

Winter
2024

Course Project Technical Report

Version 0.4
Winter 2024

COMP 2714-BCIT
COURSE PROJECT TECHNICAL REPORT

Team (Project) Name:

Team Members:

	Student Name	Student-ID
1	Will Otterbein	A01372608
2	Raymond Xie	A01343016
3	Calvin Lee	A00922317
4		

ABOUT THE PROJECT	3
PROJECT GOAL	3
PROJECT EVALUATION	3
PROJECT TIMELINE	3
MILESTONE1	4
MILESTONE1-TASK1: UNIVERSE OF DISCOURSE (MINI-WORLD) DESCRIPTION	4
MILESTONE1-TASK2: CONCEPTUAL DESIGN USING ER/EER DIAGRAM	5
MILESTONE1-TASK3: CONCEPTUAL DESIGN USING UML NOTATION [OPTIONAL BUT RECOMMENDED]	ERROR!
BOOKMARK NOT DEFINED.	
MILESTONE1-TASK4: DEFINING FUNCTIONAL DEPENDENCIES IN YOUR MINI-WORLD	5
MILESTONE2	10
MILESTONE2-TASK1: UPDATED UNIVERSE OF DISCOURSE (MINI-WORLD) DESCRIPTION	10
MILESTONE2-TASK2: UPDATED CONCEPTUAL DESIGN	10
MILESTONE2-TASK3: UPDATING FUNCTIONAL DEPENDENCIES IN YOUR MINI-WORLD	10
MILESTONE2-TASK4: CREATING THE RELATIONAL MODEL OF YOUR CONCEPTUAL MODEL	10
MILESTONE3	13

MILESTONE3-TASK1: UPDATE YOUR CONCEPTUAL AND RELATIONAL MODELS	13
MILESTONE3-TASK1: CREATE YOUR DATABASE SCHEMA	14
MILESTONE3-TASK2: POPULATE YOUR TABLES WITH SOME SAMPLE DATA	19
MILESTONE3-TASK3: WRITE SQL STATEMENT	25
 MILESTONE4	 32
 MILESTONE4-TASK1: IDENTIFY FULL, PARTIAL AND TRANSITIVE FUNCTIONAL DEPENDENCIES IN YOUR DESIGN	 32
MILESTONE4-TASK2: HIGHEST NORMAL FORM	33
MILESTONE4-TASK3: CONVERTING TO 3NF	33
MILESTONE4-TASK4: CONVERTING TO BCNF	33

About the Project

Project Goal

The objective of the project activity is to provide this opportunity to create a database system from the beginning to the database development. In this project, you will:

- 1- Come up with a universe of discourse and corresponding conceptual model
- 2- Convert your conceptual model to a relational model
- 3- Use a SQL product (such as MySQL) to create your database, populate it with some data and write some DML statements
- 4- Normalize your database schema

The project could be done in groups of up to 4 people. Depending on the number of people in the group the scope of the project would change.

Project Evaluation

Each Milestone will be evaluated independently.

Once you join a group, it is expected to stay in the group until the end of the project.

The project final grade for each individual will be based on the following metrics:

- 1- The completion of the Milestones 1-4 on time and completeness of this technical report
- 2- Evaluation of teammates in a group

Project Timeline

Milestones	Due date
Milestone 1	Please refer to Learning Hub
Milestone 2	Please refer to Learning Hub
Milestone 3	Please refer to Learning Hub
Milestone 4	Please refer to Learning Hub

Milestone1

- You need to describe a mini world.
- You can come up with a new idea for the mini-world or describe an existing application.
- A good mini-world is the one with a conceptual model including all the topics we have studies in Module 1:
 - Entities, Weak entities, Total and partial participations, Classes and sub-classes, composite attributes, derived attributes, super/sub classes, ...
- The conceptual model in 2 formats (ER/EER) and UML Class Notation
 - UML Notation is Optional.

Note: The scope of the project is adjusted based on how many people work on a project in a group.

If you work individually (group of 1) on this project:

- Then it is expected between 8-10+ entity types in your conceptual model.

If you work in a group of 2 on the project:

- Then it is expected between 12-15+ entity types in your conceptual model.

If you work in a group of 3/4 on the project:

- Then it is expected between 18-20+ entity types in your conceptual model.

Milestone1-Task1: Universe of Discourse (Mini-World) Description

[Write your answer here]

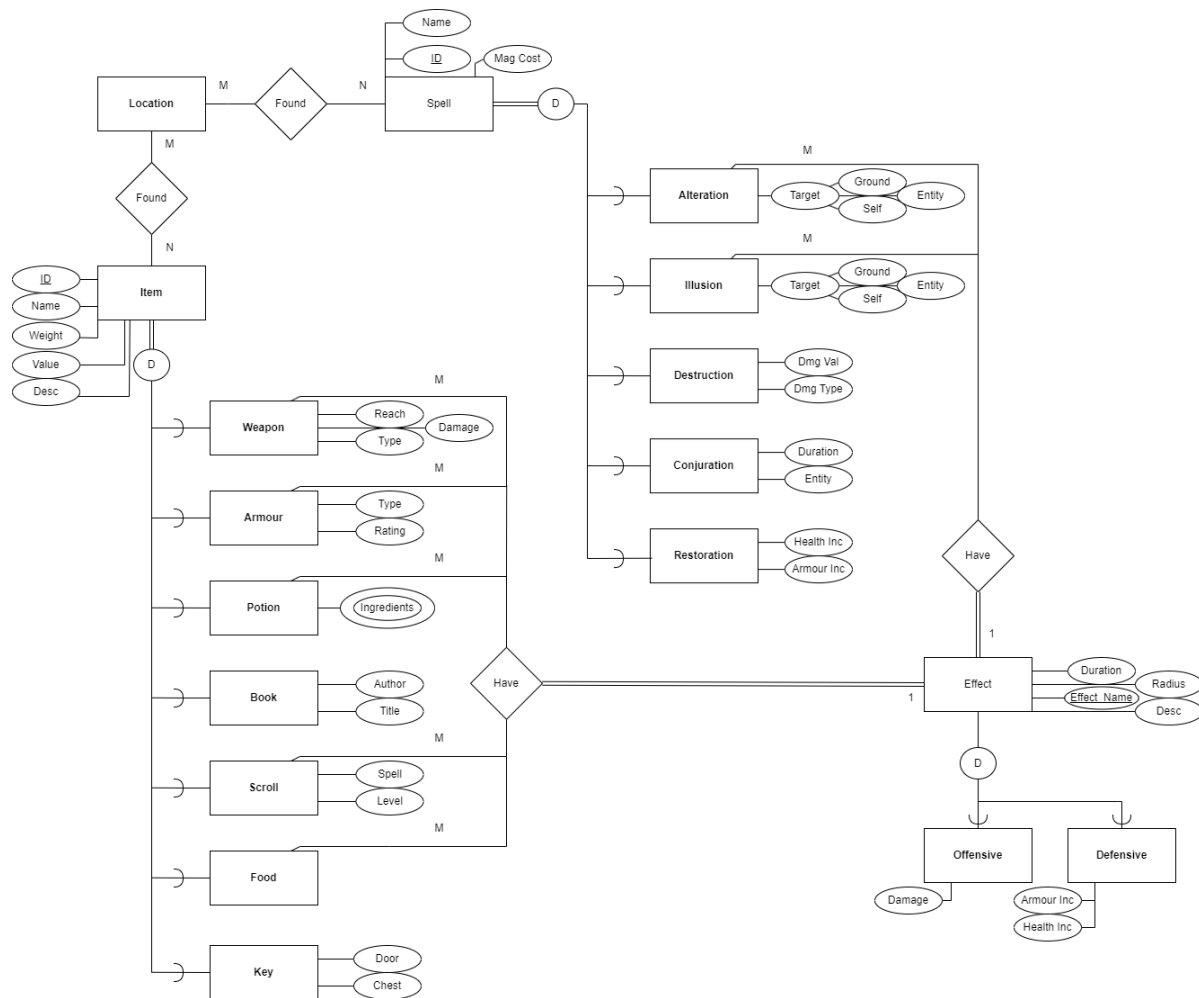
Here's a revised version of your paragraph:

The focus of our discourse is the inventory of a player character in Skyrim. This domain encompasses information about the diverse array of items and spells in the game, capturing unique details such as an item's specific weight and value, or a spell's magicka cost. Our domain will also delve into the unique effects applied to items and spells, such as bleed or blindness.

The objective of a database modeling this domain is to create a system that allows users to efficiently manage their multitude of items, spells, and their associated effects that accumulate during a game playthrough. With this system, users can effortlessly ascertain the total value of their weapons or armor, as well as the count of spells they possess that grant a specific effect. This ensures that complex decisions, such as determining which books in their possession will yield the most advantageous skills, are simplified from the user's perspective. In short, this database aims to enhance the gaming experience by providing a structured and easy-to-navigate source of information.

Milestone1-Task2: Conceptual Design using ER/EER Diagram

[Write your answer here]



Milestone1-Task3: Defining functional dependencies in your mini-world

If you work individually (group of 1) on this project:

- Then it is expected between 4+ functional dependencies in your mini-world.

If you work in a group of 2 on the project:

- Then it is expected between 6+ functional dependencies in your mini-world.

If you work in a group of 3/4 on the project:

- Then it is expected between 7+ functional dependencies in your mini-world.

List all functional dependencies here: Make sure to include visuals as discussed during the lecture:

1- Primary-Key Functional Dependencies: List them here:

- **Weapon**

- {ID} -> {Name, weight, Value, Desc, Reach, Type, Damage}

<u>ID</u>	Name	Weight	Value	Desc	Type	Reach	Damage
-----------	------	--------	-------	------	------	-------	--------

- **Armour**

- {ID} -> {Name, weight, Value, Desc, Type, Rating}

<u>ID</u>	Name	Weight	Value	Desc	Type	Rating
-----------	------	--------	-------	------	------	--------

- **Potion**

- {ID} -> {Name, weight, Value, Desc, Ingredients}

- **Book**

- {ID} -> {Name, weight, Value, Desc, Author, Title}

<u>ID</u>	Name	Weight	Value	Desc	Author	Title
-----------	------	--------	-------	------	--------	-------

- **Scroll**

- {ID} -> {Name, weight, Value, Desc, Spell, Level}

<u>ID</u>	Name	Weight	Value	Desc	Spell	Level
-----------	------	--------	-------	------	-------	-------

- **Food**

- {ID} -> {Name, weight, Value, Desc}

<u>ID</u>	Name	Weight	Value	Desc
-----------	------	--------	-------	------

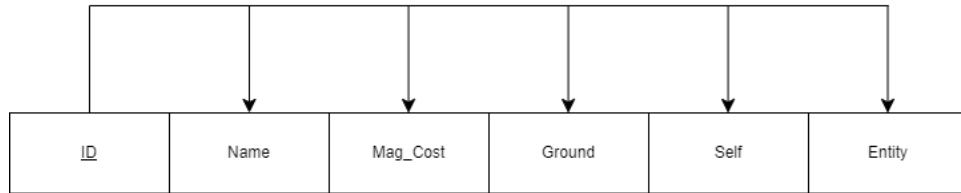
- **Key**

- {ID} -> {Name, weight, Value, Desc, Door, Chest}

<u>ID</u>	Name	Weight	Value	Desc	Door	Chest
-----------	------	--------	-------	------	------	-------

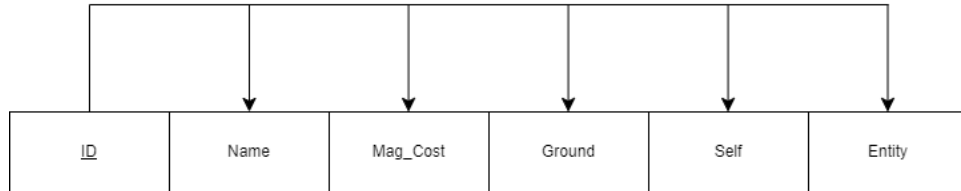
- **Alteration**

- {ID} -> {Name, Mag_Cost, Ground, Self, Entity}



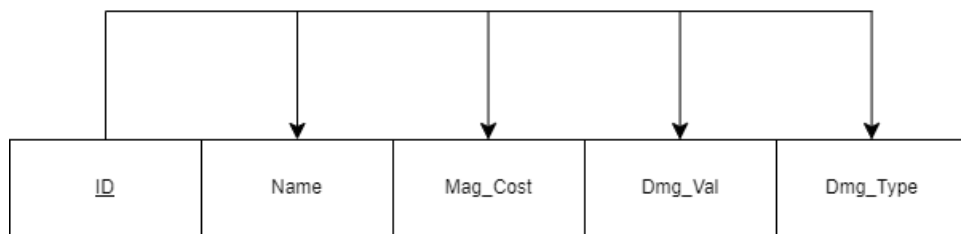
- **Illusion**

- {ID} -> {Name, Mag_Cost, Ground, Self, Entity}



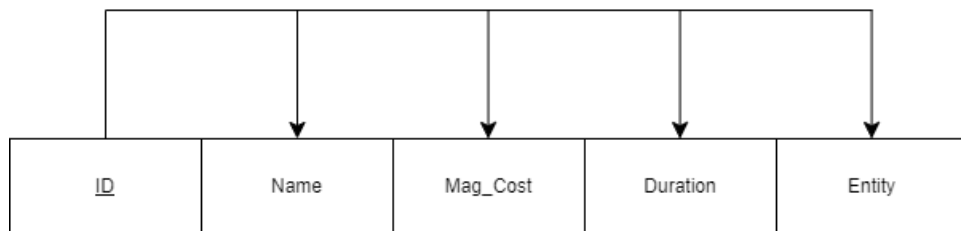
- **Destruction**

- {ID} -> {Name, Mag_Cost, Dmg_Val, Dmg_Type}



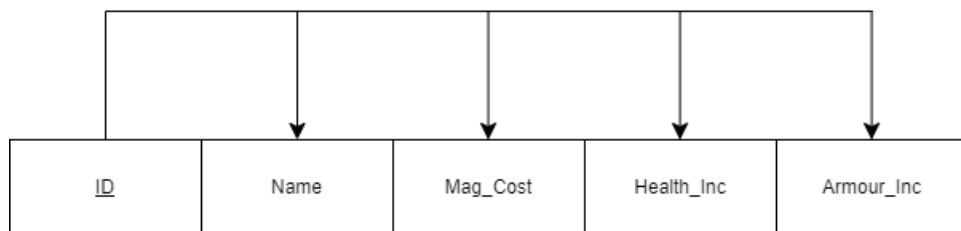
- **Conjuration**

- {ID} -> {Name, Mag_Cost, Duration, Entity}



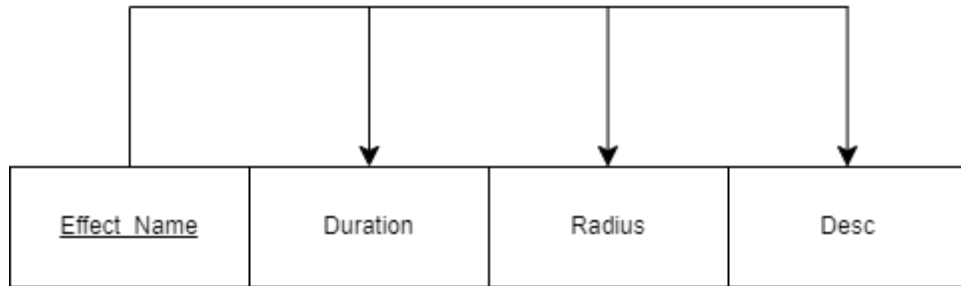
- **Restoration**

- {ID} -> {Name, Mag_Cost, Health_Inc, Armour_Inc}



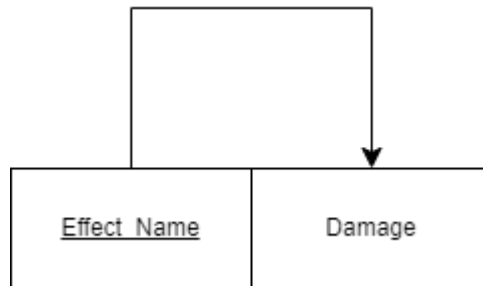
- **Effect**

- {Effect_Name} -> {Duration, Radius, Desc}



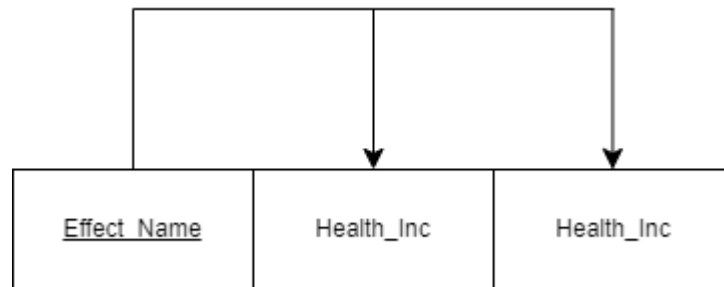
- **Offensive**

- {Effect Name} -> {Damage}



- **Defensive**

- {Effect Name} -> {Health_Inc, Armour_Inc}



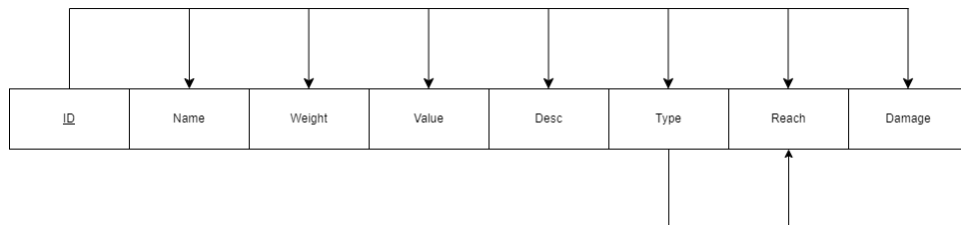
2- Partial Functional Dependencies: List them here

None, as there are no multi-attribute primary keys.

3- Transitive Functional Dependencies: List them here

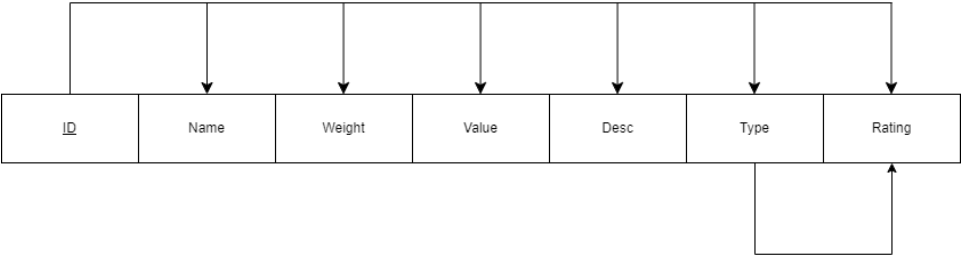
- **Weapon**

- {Type} -> {Reach}



- **Armour**

- {Type} -> {Rating}



Milestone2

In Milestone 2, you are going to continue working on what you have done in Milestone1.

Milestone2-Task1: Updated Universe of Discourse (Mini-World) Description

- [Optional]: If your mini-world needs to be updated for any reasons, you have the chance to update it here. Please notice, if the original version of the Mini-World is not comprehensive enough (having different types of entities and relationships and ... and enough number of entities) you need to update the original mini-world and submit it in this Milestone.
- The Milestone2-Task1 is not going to be graded, yet an update is a must if needed based on the comment above. Please also see the Task3 below.

As a group, we unanimously decided that we are content with our current miniworld.

[Write your answer here] N/A

Milestone2-Task2: Updated Conceptual Design

- If an update to the Conceptual design is needed, please do it and add it here. (Particularly If you have to update your original mini-world in Task1 above)

[Write your answer here] N/A

Milestone2-Task3: Updating functional dependencies in your mini-world

If you work individually (group of 1) on this project:

- Then it is expected between 4+ functional dependencies in your mini-world.

If you work in a group of 2 on the project:

- Then it is expected between 6+ functional dependencies in your mini-world.

If you work in a group of 3/4 on the project:

- Then it is expected between 7+ functional dependencies in your mini-world.

Note: If you cannot extract enough number of functional dependencies in your mini-world, you will need to update it and submit it under Milestone2-Task1.

[Write your answer here] N/A

Milestone2-Task4: Creating the relational model of your conceptual model

- You need to create the relational models following the steps we defined in the lecture.

Item [ID, locName, name, weight, value, desc] X
Item.locName references Location.locName

Location [locName] X

Item subclasses:

Weapon [itemID, damageValue, reach, type, effectID] X
Weapon.itemID references Item.ID
Weapon.effectID references Effect.effectID

Armour [itemID, type, rating, effectID] X
Armour.itemID references Item.ID
Armour.effectID references Effect.effectID

Ingredients [pltemID, itemD] X
Ingredients.item references Item.itemID
Ingredients.pltemID references Potion.itemID

Potion [itemID, effectID] X
Potion.itemID references Item.ID
Potion.effectID references Effect.effectID

Book [itemID, author, title] X
Book.itemID references Item.ID

Scroll [itemID, spell, level, effectID] X
Scroll.itemID references Item.ID
Scroll.spell references Spell.ID
Scroll.effectID references Effect.effectID

Food [itemID, effectID] X
Food.itemID references Item.ID

Key [itemID, door, chest] X

Key.itemID references Item.ID

~~~~~

Spell [ID, locName, name, cost]  
Item.locName references Location.locName

Alteration [spellID, target, effectID]  
Alteration.spellID references Spell.ID  
Alteration.effectID references Effect.effectID

Illusion [spellID, target, effectID]

Illusion.spellID references Spell.ID

Illusion.effectID references Effect.effectID

Destruction [spellID, damage, type]

Destruction.spellID references Spell.ID

Conjuration [spellID, duration, entity]

Conjuration.spellID references Spell.ID

Restoration [spellID, healthRestored, armourIncrease]

Restoration.spellID references Spell.ID

~~~~~

Effect [effectName, duration, radius, desc]

Offensive [effectName, damage]

Offensive.effectName references Effect.effectName

Defensive [effectName, healthIncrease, armourIncrease]

Defensive.effectName references Effect.effectName

Milestone3

In Milestone 3, In Milestone 3 you are doing to work with the database that you have designed.

Note: By the end of Milestone 2, it is expected that you have designed your relational model (database).

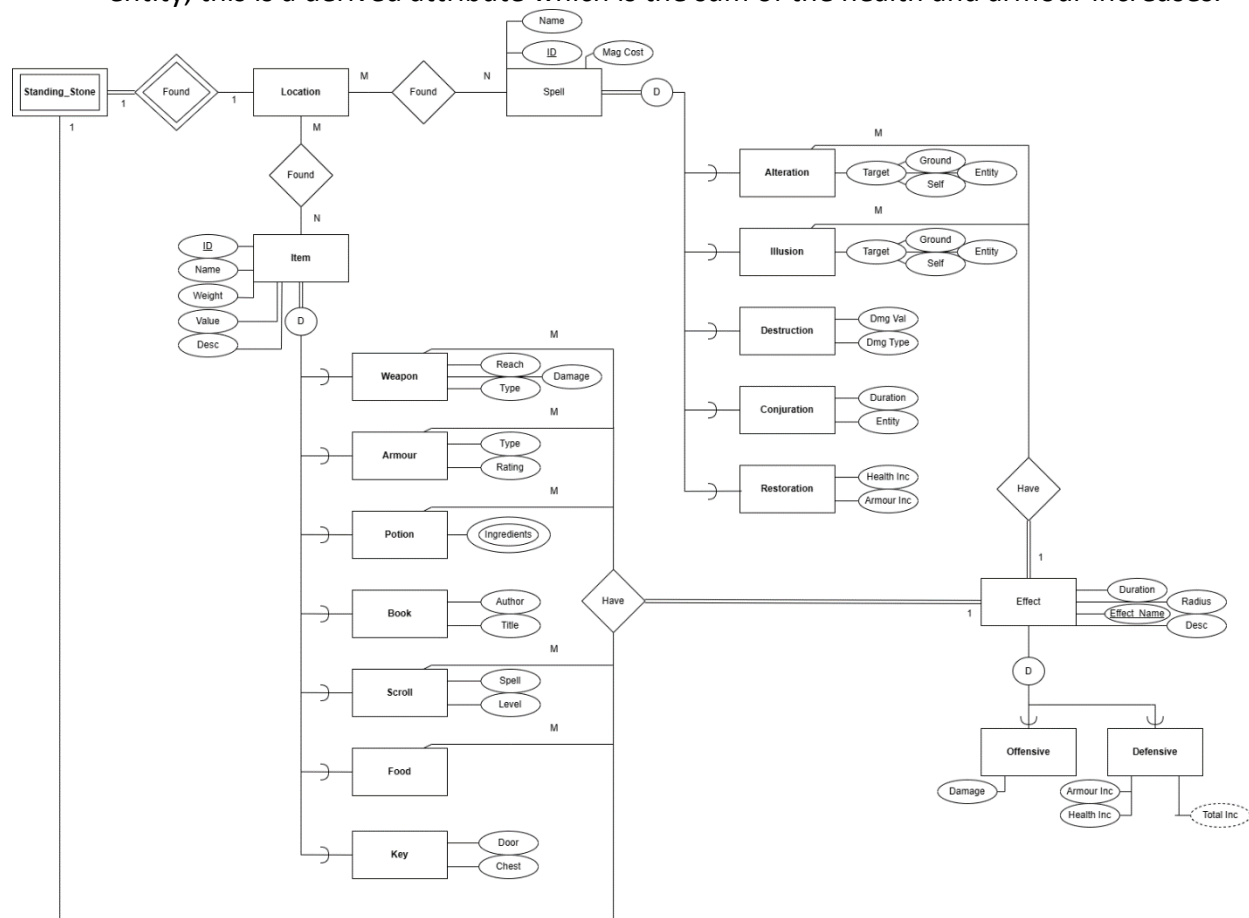
Milestone3-Task1: Update your conceptual and relational models

If you have received any feedback from the evaluator of Milestone 1 and 2, that you need to make some changes in your conceptual and relation models, you need to do it now and insert the updated answer:

[Write your answer here]

Updated Diagram

- We added a weak_entity known as Standing_stone. These stones are found throughout the Skyrim game world and only exist at specific locations. These standing stones are known to grant the player specific effects, so we though they would be an interesting addition to the database.
- Additionally, we added a Total Inc (total increase) value to the defensive effect subclass entity, this is a derived attribute which is the sum of the health and armour increases.



Updated Relational Model

- Here are the additional / changed relations. Otherwise, the model remains similar.

Standing_Stone[stoneName, stoneLocation, effectID]

Standing_Stone.stoneLocation references Location.locName

Standing_Stone.effectID references Effect.effectID

Defensive [effectName, healthIncrease, armourIncrease]

Defensive.effectName references Effect.effectName

(derived attributes do not appear in the relation, but the Total_Inc field will be accessible)

Milestone3-Task1: Create your database schema

- Please use the SQL DDL statements to create your database schema.
- Provide a name for each table.
- Include all attributes
- Define PK and FK.
- Please add a snapshot of the statement you have written and the results

[Write your answer here]

```
-- =====
-- DDL =====
-- =====

-- =====
-- LOCATION TABLE
--
CREATE TABLE Location (
    locName varchar(100),
    PRIMARY KEY (locName)
);

-- =====
-- EFFECT TABLE
--
CREATE TABLE Effect (
    effectID int,
    duration decimal,
    radius int,
    edesc varchar(100),
    PRIMARY KEY (effectID)
);
```

```

-- OFFENSIVE TABLE
--
CREATE TABLE Offensive(
    effectID int,
    damage decimal,
    PRIMARY KEY(effectID),
    FOREIGN KEY (effectID) REFERENCES Effect (effectID)
);

-- DEFENSIVE TABLE
--
CREATE TABLE Defensive(
    effectID int,
    healthIncrease decimal,
    armourIncrease decimal,
    PRIMARY KEY (effectID),
    FOREIGN KEY (effectID) REFERENCES Effect (effectID)
);

CREATE TABLE Standing_Stone (
    stoneName varchar(100),
    stoneLocation varchar(100),
    effectID int,
    PRIMARY KEY (stoneName, stoneLocation),
    FOREIGN KEY (stoneLocation) REFERENCES Location (locName),
    FOREIGN KEY (effectID) REFERENCES Effect (effectID)
);

-- XX

-- =====
-- SPELL TABLE AND SUBTABLES
--

-- SPELL TABLE
--
CREATE TABLE Spell (
    spellID int,
    locName varchar(100),
    spellName varchar(100),
    spellCost int,
    PRIMARY KEY (spellID),
    FOREIGN KEY (locName) REFERENCES Location(locName)
);

```


-- ALTERATION TABLE

--

```
CREATE TABLE Alteration (  
    spellID int,  
    effectID int,  
    sTarget varchar(100),  
    PRIMARY KEY (spellID),  
    FOREIGN KEY (spellID) REFERENCES Spell(spellID),  
    FOREIGN KEY (effectID) REFERENCES Effect(effectID)  
);
```

-- ILLUSION TABLE

--

```
CREATE TABLE Illusion (  
    spellID int,  
    effectID int,  
    starget varchar(100),  
    PRIMARY KEY (spellID),  
    FOREIGN KEY (spellID) REFERENCES Spell (spellID),  
    FOREIGN KEY (effectID) REFERENCES Effect (effectID)  
);
```

-- DESTRUCTION TABLE

--

```
CREATE TABLE Destruction (  
    spellID int,  
    damage decimal,  
    stype varchar(100),  
    PRIMARY KEY (spellID),  
    FOREIGN KEY (spellID) REFERENCES Spell (spellID)  
);
```

-- CONJURATION TABLE

--

```
CREATE TABLE Conjunction (  
    spellID int,  
    duration decimal,  
    entity varchar(100),  
    PRIMARY KEY (spellID),  
    FOREIGN KEY (spellID) REFERENCES Spell (spellID)  
);
```

-- RESTORATION TABLE

--

```
CREATE TABLE Restoration(  

```

```

    spellID int,
    healthRestored decimal,
    armourIncrease decimal,
    PRIMARY KEY (spellID),
    FOREIGN KEY (spellID) REFERENCES Spell (spellID)
);

-- =====
-- ITEM TABLE AND SUBTABLES
--
CREATE TABLE Item (
    itemID int,
    locName varchar(100),
    iName varchar(100),
    iWeight float,
    iValue int,
    iDesc varchar(100),
    PRIMARY KEY (itemID),
    FOREIGN KEY (locName) REFERENCES Location (locName)
);

-- WEAPON TABLE
--
CREATE TABLE Weapon (
    itemID int,
    damageValue int,
    reach float,
    wType varchar(100),
    effectID int,
    PRIMARY KEY (itemID),
    FOREIGN KEY (itemID) REFERENCES Item (itemID),
    FOREIGN KEY (effectID) REFERENCES Effect (effectID)
);

-- ARMOUR TABLE
--
CREATE TABLE Armour (
    itemID int,
    aType varchar(100),
    aRating int,
    effectID int,
    PRIMARY KEY (itemID),
    FOREIGN KEY (itemID) REFERENCES Item (itemID),
    FOREIGN KEY (effectID) REFERENCES Effect (effectID)
);

```

-- *POTION TABLE*

--

```
CREATE TABLE Potion (  
    itemID int,  
    effectID int,  
    PRIMARY KEY (itemID),  
    FOREIGN KEY (itemID) REFERENCES Item (itemID),  
    FOREIGN KEY (effectID) REFERENCES Effect (effectID)  
);
```

-- *INGREDIENT TABLE*

--

```
CREATE TABLE Ingredients (  
    itemID int,  
    pItemID int,  
    PRIMARY KEY (itemID, pItemID),  
    FOREIGN KEY (itemID) REFERENCES Item (itemID),  
    FOREIGN KEY (pItemID) REFERENCES Potion (itemID)  
);
```

-- *BOOK TABLE*

--

```
CREATE TABLE Book (  
    itemID int,  
    author varchar(100),  
    title varchar(100),  
    PRIMARY KEY (itemID),  
    FOREIGN KEY (itemID) REFERENCES Item (itemID)  
);
```

-- *SCROLL TABLE*

--

```
CREATE TABLE Scroll (  
    itemID int,  
    effectID int,  
    spellID int,  
    scLevel int,  
    PRIMARY KEY (itemID),  
    FOREIGN KEY (itemID) REFERENCES Item (itemID),  
    FOREIGN KEY (effectID) REFERENCES Effect (effectID),  
    FOREIGN KEY (spellID) REFERENCES Spell (spellID)  
);
```

-- *FOOD TABLE*

```
--
CREATE TABLE Food (
    itemID int,
    effectID int,
    PRIMARY KEY (itemID),
    FOREIGN KEY (itemID) REFERENCES Item (itemID),
    FOREIGN KEY (effectID) REFERENCES Effect (effectID)
);

-- KEY TABLE
--
CREATE TABLE KeyItem (
    itemID int,
    kDoor varchar(100),
    kChest varchar(100),
    PRIMARY KEY (itemID),
    FOREIGN KEY (itemID) REFERENCES Item (itemID)
);
```

```
mysql> show tables;
+-----+
| Tables_in_comp2714_termassignment |
+-----+
| alteration                          |
| armour                             |
| book                               |
| conjuration                        |
| defensive                          |
| destruction                        |
| effect                             |
| food                               |
| illusion                           |
| ingredients                        |
| item                               |
| keyitem                           |
| location                           |
| offensive                          |
| potion                             |
| restoration                        |
| scroll                             |
| spell                              |
| standing_stone                     |
| weapon                             |
+-----+
20 rows in set (0.00 sec)
```

Milestone3-Task2: Populate your tables with some sample data

- Insert some sample data to your tables
- The volume of inserted data is not important. Just insert enough amount of data that makes sense for your project.
- Please add a snapshot of the tables with some data in them

[Write your answer here]

```

-- =====
-- DML =====
-- =====

--
-- LOCATION NEW
--

-- Inserting data into the Location table
INSERT INTO Location (locName) VALUES
    ('Whiterun'),
    ('Solitude'),
    ('Riften');

--
-- EFFECT NEW
--

-- Inserting data into the Effect table
INSERT INTO Effect (effectID, duration, radius, edesc) VALUES
    (0, 5.0, 30, 'Invisibility'),
    (1, 10.5, 20, 'Fire Damage'),
    (2, 15.0, 30, 'Frost Damage'),
    (3, 60.0, 0, 'Healing'),
    (4, 120.0, 0, 'Armor Buff'),
    (5, 30.0, 10, 'Calm'),
    (6, 30.0, 10, 'Fear'),
    (7, 0, 0, 'Sword Damage'),
    (8, 0, 0, 'Armor Protection'),
    (9, 60.0, 0, 'Health Restoration'),
    (10, 60.0, 0, 'Mana Restoration'),
    (11, 0, 0, 'Ingredient Effect'),
    (12, 0, 0, 'Book Knowledge'),
    (13, 0, 0, 'Scroll Spell'),
    (14, 0, 0, 'Food Nutrition'),
    (15, 0, 0, 'Key Access');

-- Inserting data into the Offensive table
INSERT INTO Offensive (effectID, damage)
VALUES (1, 50), -- Fire Damage
       (2, 75), -- Frost Damage
       (7, 12); -- Sword Damage

-- Inserting data into the Defensive table

```

```

INSERT INTO Defensive (effectID, healthIncrease, armourIncrease)
VALUES (3, 100, 0), -- Healing
      (4, 0, 50), -- Armor Buff
      (8, NULL, 25); -- Armor Protection (NULL for health increase)

-- Inserting data into the Standing_Stone table
INSERT INTO Standing_Stone (stoneName, stoneLocation, effectID)
VALUES ('Shadow Stone', 'Riften', 0);

--
-- SPELLS NEW
--
-- Inserting data into the Spell table
INSERT INTO Spell (spellID, locName, spellName, spellCost) VALUES
(1, 'Whiterun', 'Flames', 100),
(2, 'Solitude', 'Frostbite', 200),
(3, 'Whiterun', 'Healing', 150),
(4, 'Solitude', 'Oakflesh', 50),
(5, 'Whiterun', 'Summon Familiar', 100),
(6, 'Solitude', 'Summon Atronach', 200),
(7, 'Whiterun', 'Calm', 75),
(8, 'Solitude', 'Fear', 75);

-- Inserting data into the Alteration table
INSERT INTO Alteration (spellID, effectID, sTarget) VALUES (3, 3, 'Draugr'), (4,
4, 'Dragon');

-- Inserting data into the Illusion table
INSERT INTO Illusion (spellID, effectID, starget) VALUES (7, 5, 'Bandit'), (8, 6,
'Giant');

-- Inserting data into the Destruction table
INSERT INTO Destruction (spellID, damage, stype) VALUES (1, 50.0, 'Fire'), (2,
75.0, 'Frost');

-- Inserting data into the Conjuraton table
INSERT INTO Conjuraton (spellID, duration, entity) VALUES (5, 10.5, 'Familiar'),
(6, 15.0, 'Atronach');

-- Inserting data into the Restoration table
INSERT INTO Restoration (spellID, healthRestored, armourIncrease) VALUES (3,
100.0, 50.0), (4, 150.0, 75.0);

--
-- ITEM NEW

```

--

-- *Inserting data into the Item table*

```
INSERT INTO Item (itemID, locName, iName, iWeight, iValue, iDesc) VALUES
(1, 'Whiterun', 'Iron Sword', 10.0, 25, 'A standard Iron Sword'),
(2, 'Solitude', 'Steel Sword', 12.0, 45, 'A sturdy Steel Sword'),
(3, 'Whiterun', 'Apple', 0.1, 2, 'A fresh apple'),
(4, 'Solitude', 'Cheese', 1.0, 5, 'A piece of cheese'),
(5, 'Whiterun', 'Iron Armour', 30.0, 100, 'A standard Iron Armour'),
(6, 'Solitude', 'Steel Armour', 35.0, 150, 'A sturdy Steel Armour'),
(7, 'Whiterun', 'Health Potion', 0.5, 50, 'A potion that restores health'),
(8, 'Solitude', 'Mana Potion', 0.5, 50, 'A potion that restores mana'),
(9, 'Whiterun', 'Fire Salts', 0.2, 20, 'A rare ingredient'),
(10, 'Solitude', 'Frost Salts', 0.2, 20, 'A rare ingredient'),
(11, 'Whiterun', 'Conjuration for Novices', 1.0, 30, 'A book on the basics of
Conjuration'),
(12, 'Solitude', 'Destruction for Beginners', 1.0, 30, 'A book on the basics of
Destruction'),
(13, 'Whiterun', 'Scroll of Fireball', 0.5, 100, 'A scroll that casts a powerful
fireball'),
(14, 'Solitude', 'Scroll of Frostbite', 0.5, 100, 'A scroll that casts a powerful
frostbite'),
(15, 'Whiterun', 'Key to Dragonsreach', 0.1, 0, 'A key to the Dragonsreach in
Whiterun'),
(16, 'Solitude', 'Key to Blue Palace', 0.1, 0, 'A key to the Blue Palace in
Solitude');
```

-- *Inserting data into the Weapon table*

```
INSERT INTO Weapon (itemID, damageValue, reach, wType, effectID) VALUES (1, 8,
1.0, 'One-Handed', 7), (2, 10, 1.0, 'One-Handed', 7);
```

-- *Inserting data into the Armour table*

```
INSERT INTO Armour (itemID, aType, aRating, effectID) VALUES (5, 'Light Armour',
25, 8), (6, 'Heavy Armour', 35, 8);
```

-- *Inserting data into the Potion table*

```
INSERT INTO Potion (itemID, effectID) VALUES (7, 9), (8, 10);
```

-- *Inserting data into the Ingredients table*

```
INSERT INTO Ingredients (itemID, pItemID) VALUES (9, 11), (10, 11);
```

-- *Inserting data into the Book table*

```
INSERT INTO Book (itemID, author, title) VALUES (11, 'Falion', 'Conjuration for
Novices'), (12, 'Wuunferth the Unliving', 'Destruction for Beginners');
```

-- *Inserting data into the Scroll table*

```
INSERT INTO Scroll (itemID, effectID, spellID, scLevel) VALUES (13, 13, 1, 1),  
(14, 13, 2, 2);
```

```
-- Inserting data into the Food table
```

```
INSERT INTO Food (itemID, effectID) VALUES (3, 14), (4, 14);
```

```
-- Inserting data into the KeyItem table
```

```
INSERT INTO KeyItem (itemID, kDoor, kChest) VALUES (15, 'Dragonsreach',  
'Chest1'), (16, 'Blue Palace', 'Chest2');
```

Here is the data exported to CSV format:

Alteration

"3","3","Draugr"

"4","4","Dragon"

Armour

"5","Light Armour","25","8"

"6","Heavy Armour","35","8"

Book

"11","Falion","Conjuration for Novices"

"12","Wuunferth the Unliving","Destruction for Beginners"

Conjuration

"5","11","Familiar"

"6","15","Atronach"

Defensive

"3","100","0"

"4","0","50"

"8","N","25"

Destruction

"1","50","Fire"

"2","75","Frost"

Effect

"0","5","30","Invisibility"

"1","11","20","Fire Damage"

"2","15","30","Frost Damage"

"3","60","0","Healing"

"4","120","0","Armor Buff"

"5","30","10","Calm"

"6","30","10","Fear"

"7","0","0","Sword Damage"

"8","0","0","Armor Protection"

"9","60","0","Health Restoration"
"10","60","0","Mana Restoration"
"11","0","0","Ingredient Effect"
"12","0","0","Book Knowledge"
"13","0","0","Scroll Spell"
"14","0","0","Food Nutrition"
"15","0","0","Key Access"

Food

"3","14"
"4","14"

Illusion

"7","5","Bandit"
"8","6","Draugr"

Ingredients

"9","7"
"10","8"

Item

"1","Whiterun","Iron Sword","10","25","A standard Iron Sword"
"2","Solitude","Steel Sword","12","45","A sturdy Steel Sword"
"3","Whiterun","Apple","0.1","2","A fresh apple"
"4","Solitude","Cheese","1","5","A piece of cheese"
"5","Whiterun","Iron Armour","30","100","A standard Iron Armour"
"6","Solitude","Steel Armour","35","150","A sturdy Steel Armour"
"7","Whiterun","Health Potion","0.5","50","A potion that restores health"
"8","Solitude","Mana Potion","0.5","50","A potion that restores mana"
"9","Whiterun","Fire Salts","0.2","20","A rare ingredient"
"10","Solitude","Frost Salts","0.2","20","A rare ingredient"
"11","Whiterun","Conjuration for Novices","1","30","A book on the basics of Conjuration"
"12","Solitude","Destruction for Beginners","1","30","A book on the basics of Destruction"
"13","Whiterun","Scroll of Fireball","0.5","100","A scroll that casts a powerful fireball"
"14","Solitude","Scroll of Frostbite","0.5","100","A scroll that casts a powerful frostbite"
"15","Whiterun","Key to Dragonsreach","0.1","0","A key to the Dragonsreach in Whiterun"
"16","Solitude","Key to Blue Palace","0.1","0","A key to the Blue Palace in Solitude"

Keyitem

"15","Dragonsreach","Chest1"
"16","Blue Palace","Chest2"

Location

"Riften"
"Solitude"
"Whiterun"

Offensive

"1","50"
"2","75"
"7","12"

Potion

"7","9"
"8","10"

Restoration

"3","100","50"
"4","150","75"

Scroll

"13","13","1","1"
"14","13","2","2"

Spell

"1","Whiterun","Flames","100"
"2","Solitude","Frostbite","200"
"3","Whiterun","Healing","150"
"4","Solitude","Oakflesh","50"
"5","Whiterun","Summon Familiar","100"
"6","Solitude","Summon Atronach","200"
"7","Whiterun","Calm","75"
"8","Solitude","Fear","75"
"9","Solitude","Turn Undead","75"

Standing_Stone

"Shadow Stone","Riften","0"

Weapon

"1","8","1","One-Handed","7"
"2","10","1","One-Handed","7"

Link to GitHub with all SQL code / some C code for automation:

[BetterLordWilliam/COMP2714_TermAssignment \(github.com\)](https://github.com/BetterLordWilliam/COMP2714_TermAssignment)

Milestone3-Task3: Write SQL Statement




- You need to define several use-cases and write their corresponding queries.
- At least 10 queries should be written
- For group of 2 or more people, in addition to the first 10 queries, for each group member you need to add extra 3 use-cases.
- Write the SQL Statements, run them and take a snapshot from the results and insert them here.
- So the following items are expected:
 - 1- The usecase

- 2- The SQL query
- 3- The screenshot of the data

Example:

- 1- Use case: As an admin, I need to know the name and SSN of the managers of all department.
- 2- SQL Statement: Please see below
- 3- Snapshot: Please see below

```
1 • Select D1.mgrSSN, D2.name from Department as D1, Employee as D2
2 where D1.mgrSSN=D2.ssn
```

100%	23:2
Result Grid   Filter Rows: <input type="text" value="Search"/> Export: 	
mgrSSN	name
▶ 333445555	Franklin
888665555	James
987654321	Jennifer

[Write your answer here]

1. Use case: As a Skyrim player, I want to be able to see all the items I have in my inventory.

```
SELECT * FROM Item;
```

```
MariaDB [comp2714_termassignment]> SELECT * FROM Item;
```

itemID	locName	iName	iWeight	iValue	iDesc
1	Whiterun	Iron Sword	10	25	A standard Iron Sword
2	Solitude	Steel Sword	12	45	A sturdy Steel Sword
3	Whiterun	Apple	0.1	2	A fresh apple
4	Solitude	Cheese	1	5	A piece of cheese
5	Whiterun	Iron Armour	30	100	A standard Iron Armour
6	Solitude	Steel Armour	35	150	A sturdy Steel Armour
7	Whiterun	Health Potion	0.5	50	A potion that restores health
8	Solitude	Mana Potion	0.5	50	A potion that restores mana
9	Whiterun	Fire Salts	0.2	20	A rare ingredient
10	Solitude	Frost Salts	0.2	20	A rare ingredient
11	Whiterun	Conjuration for Novices	1	30	A book on the basics of Conjuration
12	Solitude	Destruction for Beginners	1	30	A book on the basics of Destruction
13	Whiterun	Scroll of Fireball	0.5	100	A scroll that casts a powerful fireball
14	Solitude	Scroll of Frostbite	0.5	100	A scroll that casts a powerful frostbite
15	Whiterun	Key to Dragonsreach	0.1	0	A key to the Dragonsreach in Whiterun
16	Solitude	Key to Blue Palace	0.1	0	A key to the Blue Palace in Solitude

2. 16 rows in set (0.002 sec)

1. Use case: As a Skyrim player, I want to be able to see all the spells at a specific location, such as Whiterun.

```
SELECT * FROM Spell WHERE locName = 'Whiterun';
```

```
MariaDB [comp2714_termassignment]> SELECT * FROM Spell WHERE locName = 'Whiterun';
```

spellID	locName	spellName	spellCost
1	Whiterun	Flames	100
3	Whiterun	Healing	150
5	Whiterun	Summon Familiar	100
7	Whiterun	Calm	75

2. 4 rows in set (0.042 sec)

1. Use case: As a Skyrim player, I want to be able to see all the items at a specific location, such as Whiterun.

```
SELECT * FROM Item WHERE locName = 'Whiterun';
```

```
MariaDB [comp2714_termassignment]> SELECT * FROM Item WHERE locName = 'Whiterun';
```

itemID	locName	iName	iWeight	iValue	iDesc
1	Whiterun	Iron Sword	10	25	A standard Iron Sword
3	Whiterun	Apple	0.1	2	A fresh apple
5	Whiterun	Iron Armour	30	100	A standard Iron Armour
7	Whiterun	Health Potion	0.5	50	A potion that restores health
9	Whiterun	Fire Salts	0.2	20	A rare ingredient
11	Whiterun	Conjuration for Novices	1	30	A book on the basics of Conjuration
13	Whiterun	Scroll of Fireball	0.5	100	A scroll that casts a powerful fireball
15	Whiterun	Key to Dragonsreach	0.1	0	A key to the Dragonsreach in Whiterun

2. 8 rows in set (0.043 sec)

3. Use case: As a Skyrim player, I want to be able to see the alteration spells which effect certain entities, such as Draugr.

```
SELECT * FROM Alteration WHERE sTarget = 'Bandit';
```

```
MariaDB [comp2714_termassignment]> SELECT * FROM Alteration WHERE sTarget = 'Draugr';
```

spellID	effectID	sTarget
3	3	Draugr

4. 1 row in set (0.001 sec)

1. Use case: As a Skyrim player, I want to be able to see the destruction spells and their information, which do X amount of damage (example 25).

2. Query:

```
SELECT *
FROM Destruction, Spell
WHERE Destruction.spellID = Spell.spellID
AND damage > 25;
```

```
MariaDB [comp2714_termassignment]> SELECT *
-> FROM Destruction, Spell
-> WHERE Destruction.spellID = Spell.spellID
-> AND damage > 25;

+-----+-----+-----+-----+-----+-----+-----+
| spellID | damage | stype | spellID | locName | spellName | spellCost |
+-----+-----+-----+-----+-----+-----+-----+
| 1 | 50 | Fire | 1 | Whiterun | Flames | 100 |
| 2 | 75 | Frost | 2 | Solitude | Frostbite | 200 |
+-----+-----+-----+-----+-----+-----+-----+

2 rows in set (0.003 sec)
```

- 3.

1. Use case: As a Skyrim player, I want to be able to see the restoration spells and their information, which restore X amount of damage (example 25).

2. Query:

```
SELECT *
FROM Restoration, Spell
WHERE Restoration.spellID = Spell.spellID
AND healthRestored > 25;
```

```
MariaDB [comp2714_termassignment]> SELECT *
-> FROM Restoration, Spell
-> WHERE Restoration.spellID = Spell.spellID
-> AND healthRestored > 25;

+-----+-----+-----+-----+-----+-----+-----+
| spellID | healthRestored | armourIncrease | spellID | locName | spellName | spellCost |
+-----+-----+-----+-----+-----+-----+-----+
| 3 | 100 | 50 | 3 | Whiterun | Healing | 150 |
| 4 | 150 | 75 | 4 | Solitude | Oakflesh | 50 |
+-----+-----+-----+-----+-----+-----+-----+

2 rows in set (0.001 sec)
```

- 3.

1. Use case: As a Skyrim player, I want to be able to see the weapons that I have which do more than X damage (example 5).

2. Query:

```
SELECT *
FROM Weapon, Item
WHERE Weapon.itemID = Item.itemID
AND Weapon.damageValue > 5;
```

```
MariaDB [comp2714_termassignment]> SELECT *
-> FROM Weapon, Item
-> WHERE Weapon.itemID = Item.itemID
-> AND Weapon.damageValue > 5;

+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| itemID | damageValue | reach | wType | effectID | itemID | locName | iName | iWeight | iValue | iDesc |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 1 | 8 | 1 | One-Handed | 7 | 1 | Whiterun | Iron Sword | 10 | 25 | A standard Iron Sword |
| 2 | 10 | 1 | One-Handed | 7 | 2 | Solitude | Steel Sword | 12 | 45 | A sturdy Steel Sword |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+

2 rows in set (0.004 sec)
```

- 3.

1. Use case: As a Skyrim player, I want to be able to see the armpurs that I have which provide X protection (example 10).

2. Query:

```
SELECT *
FROM Armour, Item
WHERE Armour.itemID = Item.itemID
AND Armour.aRating > 10;
```

```
MariaDB [comp2714_termassignment]> SELECT *
-> FROM Armour, Item
-> WHERE Armour.itemID = Item.itemID
-> AND Armour.aRating > 10;
```

itemID	aType	aRating	effectID	itemID	locName	iName	iWeight	iValue	iDesc
5	Light Armour	25	8	5	Whiterun	Iron Armour	30	100	A standard Iron Armour
6	Heavy Armour	35	8	6	Solitude	Steel Armour	35	150	A sturdy Steel Armour

2 rows in set (0.002 sec)

- 3.

1. Use case: As a Skyrim player, it would be useful to find the potions that I have in my inventory which hold a specific effect.

2. Query:

```
SELECT * FROM Potion, Item WHERE Potion.itemID = Item.itemID
AND Potion.effectID = 9;
```

```
MariaDB [comp2714_termassignment]> SELECT * FROM Potion, Item WHERE Potion.itemID = Item.itemID
-> AND Potion.effectID = 9;
```

itemID	effectID	itemID	locName	iName	iWeight	iValue	iDesc
7	9	7	Whiterun	Health Potion	0.5	50	A potion that restores health

1 row in set (0.002 sec)

- 3.

1. Use case: As a Skyrim player, it would be useful to find the scrolls in my inventory which require a me to have a certain level.

2. Query:

```
SELECT *
FROM Scroll, Item, Spell
WHERE Scroll.itemID = Item.itemID AND Scroll.spellID = Spell.spellID
AND Scroll.scLevel > 1;
```

```
MariaDB [comp2714_termassignment]> SELECT *
-> FROM Scroll, Item, Spell
-> WHERE Scroll.itemID = Item.itemID AND Scroll.spellID = Spell.spellID
-> AND Scroll.scLevel > 1;
```

itemID	effectID	spellID	scLevel	itemID	locName	iName	iWeight	iValue	iDesc	spellID	locName	spellName	spellCost
14	13	2	2	14	Solitude	Scroll of Frostbite	0.5	100	A scroll that casts a powerful frostbite	2	Solitude	Frostbite	200

1 row in set (0.001 sec)

- 3.

1. Use case: As a Skyrim player, to quickly gauge my wealth, it would be useful to view the total value of all the items in my inventory.

2. Query:

```
SELECT SUM(iValue) AS Total_Worth FROM Item;
```

```
MariaDB [comp2714_termassignment]> SELECT SUM(iValue) AS Total_Worth FROM Item;
```

Total_Worth
727

1 row in set (0.002 sec)

- 3.

1. Use case: As a Skyrim player, I want to be able to easily know which entities can be targeted by which spells. Part of this is knowing which entities can be targeted by more than one different kind of spell, whereupon I can discern the best spell for the scenario.

2. Query:

```
SELECT sTarget
FROM (
    SELECT sTarget FROM Alteration UNION ALL
    SELECT sTarget FROM Illusion) AS combined
GROUP BY sTarget
HAVING COUNT(*) > 1;
```

```
MariaDB [comp2714_termassignment]> SELECT sTarget
-> FROM (
-> SELECT sTarget FROM Alteration UNION ALL
-> SELECT sTarget FROM Illusion) AS combined
-> GROUP BY sTarget
-> HAVING COUNT(*) > 1;
```

```
+-----+
| sTarget |
+-----+
| Dragon  |
+-----+
```

```
1 row in set (0.001 sec)
```

3.

1. Use case: As a Skyrim player, when deciding what weapon I should use I need to consider various factors. Once such factor is the reach of the weapon, so it would be useful if I could quickly determine the weapon in my inventory with the longest reach.

2. Query:

```
SELECT iName, reach
FROM Item, Weapon
WHERE Item.itemID = Weapon.itemID
ORDER BY (reach) DESC LIMIT 1;
```

```
MariaDB [comp2714_termassignment]> SELECT iName, reach FROM Item, Weapon
-> WHERE Item.itemID = Weapon.itemID
-> ORDER BY (reach) DESC LIMIT 1;
```

```
+-----+-----+
| iName | reach |
+-----+-----+
| Iron Sword | 1 |
+-----+-----+
```

```
1 row in set (0.001 sec)
```

3.

1. Use case: As a Skyrim player, sometimes I need to figure out the places where I can make the most money. As such, it is useful to see the most expensive items at an area.

2. Query

```
SELECT locName, iName, iValue
FROM Item
WHERE iValue IN (
    SELECT DISTINCT iValue
    FROM Item AS t
    WHERE t.locName = Item.locName
)
AND iValue != 0
ORDER BY Item.locName, iValue DESC;
```

locName	iName	iValue
Solitude	Steel Armour	150
Solitude	Scroll of Frostbite	100
Solitude	Mana Potion	50
Solitude	Steel Sword	45
Solitude	Destruction for Beginners	30
Solitude	Frost Salts	20
Solitude	Cheese	5
Whiterun	Iron Armour	100
Whiterun	Scroll of Fireball	100
Whiterun	Health Potion	50
Whiterun	Conjuration for Novices	30
Whiterun	Iron Sword	25
Whiterun	Fire Salts	20
Whiterun	Apple	2

14 rows in set (0.00 sec)

3.

1. Use case: As a Skyrim player, I want to be able to quickly see the destruction spells I have which can deal the most amount of damage.
2. Query:

```
SELECT *
FROM Destruction, Spell
WHERE Destruction.spellID = Spell.spellID
ORDER BY (damage, spellCost) DESC;
```

MariaDB [comp2714_termassignment]> SELECT *

```
-> FROM Destruction, Spell
-> WHERE Destruction.spellID = Spell.spellID
-> ORDER BY (damage) DESC;
```

spellID	damage	stype	spellID	locName	spellName	spellCost
2	75	Frost	2	Solitude	Frostbite	200
1	50	Fire	1	Whiterun	Flames	100

2 rows in set (0.005 sec)

3.

1. Use case: As a Skyrim player, I want to be able to quickly see which effects applied to weapons, armour, etc, would have the greatest radius. I want to be able to do so regardless of the effect type.
2. Query:

```
(SELECT E1.effectID, radius, edesc FROM Effect E1, Defensive
WHERE Defensive.effectID = E1.effectID
AND radius > 0)
UNION
(SELECT E2.effectID, radius, edesc FROM Effect E2, Offensive
WHERE Offensive.effectID = E2.effectID
AND radius > 0)
ORDER BY (radius) DESC;
```



```

MariaDB [comp2714_termassignment]> (SELECT E1.effectID, radius, edesc FROM Effect E1, Defensive
-> WHERE Defensive.effectID = E1.effectID
-> AND radius > 0)
-> UNION
-> (SELECT E2.effectID, radius, edesc FROM Effect E2, Offensive
-> WHERE Offensive.effectID = E2.effectID
-> AND radius > 0)
-> ORDER BY (radius) DESC;
+-----+-----+-----+
| effectID | radius | edesc |
+-----+-----+-----+
| 2 | 30 | Frost Damage |
| 1 | 20 | Fire Damage |
+-----+-----+-----+
2 rows in set (0.005 sec)

```

3.

Milestone4

In Milestone 4, you are going to normalize the database (relational model) you have developed in Milestone 2.

Note: It is expected in this Milestone to change the relational model you have completed in Milestone 2 and consequently the database you have developed in Milestone 3. While it is expected to normalize your database schema in this milestone, it is not expected you to re-create your database and update the sql statements you have developed in module 3.

Milestone4-Task1:Identify full, partial and transitive functional dependencies in your design

- In Module 4, we learned about full, partial and transitive functional dependencies and we discussed how normalization process deals with such functional dependencies.
- The Milestone4-Task1: List all functional dependencies you have identified in Milestone 2. For each functional dependency identify whether it is full, partial or transitive functional dependencies and briefly explain why. Complete the following table. (Expand the table as needed)

Functional Dependency	
Partial, full or transitive? And why	
Functional Dependency	
Partial, full or transitive? And why	

Milestone4-Task2: Highest Normal Form

- Take your relational models and test them against the Normalization tests and complete this section:

Relation NOT in 1NF: (List all relations that are NOT even in 1NF, if any):

Relation with 1NF as highest normal form: (List all relations that are in 1NF but not 2NF)

Relation with 2NF as highest normal form: (List all relations that are in 2NF but not 3NF)

Relation with 3NF as highest normal form: (List all relations that are in 3NF but not BCNF)

Relation with BCNF as highest normal form:

Milestone4-Task3: Converting to 3NF

In this task you will normalize all your relations to 3NF and BCNF. If a relation is already in BCNF or 3NF, no change is needed.

Milestone4-Task4: Converting to BCNF

In this task you will normalize all your relations to BCNF. If a relation is already in BCNF, no change is needed.

- Check whether there are any functional dependencies that are not preserved as a result of normalization to BCNF.