

Shubham Kumar 4th Year, Department of CSE, Gandhi Institute for Technology, BPUT, India
shubhamk2021@gift.edu.in

Under the guidance of mrs. **Smruti Smaraki Sarangi**

Department of Computer Science and Engineering, Gandhi Institute for Technology, BPUT, India

Abstract—

E-commerce has become a vital part of modern global trade, transforming how consumers and businesses interact. As the digital marketplace becomes increasingly saturated, factors such as user experience (UX) and secure transaction methods become key differentiators in determining the success of online retail platforms. This paper aims to explore the theoretical underpinnings and best practices in designing e-commerce websites that enhance user experience while maintaining high standards of payment security. Through the lens of front-end technologies such as HTML, CSS, and JavaScript, the study examines design principles, interface usability, and modern approaches to secure payment integrations like tokenization and third-party gateways. Findings suggest that intuitive design and trust-building security mechanisms are paramount to user retention, brand loyalty, and conversion rates.

I. INTRODUCTION

In the modern digital age, industries across various domains are leveraging technology to enhance efficiency, accessibility, and user experience. One such niche domain is the equine industry, which includes the buying, selling, and listing of horses for various purposes such as racing, breeding, and companionship. Traditionally, this market has relied heavily on word-of-mouth, local advertisements, and in-person auctions, often making it inaccessible to a wider audience. Recognizing this gap, the **Eminent Equine** platform has been developed to bring transparency, convenience, and broader reach to equine transactions.

Eminent Equine is a web-based application designed to simplify and modernize the way horses are traded. It acts as an online marketplace where users can create accounts, list horses for sale, browse available listings, and connect with buyers or sellers. The platform incorporates several user-centric features such as image uploads, category-based filtering, responsive design for multiple devices, and secure form-based contact options.

The system is built using standard and reliable web technologies: **HTML, CSS, JavaScript** for the front-end interface, and **PHP with MySQL** for the back-end logic and database management. This stack allows for fast, secure, and scalable web development. Additionally, the platform is live and accessible globally at <https://eminentequine.com.au>, offering real-time services to users in the equine industry.

The primary objective of this project is to develop a functional, user-friendly, and efficient horse trading platform that can be used by breeders, buyers, trainers, and horse enthusiasts. Through this system, the aim is to:

- Reduce the time and effort required in finding or selling horses
- Offer a centralized digital platform for horse trading
- Enhance trust and accessibility in the equine marketplace

This report details the entire project lifecycle including background research, system planning, design specifications, implementation methods, testing strategies, and possible future enhancements. The development process followed a structured approach to ensure quality, maintainability, and usability of the final product.

II. LITERATURE SURVEY

A literature survey involves the study and analysis of existing work related to a specific domain. In the context of this project, the domain is online horse trading and marketplace web applications.

This chapter provides insights into existing technologies, platforms, and systems, and evaluates their relevance to the goals of **Eminent Equine**. It also highlights the limitations of current solutions and the innovations introduced by this project

III. SYSTEM DESIGN

During the planning and design phase of the **Eminent Equine** platform, it was necessary to select an appropriate database system to manage structured data efficiently. After analyzing different options, **MySQL** was chosen due to its compatibility with PHP, structured relational model, and wide adoption in web development.

The system is designed around three core entities: **Users**, **Horses**, and **Messages**. These entities were identified based on user needs and listing functionality. The relationships among them follow a normalized relational model to ensure data integrity, reduce redundancy, and support scalability.

IV. IMPLEMENTATIONS

The Eminent Equine website enables users to list, buy, and sell horses through a fully functional web interface. This chapter explains the connection to the database and core parts of the code used in the implementation.

Database Connection

The backend uses **PHP** to establish a connection to a **MySQL** database. All dynamic content such as horse listings, user details, and inquiries are fetched and stored using SQL queries.

A reusable file, typically named **db.php**, is created to handle the database connection. This file is included in all PHP scripts that interact with the database.

Sample Code – db.php php

```
<?php

$host = 'localhost';

$user = 'root'; $password = "";

$dbname = 'eminentequine';

$conn = new mysqli($host, $user, $password, $dbname);

// Check connection    if ($conn-

>connect_error) { die("Connection failed: " .

$conn->connect_error);

}

?>
```

This ensures that the database connection is centralized and easy to manage or modify.

Horse Listing Display Code

To display all available horses, PHP retrieves data from the **horses** table and outputs it dynamically in HTML.

Sample Code – Display Horses php

```

<?php

include 'db.php';

$sql = "SELECT * FROM horses";

$result = $conn->query($sql);

while ($row = $result->fetch_assoc()) {      echo "<div class='horse'>"; echo "<img src='uploads/"
. $row['image'] . "' alt='Horse Image'>"; echo "<h3>" . $row['name'] . "</h3>"; echo "<p>Price: $" .
$row['price'] . "</p>"; echo "<p>" .
$row['description'] . "</p>"; echo "</div>";

}

?>

```

This dynamically generates a grid of horses with their images, names, prices, and descriptions.

Add Horse Form (Admin Panel)

Admins can add new horses for sale through a web form. This form collects input and submits it via POST to the backend.

Sample Code – HTML Form

```

html

<form action="add_horse.php" method="POST" enctype="multipart/formdata">

<input type="text" name="name" placeholder="Horse Name" required>
<input type="number" name="price" placeholder="Price" required>

<textarea name="description" placeholder="Description"></textarea>

<input type="file" name="image" required>

<button type="submit">Add Horse</button>

</form>

```

Backend Insert Logic

The form submission is handled in the file `add_horse.php`, where the image is uploaded and data is inserted into the database.

Sample Code – add_horse.php php

```

<?php include 'db.php';

$name = $_POST['name'];

```

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

$description = $_POST['description'];

$image = $_FILES['image']['name'];

$target = "uploads/" . basename($image);

// Upload image to server move_uploaded_file($_FILES['image']['tmp_name'], $target);

// Insert into database

$sql = "INSERT INTO horses (name, price, description, image) VALUES ('$name', '$price', '$description', '$image')";

$conn->query($sql);

// Redirect to dashboard header("Location: admin_dashboard.php");
?>
```

This completes the essential backend logic to manage horse listings from both the admin and user perspectives.

V. TESTING

The Waterfall Model was selected as the primary development methodology for this project. This model divides the software development process into distinct sequential phases, each of which must be completed before the next begins. It is suitable for academic and prototypebased projects due to its clarity, documentation, and ease of tracking progress.

Testing

Performed unit testing for each module.

Conducted integration testing to ensure data flows correctly across the platform. Simulated real-world usage scenarios.

Deployment

Tested the project locally using XAMPP.

Deployed the complete system to a live hosting environment: <https://eminentequine.com.au>

VI. CONCLUSIONS

The development of the Eminent Equine platform successfully demonstrates how web technologies can be used to digitize and streamline the horse trading process. The project met all core functional requirements, including secure user registration, dynamic listing creation, database integration, and responsive front-end design.

Using a combination of HTML, CSS, JavaScript, PHP, and MySQL, the system provides an efficient and accessible solution for users to buy, sell, and manage horse listings in real time. By deploying the platform at <https://eminentequine.com.au>, the project moved beyond theory and into a real-world, usable application.

The platform also reflects solid database design principles, user interface clarity, and modular coding practices. Overall, the project has fulfilled its objectives and has strong potential for realworld impact in the equine community.

VII. REFERENCES

Mozilla Developer Network (MDN). (n.d.). Using the Fetch API. The MDN Web Docs offered detailed documentation and examples for using the JavaScript Fetch API, which played a vital role in

making asynchronous server requests to retrieve or submit data without reloading the page. This helped enhance the interactivity of the website.

Available at: https://developer.mozilla.org/en-US/docs/Web/API/Fetch_API/Using_Fetch

Bootstrap. (n.d.). Bootstrap 5 Documentation. Although custom CSS was used primarily, Bootstrap's responsive grid system and UI components were consulted during the design of the admin dashboard and mobile-friendly layouts. The official documentation provided guidance on layout structures and utility classes.

Available at: <https://getbootstrap.com/docs/5.3/getting-started/introduction/> MySQL Documentation.

(n.d.). MySQL 8.0 Reference Manual. This official

documentation provided detailed insights into SQL queries, database structure, indexing, and optimization strategies used in the backend of the horse listing application. It served as a reference while designing the database schema.

Available at: <https://dev.mysql.com/doc/refman/8.0/en/>

PHP Manual. (n.d.). PHP Official Documentation. This was a primary reference for understanding server-side logic, handling form data, file uploads, and session management in the project. It also guided the use of secure coding practices and error handling. Available at:

<https://www.php.net/manual/en/>

Eminent Equine. (2025). Horse Buying, Selling, and Listing Platform. The live website developed as part of this project served as a practical implementation of all discussed technologies. It features dynamic data loading, form processing, image uploads, and a custom admin interface.

Available at: <https://www.eminentequine.com.au>