



2013

dBay

INFO 620- Information Systems Analysis &
Design



Table of Contents

Proposal.....	2
Use Case Models.....	6
Class Models.....	24
Sequence Diagrams.....	27
Design Class Diagram.....	32
RDB Schema and Extra Credit Diagrams.....	34
Final Evaluations and Independent Work.....	36

Proposal

1. Title: Analysis and Design of dBay (Drexel Version of eBay System)

The Problem Statement

Drexel would like to create its own version of eBay called dBay that could be utilized by students, faculty, and staff members of the university—various items such as textbooks and electronics can be sold and bought in this database.

a. Context and Importance of the system

Drexel is known for their fast paced culture and rigorous all year round quarters. Thus, it is necessary for a database to be created to assist students, faculty, and staff members connect and be able to sell/buy textbooks and electronics. dBay will be able to provide the fast connection to obtain a copy of the latest edition of a textbook for a class as well as needed electronics such as iPads, laptops, etc.

b. Overall goals of the system

For Drexel to be aware of what products are popular amongst their students, faculty, and staff members, as well as experience financial benefit by buying and selling used textbooks, electronics, etc.

c. Scope of the Project

IN-Scope:

Only items directly relevant to academic life will be sold on D-bay: textbooks, electronics

OUT-Scope:

D-bay will not be used to sell such things as property, automobiles or bicycles to students and staff

2. Requirements

2.1 Functional Requirements

- 1) User will have a “mydBay” dashboard
 - a. User will have four primary interfaces via the dashboard: buy, sell, watch, and message
- 2) Browse and track items for auction (watch)
- 3) Bid on items for auction (buy)
- 4) Pay for items won

- 5) Place items up for auction (sell)
- 6) Receive money for items sold
- 7) Communicate with other eBay buyers and sellers (message)

2.2 Data requirements (Partial List)

1) USER

- Unique ID of each user
- Record transaction history
- Shipping information
- Contact information
- Payment information

2) TRANSACTION

- Record each transaction
- Extract sales tax
- Time item was for sale
- Number of bidders

3) ITEMS

- Track each item bought and sold to observe trends
- Categorize items
- Classify items as appropriate (books and electronics) or inappropriate (housing and vehicles)

2.3 Business Rules and Data Logic.

1. eBay's Buyer Protection Program— If you receive a product that does not match the products description then you can contact the seller via email only. However, if the seller does not response after 3 days, the buyer can contact eBay to address the matter.
2. Transaction may be canceled if the item has not been shipped.
3. Items can be returned within 15 days of the purchase.
4. Credit cards, bank wire transfers, checks, money orders, and meet up to exchange cash is not acceptable. All transaction must be done via PayPal for documentation.
5. eBay allows buyers to rate their sellers; however, no offensive comments will be posted--including political & religious views, personal opinions, etc.
6. Before making a purchase users will need to select the "add to cart" button which temporarily places the item into a checkout "cart". Once this step is completed users can either decide to continue browsing or purchase the item(s) in their cart.
7. Users may add items to their "wish list" for future viewing or purchasing purposes.
8. Upon winning an auction a buyer confirmation email is sent to the purchaser. The email informs the buyer that they have won the item, and the next step is to pay.
9. Once an item is successfully paid for the buyer will receive an confirmation email thanking them for their purchase and informing them of where to go for details on their purchase.

10. Buyers will be given 48 hours to pay for their purchase. If money isn't received during this time frame then the user can choose to have dBay send a payment reminder the buyer.
11. Once the seller marks the item as shipped dBay will send the buyer an order update with tracking information.
12. During a live auction buyers will be allowed to submit questions directly to the seller regarding the item/listing.
13. Buyers may provide special instructions on their order during checkout.
14. Seller invoices will be listed under "My dBay" dashboard. The invoice will feature item(s) sold, description of item(s), and the current balance (transaction fee from final sale including shipping).
15. Sellers are charged a 7% transaction fee
16. Users PayPal accounts will be directly accessible via "My dBay" dashboard. This will include a link to the actual site, account history, and users profile summary.
17. As a general rule, dBay will not allow buyers to retract, remove, or cancel a bid. When users place a bid, they agree to pay for the item if they are indeed the winning bidder. However there are exceptions. Valid reasons for retracting a bid include: You accidentally entered the wrong bid amount. For example, you bid \$99.50 instead of \$9.95. - The item's description changed significantly after you entered your last bid. For example, the seller updated the details about the item's features or condition. - You can't reach the seller by telephone or email. Changing your mind isn't a valid reason for retracting a bid. Be sure to carefully review listings before bidding.
18. Buyers can either have their items shipped or arrange for pick up at central location on Drexel's main campus.

2.4 Non-functional requirements

- 1) Allow users to create an online auction account (usability)
 - a. An @drexel.edu email will be required to set up account
 - b. Drexel member validation via email verification
 - c. Usernames must be 8 characters long and include a minimum of 2 letters to ensure Drexel IDs are not used.
- 2) A valid PayPal account must be linked to auction account (security)
 - a. PayPal will be only accepted form of payment
 - b. Verification will be performed by secure web service
 - c. Customer financial data will not reside on dBay servers and instead all financial transactions will be conducted through PayPal
- 3) User will have a "myDbay" dashboard (interface)
 - a. User will have four primary interfaces via the dashboard: buy, sell, watch, and message

4) Application will be distributed across a clustered hardware infrastructure in a co-located data center with redundant critical infrastructure (reliability)

5) Users will have to agree to a TOS statement which will outline prohibited items as well as a code of conduct to regulate users interactions and behaviors (legal)

2.5 Other Assumptions

1. We assume all Drexel students, staff and faculty are tech savvy
2. dBay is hosted on Drexel's server and is accessed via a web browser
3. The back end database will be hosted on an oracle server.
4. Transactions will be facilitated via an Oracle Database.
5. Website will be hosted by Drexel owned servers.
6. Items and/or invoices will automatically be catalogued, calculated, and dispersed.
7. The system will seamlessly interface with outside courier services (USPS, UPS, FedEx, etc.)

Examples of system input/output, etc

1. Record all items bought and sold
2. Record buyer and seller
3. Classify items by category and specific item
4. Reports of most popular items
5. Report trends of when items are bought and sold
6. Reports of items that may be deemed not suitable

Knowledge Acquisition

dBay is designed to provide the convenience of an e-commerce geared towards Drexel's peers that provides the supply and demand products that are relevant to academic life. We will consult test users to ensure dBay is user-friendly.

Software and/or hardware involved

The UML diagrams will be developed in Rational Rose. The website will be developed using Java. Inventory transactions and buying history will interface with an Oracle database. It will be hosted on servers already owned by the college. Users will be able to post through a web portal

Proposed Deliverables and work plans

We will submit UML diagrams modeling the dBay system. In addition we will document our design process and the logic behind each element. We will also elicit feedback from prospective users and include that feedback to improve our process and provide thoughts on possible future developments for the project.

3. Use Case Model

3.1 Actors and Their Goals

1. Seller (primary actor): The seller creates an account with dBay, links PayPal account, and list item(s) for sale.
2. Buyer (primary actor): The buyer creates an account with dBay, links PayPal account, and bids on auction item(s) posted for sale.
3. System (secondary actor): The system auto generates email confirmations to both the seller and buyer once items are sold.
4. PayPal (secondary actor): PayPal serves as a global e-commerce business allowing payments and money transfers to be made through the Internet.

3.2 Use Case Diagram

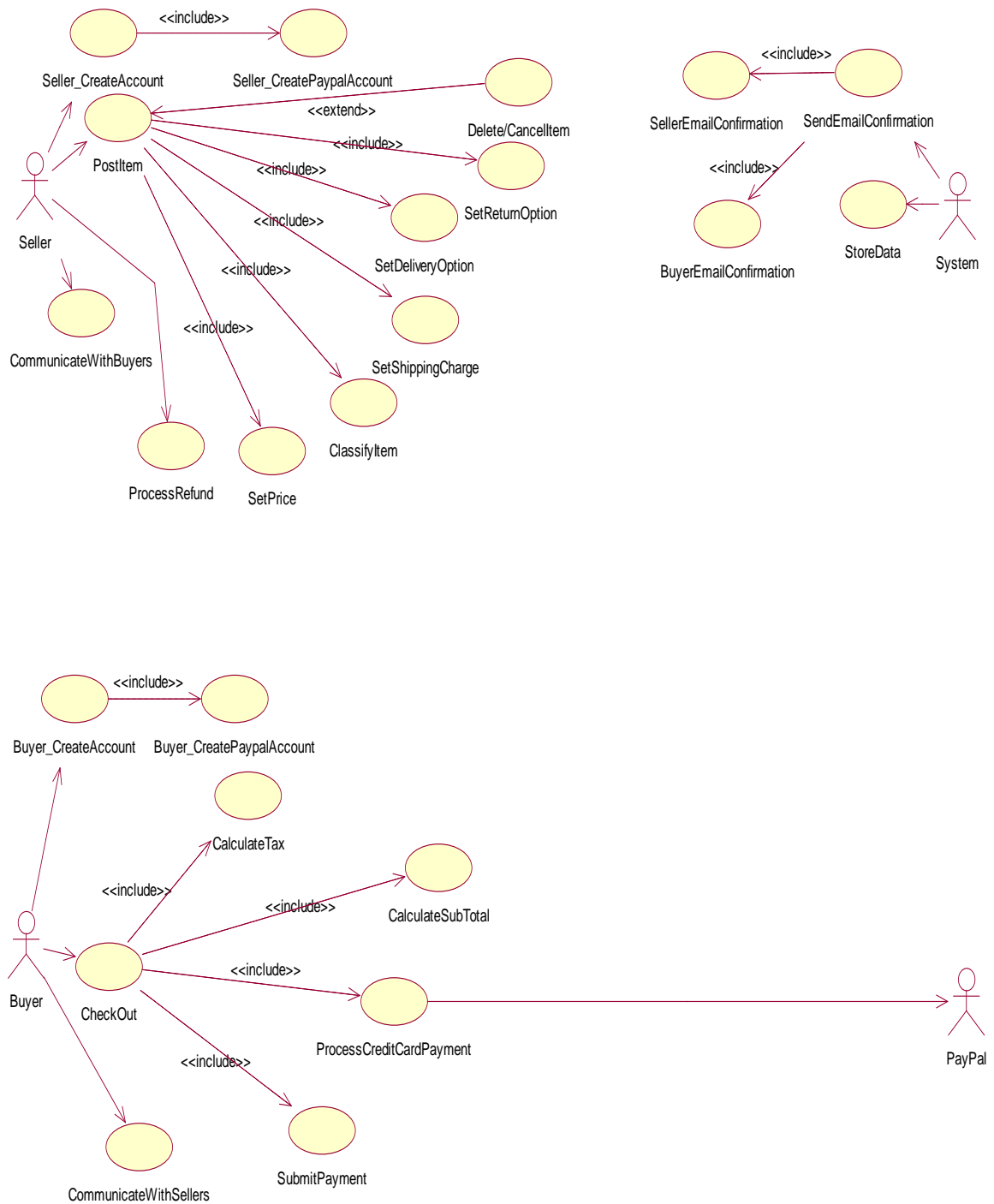


Figure 1.

3.3 Overview section of all the use cases

Use Case Name	Actor	Type	Overview and Scope	Subordinates
Seller_CreateAccount	Seller	Base	Seller creates dBay account	N/A
CommunicateWithBuyers	Seller	Base	Seller electronically communicates with buyer	N/A
StoreData	System	Base	System stores metadata.	N/A
SendEmailConfirmation	System	Base	System processes email confirmations	N/A
SellerEmailConfirmation	System	Include	System generates email confirmation for items sold	SendEmailConfirmation.
BuyerEmailConfirmation	System	Include	System generates email confirmation for items sold	SendEmailConfirmation
CommunicateWithSellers	Buyer	Base	Buyer electronically communicates with seller.	N/A
CalculateTax	Buyer	Base	System calculates tax	N/A
Buyer_CreateAccount	Buyer	Base	Buyer creates dBay account	N/A
Buyer_CreatePaypalAccount	Buyer	Include	Buyer creates PayPal account	Buyer_CreateAccount

Figure 2.

3.4 Use Case Description

Figures 3.

USE CASE #	3	
USE CASE Name	Post_Item	
ACTOR	Seller	
Goal (1 phrase)	To post (list) items on site for sale	
Overview and scope	This use case describes the process whereby a seller electronically lists his/her item on dBay for sale. The seller will be prompted by the system to undergo a series of steps before the item is ready for auction.	
Level	Primary	
Preconditions	User must be motivated to sell User must have valid checking account User must have PayPal account	
Postconditions in words(write in passive and past tense)	The system process the item. The items starting value, shipping, return, refund, and delivery options are set. The item goes “live” and is available for auction.	
Trigger	User has an item he/she wants to sell	
Included Use Cases	SetDeliveryOption, SetShippingCharge, Classify_Item, SetReturnOption, Set_Price	
Extending Use Cases	Delete/CancelItem	
MAIN SUCCESSFUL SCENARIO in numbered sequence Reference “included use cases” in this section using INCLUDE <i>ius_name</i>	Actor Action	System Action
	1. Seller selects “list item”	2. System takes seller to “classify item” page.
	3. Actor selects appropriate item classification(s).	4. System prompts seller to select “next” if satisfied with classification.
	Actor selects “next”	5 System: SetPrice, SetShippingCharge, SetDeliveryOption, SetReturnOption.
	6. Actor selects “finished/auction item”	7. System processes request and begins “live auction”
UNSUCCESSFUL	Conditions	Actions

SCENARIOS (erroneous situations)	Item cannot be listed for auction (motor vehicles, clothing, furniture, etc.)	Abort the transaction
	Seller does not have valid banking and/or PayPal account	Abort the transaction
Priority in scheduling	Primary	
Frequency	Whenever a user wishes to auction an appropriate item	
Business rules and data logic	Item must be appropriate. Seller must classify Item. Auction options must be selected. PayPal account must be linked.	
Other non-functional requirements	All information must be valid. Network connection must be established.	
Superordinates	None	
Developer	Shawn Woodson, Matthew Lyke, Betty Nguyen	
Creation date and last modified date	08/18/2013	
Other Comments	N/A	

USE CASE #	4
USE CASE Name	Set_Price
ACTOR	Seller
Goal (1 phrase)	To set a starting price for a listed item.
Overview and scope	This use case describes the process whereby a seller sets a starting sell price for their auction item. The seller will need to set the starting price at \$0.99 or higher.
Level	Primary
Preconditions	User must be motivated to sell User must have valid checking account User must have PayPal account
Postconditions in words (write in passive and past tense)	The items sale value is augmented and confirmed.
Trigger	Seller has classified the item and is ready to select his/her auction options.
Included Use Cases	SetDeliveryOption, SetShippingCharge, Classify_Item, SetReturnOption
Extending Use Cases	Delete/CancelItem

MAIN SUCCESSFUL SCENARIO in numbered sequence Reference “included use cases” in this section using INCLUDE <i>ius_name</i>	Actor Action	System Action
	1. Seller places cursor in “price” field and enters a number.	2. System displays number.
UNSUCCESSFUL SCENARIOS (<i>erroneous situations</i>)	Conditions	Actions
	Item cannot be listed for auction (motor vehicles, clothing, furniture, etc.)	Abort the transaction
	Seller does not have valid banking and/or PayPal account	Abort the transaction
Priority in scheduling	Primary	
Frequency	Whenever a user wishes to auction an valid item	
Business rules and data logic	Item must be appropriate. Seller must classify Item. Auction options must be selected. PayPal account must be linked.	
Other non-functional requirements	All information must be valid. Network connection must be established.	
Superordinates	None	
Developer	Shawn Woodson, Matthew Lyke, Betty Nguyen	
Creation date and last modified date	08/18/2013	
Other Comments	N/A	

USE CASE #	5
USE CASE Name	Classify_Item
ACTOR	Seller
Goal (1 phrase)	To properly classify the item for sale
Overview and scope	This use case describes the process whereby a seller describes and/or validates the exact item they wish to auction. The system will prompt the seller to attribute accompanying characteristics and specifications.
Level	Primary
Preconditions	User must be motivated to sell User must have valid checking account

	User must have PayPal account	
Postconditions in words (write in passive and past tense)	The items is a given an identity and attributes are subscribed.	
Trigger	User creates seller account.	
Included Use Cases	SetDeliveryOption, SetShippingCharge, Set_Price, SetReturnOption	
Extending Use Cases	Delete/CancelItem	
MAIN SUCCESSFUL SCENARIO in numbered sequence Reference “included use cases” in this section using INCLUDE <i>ius_name</i>	Actor Action	System Action
	1. Seller places cursor in “title” field and gives his/her item a name.	2. System displays and stores information.
	3. Seller places cursor in “Description” field and enters item characteristics.	4. System displays and stores information.
	5. Seller selects “save/next”	5. System saves information and displays subsequent web page.
UNSUCCESSFUL SCENARIOS (<i>erroneous situations</i>)	Conditions	Actions
	Item cannot be listed for auction (motor vehicles, clothing, furniture, etc.)	Abort the transaction
	Seller does not have valid banking and/or PayPal account	Abort the transaction
Priority in scheduling	Primary	
Frequency	Whenever a user wishes to auction an valid item	
Business rules and data logic	Item must be appropriate. Seller must classify Item. Auction options must be selected. PayPal account must be linked.	
Other non-functional requirements	All information must be valid. Network connection must be established.	
Superordinates	None	
Developer	Shawn Woodson, Matthew Lyke, Betty Nguyen	
Creation date and last modified date	08/18/2013	
Other Comments	N/A	

USE CASE #	6	
USE CASE Name	SetShippingCharge	
ACTOR	N/A	
Goal (1 phrase)	To set a shipping charge for the auctioned item	
Overview and scope	This use case describes how the user will set the shipping charge for the auctioned item.	
Level	Include	
Preconditions	1. There is at least 1 item to auction by a seller.	
Postconditions in words	1. The system stores a history of the seller's auctioned items.	
Trigger	The seller is in the process of trying to create a profile for the auction item.	
Included Use Cases	N/A	
Extending Use Cases	N/A	
MAIN SUCCESSFUL SCENARIO	Actor Action	System Action
	1. Seller clicks on "Set Shipping Charge" button	2. System generates a list of shipping charges base on the length of ship (i.e. 3 business days, etc.)
	3. Seller selects the appropriate shipping charge	4. System processes the chosen option
UNSUCCESSFUL SCENARIOS	Conditions	Actions
	1, 3. Internet connection failed	Transaction failure
Priority in scheduling	Secondary	
Frequency	Every time the seller has an item for auction.	
Business rules and data logic	Each seller must have item they wish to auction.	
Other non-functional requirements	The information should all be valid.	
Superordinates	N/A	
Developer	Matthew Lyke, Betty Nguyen, and Shawn Woodson	
Creation date and last modified date	August 13, 2013	

Other Comments	N/A	
USE CASE #	7	
USE CASE Name	SetDeliveryOption	
ACTOR	N/A	
Goal (1 phrase)	To set a delivery method for the auctioned item	
Overview and scope	This use case describes how the user will set the delivery method for the auctioned item.	
Level	Include	
Preconditions	1. There is at least 1 item to auction by a seller.	
Postconditions in words	1. The system stores a list of the seller's auctioned item with its delivery methods.	
Trigger	The seller is in the process of trying to create a profile for the auction item.	
Included Use Cases	N/A	
Extending Use Cases	N/A	
MAIN SUCCESSFUL SCENARIO	Actor Action	Actor Action
	1. Seller clicks on "Set Delivery Option" button	1. Seller clicks on "Set Delivery Option" button
	3. Seller selects the appropriate delivery method	3. Seller selects the appropriate delivery method
UNSUCCESSFUL SCENARIOS	Conditions	Conditions
	1, 3. Internet connection failed	1, 3. Internet connection failed
Priority in scheduling	Secondary	
Frequency	Every time the seller has an item for auction.	
Business rules and data logic	Each seller must have item they wish to auction.	
Other non-functional requirements	The information should all be valid	
Superordinates	N/A	
Developer	Matthew Lyke, Betty Nguyen, and Shawn Woodson	
Creation date and last modified date	August 13, 2013	

Other Comments	N/A	
USE CASE #	8	
USE CASE Name	SetReturnOption	
ACTOR	N/A	
Goal (1 phrase)	To set a return option for the auctioned item	
Overview and scope	This use case describes how the user will set the return option for the auctioned item.	
Level	Included	
Preconditions	1. There is at least 1 item to auction by a seller.	
Postconditions in words	1. The system stores a list of the seller's auctioned item with its return option.	
Trigger	The seller is in the process of trying to create a profile for the auction item.	
Included Use Cases	N/A	
Extending Use Cases	N/A	
MAIN SUCCESSFUL SCENARIO	Actor Action	System Action
	1. Seller clicks on "Set Return Option" button	2. System generates a list of return options (i.e. 7 days, 15 days, 30 days etc.)
	3. Seller selects the appropriate return option	4. System processes the chosen option
UNSUCCESSFUL SCENARIOS	Conditions	Actions
	1, 3. Internet connection failed	Transaction failure
Priority in scheduling	Secondary	
Frequency	Once	
Business rules and data logic	Every time the seller has an item for auction.	
Other non-functional requirements	Each seller must have item they wish to auction.	
Superordinates	N/A	
Developer	Matthew Lyke, Betty Nguyen, and Shawn Woodson	
Creation date and last modified date	August 13, 2013	
Other Comments	N/A	

USE CASE #	14	
USE CASE Name	CheckOut_Buyer	
ACTOR	Buyer	
Goal (1 phrase)	To finalize the purchase of an item	
Overview and scope	This use case describes how the user finalizes the purchase of items on eBay.	
Level	Primary	
Preconditions	1. The buyer must have a valid account. 2. The buyer must have a valid PayPal account. 3. The buyer must have selected an item. 4.	
Postconditions in words	1. The system stores the sale. 2. User's PayPal account is charged.	
Trigger	The buyer finds an item she wants.	
Included Use Cases	<<submit payment>> <<process credit card payment>> <<calculate sub total>> <<calculate tax>>	
Extending Use Cases		
MAIN SUCCESSFUL SCENARIO	Actor Action	System Action
	1. Buyer selects purchase option	2. System asks the buyer to approve the charges
	3. Buyer approves charges	4. System processes purchase
UNSUCCESSFUL SCENARIOS	Conditions	Actions
	1. Item is already purchased	Purchase is rejected
	3. PayPal rejects payment	Purchaser is asked to revise payment details
Priority in scheduling	Primary	
Frequency	Whenever	

Business rules and data logic	Buyer must select an option. Item must still be for sale. Paypal account must be current.
Other non-functional requirements	The information should all be valid
Superordinates	N/A
Developer	Matthew Lyke, Betty Nguyen, and Shawn Woodson
Creation date and last modified date	August 13, 2013
Other Comments	N/A

USE CASE #	15	
USE CASE Name	Calculate Tax	
ACTOR	Buyer	
Goal (1 phrase)	To add the tax to a dBay purchase	
Overview and scope	This use case describes how tax is added to the final purchase of an item from dBay.	
Level	Include	
Preconditions	1. The buyer must have a valid account. 2. The buyer must have a valid PayPal account. 3. The buyer must have selected an item. 4. The purchase must have been accepted	
Postconditions in words	1. The system stores the sale. 2. User's PayPal account is charged. 3. Tax is recorded	
Trigger	The buyer approves the payment.	
Included Use Cases	N/A	
Extending Use Cases	N/A	
MAIN SUCCESSFUL SCENARIO	Actor Action	System Action
	1. Buyer approves payment	2. System processes sale
UNSUCCESSFUL	Conditions	Actions

SCENARIOS	1. Paypal rejects payment	2. Purchaser is asked to revise payment details
Priority in scheduling	Secondary	
Frequency	Once an item is purchased	
Business rules and data logic	Buyer must select an option. Item must still be for sale. Paypal account must be current.	
Other non-functional requirements	Tax rates in the system should be up to date for city, county, state	
Superordinates	Checkout	
Developer	Matthew Lyke, Betty Nguyen, and Shawn Woodson	
Creation date and last modified date	August 13, 2013	
Other Comments	N/A	

USE CASE #	16
USE CASE Name	Calculate Sub Total
ACTOR	Buyer
Goal (1 phrase)	To process the subtotal of a purchase pretax and pre-shipping
Overview and scope	This use case describes the base cost of an item purchased on dBay prior to the addition of taxes and shipping cost.
Level	Include
Preconditions	1. The buyer must have a valid account. 2. The buyer must have a valid PayPal account. 3. The buyer must have selected an item. 4. The cost information must be accurate
Postconditions in words	1. The system records the sub total
Trigger	The buyer selects the item.
Included Use Cases	N/A
Extending Use Cases	N/A

MAIN SUCCESSFUL SCENARIO	Actor Action	System Action
	1. Buyer approves subtotal	2. System processes tax and shipping
UNSUCCESSFUL SCENARIOS	Conditions	Actions
	1. System cannot determine the cost	2. Sale is suspended
Priority in scheduling	Secondary	
Frequency	Once an item is selected	
Business rules and data logic	Buyer must select an item. Item must still be for sale. Price must be stored in the system.	
Other non-functional requirements		
Superordinates	Checkout	
Developer	Matthew Lyke, Betty Nguyen, and Shawn Woodson	
Creation date and last modified date	August 13, 2013	
Other Comments	N/A	

USE CASE #	17
USE CASE Name	ProcessCreditCardPayment
ACTOR	Buyer, PayPal
Goal (1 phrase)	To process a credit card payment through PayPal.
Overview and scope	This use case describes the process of a credit card through the users pay pal account.
Level	Include
Preconditions	<ol style="list-style-type: none"> 1. The buyer must have a valid account. 2. The buyer must have a valid PayPal account. 3. The buyer must have selected an item. 4. The cost information must be accurate 5. The user must have approved the transaction

Postconditions in words	1. The system records the sub total	
Trigger	The buyer selects the item.	
Included Use Cases	N/A	
Extending Use Cases	N/A	
MAIN SUCCESSFUL SCENARIO	Actor Action	System Action
	1. Buyer approves subtotal	2. System processes tax and shipping
	3. Buyer approves credit card payment	4. system processes payment
UNSUCCESSFUL SCENARIOS	Conditions	Actions
	1. PayPal process fails	2. System requests the user review information
Priority in scheduling	Secondary	
Frequency	Once an item is selected	
Business rules and data logic	Buyer must select an item. Item must still be for sale. Price must be stored in the system.	
Other non-functional requirements		
Superordinates	Checkout	
Developer	Matthew Lyke, Betty Nguyen, and Shawn Woodson	
Creation date and last modified date	August 13, 2013	
Other Comments	N/A	

USE CASE #	18
USE CASE Name	Submit Payment
ACTOR	Buyer, PayPal

Goal (1 phrase)	To process the payment of the item on dBay.	
Overview and scope	This describes the procedure for processing a payment from a user for a dBay purchase utilizing their PayPal account	
Level	Include	
Preconditions	1. The buyer must have a valid account. 2. The buyer must have a valid PayPal account. 3. The buyer must have selected an item. 4. The buyer must have confirmed payment 5. The connection with the PayPal server must be active	
Postconditions in words	1. User's PayPal account is charged 2. User is authorized to receive the item. 3. PayPal accesses the bank or credit card to settle the account.	
Trigger	The buyer confirms the charged.	
Included Use Cases	none	
Extending Use Cases	none	
MAIN SUCCESSFUL SCENARIO	Actor Action	System Action
	1. Buyer confirms purchase	2. System initiates connection with PayPal
	3. PayPal confirms buys account	4. System requests permission to charge buyers account
	5. PayPal initiates transaction	6. System receives payment and notifies buyer
	7. Buyer confirms receipt.	
UNSUCCESSFUL SCENARIOS	Conditions	Actions
	1. Paypal rejects payment	Buyer is asked to revise payment details
	2. Connection to PayPal fails	Buyer is asked to wait
	3.	
Priority in scheduling	Primary	
Frequency	Whenever someone purchases an item	
Business rules and data logic	Buyer must select an option. Item must still be for sale. Paypal account must be current. Paypal connection to dBay must be current.	

Other non-functional requirements	The information should all be valid
Superordinates	N/A
Developer	Matthew Lyke, Betty Nguyen, and Shawn Woodson
Creation date and last modified date	August 13, 2013
Other Comments	N/A

3.6 Discussion

One of the difficulties we had between Seller and Buyer were differentiating between several scenarios specifically, which scenario should be considered an extended or included use case. As you know, the definition for an extend use case is: A) several possible alternate scenarios, B) failure scenario that do not achieve the goal. The definition of an include use case is a functionality that stands alone that can be used by more than one use case. There were several instances where we thought an extend use case should be used as an include use case and vice versa. For example, we thought SetShippingCharge, SetDeliveryOption, and SetReturnOption could have been an extend use case. Our reasoning for this is because we believed these particular use cases could not stand alone and could not be used by more than one use case. Theoretically, since the system could also play a role in various scenarios we decided to switch our extend use cases to includes.

Another issue we had was trying to figure out which use case should be with Seller and which use case should be with Buyer. For example, there were some discussions with figuring out if SetDeliveryOption should be with the Seller or Buyer. It was then that Shawn walked us through his eBay account showing that the Buyer has the power to set his/her delivery option. After the tutorial, we as a group decided it made more sense to move SetDeliveryOption from the Buyer to the Seller, and made it an “include use case” with PostItem. We also had the same issue with SetShippingCharge, and decided it made much more sense to have both SetDeliveryOption and SetShippingCharge together, especially after viewing an example of eBay.

4. Class Model

4.1 Thinking Procedure

The step towards creating our analysis class diagram began by identifying nouns, noun phrases, and textual descriptions of the domain. We then began to walk through our category list, which presented a step-by-step analysis of our use case transactions. In order to determine what should be considered and not considered a conceptual class, we consulted your Week 4 lecture material on Chapters 8 and 9. According to Larman's rule, an attribute should be considered a number or text. Therefore, (according to your notes) a concept is an actual thing that takes space, an entity in the world. We used both forms of advice to arrive at our final conclusion as it pertains to our conceptual classes. We excluded other use cases, such as, Seller/Buyer Create Account, Buyer Checkout, etc., because we felt that these use cases would not be represented in the software layer. However, we went back and forth with this issue for quite some time, because we were unable to figure out if these particular use cases would be classified as a tangible or intangible case, because, again as mentioned in your notes, a use case is an actual thing that takes space, or an entity of the world, and there was some confusion amongst the group as to whether the aforementioned use cases qualified as such.

We also had difficulty with the cardinality, because we were unable to view the diagram from one scenario vs. multiple scenarios. There were some arguments within the group as to whether or not the cardinality between Buyer and Item should be multiple buyers purchases multiple items, or one buyer purchases multiple items. We felt that all of the included relationships denoted some semblance of actual real world relationships in which we need some memory. Therefore, we were confident our associations accurately depicts the relationships amongst the classes.

4.2 The Class Diagram

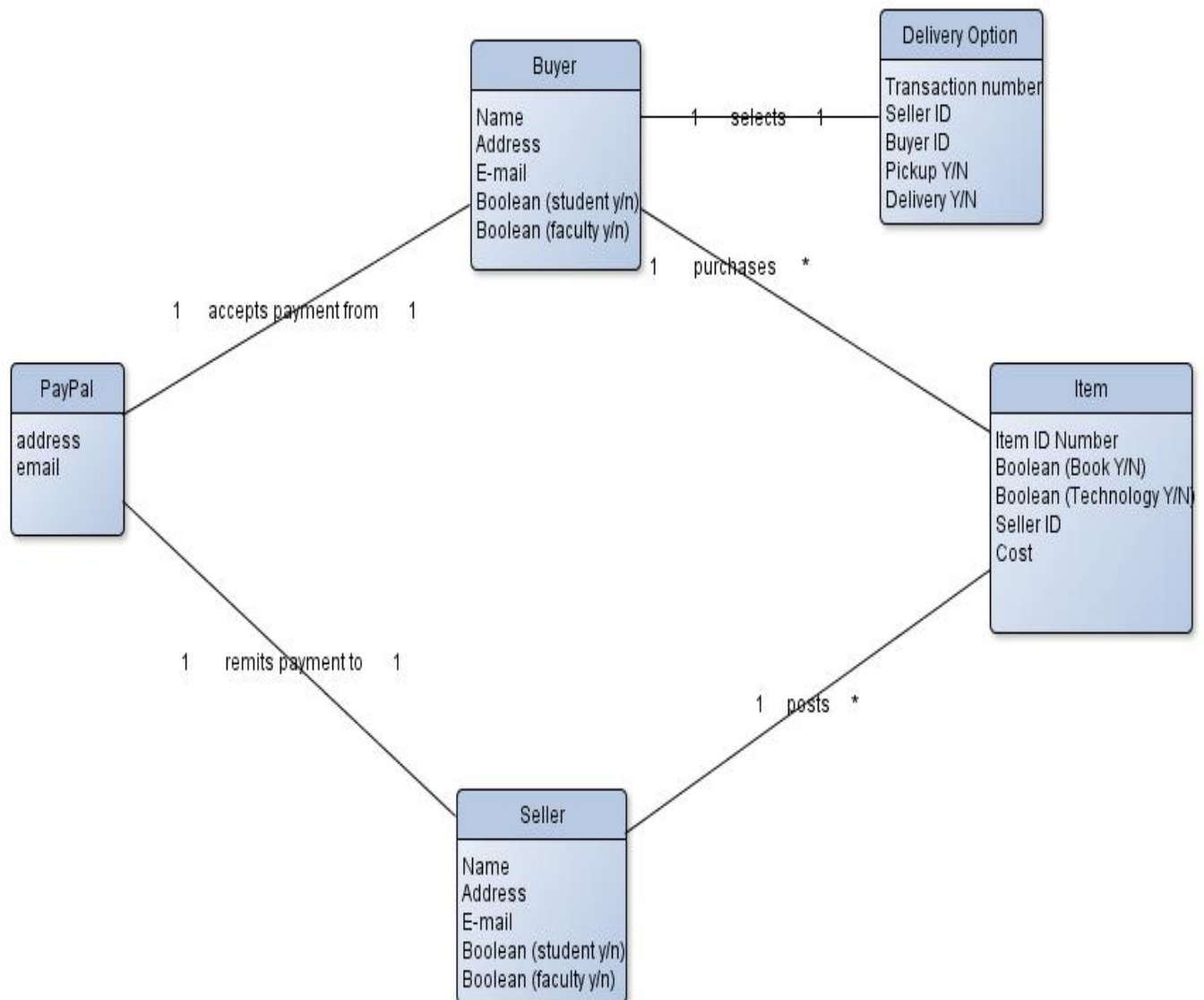


Figure 4.

4.3 Selected Class Definitions

As a group we feel that the majority of our classes can be readily learned and understood. However, if there is one class which may cause doubt it would be PayPal. Some may not be familiar with this specific e-commerce business, much less understand that it allows payments and money transfers to be made through the Internet. Further

clarification regarding PayPal banking capabilities may need to be explained to avoid further confusion.

In the context of the domain model, PayPal acts as an intermediary between Buyer and Seller. Payments are sent to and from PayPal depending upon who is making the transaction.

4.4 Selected Association Definitions

The association between Seller and Item is “Seller” posts many “Items” and many “Items” to one “Seller.” The word “post” is used to define the relationship between the two classes. For some, the association may not be clear. Theoretically, one could say Seller creates many items. However, the Seller is not physically creating the object they are selling, they are merely posting or listing the item in a database for auction.

4.5 Discussion

There are five classes in this system: Buyer, Seller, Item, PayPal, and Delivery Options. We decided that Buyer and Seller were better domains than User because it is possible for a user to be one and not the other and they have very a different functionality. We decided that transaction should not be a domain because it is a verb and because it would have to contain delivery option. We decided that it should be a class in and of itself, because PayPal is the system for processing all the transactions on dBay it is most certainly a domain class in itself.

5. Sequence Diagrams

5.1 A System Sequence Diagram that Expand the System Sequence Diagram

Shawn & Betty

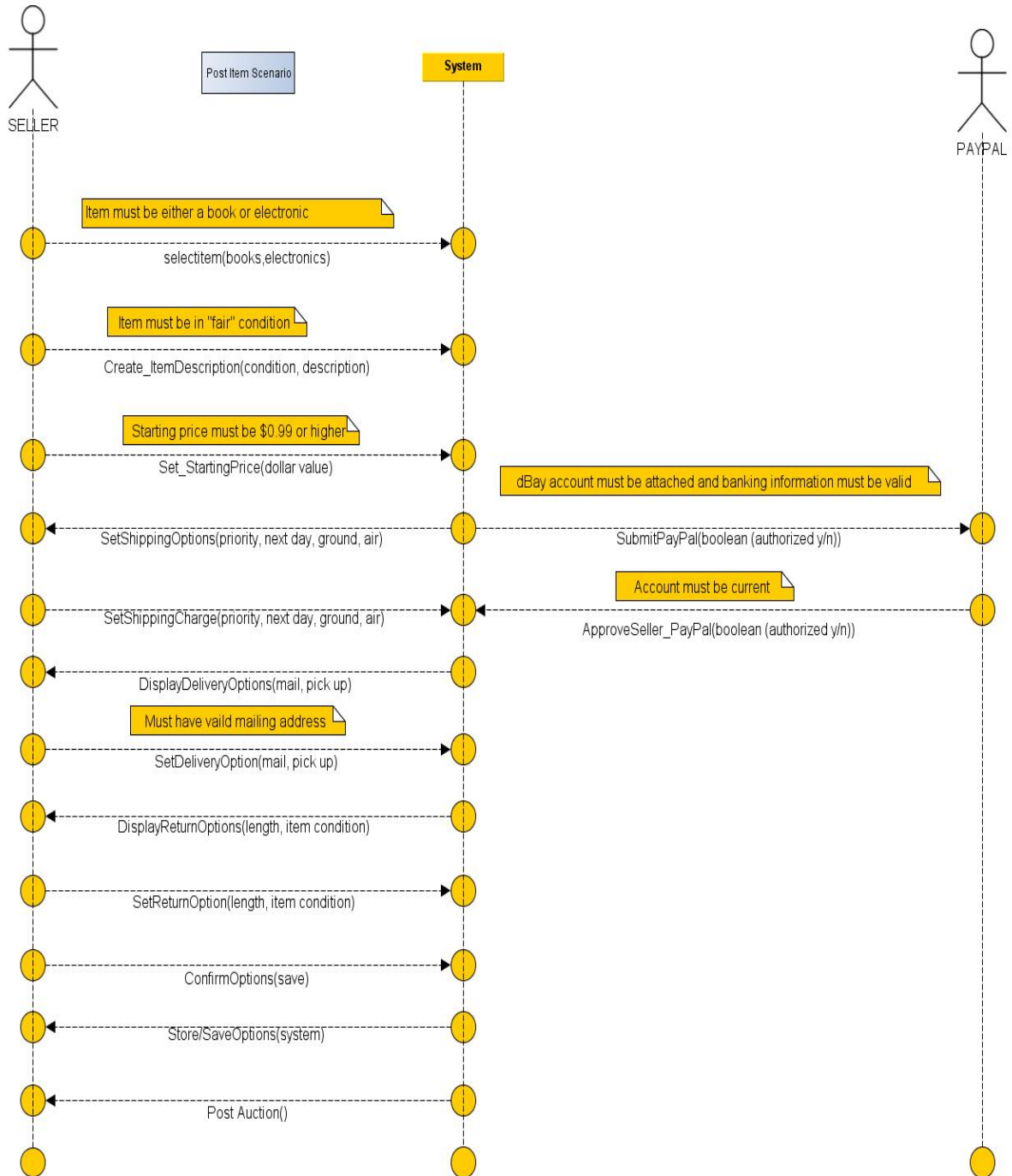


Figure 5.

Matt

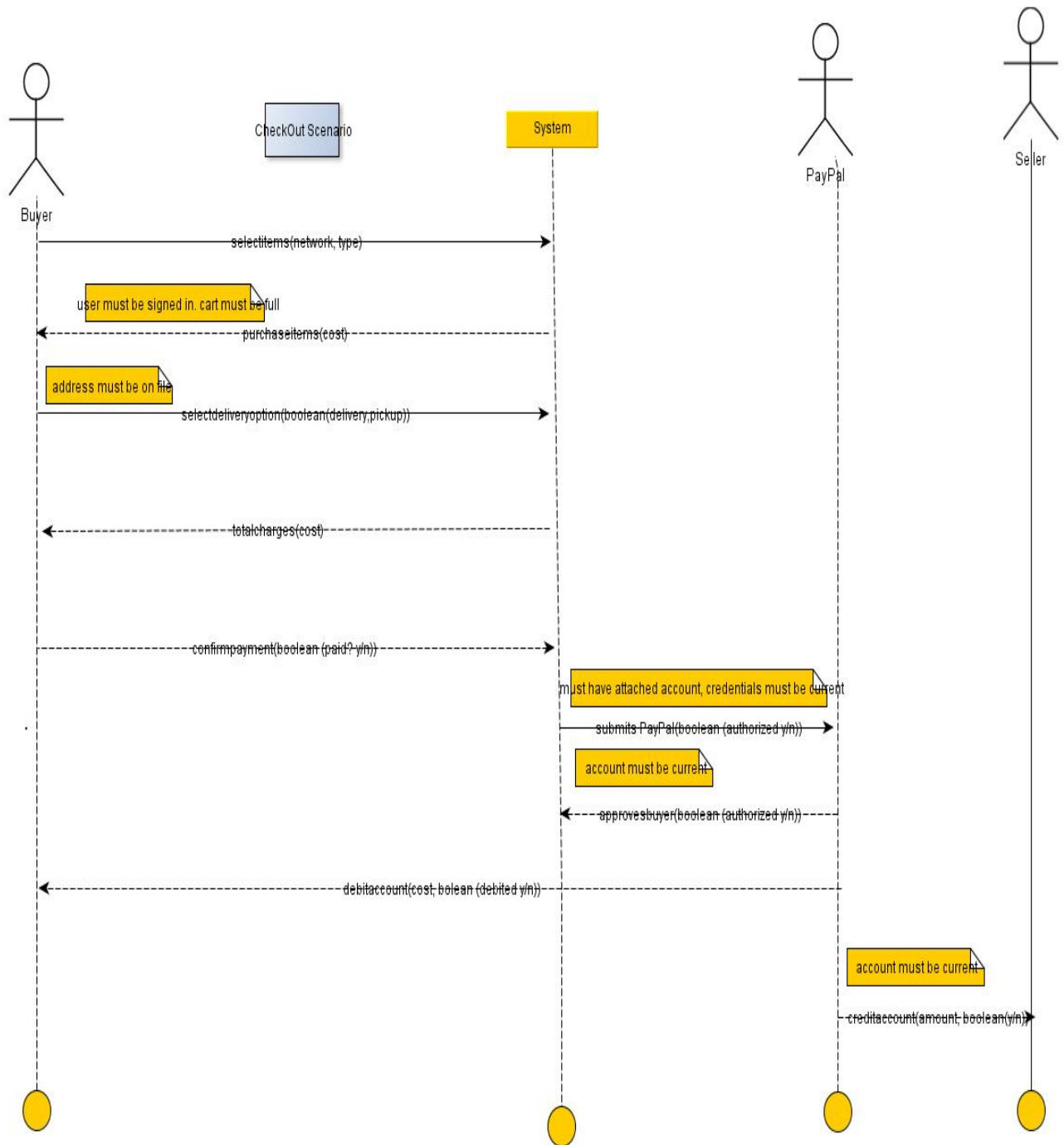


Figure 6.

5.2 A Set of System Sequence Diagrams of the Chosen Use Case

Shawn & Betty

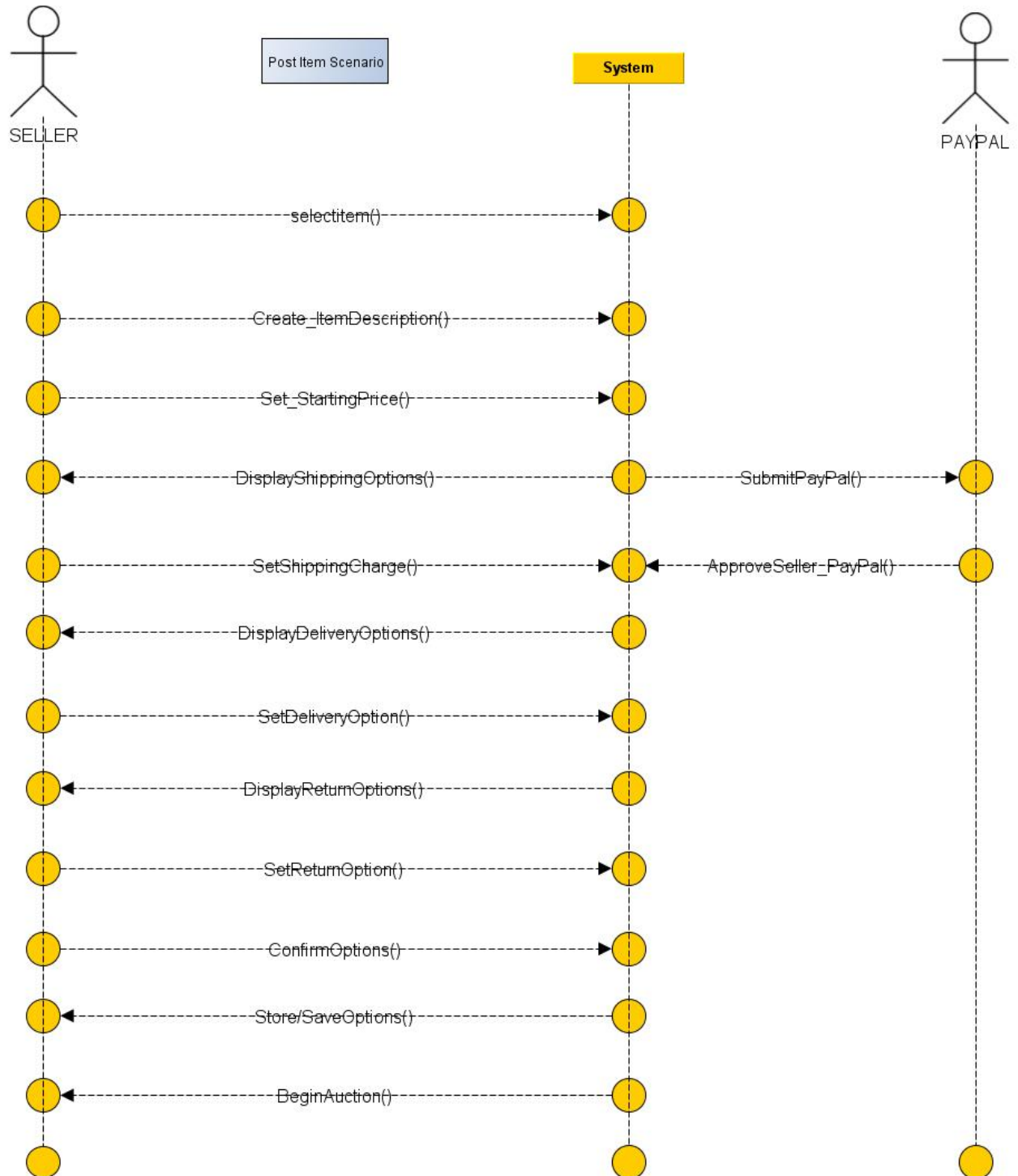


Figure 7.

Matt

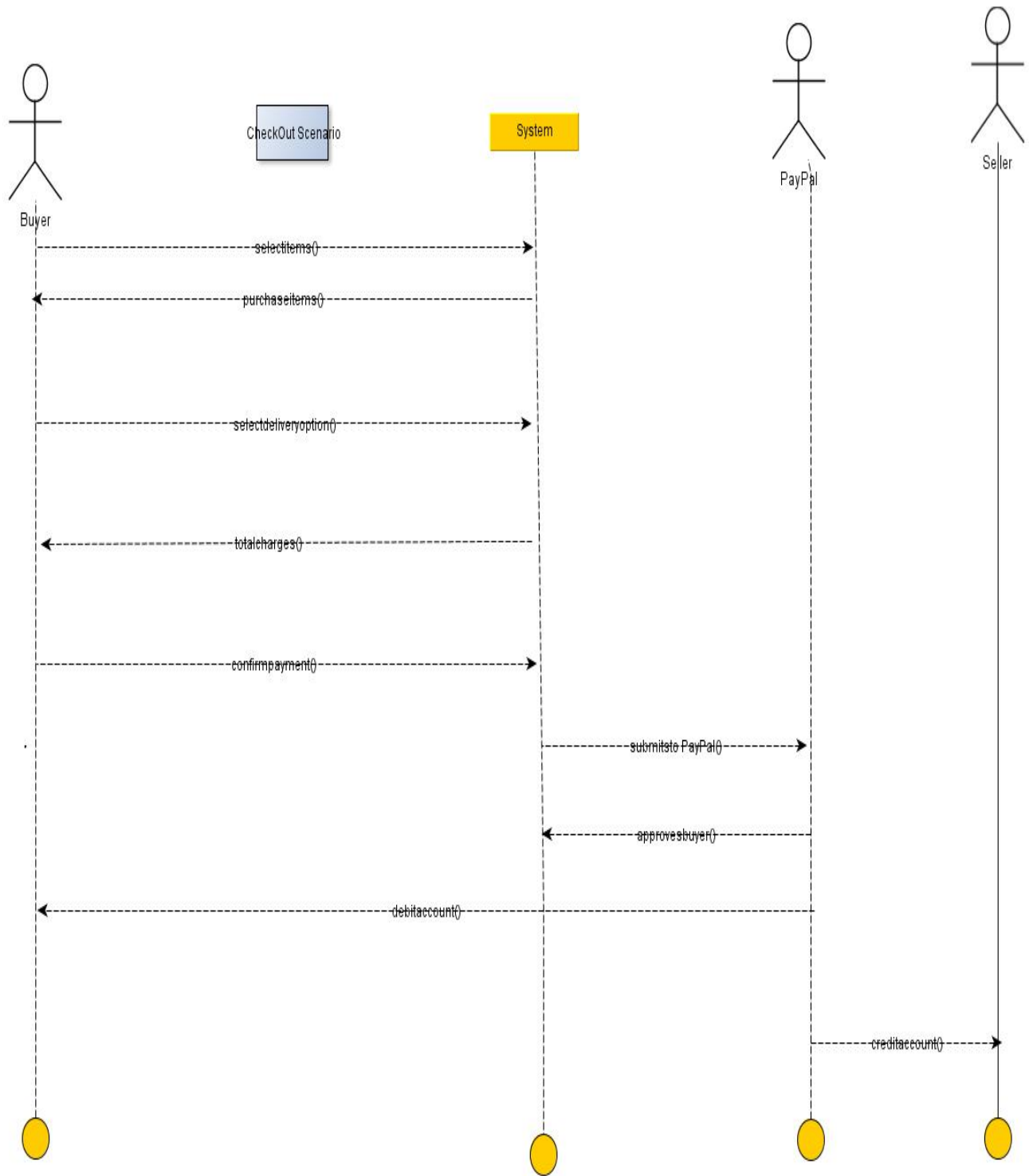


Figure 8.

5.3 Sequence Diagrams

Matt - This is the sequence model for the checkout scenario. In this scenario buyer, PayPal, seller and the system are the actors. The buyer primarily interacts with the system but the system verify the buyers' information through PayPal which submits payment to the seller.

Some additional options for this model include having the seller interact directly with the system and having the system communicating with PayPal as a separate scenario. The reason we did not like this is that we thought it added unnecessary steps and was not particularly clear.

An additional possibility would be to explore the unsuccessful scenarios. For example what if the buyer does not have the needed funds in his account? Or what if the seller had changed the checking account connected to his PayPal?

Shawn & Betty – This is the sequence diagram for the “Post Item” scenario. In this scenario seller and PayPal are the actors. Although, the buyer primarily interacts with the system, the system must verify the buyers banking information via PayPal.

Regarding the SSD – We mulled over whether or not to include “Buyer” as an actor but ultimately decided against it. The reason for this is because the buyer is not a part of the “Post Item” process, but rather a part of the general dBay scenario. Conversely, we decided to include “Buyer” as an actor in the sequence diagram because in order to pay the seller the actors account must be debited by PayPal. We also struggled to come up with attributes for some of the system transactions. For example, “Store/SaveOptions”, “ConfirmOptions(save)”, etc.

It was a challenge illustrating the order, and inter-system events that external actors generate. For this particular use case it was difficult coming up with events for the system. There were many events in which the seller interacted with the system, but maybe we were not looking at this in the proper light.

6. Design Class Diagram

6.1 Design Class Diagram with all the Attributes, Operations, and Navigability

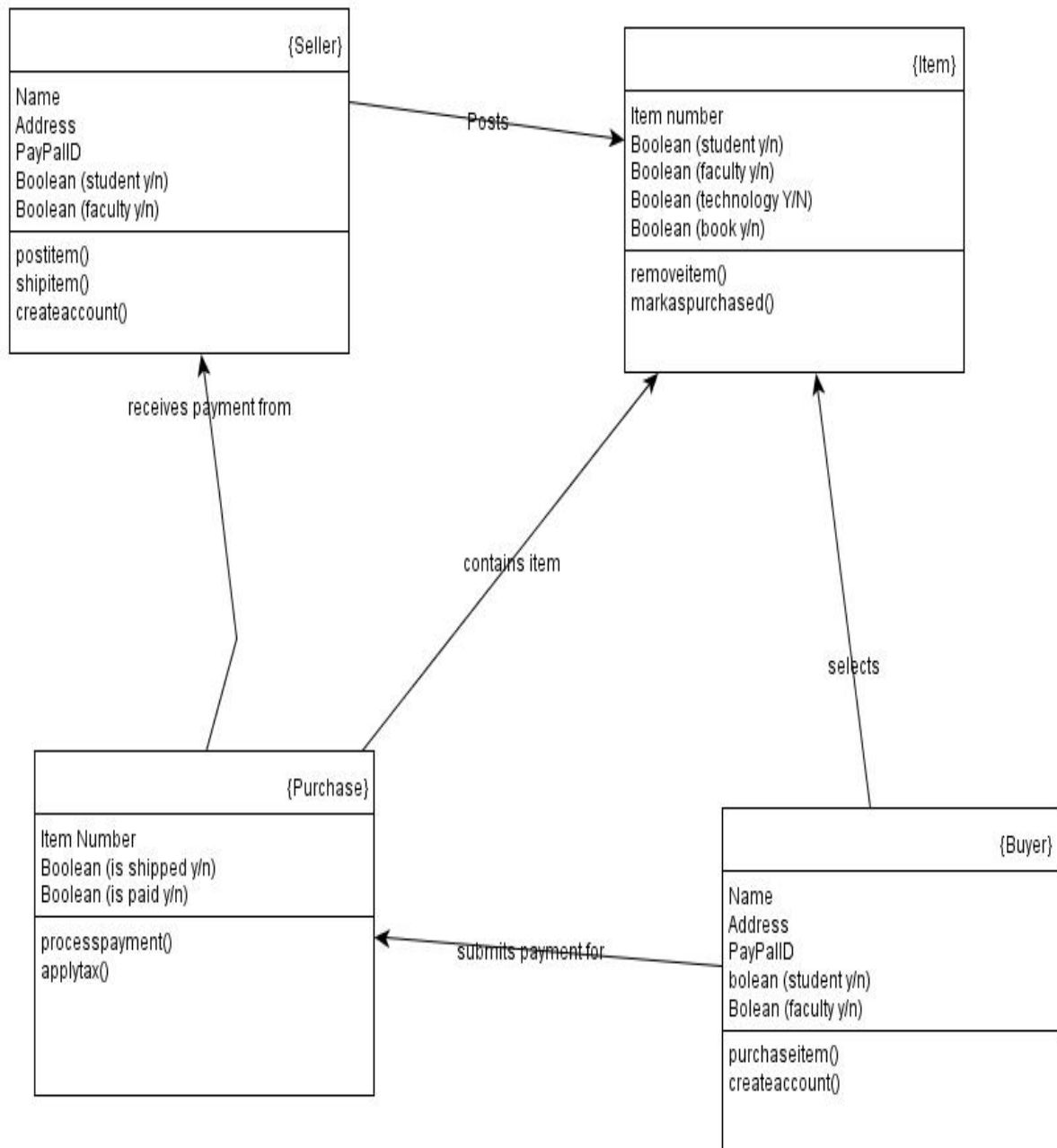


Figure 9.

6.2 Validation and Discussion

This model defines the interaction between the four classes of Buyer, Seller, Item and Purchase. In this program class of buyer selects an item, pays for the item and receives payment from the buyer via the class purchase. The buyer posts an item and receives payment from the buyer via the class purchase. Both the Buyer and Seller can create accounts. The seller can post and ship items simultaneously. The Buyer can purchase individual items.

We could have defined a single class as User and have “Buyer” and “Seller” as a Boolean option. However because it is an auction site someone can be one or the other and not have any interest in being a seller or buyer. Also, PayPal could have been its own class but we decided that the class of purchase contained the functionality of PayPal for this purchase.

There are a few possibilities that could be further explored. As I suggested PayPal could be its own class. Another option it to add a class for the system and that could extract tax and process payment either in addition to or connected to the class of purchase.

7. RDB and Extra Credit Diagrams

7.1 Relational Database Schema

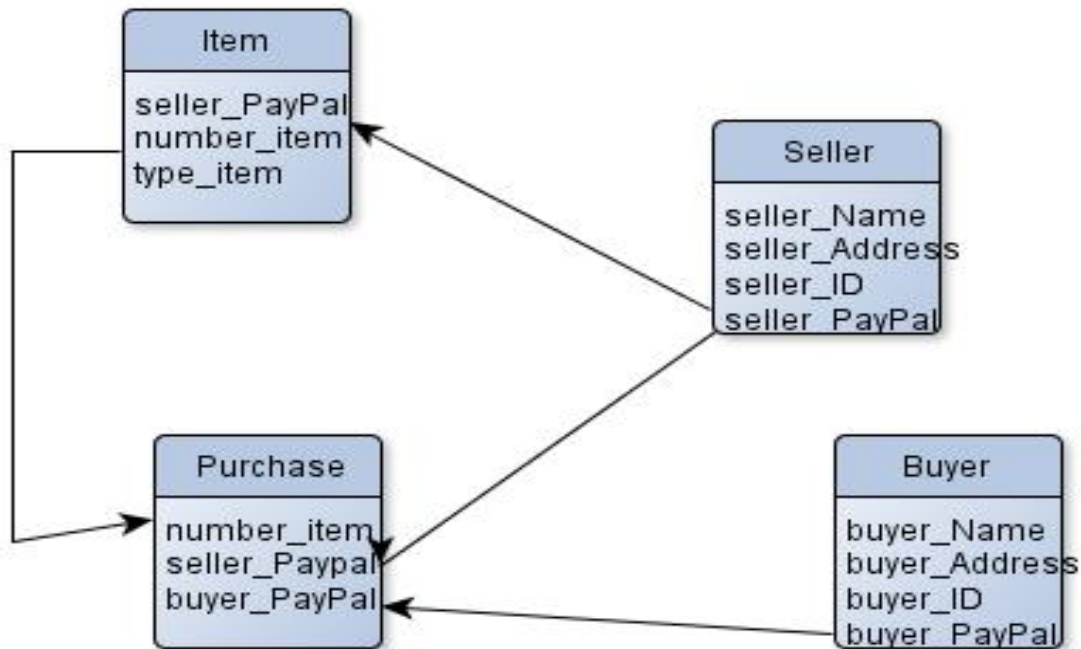


Figure 10.

7.2 Discussion

The relationships in the data relational model are based off connections that are the most simple and the most intuitive. What connects the buyer and seller is the monetary exchange through PayPal so their PayPal ID's are the keys. Likewise the item number provides the identification for the purchase which connects back to the buyer and the seller. Because the PayPal ID connects back to the buyer and seller it allows the tracking of purchase activity.

7.5 ERD—Entity Relational Diagram

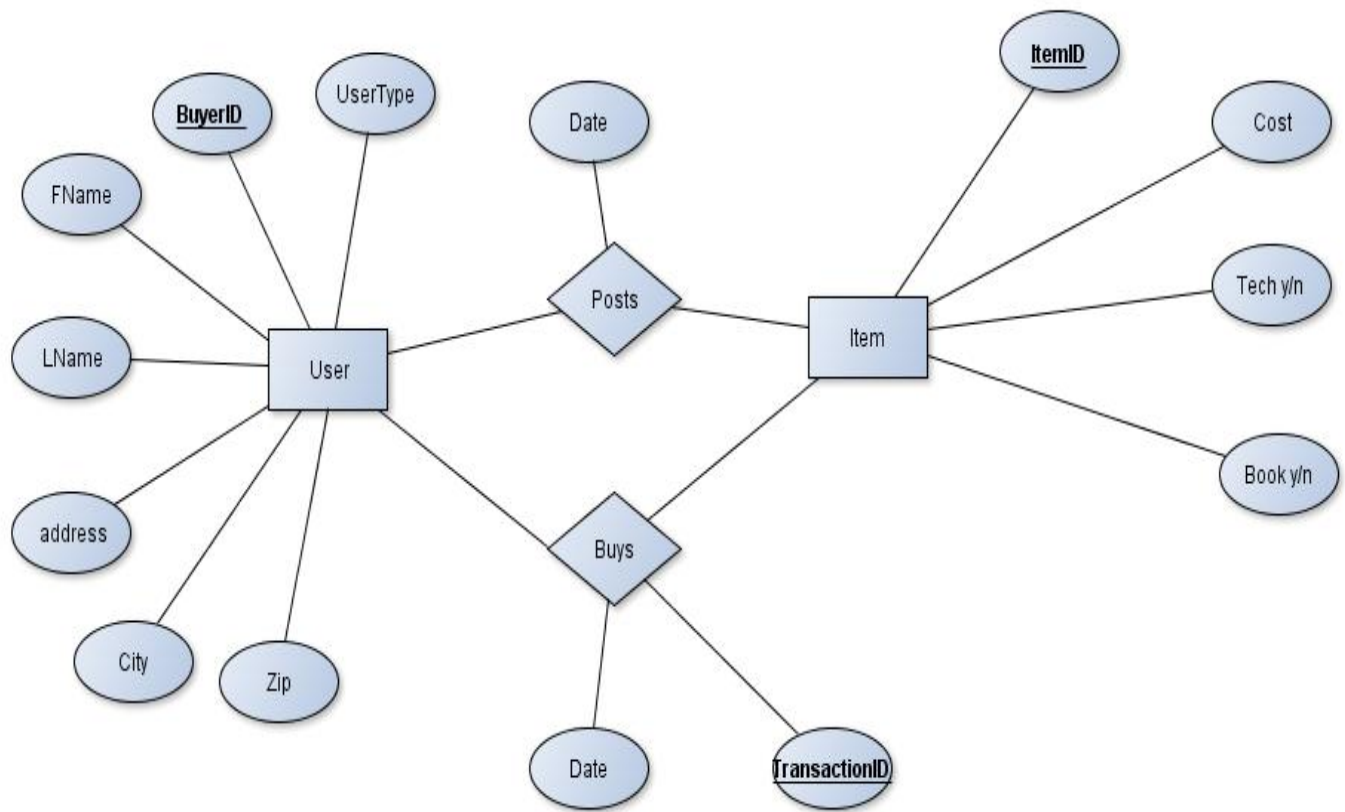


Figure 11.

8. Final Evaluations and Independent Work

8.1 Evaluations of Project

I think our modes do a good job of showing the process of creating an account posting and selling an item. The only additional scenarios to explore are how the system would handle unsuccessful scenarios such as if someone who is not at Drexel student/faculty member attempts to sign up or the item is rejected for not being an electronic item or book. As we mentioned above, unsuccessful scenarios of buyer or seller feature PayPal errors such as “out of funds” or “no connected account.”

8.2 Evaluation of UML and Tool

Learning to use Rational Rose has been quite the love/hate relationship, because it is not user friendly. Trying to learn how to draw the diagrams were tedious enough, but learning how to use the software added to this frustration. One particular process we had an issue with was trying to figure out how to draft Sequence Diagrams. Although the class material was good at explaining what Sequence Diagrams are, we needed a user guide to help us actually create the Sequence Diagram in Rational Rose. We believe video tutorials specifically designed for this course would have been extremely helpful and useful on various levels.

In addition, not just tutorial videos for Rational Rose, but perhaps even video recorded lectures of UML. This course may be one of the hardest courses for most of us. The amount of provided material was more than adequate, that said, we feel that it would have been beneficial to have someone explain it to us.

For part of our project, we used a software program called yEd to draft our Sequence Diagrams. It was a much more intuitive and much less cumbersome than Rational Rose. For instance, we were able to create lines without arrows. Although, we’re sure this is possible in Rational Rose, we were not able to figure out how to do it.

8.3 Unsolved Problems

Upon consulting our group for the last time it was apparent that we had sufficiently addressed most, if not all of our issues. That said, we could have explored our unsuccessful scenarios a bit further. However, we do not look at this as problem per se, but rather functions that were insignificant to our general use case.

8.4 Lessons Learned

Matthew Lyke

One of the most important lessons I learned in this project is the importance of iterations. We met a few times and Betty and Shawn were able to meet more frequently but with my situation we weren't able to get together very often. Communicating through email we were able to bounce suggestions back and forth of each other improving our designs incrementally and probably result in a better project than if we worked on them all together.

I am by nature a visual learner but I learned from this course that I do not connect as equally with all diagrams or models. For example the models connected to data and data analysis make sense to me I think because I understand the uses of and relationship between data intuitively. Likewise I found sequence diagrams helpful because I enjoy thinking in terms of processes. In contrast I really struggle with domains and classes which is probably evident from my diagrams!

It would be interesting to learn more about how large complex systems fit together. For example when our project would be designed there would be multiple and languages involved. I imagine a model that encompassed every step and all the code for the interface the database and the actions steps of the site would be 100's of pages. How does a team use a model like this and avoid analysis paralysis?

Betty C Nguyen

This project has taught me a variety of lessons, but the two main ones are team work and software designing is one tedious process. This is my first taste of software designing, and I did not expect the level of detail, time, and frustration. Now looking

back, I realize our project has finally come to a full circle and understand where each piece fits and why. The material in this class is quite chewy as a whole, but working on this project allowed the theories, rules, and restrictions come alive from multiple diagram drafting sessions and extensive discussion within the team.

I am usually better at interacting with peers in person, but this project as pushed me outside of my comfort zone. Shawn and I were able to meet frequently since we both live in the city. Matt lived quite a distance, and has a small child to care for, so it was difficult at first. However, we pulled through, and many times Matt chimed into our discussions via email—real time. This project has been quite intense, but Matt contributed significantly that distance was the least of our worries, and did not impact our team in any negative way.

My second lesson, I am a better visual learner than verbal learner. With that being said, the vast amount of discussions regarding why a use case should be <<include>> or <<extend>> helped built a small foundation for the various diagramming we did throughout this class. My main problem was not being able to think as a program. I can be quite wordy, and it shows. Use Cases were particularly difficult for me at first, because I failed at thinking more technical, and understanding what a human would be using the program for and how the program would respond.

One of the things I would love to learn and see next would be how the program would start its building stage after its designing stage, and what steps are required to be completed there.

Shawn A. Woodson

In the beginning our project was off to a promising start. We started this journey with four group members, a lot of ideas and several question marks. However, our anxieties were eased once we met for the first time as a group. We managed to hash out individual responsibilities and we also created a project documentation guideline, a document that has been instrumental in keeping the group on track. As you know, we lost a group member 4 weeks into the course and this halted our momentum for a few weeks. The responsibilities that were ironed out during our initial meeting had to be

revised and reassigned. Furthermore, we realized that Matt would not be available to meet in person as often we would have liked. That said, we persevered and managed to organize ourselves into functioning unit. Betty and I met regularly the last half of the quarter and Matt emailed us several times throughout the weeks with different iterations of our diagrams and models.

As Matt mentioned above – the importance of iteration in regards to UML cannot be overlooked. This was especially true when it came to building our (long form) use case diagrams. I failed to take to recognize that each use case is a complete scenario, and an <<extended>> use case is not a subset of the base case. It is a complete scenario in its own right. Therefore the entire point of the <<extended>> use cases is to list the functionality that makes the use case different from the base case. I also had trouble differentiating between a conceptual class and the attributes of said concept. When I took Larman's rule that attributes should be viewed as numbers or text into account then everything made sense.

Last – in looking at the UML Roadmap I would be interested to learn how the other models build off of one another. I find that diagram to be very interesting. The fact that everything originates from the problem statement and use case diagrams is truly fascinating. It would have been interesting to explore the right side of the diagram (the implementation phase), and how the persistent classes help build the diagrams and schemas on that side of the equation.

8.5 Peer Evaluation

Project Peer Evaluation Form (PEF) of Team Members for INFO 620

Term: Summer 2013

Project Title: dBay Analysis and Design

Your name: Matt Lyke

Other Team Members: TM-1=Betty Nguyen, TM-2=Shawn Woodson

Who's Responsible?

This form needs to be submitted for all group project members. At the submission of each team project deliverable, every team member evaluates himself/herself and each fellow team member. Peer evaluations could affect your grade, so it is in your best interest to ensure that these evaluations are conducted fairly and submitted to the professor at the time you submit your project report. If an individual has taken on extra responsibility, such as organizing the group, or editing submissions for consistency, this should be indicated by group members and extra credit will be assigned to that person.

Each team member must complete a peer evaluation form. The peer reviews will be used to identify team problems and, in cases where an individual or group collaboration was uneven, to apportion the group grade. These forms can be included in the project package or, if you wish to treat the form as confidential, please feel free to email it to me.

The form is mandatory for all team members. It must be submitted within 24 hrs after the project documentation is submitted. Without this form, your project grade will be zero.

Peer Evaluation Form

Your peer evaluation form should adhere to the following format. Add more rows if necessary.

PERCENTAGE OF 100

Collaboration Items	Your Name	Other Team Member(TM)Name 1	Other Team Member(TM)Name 2
Proposal Development	33	33	33
Group leadership or ability to plan project and help keep team on track	33	33	33
Contribution over and above the call of duty, such as: willingness to help other team members, or creating an atmosphere/being a pleasure to work with.	20	40	40
Quality and timeliness of individual work submitted towards assignment	33	33	33
Revising and developing the final problem statement and	20	40	40

requirements			
Leadership on finalizing the use case diagram	20	40	40
Documentation on the use case diagram such as actor definition, use case overview section, and discussion on use case diagram	20	40	40
Leadership on finalizing the class diagram	60	20	20
Documentation on class diagram such as class definition, association definition, and discussion on the finalized class diagram	60	20	20
Leadership on collecting sequence diagrams	20	40	40
Design class diagram and documentation on the diagram	60	20	20
Creating the relational DB schema	60	20	20
Evaluation and Summary Sections	33	33	33
Organizing the final report, editing for consistency and formatting, creating Contents page	20	40	40
<i>(any other piece of work you want to bring out to my attention; add more rows if necessary)</i>	Your score	TM1's Score	TM2's Score
Final Percentage of effort in completing the project	33	33	33
Average	35	35	35

Individual scores on each criterion should range from 1 (*very poor*) to 5 (*outstanding*). Generally speaking, 1's should be as rare as 5's, unless team members are exceptionally bad or good. Provide a score for each team member (including yourself) on each criterion, and provide comments below to justify any very low or very high scores. Also compute an average score for each team member.

Mandatory: ***Overall comments on your team members (Write qualitative comments on your collaboration and any comments you missed above):*** You **must** write a paragraph here about the overall evaluation **and** any qualitative comments you want to pass me. (Without this section, the validity of your scores are questionable.)

Project Peer Evaluation Form (PEF) of Team Members for INFO 620

Term: Summer 2013

Project Title: dBay Analysis and Design

Your name: Betty C Nguyen

Other Team Members: TM-1=Matt Lyke, TM-2=Shawn Woodson

Who's Responsible?

This form needs to be submitted for all group project members. At the submission of each team project deliverable, every team member evaluates himself/herself and each fellow team member. Peer evaluations could affect your grade, so it is in your best interest to ensure that these evaluations are conducted fairly and submitted to the professor at the time you submit your project report. If an individual has taken on extra responsibility, such as organizing the group, or editing submissions for consistency, this should be indicated by group members and extra credit will be assigned to that person.

Each team member must complete a peer evaluation form. The peer reviews will be used to identify team problems and, in cases where an individual or group collaboration was uneven, to apportion the group grade. These forms can be included in the project package or, if you wish to treat the form as confidential, please feel free to email it to me.

The form is mandatory for all team members. It must be submitted within 24 hrs after the project documentation is submitted. Without this form, your project grade will be zero.

Peer Evaluation Form

Your peer evaluation form should adhere to the following format. Add more rows if necessary.

PERCENTAGE OF 100

Collaboration Items	Your Name	Other Team Member(TM)Name 1	Other Team Member(TM)Name 2
Proposal Development	33	33	33
Group leadership or ability to plan project and help keep team on track	33	33	33
Contribution over and above the call of duty, such as: willingness to help other team members, or creating an atmosphere/being a pleasure to work with.	20	40	40
Quality and timeliness of individual work submitted towards assignment	33	33	33
Revising and developing the final problem statement and requirements	20	40	40

Leadership on finalizing the use case diagram	20	40	40
Documentation on the use case diagram such as actor definition, use case overview section, and discussion on use case diagram	20	40	40
Leadership on finalizing the class diagram	60	20	20
Documentation on class diagram such as class definition, association definition, and discussion on the finalized class diagram	60	20	20
Leadership on collecting sequence diagrams	20	40	40
Design class diagram and documentation on the diagram	60	20	20
Creating the relational DB schema	60	20	20
Evaluation and Summary Sections	33	33	33
Organizing the final report, editing for consistency and formatting, creating Contents page	20	40	40
<i>(any other piece of work you want to bring out to my attention; add more rows if necessary)</i>	Your score	TM1's Score	TM2's Score
Final Percentage of effort in completing the project	33	33	33
Average	35	35	35

Individual scores on each criterion should range from 1 (*very poor*) to 5 (*outstanding*). Generally speaking, 1's should be as rare as 5's, unless team members are exceptionally bad or good. Provide a score for each team member (including yourself) on each criterion, and provide comments below to justify any very low or very high scores. Also compute an average score for each team member.

Mandatory: *Overall comments on your team members (Write qualitative comments on your collaboration and any comments you missed above):* You **must** write a paragraph here about the overall evaluation **and** any qualitative comments you want to pass me. (Without this section, the validity of your scores are questionable.)

Project Peer Evaluation Form (PEF) of Team Members for INFO 620

Term: Summer 2013

Project Title: dBay Analysis and Design

Your name: Shawn Woodson

Other Team Members: TM-1=Betty Nguyen, TM-2=Shawn Woodson

Who's Responsible?

This form needs to be submitted for all group project members. At the submission of each team project deliverable, every team member evaluates himself/herself and each fellow team member. Peer evaluations could affect your grade, so it is in your best interest to ensure that these evaluations are conducted fairly and submitted to the professor at the time you submit your project report. If an individual has taken on extra responsibility, such as organizing the group, or editing submissions for consistency, this should be indicated by group members and extra credit will be assigned to that person.

Each team member must complete a peer evaluation form. The peer reviews will be used to identify team problems and, in cases where an individual or group collaboration was uneven, to apportion the group grade. These forms can be included in the project package or, if you wish to treat the form as confidential, please feel free to email it to me.

The form is mandatory for all team members. It must be submitted within 24 hrs after the project documentation is submitted. Without this form, your project grade will be zero.

Peer Evaluation Form

Your peer evaluation form should adhere to the following format. Add more rows if necessary.

PERCENTAGE OF 100

Collaboration Items	Your Name	Other Team Member(TM)Name 1	Other Team Member(TM)Name 2
Proposal Development	33	33	33
Group leadership or ability to plan project and help keep team on track	33	33	33
Contribution over and above the call of duty, such as: willingness to help other team members, or creating an atmosphere/being a pleasure to work with.	20	40	40
Quality and timeliness of individual work submitted towards assignment	33	33	33
Revising and developing the final problem statement and requirements	20	40	40

Leadership on finalizing the use case diagram	20	40	40
Documentation on the use case diagram such as actor definition, use case overview section, and discussion on use case diagram	20	40	40
Leadership on finalizing the class diagram	60	20	20
Documentation on class diagram such as class definition, association definition, and discussion on the finalized class diagram	60	20	20
Leadership on collecting sequence diagrams	20	40	40
Design class diagram and documentation on the diagram	60	20	20
Creating the relational DB schema	60	20	20
Evaluation and Summary Sections	33	33	33
Organizing the final report, editing for consistency and formatting, creating Contents page	20	40	40
<i>(any other piece of work you want to bring out to my attention; add more rows if necessary)</i>	Your score	TM1's Score	TM2's Score
Final Percentage of effort in completing the project	33	33	33
Average	35	35	35

Individual scores on each criterion should range from 1 (*very poor*) to 5 (*outstanding*). Generally speaking, 1's should be as rare as 5's, unless team members are exceptionally bad or good. Provide a score for each team member (including yourself) on each criterion, and provide comments below to justify any very low or very high scores. Also compute an average score for each team member.

Mandatory: ***Overall comments on your team members (Write qualitative comments on your collaboration and any comments you missed above):*** You **must** write a paragraph here about the overall evaluation **and** any qualitative comments you want to pass me. (Without this section, the validity of your scores are questionable.)

Appendix

Figure 1.....	7
Figure 2.....	8
Figure 3.....	9
Figure 4.....	25
Figure 5.....	27
Figure 6.....	28
Figure 7.....	29
Figure 8.....	30
Figure 9.....	32
Figure 10.....	34
Figure 11.....	35

References

Larman, C., *Applying UML and Patterns*, 3rd edition, Prentice Hall, 2005.

Yunan Chen and Il-Yeol Song, "Guidelines for Developing Quality Use Case Descriptions," in the *Proc. of 2007 IRMA International Conference*. **(UCD paper)**

Il-Yeol Song, Kurt Yano, Juan Trujillo, and Sergio Luján-Mora. "A Taxonomic Class Modeling Methodology for Object-Oriented Analysis," In *Information Modeling Methods and Methodologies*, Advanced Topics in Databases Series, Ed. (J Krostige, T. Halpin, K. Siau), Idea Group Publishing, 2004, pp. 216-240. **(TCM paper)**

Il-Yeol Song. "Developing Sequence Diagrams in UML," in *Proc. of 20th International Conference on Conceptual Modeling (ER2001)*, (Lecture Notes in Computer Science, Vol. 2224), Yokohama, Japan, Nov. 27-30, 2001, pp. 368-382. **(SQD2001 paper)**