to a 360° virtual tour or video (optional).
* SEO Metadata:
  + Meta Title (string).
  + Meta Description (string).
* Branding:
  + Primary Color (hex code, e.g., #FF5733).
  + Secondary Color (hex code).
  + Font Family (string, e.g., "Roboto").
* Language Translations:
  + Name (object).
  + Description (object).
* Is Multi-Location: Boolean indicating if the restaurant has multiple branches.

2.2 Operating Hours

* Restaurant ID: Reference to the restaurant.
* Day of Week: Enum (e.g., Monday, Tuesday, ..., Sunday).
* Opening Time: Time in HH:MM format (e.g., 09:00).
* Closing Time: Time in HH:MM format (e.g., 22:00).
* Is Closed: Boolean indicating if the restaurant is closed on this day.
* Special Hours:
  + Date (datetime, for holidays or events).
  + Opening Time (HH:MM).
  + Closing Time (HH:MM).
  + Is Closed (boolean).

3. Order Management

3.1 Orders

* Order ID: Unique identifier (auto-generated).
* Customer ID: Reference to the customer (optional for guest orders).
* Customer Name: Name provided during checkout (string).
* Customer Email: Email for order confirmation (string, optional).
* Customer Phone: Phone number for contact (string, optional).
* Order Items:
  + Array of objects:
    - Item ID (reference to menu item).
    - Variant (string, e.g., "Large").
    - Quantity (integer).
    - Customizations (array, e.g., ["Extra Cheese", "No Onions"]).
    - Unit Price (decimal).
    - Total Price (decimal).
* Total Price: Sum of all item prices, including customizations (decimal).
* Discount Applied: Discount amount (decimal, optional).
* Tax Amount: Calculated tax based on location (decimal).
* Delivery Fee: Fee for delivery (decimal, optional).
* Tip Amount: Tip for staff or driver (decimal, optional).
* Final Amount: Total after discounts, taxes, and fees (decimal).
* Order Type: Enum (e.g., Pickup, Delivery, Dine-In).
* Delivery Address (if Delivery):
  + Street Address (string).
  + City (string).
  + State/Province (string).
  + Postal Code (string).
  + Country (string).
  + Latitude (decimal).
  + Longitude (decimal).
* Order Status: Enum (e.g., Pending, Preparing, Out for Delivery, Delivered, Cancelled).
* Payment Method: Enum (e.g., Credit Card, Apple Pay, Cash on Delivery).
* Payment Status: Enum (e.g., Paid, Pending, Failed).
* Payment ID: Reference to payment gateway transaction (string).
* Order Date/Time: Timestamp of order placement (datetime).
* Estimated Prep Time: AI-predicted preparation time (integer, minutes).
* Estimated Delivery Time: AI-predicted delivery time (datetime, optional).
* Driver ID: Reference to delivery driver (optional).
* Notes: Special instructions from customer (text, optional).
* Channel: Source of the order (e.g., App, Website, Third-Party, In-Store).

3.2 Order History

* Same fields as Orders, plus:
* Review: Customer rating (1-5 stars) and feedback (text, optional).
* Review Images: URLs to customer-uploaded photos (array, optional).
* Reorder Count: Number of times the order was reordered (integer).

4. Reporting and Analytics

4.1 Sales Report

* Report ID: Unique identifier (auto-generated).
* Date Range: Start and end dates for the report (datetime).
* Total Sales: Total revenue (decimal).
* Total Orders: Number of orders (integer).
* Top-Selling Items:
  + Array of objects: {Item ID, Name, Quantity Sold, Revenue}.
* Sales by Category:
  + Array of objects: {Category ID, Name, Revenue, Percentage of Total}.
* Peak Hours:
  + Array of objects: {Hour, Order Count, Revenue}.
* Location ID: Reference to restaurant branch (for multi-location).

4.2 Customer Insights

* Customer ID: Reference to customer.
* Order Frequency: Average orders per month (decimal).
* Average Order Value: Average spend per order (decimal).
* Favorite Items: Array of Item IDs based on order history.
* Dietary Preferences: Array of tags (e.g., ["vegetarian", "gluten-free"]).
* Churn Risk: AI-predicted likelihood of churn (percentage).
* RFM Metrics:
  + Recency (days since last order, integer).
  + Frequency (total orders, integer).
  + Monetary (total spend, decimal).

4.3 Operational Analytics

* Ingredient Usage:
  + Ingredient ID (unique identifier).
  + Name (string).
  + Quantity Used (decimal, e.g., 10 kg).
  + Cost (decimal).
* Staff Performance:
  + Staff ID (reference to admin user).
  + Orders Processed (integer).
  + Average Processing Time (integer, minutes).
* Delivery Performance:
  + Driver ID (reference to driver).
  + Deliveries Completed (integer).
  + Average Delivery Time (integer, minutes).

5. User Management

5.1 Admin Users

* User ID: Unique identifier (auto-generated).
* Username: Unique login name (string).
* Password: Hashed password (string).
* Email: Contact email (string).
* Full Name: Admin’s full name (string).
* Role: Enum (e.g., Admin, Manager, Staff, Chef).
* Permissions: Array of permissions (e.g., ["manage\_menu", "view\_orders"]).
* Location ID: Reference to assigned restaurant branch (optional).
* Last Login: Timestamp of last login (datetime).
* MFA Enabled: Boolean for multi-factor authentication.
* Status: Enum (e.g., Active, Suspended).

5.2 Customer Management

* Customer ID: Unique identifier (auto-generated).
* Email: Unique email for login (string).
* Password: Hashed password (string, optional for social logins).
* Full Name: Customer’s name (string).
* Phone Number: Contact number (string, optional).
* Address:
  + Array of objects: {Street Address, City, State, Postal Code, Country, Latitude, Longitude, Is Default}.
* Dietary Preferences: Array of tags (e.g., ["vegan", "no nuts"]).
* Loyalty Points: Total points earned (integer).
* Loyalty Tier: Enum (e.g., Bronze, Silver, Gold).
* Order History: Array of Order IDs.
* Consent:
  + Marketing Consent (boolean).
  + Data Sharing Consent (boolean).
* Last Login: Timestamp of last login (datetime).
* Referral Code: Unique code for refer-a-friend program (string).
* Referred By: Customer ID of the referrer (optional).

6. Settings

6.1 Payment Gateway Integration

* Gateway ID: Unique identifier (auto-generated).
* Provider: Enum (e.g., Stripe, PayPal, Square).
* API Key: Secure key for gateway integration (string).
* Secret Key: Secure secret for gateway (string).
* Supported Methods: Array (e.g., ["Credit Card", "Apple Pay"]).
* Currency: Default currency (e.g., USD, EUR).
* Is Active: Boolean to enable/disable the gateway.

6.2 Notification Settings

* Notification ID: Unique identifier (auto-generated).
* Channel: Enum (e.g., Email, SMS, Push, WhatsApp).
* Event: Enum (e.g., Order Placed, Order Delivered, Promotion).
* Template: Custom message template (text, supports variables like {order\_id}).
* Is Enabled: Boolean to enable/disable the notification.
* Provider:
  + Provider Name (e.g., Twilio, Firebase).
  + API Key (string).
  + Secret Key (string).

6.3 Tax & Compliance

* Tax Rule ID: Unique identifier (auto-generated).
* Location: Reference to city/state/country.
* Tax Rate: Percentage (decimal, e.g., 8.5).
* Tax Name: Name of the tax (e.g., "Sales Tax").
* Is Active: Boolean to enable/disable the tax rule.

User App Fields

1. Menu Browsing

1.1 Menu Items

* Same fields as Admin Panel > Menu Items, but filtered for customer view:
  + Expose: Item ID, Name, Description, Price, Variant Prices, Image, Video, Category ID, Subcategory ID, Tags, Nutritional Information, Customizations, Preparation Time, Availability, Language Translations.
  + Hide: Inventory Status, Stock Quantity, SEO Metadata, Is Active (handled server-side).

1.2 Menu Categories

* Same fields as Admin Panel > Menu Categories, exposing:
  + Category ID, Name, Description, Image, Priority, Language Translations, Discount.
  + Hide: Parent Category ID, Is Active (handled server-side).

1.3 Menu Subcategories

* Same fields as Admin Panel > Menu Subcategories, exposing:
  + Subcategory ID, Name, Description, Image, Priority, Language Translations, Discount.
  + Hide: Parent Category ID, Is Active (handled server-side).

2. Ordering System

2.1 Cart Management

* Cart ID: Unique identifier (auto-generated, session-based).
* Customer ID: Reference to customer (optional for guest users).
* Items:
  + Array of objects:
    - Item ID (reference to menu item).
    - Variant (string).
    - Quantity (integer).
    - Customizations (array).
    - Unit Price (decimal).
    - Total Price (decimal).
* Subtotal: Sum of item prices (decimal).
* Discount Applied: Discount amount (decimal, optional).
* Tax Amount: Calculated tax (decimal).
* Delivery Fee: Fee for delivery (decimal, optional).
* Tip Amount: Tip amount (decimal, optional).
* Final Amount: Total after discounts, taxes, and fees (decimal).
* Saved Cart Name: Name for saved carts (string, optional).

2.2 Checkout

* Order ID: Generated upon checkout (auto-generated).
* Customer ID: Reference to customer (optional).
* Customer Name: Name for guest checkout (string).
* Customer Email: Email for confirmation (string).
* Customer Phone: Phone for contact (string, optional).
* Delivery Address (if Delivery):
  + Street Address, City, State, Postal Code, Country, Latitude, Longitude.
* Order Type: Enum (Pickup, Delivery, Dine-In).
* Payment Method: Enum (e.g., Credit Card, Apple Pay).
* Payment Details:
  + Card Number (masked, e.g., \*\*\*\* \*\*\*\* \*\*\*\* 1234).
  + Expiry Date (MM/YY).
  + Cardholder Name (string).
* Loyalty Points Used: Points applied to discount (integer, optional).
* Promo Code: Code for discounts (string, optional).
* Notes: Special instructions (text, optional).

3. Order Tracking

3.1 Order Status

* Order ID: Reference to the order.
* Status: Enum (Pending, Preparing, Out for Delivery, Delivered).
* Status Updates:
  + Array of objects: {Status, Timestamp, Notes}.
* Estimated Delivery Time: Predicted delivery time (datetime).
* Driver Details (if Delivery):
  + Driver ID (reference).
  + Driver Name (string).
  + Driver Phone (string, optional).
  + Current Location (latitude, longitude, updated in real-time).
* ETA: Estimated time of arrival (datetime).

3.2 Order History

* Same fields as Admin Panel > Order History, exposing:
  + Order ID, Order Items, Total Price, Order Type, Delivery Address, Order Status, Payment Method, Order Date/Time, Review, Review Images, Reorder Count.

4. Personalization

4.1 User Profiles

* Same fields as Admin Panel > Customer Management, exposing:
  + Customer ID, Email, Full Name, Phone Number, Address, Dietary Preferences, Loyalty Points, Loyalty Tier, Order History, Referral Code.
  + Hide: Password, Consent, Last Login, Referred By (managed server-side).

4.2 Recommendations

* Recommendation ID: Unique identifier (auto-generated).
* Customer ID: Reference to customer.
* Recommended Items: Array of Item IDs.
* Recommendation Type: Enum (e.g., Based on History, Popular, Complementary).
* Timestamp: When the recommendation was generated (datetime).

4.3 Gamification

* Badge ID: Unique identifier (auto-generated).
* Customer ID: Reference to customer.
* Badge Name: Name of the badge (e.g., "Loyal Customer").
* Badge Description: Description of the badge (text).
* Earned Date: Timestamp when earned (datetime).
* Reward Points: Points awarded for the badge (integer, optional).

5. Social Features

5.1 Community Reviews

* Review ID: Unique identifier (auto-generated).
* Order ID: Reference to the order.
* Customer ID: Reference to the customer.
* Item ID: Reference to the menu item (optional).
* Rating: 1-5 stars (integer).
* Comment: Review text (text).
* Images: URLs to uploaded photos (array).
* Timestamp: When the review was posted (datetime).
* Is Verified: Boolean indicating if the review is from a confirmed order.

5.2 Refer-a-Friend

* Referral ID: Unique identifier (auto-generated).
* Referrer ID: Customer ID of the referrer.
* Referred Customer ID: Customer ID of the referred user.
* Reward Points: Points awarded to referrer (integer).
* Status: Enum (e.g., Pending, Completed).
* Timestamp: When the referral was made (datetime).

Additional Notes

* Data Types: Fields specify types like string, integer, decimal, boolean, datetime, enum, array, or object for clarity.
* Required vs. Optional: Most fields are required unless marked as optional (e.g., video, notes). Validation rules should enforce this.
* Scalability: Fields are designed to support indexing (e.g., Item ID, Order ID) for fast queries in large datasets.
* Localization: Language Translations fields ensure multi-language support for global markets.
* Security: Sensitive fields (e.g., Password, API Key) must be encrypted; payment details should be tokenized via the payment gateway.
* Extensibility: Fields like Tags and Customizations are flexible to accommodate future requirements.

This list covers all necessary fields for the application’s functionality, from menu browsing to analytics. If you need a specific subset of fields (e.g., only for a particular module) or additional details (e.g., database schema design), let me know!