DNA Assignment 1

Team no - 10

Team Members: Prisha, Vanshika Dhingra, Aaryan Ajay Sharma

Introduction to Miniworld

Our mini world (primarily derived from the “7 million contract scene “ in John Wick part 2) is a database for an assassination management organisation. In that scene, the client requests an operator for an account to be opened in the name of the assassinee (here, ‘account’ is used interchangeably with contract). The account details contain the contract’s status, the person to be assassinated and the denomination for the same, which the client specifies. The operator further conveys the message to the hitman(s) who take up the job.

Purpose Of Database

The purpose of this database is:

- To provide a general synopsis of assassinations based on the type and status of contract
- To safely and reliably store the information regarding the contracts, hitman, assassinee and client.
- Organise data intuitively and logically for easy and quick access

Users of the database

ADMIN: The database owner who keeps track of the progress of management and updates the database.

CLIENT: Clients are end users of the database. They can view the database to see which hitmen are allotted their tasks and know the status of the assassinee.

OPERATOR: The operator’s job is to respond to the client’s request and make an account, depending upon the requirements.

Applications of the database

The database can be used by

- The client to view the information about the various assassins assigned to kill the assassinee.
- The client to see whether the assassination has been completed or not.
- Operator to see which hitmen are occupied so that messages of the next task can be sent accordingly.
- Which weapon did the hitman use to kill the assassinee and rank the hitmen based on the number of successful assassinations they made by that time.
- To keep track of management's investment.

Database requirements

Assumptions

- A hitman can take up multiple jobs at once or have no job at all.
- An assassinee can have one or more hitmen assigned to kill them.
- An operator is someone who opens an account for the client.
- An account will be uniquely determined by its contract ID.
- The account name will be the name of the assassinee.
- Clients will have to verify their identity while opening an account by giving a verification ID.
- There will be states of the accounts opened which can be:
  - Open - The contract is open for all the hitmen.
  - Closed - The contract is open for only the hitmen specified by the client.
- Clients will have to specify the denomination of the account opened.
- Clients can specify the name(s) of the hitman.
- A client is someone who will give hit money to an operator to assassinate.
- All of the people involved (Operator, Client, hitmen) will have a phone number which will consist of:
  - Country code
  - Phone number - Country code

Strong Entity Types

- **Hitman**
  - Name: string → Composite
    - First Name → Not Null
    - Middle Name → Null
    - Last Name → Null
  - Height(in metres): float → Not Null
  - *Age: Int* → Not Null, Derived
  - Weight (in kgs): Int
  - Hitman ID: Int → 6 digit Int, Primary Key
  - Phone Number: string → Not Null, Multivalued
  - Address: string → Composite
    - Street name → Not Null
    - Address line 1 → Not Null
    - Address line 2 → Null
    - Pincode → Not Null
  - Weapon (To be described in detail): string → Not Null
  - Date of Birth: string → Composite, Not Null
    - Date of birth
    - Month of birth
    - Year of birth

    - Kill Count: Int → Not Null
    - Occupancy Status: Boolean → enum Int {0, 1}
- **Operator**
    - Operator ID: Int → 3 digit Int, Not Null, Primary Key
    - Name: string → Composite
        - First Name → Not Null
        - Middle Name → Null
        - Last Name → Null
    - Date of Birth: string → Composite, Not Null
        - Date of birth
        - Month of birth
        - Year of birth
    - *Age: Int* → Not Null, Derived
    - Phone number: string → Not Null, Multivalued
        - Country Code → Not Null
        - Phone number - Country code → Not Null
- **Client**
    - Client ID: Int → 5 digit Int, Not Null, Primary Key
    - Name: string → Composite
        - First Name → Not Null
        - Middle Name → Null
        - Last Name → Null
    - Phone number: string → Not Null, Multivalued
        - Country Code → Not Null
        - Phone number - Country code → Not Null
    - Collateral: string → Not Null, Multivalued
- **Contract**
    - Name: string
    - Type of contract: string
    - Assassinee Name: string → Composite
        - First Name → Not Null
        - Middle Name → Null
        - Last Name → Null
    - Denomination: Int (USD) → Not Null
    - Assassinee Address: string → Composite
        - Street name → Not Null
        - Address line 1 → Not Null
        - Address line 2 → Null
        - Pincode → Not Null
    - Contract date/time —> Composite
        - Time → Not Null, string of type HH:MM:SS (Hours, Minutes, Seconds)
        - Date → Not Null, string of type DD/MM/YYYY (Date, Month, Year)
    - Contract ID: Int → 6 digit Int, Primary Key
    - HitmanID(s): Int → Not Null, 6 digits Int, Multivalued
    - Status (completed/ongoing): boolean → enum Int {0, 1}

Weak Entity Types

- **Assassinee** (as described by the client)
    - Name: string → Composite
        - First Name → Not Null
        - Middle Name → Null
        - Last Name → Null
    - Height (in metres): float → Not Null
    - *Age: Int* → Not Null
    - Assassinee Address: string → Composite
        - Street name → Not Null
        - Address line 1 → Not Null
        - Address line 2 → Null
        - Pincode → Not Null
    - Physical Description: string → Not Null

Relationship Types

- ***assassinates***
    - Degree: 2 (Binary)
    - Entity Types: **Hitman** → **Assassinee**
    - Min-max ratio
        - Hitman → (0, 1)
        - Assassinee → (1, N)
- ***gives contract***
    - Degree: 2 (Binary)
    - Entity Types: **Client** → **Operator**
    - Min-max ratio
        - Client → (0, 1)
        - Operator → (0, N)
- ***assigns contract***
    - Degree: 2 (Binary)
    - Entity Types: Operator → Hitman
    - Min-max ratio
        - Operator → (1, N)
        - Hitman → (1, 1)
- ***works for***
    - Degree: 2 (Binary)
    - Entity Types: Hitman → Client
    - Min-max ratio
        - Hitman → (0, 1)
        - Client → (0, N)

- ***accomplice in assassination***
    - Degree: 3 (Ternary)

- Entity Types: Operator, Client, Hitman
- Min-max ratio
  - operator → (1, N)
  - client → (1, 1)
  - Hitman → (1,1)

Functional Requirements

- **Modification**
  1. **INSERT:**
     - account details
     - hitman details.
  2. **UPDATE:**
     - status of assassination by a hitman
     - status of occupancy of hitman

- **Retrievals**
  1. **SELECTION:**
     - Retrieve the account whose type of contract is open.
     - Retrieve the hitman who is not occupied
  2. **PROJECTION:**
     - Retrieve the phone numbers of all the hitmen
     - Retrieve the hit money of all the contracts
  3. **SEARCH**:
     - Searching for all the bid amount greater than a particular value.
  4. **AGGREGATE:**
     - Calculate the total money earned by the assassinating company
     - Get the contract with the minimum bid or Calculate the average value of the bid.
  5. **ANALYSIS:**
     - Analysing which age group and gender are most targeted upon
     - Analysing which hitmen had maximum and minimum assassins, respectively

Summary

We have created a database for an assassination management agency whose primary purpose is to store data safely and reliably for easy and quick retrieval. There are many applications, like retrieving different rows and columns based on the requirements.