Name: Mahmoud Mansour

Course: CIS 344 Spring 2026

Professor: Yanilda Peraita Ramos

Date: 02/15/2026

System Design Requirements for my Mini World project (Adventure Tour Company)

For my individual project, I selected adventure tour company. I chose this mini world because it represents a realistic service based business that involves customers, employees, bookings, tour packages, and locations.

This scenario fits the scope of the project because it includes multiple entities with clear relationships, such as customers making bookings, guides leading tours, and tour packages including multiple landmarks. It is not too simple, but also not overly complex. This will help me practice most of the skills learned in class.

After selecting my project from the list provided, I carefully look for a company that operates in NYC and serves tourist daily that reflects a real world business environment where data management is important for scheduling, pricing and customer tracking. To gather requirements for the NYC Adventure Platform database, I conducted online research by carefully reviewing the company's official website (https://www.nyc-adventure.com/) and its TripAdvisor page. I examined the booking interface, tour descriptions, pricing structure, itinerary details, and customer reviews. Instead of just reading the information, I walked through the booking process step by step as if I were a customer. This helped me understand how the system functions from a user perspective.

From the booking page, I identified that customers must provide contact information before completing a reservation. This led me to create the CUSTOMERS entity. I also observed that the company offers multiple tour options with different durations and prices, such as 30-minute, 1-hour, and 2-hour tours. This helped me define the TOUR_PACKAGES entity.

Customer reviews frequently mentioned specific tour guides by name, such as Abdul and Jay. This showed that guides play an important role in the service experience. Based on this, I decided to create a GUIDES entity to allow each booking to be assigned to a specific guide.

The itinerary descriptions listed multiple landmarks visited during each tour, such as Bethesda Fountain and Bow Bridge. Since one tour visits many landmarks and one

landmark appears in multiple tours, I determined this is a many-to-many relationship. To handle this properly, I created a bridge table called TOUR_ITINERARY.

Through analyzing pricing information, I determined that price must be stored using a DECIMAL data type to ensure accuracy. I also implemented constraints such as NOT NULL on required fields and UNIQUE on customer email addresses to maintain data integrity.

By carefully reviewing the company's public information and mapping out how bookings flow from customer selection to tour completion, I was able to design a structured database that reflects the real-world operation of NYC Adventure Platform.

Database Entity-Relationship Summary

- **Customers & Bookings**: A **One-to-Many** relationship; one customer can book multiple tours over time, but each booking belongs to a specific customer.
- **Guides & Bookings**: A **One-to-Many** relationship; each booking is assigned to one guide (like Abdul or Jay) to ensure accountability and scheduling.
- **Guides & Languages**: A **Many-to-Many** relationship managed via the guide_languages bridge table, allowing the system to track all languages a guide speaks (e.g., English and Arabic).
- **Tour Packages & Itineraries**: A **One-to-Many** relationship; a single tour package (like the "2-hour Movie Spot Tour") follows a specific itinerary of multiple landmarks.
- **Landmarks & Itineraries**: A **Many-to-Many** relationship where landmarks (like Bethesda Fountain) can be part of multiple different tour packages.