Database Project Report

Author: Your Name

Date: 2024-07-18

Database Project Report

1. Description of what the database does

This database manages a game world where players, NPCs (non-player characters), and quests interact.

Players can join guilds, complete quests, and earn achievements that unlock dimensions.

NPCs can belong to guilds and initiate quests. The database ensures data integrity and proper relationships

between entities using triggers and constraints.

2. Queries

Triggers:

1. enforce_single_guild_affiliation: Ensures each guild has only one affiliation by checking the count of existing affiliations before inserting a new one.

Queries:

1. ViewNPCGuilds: Retrieves and displays NPCs along with their guild names, defaulting to an empty string if they do not belong to a guild.

3. Conceptual schema

Database Project Report

Entities and Attributes:

Entity	Description
Player	Represents players in the game.
NPC	Represents non-player characters.
Guild	Represents guilds.
Quest	Represents quests.
Achievement	Represents achievements.
Dimension	Represents dimensions.
Item	Represents items.
Player_Item	Represents items owned by players.

Entity (0/1;1/n)	Relationship	Entity (0/1;1/n)
Player (0/1)	Complete	Quest (1/n)
Player (0/1)	Talks	NPC (1/n)
Player (0/1)	Belong	Guild (1/n)
Player (0/1)	Own	Player_Item (1/n)
Player (0/1)	Travel	Dimension (1/n)
Guild (0/1)	Affiliated	NPC (1/n)
NPC (0/1)	Initiate	Quest (1/n)
Quest (0/1)	Check Achievement	Achievement (1/n)
Quest (0/1)	Check Player_Item	Player_Item (1/n)
Player_Item (0/1)	Check Achievement	Achievement (1/n)
Player_Item (0/1)	Refers	Item (1/n)
Achievement (0/1)	Unlocks	Dimension (1/n)

4. Detailed Redundancy Analysis

1. Player_Item and Quest:

- Redundancy in Checking Items and Achievements:

  - Quest table includes attributes to check player items and achievements.

- Player_Item also checks achievements, leading to redundancy.

- Separate tables are used to check relationships between quests, items, and achievements, causing overlapping data.

- Overlapping Attributes:

- Quest and Check_Achievement both reference achievement requirements for quests.

- Player_Item and Check_Achievement both reference player item checks.

2. Guild and NPC:

- Redundant Relationships:

- Guild-Affiliated-NPC and Player-Belong-Guild relationships overlap.

- Information about guild affiliation is duplicated.

5. Removal/addition caused by redundancy analysis

- Removal of Overlapping Attributes:

- Merged the relationships between Quest, Player_Item, and Check_Achievement into a unified structure.

- Simplification of Relationships:

- Unified guild affiliation information to avoid redundancy.

Resulting Conceptual Schema:

Entities and Attributes:

Database Project Report

Entity	Description
Player	Represents players in the game.
NPC	Represents non-player characters.
Guild	Represents guilds.
Quest	Represents quests.
Achievement	Represents achievements.
Dimension	Represents dimensions.
Item	Represents items.
Player_Item	Represents items owned by players.

Entity (0/1;1/n)	Relationship	Entity (0/1;1/n)
Player (0/1)	Complete	Quest (1/n)
Player (0/1)	Talks	NPC (1/n)
Player (0/1)	Belong	Guild (1/n)
Player (0/1)	Own	Player_Item (1/n)
Player (0/1)	Travel	Dimension (1/n)
Guild (0/1)	Affiliated	NPC (1/n)
NPC (0/1)	Initiate	Quest (1/n)
Quest (0/1)	Check Achievement	Achievement (1/n)
Quest (0/1)	Check Player_Item	Player_Item (1/n)
Player_Item (0/1)	Check Achievement	Achievement (1/n)
Player_Item (0/1)	Refers	Item (1/n)
Achievement (0/1)	Unlocks	Dimension (1/n)

6. Logical schema

Entities and Attributes:

Entity	Attributes
Player	player_id (INT, PK), player_name (VARCHAR), guild_id (INT, FK)
NPC	npc_id (INT, PK), npc_name (VARCHAR), guild_id (INT, FK)
Guild	guild_id (INT, PK), guild_name (VARCHAR)
Quest	quest_id (INT, PK), quest_name (VARCHAR), state (BOOLEAN)
Achievement	achievement_id (INT, PK), achievement_name

	(VARCHAR)
Dimension	dimension_id (INT, PK), dimension_name (VARCHAR)
Item	item_id (INT, PK), item_name (VARCHAR)
Player_Item	player_item_id (INT, PK), player_id (INT, FK), item_id (INT, FK), state (BOOLEAN)
Relationship	**Attributes**
Complete	player_id (INT, FK), quest_id (INT, FK)
Talks	player_id (INT, FK), npc_id (INT, FK)
Belong	player_id (INT, FK), guild_id (INT, FK)
Own	player_id (INT, FK), player_item_id (INT, FK)
Travel	player_id (INT, FK), dimension_id (INT, FK)
Affiliated	guild_id (INT, FK), npc_id (INT, FK), affiliation (VARCHAR)
Initiate	npc_id (INT, FK), quest_id (INT, FK)
Check_Achievement	quest_id (INT, FK), achievement_id (INT, FK), quest_id (INT, FK), player_id (INT, FK), state (BOOLEAN)
Check_Player_Item	player_id (INT, FK), item_id (INT, FK)
Refers	player_item_id (INT, FK), item_id (INT, FK)
Unlock	achievement_id (INT, FK), dimension_id (INT, FK)

7. Normalization of logical schema

First Normal Form (1NF):

- Player: Each player has a unique player_id, player_name, and guild_id.

- NPC: Each NPC has a unique npc_id, npc_name, and guild_id.

- Guild: Each guild has a unique guild_id and guild_name.

- Quest: Each quest has a unique quest_id, quest_name, and state.

- Achievement: Each achievement has a unique achievement_id and achievement_name.

- Dimension: Each dimension has a unique dimension_id and dimension_name.

- Item: Each item has a unique item_id and item_name.

- Player_Item: Each player item has a unique player_item_id, player_id, item_id, and state.

Second Normal Form (2NF):

- Player: player_name and guild_id are fully dependent on player_id.
- NPC: npc_name and guild_id are fully dependent on npc_id.
- Guild: guild_name is fully dependent on guild_id.
- Quest: quest_name and state are fully dependent on quest_id.
- Achievement: achievement_name is fully dependent on achievement_id.
- Dimension: dimension_name is fully dependent on dimension_id.
- Item: item_name is fully dependent on item_id.
- Player_Item: player_id, item_id, and state are fully dependent on player_item_id.

Third Normal Form (3NF):

- Player: No transitive dependencies exist.
- NPC: No transitive dependencies exist.
- Guild: No transitive dependencies exist.
- Quest: No transitive dependencies exist.
- Achievement: No transitive dependencies exist.
- Dimension: No transitive dependencies exist.
- Item: No transitive dependencies exist.
- Player_Item: No transitive dependencies exist.