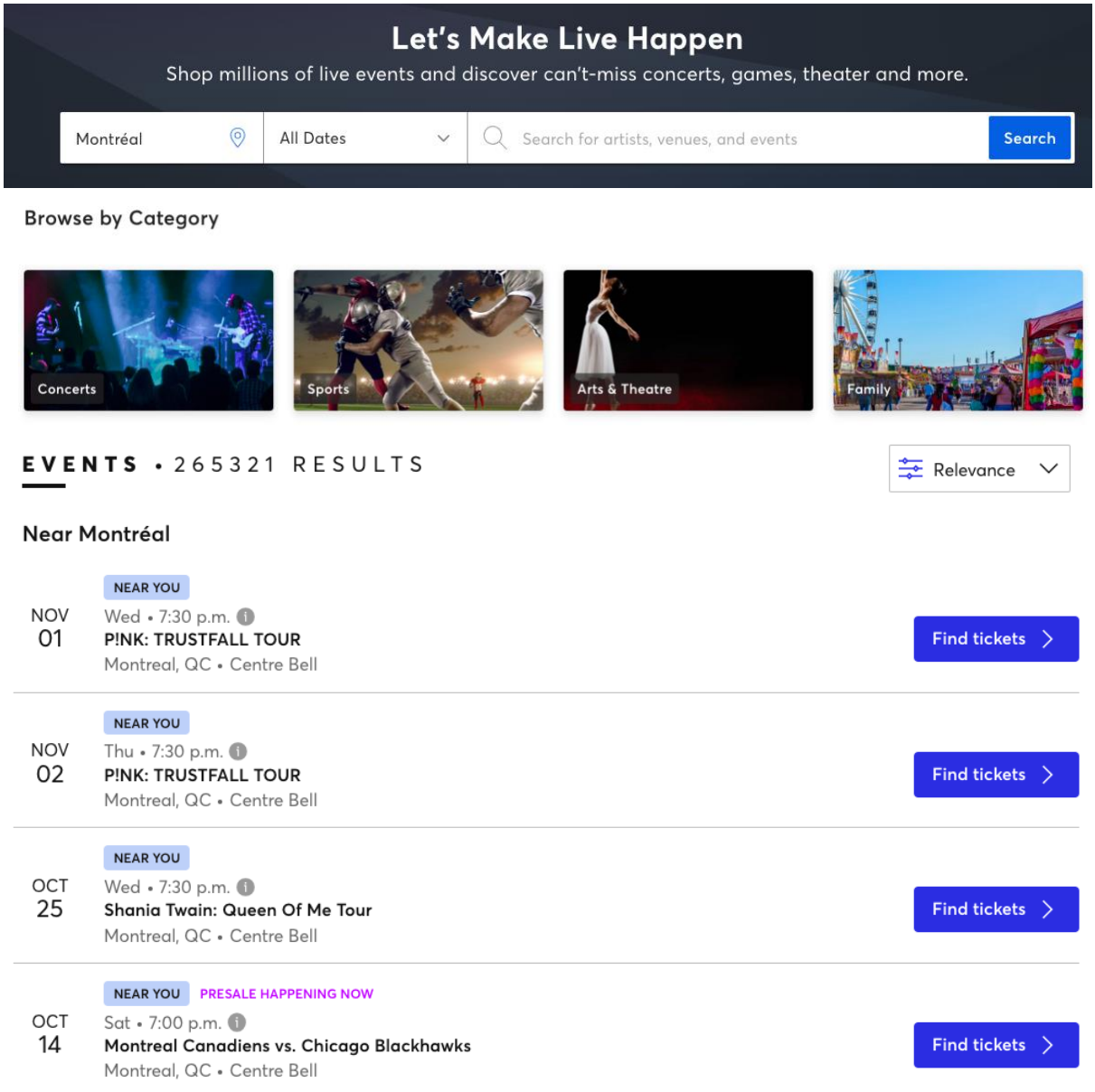
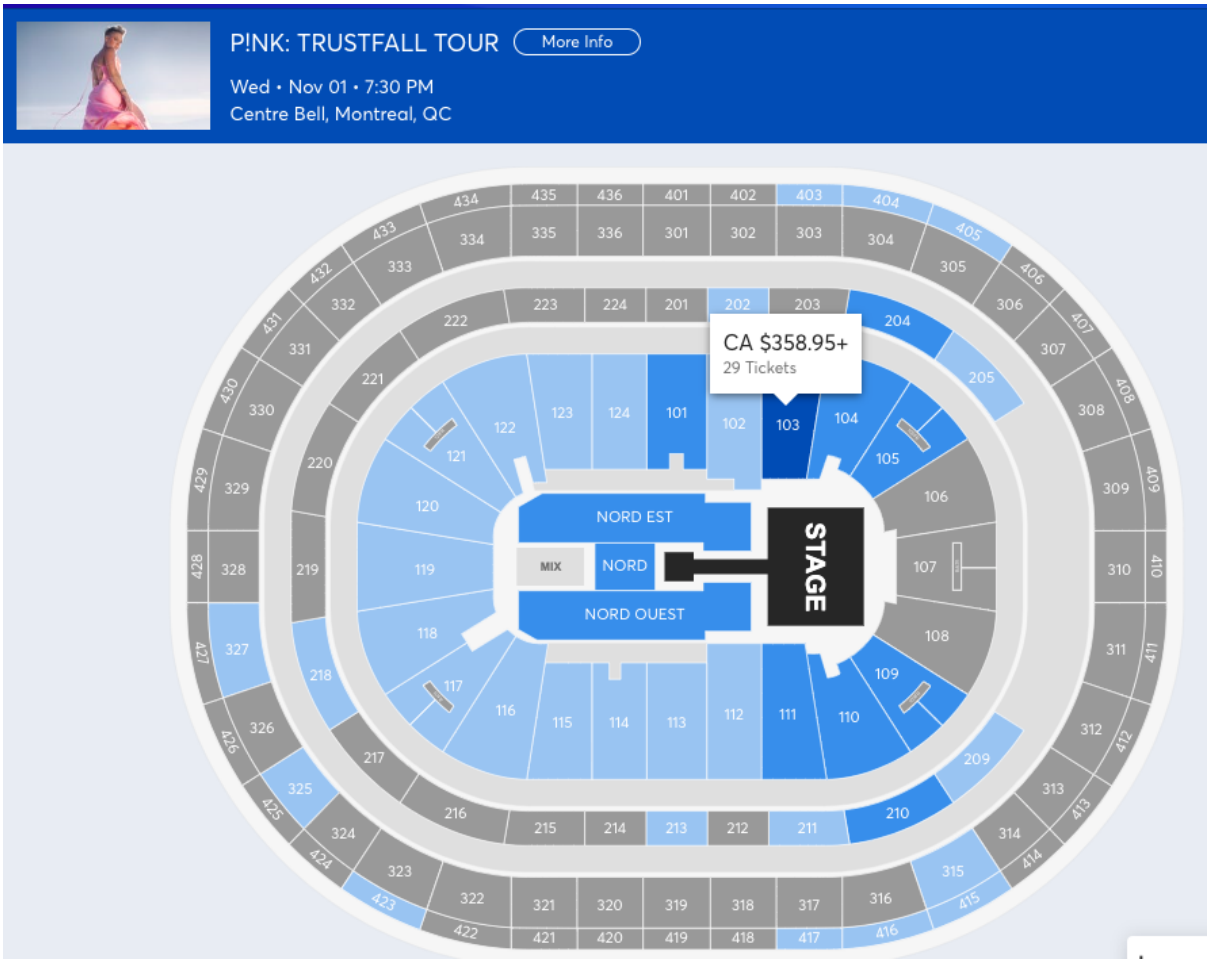
Overview of the business scenario

The business is similar to Ticketmaster, a comprehensive event management and ticketing solution platform. With an expansive reach spanning concerts, sports events, theatre productions, and more, our platform connects event clients, venues, and customers in a seamless ecosystem. Customers can explore a diverse range of events, check tickets with ease, and gain access to the events they're passionate about.





Mission statement

The purpose of the Ticketmaster-like database system is to effectively store, manage, and facilitate the exchange of event-related data. Our database fosters seamless collaboration and information sharing among clients, venues, and customers.

Mission objectives

To maintain (enter, update and delete) data on events
To maintain (enter, update and delete) data on venues
To maintain (enter, update and delete) data on address
To maintain (enter, update and delete) data on customers
To maintain (enter, update and delete) data on clients
To maintain (enter, update and delete) data on orders
To maintain (enter, update and delete) data on tickets
To maintain (enter, update and delete) data on seats
To maintain (enter, update and delete) data on reviews

To perform searches on events
To perform searches on venues

To perform searches on address
To perform searches on customers
To perform searches on clients
To perform searches on orders
To perform searches on tickets
To perform searches on seats
To perform searches on reviews

To track the status of events at venues
To track the status of orders and payments

To report on events
To report on venues
To report on address
To report on customers
To report on clients
To report on orders
To report on tickets
To report on seats
To report on reviews

Business rules:

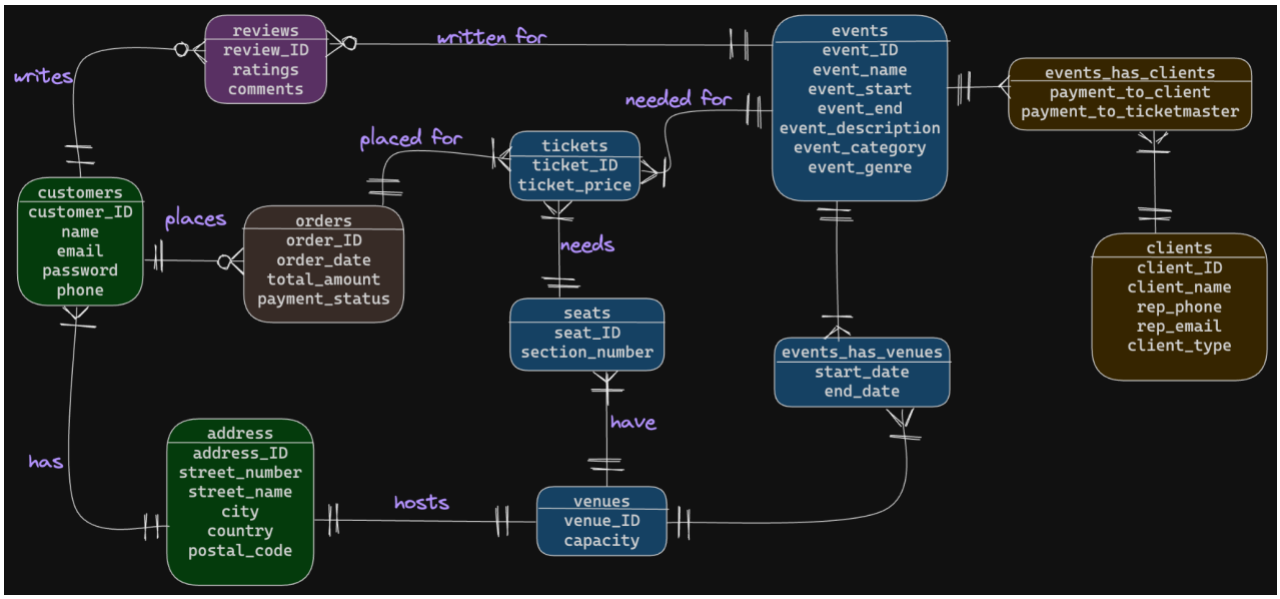
- A customer must be associated with an address.
- An event must have at least one venue and can have multiple venues.
- An event can have multiple reviews from different customers.
- An order must be associated with a customer, and each order can have multip le tickets.
- A ticket must be associated with an event, a seat, and an order.
- Clients must have events associated with them, and each event must have pay ments to the client and Ticketmaster.
- Events must have clients, indicating partnerships between clients and event s.
- Events must have a start and end date, and venues must host events during s pecific time periods.

Business assumptions:

- Customers have a single address associated with them (billing address), ass uming customers cannot have multiple addresses. But multiple customers can have the same address.
- The system assumes that events can take place in multiple venues and across different categories and genres.

- Ticket prices are stored at the ticket level, assuming they remain constant regardless of the seat or order.
- Clients are assumed to be external entities or event organizers who collaborate with the event management system.
- The system assumes that events, orders, and tickets are linked together accurately for tracking purposes.
- Reviews are associated with customers and events, assuming customers can only leave reviews for events they attended.

ERD



Data Dictionary

- Description of Entities

Entity Name	Description	Aliases	Occurrence
events	Contains the information needed to identify a particular event held	activities	1. One event can have many reviews; 2. Many events can be held in many venues (has "events_has_venues" relationship attribute); 3. Many events can be held by many clients (has "events_has_clients" relationship attribute)
venues	Contains the information needed to identify a particular venue tha	sites	1. One venue can have many Seats; 2. One venue can only have

	t can hold events		one address; 3. Many venues can hold many events (has “events_has_venues” relationship attribute)
address (we refer to billing address here specifically)	Address information that can be used for both customers and venues	location	1. One address can only have one venue; 2. One address can have many customers (who may live together)
customers	Contains the customer’s personal information who registered with our business	users	1. Many customers can have one address (may live together); 2. One customer can place many orders; 3. One customer can post many reviews
clients	Client information about who is holding their events and selling tickets through our business	guests	1. Many clients can hold many events
orders	Contains the order information of customers with dates, status and payment, etc.	purchase	1. Many orders can belong to one customer; 2. One order can contain many tickets
tickets	Contains the ticket information such as the price of an event for customers	pass	1. Many tickets can be in one order; 2. Many tickets can belong to one event; 3. Many tickets can have one Seat (for different events at the same venue)
seats	Contains information about the seats of customers at an event	spots	1. Many seats can be in one venue; 2. One seat can be in many tickets (for different events at the same venue)
reviews	An entity that holds r	feedback	1. Many reviews can be pos

	atings and comments made by the customer for events		ted for one event; 2. Many reviews can be posted by one customer

- Description of Attributes

(Please see the Google sheet submitted thank you)

Relational Schema

address(address_ID, street_number, street_name, city, country, postal_code)
Primary Key: address_ID

customers(customer_ID, name, email, password, phone, address_ID)
Primary Key: customer_ID
Foreign Key: address_ID References address(address_ID)

events(event_ID, event_name, event_start, event_description, event_end, event_category, event_genre)
Primary Key: event_ID

reviews(review_ID, rating, comments, customer_ID, event_ID)
Primary Key: review_ID
Foreign Key: customer_ID References customers(customer_ID)
Foreign Key: event_ID References events(event_ID)

orders(order_ID, order_date, total_amount, payment_status, customer_ID)
Primary Key: order_ID
Foreign Key: customer_ID References customers(customer_ID)

venues(venue_ID, name, capacity, address_ID)
Primary Key: venue_ID
Foreign Key: address_ID References address(address_ID)

seats(seat_ID, section_number, venue_ID)
Primary Key: seat_ID
Foreign Key: venue_ID References venues(venue_ID)

tickets(ticket_ID, ticket_price, event_ID, order_ID, seat_ID)
Primary Key: ticket_ID
Foreign Key: event_ID References events(event_ID)
Foreign Key: order_ID References orders(order_ID)
Foreign Key: seat_ID References seats(seat_ID)

clients(client_ID, client_name, rep_phone, rep_email, client_type)
Primary Key: client_ID

events_has_clients(event_ID, client_ID, payment_to_client, payment_to_ticketmaster)
Primary Key: event_ID, client_ID
Foreign Key: event_ID References events(event_ID)
Foreign Key: client_ID References clients(client_ID)

events_has_venues(event_ID, venue_ID, start_date, end_date)
Primary Key: event_ID, venue_ID
Foreign Key: event_ID References events(event_ID)
Foreign Key: venue_ID References venues(venue_ID)