2/25/2016

Kerr and Brown Shoe Shop

Software Design (F28SD2) Coursework 2016

Ronan Smith, rs6@hw.ac.uk, H00189534

Kerr and Brown Shoe Shop

Software Design (F28SD2) Coursework 2016

D1 – Assumptions……

Assumptions made about the informal requirements include a small customer base as the requirements mention the shop operates in a small country town. The requirements also say that it can be extended though, to deal with future growth for the company. Another assumption is that all users with an account can use the system. I have not described any help systems or help pages within my diagrams. It is also assumed that the company already owns all the necessary hardware needed such as servers and computer monitors in-store as well as a way to access the internet.

D2 – Design Models…….

Functional requirements are shown in figure 1 and the non-functional requirements are shown in figure 2.

ID	Details	Type	Priority
R1	The system shall display a list of all products offered by the shoe shop.	Products	M
R2	The system shall list all products by category	Products	M
R3	The system shall display detailed product descriptions on demand	Products	M
	including a name, photograph, price and description for each product.		
R4	The system shall have a shopping basket or equivalent.	Products	M
R5	The shopping basket shall display the number of items inside it.	Products	C
R6	The system shall allow customers to buy/order products.	Orders	M
R7	The system shall allow staff in the store to view detailed information on	Products	M
	sales and stock.		
R8	The system shall warn the staff when certain stock is low or if a certain	Products	C
	item is out of stock.		
R9	The system shall allow the store staff to order stock from the	Products	M
	workshop.		
R10	The system shall allow customers to bring back shoes for repair	Products	M

R11	Shall allow the staff in store to process any orders made online, over	Products	M
	the phone or by post.		
R12	Shall allow for products to be returned and the information on the return	Products	M
	to be processed.		
R13	The system shall give recommendations to customers based on their	Interface	W
	past purchases.		
R14	The system shall allow staff to reduce the prices of certain product lines	Products	M
	during times of sales.		
R15	The system shall allow the user to enter their credit/debit card	Payment	M
	details securely for payment online.		
R16	The system shall allow for staff to enter any credit/debit card details	Payment	M
	into the system for orders placed over the phone or by mail.		

Figure 1 – Functional Requirements (T1.1)

	Non - functional requirements		
ID	**Details**	**Type**	**Priority**
R30	The system shall support concurrent sessions.	Capacity	M
R31	The system shall store sales transaction data.	Availability	M
R32	The system shall use a browser as it's	Compliance to	M
	user interface.	standards.	
R33	The system shall be able to work over many	Availability	M
	different browsers/platforms.		
R34	The system should be able to store customer	Security	M
	information - including card details - safely and		
	securely.		
R35	The system shall work quickly with all important	Performance	S
	information no more than 3 mouse-clicks away.		
R36	The system shall be available 24/7	Availability	M
R37	The system shall be able to support a fairly small	Maintainability	S
	number of users but with the ability to grow.		

Figure 2 – Non-functional requirements (T1.1)

Next is the use case diagram, which I created using an online tool. It contains 14 different use cases and various connections between them. For example, the 'Make Order' case is an extension from the 'View Basket' and 'View Items' case to allow the user to make an order from each of these cases at the click of a button. It also contains four actors; Customer, Staff, Office and Workshop. These actors are all mentioned in the coursework specification.

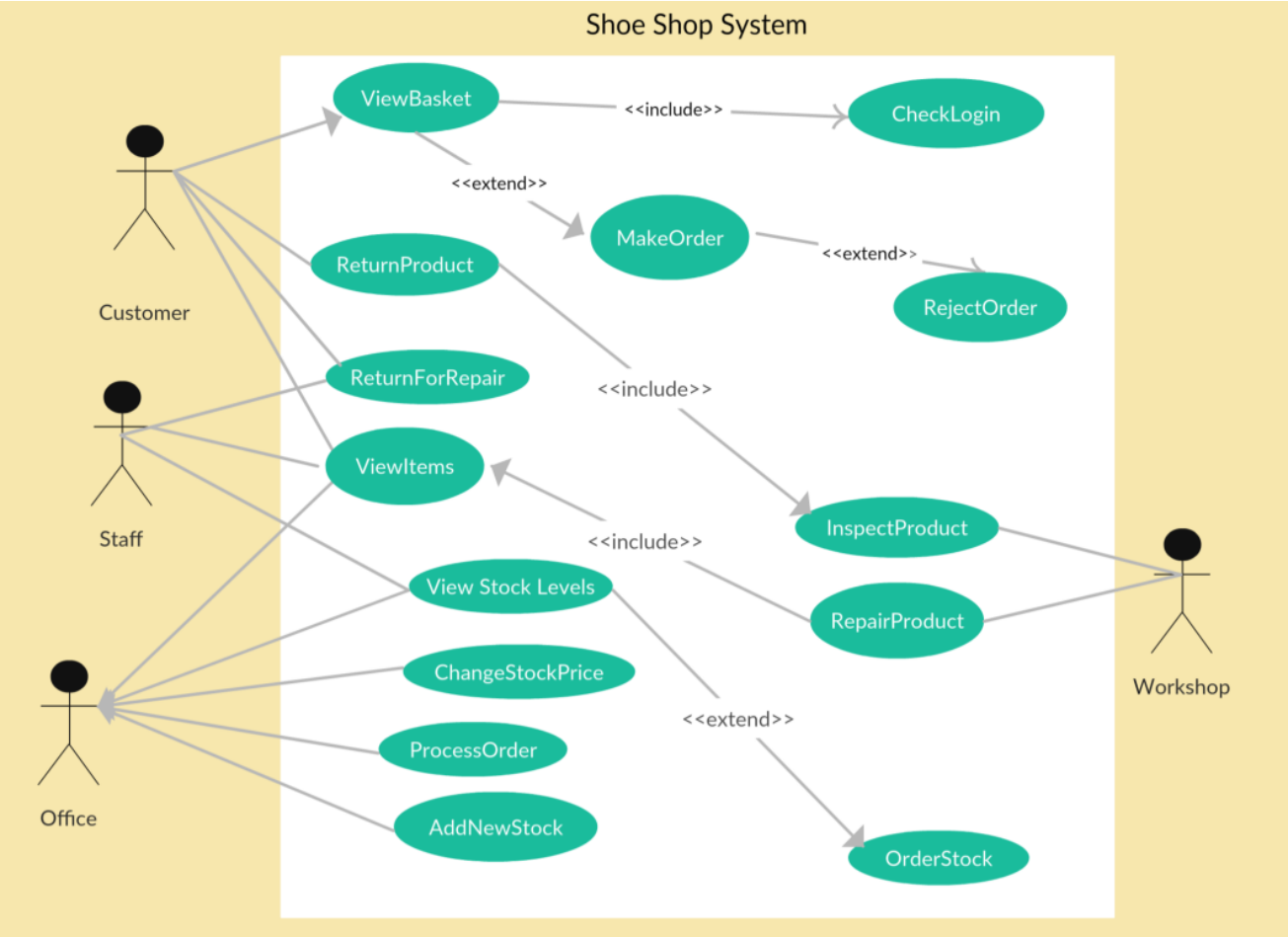

Figure 3 – (T1.2) Use Case Diagram for the shoe shop. Implemented using creately.com

The textual descriptions are attached at the end of this report.

On the next page is my final class diagram, which was created using Microsoft Word. It contains a variety of different objects and many of the attributes were specified in the coursework requirements. I had to create some extra attributes and objects to help it to work better for the shoe shop.

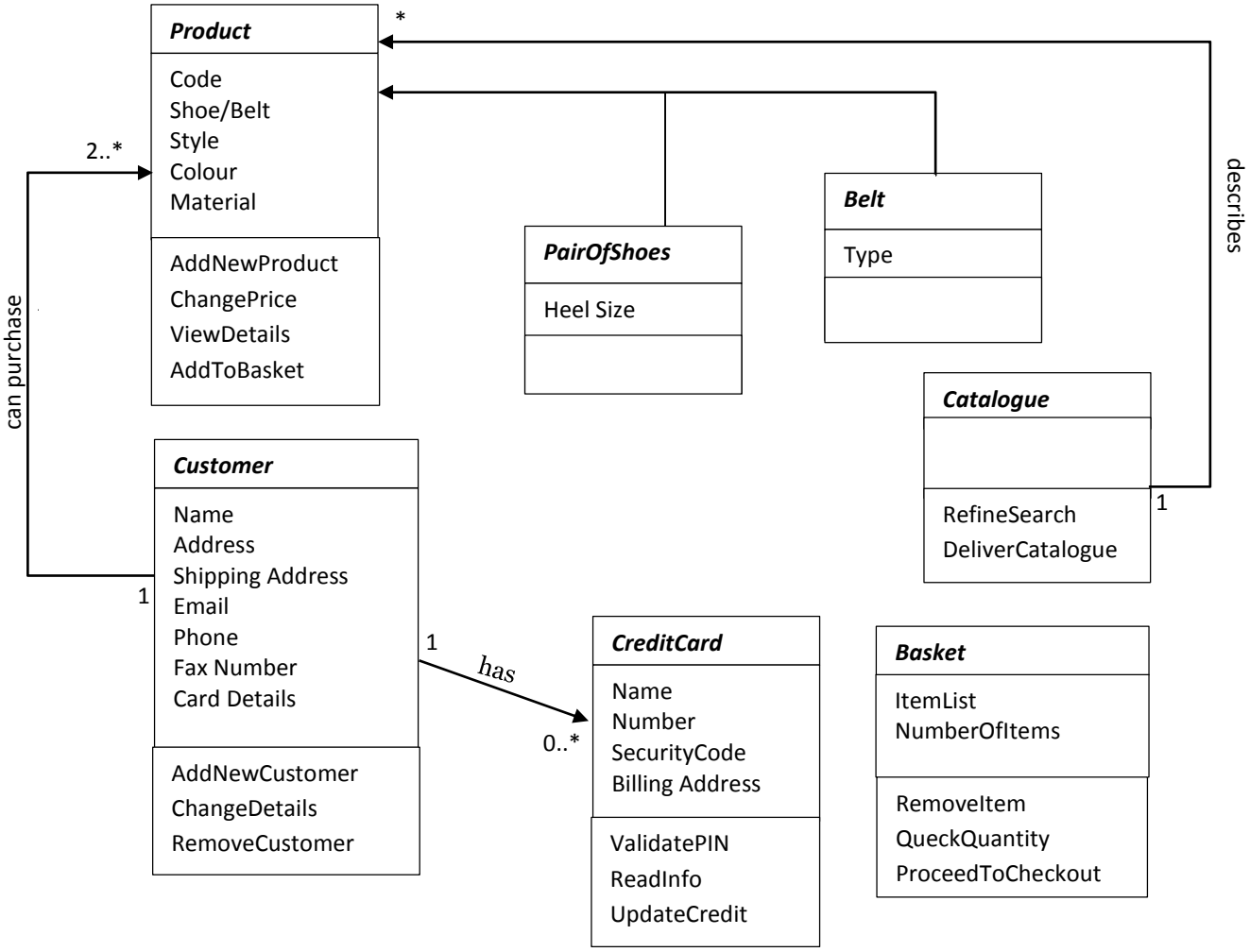

Figure 4 - Class Diagram (T2), used to model the static aspects of the subsystem

For T3 I decided to use the 'ViewBasket' use case as this is the only one which contains both an <<include>> and an <<extend>> relationship. The activity diagram for 'ViewBasket' is shown in figure 5 on the next page. To implement this, I also used Microsoft Word.

As described in the textual description the use case starts when the Customer opens the basket. Using swim-lanes I have represented how the system checks the users' identity and asks for a user name and password if they have not already entered it. Apart from the 'Check Login' activity, all activities are carried out by the customer.

The fork and join in the diagram shows that the user can 'View Products' and 'Add/Remove Items' at the same time but they both must be completed before moving on to the next stage (i.e a customer cannot make a purchase without first knowing what is in their basket or without adding something to it). Although we were only required to create one activity diagram for this coursework, I have used the following symbol to represent an activity that would have been represented in another diagram: ⏚

For example, the 'Proceed to Checkout' activity which would deal with making an order/purchase and any problems which might arise such as an item being out of stock. This diagram is shown below.

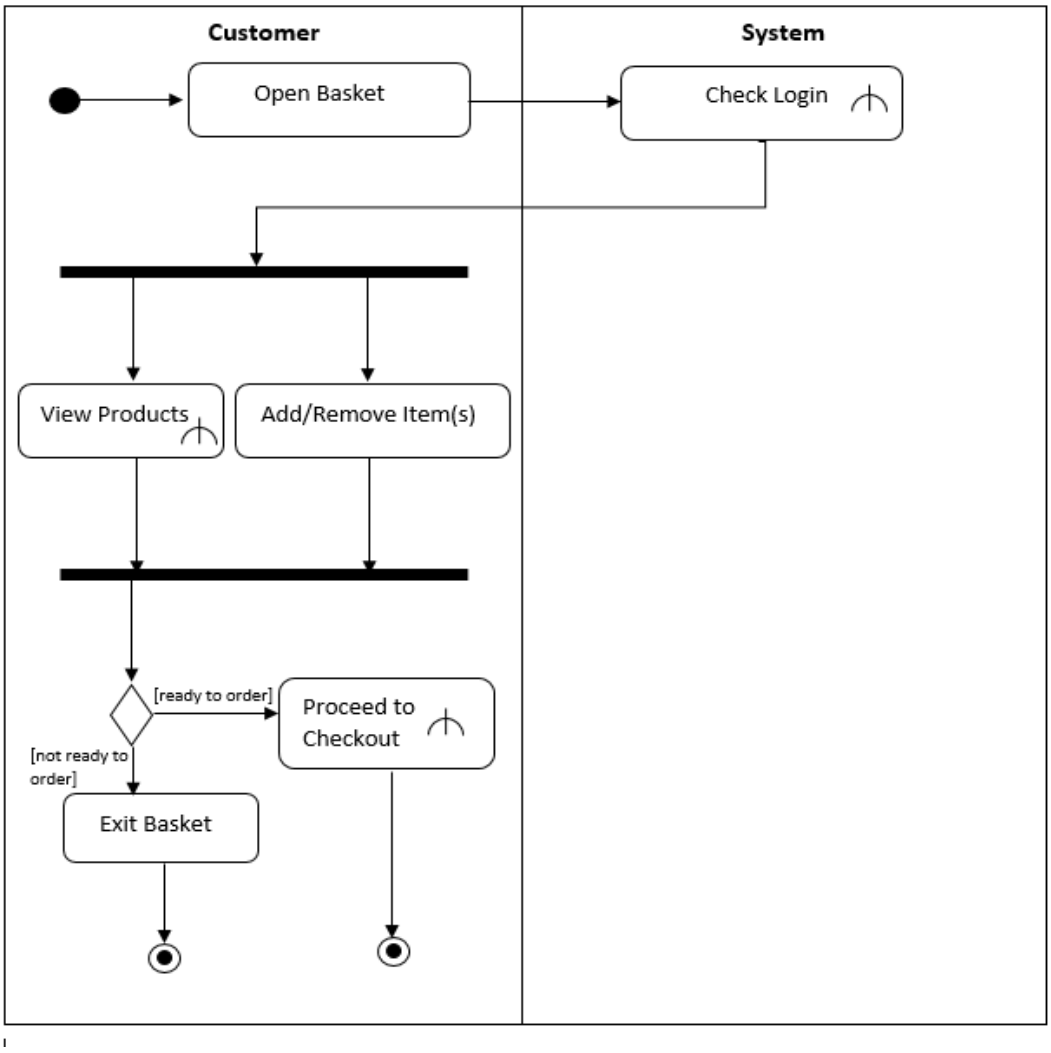

Figure 5 – Activity Diagram (T3.1) for the 'ViewBasket' use case.

The sequence diagram is shown in figure 6. Again, I used Microsoft Word to create this and I used an online resource, Stack Overflow, to find a way to represent the if… else condition. A user can only access any of the information in the basket of they are logged in and the 'else' part is only ever entered if the user is denied access, for example if they enter an incorrect password. This is the full link to the Stack Overflow page:

http://stackoverflow.com/questions/8114770/how-to-show-if-condition-on-a-sequence-diagram

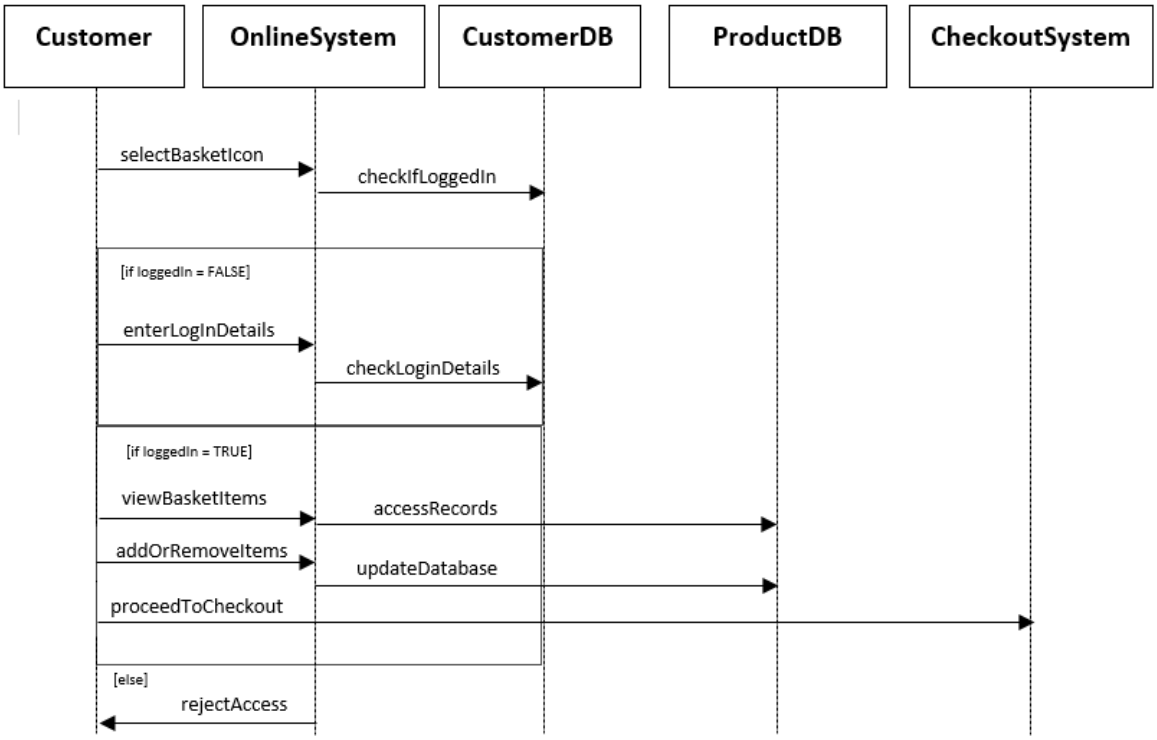

Figure 6 – Sequence Diagram (T3.2) for the 'ViewBasket' use case.

My final diagram, the state machine diagram, is shown in figure 7 and consists of 4 states:

- Guest Browse – A state of limited functionality for users who are not logged in/don't have an account.
- Authentication – A screen which asks for the users' username and password.
- View Items – A state where the user can view all items in their basket and add or remove items also.
- Rejection – After 3 failed username/password attempts the user is rejected.

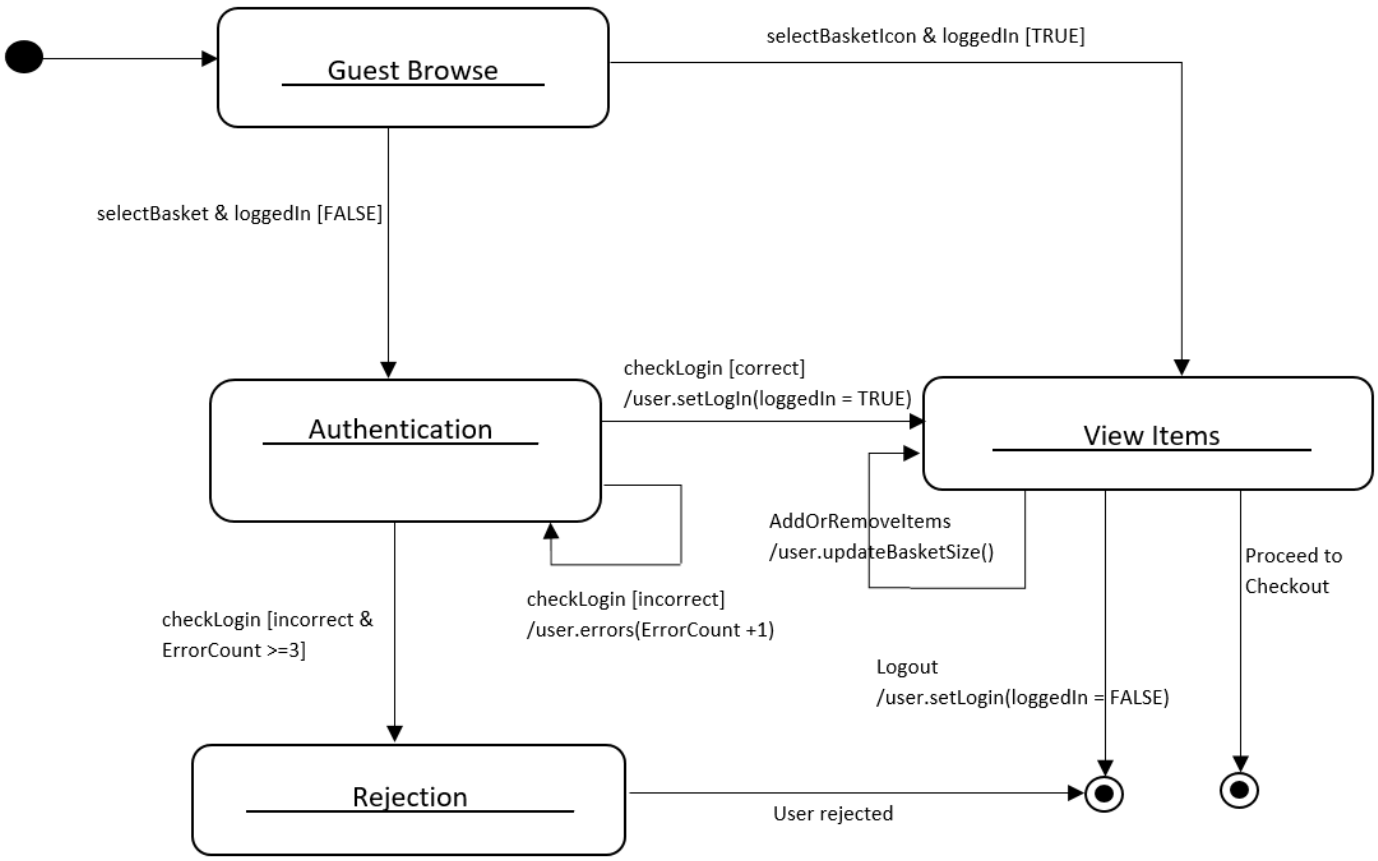

Figure 7 – State Machine Diagram (T3.3) for 'ViewBasket' use case

To create the diagram in figure 7 I used Microsoft Word and I used the ATM example from lecture 12 as a reference to get started as both mines and the one in the slides are examples where authorization is needed to access the system.

If the user is browsing the site and has already logged in at an earlier stage, then they can access the basket directly but if they are not already logged in then they must go through the authorization state first. Furthermore, there are two final states. One of which represents the user being logged out of the system and one which allows the user to continue to the 'checkout' where they can make orders. To proceed to checkout and make orders the user is required to have an account and be logged in.

D3 – Strengths and Limitations……

One of the main difficulties I encountered was representing that the user must purchase two or more products as is specified in the coursework requirements. I did manage to do this in the class diagram but didn't feel like the other diagrams allowed me to do so. Secondly, there was the problem of signifying the difference between what would happen if a user was already logged in and if they weren't. For example, if a user tried to access account information, their basket or some other area of the site that is limited to users who have an account, the system would automatically work out if they were already logged in or not, and if not, it would prompt them to enter their username and password. I feel that I got this idea across in the state machine diagram. I had to use an if… else condition in the sequence diagram to show this but to my knowledge this was not covered in the lectures. Since the sequence diagram, activity diagram and state machine diagram all had to focus on the same use case, I felt this made it more difficult as the specific use case is suited more to some diagrams than others. For example, here I felt that the activity diagram didn't convey much useful information that wouldn't have been in the other two diagrams. Due to the fact that they are all based on 'ViewBasket', none of the actors are really represented in this report much apart from 'Customer'. One more limitation of these diagrams is that many of them do not make it easy to represent real-life objects such as the Shoe Shop's catalogue. The only way I was able to represent this was in the class diagram, which is of course, designed for representing objects.

For the 'ViewBasket' use case in particular, I feel the state machine diagram represents this the best as it allows you to represent a users' log in status and different states which relate to this case.

Textual Descriptions for Use Cases form T1.2……

Use case: ViewBasket
ID: 1
Brief description: The act of a customer viewing their 'basket', which is a representation of the items they are holding for a later purchase.
Primary actors: Customer
Secondary actors: None
Preconditions :
1. The customer must have an account with the system (i.e a username and password)
2. The customer must already be logged into the system through their own name and password to access the basket.

Main Flow:

1. The use case starts when the Customer selects the 'basket' icon on the Kerr & Brown website.
2. include (CheckLogin)
3. The system displays all products that are currently being stored in the basket and allows the user to proceed to the 'checkout' with the products.
4. A Customer can add and delete items from their basket.
5. extensionPoint: MakeOrder

Postconditions :

1. The customer is aware of the items being stored in their basket.
2. They have updated the basket as they feel necessary.

Extension Use Case: CheckLogin

ID: 1.1

Brief description: A customers' login details must be checked before they can have any access to their personal account.

Primary actors: Customer

Secondary actors: None

Segment 1 preconditions :

1. The customer wants to access information which is personal to an account. For example their shopping basket or purchase history.

Segment 1 flow:

1. The use case begins when the Customer tries to access an area of the system which is only accessible if you have an account.
2. The system displays a message saying that login details are required and offers inout text boxes for the Customer to enter their details.
3. The system checks the login details and allows/doesn't allow access.

Segment 1 postconditions :

1. The customer is now logged into their account and has access to their personal information.

Extension use case: MakeOrder
ID: 1.2
Brief description: The act of a customer placing an order through the online system.
Primary actors: Customer
Secondary actors: None
Segment 1 preconditions :

1. The customer must have an account.
2. The customer may or may not have added the product(s) to their basket first.

Segment 1 Flow:

1. The customer will see a list of the products they have selected to order and then would be asked to confirm that these are the correct products.
2. The customer will then be asked to enter their payments details and contact information. If they have used the system before, this information will have already been saved.
3. extentionPoint: RejectOrder
4. The system then creates a confirmation document which is sent to the customer via email

Segment 1 postconditions :

1. The customer has placed an order.
2. The customer has a confirmation document which contains all the important information they need.

Extension Use Case: RejectOrder

ID: 1.3

Brief description: The act of an order being declined by the system. This may be due to incorrect payment details being given, insufficient funds on the customers' card, or the product(s) they are looking to buy being out of stock.

Primary actors: Customer

Secondary actors: None

Segment 1 preconditions :

1. There must be a problem with the order.

Segment 1 flow:

1. The system displays a message to the user/customer explaining what the problem is.
2. The system then offers an overview of the details that have been entered so that the customer can make necessary fixes to them.

3. The system offers a contact number for the staff at the store so the customer can get in contact of they need to.

Segment 1 postconditions :

1. The order is cancelled if the problem cannot be solved or resumed if a fix has been made.

Use case: ReturnProduct
ID: 2
Brief description: The act of a customer returning a product to the store due to them being unsatisfied with the product.
Primary actors: Customer
Secondary actors: None
Preconditions :
1. The customer must have purchased the product from the store or website
Main Flow:

1. The use case begins when the Customer explains to a member of staff that they would like to return an item.
2. The customer is then asked to fill in a 'return sheet' which specifies their reasons for returning the product.
3. The product in question is sent to the workshop to be inspected.
4. include(InspectProduct)
5. If there is no damage to the product being returned, then the customer is refunded.

Postconditions :

1. The product has been returned.
2. The customer may or may not have been refunded.

Extension Use Case: InspectProduct

ID: 2.1

Brief description: The act of a product being inspected in the workshop. This can be any of the produced produced by Kerr and Brown and can be before the product is sent to the shop or after the product is returned by a customer.

Primary actors: Workshop

Secondary actors: None

Segment 1 preconditions :

1. The product must either be new or must have been returned by an unsatisfied customer.

Segment 1 flow:

1. The extension case begins when a product is sent to the workshop from the company's store or when a new product has been made.
2. The product is inspected for quality by the team in the workshop.
3. Any problems are logged into the system and further actions are taken. (Either sent to the shop or the problem is fixed.)

Segment 1 postconditions :

1. Any problems with the product have been identified.

Use case: ReturnForRepair
ID: 3
Brief description: The act of a customer taking shoes or belts back to the store for repair.
Primary actors: Customer, Staff
Secondary actors: Workshop
Preconditions :

1. The customer must have bought the product from Kerr & Brown.
2. The product must need some sort of repairing.

Main Flow:

1. The use case starts when the customer returns a pair of shoes or a belt for repair.
2. The customer fills out a form explaining what exactly they would like to be repaired.
3. The store staff member takes the product and give the customer a receipt to be shown when the shoes are collected again The customers' details are also taken in case contact is needed.
4. The member of staff sends the shoes or belt to the workshop.
5. include(RepairProduct)
6. The customer pays for the repair when they return to collect the shoes.

Postconditions :

1. The shoes are repaired.
2. The shoes are returned to the customer

Extension Use Case: RepairProduct

ID: 3.1

Brief description: The act of the workshop repairing a product, for example the resoling of shoes.

Primary actors: Workshop

Secondary actors: None

Segment 1 preconditions :

1. The product has been sent to the workshop for repair.

Segment 1 flow:

1. The use case begins when the workshop receives a product to be repaired.
2. The workshop will then read the form attached with the product which explains what needs to be repaired.
3. The repair is made.
4. A price is calculated for the customer to pay based on the work that has been done and how long it took to carry out.
5. The product is returned to the store to be collected by the customer.

Segment 1 postconditions :

1. The product has been repaired.
2. The product has been sent back to the store.

Use case: ViewItems

ID: 4

Brief description: The act of viewing all products available from Kerr & Brown through the online system.

Primary actors: Customer, Staff, Office

Secondary actors: None

Preconditions :

1. The user/actor must have a connection to the system.
2. The products must be sold by the company to be shown on the system.

Main Flow:

1. The use case starts when the user/actor starts using the system. This can be done through the website or through the internal system which is available to staff and the office.

3. The system displays all products that are currently available to buy from the shoe shop.

4. The user can view a description of the product, including a price and a photograph

5. They can also add a product to their basket or place an order for the product.

5. extensionPoint: MakeOrder

Postconditions :

3. The customer is aware of the items available in the store.

Use case: ViewStockLevels
ID: 5
Brief description: The act of an actor/user viewing the levels of stock available in the store at the time of use.
Primary actors: Staff, Office
Secondary actors: None
Preconditions :

1. The user must have the authority to use this area of the system.
2. The user must be logged in as a member of staff or a member of the office.

Main Flow:

1. The use case begins when the Staff or Office log into the system to check stock levels.
2. They will be asked to confirm their username and password.
3. The system will return information on stock based on the actor/users search criteria.
4. extensionPoint: OrderStock

Postconditions :

1. The actor is now more aware of stock levels in the shoe store.

Extension Use Case: OrderStock
ID: 5.1
Brief description: The act of ordering more stock as necessary after viewing the stock levels in the store.
Primary actors: Staff, Office
Secondary actors: None
Segment 1 preconditions :

1. The actor must be authorised to order more stock.

Segment 1 flow:

1. The extension case begins after the Staff or Office actor has viewed stock levels and decided that more must be ordered.
2. The actor selects which items of stock they require and what quantity of these items.
3. If the stock is available, a confirmation message will appear and a receipt can be printed for documentation which will be stored in the shop's office.

4. Otherwise, the actor is told that this stock is not available at this current time.

Segment 1 postconditions :

1. Stock that is low or needed has been ordered if available.

Use case: ChangeStockPrice
ID: 6
Brief description: The act of the price of certain stock items being changed. This may be due to a sale or simply changes in process proposed by the company.
Primary actors: Office
Secondary actors: None
Preconditions :
1. The stock item(s) being manipulated must already be in the system.
Main Flow:

1. The use case begins when the Office logs into the system to change the price of stock items.
2. They are asked by the system if the price change is to be permanent or temporary.
3. If permanent, the actor is asked to enter the new value of the stock item's proce and this is updated on the next day of sales.
4. If this is a sales discount over a period of time, the actor will enter the time period and the discount that they want the item(s) to have. The system then changes the price for the given time period then back to its original price automatically.

Postconditions :

1. A stock item or series of stock items has had its price changed.

Use case: ProcessOrder
ID: 7
Brief description: The act of an order being processed.
Primary actors: Office
Secondary actors: None
Preconditions :

1. The order must have been placed by a customer either by phone, mail, online or in-store.

Main Flow:

1. The use case begins when the Office actor receives information about an offer made by a customer in one of the ways mentioned above.
2. The office actor fills in the order details online and sends them away to the workshop.
3. Any problems that arise will result in the customer having to be contacted either by phone or by email.

Postconditions :

1. The customers' order has been fully processed and the products are ready to be made.

Use case: AddNewStock
ID: 8
Brief description: The act of new item(s) of stock being added to the system to be viewed and purchased by customers.
Primary actors: Office
Secondary actors: None
Preconditions :

1. The stock being added cannot be a part of the current stock.

Main Flow:

1. The use case starts when the Office adds a new item of stock to the system.
2. They are asked to provide information about the product including a price and a photographic image. There must also be an overview description.

Postconditions :

1. A new product has now been added to the systems' database.
2. Customers can now view and purchase this product online and also in-store.