

TÀI LIỆU THIẾT KẾ HƯỚNG ĐÓI TUỢNG (MÔ HÌNH LỚP)

Nhóm 02 - Thành viên nhóm:

1. Nguyễn Thanh Cảnh (Nhóm trưởng)
2. Lê Hoàng Khang
3. Trần Đình Kiên
4. Hồ Thị Như Tâm

Tên ứng dụng: OmegaBook - Ứng dụng quản lý nhà sách

Thời gian thực hiện: Từ 21/08/2023 đến 19/11/2023 (13 tuần)

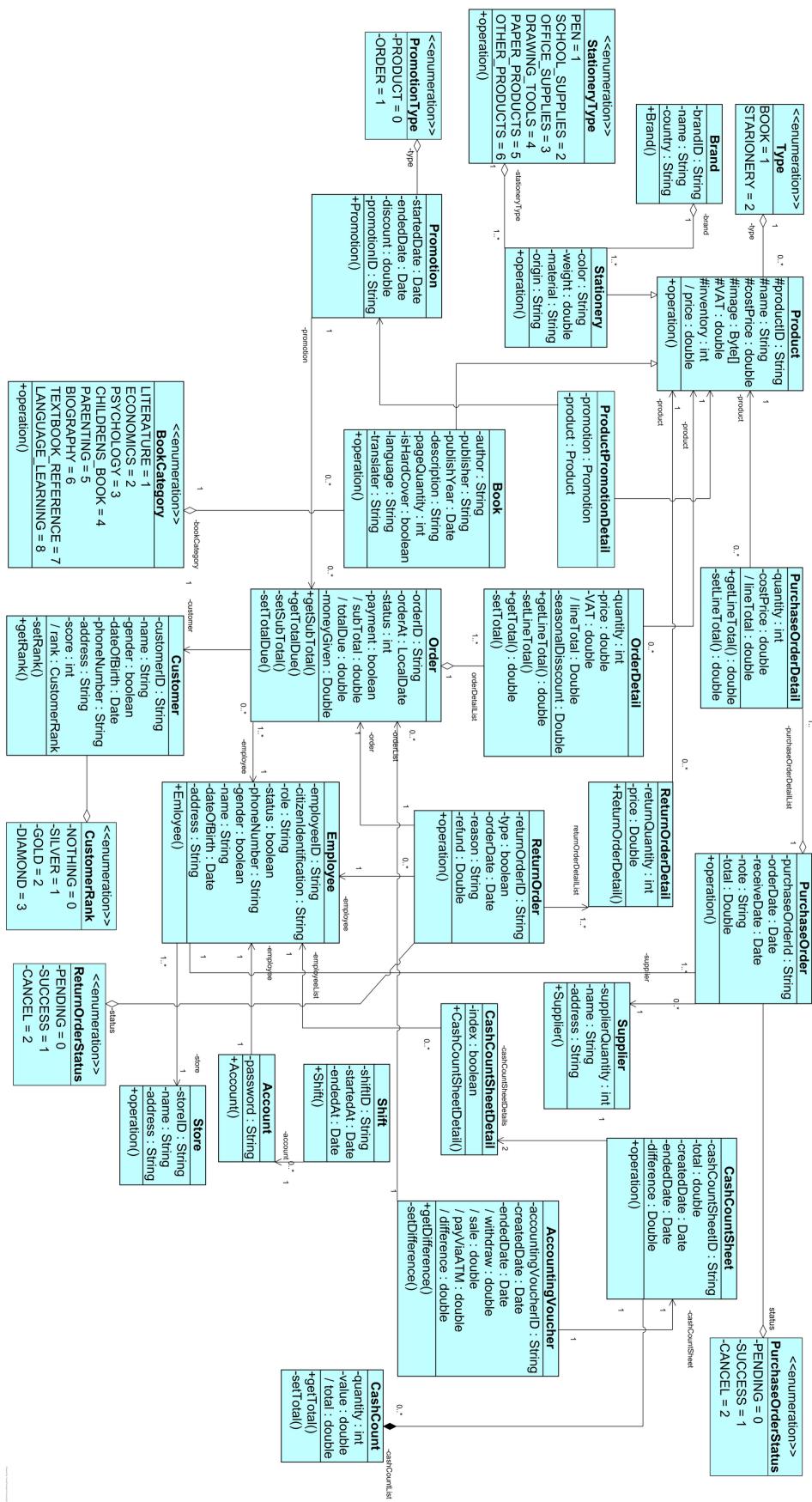
Mục lục

1. Mô hình lớp (Class Diagram).....	4
2. Đặc tả sơ đồ lớp:.....	6
2.1. Thực thể Account (Tài khoản)	6
2.2. Thực thể AccountingVoucher (Phiếu kết toán)	7
2.3. Thực thể AccountingVoucherDetail (Chi tiết phiếu kết toán)	Error! Bookmark not defined.
2.4. Thực thể Address (Địa chỉ)	Error! Bookmark not defined.
2.5. Thực thể Book (Sách).....	10
2.6. Thực thể Brand (Thương hiệu).....	12
2.7. Thực thể CashCount (Chi tiết kiểm tiền)	14
2.8. Thực thể CashCountSheet (Phiếu kiểm tiền)	16
2.9. Thực thể CashCountSheetDetail (Chi tiết phiếu kiểm tiền).....	18
2.10. Thực thể Customer (Khách hàng)	20
2.11. Thực thể Employee (Nhân viên)	23
2.12. Thực thể ForeingBook (Sách ngoại văn)	28
2.13. Thực thể Order (Hóa đơn).....	29
2.14. Thực thể OrderDetail (Chi tiết hóa đơn)	32
2.15. Thực thể Product (Sản phẩm).....	35
2.16. Thực thể Promotion (Chương trình khuyến mãi).....	37
2.17. Thực thể PuchaseOrder (Đơn nhập hàng).....	38
2.18. Thực thể PuchaseOrderDetail (Chi tiết đơn nhập hàng)	42
2.19. Thực thể ReturnOrder (Đơn đổi trả hàng).....	43
2.20. Thực thể Shift (Phiên đăng nhập).....	46
2.21. Thực thể Stationery (Văn phòng phẩm).....	48
2.22. Thực thể Store (Cửa hàng)	51
2.23. Thực thể Supplier (Nhà cung cấp)	54
2.24. Thực thể VietnameseBook (Sách tiếng việt).....	55

DANH MỤC BẢNG

Bảng 2.1 Đặc tả lớp Account - Tài khoản.....	6
Bảng 2.2 Đặc tả lớp AccoutingVoucher - Phiếu kết toán	7
Bảng 2.3 Đặc tả lớp AcoutingVoucherDetail - Chi tiết phiếu kết toán.....	Error! Bookmark not defined.
Bảng 2.4 Đặc tả lớp Address - Địa chỉ.....	Error! Bookmark not defined.
Bảng 2.5 Đặc tả lớp Book - Sách.....	10
Bảng 2.6 Đặc tả lớp Brand - Thương hiệu	12
Bảng 2.7 Đặc tả lớp CashCount - Chi tiết kiểm tiền.....	14
Bảng 2.8 Đặc tả lớp CashCountSheet - Phiếu kiểm tiền	16
Bảng 2.9 Đặc tả lớp CashCountSheetDetail - Chi tiết phiếu kiểm tiền.....	18
Bảng 2.10 Đặc tả lớp Customer - Khách hàng.....	20
Bảng 2.11 Đặc tả lớp Employee - Nhân viên.....	23
Bảng 2.12 Đặc tả lớp ForeingBook - Sách ngoại văn	28
Bảng 2.13 Đặc tả lớp Order - Hóa đơn	29
Bảng 2.14 Đặc tả lớp OrderDetail.....	32
Bảng 2.15 Đặc tả lớp Product - Sản phẩm	35
Bảng 2.16 Đặc tả lớp Promotion - Chương trình khuyến mãi.....	37
Bảng 2.17 Đặc tả lớp PurchaseOrder - Đơn nhập hàng	38
Bảng 2.18 Đặc tả lớp PurchaseOrderDetail - Chi tiết đơn nhập hàng.....	42
Bảng 2.19 Đặc tả lớp ReturnOrder - Đơn đổi trả hàng	43
Bảng 2.20 Đặc tả lớp Shift - Phiên đăng nhập	46
Bảng 2.21 Đặc tả lớp Stationery - Văn phòng phẩm.....	48
Bảng 2.22 Đặc tả lớp Store - Cửa hàng.....	51
Bảng 2.23 Đặc tả lớp Supplier - Nhà cung cấp	54
Bảng 2.24 Đặc tả lớp VietnameseBook - Sách tiếng việt	55

1. Mô hình lớp (Class Diagram)



2. Đặc tả sơ đồ lớp:

2.1. Thực thể Account (Tài khoản)

Bảng 2.1 Đặc tả lớp Account - Tài khoản

STT	Công việc	Kiểu dữ liệu	Ý nghĩa	Ràng buộc	Ghi chú
1	Khai báo thuộc tính:				
1.1	password	String	Mật khẩu tài khoản	Từ 8 ký tự trên, ký tự đầu là chữ in hoa, có ít nhất 1 trong các ký tự “.,@&*^”, có ít nhất 1 chữ số	
1.2	employee	Employee	Nhân viên ứng với tài khoản		
2	Viết các phương thức getter, setter				
2.1	setPassword(password)	void		Từ 8 ký tự trên, ký tự đầu là chữ in hoa, có ít nhất 1 trong các ký tự “.,@&*^”, có ít nhất 1 chữ số	throw exception “password không hợp lệ” nếu tham số password không hợp lệ
2.2	setEmployee(employee)	void		Không được rỗng	throw exception “employee không được rỗng” nếu

					tham số employee không hợp lệ
3	Viết các constructor				
3.1	Password(password, employee)			Hàm khởi tạo đầy đủ tham số	
3.2	Password()			Hàm khởi tạo mặc định	
4	Viết phương thức toString()	string			trả về chuỗi chứa giá trị các thuộc tính của tài khoản

2.2. Thực thể AccountingVoucher (Phiếu kết toán)

Bảng 2.2 Đặc tả lớp AccountingVoucher - Phiếu kết toán

ST T	Công việc	Kiểu dữ liệu	Ý nghĩa	Ràng buộc	Ghi chú
1	Khai báo thuộc tính:				
1.1	accountingVoucherID	String	Mã phiếu kết toán	Bao gồm 15 ký tự có dạng KTOhhppddmmyyyxx. Trong đó: - hh là giờ lập phiếu - pp: là ngày lập phiếu - dd: là ngày lập phiếu - mm: là tháng lập phiếu - yyyy: là năm lập phiếu	{id}

				- xx: là hai số nguyên dương được hệ thống phát sinh tăng dần.	
1.2	createDate	DateTime	Thời gian bắt đầu bán hàng	Thời gian lập phiếu kết toán gần nhất trước đó.	
1.3	endDate	DateTime	Thời gian kết thúc bán hàng	Thời điểm hiện tại	
1.4	cashCountSheet	CashCountSheet	Phiếu kiểm tiền gần nhất và ở trạng thái “Chờ kết toán”		
1.6	sale	double	Tổng tiền bán hàng	Lớn hơn hoặc bằng 0	(*)
1.7	payViaATM	double	Tổng tiền được thanh toán qua ATM	Lớn hơn hoặc bằng 0	(*)

1.8	withDraw	double	Số tiền lấy ra khỏi két bán hàng	Lớn hơn hoặc bằng 0	(*)
1.9	difference	double	Tiền chênh lệch		(*)
2	Viết các phương thức getter, setter				
2.1	setTotal	void			throw exception “password không hợp lệ” nếu tham số password không hợp lệ
3	Viết các constructor				
3.1	AcountingVoucher(accountingVoucherID, createDate, endDate, cashCountSheet, accountingVoucherDetails, sale, payViaATM, withDraw, difference)			Hàm khởi tạo đầy đủ tham số	
3.2	AcountingVoucher ()			Hàm khởi tạo mặc định	

4	Viết phương thức toString()	string			trả về chuỗi chứa giá trị các thuộc tính của phiếu kiểm tiền
---	--	---------------	--	--	---

Ghi chú:

- sale: là doanh thu bán hàng. Được tính từ tổng các hóa đơn ở trạng thái đã thanh toán từ thời điểm lập phiếu kết toán gần nhất trước đó đến thời điểm hiện tại.
- payViaATM: là tổng tiền đã được thanh toán qua ATM. Được tính từ tổng các hóa đơn ở trạng thái đã thanh toán từ thời điểm lập phiếu kết toán gần nhất trước đó đến thời điểm hiện tại.
- withdraw = sale - payViaATM (Số tiền mặt cần lấy ra khỏi két)
- difference = cashCountSheet.total - withdraw - 1.765.000 (Chênh lệch so với tiền dự phòng sau khi lấy tiền ra khỏi két)

2.3. Thực thể Book (Sách)

Bảng 2.3 Đặc tả lớp Book - Sách

STT	Công việc	Kiểu dữ liệu	Ý nghĩa	Ràng buộc	Ghi chú
1	Khai báo thuộc tính:				
1.1	author	String	Tên tác giả	Không rỗng	
1.2	publisher	String	Tên nhà xuất bản	Không rỗng	
1.3	publishYear	Integer	Năm phát hành	1900 < publishYear <= Năm hiện tại	
1.4	description	String	Mô tả		
1.5	pageQuantity	Number	Số trang	Lớn hơn 0	
1.6	isHardCover	Boolean	Có phải bìa cứng không		

1.7	bookCategory	BookCategory	Thể loại sách	<<Enum>>	(*)
2	Viết các phương thức getter, setter				
2.1	setAuthor(String author)	void		Kiểm tra tham số author không được rỗng	throw exception “Author can't empty” nếu tham số author không hợp lệ
2.2	setPublisher(String publisher)	void		Kiểm tra tham số publisher không được rỗng.	throw exception “Publisher can't empty” nếu tham số publisher không hợp lệ
2.3	setPublishYear(Integer year)	void		Kiểm tra tham số year lớn hơn 1900 và \leq năm hiện tại.	Throw exception “publishYear must > 1900 && \leq current Year” nếu tham số year không hợp lệ
2.5	setPageQuantity(Integer pageQuantity)	Void		Kiểm tra tham số pageQuantity > 0	Throw exception “pageQuantity must > 0” nếu tham số pageQuantity không hợp lệ

3	Viết các constructor				
	Book()				Mặc định
	Book(String id)				Chỉ chứa mã
	Book(String productId, String name, Double costPrice, Double price, String author, String publisher, Integer publishYear, String description, Integer PageQuantity, boolean isHardCover)				Đầy đủ tham số
4	Viết phương thức toString()	string			

Ghi chú: Enum “BookCategory”

- LITERATURE: Sách văn học
- ECONOMICS: Sách kinh tế
- PSYCHOLOGY: Sách Tâm lý – Kỹ năng sống
- CHILDRENS_BOOK: Sách thiếu nhi
- PARENTING: Sách nuôi dạy con
- BIOGRAPHY: Sách tiểu sử, hồi ký
- TEXTBOOK_REFERENCE: Sách giáo khoa – Tham khảo
- LANGUAGE_LEARNING: Sách học ngoại ngữ

2.4. Thực thể Brand (Thương hiệu)

Bảng 2.4 Đặc tả lớp Brand - Thương hiệu

STT	Công việc	Kiểu dữ liệu	Ý nghĩa	Ràng buộc	Ghi chú

1	Khai báo thuộc tính:				
1.1	name	String	Tên thương hiệu	Không rỗng	
1.2	brandID	String	Mã thương hiệu	Mã thương hiệu có dạng THxxx Trong đó:xxx là 3 số nguyên dương phát sinh	{id}
1.3	country	String	Quốc gia của thương hiệu	Không rỗng	
2	Viết các phương thức getter, setter				
2.1	setName(String name)	void		Kiểm tra tham số name không được rỗng	throw exception “name can’t empty” nếu tham số name không hợp lệ
2.2	setBrandID(String id)	void		Kiểm tra tham số id không được rỗng	throw exception “id can’t empty” nếu tham số id không hợp lệ

2.3	setCountry(String country)	void		Kiểm tra tham số country không rỗng	Throw exception “country can’t empty” nếu tham số country không hợp lệ
3	Viết các constructor				
	Brand ()				Mặc định
	Brand (String id)				Chỉ chứa mã
	Brand (String brandID, String name, String country)				Đầy đủ tham số
4	Viết phương thức toString()	string			

2.5. Thực thể CashCount (Chi tiết kiểm tiền)

Bảng 2.5 Đặc tả lớp CashCount - Chi tiết kiểm tiền

STT	Công việc	Kiểu dữ liệu	Ý nghĩa	Ràng buộc	Ghi chú
1	Khai báo thuộc tính:				
1.1	quantity	int	Số lượng tờ tiền	Số nguyên dương	
1.2	value	Double	Mệnh giá của tiền	Số thực dương	
1.3	total	double	Tổng tiền của mệnh giá	Số thực dương	
2	Viết các phương thức getter, setter				

2.1	setQuantity(quantity)	void		Kiểm tra số truyền vào phải là số nguyên dương	throw exception “password không hợp lệ” nếu tham số password không hợp lệ
2.2	setValue	void		Kiểm tra mệnh giá phải là số dương	throw exception “employee không được rỗng” nếu tham số employee không hợp lệ
3	Viết các constructor				
3.1	CashCount(quantity, value)			Hàm khởi tạo đầy đủ tham số	
3.2	CashCount()			Hàm khởi tạo mặc định	
4	Viết phương thức toString()	string			trả về chuỗi chứa giá trị các thuộc tính của CashCount

Ghi chú: total = quantity * value

2.6. Thực thể CashCountSheet (Phiếu kiểm tiền)

Bảng 2.6 Đặc tả lớp CashCountSheet - Phiếu kiểm tiền

STT	Công việc	Kiểu dữ liệu	Ý nghĩa	Ràng buộc	Ghi chú
1	Khai báo thuộc tính:				
1.1	cashCountSheetID	String	Mã phiếu kiểm tiền	Mã phiếu kiểm tiền có dạng KTIhhppddmmyyyyxx. Trong đó: - hh là giờ lập phiếu - pp: là ngày lập phiếu - dd: là ngày lập phiếu - mm: là tháng lập phiếu - yyyy: là năm lập phiếu - xx: là hai số nguyên dương được hệ thống phát sinh tăng dần.	
1.1	total	Double	Tổng tiền kiểm được	Số thực dương	
1.2	createDate	DateTime	Thời gian khởi tạo kiểm tiền	< Thời điểm hiện tại	
1.3	endDate	DateTime	Thời gian kết thúc kiểm tiền	> createDate	
1.4	cashCounts	CashCount	Danh sách các tờ tiền		

			ứng với số lượng		
1.5	cashCountSheetDetails		Thông tin về các nhân viên kiểm tiền ứng với phiếu kiểm tiền		
1.6	difference	double	Chênh lệch so với tiền dự phòng		
2	Viết các phương thức getter, setter				
2.1	setTotal	void			throw exception “password không hợp lệ” nếu tham số password không hợp lệ
3	Viết các constructor				
3.1	CashCount(quantity, value)			Hàm khởi tạo đầy đủ tham số	
3.2	CashCount()			Hàm khởi tạo mặc định	
4	Viết phương thức toString()	string			trả về chuỗi chứa

					giá trị các thuộc tính của phiếu kiểm tiền
--	--	--	--	--	--

Ghi chú: total = Tổng các cashCount

2.7. Thực thể CashCountSheetDetail (Chi tiết phiếu kiểm tiền)

Bảng 2.7 Đặc tả lớp CashCountSheetDetail - Chi tiết phiếu kiểm tiền

STT	Công việc	Kiểu dữ liệu	Ý nghĩa	Ràng buộc	Ghi chú
1	Khai báo thuộc tính:				
1.1					
1.2	employee	Employee	Nhân viên trong phiếu kiểm tiền		
2	Viết các phương thức getter, setter				
2.1	setEmployee	void		Kiểm tra nhân viên phải tồn tại trong hệ thống	

3	Viết các constructor				
3.1	CashCountSheetDetail(employee)			Hàm khởi tạo với nhân viên	
3.2	CashCountSheetDetail ()			Hàm khởi tạo mặc định	
4	Viết phương thức toString()	string			trả về chuỗi chứa giá trị các thuộc tính của CashCountSheetDetail

2.8. Thực thể Customer (Khách hàng)

Bảng 2.8 ĐẶC TÍNH LỚP Customer - Khách hàng

STT	Công việc	Kiểu dữ liệu	Ý nghĩa	Ràng buộc	Ghi chú
1	Khai báo thuộc tính:				
1.1	customerID	String	Mã khách hàng	Mã khách hàng bao gồm chữ số có dạng: KHaabxxxx. Trong đó: - aaaa: là năm sinh của khách hàng. - b: là giới tính của khách hàng (0 là nữ và 1 là nam) - xxxx: là 4 số nguyên	{id}
1.2	name	String	Tên khách hàng		
1.3	dateOfBirth	Datetime	Ngày sinh khách hàng		
1.4	phoneNumber	String	Số điện thoại khách hàng	Dãy số gồm 10 chữ số	
1.5	gender	Boolean	Giới tính khách hàng		True = Female, False = Male
1.6	address	String	Địa chỉ khách hàng		
1.7	Score	Int	Điểm tích thưởng		

1.8	Rank	String	Hạng khách hàng		
2	Viết các phương thức getter, setter				
2.1	setCustomerID(String customerID)	void			Tự động phát sinh
2.1	setName(String name)	void		Kiểm tra tham số name không được rỗng	throw exception “Họ tên không được rỗng” nếu tham số name không hợp lệ
2.2	setDateOfBirth(Datetime dateOfBirth)	void		Kiểm tra ngày sinh phải nhỏ hơn ngày hiện tại.	Throw exception “Ngày sinh phải nhỏ hơn ngày hiện tại” nếu tham số dateOfBirth không hợp lệ
2.3	setPhoneNumber(String phoneNumber)	void		Kiểm tra số điện thoại phải có đủ 10 chữ số và bắt đầu bằng số 0.	Throw exception “Số điện thoại không hợp lệ” nếu tham số phoneNumber không hợp lệ

2.4	setGender(Boolean gender)	void		Kiểm tra tham số gender không được rỗng	throw exception “Giới tính không không được rỗng” nếu tham số gender không hợp lệ
2.5	setAddress(Address address)	void		Kiểm tra tham số address không được rỗng	throw exception “Địa chỉ không không được rỗng” nếu tham số address không hợp lệ
2.6	setScore(int score)	void		Kiểm tra tham số score ≥ 0	throw exception “score ≥ 0 ” nếu tham số score không hợp lệ
2.7	setRank()	void		Được phân loại dựa vào giá trị của tham số score	Tự phát sinh
3	Viết các constructor				
3.1	Customer(String name, Datetime dateOfBirth, String phoneNumber, Boolean gender, Address address, String			Hàm khởi tạo đầy đủ tham số	

	customerID, int score, String rank)				
3.2	Customer()			Hàm khởi tạo mặc định	
3.3	Customer(String customerID)			Hàm khởi tạo bằng mã	
4	Viết phương thức <code>toString()</code>	string			trả về chuỗi chứa giá trị các thuộc tính của khách hàng

2.9. Thực thể Employee (Nhân viên)

Bảng 2.9 *Đặc tả lớp Employee - Nhân viên*

ST T	Công việc	Kiểu dữ liệu	Ý nghĩa	Ràng buộc	Ghi chú
1	Khai báo thuộc tính:				
1.1	employeeID	String	Mã nhân viên	Là chuỗi có dạng NVaaaaaccccx xx. Trong đó: - a: là giới tính (nam là 1 nữ là 0) - bbbb: là năm sinh - cccc: là năm vào làm	{id}

				- xxx: là 3 số nguyên dương phát sinh	
1.2	name	String	Tên nhân viên		
1.3	dateOfBirth	Datetime	Ngày sinh nhân viên		
1.4	phoneNumber	String	Số điện thoại nhân viên	Dãy số gồm 10 chữ số	
1.5	gender	Boolean	Giới tính nhân viên		True = Female, False = Male
1.6	address	String	Địa chỉ		
1.7	citizenIdentification	String	Số căn cước công dân	Dãy số gồm 12 chữ số	
1.8	role	String	Chức vụ		
1.9	status	Boolean	Trạng thái làm việc		True = Đang làm, False = Đã nghỉ
1.10	account	Account	Tài khoản		
1.11	store	Store	Cửa hàng		
2	Viết các phương thức getter, setter				
2.1	setEmployeeID(String employeeID)	void			Phát sinh tự động
2.1	setName(String name)	void		Kiểm tra tham số name không được rỗng	throw exception “Họ tên không được rỗng” nếu

					tham số name không hợp lệ
2.2	setDateOfBirth(Datet ime dateOfBirth)	void		Kiểm tra ngày sinh phải nhỏ hơn ngày hiện tại.	Throw exception “Ngày sinh phải nhỏ hơn ngày hiện tại” nếu tham số dateOfBirth không hợp lệ
2.3	setPhoneNumber(Str ing phoneNumber)	void		Kiểm tra số điện thoại phải có đủ 10 chữ số và bắt đầu bằng số 0.	Throw exception “Số điện thoại không hợp lệ” nếu tham số phoneNumber không hợp lệ
2.4	setGender(Boolean gender)	void		Kiểm tra tham số gender không được rỗng	throw exception “Giới tính không không được rỗng” nếu tham số gender không hợp lệ
2.5	setAddress(Address address)	void		Kiểm tra tham số address không được rỗng	throw exception “Địa chi không không được rỗng” nếu

					tham số address không hợp lệ
2.6	setCitizenIdentification(String citizenIdentification)	void		Kiểm tra tham số phải đủ 12 chữ số không gồm các kí tự khác	Throw exception “Số CCCD phải đủ 12 chữ số” nếu tham số citizenIdentification không hợp lệ
2.7	setRole(String role)	void		Kiểm tra tham số store không được rỗng	throw exception “Chức vụ không không được rỗng” nếu tham số role không hợp lệ
2.8	setStatus(Boolean status)	void		Kiểm tra tham số status không được rỗng	throw exception “Trạng thái làm việc không không được rỗng” nếu tham số status không hợp lệ
2.9	setAccount(Account account)	void		Kiểm tra tham số account	throw exception “Tài khoản không

				không được rỗng	không được rỗng” nếu tham số account không hợp lệ
2.10	setStore(Store store)	void		Kiểm tra tham số store không được rỗng	throw exception “Cửa hàng không được rỗng” nếu tham số store không hợp lệ
3	Viết các constructor				
3.1	Employee(String employeeID, String name, Datetime dateOfBirth, String phoneNumber, Boolean gender, Address address, String citizenIdentification, String role, String status, Account account, Store store)			Hàm khởi tạo đầy đủ tham số	
3.2	Employee()			Hàm khởi tạo mặc định	
3.3.	Employee(String employeeID)			Hàm khởi tạo bằng mã	

4	Viết phương thức toString()	string			trả về chuỗi chứa giá trị các thuộc tính của nhân viên
---	--	--------	--	--	---

2.10. Thực thể ForeingBook (Sách ngoại văn)

Bảng 2.10 Đặc tả lớp ForeignBook - Sách ngoại văn

STT	Công việc	Kiểu dữ liệu	Ý nghĩa	Ràng buộc	Ghi chú
1	Khai báo thuộc tính:				
1.1	language	String	Ngôn ngữ	Không rỗng	
1.2	translator	String	Tên dịch giả	Không rỗng	
2	Viết các phương thức getter, setter				
2.1	setLanguage(String lang)	void		Kiểm tra tham số lang không được rỗng	throw exception “Lang can't empty” nếu tham số lang không hợp lệ
2.2	setTranslator (String translator)	void		Kiểm tra tham số translator không được rỗng.	throw exception “Translator can't empty” nếu tham số translator không hợp lệ
3	Viết các constructor				
	ForeignBook()				Mặc định
	ForeignBook (String id)				Chỉ chứa mã

	ForeignBook (String productId, String name, Double costPrice, Double price, String author, String publisher, Integer publishYear, String description, Integer PageQuantity, boolean isHardCover, String language, String translator)				Đầy đủ tham số
4	Viết phương thức toString()	string			

2.11. Thực thể Order (Hóa đơn)

Bảng 2.11 Đặc tả lớp Order - Hóa đơn

STT	Công việc	Kiểu dữ liệu	Ý nghĩa	Ràng buộc	Ghi chú
1	Khai báo thuộc tính:				
1.1	orderID	String	Mã hóa đơn	Dãy số gồm 15 kí tự có dạng HDddmmyyyxxxxx - dd: ngày lập hóa đơn - mm: tháng lập hóa đơn - yyyy: năm lập hóa đơn - xxxx: 5 số nguyên dương phát sinh	{id}

1.2	orderAt	Datetime	Ngày lập hóa đơn	Được lấy tại thời điểm hóa đơn thanh toán thành công	
1.3	status	boolean	Trạng thái thanh toán	Bao gồm hai trạng thái tương ứng là “Chờ thanh toán” ứng với True và “Đã thanh toán” ứng với False	
1.4	subTotal	float	Tổng tiền hóa đơn	Là số thực dương, được tính bằng tổng tiền các sản phẩm trong hóa đơn	
1.5	totalDue	float	Tổng hóa đơn sau khi áp dụng khuyến mãi, thuế.	Là số dương, được tính sau khi trừ đi thuế và áp dụng chương trình khuyến mãi	
1.6	promotion	Promotion	Chương trình khuyến mãi áp dụng vào hóa đơn		
1.7	employee	Employee	Nhân viên lập hóa đơn		

1.8	customer	Customer	Khách hàng thanh toán hóa đơn		
	moneyGiven	Double	Số tiền khách đã đưa		
2	Viết các phương thức getter, setter				
2.1	setOrderID	void	Gán giá trị cho mã hóa đơn	Phát sinh day số theo quy luật HDddmmyyyxxxxx .Trong đó bắt đầu bằng HD + ngày + tháng + năm + 4 số nguyên (Được phát sinh tăng dần). Ví dụ HD2709202300000)	Phát sinh tự động
2.2	setOrderAt(orderAt)	void	Gán thời điểm lập hóa đơn	Kiểm tra thời gian lập hóa đơn <= thời gian hiện tại	
2.4	subTotal	void	Tính tổng hóa đơn		
2.5	totalDue	void	Tính tổng	Tiền thanh toán = Tổng hóa đơn -	[*]

			hóa đơn sau khi áp dụng khuyến mãi, thuế.	Khuyến mãi + Phần trăm thuế*(Tổng hóa đơn-Khuyến mãi)	
3	Viết các constructor				
3.1	OrderDetail()			Hàm khởi tạo mặc định	
3.2	Order(orderID, orderAt, status, subtotal, totalDue, promotion, employee, customer)			Hàm khởi tạo đầy đủ tham số	
3.3	Order(orderID)			Hàm khởi tạo bằng mã.	
4	Viết phương thức toString()	String			trả về chuỗi chứa giá trị các thuộc tính của hóa đơn

2.12.Thực thi OrderDetail (Chi tiết hóa đơn)

Bảng 2.12 Đặc tả lớp OrderDetail

STT	Công việc	Kiểu dữ liệu	Ý nghĩa	Ràng buộc	Ghi chú
1	Khai báo thuộc tính:				

1.1	quantity	int	Số lượng sản phẩm	Số nguyên dương.	
1.2	price	float	Giá bán	Là số thực, dương	
1.3	lineTotal	float	Tổng tiền sản phẩm	Là số thực, dương	
2	Viết các phương thức getter, setter				
2.1	setQuantity	void		Kiểm tra số lượng nhập vào không được rỗng và lớn hơn 0.	throw exception “Số lượng sản phẩm không được nhỏ hơn 1”
2.2	setPrice(orderAt)	void		Kiểm tra giá bán phải lớn hơn giá nhập, không được rỗng và lớn hơn 0	throw exception “Giá bán phải lớn hơn giá nhập, không được rỗng và lớn hơn 0”

2.3	lineTotal	float		Tính tổng tiền của 1 sản phẩm	
3	Viết các constructor				
3.1	OrderDetail()			Hàm khởi tạo mặc định	
3.2	OrderDetail(product, quantity, price, lineTotal)			Hàm khởi tạo đầy đủ tham số	
3.3	Order(product)			Hàm khởi tạo bằng mã.	
4	Viết phương thức toString()	String			trả về chuỗi chứa giá trị các thuộc tính của chi tiết hóa đơn

2.13.Thực thể Product (Sản phẩm)

Bảng 2.13 Đặc tả lớp Product - Sản phẩm

STT	Công việc	Kiểu dữ liệu	Ý nghĩa	Ràng buộc	Ghi chú
1	Khai báo thuộc tính:				
1.1	productID	String	Mã sản phẩm	Mã sản phẩm là chuỗi gồm 9 ký tự có dạng SPabbxxxx. Trong đó: - a: là 1 nếu là sách, 2 nếu là văn phòng phẩm và các sản phẩm khác là 3 - bb: là loại sản phẩm - xxxx: là 4 số nguyên dương	{id}
1.2	name	String	Tên sản phẩm	Không rỗng	
1.3	costPrice	Double	Giá nhập	Lớn hơn 0	
1.4	price	Double	Giá bán		(*)
1.5	type	Type	Loại sản phẩm	<>Enum>>	(*)
2	Viết các phương thức getter, setter				
2.1	setProductID(String ProductID)	void		Phát sinh day số theo quy luật.....	Phát sinh tự động

2.2	setName (String name)	void		Kiểm tra tham số name không được rỗng	throw exception “Name can’t empty” nếu tham số name không hợp lệ
2.3	setCostPrice(Double costPrice)	void		Kiểm tra tham số costPrice lớn hơn 0.	Throw exception “costPrice must > 0” nếu tham số costPrice không hợp lệ
2.4	setPrice(String price)	void		Kiểm tra tham số price > 0 và > costPrice	Throw exception “price must > 0 and > costPrice” nếu tham số price không hợp lệ
3	Viết các constructor				
	Product()				Mặc định
	Product(String name, Double costPrice, Double price)				Đầy đủ tham số
	Product(id)				Chỉ chứa mã

4	Viết phương thức toString()	string			
---	--	--------	--	--	--

(*) Giá bán được tính trên giá nhập kèm theo lợi nhuận mặc định theo từng nhóm sản phẩm sau:

Sách ngoại văn: 60%

Sách trong nước: 50%

Đối với văn phòng phẩm nếu giá nhập dưới 20.000Vnd thì lợi nhuận là 60% và lợi nhuận các sản phẩm từ 20.000Vnd trở lên là 50%

Ghi chú: Enum “Type”:

- VIETNAMESEBOOK: Sách tiếng Việt
- FOREIGNBOOK: Sách ngoại văn
- STARIONERY: Văn phòng phẩm

2.14. Thực thể Promotion (Chương trình khuyến mãi)

Bảng 2.14 Đặc tả lớp Promotion - Chương trình khuyến mãi

STT	Công việc	Kiểu dữ liệu	Ý nghĩa	Ràng buộc	Ghi chú
1	Khai báo thuộc tính:				
1.1	startedDate	Datetime	Ngày bắt đầu	Trước hoặc ngay ngày hiện tại	
1.2	endedDate	Datetime	Ngày kết thúc	Sau ngày bắt đầu	
1.3	type	Int	Loại khuyến mãi	Không rỗng	
1.4	discount	Double	Giảm giá	Là số thực và không âm	
2	Khai báo phương thức :				

2.1	setStartedDate(datetime startedDate)	Void		Kiểm tra tham số startedDate không được rỗng	
2.2	setEndEdDate(datetime endedDate)	Void		Kiểm tra tham số endedDate không được rỗng	
2.3	setType(int type)	Void		Kiểm tra tham số type không được rỗng	
2.4	setDiscount(double discount)	Void			
3	Viết các Constructor				
3.1	Promotion ()			Hàm khởi tạo mặc định	
3.2	Promotion(startedDate, endedDate, type, discount)			Hàm khởi tạo đầy đủ tham số	
4	Viết phương thức toString()	String		Trả về chuỗi chứa giá trị thuộc tính của khuyến mãi	

2.15.Thực thể PurchaseOrder (Đơn nhập hàng)

Bảng 2.15 Đặc tả lớp PurchaseOrder - Đơn nhập hàng

STT	Công việc	Kiểu dữ liệu	Ý nghĩa	Ràng buộc	Ghi chú

1	Khai báo thuộc tính:					
1.1	purchaseOrderID	String	Mã đơn nhập hàng	Là chuỗi gồm x ký tự có dạng DNmmyyyyxx. Trong đó: - mm: là tháng tạo đơn nhập - yyyy: là năm tạo đơn nhập - xx: là hai số nguyên dương phát sinh.	{id}	
1.2	orderDate	Datetime	Thời gian tạo đơn nhập	Được tạo ngay khi tạo đơn nhập		
1.3	receiveDate	Datetime	Thời gian nhận hàng	Sau thời gian tạo đơn nhập		
1.4	note	String	Ghi chú	Không quá 200 ký tự		
1.5	status	Int	Trạng thái đơn nhập	Bao gồm 3 trạng thái tương ứng 0 hoặc 1 hoặc 2	Chứa thông tin trạng thái đơn nhập (2. đã huỷ, 0. chưa	

					nhận, 1.đã nhận)
2	Khai báo phương thức :				
2.1	setPurchaseOrderID(string purchaseOrderID)	Void		Phát sinh mã theo quy luật	Phát sinh tự động
2.2	setOrderDate(datetime orderDate)	Void		Kiểm tra điều kiện thời gian nhập hàng	
2.3	setReceiveDate(datetime receiveDate)	Void		Kiểm tra điều kiện thời gian nhận đơn nhập	
2.4	setNote(string note)	Void			
2.5	setStatus(int status)	Void		Kiểm tra tham số status không được rỗng và là số nguyên trong dãy [0,1,2]	Throw exception “Trạng thái đơn hàng không được rỗng và phải là 0 hoặc 1 hoặc 2” nếu tham số status không hợp lệ
3	Viết các Constructor				

3.1	PurchaseOrder()				Hàm khởi tạo mặc định
3.2	PurchaseOrder(purchaseOrderID, orderDate, receiveDate, status, note)				Hàm khởi tạo đầy đủ tham số
3.3	PurchaseOrder(purchaseOrderID)				Hàm khởi tạo bằng mã
4	Viết phương thức <code>toString()</code>	String			Trả về chuỗi chứa giá trị thuộc tính của đơn nhập hàng

2.16. Thực thể PurchaseOrderDetail (Chi tiết đơn nhập hàng)

Bảng 2.16 Đặc tả lớp PurchaseOrderDetail - Chi tiết đơn nhập hàng

STT	Công việc	Kiểu dữ liệu	Ý nghĩa	Ràng buộc	Ghi chú
1	Khai báo thuộc tính:				
1.1	quantity	Int	Số lượng	Không rỗng và lớn hơn hoặc bằng 1	
1.2	costPrice	Double	Giá gốc	Số thực, không âm	
1.3	lineTotal	Double	Tổng tiền trên một dòng	Số thực, không âm	
1.4	product	Product	Thông tin sản phẩm		
1.5	purchaseOrder	PurchaseOrder	Thông tin đơn nhập hàng		
2	Khai báo phương thức :				
2.1	getLineTotal()	Double	Lấy tổng tiền trên một dòng	Là số thực, không âm	
2.2	setLineTotal(double lineTotal)	void	Tính tổng tiền trên một dòng	Kiểm tra tham số lineTotal >0 và không được rỗng	Tính tổng tiền trên một dòng của sản phẩm dựa vào giá gốc
2.3	setQuantity(int quantity)	Void		Kiểm tra tham số quantity không được rỗng và lớn hơn hoặc bằng 1	Throw exception “Số lượng sản phẩm không hợp lệ” nếu

					tham số quantity không hợp lệ
2.4	setCostPrice(double costPrice)	Void		Kiểm tra tham số costPrice không rỗng và lớn hơn 0	
2.5	setProduct(Product product)	Void			
2.6	setPurchaseOrder(PurchaseOrder purchaseOrder)	Void			
3	Viết các Constructor				
3.1	PurchaseOrderDetail()				Hàm khởi tạo mặc định
3.2	PurchaseOrderDetail(quantity, costPrice, lineTotal, product, purchaseOrder)				Hàm khởi tạo đầy đủ tham số
4	Viết phương thức toString()	String			Trả về chuỗi chứa giá trị thuộc tính của đơn nhập hàng

2.17. Thực thể ReturnOrder (Đơn đổi trả hàng)

Bảng 2.17 Đặc tả lớp ReturnOrder - Đơn đổi trả hàng

STT	Công việc	Kiểu dữ liệu	Ý nghĩa	Ràng buộc	Ghi chú
-----	-----------	--------------	---------	-----------	---------

1	Khai báo thuộc tính:				
1.1	orderDate	Datetime	Thời gian tạo đơn đổi trả	Được khởi tạo vào thời điểm tạo đơn nhập	
1.2	status	Int	Trạng thái đơn đổi trả	Bằng 0 hoặc 1 hoặc 2	Lưu trạng thái đơn đổi trả (0. Chưa giao, 1. Đã giao, 2. Đã huỷ)
1.3	returnOrderID	String	Mã đơn đổi trả	Là chuỗi gồm x ký tự trong đó...	Được phát sinh tự động
1.4	employee	Employee	Nhân viên		Chứa thông tin nhân viên tạo đơn đổi trả
1.5	order	Order	Đơn hàng		Chứa thông tin đơn hàng cần đổi trả
2	Khai báo phương thức :				
2.1	setReturnOrderID(string returnOrderID)	Void			Phát sinh mã tự động theo quy luật

2.2	setOrderDate(datetime orderDate)	Void		Kiểm tra tham số thời gian tạo đơn hàng	
2.3	setStatus(int status)	Void		Kiểm tra tham số status không được rỗng và là số nguyên trong các số [1,2,3]	Throw exception “Trạng thái đơn đổi trả không hợp lệ” khi tham số status không hợp lệ
2.4	setOrder(Order order)	Void			
2.5	setEmployee(Employee employee)	Void			
3	Viết các Constructor				
3.1	ReturnOrder()				Hàm khởi tạo mặc định
3.2	ReturnOrder(returnOrderID, orderDate, status,employee, order)				Hàm khởi tạo đầy đủ tham số
3.3	ReturnOrder(returnOrderID)				Hàm khởi tạo bằng mã đơn đổi trả
4	Viết phương thức toString()		String		Trả về chuỗi

					chứa giá trị thuộc tính của đơn đối trả hàng
--	--	--	--	--	--

2.18.Thực thể Shift (Phiên đăng nhập)

Bảng 2.18 Đặc tả lớp Shift - Phiên đăng nhập

STT	Công việc	Kiểu dữ liệu	Ý nghĩa	Ràng buộc	Ghi chú
1	Khai báo thuộc tính:				
1.1	shiftID	String	Mã phiên đăng nhập	Là chuỗi gồm , có dạng PHaaddmmYYYYxxxx Trong đó: - aa: ca làm (01, 02) - dd: ngày đăng nhập - mm: tháng đăng nhập - yyyy: năm đăng nhập - xxxx: 4 số nguyên dương phát sinh	{id}
1.2	startedAt	DateTime	Thời gian bắt đầu ca làm		
1.3	endedAt	DateTime	Thời gian kết thúc ca làm		

1.4	account	Account	Tài khoản trong ca làm		
2	Viết các phương thức getter, setter				
2.1	setShiftID(String shiftID)	void			Tự động phát sinh
2.2	setStartedAt(DateTime startedAt)	void		Không được rỗng	throw exception “startedAt không được rỗng” nếu tham số startedAt không hợp lệ
2.3	setEndedAt(DateTime endedAt)	void		Không được rỗng	throw exception “endedAt không được rỗng” nếu tham số endedAt không hợp lệ
2.4	setAccount(Account account)	void		Không được rỗng	throw exception “account không được

					rỗng” nếu tham số account không hợp lệ
3	Viết các constructor				
3.1	Shift(String shiftID, DateTime startedAt, DateTime endedAt, Account account)			Hàm khởi tạo đầy đủ tham số	
3.2	Shift()			Hàm khởi tạo mặc định	
3.3	Shift(String shiftID)			Hàm khởi tạo bằng mã	
4	Viết phương thức toString()	string			trả về chuỗi chứa giá trị các thuộc tính của ca làm việc

2.19. Thực thể Stationery (Văn phòng phẩm)

Bảng 2.19 Đặc tả lớp Stationery - Văn phòng phẩm

STT	Công việc	Kiểu dữ liệu	Ý nghĩa	Ràng buộc	Ghi chú
1	Khai báo thuộc tính:				
1.1	color	String	Màu sắc	Không rỗng	
1.2	weight	Double	Trọng lượng	Lớn hơn 0	
1.3	material	String	Chất liệu	Không rỗng	
1.4	origin	String	Xuất xứ	Không rỗng	

1.5	stationeryType	StationeryType	Loại dụng cụ văn phòng phẩm	<<Enum>>	(*)
1.6	Brand	Brand			
2	Viết các phương thức getter, setter				
2.1	setColor(String color)	void		Kiểm tra tham số color không được rỗng	throw exception “color can't empty” nếu tham số color không hợp lệ
2.2	setWeight(Double publisher)	void		Kiểm tra tham số weight phải lớn hơn 0.	throw exception “Weight can't lower 0” nếu tham số weight không hợp lệ
2.3	setMaterial(String material)	void		Kiểm tra tham số material không rỗng	Throw exception “material can't empty” nếu tham số material không hợp lệ

2.5	setOrigin(String origin)	Void		Kiểm tra tham số origin không rỗng	Throw exception “origin can't empty” nếu tham số origin không hợp lệ
3	Viết các constructor				
	Stationery()				Mặc định
	Stationery (String id)				Chỉ chứa mã
	Stationery (String productId, String name, Double costPrice, Double price, String color, Double weight, String material, String origin)				Đầy đủ tham số
4	Viết phương thức toString()	string			

Ghi chú:

- PEN: Viết
- SCHOOL_SUPPLIES: Dụng cụ học tập
- OFFICE_SUPPLIES: Dụng cụ văn phòng
- DRAWING_TOOLS: Dụng cụ vẽ
- PAPER_PRODUCTS: Sản phẩm về giấy
- OTHER_PRODUCTS: Sản phẩm khác

2.20.Thực thể Store (Cửa hàng)

Bảng 2.20 Đặc tả lớp Store - Cửa hàng

STT	Công việc	Kiểu dữ liệu	Ý nghĩa	Ràng buộc	Ghi chú
1	Khai báo thuộc tính:				
1.1	storeID	String	Mã nhà sách		{id}
1.2	name	String	Tên nhà sách		
1.3	address	Address	Địa chỉ nhà sách		
2	Viết các phương thức getter, setter				
2.1	setStoreID(String storeID)	void			Tự động phát sinh
2.2	setName(String name)	void	Kiểm tra tham số name không được rỗng		throw exception “name không được rỗng” nếu tham số name không hợp lệ
2.3	setAddress(Address address)	void			
3	Viết các constructor				
3.1	Store(String storeID, String name, Address address)			Hàm khởi tạo đầy đủ tham số	
3.2	Store()			Hàm khởi tạo mặc định	

3.3	Store(String storeID)			Hàm khởi tạo bằng mã	
4	Viết phương thức toString()	string			trả về chuỗi chứa giá trị các thuộc tính của cửa hàng
STT	Công việc	Kiểu dữ liệu	Ý nghĩa	Ràng buộc	Ghi chú
1	Khai báo thuộc tính:				
1.1	addressID	String	Mã địa chỉ		{id}
1.2	town	String	Xã/thị trấn/phường		
1.3	district	String	Quận/huyện		
1.4	city	String	Tỉnh/thành phố		
1.5	Detail	String	Số nhà, tên toàn..		
2	Viết các phương thức getter, setter				
2.1	setAddressID(String addressID)	void			Tự động phát sinh
2.2	setTown(String town)	void		Không được rỗng	throw exception “town không được rỗng” nếu tham số town không hợp lệ

2.3	setDistrict(String district)	void		Không được rỗng	throw exception “district không được rỗng” nếu tham số district không hợp lệ
2.4	setCity(String city)			Không được rỗng	throw exception “city không được rỗng” nếu tham số city không hợp lệ
2.5	setDetail(String detail0)				
3	Viết các constructor				
3.1	Address(String addressID, String town, String district, String city, String detail)			Hàm khởi tạo đầy đủ tham số	
3.2	Address()			Hàm khởi tạo mặc định	
3.3	Address(String addressID, String town, String district, String city)			Hàm khởi tạo không bao gồm tham số detail	
3.4	Address(String addressID)			Hàm khởi tạo bằng mã	
4	Viết phương thức <code>toString()</code>	string			trả về chuỗi chứa giá trị

					các thuộc tính của địa chỉ
--	--	--	--	--	----------------------------

2.21.Thực thể Supplier (Nhà cung cấp)

Bảng 2.21 Đặc tả lớp Supplier - Nhà cung cấp

STT	Công việc	Kiểu dữ liệu	Ý nghĩa	Ràng buộc	Ghi chú
1	Khai báo thuộc tính:				
1.1	supplierID	String	Mã nhà cung cấp	Là chuỗi gồm x ký tự trong đó ...	
1.2	name	String	Tên nhà cung cấp	Không rỗng	
1.3	address	Address	Địa chỉ nhà cung cấp	Không rỗng	
2	Khai báo phương thức :				
2.1	setSupplierID(string supplierID)	Void			Phát sinh tự động mã nhà cung cấp theo quy luật
2.2	setName(string name)	Void		Kiểm tra tham số name là chuỗi không được rỗng	
2.3	setAddress(Address address)	Void			
3	Viết các Constructor				

3.1	Supplier ()				Hàm khởi tạo mặc định
3.2	Supplier(supplierID, name, address)				Hàm khởi tạo đầy đủ tham số
3.3	Supplier(supplierID)				Hàm khởi tạo bằng mã nhà cung cấp
4	Viết phương thức toString()	String			Trả về chuỗi chứa giá trị thuộc tính của nhà cung cấp

2.22.Thực thể VietnameseBook (Sách tiếng việt)

Bảng 2.22 Đặc tả lớp VietnameseBook - Sách tiếng việt

STT	Công việc	Kiểu dữ liệu	Ý nghĩa	Ràng buộc	Ghi chú
1	Khai báo thuộc tính:				
2	Viết các phương thức getter, setter				
3	Viết các constructor				
	VietnameseBook ()				Mặc định
	VietnameseBook (String id)				Chỉ chứa mã

	VietnameseBook (String productId, String name, Double costPrice, Double price, String author, String publisher, Integer publishYear, String description, Integer PageQuantity, boolean isHardCover)				Đầy đủ tham số
4	Viết phương thức <code>toString()</code>	string			