Design and Implementation of an E-commerce Platform for New and Used Products

Joan Sebastian Duran Pradilla
Maiker Alejandro Hernandez Archila
Systems Engineering
Universidad Distrirtal

***Abstract*—This work presents an e-commerce system that allows the sale of new and used products through a centralized platform. This solution improves the management of users, items, payments, and orders through well-defined classes. Additionally, the results demonstrate the efficiency of the design in handling multiple types of products, from cars to video games.**

I. Introduction

E-commerce has grown exponentially, and managing new and used products is a crucial challenge for e-commerce platforms. Many existing solutions focus solely on new products, limiting the options for resale platforms. Our proposal aims to identify and fill this gap. For this purpose, an application has been developed that allows the buying and selling of both types of products. There are solutions such as *Mercado Libre* and *eBay* that allow the resale of used products. However, the proposed architecture offers greater flexibility in managing product types through a hierarchical structure based on class inheritance, allowing the system to easily scale to new product categories.

II. Goals

The main research question is: *How to design an efficient system that manages the sale of new and resale products simultaneously on an e-commerce platform?* The expected final product is a market system that can adapt to different product categories without compromising scalability or code complexity.

A. UML System Diagram

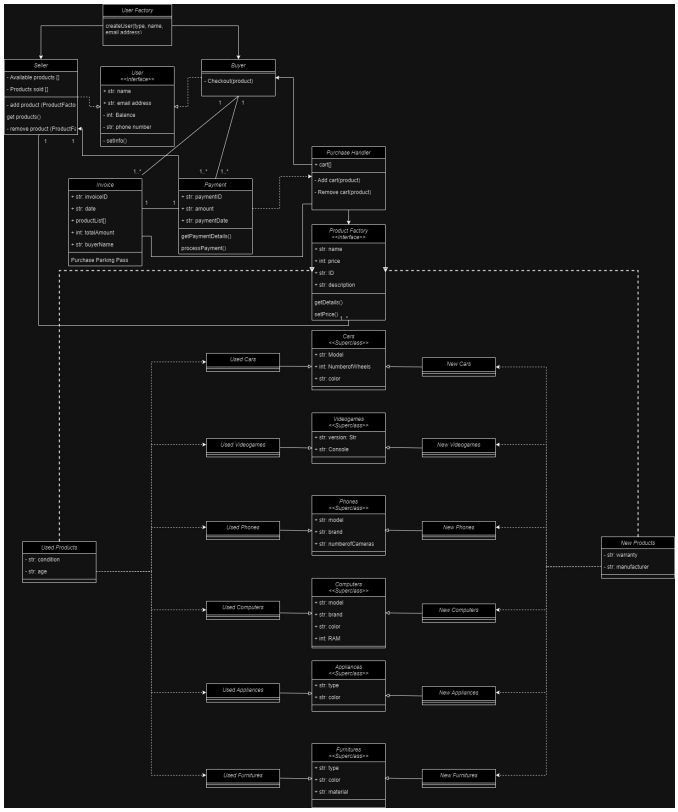

Fig. 1: UML Diagram of the proposed system

B. Solution Description

The proposed solution is based on a program model called *Shopi*. Shopi primarily handles object-oriented programming (OOP) and some core programming principles. It deals with both new and used products through a set of generic and specialized classes. The system utilizes a `ProductFactory` class, which acts as a product factory, along with subclasses like `Cars`, `Videogames`, and `Phones`. By doing so, it can create specific instances of products. Ultimately, this approach aims to develop a sales page, making it easier for users to buy and sell items.

III. Results

Unit tests were conducted on key functionalities such as product creation, adding products to the cart, and payment completion. Throughout 10 tests, covering the most common buying and selling scenarios, a success rate of approximately 95% was achieved. Additionally, integration tests were implemented to verify the successful interaction between user, product, and payment components. Table I summarizes the results.

TABLE I: Summary of Results from Unit Tests

Test	**Result**	**percentage of users liking**
Product creation	Success	100%
Add to cart	Success	60%
Processed payment	Success	100%

Acceptance tests were also conducted with end-users, who reported ease of use and the system's ability to handle large volumes of products.

IV. Conclusion

The proposed marketplace system is structured to effectively address the need for a system that manages both new and used products efficiently. Its structure, based on inheritance and abstract classes, allows for scalability, and the test results demonstrate that it is reliable and easy to maintain. As such, it is presented as an ideal platform for future expansions, such as the addition of new product categories.