

1. Class “Product”

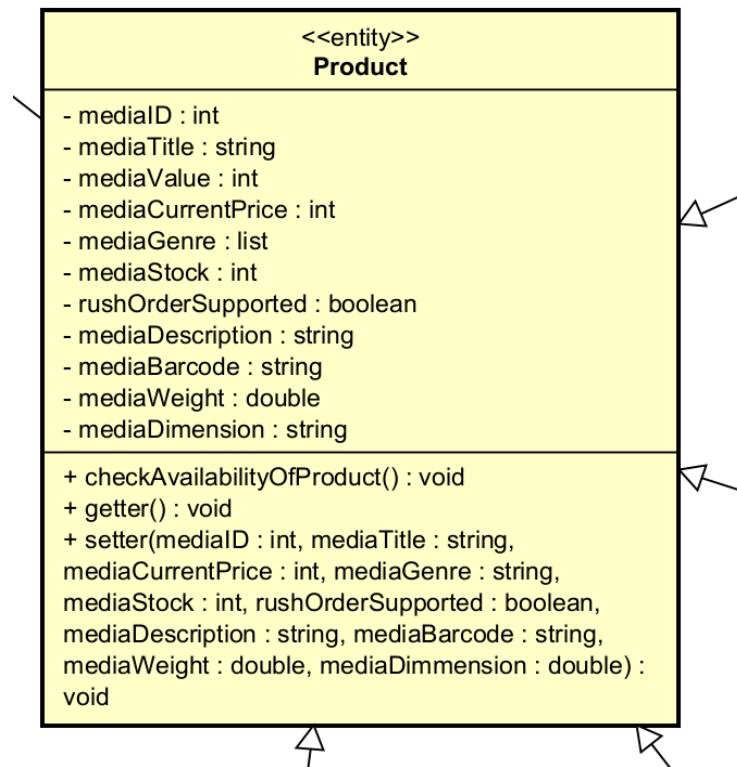


Table 1. Example of attribute design

#	Name	Data type	Default value	Description
1	mediaID	int	N/A	Unique identifier for the product
2	mediaTitle	string	N/A	The name of the product
3	mediaValue			
4	mediaCurrentPrice	int	N/A	The current price of product (VND)
5	mediaGenre	list<String>	N/A	The list of product's genres
6	mediaStock	int	N/A	The quantity of a particular product available in stock
7	rushOrderSupported	boolean	N/A	Indicates whether the product supports rush orders (true/false)
8	mediaDescription	string	N/A	Detailed description of the product
9	mediaBarcode	string	N/A	Barcode number associated with the product

10	mediaWeight	string	N/A	Weight of the product
11	mediaDimmension	string	N/A	Dimensions of the product

Table 2. Example of operation design

#	Name	Return type	Description (purpose)
1	checkAvailabilityOfProduct()	void	Used to check if a product is available in stock
2	getter()	v.v	Used to retrieve the values of the attributes in the Product class
3	setter(...)	void	Used to set the values for the attributes in the Product class

1. Method “CheckAvailabilityOfProduct”

- Parameter

#	Name	Description
1	mediaStock: int	The quantity of a particular product available in stock

- Exception

#	Name	Description
1	OutOfStockException	If the stock value is 0 or less

- How to use parameter/attribute:** *checkAvailabilityOfProduct()* uses the parameter to check if the inventory stock is sufficient for the customer's order.

2. Class “DVD”

DVD
- dvdDiscType : char - dvdDirector : string - dvdRuntime : double - dvdStudio : string - dvdLanguage : string - dvdSubtitles : string - dvdReleaseDate : date - dvdID : int
+ getDvdDiscType() : char + setDvdDiscType(dvdDiscType : char) : void + getDvdDirector() : string + setDvdDirector(dvdDirector : string) : void + getDvdRuntime() : double + setDvdRuntime(dvdRuntime : double) : void + getDvdStudio() : string + setDvdStudio(dvdStudio : string) : void + getDvdLanguage() : string + setDvdLanguage(dvdLanguage : string) : void + getDvdSubtitles() : string + setDvdSubtitles(dvdSubtitles : string) : void + getDvdReleaseDate() : date + setDvdReleaseDate(dvdReleaseDate : date) : void + getDvdID() : int + setDvdID(dvdID : int) : void

Table 1. Example of attribute design

#	Name	Data type	Default value	Description
1	dvdID	int	N/A	Unique identifier for the DVD
2	dvdDiscType	char	N/A	Represents the type of DVD disc (e.g., Blu-ray, HD-DVD)
3	dvdDirector	string	N/A	The director of the movie or content on the DVD
4	dvdRuntime	double	N/A	The total runtime of the DVD
5	dvdStudio	string	N/A	The studio that produced or released the DVD
6	dvdLanguage	string	N/A	The language of the audio on the DVD
7	dvdSubtitle	string	N/A	The subtitle of the DVD
8	dvdReleaseDate	date	N/A	The release date of the DVD

Table 2. Example of operation design

#	Name	Return type	Description (purpose)
1	getter()	void	Used to retrieve the values of the attributes in the DVD class
2	setter()	v.v	Used to set the values for the attributes in the DVD class

3. Class “Book”

Book
- bookID : int - bookPublisher : string - bookCoverType : string - bookPublicationDate : date - bookNumPages : int - bookLanguage : string
+ getBookID() : int + setBookID(bookID : int) : void + getBookPublisher() : string + setBookPublisher(bookPublisher : string) : void + getBookCoverType() : string + setBookCoverType(bookCoverType : string) : void + getBookPublicationDate() : date + setBookPublicationDate(bookPublicationDate : date) : void + getBookNumPages() : int + setBookNumPages(bookNumPages : int) : void + getBookLanguage() : string + setBookLanguage(bookLanguage : string) : void

Table 1. Example of attribute design

#	Name	Data type	Default value	Description
1	bookID	int	N/A	Unique identifier for the book
2	bookPublisher	string	N/A	The publisher of the book
3	bookCoverType	string	N/A	The type of cover for the book
4	bookPublicationDate	date	N/A	The date when the book was published
5	bookNumPages	int	N/A	The total number of pages in the book
6	bookLanguage	string	N/A	The language in which the book is written

Table 2. Example of operation design

#	Name	Return type	Description (purpose)
1	getter()	void	Used to retrieve the values of the attributes in the book class
2	setter()	v.v	Used to set the values for the attributes in the book class

4. Class “LP”

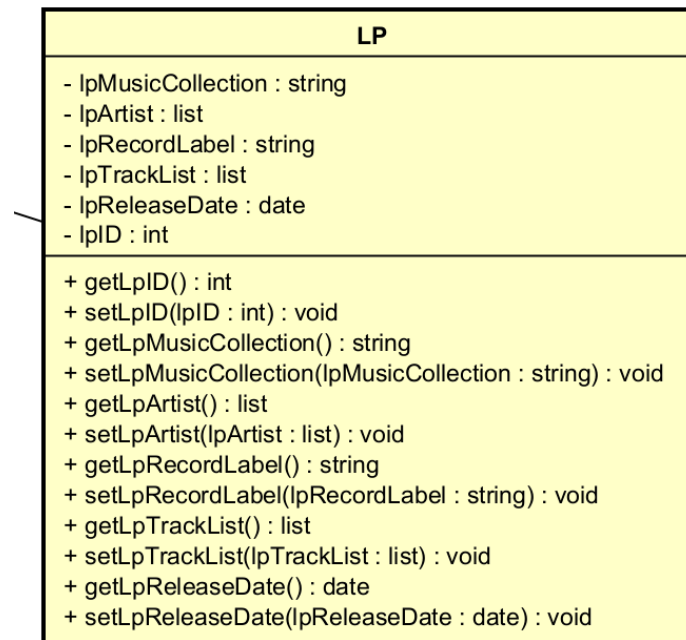


Table 1. Example of attribute design

#	Name	Data type	Default value	Description
1	lpID	int	N/A	Unique identifier for the LP
2	lpMusicCollection	string	N/A	Represents the name or type of the music collection in the LP
3	lpArtist	list	N/A	A list of artists associated with the LP
4	lpRecordLabel	string	N/A	The record label that produced or released the LP
5	lpTrackList	list	N/A	A list of tracks (songs) included on the LP
6	lpReleaseDate	date	N/A	he release date of the LP

Table 2. Example of operation design

#	Name	Return type	Description (purpose)
1	getter()	void	Used to retrieve the values of the attributes in the LP class
2	setter()	v.v	Used to set the values for the attributes in the LP class

5. Class “CD”

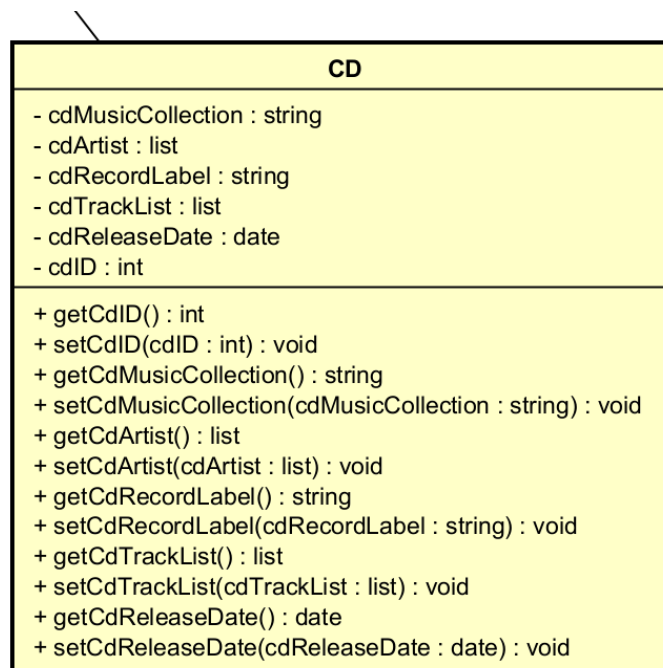


Table 1. Example of attribute design

#	Name	Data type	Default value	Description
1	cdID	int	N/A	Unique identifier for the CD
2	cdMusicCollection	string	N/A	Represents the name or type of the music collection in the CD
3	cdArtist	list	N/A	A list of artists associated with the CD
4	cdRecordLabel	string	N/A	The record label that produced or released the CD
5	cdTrackList	list	N/A	A list of tracks (songs)

				included on the CD
6	cdReleaseDate	date	N/A	he release date of the CD

Table 2. Example of operation design

#	Name	Return type	Description (purpose)
1	getter()	void	Used to retrieve the values of the attributes in the LP class
2	setter()	v.v	Used to set the values for the attributes in the LP class

6. Class “Order”

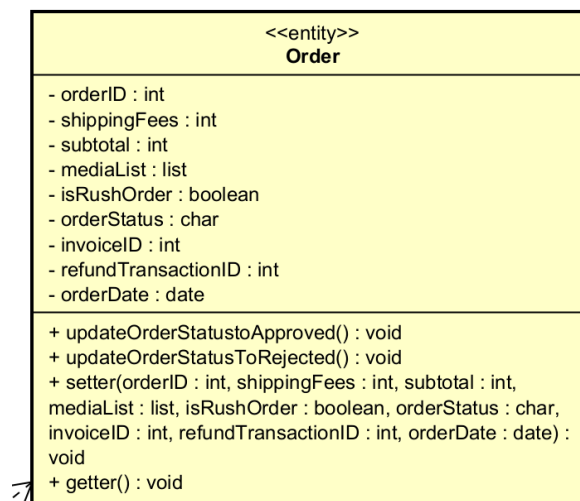


Table 1. Example of attribute design

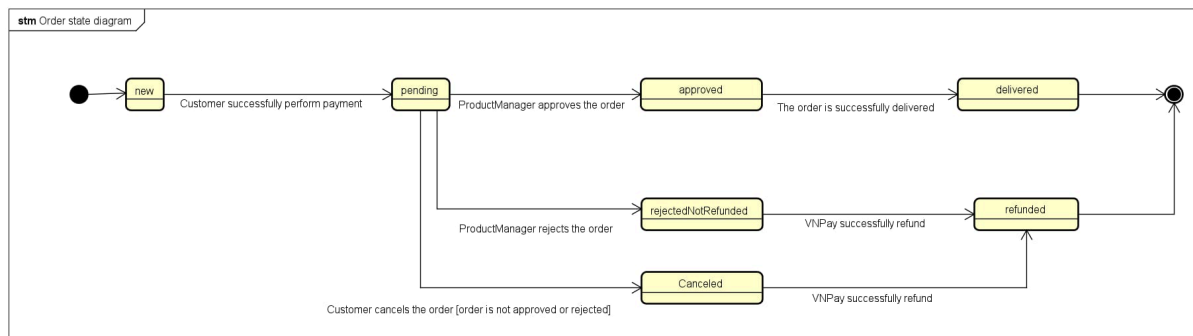
#	Name	Data type	Default value	Description
1	orderID	int	N/A	Unique identifier for the order
2	shippingFees	int	N/A	The cost of shipping for the order
3	subTotal	int	N/A	The total cost of the items before adding shipping fees
4	mediaList	list	N/A	A list of media items associated with the order
5	isRushOrder	boolean	N/A	A flag indicating whether the order is a rush order (true or false)

6	orderStatus	char	N/A	Status of the order
7	invoiceID	int	N/A	The invoice identifier associated with the order
8	refundTransactionID	int	N/A	The transaction ID used for refunds, if applicable
9	orderDate	date	N/A	The date when the order was placed

Table 2. Example of operation design

#	Name	Return type	Description (purpose)
1	updateOrderStatusToApproved()	void	A method to update the order's status to "Approved."
2	updateOrderStatusToRejected()	void	A method to update the order's status to "Rejected."
3	getter()	v.v	Used to retrieve the values of the attributes in the Product class
4	setter(...)	void	Used to set the values for the attributes in the Product class

State Diagram



7. Class “OrderManagementController”

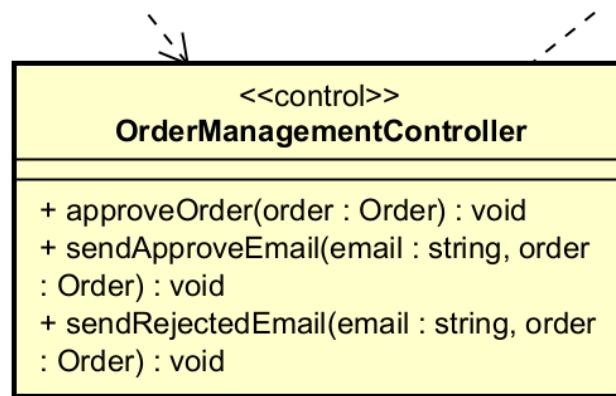


Table 1. Example of attribute design

Table 2. Example of operation design

#	Name	Return type	Description (purpose)
1	approveOrder	void	A method to request to approve order
2	sendApprovedEmail	void	This method is responsible for sending an email to the customer informing them that their order has been approved
3	sendRejectedEmail	void	This method is responsible for sending an email to the customer informing them that their order has been rejected

1. Method “sendApprovedEmail”

- Parameter

#	Name	Description
1	email: string	The email of customer
2	order: Order	This is an object of the Order class

- Exception: N/A

- **How to use parameter/attribute:** *sendApprovedEmail()* uses the parameter to sending an email to the customer informing them that their order has been approved

2. Method “sendRejectedEmail”

- Parameter

#	Name	Description
1	email: string	The email of customer

2	order: Order	This is an object of the Order class
---	--------------	--------------------------------------

- **Exception: N/A**
- **How to use parameter/attribute:** *sendRejectedEmail()* uses the parameter to sending an email to the customer informing them that their order has been rejected

3. Method “approveOrder”

- **Parameter**

#	Name	Description
1	order: Order	This is an object of the Order class

- **Exception: N/A**

Class “OrderDetailScreen”

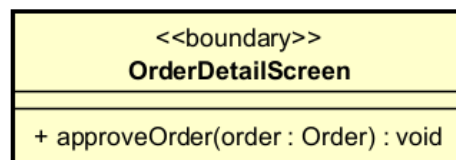


Table 1. Example of attribute design

Table 2. Example of operation design

#	Name	Return type	Description (purpose)
1	approveOrder	void	A method to request to approve order