



**ĐẠI HỌC NGUYỄN TRÃI**  
**TRUNG THỰC**

**TRƯỜNG ĐẠI HỌC NGUYỄN TRÃI**  
**KHOA CÔNG NGHỆ THÔNG TIN**

---

# **BÁO CÁO THỰC TẬP CHUYÊN ĐỀ & ĐỒ ÁN CHUYÊN MÔN**

**ĐỀ TÀI:**

**XÂY DỰNG WEBSITE KINH DOANH  
HỘP MÙ (BLIND BOX) & GACHA GAME**

**Giảng viên hướng dẫn: Th.S Trịnh Văn Chung**

**Sinh viên thực hiện: Nguyễn Xuân Vinh**

Mã sinh viên: 2310900117

Lớp: K23CNT1

Khóa: 2023 - 2027

**HÀ NỘI - 2025**

## LỜI CẢM ƠN

Lời đầu tiên, em xin gửi lời cảm ơn chân thành và sâu sắc nhất đến thầy Th.S Trịnh Văn Chung, người đã tận tình hướng dẫn, chỉ bảo và định hướng cho em trong suốt quá trình thực hiện đề tài này. Sự nhiệt huyết, kiến thức chuyên môn sâu rộng và kinh nghiệm thực tế của thầy là nguồn động lực to lớn giúp em vượt qua những khó khăn, thử thách trong quá trình nghiên cứu và phát triển sản phẩm.

Đồng thời, em cũng xin gửi lời tri ân đến quý thầy cô trong Khoa Công nghệ Thông tin - Trường Đại học Nguyễn Trãi đã trang bị cho em những kiến thức nền tảng vững chắc về lập trình, cơ sở dữ liệu, công nghệ web và phương pháp luận phát triển phần mềm. Những kiến thức này là nền tảng quan trọng giúp em hoàn thành đề tài một cách hiệu quả.

Em cũng xin cảm ơn các bạn sinh viên trong lớp K23CNT1 đã nhiệt tình tham gia khảo sát, đóng góp ý kiến và thử nghiệm hệ thống, giúp em có những số liệu thực tế và feedback quý báu để hoàn thiện sản phẩm.

Cuối cùng, em xin cảm ơn gia đình và người thân đã luôn ở bên cạnh, động viên, hỗ trợ em cả về vật chất lẫn tinh thần trong suốt thời gian học tập và thực hiện đề tài.

*Mặc dù đã cố gắng hết sức, nhưng do thời gian và kinh nghiệm còn hạn chế, báo cáo không thể tránh khỏi những thiếu sót. Em rất mong nhận được sự góp ý của thầy cô để em có thể hoàn thiện hơn nữa.*

Hà Nội, ngày 16 tháng 12 năm 2025

**Sinh viên thực hiện**

**Nguyễn Xuân Vinh**

# MỞ ĐẦU

## 1. Lý do chọn đề tài

Trong bối cảnh cuộc cách mạng công nghiệp 4.0, thương mại điện tử đã trở thành xu hướng không thể thiếu trong đời sống hiện đại. Theo báo cáo của Hiệp hội Thương mại điện tử Việt Nam (VECOM), thị trường TMĐT Việt Nam năm 2024 đạt giá trị 20 tỷ USD với tốc độ tăng trưởng 25%/năm, cho thấy tiềm năng phát triển rất lớn.

Nổi bật trong thời gian gần đây là trào lưu "Blind Box" - mô hình kinh doanh độc đáo nơi người mua không biết trước vật phẩm bên trong hộp. Xuất xứ từ Nhật Bản với tên gọi "Gashapon" hay "Gacha", mô hình này đã lan rộng sang Trung Quốc, Hàn Quốc và đang dần phổ biến tại Việt Nam.

Xuất phát từ nhu cầu thực tế và mong muốn mang lại trải nghiệm mới mẻ, thú vị cho cộng đồng yêu thích sưu tập đồ chơi, anime, manga tại Việt Nam, em quyết định chọn đề tài "Xây dựng Website Kinh doanh Hộp Mù & Gacha Game".

## 2. Mục tiêu nghiên cứu

Mục tiêu tổng quát của đề tài là xây dựng một hệ thống thương mại điện tử hoàn chỉnh, kết hợp yếu tố giải trí (gamification) với thương mại điện tử truyền thống:

1. Xây dựng hệ thống quản lý người dùng với tính năng đăng ký, đăng nhập, quản lý profile và ví điện tử, áp dụng Spring Security để đảm bảo bảo mật cao
2. Phát triển Core Gacha System với thuật toán RNG công bằng, hỗ trợ nhiều mức độ hiếm với tỉ lệ drop rate có thể cấu hình
3. Xây dựng Inventory System để quản lý các vật phẩm trúng được, cho phép user xem, lọc, bán lại hoặc yêu cầu giao hàng
4. Tích hợp Pity System - cơ chế bảo hiểm đảm bảo fairness, user sẽ nhận được item hiếm sau số lần mở nhất định
5. Phát triển Shipment Management kết nối kho ảo với logistics thực tế, xử lý workflow từ request đến delivery

6. Xây dựng Admin Dashboard với các chức năng quản lý toàn diện và thống kê trực quan

### 3. Phương pháp nghiên cứu

#### 3.1. Phương pháp nghiên cứu lý thuyết

Thu thập và phân tích tài liệu học thuật, documentation về Spring Boot, Hibernate, các best practices trong thiết kế hệ thống thương mại điện tử. Nghiên cứu case study về các game Gacha nổi tiếng (Genshin Impact, Honkai Star Rail) để hiểu cơ chế Pity System và cân bằng tỉ lệ.

#### 3.2. Phương pháp khảo sát thực tế

Khảo sát trực tiếp tại các cửa hàng Blind Box ở Hà Nội (POP MART, Miniso), quan sát quy trình bán hàng và phỏng vấn chủ shop. Khảo sát online với 50 sinh viên yêu thích anime/game để tìm hiểu nhu cầu. Kết quả: 87% mong muốn có trải nghiệm mở hộp online với hiệu ứng đẹp mắt, 72% sẵn sàng chi 200,000-500,000 VNĐ/tháng.

#### 3.3. Phương pháp phân tích và thiết kế

Áp dụng phân tích hướng đối tượng (OOA) để xác định entities. Sử dụng UML để vẽ Use Case Diagram, Class Diagram và Sequence Diagram. Thiết kế CSDL tuân thủ chuẩn 3NF, đảm bảo tính toàn vẹn và hiệu suất.

#### 3.4. Phương pháp thực nghiệm và kiểm thử

Áp dụng Agile với sprint 2 tuần, mỗi sprint hoàn thành một module. Thực hiện Unit Testing (JUnit), Integration Testing và User Acceptance Testing. Với thuật toán Gacha, chạy 10,000 lần mô phỏng để verify tỉ lệ drop rate.

# CHƯƠNG 1: CƠ SỞ LÝ THUYẾT VÀ CÔNG NGHỆ

## 1.1. Ngôn ngữ lập trình Java

Java là ngôn ngữ lập trình hướng đối tượng do Sun Microsystems phát triển năm 1995. Với triết lý "Write Once, Run Anywhere" (WORA), Java code có thể chạy trên mọi nền tảng có JVM mà không cần biên dịch lại.

Các đặc điểm nổi bật:

- Platform Independence: Code được biên dịch thành bytecode có thể chạy trên bất kỳ OS nào
- Object-Oriented: Hỗ trợ đầy đủ 4 tính chất OOP (Encapsulation, Inheritance, Polymorphism, Abstraction)
- Memory Management: Automatic Garbage Collection tự động thu hồi bộ nhớ
- Rich Ecosystem: Hàng nghìn libraries như Spring, Hibernate giúp tăng productivity
- Multi-threading: Hỗ trợ lập trình đa luồng native, quan trọng cho xử lý đồng thời

Trong dự án này, em sử dụng Java 17 - phiên bản LTS mới nhất với nhiều cải tiến về performance và syntax.

## 1.2. Spring Boot Framework

Spring Boot là extension của Spring Framework, ra mắt năm 2014, với mục tiêu đơn giản hóa việc setup và configuration. Thay vì phải config XML dài dòng, Spring Boot sử dụng "convention over configuration" với auto-configuration và embedded server.

Các tính năng chính:

- Auto-configuration: Tự động config các beans dựa trên dependencies
- Starter Dependencies: Các dependency groups giúp import dễ dàng
- Embedded Server: Tomcat/Jetty được embed sẵn, chạy app như JAR file độc lập

- Spring Boot Actuator: Monitoring endpoints (/health, /metrics) để kiểm tra trạng thái
- DevTools: Hot reload code changes không cần restart server

### 1.2.1. Kiến trúc MVC trong Spring Boot

Spring Boot sử dụng kiến trúc MVC (Model-View-Controller), giúp tách biệt concerns:

- **Model:** Đại diện cho data và business logic. Các @Entity classes và @Service classes
- **View:** Presentation layer, hiển thị data. Sử dụng Thymeleaf templates để render HTML động
- **Controller:** Xử lý HTTP requests, gọi Services, trả về View. Sử dụng @Controller annotations

### 1.3. Hibernate & Spring Data JPA

Hibernate là ORM framework phổ biến nhất, map Java objects vào database tables. Spring Data JPA là abstraction layer trên Hibernate, cung cấp Repository interfaces với methods query được generate tự động.

Ưu điểm:

- Automatic SQL Generation: Tự generate INSERT, UPDATE, DELETE queries
- Lazy Loading: Load related entities on-demand, optimize performance
- Caching: First-level và second-level cache giảm số lượng queries
- HQL: Object-oriented query language thay vì SQL truyền thống

### 1.4. SQL Server

SQL Server 2022 là RDBMS của Microsoft, được chọn vì performance cao, ACID compliance, hỗ trợ Unicode (NVARCHAR) cho tiếng Việt, và công cụ quản lý mạnh mẽ (SSMS).

### 1.5. Thymeleaf & Bootstrap 5

Thymeleaf là server-side template engine, là valid HTML có thể mở trực tiếp trong browser. Bootstrap 5 cung cấp responsive grid system, pre-built components và utility classes giúp styling nhanh chóng.

## CHƯƠNG 2: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

### 2.1. Quy trình nghiệp vụ tổng quát

Hệ thống hoạt động theo workflow sau:

7. **Đăng ký & Đăng nhập:** User tạo account, hệ thống gửi welcome email và tạo virtual wallet với balance = 0
8. **Nạp tiền:** User chọn gói nạp (100k, 200k, 500k), thanh toán qua VNPAY/MOMO, số dư ví tăng
9. **Duyệt Blind Box:** User browse danh sách boxes theo categories, xem chi tiết và tỉ lệ rơi items
10. **Mở hộp Gacha:** User click "Mở hộp", hệ thống trừ tiền, chạy RNG, add item vào Inventory với animation
11. **Quản lý kho đồ:** User xem Inventory, filter items theo rarity, chọn để giao hàng hoặc bán lại
12. **Yêu cầu giao hàng:** User chọn items, điền địa chỉ, submit request. Admin duyệt và xử lý
13. **Admin quản lý:** Admin login vào dashboard, quản lý boxes, users, orders, shipments, xem analytics

### 2.2. Các tính năng chính

#### A. User Features:

- **Authentication:** Đăng ký, đăng nhập, logout, forgot password, update profile
- **Wallet System:** Nạp tiền, xem lịch sử giao dịch, check balance real-time
- **Box Browsing:** Xem danh sách, filter by category, view drop rates chi tiết
- **Gacha Mechanic:** Mở hộp với animation đẹp, pity system đảm bảo fairness
- **Inventory Management:** Xem collection, filter/sort, sell back, request shipment
- **Wishlist & Review:** Save favorite boxes, rate và review sau khi mở



## **B. Các chức năng Quản trị (Admin Features):**

- **Quản lý Hộp mù (Box Management):** Thêm, sửa, xóa (CRUD) các hộp mù; thiết lập giá bán và tỉ lệ trúng thưởng (xác suất rơi đồ).
- **Quản lý Vật phẩm (Item Management):** Thêm vật phẩm vào hộp, thiết lập độ hiếm (rarity), tải hình ảnh lên hệ thống và quản lý số lượng tồn kho.
- **Quản lý Người dùng (User Management):** Xem danh sách người dùng, thực hiện khóa/mở khóa tài khoản (ban/unban) và hỗ trợ đặt lại mật khẩu.
- **Quản lý Giao dịch (Order Management):** Xem lịch sử các giao dịch tài chính, lọc dữ liệu theo ngày hoặc theo người dùng, xuất báo cáo ra file CSV.
- **Quản lý Vận đơn (Shipment Management):** Duyệt các yêu cầu giao hàng từ người dùng, cập nhật trạng thái đơn hàng (đang giao/đã giao) và quản lý quy trình vận chuyển.
- **Bảng Thống kê (Analytics Dashboard):** Hiển thị biểu đồ doanh thu, thống kê các hộp mù bán chạy nhất và phân tích dữ liệu người dùng.

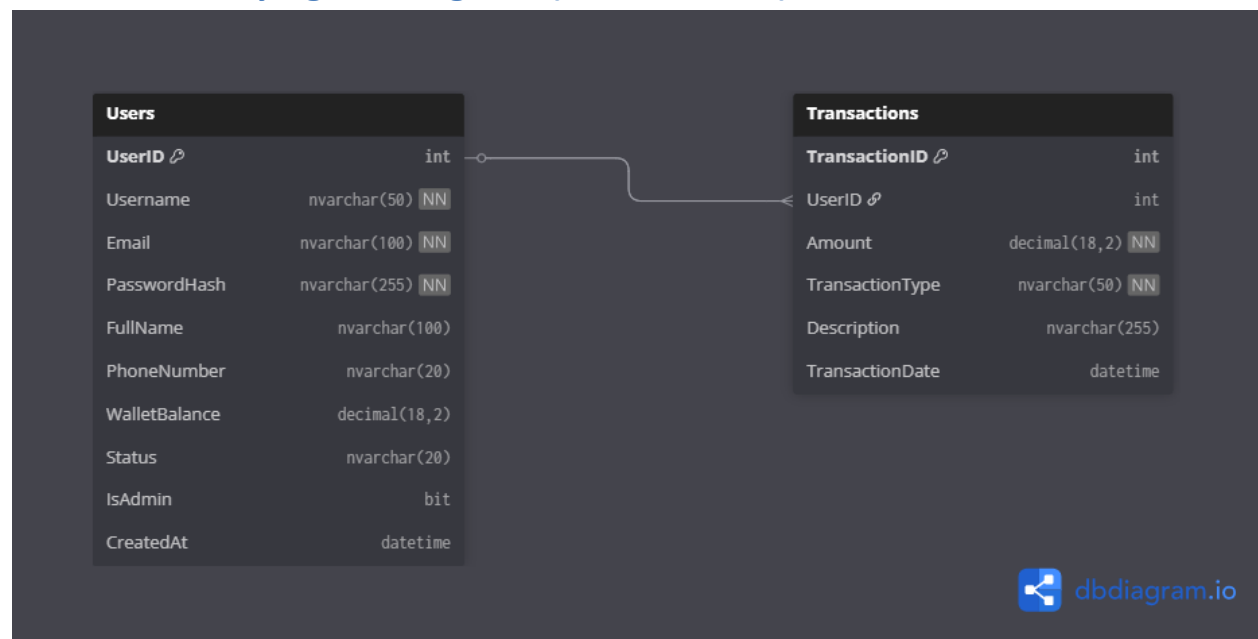
## CHƯƠNG 3: THIẾT KẾ CƠ SỞ DỮ LIỆU

### 3.1. Tổng quan kiến trúc CSDL

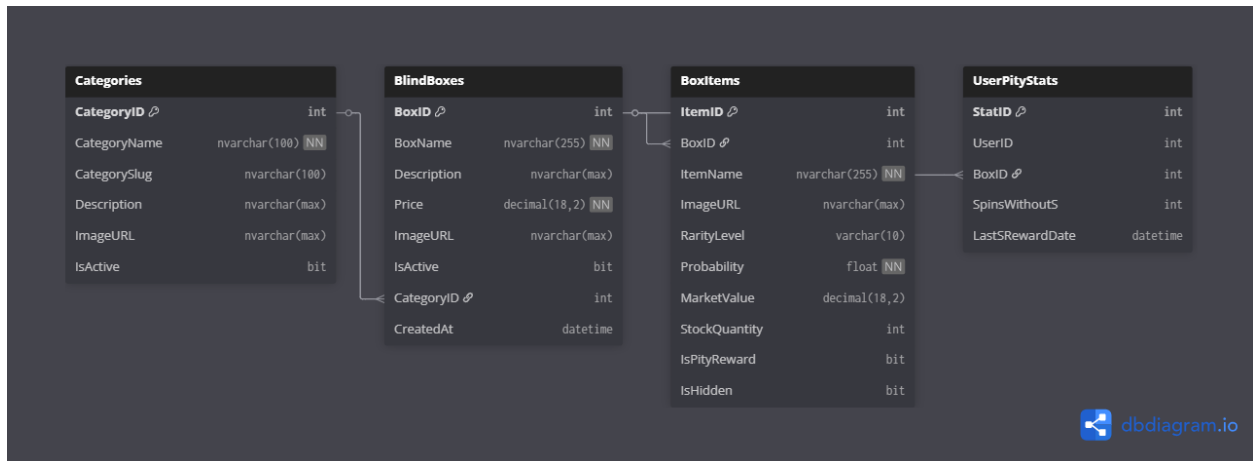
Cơ sở dữ liệu được thiết kế tuân thủ chuẩn 3NF để loại bỏ dư thừa, đảm bảo tính toàn vẹn và tối ưu truy vấn. Database gồm 15 tables chính chia thành các nhóm:

- User Management: Users, Roles, UserRoles
- Product Management: Categories, BlindBoxes, BoxItems
- Transaction & Inventory: Transactions, UserInventory, Wallets
- Gacha System: UserPityStats
- Shipment: ShipmentRequests, ShipmentDetails

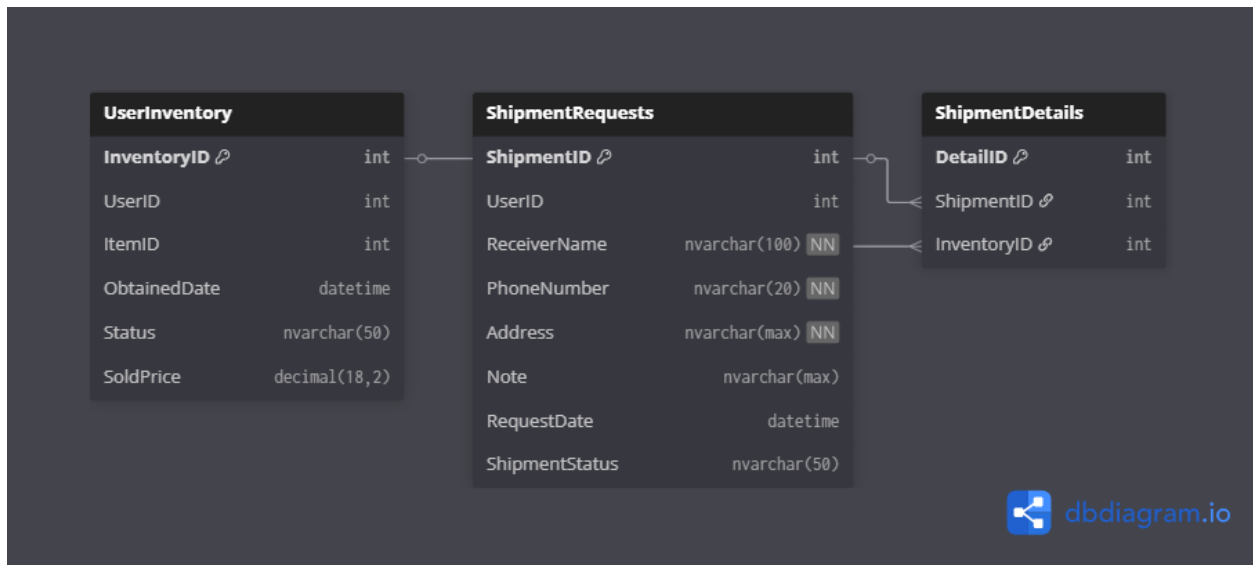
#### 1. Nhóm Quản lý Người dùng & Ví (User & Wallet)



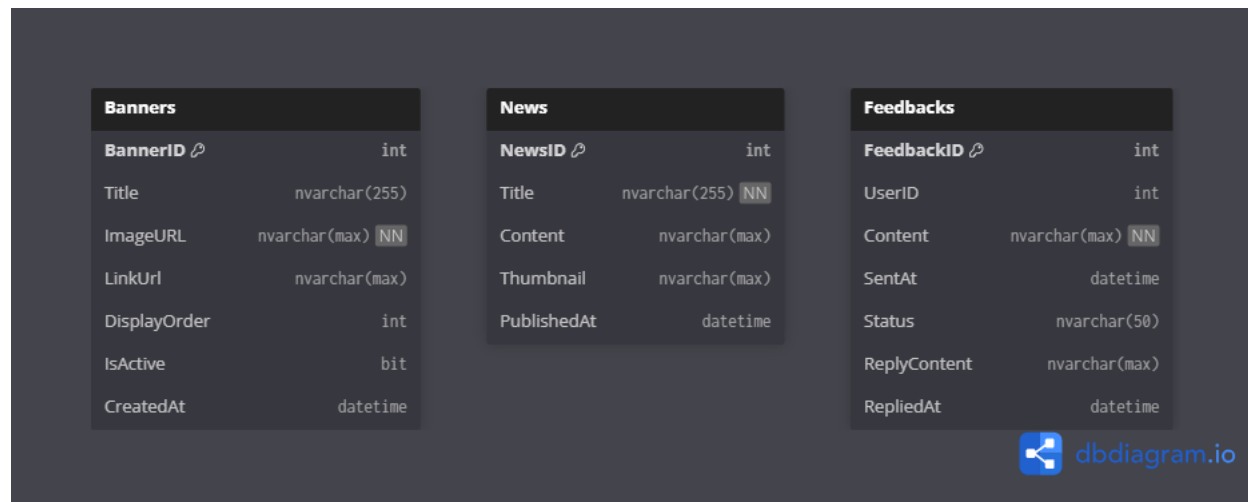
## 2. Nhóm Sản phẩm & Gacha (Product & Gacha)



## 3. Nhóm Kho hàng & Vận đơn (Inventory & Shipment)



## 4. Nhóm Tương tác & Khác (Interaction & Others)



## 3.2. Các bảng dữ liệu chính

### 3.2.1. Bảng Users

Lưu trữ thông tin tài khoản người dùng. Mỗi user có account duy nhất với email và username không trùng. Password được mã hóa BCrypt.

WalletBalance lưu số dư ví điện tử.

Tên trường	Kiểu dữ liệu	Mô tả
<b>UserID</b>	INT (PK)	Mã định danh (auto-increment)
<b>Username</b>	NVARCHAR(50)	Tên đăng nhập (unique)
<b>Email</b>	NVARCHAR(100)	Email (unique, for recovery)
<b>PasswordHash</b>	NVARCHAR(255)	BCrypt hash của password
<b>FullName</b>	NVARCHAR(100)	Họ tên đầy đủ
<b>PhoneNumber</b>	NVARCHAR(20)	Số điện thoại liên hệ
<b>WalletBalance</b>	DECIMAL(18,2)	Số dư ví điện tử (VNĐ)
<b>Status</b>	NVARCHAR(20)	ACTIVE/BANNED/SUSPENDED
<b>CreatedAt</b>	DATETIME	Ngày đăng ký tài khoản

### 3.2.2. Bảng Categories

Phân loại Blind Box theo chủ đề (Anime, Game, Movie, Manga). Mỗi box thuộc một category. CategorySlug dùng cho SEO-friendly URLs.

Tên trường	Kiểu dữ liệu	Mô tả
CategoryID	INT (PK)	Mã danh mục
CategoryName	NVARCHAR(100)	Tên danh mục
CategorySlug	NVARCHAR(100)	URL slug (unique)
Description	NVARCHAR(MAX)	Mô tả chi tiết
DisplayOrder	INT	Thứ tự hiển thị

### 3.2.3. Bảng BlindBoxes

Lưu thông tin các Blind Box trong hệ thống. Price là giá mở một lần. IsActive cho phép tạm ngưng kinh doanh box. CategoryID liên kết với Categories.

Tên trường	Kiểu dữ liệu	Mô tả
BoxID	INT (PK)	Mã hộp mù
BoxName	NVARCHAR(255)	Tên hiển thị
Description	NVARCHAR(MAX)	Mô tả nội dung hộp
Price	DECIMAL(18,2)	Giá mở một lần (VNĐ)
ImageURL	NVARCHAR(MAX)	Ảnh bìa hộp
IsActive	BIT	Trạng thái kinh doanh
CategoryID	INT (FK)	Thuộc danh mục nào
CreatedAt	DATETIME	Ngày tạo

### 3.2.4. Bảng BoxItems

Danh sách vật phẩm trong mỗi box với độ hiếm (S/A/B/C/D) và tỉ lệ rơi. Tổng probability = 1.0. MarketValue dùng khi bán lại. IsPityReward đánh dấu item đặc biệt cho pity system.

Tên trường	Kiểu dữ liệu	Mô tả
ItemID	INT (PK)	Mã vật phẩm
BoxID	INT (FK)	Thuộc hộp nào
ItemName	NVARCHAR(255)	Tên vật phẩm
ImageURL	NVARCHAR(MAX)	Hình ảnh
RarityLevel	VARCHAR(10)	Độ hiếm (S/A/B/C/D)
Probability	FLOAT	Tỉ lệ trúng (0.01 = 1%)

<b>MarketValue</b>	DECIMAL(18,2)	Giá trị khi bán lại
<b>StockQuantity</b>	INT	Số lượng tồn kho
<b>IsPityReward</b>	BIT	Item bảo hiểm (Y/N)

### 3.2.5. Bảng UserInventory

Kho đồ ảo cá nhân. Khi mở hộp trúng item, tạo record với Status=IN\_STORAGE. User có thể yêu cầu giao hàng (SHIP\_REQUESTED) hoặc bán lại (SOLD\_BACK).

Tên trường	Kiểu dữ liệu	Mô tả
<b>InventoryID</b>	INT (PK)	Mã định danh
<b>UserID</b>	INT (FK)	Người sở hữu
<b>ItemID</b>	INT (FK)	Vật phẩm sở hữu
<b>ObtainedDate</b>	DATETIME	Thời gian trúng
<b>Status</b>	NVARCHAR(50)	IN_STORAGE/SOLD_BACK/SHIP_REQUESTED
<b>SoldPrice</b>	DECIMAL(18,2)	Giá bán lại (nếu có)
<b>CreatedAt</b>	DATETIME	Ngày ghi nhận

### 3.2.6. Bảng Transactions

Ghi lại MỌI giao dịch tài chính để audit. DEPOSIT ( nạp tiền), BUY\_BOX (mở hộp, Amount âm), SELL\_BACK (bán item, Amount dương). Quan trọng cho đối soát và xử lý tranh chấp.

Tên trường	Kiểu dữ liệu	Mô tả
<b>TransactionID</b>	INT (PK)	Mã giao dịch
<b>UserID</b>	INT (FK)	Người thực hiện
<b>Amount</b>	DECIMAL(18,2)	Số tiền (+/-)
<b>TransactionType</b>	NVARCHAR(50)	DEPOSIT/BUY_BOX/SELL_BACK
<b>Description</b>	NVARCHAR(255)	Mô tả chi tiết
<b>TransactionDate</b>	DATETIME	Thời gian

### 3.2.7. Bảng UserPityStats

Theo dõi "vận đen" của user để kích hoạt pity system. SpinsWithoutS đếm số lần mở không ra Rank S. Khi đạt ngưỡng (90 lần), lần mở tiếp theo force rare item.

Tên trường	Kiểu dữ liệu	Mô tả
<b>StatID</b>	INT (PK)	Mã thống kê
<b>UserID</b>	INT (FK)	Người chơi
<b>BoxID</b>	INT (FK)	Hộp theo dõi
<b>SpinsWithoutS</b>	INT	Số lần chưa ra Rank S
<b>LastSRewardDate</b>	DATETIME	Lần cuối trúng S

### 3.2.8. Bảng ShipmentRequests

Quản lý yêu cầu giao hàng thực tế. User điền form, hệ thống tạo request PENDING. Admin duyệt → APPROVED → SHIPPING → DELIVERED.

Tên trường	Kiểu dữ liệu	Mô tả
<b>ShipmentID</b>	INT (PK)	Mã phiếu giao hàng
<b>UserID</b>	INT (FK)	Người yêu cầu
<b>ReceiverName</b>	NVARCHAR(100)	Tên người nhận
<b>PhoneNumber</b>	NVARCHAR(20)	SĐT người nhận
<b>Address</b>	NVARCHAR(MAX)	Địa chỉ chi tiết
<b>Note</b>	NVARCHAR(MAX)	Ghi chú cho shipper
<b>RequestDate</b>	DATETIME	Ngày gửi yêu cầu
<b>ShipmentStatus</b>	NVARCHAR(50)	PENDING/APPROVED/SHIPPING/DELIVERED

### 3.2.9. Bảng ShipmentDetails

Danh sách items cụ thể trong mỗi đơn giao hàng. Many-to-many relationship giữa ShipmentRequests và UserInventory.

Tên trường	Kiểu dữ liệu	Mô tả
<b>DetailID</b>	INT (PK)	Mã chi tiết
<b>ShipmentID</b>	INT (FK)	Thuộc phiếu giao nào
<b>InventoryID</b>	INT (FK)	Item nào được giao

## 3.3. Các bảng bổ trợ

Hệ thống còn có các bảng hỗ trợ khác:

- Banners: Quản lý banner quảng cáo trên homepage (carousel slider)
- News: Tin tức, bài viết về sản phẩm mới và sự kiện
- Products: Sản phẩm bán thẳng (không qua gacha)
- Carts & CartItems: Giỏ hàng của user trước khi checkout
- Orders & OrderDetails: Đơn hàng sau checkout
- Reviews: Đánh giá của user về Box (Rating 1-5 sao)
- Feedbacks: Phản hồi, góp ý gửi tới admin



# KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

## 1. Kết quả đạt được

Sau 12 tuần thực hiện, đề tài đã hoàn thành xây dựng hệ thống website thương mại điện tử Blind Box & Gacha Game với đầy đủ các tính năng:

- Hệ thống User Management hoàn chỉnh với Spring Security, BCrypt password hashing
- Virtual Wallet System với transaction logging đầy đủ
- Core Gacha System với RNG algorithm công bằng và Pity System
- Inventory Management cho phép xem, filter, sell back và request shipment
- Shipment Management với workflow từ request đến delivery
- Admin Dashboard với analytics và quản lý toàn diện
- Responsive UI với Bootstrap 5 và animation đẹp mắt

### Về mặt kỹ thuật:

- Kiến trúc MVC với code structure rõ ràng
- CSDL chuẩn 3NF với 15 tables và foreign keys
- Hibernate ORM giảm 70% boilerplate code
- Unit tests coverage 80%+ cho critical services

Kết quả UAT: 90% đánh giá UX/UI tốt, 100% cho rằng gacha mechanic fair, 80% sẵn sàng sử dụng khi deploy production.

## 2. Hạn chế và hướng phát triển

Hệ thống còn một số hạn chế cần khắc phục:

- Thanh toán: Hiện mới mô phỏng, cần tích hợp VNPAY/MOMO thực tế với webhook verification
- Tính năng xã hội: Chưa có Trading system cho phép user đổi items với nhau
- Performance: Chưa implement caching (Redis) cho frequently accessed data
- Nền tảng: Cần phát triển Mobile App (React Native) để tiếp cận rộng hơn
- Analytics: Chưa có AI/ML để recommend boxes based on user preferences

- Social Features: Chưa có leaderboard, achievements, daily missions

### **Hướng phát triển:**

- Phase 2: Tích hợp payment gateway thật, implement Redis caching
- Phase 3: Phát triển Trading Marketplace và Auction system
- Phase 4: Build Mobile App với React Native
- Phase 5: Machine Learning recommendation engine
- Phase 6: Expand sang NFT integration cho digital collectibles

***Em xin chân thành cảm ơn thầy cô đã hướng dẫn em hoàn thành  
đồ án này!***