**Database Design and Initial ERD**

**Group Number:** 5

**Group Members:** Ashwini Somani, Ankit Singh, Chintan Agrawal, Manisha Dashrath Nakawe, Nidhi Dahiya, Priya Saraf

**Database Specification: Purpose, Business Problems Addressed & Business Rules**

**Database Purpose:** The purpose of this application is to automate the existing manual system by the help of computerized equipment and a software that is fulfilling their requirement so their information can be stored for a longer period along with ease of access to that information. Databases are specially built software applications that record and analyze data through interacting with the user, other applications, and the database itself. We have an event management database for event planners. This database manager will have access to all information about all events that have occurred and will occur in the future.

The majority of 3NF tables have no insertion, update, or deletion abnormalities. This project offers a software that allows us to examine which of a table's normal form dependencies are satisfied. The goal of this project is to create a database for event management that organizes and saves all available event-related data. To determine which Normal Form a relation's dependencies, belong to.

This database is especially designed for event managers. The event manager will be able to feed in keep record and project reports of his work. It provides a user-friendly approach for handling all the services. Some of the important features of the project are Events, Reports, Vendor.

**Business Problem Addressed:** When managing many events at once, event organizers (usually on a small scale) have difficulty keeping track of the requirements of each event. Special staff are also assigned to keep track of the events' records. The project offers a simple and automated solution that saves time, money, and mental energy. The project enables real-time data feeding into a safe location, preventing data loss in offline tracking due to physical mishaps and saving time spent on unnecessary paperwork. This project was created to address issues with current systems that make them difficult to use. The heads of event management companies, managers, and staff are among the system's users.

1. An easy-to-use platform for connecting with the organizers will be provided by the Event Management system. Because all information will be stored in a database and clients will access it according to their needs, clients will be able to choose from pre-defined packages or customize packages to meet their specific needs.
2. The event management system not only digitalizes the booking of orders and provides client convenience, but it also digitalizes the firm's functions. Employees will be able to digitalize all their personal records, while managers will be able to add, update, and remove employees, as well as examine orders, vendors, and other information.
3. The project ensures real time feeding of data in a secure place which avoids data loss in offline logging due to any physical mishap and save time wasted in unnecessary paperwork.

**Business Rules:**

1. One vendor can provide only one service at a time. There should not be overlapping of services. For different services different vendor id will be generated.
2. One booking can have only one event. One event can have multiple bookings. But one booking can not have more than one event.
3. One payment can be done against one invoice. There can not be multiple payments for one invoice.
4. One Customer can choose from a list of packages and accordingly through vendors services will be booked. Customers can not do customization of events.

**Design Decisions**

|  |  |  |  |
| --- | --- | --- | --- |
| **SR No.** | **Entity  Name** | **Why Entity is included** | **How Entity is related to other entities** |
| 1 | Manager | The event manager will be able to feed in, keep record, and project reports of his work. It provides a user-friendly approach for handling all the services and keeping track of employees. | Manager entity is related to the employee data to keep track of different employees under him and their contact information as well as salaries. It has a one to many relation with it since one manager will have multiple logistical employees working for him. |
| 2 | Employee | Various employees are appointed to an event having first name, last name, unique employee ID, salary, phone number and their email ids.  These employees could be either managers or a part of the logistics team.The key function here is to keep track of customers, managers and the employees | Employee is connected to managers and the customer table with a one to many relationship.Its main functionality is to store the employee data and relate it to the pertaining manager. |
| 3 | Customer | Capturing the customer details is done through the customer login interface to store their details as well as their geolocation.Customer table shows customers who have booked various events and have their contact details and their login details.The main reason the table was created is to store customer information and the employee that is in contact with them throughout the event to keep the manager abreast of which customer has booked what kind of event.Its also used in collecting their feedback, payments and bookings\_ID. | This table is one of three main data storages  that contains customer details and is related to their payments, bookings, employee and feedback databases. It has a one to many relation with bookings, payment and feedback while having a many to one relation with the employee database using Employee\_ID and Customer\_ID. |
| 4 | Venue | Venue provides a list of all the Venues present in our Database for an event along with its details. Venue is provided for an event to accommodate guests.  Vendor attribute includes Venue\_Id, Vendor\_ID, Name, Capacity, Address, Rent, Phone | Venue table has zero to one, and many to one relationship with the Vendor table using Vendor\_Id. All relationships are non-identifying.  Services can be provided by many vendors in the same location or in multiple venues. |
| 5 | Vendor | The Vendor table keeps the list of all the Vendors that provide different services in our events. It has the following attributes:  Vendor\_Id, Vendor\_Type , Firstname , Lastname, Phone , Address , Email | The Vendor table has-  - one to zero, and one-to-many relationship with the Venue table. Connected through Vendor\_Id.  - one to zero, and one-to-many relationship with the Performer table. Connected through Vendor\_Id.  - one to zero, and one-to-many relationship with the Photographer table. Connected through Vendor\_Id.  - one to zero, and one-to-many relationship with the Decoration table. Connected through Vendor\_Id.  - one to zero, and one-to-many relationship with the Transportation table. Connected through Vendor\_Id.  - one to zero, and one-to-many relationship with the Equipment table. Connected through Vendor\_Id.  - one to zero, and one-to-many relationship with the Makeup Artist table. Connected through Vendor\_Id.  All relationships are non-identifying. |
| 6 | Event | Event table gives information about all the all the types of events organized. Every event has its own type, and a unique event ID.  Event\_ID, Event\_type | Event table has one-to-one and one-to-many relationship with the Packages table. Connected by Event\_Id.  It has one-to-one and one-to-zero relationship with the Customer Feedback table. Both are connected by Event\_Id.  It has one-to-one and one-to-many relationship with the Bookings table. Both are connected by Event\_Id.  All relationships are non-identifying. |
| 7 | Packages | Describes the type and name of the package for a given event id.  Package\_Id,Event\_Id,Package\_Name, Package\_Charges |  |
| 8 | Transportation | Transportation provided to an event is defined by the type of vehicle the customer requires, cost per km and the number of seats that vehicle has. Transportation is provided by the Transport Company with name, id, phone number, address.  Transportation attribute include Vendor\_ID , cost\_per\_km, no\_of\_seats, vehicle\_name, License plate, |  |
| 9 | Performer | For entertaining the guests in the event, different performances are held. These performances are performed in the event by performers who have their unique ID, name, salary, mail id and address.  Performer attributes include Vendor\_ID , Performer\_ID, P\_Category, P\_Firstname, P\_Lastname, P\_Add, P\_mail\_id, P\_charges |  |
| 10 | Caterers/Food | Events orders for the food of different types and their respective cost which is supplied by caterers who have their unique id, name, address, mail id.  Caterer\_ID, Vendor\_ID, Cat\_FirstName, Cat\_LastName, Cat\_Mail\_ID,  Cat\_Package, Cat\_Charges. |  |
| 11 | Decoration | For decoration purposes there are different types of decoration and their respective costs which will be supplied by different decorators having name, unique ID, mail id and address.  Decor\_ID, Vendor\_ID, Decor\_FirstName, Dc\_LastName, Dc\_Mail\_ID,  Dc\_Package, Dc\_ Charges. |  |
| 12 | Electric Equipment | Every event needs various Electronic Equipments containing name, cost, unique ID, which are supplied by various electronic companies having their unique ID, Address, email id, name.  Eq\_ID, Vendor\_ID, Eq\_FirstName, Eq\_LastName, Eq\_Mail\_ID,  Eq\_Package, Eq\_Charges. |  |
| 13 | Bookings | Customers as well as managers should be able to see bookings of events to track which event is popular with customers.  Booking table will have Booking\_Id, Customer\_Id, Package\_Id, Event\_Id. | Bookings table is the intermediate table which keeps track of events booking. Booking is an associative entity. One event can have many bookings. And one booking id will be only one for an event.So booking and event is one to one relationship. |
| 14 | Makeup Artist | Some events like stage plays, marriages require makeup. There are different make up services which will be provided and each of the makeup services has its own cost and type. The makeup artists are from parlors, since there are different parlors, hence each of them has its own specific name, unique id, mail id and address.  Artist\_Id, Vendor\_Id, Firstname, Lastname, Email id, Package, Charge. | Makeup Artist is a service table which is connected to Vendor table by  many to one i.e. One vendor can have multiple Makeup Artists. |
| 15 | Photographer | Since moments in the event must be captured, these are done by the photographers who cover the entire event. Each photographer has mail id, address, pay, ID, first name, last name for further contact.  Photographer \_Id, Vendor\_Id, Firstname, Lastname, Emai\_ld, Package, Charge. | Photographer is a service table  which is connected to Vendor table by  many to one i.e. One vendor can have multiple photographers. |
| 16 | Customer Feedback | To track and improve the overall experience of the customers.  Event\_ID, Customer\_ID, Feedback. | The Feedback table is  connected to the customer table with one to one relationships and event table with one to one relationship. When the event is over, customers can leave feedback about the event . Feedback can be referenced by the foriegn key customer\_id and event\_id in the feedback table. |
| 17 | Payment | Payment table records each payment made by the customer ,  table attribute includes  Customer\_id, Invoice\_id, Event\_id, ModeOfPayment, Pay\_Date. | Payment\_id primary key used to uniquely identify each payment. Customer\_id is foriegn key reference to customer table . Payment table which is connected to Customer table by many to one relationship i.e one customer can have multiple payment . |
| 18 | Invoice | Invoice table records each payment status and amount made by the customer. Invoice\_Id, Customer\_Id Amount, Inv\_Date, | Invoice id is the primary key used to uniquely identify each invoice. Invoice table which is connected to payment table by one to one relation . |