

Introducción: Este proyecto consiste en la creación de un juego tipo dungeon cuya lógica está completamente basada en consultas SQL.

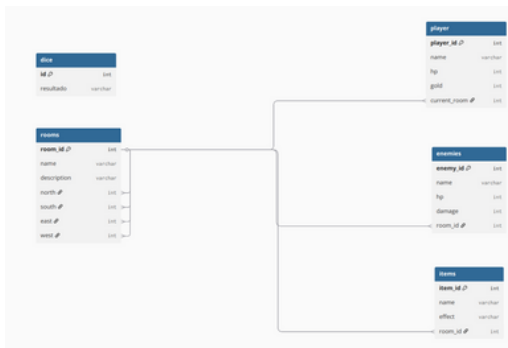
Tecnología utilizada:

- XAMPP
- MySQL Workbench
- PHP
- HTML, CSS y JavaScript

Descripción: En el juego avanzas por salas, combates enemigos, recoges ítems y tienes que derrotar a un jefe final.

Base de datos

La base de datos esta compuesta por tablas las cuales son player, rooms, enemies, items y dice, relacionadas entre sí para permitir el funcionamiento del juego.



Creación de las tablas.

Las tablas del proyecto fueron creadas utilizando sentencias CREATE TABLE, definiendo correctamente sus claves primarias y relaciones entre ellas.

```
CREATE DATABASE dungeon_game;  
USE dungeon_game;
```

```
CREATE TABLE rooms (  
    room_id INT PRIMARY KEY AUTO_INCREMENT,  
    name VARCHAR(50) NOT NULL,  
    description VARCHAR(255),  
    north INT,  
    south INT,  
    east INT,  
    west INT,  
    KEY (north),  
    KEY (south),  
    KEY (east),  
    KEY (west)  
);
```

```
CREATE TABLE player (  
    player_id INT PRIMARY KEY AUTO_INCREMENT,  
    name VARCHAR(50),  
    current_room INT,  
    hp INT DEFAULT 100,  
    gold INT DEFAULT 0  
);
```

```
CREATE TABLE enemies (  
    enemy_id INT PRIMARY KEY AUTO_INCREMENT,  
    name VARCHAR(50),  
    hp INT,  
    damage INT,  
    room_id INT  
);
```

```
CREATE TABLE items (  
    item_id INT PRIMARY KEY AUTO_INCREMENT,  
    name VARCHAR(50),  
    effect VARCHAR(100),  
    room_id INT  
);
```

Inserts dentro de las tablas.

Para inicializar el juego se utilizaron sentencias INSERT INTO que permiten definir el estado inicial del mapa, los enemigos, los ítems y el jugador.

Estos valores no se generan de manera aleatoria, ya que el mapa está diseñado previamente con el objetivo de que el jugador aprenda a optimizar rutas y estrategias mediante la repetición del juego.

```
INSERT INTO rooms (name, description) VALUES
('Entrada', 'El punto de inicio'),
('Pasillo', 'Un corredor oscuro'),
('Armería', 'Llena de armas viejas'),
('Sala de cofres', 'Hay un cofre misterioso'),
('Pre-jefe', 'La atmósfera es pesada'),
('Sala del Jefe', 'El enemigo final aguarda aquí');

UPDATE rooms SET north = 2 WHERE room_id = 1;
UPDATE rooms SET south = 1 WHERE room_id = 2;
UPDATE rooms SET east = 3 WHERE room_id = 2;
UPDATE rooms SET west = 2 WHERE room_id = 3;
UPDATE rooms SET north = 4 WHERE room_id = 2;
UPDATE rooms SET south = 2 WHERE room_id = 4;
UPDATE rooms SET north = 5 WHERE room_id = 4;
UPDATE rooms SET south = 4 WHERE room_id = 5;
UPDATE rooms SET north = 6 WHERE room_id = 5;
UPDATE rooms SET south = 5 WHERE room_id = 6;

INSERT INTO player (name, current_room) VALUES ('Jugador1', 1);

INSERT INTO enemies (name, hp, damage, room_id) VALUES
('Slime', 30, 5, 2),
('Esqueleto', 50, 10, 3),
('Mimic', 75, 12, 4),
('Mini-jefe', 120, 15, 5),
('Jefe Final', 200, 25, 6);

INSERT INTO items (name, effect, room_id) VALUES
('Poción', '+30 HP', 3),
('Cofre de Oro', '+50 GOLD', 4);
```

Relación entre tablas para la creación del mapa y mecánicas.

```
ALTER TABLE rooms
  ADD CONSTRAINT fk_room_north FOREIGN KEY (north) REFERENCES rooms(room_id),
  ADD CONSTRAINT fk_room_south FOREIGN KEY (south) REFERENCES rooms(room_id),
  ADD CONSTRAINT fk_room_east FOREIGN KEY (east) REFERENCES rooms(room_id),
  ADD CONSTRAINT fk_room_west FOREIGN KEY (west) REFERENCES rooms(room_id);

ALTER TABLE player
  ADD CONSTRAINT fk_player_room FOREIGN KEY (current_room) REFERENCES rooms(room_id);

ALTER TABLE enemies
  ADD CONSTRAINT fk_enemy_room FOREIGN KEY (room_id) REFERENCES rooms(room_id);

ALTER TABLE items
  ADD CONSTRAINT fk_item_room FOREIGN KEY (room_id) REFERENCES rooms(room_id);
```

Las tablas del sistema están relacionadas mediante claves foráneas, lo que permite mantener la integridad de los datos y asegurar que los eventos del juego se ejecuten correctamente.

Mecánicas principales.

Movimiento (x = 'north, south, east, west'):

UPDATE player

```
SET current_room = (
  SELECT x FROM rooms WHERE room_id =
current_room
)
WHERE (
  SELECT x FROM rooms WHERE room_id =
current_room
) IS NOT NULL;
```

Información:

Ver estado del jugador:

```
SELECT * FROM player;
```

Ver sala actual:

```
SELECT *  
FROM rooms  
WHERE room_id = (SELECT current_room FROM player);
```

Ver enemigos en la sala:

```
SELECT *  
FROM enemies  
WHERE room_id = (SELECT current_room FROM player);
```

Ver items en la sala:

```
SELECT *  
FROM items  
WHERE room_id = (SELECT current_room FROM player);
```

Combate:

Atacar enemigo (X = enemy_id)

```
UPDATE enemies
SET hp = hp - 20
WHERE enemy_id = X
AND room_id = (SELECT current_room FROM player)
AND (
    SELECT resultado
    FROM dice
    WHERE id = FLOOR(RAND() * 6) + 1
) = 'golpe';
```

Ataque del enemigo al jugador

```
UPDATE player
SET hp = hp - (
    SELECT damage
    FROM enemies
    WHERE enemy_id = X
    AND room_id = (SELECT current_room FROM player)
)
```

```
WHERE EXISTS (  
  SELECT 1  
  FROM enemies  
  WHERE enemy_id = X  
  AND room_id = (SELECT current_room FROM player)  
)  
AND (  
  SELECT resultado  
  FROM dice  
  WHERE id = FLOOR(RAND() * 6) + 1  
) = 'golpe';
```

Interfaz web.

Se desarrolló una pagina web que permite ejecutar consultas sql, visualizar resultados y ver el manual de comandos.

Limitaciones y alcance.

El proyecto funciona únicamente de manera local y no cuenta con seguridad avanzada.

Execute.php

```
1 <?php
2 // execute.php - simple single y multi-statement
3 header('Content-type: text/plain; charset=utf-8');
4
5 $host = "127.0.0.1";
6 $user = "root";
7 $password = "123456789";
8 $database = "dungeon_game";
9
10 $conn = new mysqli($host, $user, $password, $database);
11 if ($conn->connect_error) {
12     http_response_code(500);
13     die("Error de conexión: " . $conn->connect_error);
14 }
15
16 $query = trim($_POST['query']) ?? "";
17 if ($query === "") {
18     die("No se recibió ninguna query.");
19 }
20
21 // Seguridad básica: bloquear comandos peligrosos
22 $forbidden = ["drop", "truncate", "alter table", "create user", "grant", "revoke"];
23 $is_upper = strtoupper($query);
24 foreach ($forbidden as $bad) {
25     if (strpos($is_upper, $bad) !== false) {
26         die("Comando no permitido: $bad");
27     }
28 }
29
30 // Función para imprimir un result set en HTML
31 function fetch_result_rows($conn) {
32     $stmt = "";
33     while ($row = $conn->fetch_assoc()) {
34         $stmt .= json_encode($row, JSON_PRETTY_PRINT) . "<br>";
35     }
36     return $stmt;
37 }
38
39 // Si hay múltiples sentencias separadas por ';' usamos mysqli_query
40 if (strpos($query, ';') !== false) {
41     // Desprecupado de que la última sentencia termine sin espacios superfluos
42     $query = trim($query);
43     // Ejecutar todas las sentencias que genera sentencias vacías
44     $result = mysqli_query($conn, $query) or die(mysqli_error($conn));
45     $result = "";
46     if ($conn->mysqli_query($query)) {
47         do {
48             $result .= fetch_result_rows($conn);
49         } while ($result = mysqli_fetch_result($result));
50     }
51 }
```

Index.html

```
1 <!DOCTYPE html>
2 <html lang="es">
3 <head>
4     <meta charset="UTF-8">
5     <title>Dungeon Game</title>
6     <link rel="stylesheet" href="style.css">
7 </head>
8 <body>
9
10 <div>Dungeon Game</div>
11
12 <section>
13     <div>Reglamento del Juego</div>
14     <div>
15         <div>El jugador se mueve usando consultas SQL.</div>
16         <div>Si las acciones dependen del resultado del SQL.</div>
17         <div>Si el jugador afecta a filas, el ataque falló.</div>
18         <div>Si no se permite DROP el resultado de tablas.</div>
19     </div>
20 </section>
21
22 <section id="Manual">
23     <div>Manual de Comandos del Juego</div>
24     <div>
25         <div>Movimiento del jugador</div>
26         <div>
27             <div>-- Mover al norte</div>
28             <div>UPDATE player</div>
29             <div>SELECT north FROM rooms WHERE room_id = current_room</div>
30         </div>
31         <div>
32             <div>-- Mover al sur</div>
33             <div>UPDATE player</div>
34             <div>SELECT south FROM rooms WHERE room_id = current_room</div>
35         </div>
36         <div>
37             <div>-- Mover al este</div>
38             <div>UPDATE player</div>
39             <div>SELECT east FROM rooms WHERE room_id = current_room</div>
40         </div>
41         <div>
42             <div>-- Mover al oeste</div>
43             <div>UPDATE player</div>
44             <div>SELECT west FROM rooms WHERE room_id = current_room</div>
45         </div>
46     </div>
47 </section>
```

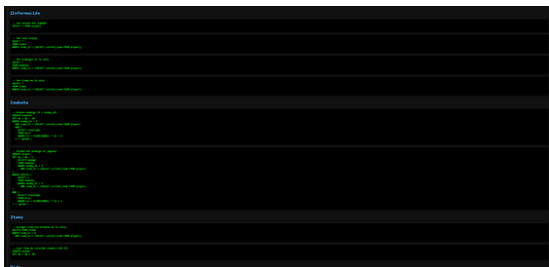
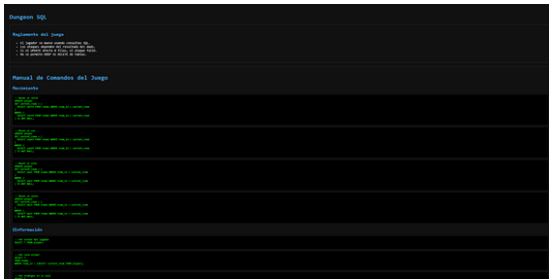

Script.js

```
1 function runQuery() {  
2   const query = document.getElementById("queryInput").value;  
3  
4   fetch("execute.php", {  
5     method: "POST",  
6     headers: {  
7       "Content-Type": "application/x-www-form-urlencoded"  
8     },  
9     body: "query=" + encodeURIComponent(query)  
10  })  
11  .then(response => response.text())  
12  .then(data => {  
13    document.getElementById("result").textContent = data;  
14  })  
15  .catch(error => {  
16    document.getElementById("result").textContent = error;  
17  });  
18 }  
19
```

Style.css

```
1 body {  
2   background-color: #f1f1f1;  
3   color: #333;  
4   font-family: Consolas, monospace;  
5   padding: 20px;  
6 }  
7  
8 h1, h2 {  
9   color: #000000;  
10 }  
11  
12 section {  
13   border: 1px solid #ccc;  
14   padding: 10px;  
15   margin-bottom: 20px;  
16 }  
17  
18 textarea {  
19   width: 100%;  
20   height: 120px;  
21   background-color: #f0f0f0;  
22   color: #000;  
23   border: 1px solid #ccc;  
24   padding: 10px;  
25 }  
26  
27 button {  
28   background-color: #000000;  
29   color: #fff;  
30   border: none;  
31   padding: 10px 20px;  
32   cursor: pointer;  
33   margin-top: 10px;  
34 }  
35  
36 button:hover {  
37   background-color: #000000;  
38 }  
39  
40 pre {  
41   background-color: #f0f0f0;  
42   padding: 10px;  
43   color: #000;  
44   overflow-x: auto;  
45 }  
46
```

Resultado final



El resultado final es un juego funcional controlado únicamente mediante consultas SQL, integrando una interfaz web para facilitar su uso y visualización.

