

The background of the slide is a circular cutout of a scene from the game Fallout Shelter. It shows the exterior of a vault entrance, which is a large, rusted metal structure with a prominent circular air filter on the left and a smaller one on the right. The entrance is set in a dark, rocky, and somewhat desolate landscape under a cloudy sky. The overall tone is dark and gritty, characteristic of the Fallout series.

Database Vault 101

A Database Administration
Project for a Fallout Shelter
vault.

PEDRO MAGALHÃES

DATABASE VAULT 101

Database project introduced in my portfolio as a demonstration of my SQL skills, focusing on my future DBA internship.

BRASÍLIA - DF

2023

Summary

1. INTRODUCTION	4
2. ENTITY-RELATIONSHIP MODEL	4
Entity list:.....	4
Attributes list:	5
Relationships list:.....	6
3. STATEMENTS.....	7
Functions	7
Stored Procedures (for strong entities)	12
Stored Procedures (for associative entities).....	20
4. CONCEPTUAL SCHEMA DESIGN	21
5. LOGICAL SCHEMA DESIGN.....	24

1. INTRODUCTION

Fallout Shelter is a strategy game where you play as a vault manager. Your mission is to keep dwellers alive, explore the post-apocalyptic world and manage resources. In this game a lot happens, a lot of data is generated, and that's where the project is involved.

The propose of db_vault_101 ('101' has no special meaning) is to store events and resources, using stored procedures, functions and analyzing this data into charts and graphs (using Power BI).

2. ENTITY-RELATIONSHIP MODEL

Entity list:

- Decease.
- Decease_type.
- Diary.
- Dweller.
- Dweller_quest.
- Exploration.
- Incident.
- Incident_type.
- Junk.
- Kinship.
- Outfit.
- Pet.
- Quest.
- Room.
- Room_type.
- Sale.
- Scrap.
- Special.
- Storage.
- Storage_exploration.
- Storage_scrap.
- Storage_quest.
- Weapon.

Attributes list:

- Decease: id_decease, date.
- Decease_type: id_decease_type, description.
- Diary: id_diary, date, dweller_count, money, room_count, quest_count, incident_count, outfit_count, weapon_count, pet_count, energy, water, food.
- Dweller: id_dweller, name, gender, level, status, id_special, id_outfit, id_weapon, id_pet, id_kinship, id_decease, id_room.
- Dweller_quest: id_dweller, id_quest.
- Exploration: id_exploration, duration, id_dweller.
- Incident: id_incident, date, id_incident_type, id_room.
- Incident_type: id_incident_type, description.
- Junk: id_junk, name, quantity, id_storage.
- Kinship: id_kinship, id_mother, id_father.
- Outfit: id_outfit, name, quantity, id_storage.
- Pet: id_pet, name, description, quantity, id_storage.
- Quest: id_quest, description, date_begin, date_end.
- Room: id_room, size, floor, id_room_type.
- Room_type: id_room_type, name, job.
- Sale: id_sale, id_product, quantity, profit, date.
- Scrap: id_scrap, id_product, scrap_quantity, date.
- Special: id_special, strength, perception, endurance, charisma, intelligence, agility, luck, category.
- Storage: id_storage, category.
- Storage_exploration: id_storage, id_exploration, quantity.
- Storage_scrap: id_scrap, id_storage, junk_quantity.
- Storage_quest: id_storage, id_quest, quantity.
- Weapon: id_weapon, name, damage, quantity, id_storage.

Relationships list:

- Decease - has - Decease_type.
- Decease - has - Incident_type.
- Diary - counts - Dweller.
- Diary - counts - Incident.
- Diary - counts - Outfit.
- Diary - counts - Pet.
- Diary - counts - Quest.
- Diary - counts - Room.
- Diary - counts - Weapon.
- Dweller - has - Decease.
- Dweller - goes - Dweller_quest.
- Dweller - goes - Exploration.
- Dweller - has - Kinship.
- Dweller - has - Outfit.
- Dweller - has - Pet.
- Dweller - belongs - Room.
- Dweller - has - Special.
- Dweller - has - Weapon.
- Exploration - gets - Storage_exploration.
- Incident - has - Incident_type.
- Incident - has - Room.
- Junk - belongs - Storage.
- Kinship - Involves - Dweller.
- Outfit - has - Special.
- Outfit - belongs - Storage.
- Pet - belongs - Storage.
- Quest - has - Dweller_quest.
- Quest - gets - Storage_quest.
- Room - has - Room_type.
- Sale - has - Storage.
- Scrap - has - Storage.

- Scrap - make - Storage_scrap.
- Storage - belongs - Dweller_quest.
- Storage - belongs - Storage_exploration.
- Storage - belongs - Storage_scrap.
- Storage - belongs - Storage_quest.
- Weapon - belongs - Storage.

3. STATEMENTS

Functions

Name	fn_calculate_junk
Description	Calculates how much junk is available in storage for sale and scrap
Parameters	
Name	Description
ct_junk	Represents id_junk
accept_zero	If True, will allow to return 0 in case there is no junk available instead to return a Error Message
Returns	
Value	Description
0	When ct_junk is NULL
Error Message	When junk doesn't exist or when there isn't junk available for usage (in case accept_zero is False)
Junk quantity	The number of junk available at storage

Name	fn_calculate_outfit
Description	Calculates how much outfit is available in storage for use, sale, and scrap
Parameters	
Name	Description
ct_outfit	Represents id_outfit
accept_zero	If True, will allow to return 0 in case there is no outfit available instead to return a Error Message
Returns	
Value	Description
0	When ct_outfit is NULL
Error Message	When outfit doesn't exist or when there isn't outfit available for usage (in case accept_zero is False)
Outfit quantity	The number of outfit available at storage

Name	fn_calculate_pet
Description	Calculates how much pet is available in storage for use, sale, and scrap
Parameters	
Name	Description
ct_pet	Represents id_pet
accept_zero	If True, will allow to return 0 in case there is no outfit available instead to return a Error Message
Returns	
Value	Description
0	When ct_pet is NULL
Error Message	When pet doesn't exist or when there isn't pet available for usage (in case accept_zero is False)
Pet quantity	The number of pet available at storage

Name	fn_calculate_weapon
Description	Calculates how much weapon is available in storage for use, sale, and scrap
Parameters	
Name	Description
ct_weapon	Represents id_weapon
accept_zero	If True, will allow to return 0 in case there is no outfit available instead to return a Error Message
Returns	
Value	Description
0	When ct_weapon is NULL
Error Message	When weapon doesn't exist or when there isn't weapon available for usage (in case accept_zero is False)
Weapon quantity	The number of weapon available at storage

Name	fn_count_floor_size
Description	Counts the sizes of the floor, to make sure there isn't too much room in it
Parameters	
Name	Description
in_floor	Represents id_room
Returns	
Value	Description
Floor sizes	Returns how much space was taken up on the floor
Floor sizes + 1	Returns how much space was taken up on the floor (if it's on the first floor)

Name	fn_insert_kinship
Description	Inserts a new kinship or returns the id_kinship for the existing kinship
Parameters	
Name	Description
in_id_mom	Represents the id_dweller from the mother
in_id_dad	Represents the id_dweller from the father
Returns	
Value	Description
id_kinship	If the kinship already exists, returns the id_kinship
Last Insert Id	Inserts the new kinship and then returns its id
Error Message	In case there is any mistakes at the parameters

Name	fn_string_filter
Description	Filters a string, removing invalid characters
Parameters	
Name	Description
in_string	Represents the string to be filtered
Returns	
Value	Description
TRIM(in_string)	The filtered string

Name	fn_string_length
Description	Filters a string, calculating its real length
Parameters	
Name	Description
in_string	Represents the string to be filtered
Returns	
Value	Description
String Length	The string sizes, including the filter from fn_string_filter()

Name	fn_verify_special
Description	Verify if SPECIAL is valid
Parameters	
Name	Description
in_str	Represents <i>strength</i> and verify if it is between 0 and 10
in_per	Represents <i>perception</i> and verify if it is between 0 and 10
in_end	Represents <i>endurance</i> and verify if it is between 0 and 10
in_cha	Represents <i>charisma</i> and verify if it is between 0 and 10
in_int	Represents <i>intelligence</i> and verify if it is between 0 and 10
in_agi	Represents <i>agility</i> and verify if it is between 0 and 10
in_luc	Represents <i>luck</i> and verify if it is between 0 and 10
in_category	Verify if SPECIAL's category is 'dweller' or 'outfit'
Returns	
Value	Description
Error Message	When any parameter is invalid
TRUE	When all parameters are valid

Name	fn_verify_storage
Description	Calculates how much product (it can be outfit, weapon, pet, and junk) has at storage
Parameters	
Name	Description
in_product	The name of the product
return_type	The type of value to return
accept_zero	If True, will allow to return 0 in case there is no outfit available instead to return a Error Message
Returns	
Value	Description
Error Message	When product doesn't exist or when return_type not between 1-3
id_storage	The id_storage of the in_product
Product category	Describes if the category is outfit, weapon, pet, or junk
Storage quantity	Product quantity at storage

Stored Procedures (for strong entities)

Name	sp_add_product
Description	Adds more products to storage based on product id_storage
Parameters	
Name	Description
in_product_category	Defines the product's category
in_id_storage	Represents the product id_storage
in_qnt_storage	Actual product's quantity at storage
in_quantity	The quantity of product to add at storage

Name	sp_insert_baby
Description	Inserts a new dweller as a baby
Parameters	
Name	Description
in_name	Defines the name of the baby
in_gender	Defines the baby's gender
in_id_mom	Defines the id_dweller from the mother
in_id_dad	Defines the id_dweller from the father
in_str	Defines <i>strength</i>
in_per	Defines <i>perception</i>
in_end	Defines <i>endurance</i>
in_cha	Defines <i>charisma</i>
in_int	Defines <i>intelligence</i>
in_agi	Defines <i>agility</i>
in_luc	Defines <i>luck</i>

Name	sp_insert_decease
Description	Inserts a new decease (used with sp_insert_dweller_decease)
Parameters	
Name	Description
in_decease_type	Defines the local where the decease happened
in_incident_type	Defines the type of incident

Name	sp_insert_diary
Description	Records the number of items, resources, events, and dwellers at the vault
Parameters	
Name	Description
in_money	Defines the quantity of money at the moment
in_energy	Defines the percents of energy at the moment
in_water	Defines the percents of water at the moment
in_food	Defines the percents of pet at the moment

Name	sp_insert_dweller
Description	Inserts a new dweller to the vault
Parameters	
Name	Description
in_name	Defines the name of the dweller
in_gender	Defines the dweller's gander
in_level	Defines the dweller's level
in_room	Defines the id_room where dweller is
in_str	Defines <i>strength</i>
in_per	Defines <i>perception</i>
in_end	Defines <i>endurance</i>
in_cha	Defines <i>charisma</i>
in_int	Defines <i>intelligence</i>
in_agi	Defines <i>agility</i>
in_luc	Defines <i>luck</i>

Name	sp_insert_dweller_decease
Description	Inserts the last id_decease into the deceased dweller
Parameters	
Name	Description
in_dweller	Represents the dweller's id/name
in_decease_type	Defines the local where the decease happened
in_incident_type	Defines the type of incident
in_save_product	If True, will save dweller's items at storage, otherwise will exclude them

Name	sp_insert_exploration
Description	Inserts a new exploration
Parameters	
Name	Description
in_dweller	Represents the dweller id
in_duration	Represents the time (in minutes) the dweller was exploring
in_date	Represents the date dweller came back from exploration

Name	sp_insert_incident
Description	Inserts a new incident
Parameters	
Name	Description
in_date	Define the date when the incident happens, not accepting future dates
in_incident_type	Define the type of the incident
in_room	Defines in which room this happened

Name	sp_insert_junk
Description	Inserts a new junk at the storage
Parameters	
Name	Description
in_name	Defines the name of the junk
in_qnt	Define how many junks is being inserted

Name	sp_insert_outfit
Description	Inserts a new outfit at the storage
Parameters	
Name	Description
in_name	Defines the name of the outfit
in_qnt	Define how many outfits is being inserted
in_str	Defines <i>strength</i>
in_per	Defines <i>perception</i>
in_end	Defines <i>endurance</i>
in_cha	Defines <i>charisma</i>
in_int	Defines <i>intelligence</i>
in_agi	Defines <i>agility</i>
in_luc	Defines <i>luck</i>

Name	sp_insert_pet
Description	Inserts a new pet at the storage
Parameters	
Name	Description
in_name	Defines the name of the pet
in_description	Describes the pet's function
in_qnt	Define how many pets are being inserted

Name	sp_insert_quest
Description	Inserts a new quest at tb_quest and the dwellers who went at tb_dweller_quest
Parameters	
Name	Description
in_description	Represents a small description about the quest
in_date_begin	Defines the date that the quest began
in_dweller_1	Represents the first dweller's id who went to the quest
in_dweller_2	Represents the second dweller's id who went to the quest
in_dweller_3	Represents the third dweller's id who went to the quest

Name	sp_insert_room
Description	Inserts a new room
Parameters	
Name	Description
in_size	Define the size of the room, from 1 to 3
in_floor	Define the floor in which the room is, from 1 to 25
in_room_type	Define the room type and what job is in it

Name	sp_insert_sale
Description	Inserts a new sale
Parameters	
Name	Description
in_product	Represents the product which was sold
in_quantity	Defines the quantity of product sold
in_profit	Defines the profit obtained from the sale
in_date	Defines the sale's date

Name	sp_insert_scrap
Description	Inserts a new scrap
Parameters	
Name	Description
in_product	Represents the product which was scraped
in_quantity	Defines the quantity of product scraped
in_date	Defines the scrap's date

Name	sp_insert_special
Description	Inserts a new special
Parameters	
Name	Description
in_str	Defines <i>strength</i> to 'dweller' or 'outfit'
in_per	Defines <i>perception</i> to 'dweller' or 'outfit'
in_end	Defines <i>endurance</i> to 'dweller' or 'outfit'
in_cha	Defines <i>charisma</i> to 'dweller' or 'outfit'
in_int	Defines <i>intelligence</i> to 'dweller' or 'outfit'
in_agi	Defines <i>agility</i> to 'dweller' or 'outfit'
in_luc	Defines <i>luck</i> to 'dweller' or 'outfit'
in_category	Define for which category the SPECIAL belongs

Name	sp_insert_storage
Description	Inserts a new product at the storage
Parameters	
Name	Description
in_category	Define which is the product category, accepting 'outfit', 'weapon', 'pet' and 'junk'

Name	sp_insert_weapon
Description	Inserts a new weapon at the storage
Parameters	
Name	Description
in_name	Defines the name of the weapon
in_damage	Describes the weapon's damage
in_qnt	Define how many weapons are being inserted

Name	sp_subtract_product
Description	Subtracts some products from storage based on product id_storage
Parameters	
Name	Description
in_product_category	Defines the product's category
in_id_storage	Represents the product id_storage
in_qnt_storage	Actual product's quantity at storage
in_quantity	The quantity of product to subtract at storage

Name	sp_update_dweller_bonus
Description	Inserts new items in dweller
Parameters	
Name	Description
up_dweller	Represents the dweller's id/name
up_outfit	Represents the outfit id
up_weapon	Represents the weapon id
up_pet	Represents the pet id

Name	sp_update_dweller_special
Description	Updates dwellers's SPECIAL
Parameters	
Name	Description
up_dweller	Represents the dweller's id/name
up_str	Defines <i>strength</i>
up_per	Defines <i>perception</i>
up_end	Defines <i>endurance</i>
up_cha	Defines <i>charisma</i>
up_int	Defines <i>intelligence</i>
up_agi	Defines <i>agility</i>
up_luc	Defines <i>luck</i>

Name	sp_dweller_status
Description	Updates dweller's status
Parameters	
Name	Description
up_dweller	Represents the dweller's id/name
up_level	Defines the dweller's level
up_room	Defines the id_room where dweller is

Name	sp_update_room
Description	Updates room's size
Parameters	
Name	Description
up_room	Represents room's id
up_size	Defines room's new size

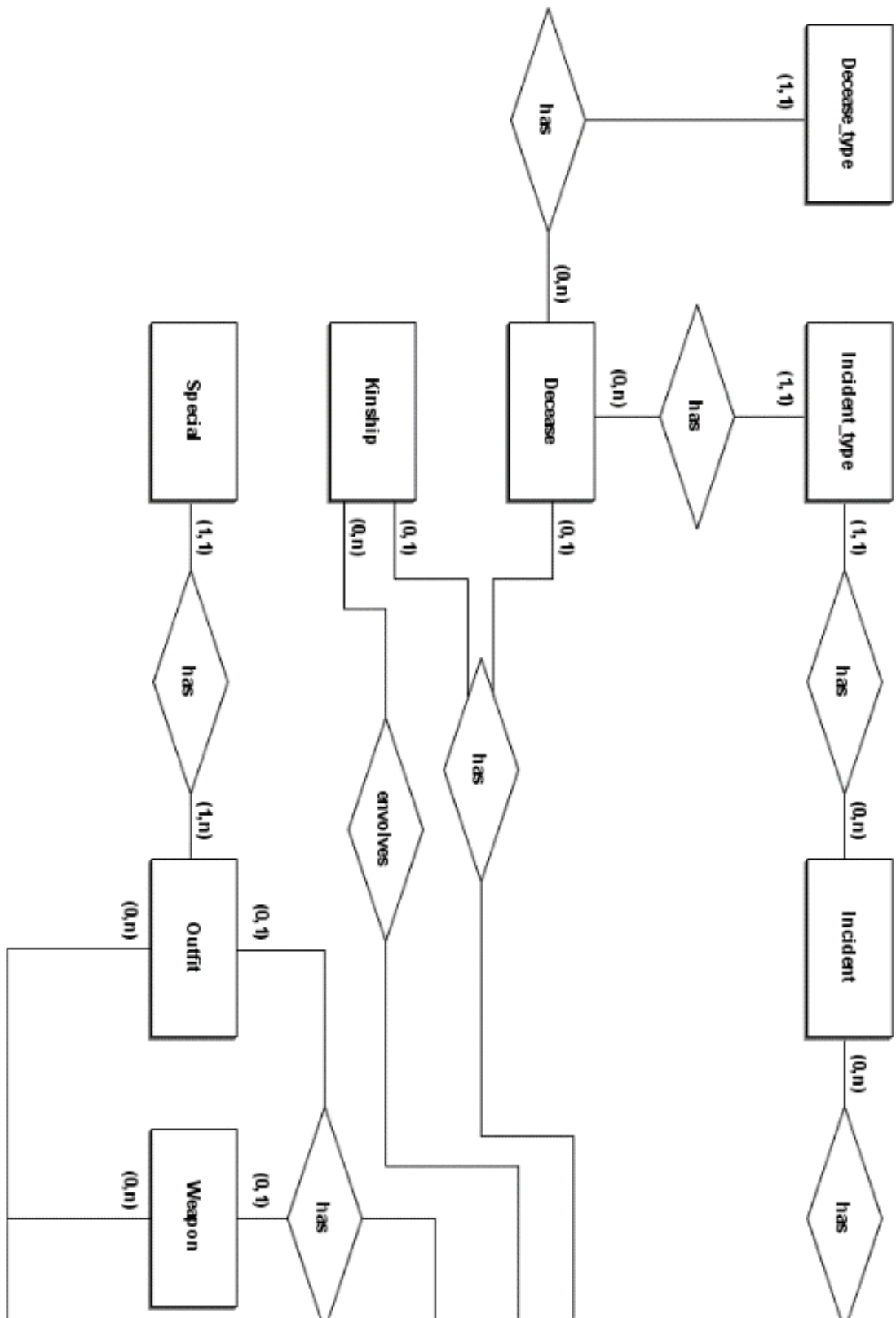
Stored Procedures (for associative entities)

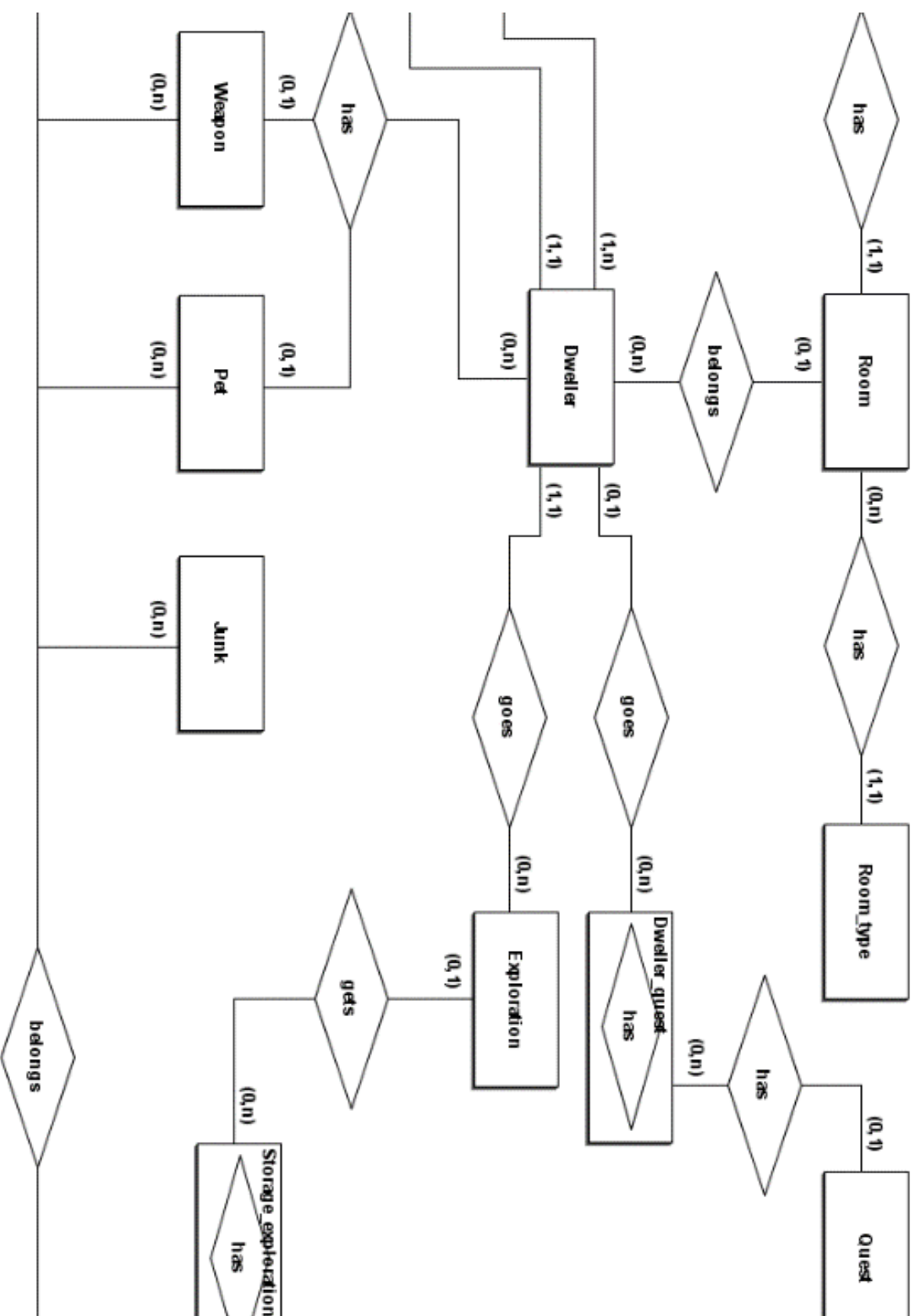
Name	sp_insert_storage_exploration
Description	Insert earnings in exploration
Parameters	
Name	Description
in_exploration	References the exploration's id
in_product	References the product's id
in_quantity	Defines how many products earns

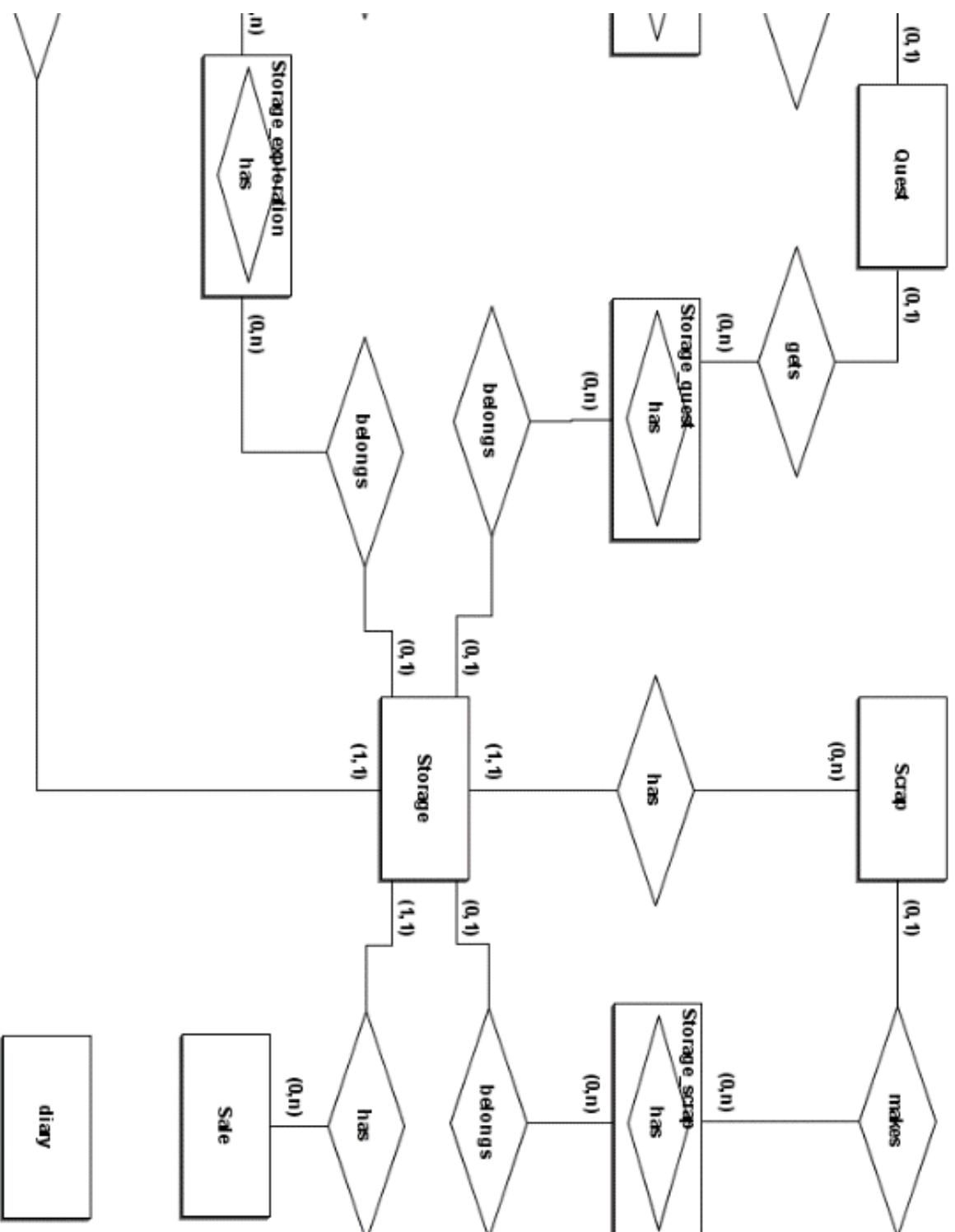
Name	sp_insert_storage_quest
Description	Insert earnings in quest
Parameters	
Name	Description
in_quest	References the quest's id
in_product	References the product's id
in_quantity	Defines how many products earns

Name	sp_insert_storage_scrap
Description	Insert earnings in scrap
Parameters	
Name	Description
in_scrap	References the scrap's id
in_product	References the product's id
in_quantity	Defines how many products earns

4. CONCEPTUAL SCHEMA DESING







5. LOGICAL SCHEMA DESIGN

