# **TevoKicks Final Project Report**



## 1. Executive Summary

TevoKicks is a sneaker e-commerce website designed and built as a final project to demonstrate full-stack web development skills using HTML, CSS, PHP, and MySQL. The site features dynamic product pages, user authentication, a shopping cart system, and payment integration. It includes product filtering by gender and size, a structured database, and customer feedback through a rating system. The project aimed to create a functional, visually appealing, and secure online shopping experience tailored to sneaker enthusiasts.

# 2. Background and Motivation

My love for sneakers and the need for user-friendly platforms that cater to various age groups and demographics inspired me to create TevoKicks. This project was chosen to enhance my practical web development experience, simulate real-world application functionality, and explore payment system integrations. It also reflects my personal interest in fashion and e-commerce trends.

# 3. Problem Statement and Objectives

The primary objective was to create a web-based sneaker store that supports product listing, customer reviews, shopping cart functionality, and secure checkout. Specific goals included:

- Implementing a user registration and login system.
- Creating separate pages for the Men, Women, and Kids categories.
- Allowing users to select sneaker sizes and view availability.
- Supporting ratings and reviews for sneakers.
- Integrating PayPal for live transaction processing.
- Developing a structured and normalized database schema.

# 4. Project Design and Implementation

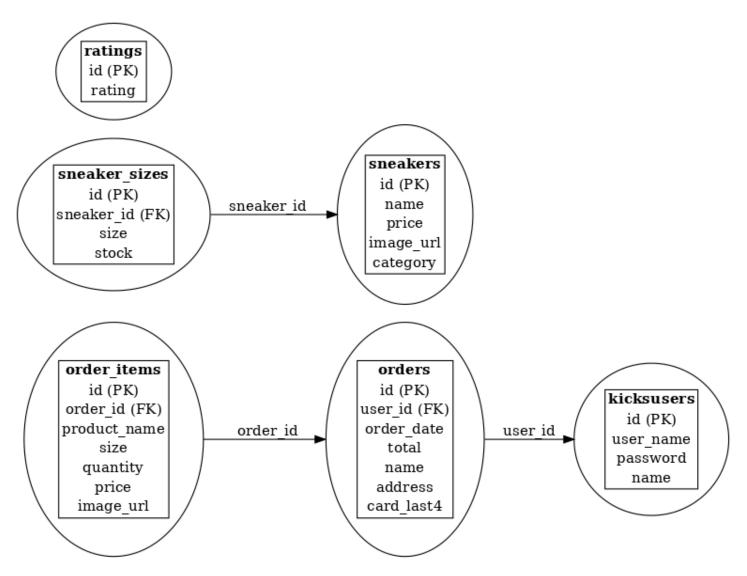
#### Technologies Used:

- Frontend: HTML, CSS (with Bootstrap for layout)
- Backend: PHP using procedural MySQLi
- Database: MySQL via phpMyAdmin
- Other: PayPal Sandbox (for testing payments)

#### Major Components:

- User System: kicks users table storing login credentials
- Order Management: orders and order\_items tables tracking user purchases
- **Product Catalog**: sneakers and sneaker\_sizes tables storing sneaker info and inventory
- Feedback Mechanism: ratings table for user-submitted reviews

#### Database Schema:



**UI Snapshots:** 

Login/Register Page:

Continue as guest	Login
l	Register
	Continue as guest

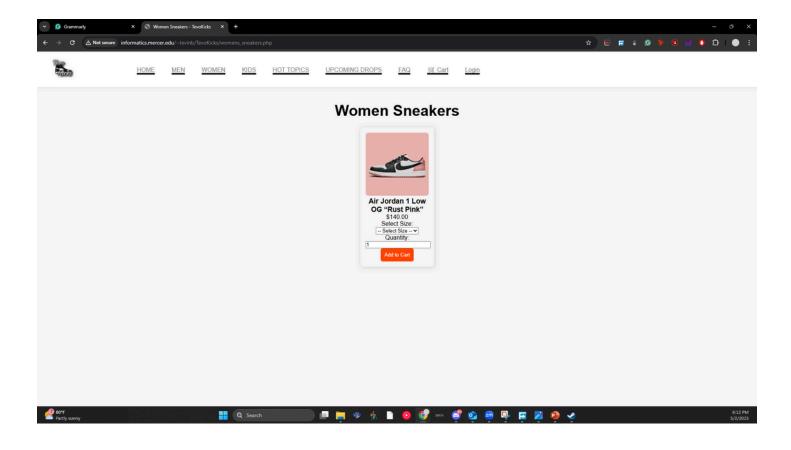
# **User Registration**

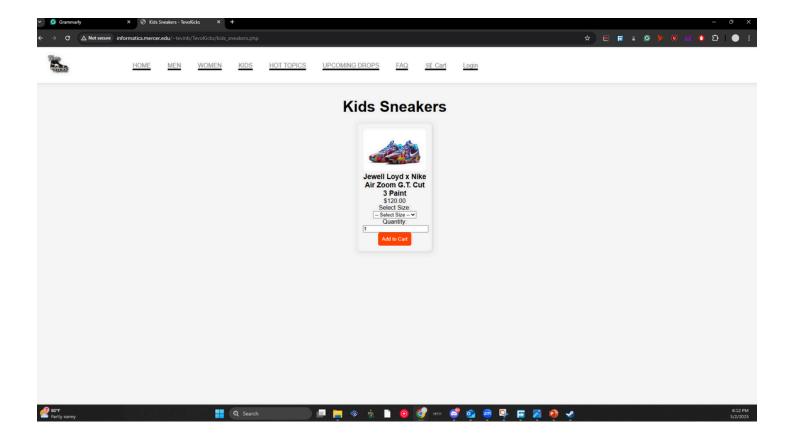
Username:	
Password:	
Name:	
	Register

Sneaker Listing (Men, Women, Kids):

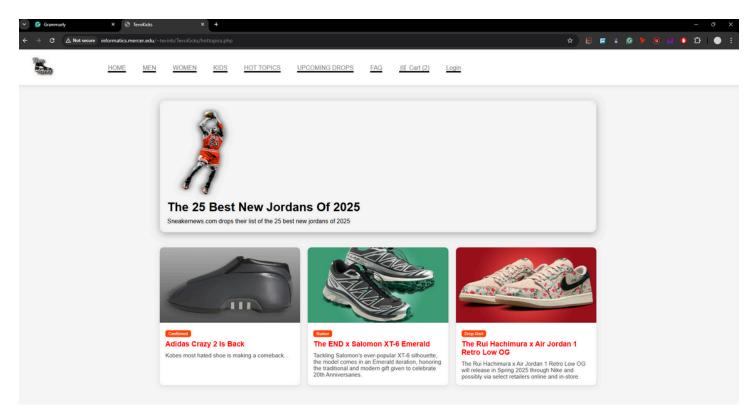
#### Men Sneakers



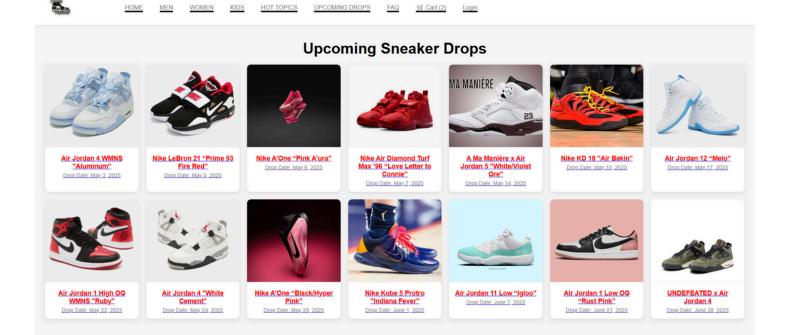




#### Hot Topics:



**Upcoming Drops:** 



#### **Shopping Cart:**

Your Shopping Cart Air Jordan 1 Low OG "Rust Pink"	
Size: 7.5 — \$140.00 × 1 Total: \$140.00	Remove
Continue Shopping Proceed to Checkout	Cart Total: \$140.00

#### Checkout:

Checkout								
Product		Size	Price	Quantity	Total			
Air Jordan 1 Low OG "Rust Pink"		7.5	\$140.00	1	\$140.00			
Card N	ame: ng Address: Number: tion Date (MM/	Place Order	nent Info					
		← Back to Ca	irt					

# 5. Testing / Evaluation / Validation

Extensive manual testing was performed to ensure that each feature functioned correctly:

- Verified user login and registration with session persistence
- Confirmed product filtering and size availability across all pages
- Simulated cart operations and validated order insertions
- Ensured ratings were stored and retrieved correctly from the database

Issues like empty cart sessions and NULL values in order fields were encountered and resolved through proper input validation and session handling.

# 6. Discussions

#### **Challenges Encountered:**

- Debugging NULL values in order insertion
- Handling session management between cart and checkout
- Linking ratings properly to sneaker entries

#### Solutions:

- Refined form handling and POST variable checks
- Ensured session\_start() and isset() were used correctly
- Normalized tables and connected sneaker IDs properly

#### Lessons Learned:

- Importance of ERD planning before coding
- How to debug PHP session-related issues
- Best practices in structuring e-commerce logic

#### Suggestions for Future Work:

- Add admin backend for product uploads.
- Support user profile editing and order history
- Include advanced filters (brand, price range)

## 7. References

- W3Schools (PHP, HTML, CSS)
- PayPal Developer Documentation
- Stack Overflow
- ChatGPT for debugging and guidance