Mehreen Fatima 23F-0611

Meerab Awais 23F-0515

Fatima Nasir 23L-0714

**Database systems-Lab Final project**

**MY GAME STORE APPLICATION**

**MY GAMES STORE APPLICATION**

**Introduction**

The GameStore database project is a simulation of a digital storefront designed to manage the buying and selling of video games. The main purpose of the application is to create a structured, efficient, and scalable system that handles essential operations such as managing game inventory, storing user information, recording purchase transactions, and generating sales reports. This project demonstrates the practical application of relational database concepts in a real-world scenario, reflecting how e-commerce platforms manage large volumes of structured data. Through this project, key database design principles such as entity-relationship modeling, normalization, and query optimization are explored and implemented.

**Objectives**

* To design and implement a relational database system tailored for a video game store.
* To manage core data entities such as users, games, orders, and inventory.
* To apply normalization techniques to reduce redundancy and improve data consistency.
* To enforce data integrity using primary and foreign key constraints.
* To enable efficient querying for common use cases such as product searches, order history, and sales tracking.
* To provide a foundation for further development, such as adding user reviews, shopping carts, or admin analytics.

**Main Codes and Database section:**

**Index.php:**

<!DOCTYPE html>

<html lang="en">

<head>

<title>Techlinked</title>

<style>

body {

margin: 0;

padding: 0;

font-family: 'Arial', sans-serif;

}

header {

background-image: url('Purple\_Gaming\_Background.png');

background-size: cover;

background-position: center;

color: rgb(255, 255, 255);

text-align: center;

padding: 200px;

}

nav {

background-color: #490e4e;

color: #fff;

padding: 20px;

text-align: center;

}

nav a {

color: #fff;

text-decoration: none;

padding: 20px;

margin: 10px;

font-weight: bold;

}

nav a:hover {

background-color: rgb(192, 7, 230);

}

section {

padding: 20px;

text-align: center;

}

</style>

</head>

<body>

<nav>

<a href="index.php">Home</a>

<a href="products.php">Products</a>

<a href="services.php">Services</a>

<a href="contact.php">Contact</a>

<a href="cart.php">Cart</a>

<a href="tracking.php">Order Tracking</a>

<a href="user\_logout.php">Logout</a>

</nav>

<header>

<h1>Welcome to Techlinked Games</h1>

<p>Bringing Exciting Games to you without the middle man</p>

</header>

<section>

<h2>About Us</h2>

<p>

We here at techlinked believe that customers should pay exactly the amount the product costs

and not a penny more. Unfortunately, more and more retailers have started selling games

at a higher price due to low stock and high demand. But worry not — that's where we come in.

We've partnered directly with some of the top game manufacturers to bring you the newest and

latest games at their original prices — even discounted sometimes! You can trust us for

authenticity and quality.

</p>

</section>

</body>

</html>

**Userlogin.php:**

<?php

session\_start();

require 'db.php';

if ($\_SERVER["REQUEST\_METHOD"] == "POST") {

$username = $\_POST['email'];

$ad\_password = $\_POST['password'];

// Use a prepared statement with PDO5

$stmt = $pdo->prepare("SELECT \* FROM admins WHERE ad\_email = ?");

$stmt->execute([$username]);

$admin = $stmt->fetch();

if ($admin && password\_verify($ad\_password, $admin['ad\_password'])) {

$\_SESSION['admin'] = $admin['username']; // Keeps the admin username in the session

$\_SESSION['admin\_logged\_in'] = true; // Set this session variable to indicate logged-in status

header("Location: admin\_dashboard.php");

exit();

}

else {

echo "Invalid username or password.";

}

}

?>

<!DOCTYPE html>

<html lang="en">

<head>

<title>Login - Techlinked</title>

<style>

body { font-family: Arial; background-image: url('Purple\_Gaming\_Background.png'); background-size: cover; text-align: center; padding: 50px; color: white; }

form { background: rgb(191, 41, 237); padding: 20px; display: inline-block; border-radius: 8px; box-shadow: 0 0 10px rgba(0,0,0,0.1); }

input { padding: 5px; margin: 10px 0; width: 100%; }

button { padding: 10px; background: #490e4e; color: white; border: none; cursor: pointer; }

button:hover { background: rgb(205, 41, 237); }

.error { color: yellow; }

</style>

</head>

<body>

<h2>Login</h2>

<?php if (!empty($error)): ?>

<p class="error"><?php echo $error; ?></p>

<?php endif; ?>

<form method="POST" action="">

<input type="email" name="email" placeholder="Email" required><br>

<input type="password" name="password" placeholder="Password" required><br>

<button type="submit">Login</button><br>

<p>Don't have an account? <a href="signup.php">Sign up here</a></p>

</form>

</body>

</html>

**Products.php:**

<?php

session\_start();

if (!isset($\_SESSION['user\_id'])) {

header('Location: user\_login.php');

exit();

}

?>

<?php

include 'db.php';

$session\_id = session\_id();

// Handle Add to Cart

if ($\_SERVER['REQUEST\_METHOD'] === 'POST' && isset($\_POST['add\_to\_cart'])) {

$name = $\_POST['name'];

$price = $\_POST['price'];

$image = $\_POST['image'];

// Fetch g\_id and available quantity

$stmt = $pdo->prepare("SELECT g\_id, g\_quantity FROM game WHERE g\_name = :name");

$stmt->execute(['name' => $name]);

$game = $stmt->fetch();

if ($game && $game['g\_quantity'] > 0) {

$g\_id = $game['g\_id'];

// Update session cart

if (!isset($\_SESSION['cart'])) {

$\_SESSION['cart'] = [];

}

$found = false;

foreach ($\_SESSION['cart'] as &$item) {

if ($item['name'] === $name) {

$item['quantity'] += 1;

$found = true;

break;

}

}

if (!$found) {

$\_SESSION['cart'][] = [

'name' => $name,

'price' => $price,

'image' => $image,

'quantity' => 1

];

}

// Update quantity in `game` table

$stmt = $pdo->prepare("UPDATE game SET g\_quantity = g\_quantity - 1 WHERE g\_id = :g\_id AND g\_quantity > 0");

$stmt->execute(['g\_id' => $g\_id]);

// Insert into `cart` table or update quantity

$stmt = $pdo->prepare("SELECT \* FROM cart WHERE g\_id = :g\_id AND session\_id = :session\_id");

$stmt->execute(['g\_id' => $g\_id, 'session\_id' => $session\_id]);

$cartItem = $stmt->fetch();

if ($cartItem) {

$stmt = $pdo->prepare("UPDATE cart SET quantity = quantity + 1 WHERE g\_id = :g\_id AND session\_id = :session\_id");

} else {

$stmt = $pdo->prepare("INSERT INTO cart (g\_id, quantity, session\_id) VALUES (:g\_id, 1, :session\_id)");

}

$stmt->execute(['g\_id' => $g\_id, 'session\_id' => $session\_id]);

}

header('Location: products.php');

exit();

}

// Handle Clear Cart

if ($\_SERVER['REQUEST\_METHOD'] === 'POST' && isset($\_POST['clear\_cart'])) {

if (isset($\_SESSION['cart']) && !empty($\_SESSION['cart'])) {

foreach ($\_SESSION['cart'] as $item) {

$stmt = $pdo->prepare("SELECT g\_id FROM game WHERE g\_name = :name");

$stmt->execute(['name' => $item['name']]);

$game = $stmt->fetch();

if ($game) {

$g\_id = $game['g\_id'];

$stmt = $pdo->prepare("UPDATE game SET g\_quantity = g\_quantity + :quantity WHERE g\_id = :g\_id");

$stmt->execute([

'quantity' => $item['quantity'],

'g\_id' => $g\_id

]);

// Remove from cart table

$stmt = $pdo->prepare("DELETE FROM cart WHERE g\_id = :g\_id AND session\_id = :session\_id");

$stmt->execute([

'g\_id' => $g\_id,

'session\_id' => $session\_id

]);

}

}

unset($\_SESSION['cart']);

}

header('Location: products.php');

exit();

}

// Fetch products

$query = 'SELECT g\_id, g\_name, g\_price, g\_image, g\_description, g\_quantity FROM game';

$stmt = $pdo->query($query);

$products = $stmt->fetchAll();

?>

<!DOCTYPE html>

<html lang="en">

<head>

<title>Techlinked - Products</title>

<link rel="stylesheet" href="css\_styling/services.css">

<style>

body {

margin: 0;

padding: 0;

font-family: 'Arial', sans-serif;

background-color: #48114d;

}

header {

background-image: url('Purple\_Gaming\_Background.png');

background-size: cover;

background-position: center;

color: rgb(255, 255, 255);

text-align: center;

padding: 100px;

}

nav {

background-color: #490e4e;

color: #d62222;

padding: 15px;

text-align: center;

}

nav a {

color: #fff;

text-decoration: none;

padding: 15px;

margin: 0 10px;

font-weight: bold;

}

nav a:hover {

background-color: rgb(205, 41, 237);

}

section {

padding: 20px;

text-align: center;

}

.product {

display: inline-block;

margin: 20px;

text-align: center;

background-color: #f2f2f2;

border-radius: 10px;

padding: 20px;

width: 250px;

box-shadow: 0 4px 8px rgba(0, 0, 0, 0.1);

}

.product img {

max-width: 100%;

border-radius: 10px;

}

.product h2 {

font-size: 1.2em;

margin: 10px 0;

}

.product p {

font-size: 1em;

color: #555;

margin-bottom: 10px;

}

.order-button {

padding: 10px 20px;

background-color: #d62222;

color: #fff;

border: none;

border-radius: 5px;

cursor: pointer;

}

.order-button:hover {

background-color: #b01616;

}

.clear-cart-button {

padding: 10px 20px;

background-color: #28a745;

color: #fff;

border: none;

border-radius: 5px;

cursor: pointer;

}

.clear-cart-button:hover {

background-color: #218838;

}

</style>

</head>

<body>

<nav>

<a href="index.php">Home</a>

<a href="products.php">Product</a>

<a href="services.php">Services</a>

<a href="contact.php">Contact</a>

<a href="cart.php">Cart</a>

<a href="tracking.php">Order Tracking</a>

<a href="user\_logout.php">Logout</a>

</nav>

<header>

<h1>Latest Games</h1>

<p>Explore the newest games with exciting features.</p>

</header>

<?php foreach ($products as $product): ?>

<section class="product">

<img src="<?= htmlspecialchars($product['g\_image']); ?>" alt="<?= htmlspecialchars($product['g\_name']); ?>">

<div class="product-info">

<h2 style="color: black"><?= htmlspecialchars($product['g\_name']); ?></h2>

<p><?= htmlspecialchars($product['g\_description']); ?></p>

<p>Price: $<?= number\_format($product['g\_price'], 2); ?></p>

<p>Available Quantity: <?= $product['g\_quantity']; ?></p>

<form method="post">

<input type="hidden" name="name" value="<?= htmlspecialchars($product['g\_name']); ?>">

<input type="hidden" name="price" value="<?= htmlspecialchars($product['g\_price']); ?>">

<input type="hidden" name="image" value="<?= htmlspecialchars($product['g\_image']); ?>">

<button class="order-button" type="submit" name="add\_to\_cart">Add to Cart</button>

</form>

</div>

</section>

<?php endforeach; ?>

<?php if (isset($\_SESSION['cart']) && count($\_SESSION['cart']) > 0): ?>

<form method="post">

<button class="clear-cart-button" type="submit" name="clear\_cart">Clear Cart</button>

</form>

<?php endif; ?>

</body>

</html>

**Database queries section:**

**Tables:**

CREATE TABLE users (

id INT AUTO\_INCREMENT PRIMARY KEY,

email VARCHAR(100) UNIQUE NOT NULL,

password VARCHAR(255) NOT NULL,

name VARCHAR(100) NOT NULL,

created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP

);

CREATE TABLE admins (

id INT AUTO\_INCREMENT PRIMARY KEY,

ad\_email VARCHAR(100) UNIQUE NOT NULL,

ad\_password VARCHAR(255) NOT NULL,

username VARCHAR(100) NOT NULL,

created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP

);

CREATE TABLE orders (

order\_id INT AUTO\_INCREMENT PRIMARY KEY,

user\_id INT,

order\_date DATETIME DEFAULT CURRENT\_TIMESTAMP,

total\_amount DECIMAL(10, 2),

status VARCHAR(50) DEFAULT 'Pending'

);

CREATE TABLE game (

g\_id INT PRIMARY KEY AUTO\_INCREMENT,

g\_name VARCHAR(100),

g\_price DECIMAL(10,2),

g\_image VARCHAR(100),

g\_description TEXT,

g\_quantity INT

);

CREATE TABLE cart (

cart\_id INT AUTO\_INCREMENT PRIMARY KEY,

g\_id INT NOT NULL,

quantity INT NOT NULL DEFAULT 1,

session\_id VARCHAR(255),

added\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

FOREIGN KEY (g\_id) REFERENCES game(g\_id)

);

CREATE TABLE payment (

payment\_id INT AUTO\_INCREMENT PRIMARY KEY,

order\_id INT,

payment\_date DATETIME DEFAULT CURRENT\_TIMESTAMP,

payment\_status VARCHAR(50),

payment\_method VARCHAR(50),

amount DECIMAL(10, 2)

);

ALTER TABLE payment

ADD COLUMN user\_id INT,

ADD COLUMN created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP;

**Insertion query:**

INSERT INTO game (g\_name, g\_price, g\_image, g\_description, g\_quantity) VALUES

('GTA V', 999.99, 'gta5.png', 'An open-world action-adventure game where you complete missions and explore a crime-filled city as three different characters.', 25),

('FC 25', 1499.99, 'fc25.png', 'The latest football simulation game offering realistic gameplay, updated teams, and career modes.', 30),

('Forza Horizon 5', 899.99, 'forza.png', 'An open-world racing game set in Mexico with dynamic seasons and a vast variety of cars.', 15),

('Spider Man 2', 899.99, 'sman.png', 'An open-world racing game set in Mexico with dynamic seasons and a vast variety of cars.', 10),

('Tekken 8', 899.99, 'tekken8.png', 'An open-world racing game set in Mexico with dynamic seasons and a vast variety of cars.', 20),

('Tekken 7', 899.99, 'tekken7.png', 'An open-world racing game set in Mexico with dynamic seasons and a vast variety of cars.', 18),

('Call of Duty: Black Ops 6', 999.99, 'cod.png', 'A free-to-play battle royale game set in the Call of Duty universe, offering intense combat and large-scale multiplayer action.', 22),

('Call of Duty: Modern Warfare', 899.99, 'codmw.png', 'An open-world racing game set in Mexico with dynamic seasons and a vast variety of cars.', 16),

('Minecraft', 799.99, 'mine.png', 'A sandbox game where players build and explore virtual worlds made of blocks, with endless possibilities for creativity and survival.', 40),

('Cricket 24', 899.99, 'cri.png', 'An open-world racing game set in Mexico with dynamic seasons and a vast variety of cars.', 14),

('Assassins Creed Valhalla', 1499.99, 'ac.png', 'A historical action RPG that follows the Viking invasion of England, offering open-world exploration and strategic combat.', 13),

('Assasins Creed Odyssey', 899.99, 'aco.png', 'An open-world racing game set in Mexico with dynamic seasons and a vast variety of cars.', 12),

('WWE 2K25', 899.99, 'wwe.png', 'An open-world racing game set in Mexico with dynamic seasons and a vast variety of cars.', 15),

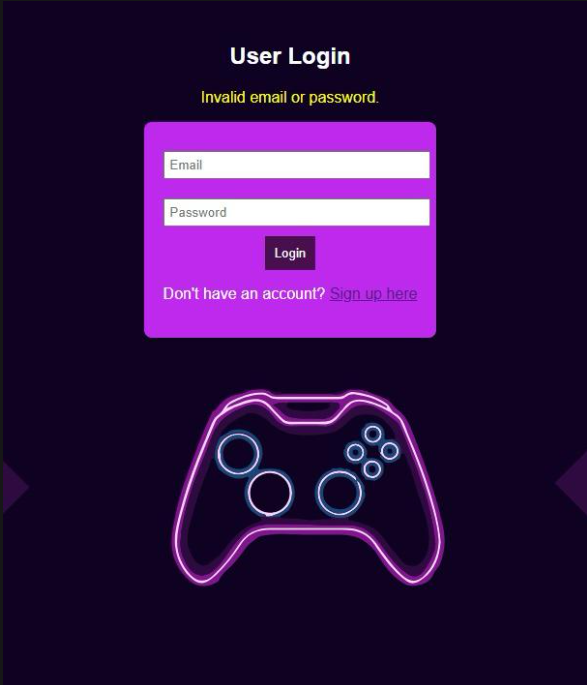
('Valorant', 899.99, 'valo.png', 'An open-world racing game set in Mexico with dynamic seasons and a vast variety of cars.', 17),

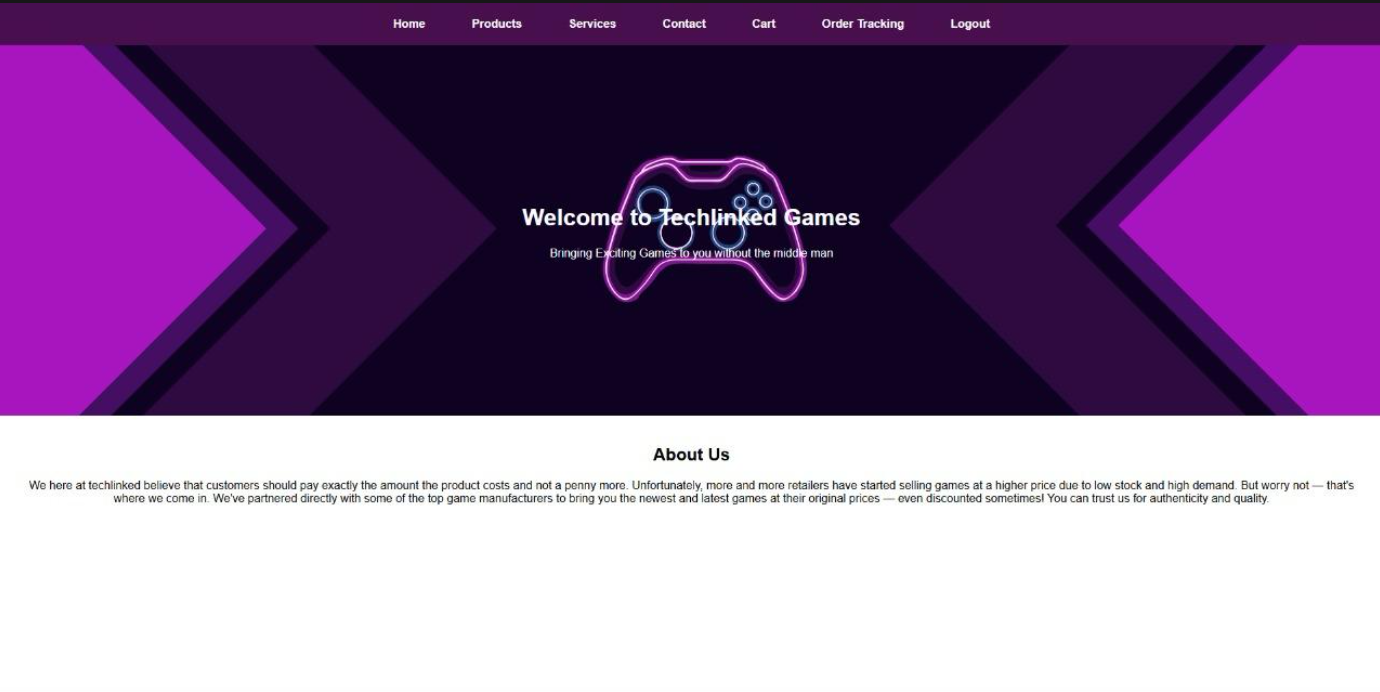
('Fortnite', 899.99, 'fort.png', 'An open-world racing game set in Mexico with dynamic seasons and a vast variety of cars.', 21),

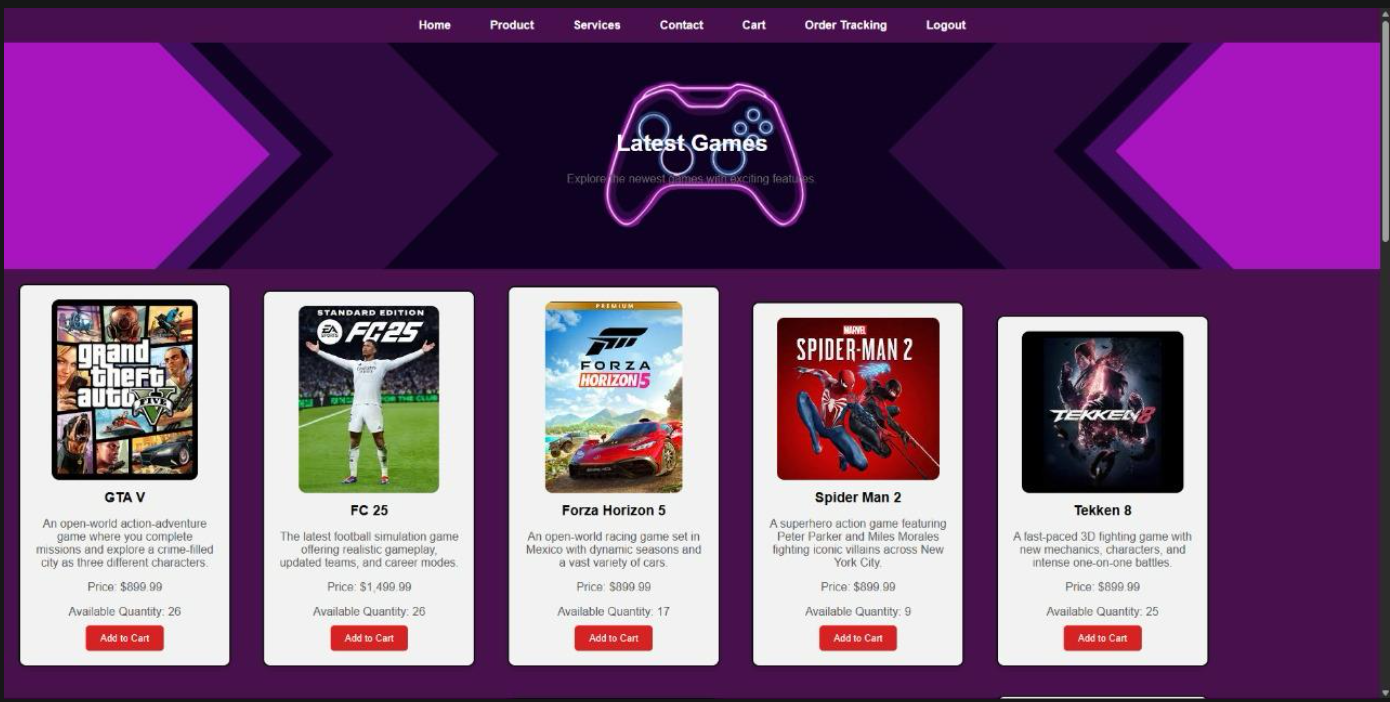
('NBA 2K25', 899.99, 'nba.png', 'An open-world racing game set in Mexico with dynamic seasons and a vast variety of cars.', 11),

('Red Dead Redemption 2', 1799.99, 'rdr.png', 'An open-world action-adventure game set in the American Wild West, following a gang of outlaws on their journey through the changing frontier.', 8),

('The Witcher 3: Wild Hunt', 1499.99, 'tw.png', 'An action RPG set in a vast open world, where you play as Geralt of Rivia, a monster hunter, as you embark on a quest to find your adopted daughter.', 9); **Error handling and outputs:**







**Database::techlinked**

