MyBrew Café System Analysis & Design Documentation

Ngcongo, SB, Mr
[26409852@sun.ac.za]

5/5/25

OI318

Table of Contents

1. Scenario Development

Case Description

MyBrew (**real-life client**) is a small, independent coffee shop located in the heart of Stellenbosch university (the Neelsie) as well as on Tygerberg campus owned by Freddie Krugar. MyBrew serves a wide range of hot beverages, tea, cold beverages, and snacks. MyBrew caters to university students and its stuff. I selected MyBrew due to its interaction with suppliers, customers and internal operations that are ideal for modelling use cases.

To gain insights and better understanding of how they operate, I then messaged the owner Freddie on WhatsApp to set up a meeting so I could engage in a formal discussion and get permission to engage with some of the staff. I told him that I was a student at the university, conducting a system modelling assignment and requested permission to ask a few questions about how they operate. He answered a few questions and allowed me to speak to one of the managers who also allowed me to speak to one of the baristas and the person who was responsible for managing stock (see appendix for Transcript).

How the MyBrew system works

Customers place their orders at the counter, the barista on duty then inputs the order into the roastery's point of sale (POS) system. The system will then record the order, send the relevant item to the barista at the back, and will calculate the total cost of the order. The customer can then pay, using cash, card or COB. Once the payment is processed, the system then updates the store's inventory based on what was sold and logs the transaction. My Brew allows students to tap their student cards to receive loyalty points and after a certain number of points, they get a coffee or snack for free.

To promote sales, Freddie partners up with various communities on campus. He has gone as far as creating drinks with the different communities he has partnered with. These drinks were then entered into the POS system and shared via My Brews Instagram page as well as the communities Instagram page. The menu items are synced between the display menu and the POS system, ensuring that there's a link between customer communication and transaction processing.

MyBrew restocks its inventory once a week. The manager is the person who's in charge of restocks, orders the items needed manually via email, WhatsApp or phone call. The baristas are the ones who normally check the inventory. Once the supplier delivers the ordered items, the inventory barista will then check the items against the order list. Once the items ordered are accepted, they are then manually logged and updated on the POS system to reflect the current stock available. If there are any items that are missing, damaged, or wrongly supplied, the supplier is contacted and often the items may need reordering.

If a product is out of stock when the customers' orders it, the barista will notify the customer and will offer an alternative. Substations are manually handled at the point of sale as the system does not show real-time stock visibility to the customers.

If they are customers that are unhappy about the service or product quality, the staff are trained to handle the situation politely and immediately. If the need arises, depending on the complaint, it is escalated to the manager and logged so that it's reviewed later.

The MyBrew employees play a crucial role in the running of the shops system. The manager oversees the staffing, operations and ordering, while the baristas oversee taking customers' orders and processing them, while the inventory assistant handles the items being delivered by the supplier. My Brew ensures that their coffee beans are imported from the GGHIBLIR15 commercial Coffee Roaster from Coffee Tech to ensure that the latest unrivalled superior technology in coffee roasting is used to unlock each bean to its full potential. They use 100% Arabica beans to blend these single origin beans after the roast into perfectly sought after flavour and taste that makes many of their customers shout for more. ("*Our Community | My Brew*," https://mybrew.capetown/our-community/.)

2. Use Case System Analysis

2.1. Identified requirements.

Functional Requirements (What the system must do)

1. Order Processing/Capture
    - The system must allow the baristas to input the customer's order into the POS, while automatically calculating the total cost of the customer's entire order.
    - The system should further support multiple payments methods such as cash, card, COB.
2. Inventory Update
    - The system should update inventory levels automatically after an order is placed and paid for. It Should further allow the manual stock update once they are new items received from the supplier. Lastly the system should be able to alert the baristas once their inventory is running low.
3. Loyalty Points Program
    - The system should be able to track the student's loyalty via student card; by allowing the students to tap their student cards to earn loyalty points, it should further alert the barista when the student has reached a certain threshold to receive a free item.
4. Menu Sync
    - The system should ensure that there is sync between the display menu items as well as the POS. It should further allow the removal or addition of custom drinks when partnerships with communities occur or promotions.
5. Supplier Ordering

- The system should allow manual stock taking, and automatically adjust the current available stock, while creating restocking reports for the manager.

6. Product Substitution
    - The system should allow the barista to change the unavailable items at the point sale.

Non-Functional Requirements (how the system should perform)

1. Usability
    - The system should be user-friendly so that the barista is efficient when inputting orders. It also should minimize any data errors the Barista might enter during the point of sale.
2. Reliability
    - The system should be able to function properly without breaking down during operating hours as well as peak times. It should ensure that no data is lost during processing.
3. Performance
    - The system should process orders quickly to avoid long waiting times and update inventory in real time.
4. Security
    - The system should ensure that only authorized personnel have access to the inventory and financial information of the customers. System should adhere to PCI compliance standards
5. Integration
    - The system should be able to be integrated with 3rd party payment systems, such as card machines and COB
6. Maintainability

- The system should be able to update itself and be able to correct menu items and inventory records. It should further be able to provide troubleshooting issue logs.

7. Accuracy
    - The system should ensure that all data recorded such as loyalty points, stock count and the like is accurate and always up to date.
8. Accessibility
    - The system should be accessible on both the Stellenbosch university campus and Tygerberg campus.
9. Compatibility
    - The system should be able to be integrated with receipt printers as well as the student card tap readers.

2.2. Use Case Diagram

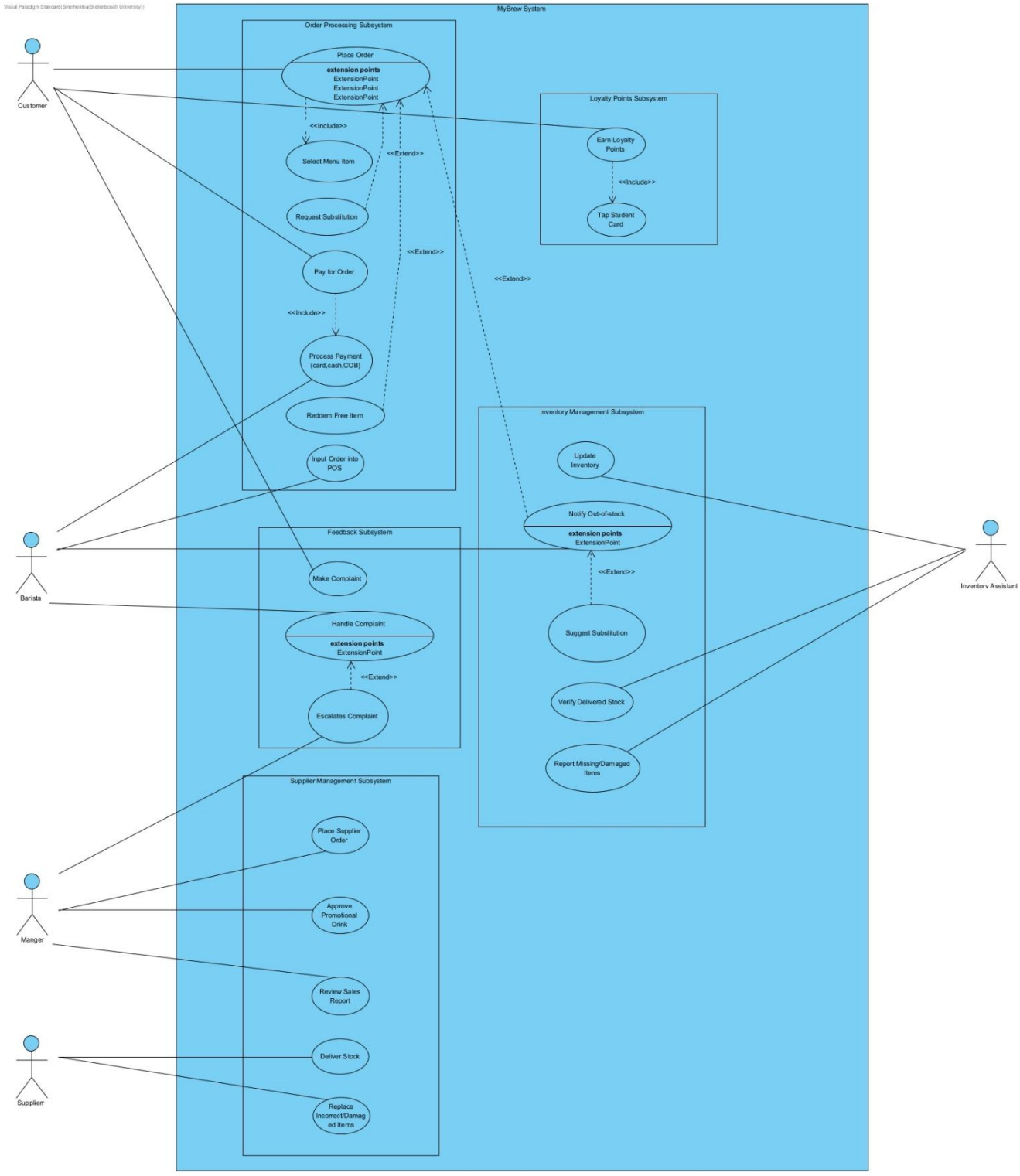

Figure 1: MyBrew Usecase Diagram

2.3. Use Case Descriptions

Place order

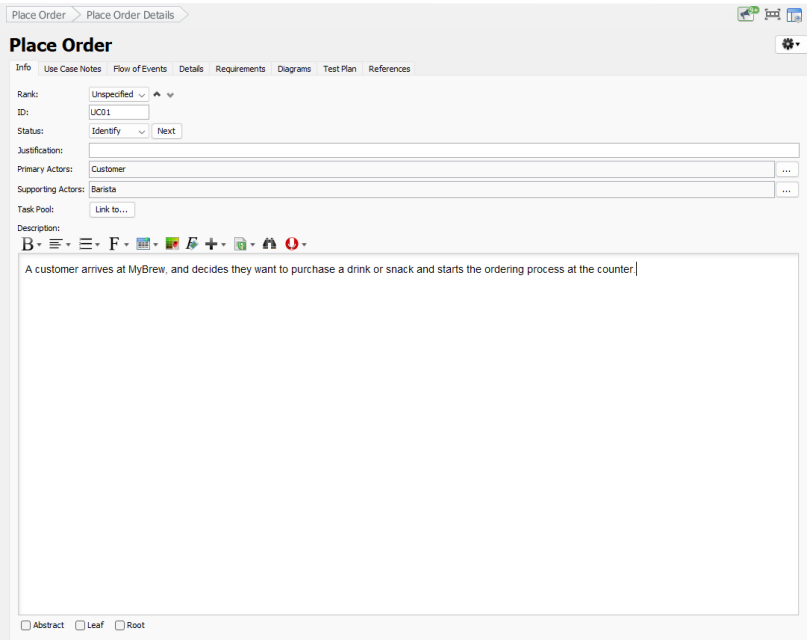

Figure 2: Place order Info

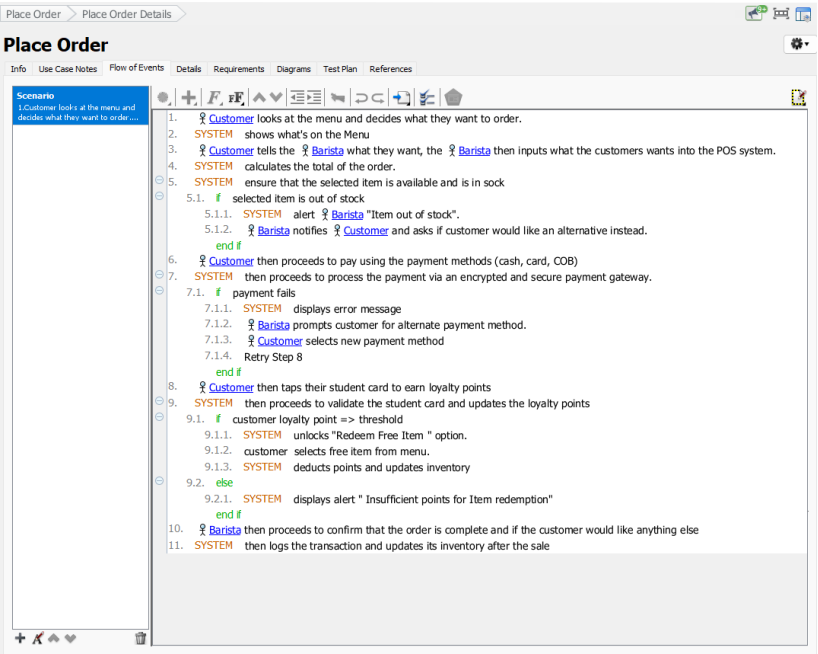

Figure 3: Place order Flow of events

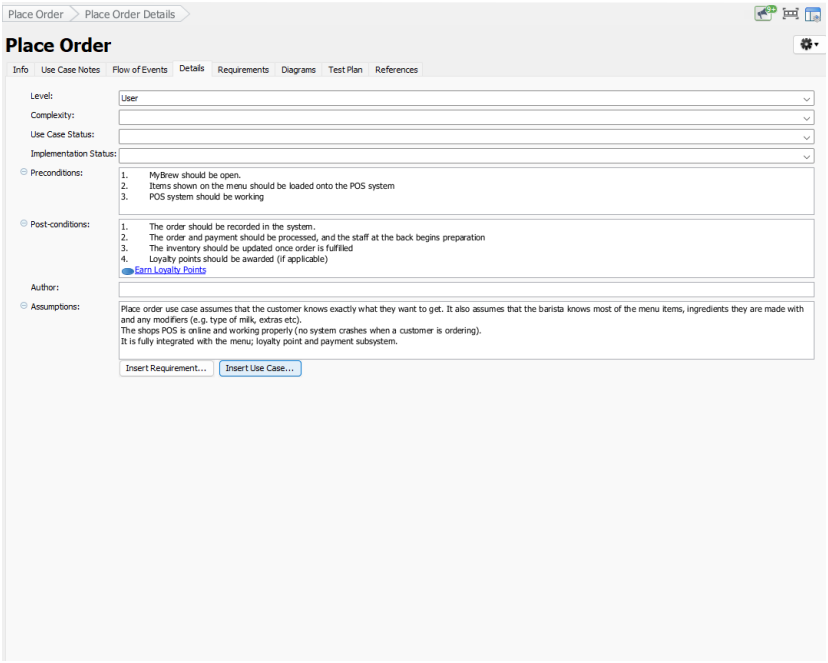

Figure 4: Place Order Details

Process Paymment

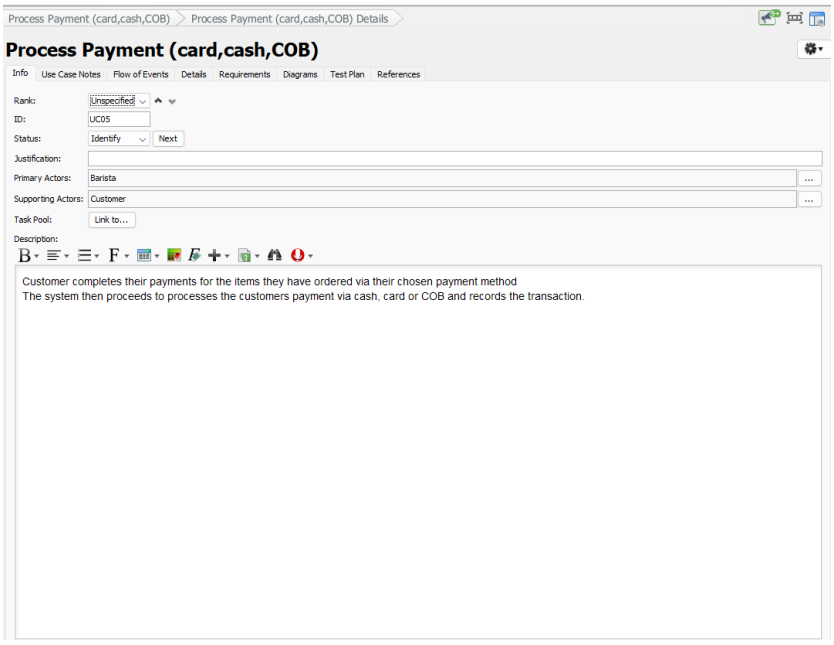

Figure 5: Process Payment info

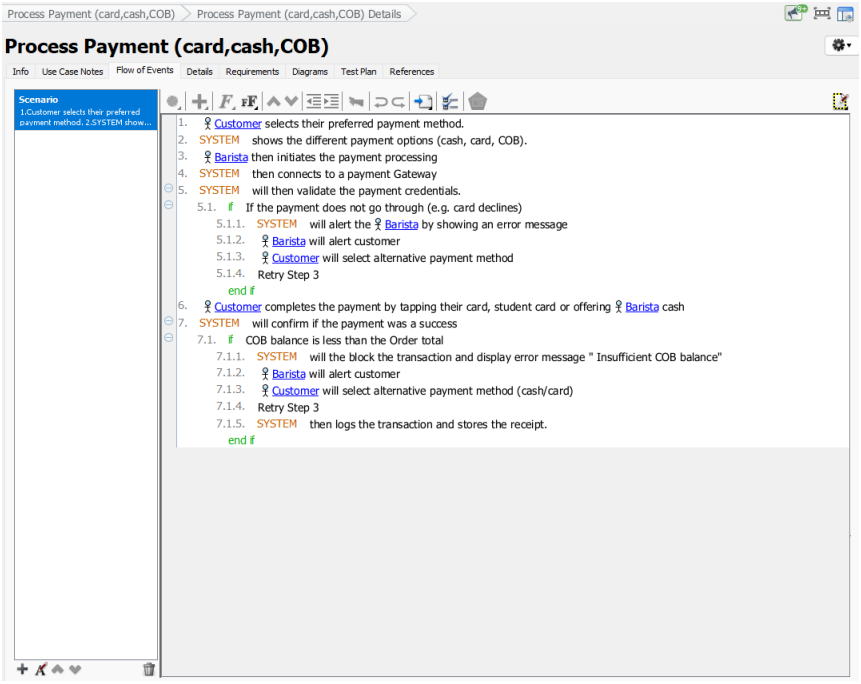

Figure 6: Process Payment flow of events

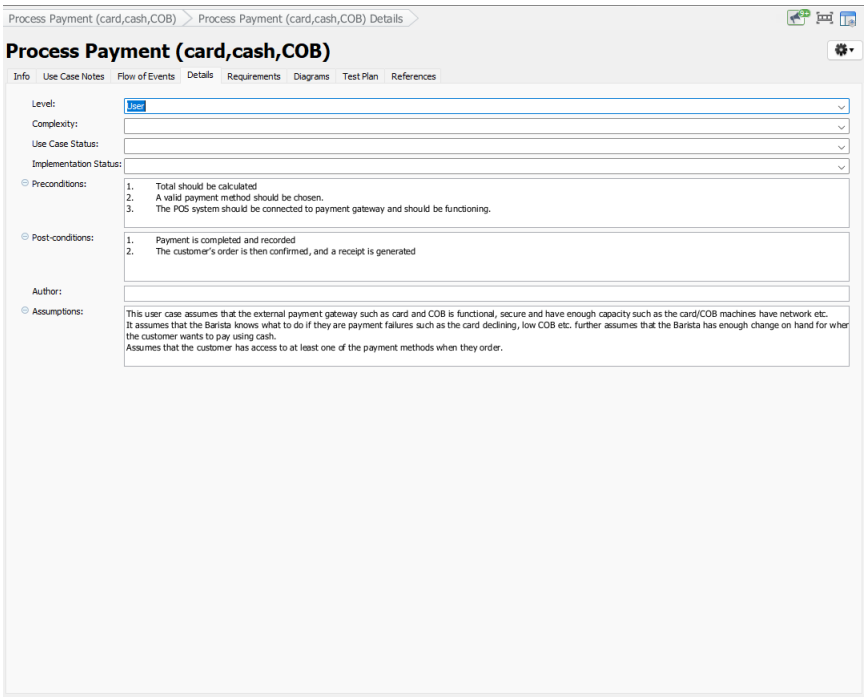

Figure 7: Process Payment Details

Redeem Free Item

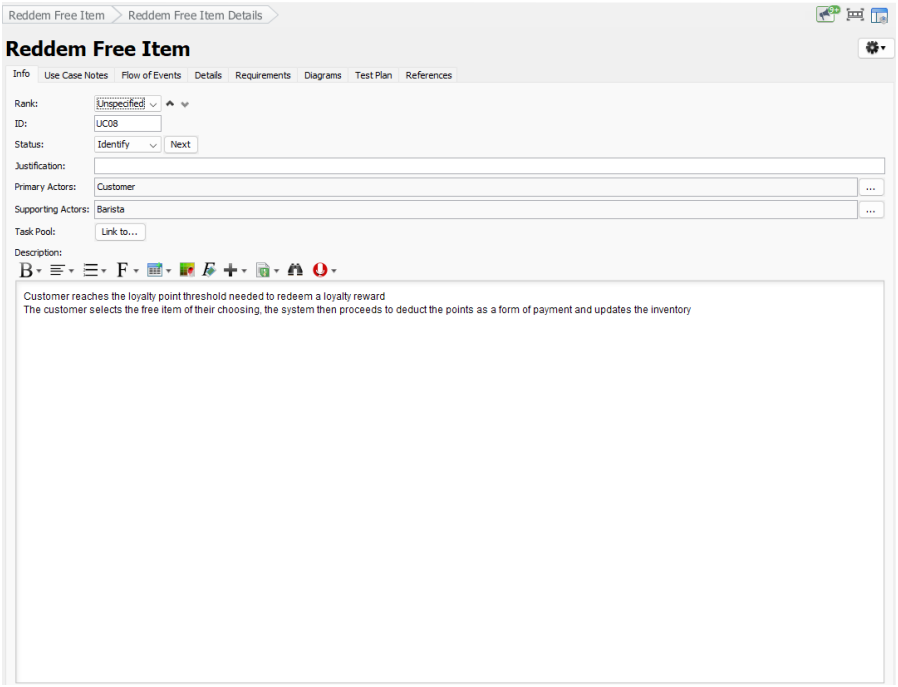

Figure 8: Redeem Free Item info

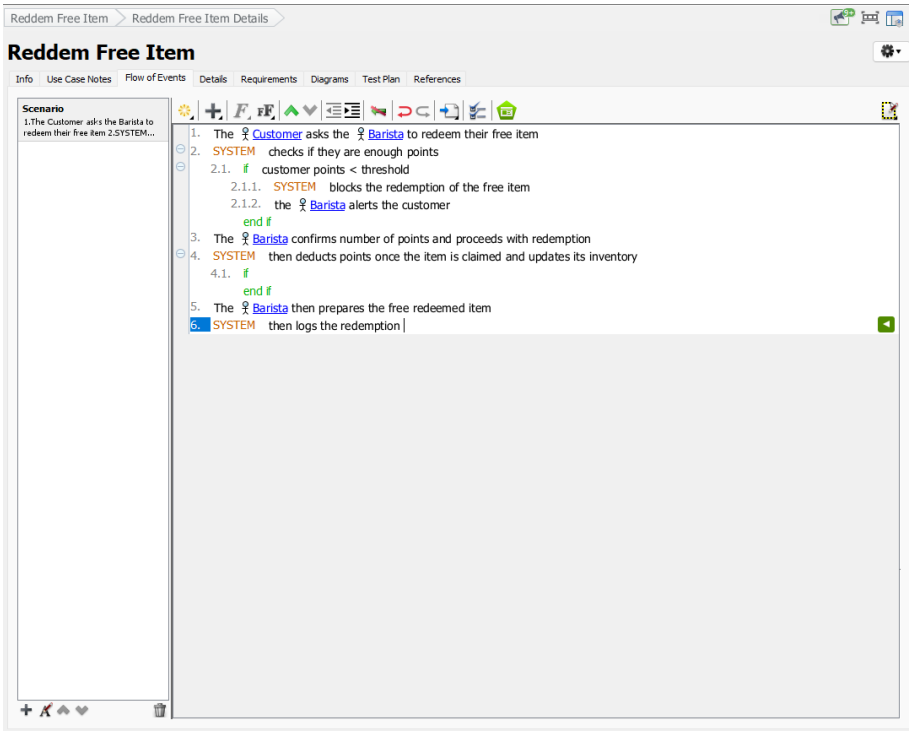

Figure 9:Redeem Free Item flow of events

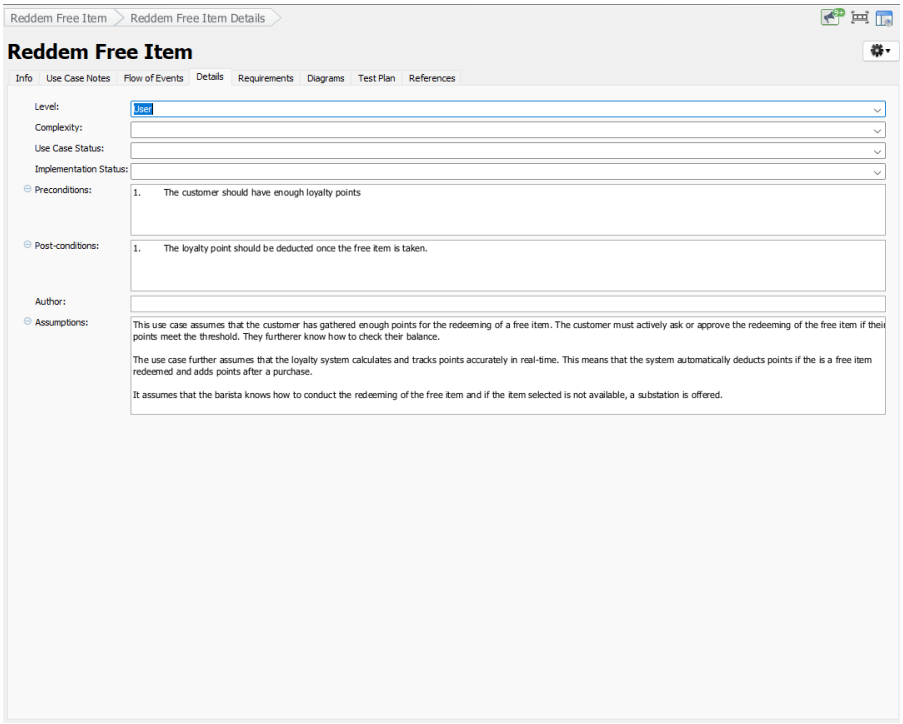

Figure 10: Redeem Free Item Details

Make Complaint

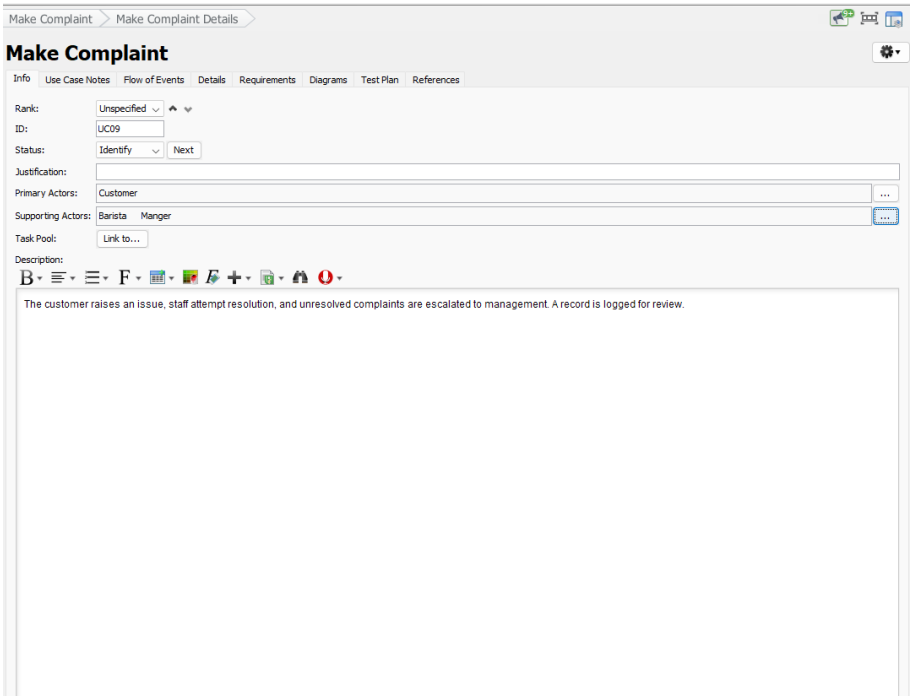

Figure 11: Make Complaint info

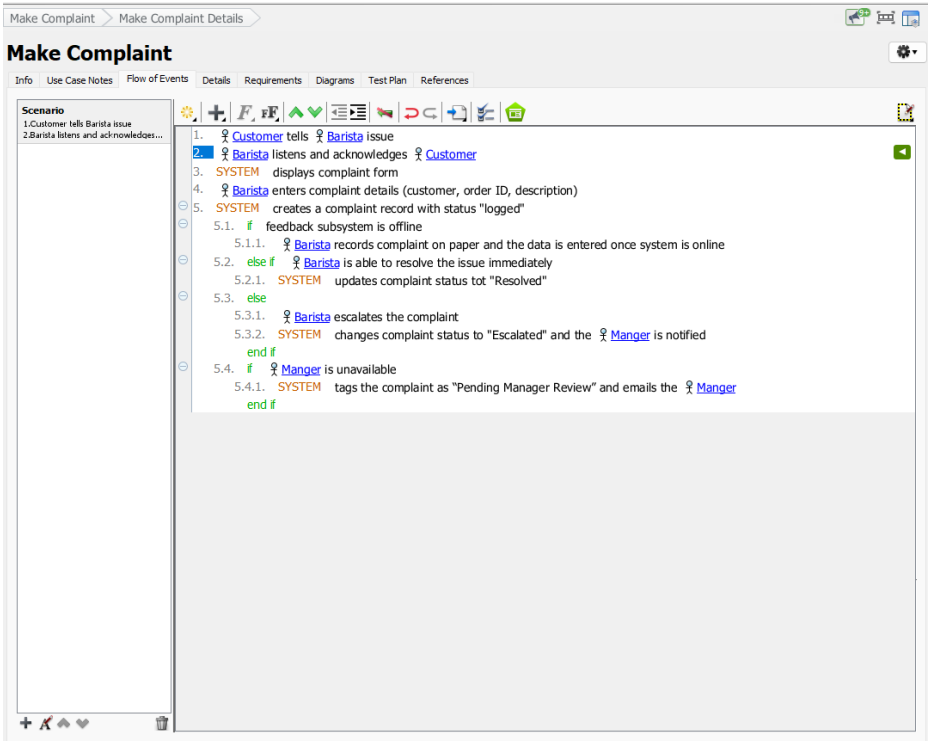

Figure 12: Make Complaint flow of events

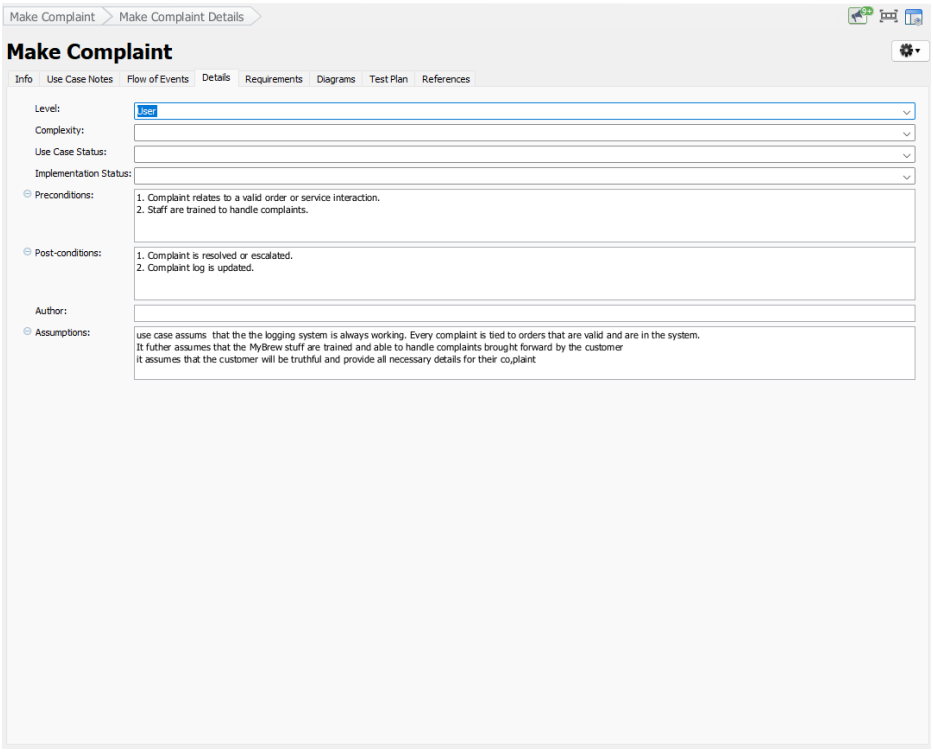

Figure 13 Make Complaint details

2.3.1 SSD Diagrams for Flow events of the Use case descriptions

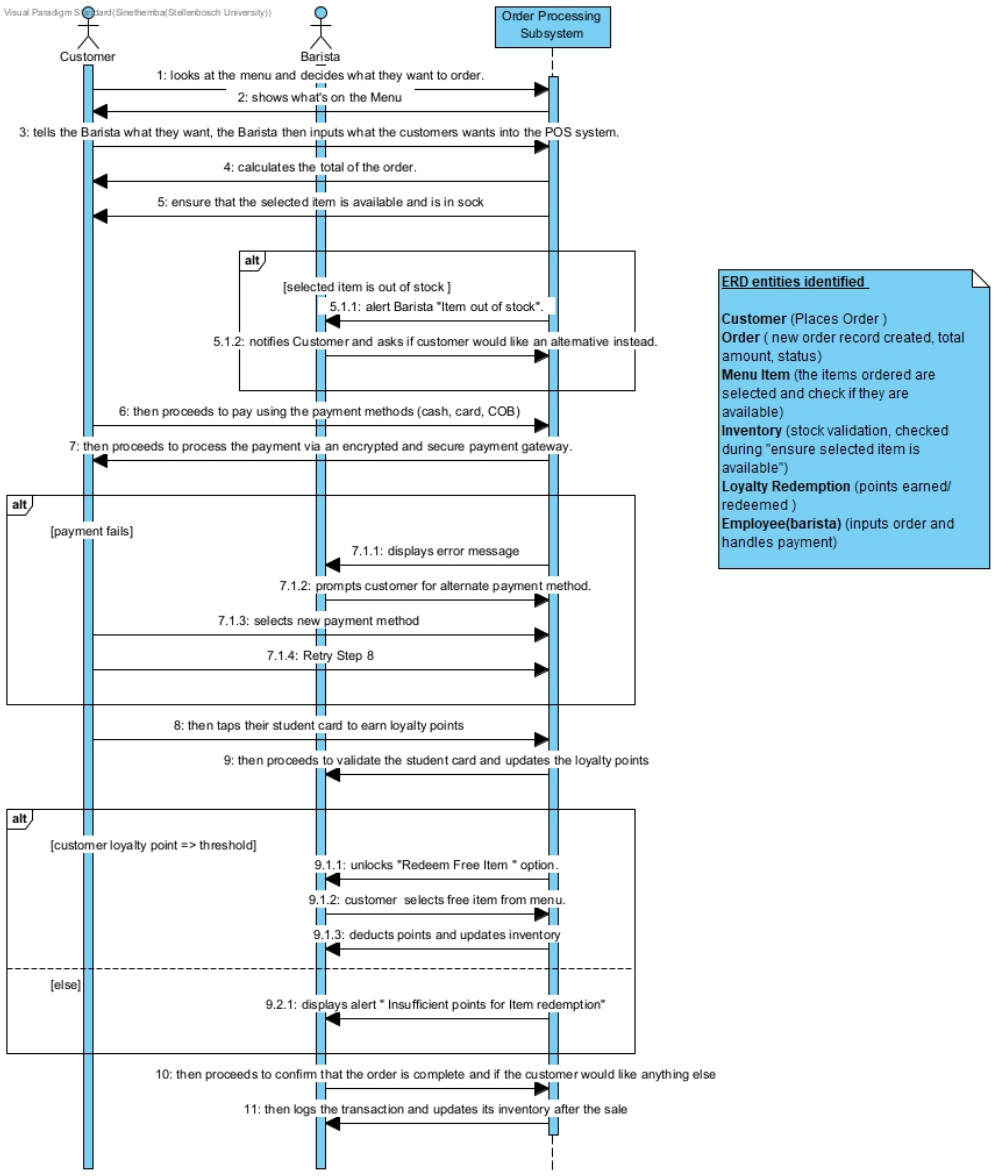

Figure 14:Place Order SSD

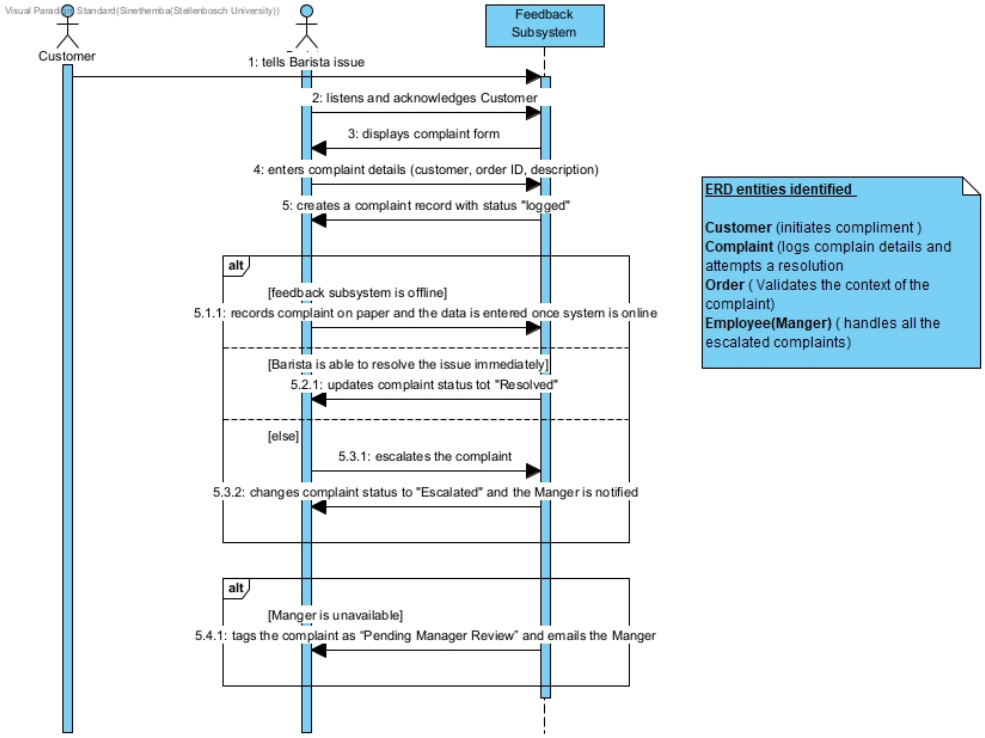

Figure 15: Make Complaint SSD

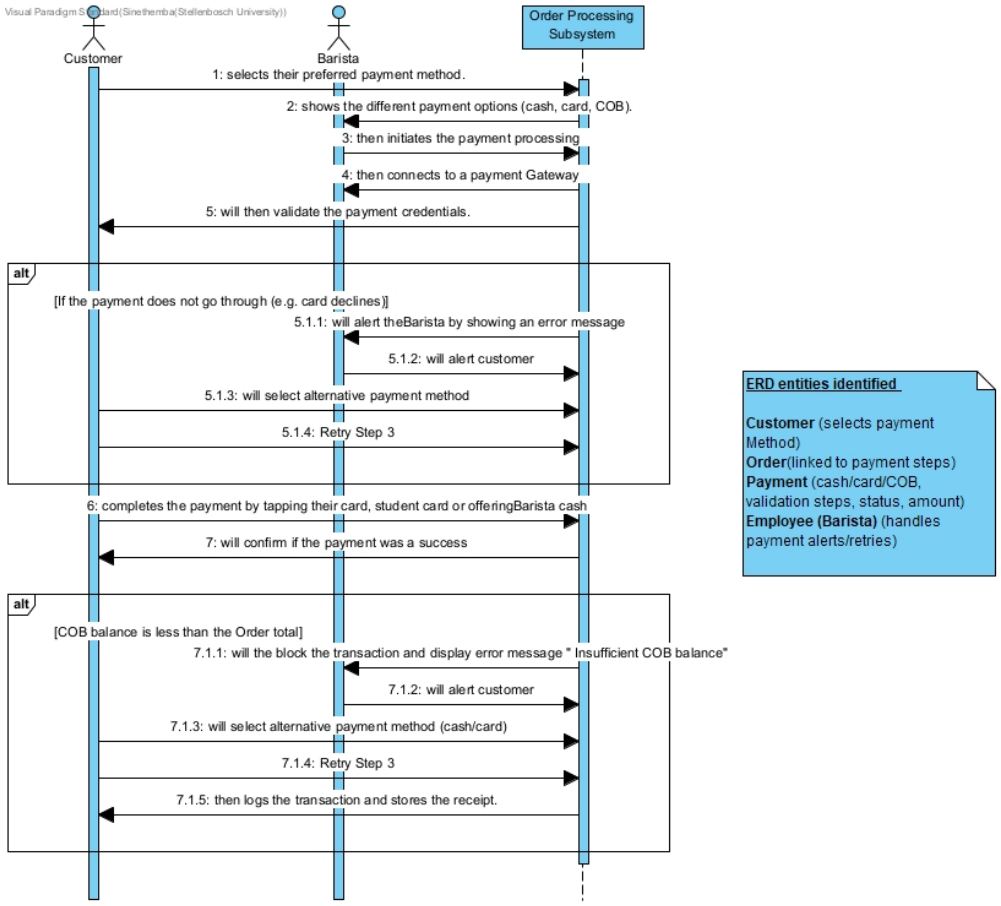

Figure 16: Payment Process SSD

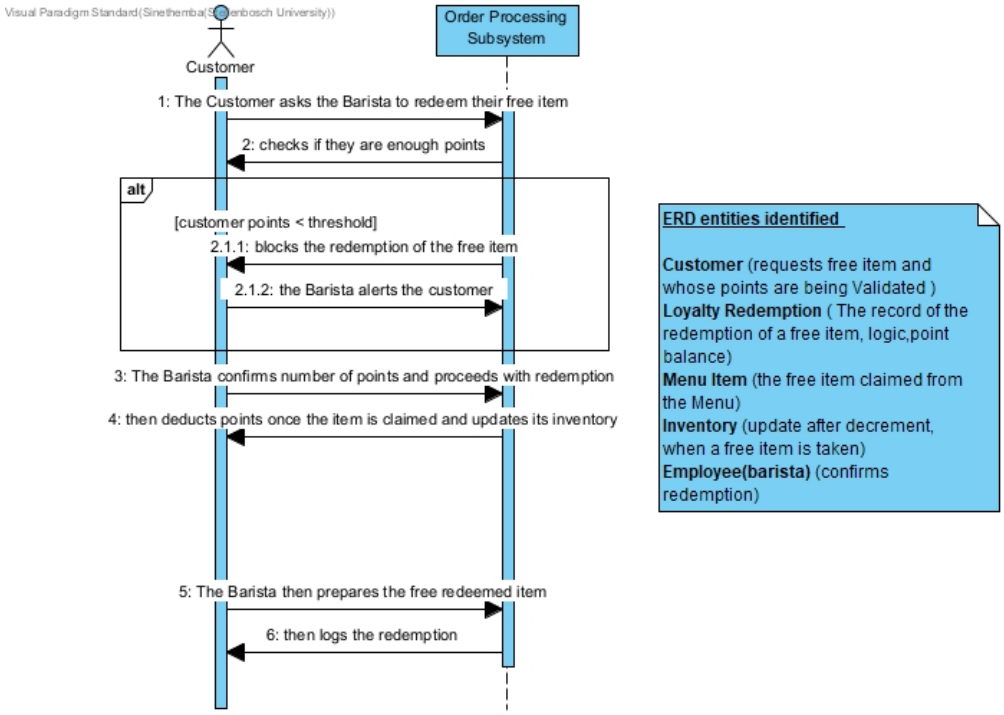

Figure 17:Redem Free Item SSD

3. Database Design

3.1 Entity Relationships Diagram

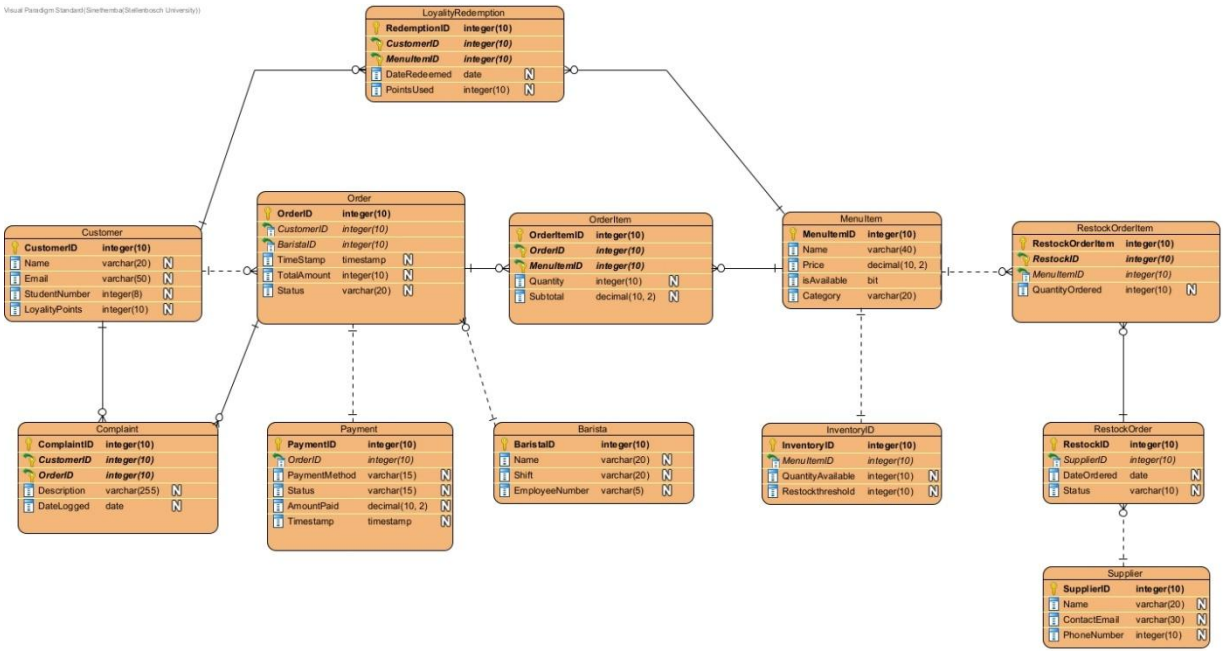

Figure 18: MyBrew ERD

3.2 Data Population

The data of the above ERD (3.1) was entered the following way:

1. The core “Lookup” Tables were filled in first. These are tables with no incoming foreign keys (FK). These tables were the **Customer**, **Barista, Supplier**, **MenuItem** entities. (see below for the screenshot of the data for each of the lookup entities)

Data of Customer

CustomerID (PK)	Name	Email	StudentNumber	LoyalityPoints
1	Sinethemba	Sinethemba@gmail.com	26409852	120
2	Jabu	jabu@sun.ac.za	24345678	45
3	Mike	Mike@yahoo.com	25678998	450

Figure 19: Customer data/table.

Data of Barista

BaristaID (PK)	Name	Shift	EmployeeNumber
1	Thulani J	Afternoon	B1001
2	Siya M	Morning	B1002

Figure 20: Barista data/table

Data of Supplier

SupplierID ...	Name	ContactEmail	PhoneNumber
1	FreshRoast	orders@freshroast.c...	0112220001
2	GGHIBLIR15	order@gghiblir@.co.za	0215550202
3	MrB bakeries	supply@mrbbake.co.za	067123456789

Figure 21: Supplier data/table

Data of MenuItem

MenuItemID (PK)	Name	Price	isAvailable	Category
1	Americano	25.00	true	Hot Beverage
2	wrap (Chicken cranb...	39.00	false	Snack
3	CarrotCake Muffin	34.00	true	Snack

Figure 22: MenuItem data/table

2. The **Inventory** table/entity was populated next. This is because the table refences and depends on the **MenuItem** records (FK references MenuItem)

Data of InventoryID

InventoryID (PK)	MenuItemID (FK)	QuantityAvailable	Restockthreshold
1	1	20	10
2	2	30	5
3	3	10	10

Figure 23: Inventory data/table

3. **RestockOrder** and **RestockOrderItem** (Supplier Orders) were populated next after the population of Inventory because they both reference and depend on the **Supplier**. This was done so the supplier-side stock can be tracked (N.B RestockOrder and RestockOrderItem do not depend on Inventory, thus making their placement flexible if the lookup tables are populated)

Data of RestockOrder

sion Table - insert a decision table in diagram for documenting business decision

RestockID (PK)	SupplierID (FK)	DateOrdered	Status
1	1	05/05/2025	Pending
2	3	30/04/2025	Success

Figure 24:Restock data/table

Data of RestockOrderItem

RestockOrderItem (PK)	RestockID (FK)	MenuItemID (FK)	QuantityOrdered
1	1	1	20
2	2	2	10

Figure 25: RestockOrderItem data/table

4. **Order** and **OrderItem** (customer transaction) were populated next after supplier orders because they reference and depend on **Customer**, **Barista** and **MenuItem** records.

Data of Order

OrderID (PK)	CustomerID (FK)	BaristaID (FK)	TimeStamp	TotalAmount	Status
1	1	1	06/05/2025 09:53	64	Processing
2	2	2	03/05/2025 10:00	34	Complete

Figure 26:Order data/table

Data of OrderItem

OrderItemID (PK)	OrderID (PK+FK)	MenuItemID (PK+FK)	Quantity	Subtotal
1	1	1	1	25.00
2	1	2	1	39.00
3	2	1	1	34.00

Figure 27: OrderItem data/table

5. **Payment** was populated next after customer transactions since orders must exist before you record items or transactions

Data of Payment

PaymentID (PK)	OrderID (FK)	PaymentMethod	Status	AmountPaid	Timestamp
1	1	COB	Success	64.00	06/05/2025 09:56
2	2	Card	Pending	39.40	23:27 2025/05/05

Figure 28:Payment data/table

6. **Complaint** and **LoyaltyRedemption** were populated last, as they reference and depend on **Customer**, **Order** and **MenuItem** records.

Data of Complaint

ComplaintID (PK)	CustomerID (FK)	OrderID (FK)	Description	DateLogged
1	1	1	chicken was raw	06/05/2025

Figure 29:Complaint data/table

Data of LoyalityRedemption

RedemptionID (PK)	CustomerID (FK)	MenuItemID (FK)	DateRedeemed	PointsUsed
1	1	3	06/05/2025	100

Figure 30: LoyalityRedemption data/table

Following this order ensure that every FK inserted would be able to point to an existing Primary Key (PK) thus maintaining referential Integrity.

Workflow Example using populated data in the Tables above (Figure 19-30):

Sinethemba N. (CustomerID 1, studentNumber: 26409852) walks into MyBrew and places an order with Barista Thulani J (BaristaID 1). The POS system then creates a new order record (OrderID 1) which is linked to CustomerID 1and BaristaID 1. It further timestamps it to 06/05/2025 09:53 and sets the status to “Processing”.

Two OrderItem entries are then added for that order, one for an Americano (MenuItemID 1, Quantity 1, subtotal R25.00) and one for the Chicken cranberry wrap (MenuItemID 2, quantity 1, subtotal R39.00). The Barista confirms the total R64.00.

Sinethemba proceeds to pay by COB; the system records the payment (PaymentID 1) which is tied to OrderID 1 with the status being “Success”, amount paid R64.00 and the timestamp 06/05/2025 09:56.

The inventory levels for MenuItem1 and 2 are immediately decremented (from 20 to 19 and 30 to 29, respectively) ensuring that there's referential integrity with the Inventory Table.

Sinethemba then taps his student card, this triggers the POS to include an "Earn Loyalty Points" action: His points increase from 120 to 183. Since Sinethemba has more than 100 points, the system offers a free Carrot Cake Muffin (MenuID 3, and 100 points are deducted when he claims the free muffin (new balance 83 points); Inventory for MenuID 3 is also decremented from 10 to 9.

Lastly the order status changes to "Paid" and Sinethemba leaves with his free Carrot cake muffin, Americano and cranberry chicken wrap.

Conclusion

The aim of the project was to analyse and design a complete system for MyBrew, a coffee shop located across Stellenbosch University campuses. Industry standard UML modelling techniques, databases design principles were used to develop the system. To ensure that both stakeholder needs and requirements needs were met, a structured approach was used.

MyBrew's everyday operations were studied. The focus was on the connection between the customers, baristas, managers and their suppliers. The system addresses core workflows such as order processing, loyalty programs, inventory management, and complaint resolution.

In order to Analysis the use cases, four were considered (Place Order, Process Payment, Redeem Free Item, and Make Complaint). The uses cases were then modelled with detailed descriptions, relationships (<<include>>/<<extend>>) and SSD diagrams. In doing so, the diagrams were able to show a clear interaction between the actor, subsystem boundaries and exception handling.

An ERD diagram was created. The diagram integrated different entities such as Supplier, Customer, Order, Menu Item etc. The ERD ensured that transactional workflows such as loyalty point redemption, order tracking etc. were supported as well as referential integrity.

In order to show how real-world transactions would function, sample data was systematically populated into the different tables such as order, payment, customer etc. The population of data ensure that a dependency of hierarchies was followed to ensure that they is referential integrity.

These were the key learning and reflections from the project:

1. Using Visual Paradigm showed me how it's possible to simplify complex workflows by creating detailed use case diagrams and SSDs.

2. Designing the ERD showed me the importance of normalization such as separating the Order from the Order item tables therefore archiving less redundancy and ensuring they is data integrity.

3. When it came to populating the data, I noticed that poor data population could lead to system errors and how important it is to populate the tables in sequences. An example is ensuring that the customer table is populated before the order table.

4. Lastly this project made me realize the gap between practical implementation and theoretical models.

The recommendation I have is that the system should expand to support delivery and online orders. This will allow MyBrew to reach a broader customer base outside of the university. Another is that they should add reporting modules, so they are able to analyse performance, sale trends and loyalty program engagement.

In essence this project provided insights into system analysis and design, showing how the theory ties together with real-world applications. It further highlighted the importance of stakeholder feedback, iterative design and ensuring that a robust business system is developed, to ensure that it's always functioning.

Appendix

Transcript of Interactions with MyBrew Staff

Interview with Manager

Sinethemba: *"How do you usually plan weekly or daily specials?"*
Manager: "We don't run daily or weekly specials. Instead, we focus on seasonal collaborations with campus communities, like creating custom drinks. For example, this coming Month, we are partnering with Wimbledon Cluster to have a cluster drink.

Sinethemba: *"How do you update prices or menu items on the POS?"*
Manager: "We use a custom system built by a local tech company. It's basic but reliable. We however have to manually input new items or adjust prices. For promotions, we also update the digital menu boards and social media posts ourselves."

Sinethemba: *"How do you manage supplier relationships and stock deliveries?"*
Manager: "We email or WhatsApp suppliers weekly. The inventory assistant checks stock levels, and I place orders based on their report. When deliveries arrive, they're cross-checked against the order list. If there's a mismatch, we contact the supplier immediately, sometimes they reship, sometimes we reorder."

Sinethemba: *"How are customer complaints handled?"*
Manager: "Baristas resolve minor issues on the spot, like remaking a drink. For serious complaints, they escalate to me. We log complaints in the POS with the order number, but it's not automated. Most feedback is verbal, though some students tag us on Instagram."

Sinethemba: *"Any systems you'd like to improve?"*
Manager: "Real-time inventory tracking. The baristas check stock manually, and sometimes we run out mid-day. Automating that would save time and reduce errors."

Interview with Barista

Sinethemba: *"Walk me through taking an order."*
Barista: "The customer tells me their order, I input it into the POS. The system calculates the total and checks stock. If something's out, I suggest alternatives, like a cappuccino instead of a latte. After payment, they tap their student card for loyalty points. Every 100 points earns a free item. The POS prints a docket for the kitchen, and we update inventory after each sale."

Sinethemba: *"What happens if a payment fails?"*
Barista: "The POS shows an error, like 'Insufficient COB balance' or 'Card declined.' I ask the customer to try another method. For loyalty redemptions, the system unlocks a free item option if they have enough points."

Sinethemba: *"Do customers ask about loyalty programs?"*
Barista: "All the time! They love earning points but sometimes get confused about thresholds. A few suggested an app to track points or pre-order drinks."

Interview with Inventory Assistant

Sinethemba: *"How do you track inventory?"*
Inventory Assistant: "I check stock weekly and report to the manager for supplier orders. When deliveries arrive, I count items against the order list and update the POS manually. If something's damaged, I email the supplier, they usually fix it within days."

Sinethemba: *"How do you communicate with suppliers?"*
Inventory Assistant: "Mostly WhatsApp or email. The manager handles ordering, but I send restock reports. A system that auto-alerts suppliers would help."

Sinethemba: *"Biggest challenges?"*
Inventory Assistant: "Delayed deliveries and manual updates. If the POS could track stock in real-time, we'd avoid selling out of popular items like our signature carrot cake muffins"