

# Street Food and Local Eats Database Design Document

TEAM 6

RISHAB SHAH  
HARSHITHA CHANDRASHEKAR  
YASH ZAVERI  
SUHAS SHETTY  
RAHUL POTHIRENDI

## 1. Database Purpose

The purpose of the 'Street Food and Local Eats' database is to create a comprehensive and accessible digital platform that catalogs street food vendors and local eateries in the Boston area. It aims to facilitate the discovery of diverse culinary experiences, promote small food businesses, and provide valuable insights for food enthusiasts and tourists. The database serves as a bridge connecting customers with a wide range of dining options, enriching the local food culture and economy.

## 2. Business Problem Description

The database addresses the challenge of locating diverse street food options and local eateries in Boston. It aims to support small food businesses by increasing their visibility and accessibility, helping them to attract a larger customer base. Furthermore, the database seeks to enhance the culinary experience for users by providing detailed information on various dining options, including menus, reviews, and health inspection records.

## 3. Business Rules

- ☐ Health inspections are conducted for vendors and recorded with a score and comments.
- ☐ Each **Dish** must be associated with at least one **Menu** but can be part of multiple menus.
- ☐ A **Menu** contains one or more dishes and is offered by a specific **Vendor**.

- ☐ Each **Vendor** can have multiple menus but must have at least one menu.
- ☐ A **User** can write multiple **Rating/Reviews**, but each review is associated with only one user.
- ☐ Each review is about a single vendor, and a vendor can have multiple reviews.
- ☐ A vendor operates in a specific location, and each location can host multiple vendors.
- ☐ An event organizer organizes events, and each event must be organized by an event organizer.
- ☐ Users register for events, and an event registration record is created for each user-event pair.
- ☐ A user can make multiple payments for different event registrations, but each payment is linked to only one registration.
- ☐ An event can have multiple payment types, such as credit card, cash, or cheque.
- ☐ A payment has a status, such as pending, confirmed, or cancelled.
- ☐ A vendor can offer multiple cuisines, but each cuisine is offered by only one vendor.

#### 4. Entities Explanation

**Dish:** This entity represents a food item that can be ordered by a user. It has attributes such as DishID, Name, Ingredient, and Allergen. A dish can be part of one or more menus, and a menu can contain one or more dishes. This is a many-to-many relationship, resolved by the associative entity **contains**.

**Menu:** This entity represents a collection of dishes offered by a vendor. It has attributes such as MenuID and Name. A menu is offered by one and only one vendor, and a vendor can have one or more menus. This is a one-to-many relationship, labeled as **offered by**.

**Vendor:** This entity represents a food service provider that can deliver dishes to users. It has attributes such as VendorID, Name, CuisineID, ServingHours, and Contact#. A vendor can offer one or more cuisines, and a cuisine can be offered by one and only one vendor. This is a one-to-many

relationship, labeled as **offers**. A vendor can also operate in one and only one location, and a location can host one or more vendors. This is a one-to-many relationship, labeled as **operates in**. A vendor can also be rated and reviewed by one or more users, and a user can rate and review one or more vendors. This is a many-to-many relationship, resolved by the associative entity **Rating/Review**.

**Cuisine:** This entity represents a type of food, such as Italian, Chinese, or Indian. It has attributes such as CuisineID and CuisineType. A cuisine can be offered by one and only one vendor, and a vendor can offer one or more cuisines. This is a one-to-many relationship, labeled as **offered by**.

**Location:** This entity represents a geographical area where vendors can operate and deliver food to users. It has attributes such as LocationID, Address, City, and Zipcode. A location can host one or more vendors, and a vendor can operate in one and only one location. This is a one-to-many relationship, labeled as **hosts**.

**User:** This entity represents a person who can order food, register for events, and rate and review vendors. It has attributes such as UserID, UserName, Email, Phone#, Preference, and Address. A user can rate and review one or more vendors, and a vendor can be rated and reviewed by one or more users. This is a many-to-many relationship, resolved by the associative entity

**Rating/Review.** A user can also register for one or more events, and an event can have one or more registered users. This is a many-to-many relationship, resolved by the associative entity

**Event Registration.** A user can also make one or more payments for different event registrations, and a payment can be made by one and only one user. This is a one-to-many relationship, labeled as **makes**.

**Event:** This entity represents a social gathering or a special occasion that users can register for. It has attributes such as EventID, Name, OrganizerID, StartDate, Duration, and Invitation. An event can be organized by one and only one event organizer, and an event organizer can organize one or more events. This is a one-to-many relationship, labeled as **organized by**. An event can also have one or more registered users, and a user can register for one or more events. This is a many-to-many relationship, resolved by the associative entity **Event Registration**.

**Event Registration:** This entity represents the reservation made by a user for an event. It has attributes such as RegistrationID, UserID, EventID, ReservationDate, and NumberOfPeople. An event registration is associated with one and only one user, and one and only one event. This is a one-to-one relationship, labeled as **for** and **by**. An event registration can also have one and only one payment, and a payment can be linked to one and only one event registration. This is a one-to-one relationship, labeled as **has**.

**Payment:** This entity represents the transaction made by a user for an event registration. It has attributes such as PaymentID, RegistrationID, PaymentType, and PaymentStatus. A payment is made by one and only one user, and a user can make one or more payments. This is a one-to-many relationship, labeled as **made by**. A payment is also linked to one and only one event registration, and an event registration can have one and only one payment. This is a one-to-one relationship, labeled as **for**. A payment can also have one or more payment types, such as credit card, cash, or cheque. This is a one-to-many relationship, labeled as **of**.

**DESIGN DECISION :**

Entity Name	Reason for Inclusion	Relationship with Other Entities
Dish	Represents the restaurant's menu items.	Belongs to a vendor, has a cuisine type, and can be included in menus.
Vendor	Represents companies or individuals providing dishes.	Offers dishes, has a location, and can organize events.
Location	Represents physical addresses associated with vendors and users.	Belongs to a vendor (optional) and has a user as its owner (optional).
User	Represents users of the system, including customers and organizers.	Can write reviews, register for events, and have preferred cuisines.
Cuisine	Represents different cuisine types (e.g., Italian, Thai).	Dishes belong to a cuisine.
Review	Represents customer reviews of vendors.	Written by a user about a vendor.
Event	Represents events organized by vendors.	Organized by a vendor, can have multiple registrations, and has a location.
Event Registration	Represents user reservations for events.	Made by a user for an event.
Payment	Represents payment information for event registrations.	Belongs to a user and a registration.
Menu	Represents menus offered by vendors.	Contains dishes offered by a vendor.
Health Inspection	Represents health inspections conducted at vendor locations.	Conducted at a location and has a score and comments.

**ENTITIES RELATIONSHIPS –**

Entity Name	Reason for inclusion	Relationship with other entities	Cardinality	Notes
User	Represents the users of the system.	Can have preferences, register for events, and write reviews.		
Vendor	Represents the vendors that provide the dishes.	Can offer dishes in menus and organize events.		
Dish	Represents the dishes offered by vendors.	Can be part of menus and reviewed by users.		
Menu	Represents the menus offered by vendors.	Can contain dishes and be offered by vendors.	Many to many	A menu can contain many dishes, and a dish can be part of many menus.
Cuisine	Represents the type of cuisine a dish belongs to.	Can have dishes and be preferred by users.	Many to many	A cuisine can have many dishes, and a dish can belong to many cuisines.
Location	Represents the location of a vendor.	Vendors can have a location, and users can search for vendors by location.	One to many	A vendor can have one location, but a location can have many vendors.
Inspection	Represents the health inspections conducted on vendors.	Conducted by inspectors for a specific vendor, location, and date.	One to many	An inspector can conduct many inspections, and a vendor can have many inspections.
Review	Represents	Written by users	One to	A user can write

	reviews written by users about dishes.	for dishes, with a rating and comment.	many	many reviews, and a dish can have many reviews.
Event	Represents events organized by vendors.	Can have dishes and be registered for by users.	Many to many	An event can have many dishes, and a dish can be part of many events.
Event Registration	Represents user registrations for events.	Made by users for events, specifying the number of people and payment information.	One to many	A user can register for many events, and an event can have many registrations.

## 5. Key Design Decisions

- ☐ Use of a relational database model to effectively organize and manage data.
- ☐ Inclusion of a comprehensive set of entities to cover all aspects of the street food experience.
- ☐ Implementation of a user-friendly interface for easy navigation and interaction with the database.
- ☐ Incorporation of customer reviews and ratings to enhance trust and reliability.
- ☐ Integration of location data to facilitate easy discovery of vendors.