1CWK100 Case Study Report

SYSTEM ANALYSIS AND DESIGN

Eniola Habeeb-Lahi Bakare

21386088

Table of Contents

Introduction

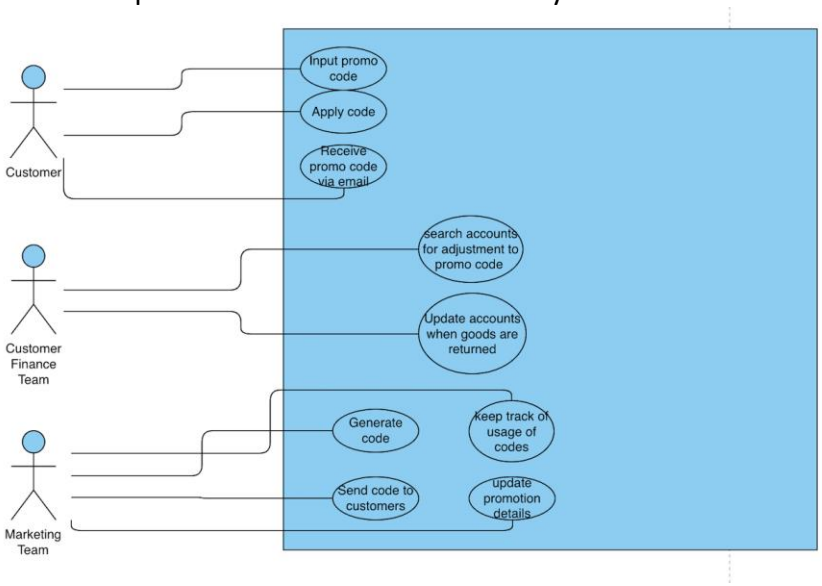
The report below is based on the marketing team of an internet shopping company wanting to introduce a new design to their current system where the team is able to send offers to customers that includes a promotion code or discount code such that the customer receives a discount on the website when certain conditions are met by the order. This process is all done before payment is requested from the customer. After the promotion code is applied by the customer, there is a choice to pay immediately through a credit/debit card method or the order can be added to the customer's shopping account where a monthly bill is produced for their shopping. The shopping company has requested the new features in the system which are listed below:

- The marketing team being able to flexibly change the codes or promotions without the need to request further program changes to the prices and checkout.
- The team wants to be able to keep track of the usage of each code.
- Team wants to be able to communicate details of a promotion with the customers through email.
- The customer finance team wants accounts to adjust automatically and appropriately when a promo code is applied as well as when customers return goods.

The features listed above would help improve the company in different areas. The flexibility in changing the promo codes will help the team update codes easily without experienced personnel and this would save the company cost of hiring someone for that. Keeping track of the codes helps with an organized database and knowing which codes are expired or still active.

User and system Requirements

A prioritized Use Case diagram with all new features needed is down below. The diagram showcases the importance of the new function or features in the system and how to go about it. The Use Case Diagram also helps in identifying the relationships between the actors and the system.



Below are two high priority full use case descriptions.

Use Case: Creating Promotion Details
Actor: Marketing Team
Brief Description
Marketing team creates promotion code, durations and conditions needed for code to apply.
Pre-Conditions
New promotion codes are needed.
Post-Conditions
Code is generated, updated, and saved to database. Website accepts code to be applied when inputted. Customers are emailed the code.
Primary Path
1. Marketing team accesses the database. 2. Conditions for promotion is entered into the database. 3. Duration of code is entered. 4. Code is generated in the database. 5. Database links code generated with the promotion. 6. A list of customers who qualify for the promoting is initiated.

7. An automated email containing the details of the promotion including the promotion code is sent out to all qualifying customers.
Alternate Path
4. Code is manually inputted by the marketing team and then linked to promotion. 5. All registered customers are valid for every promotion.

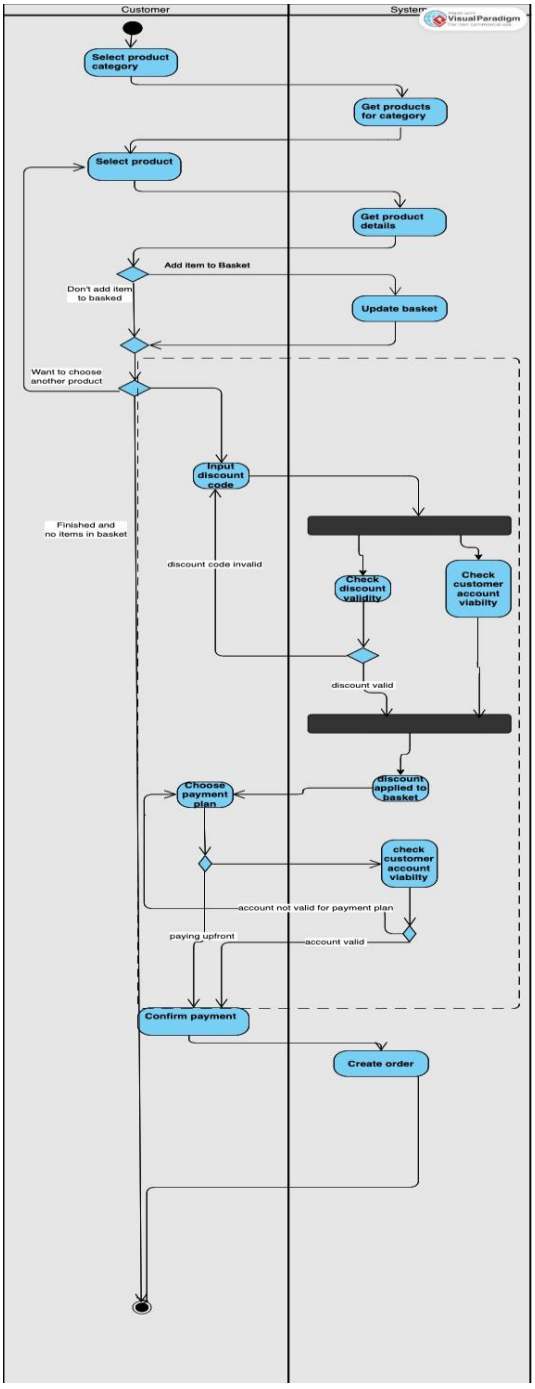
Use Case: Updating Promotion Details
Actor: Marketing Team
Brief Description
Team updating and making changes to promotions
Pre-Conditions
Promotion already exists.
Post-Conditions
New changes made to the promotion are saved in the database. New promotion details are emailed to customers. Customers can make use of the new changes.
Primary Path
1. Marketing team accesses the database. 2. Team chooses which promotions are going to be modified. 3. The new changes are inputted into the database. 4. New code is generated for the promotion changes. 5. Duration of new promotion is inserted into system. 6. New changes are saved to the database. 7. New email containing changes made to promotions and code are composed. 8. Emails are sent out to customers.
Alternate Path
4. Code is not changed and remains the same. 5. Duration remains the same since it's not close to ending.

Non-functional requirements are URPS section of the FURPS categories. FURPS is an abbreviation for **F**unctions, **U**sability, **R**eliability, **P**erformance and **S**ecurity. The **F**unctions category covers functional requirements for the company which includes the business rules and processes. The non-functional requirements are sets of specifications that covers how well the system's operational abilities are and these consists of the user's interface, integrity, failure rate and speed.
A list of non-functional requirements that would need to be met in this case study are listed below:

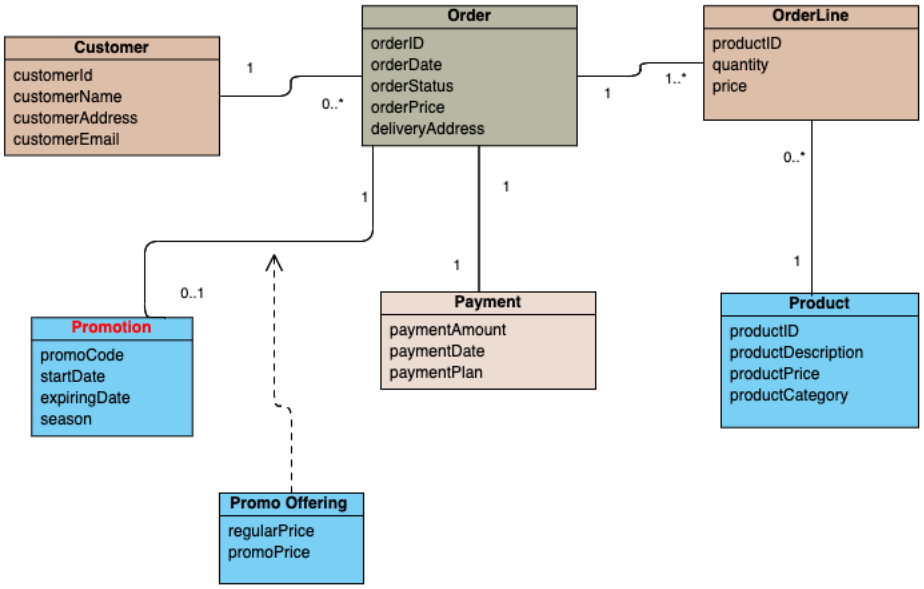
1. *A user-friendly interface*.
2. *A discount (as a percentage or a fixed amount) off all items in the order when the order is above some threshold.*
3. *A discount off items that have a value greater than a stated amount or are a specified category of product.*

4. *Promo codes that only specific customers can use when they have been sent that offer (perhaps a limited number of times)*
5. *Promo codes only valid on orders placed up to a stated date.*

Once the requirements have been gathered, an activity diagram explaining the process the customer uses the promo code is also needed and this is provided below.
The dotted lined square in the image below encompasses the new changes made to the original database to allow the use of promo codes by the customers.



An image of the updated domain model class diagram which contains the promotion table and its relationship to the order table in the database is also shown below.



The promotion class is the new class added to the domain model class. Some other changes were also made. A payment-plan was added to the Payment class as customers have options to either pay upfront or in intervals.
The relationship between the promotion class and order class is also portrayed. It is a 0..1 relation from the promotion class as an order could not have any promotion code in it and the website only allows a maximum of one code per order at the moment.
On the other end of the relationship in the order class, it has a 1..* relation. This is because the minimum amount of order a promotion code could be used for is 1.
The new relationship is only between the promotion and order class and not between promotion and product classes because the promo code is for the entire order and not individual products.
New attributes were also added to already existing tables like delivery Address on the Order class. This was introduced in case the customer wants the product delivered to a different address to the one registered.

Design Aspects

Only one new table was added into the database and that is the promotion table which contains details about all promo codes made by the company. There is also a table dependent on the promotion code called promo offering. This table entails the original price of the order and the price after the promotion has been applied. The data dictionary for each of the new tables added to the database.

Promotion

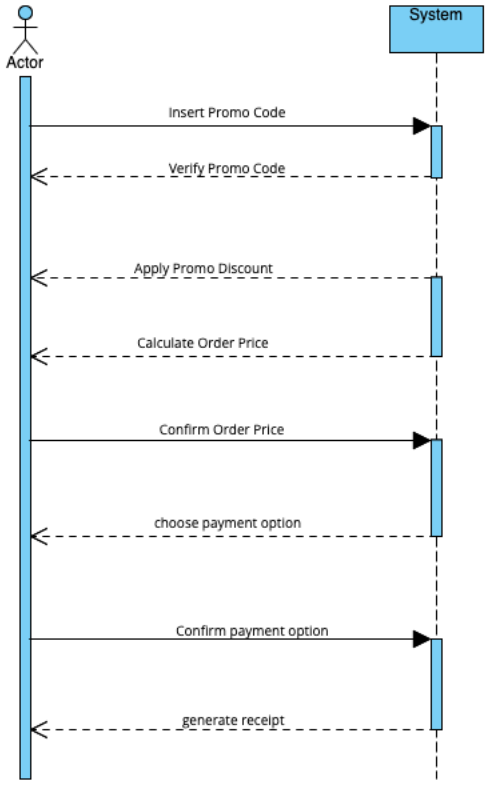
Field name	**Data Type**	**Constraint(s)**	**Key**
promoCode	Varchar(10)	notNull	Primary
startDate	Date(yyyy-mm-dd)	Mandatory(must be a valid date)	
ExpiringDate	Date(yyyy-mm-dd)	Must be a date after the startDate	
season	Varchar(80)		

Promo Offering

Field name	**Data Type**	**Constraint(s)**	**Key**
promoCode	Varchar(10)	notNull	Primary,Foreign
orderID	Integer(10)	notNull	Primary,Foreign
regularPrice	Integer(9)		
promoPrice	Integer(9)		

Tables	**Attributes**
Customer	**customerID,** customerName, customerAddress, customerEmail
Promotion	**promocode,** startDate, expiringDate, season
Promo Offering	**promoOfferingID,** regularPrice, promoPrice
Payment	**paymentID,** paymentAmount, paymentDate, paymentPlan
Order	**orderID,** orderDate, orderStatus, orderPrice, deliveryAddress
Order Line	**orderLineID,** *orderID, productID,* quantity, price
Product	**productID,** productDescription, productPrice, productCategory

In the table above, the bold attributes act as the Primary keys for the table portrayed and the attributes written in italics act as a way of identifying the foreign keys in the table when there is a one -to – many relationships present in any of the table.



A system sequence diagram detailing the interaction and data being passed between the customer and the system in the instance of the customer inputting a code to be used on their order and this has been depicted above.
Supporting the diagram, a list of information needed for the interaction both the inputs and outputs are also provided below.

INPUTS	OUTPUTS
Promo code	Valid code
Enter button	Total after applying code in percentage/price
Add/Remove item button	Error message if code isn't valid
Choose payment option	Payment option accepted
Checkout button	Final receipt

Appendices

In line with the marketing team's request to introduce a new design to handle and store promotional codes in their current system, a case study based on those criteria is shown in the report above. The report includes the system's proposed design, desirable features and functionality, user and system requirements, and non-functional needs. The report offers a thorough analysis of the project as a whole.

The report's set is provided in the introduction, which also underlines the new system's objective of enabling the marketing team to send clients offers that include discount codes. The needs are clearly stated and in line with the business' objectives. They include the ability to alter the promo codes, monitor code usage, share promotions with customers, and modify customer accounts.

The prioritised Use Case diagram accurately shows the new system functionalities that are required as well as their interactions with the actors. In addition to helping to understand the flow of interactions between the marketing team and the system, it provides a visual depiction of the proposed features. The descriptions of the two high-priority use cases, "Creating Promotion Details" and "Updating Promotion Details," are given in a straightforward and brief way. The process of developing and updating promotions is described in detail, along with the creation of promotion codes, database adjustments, and email communication with customers. The alternative routes offer more scenarios and assistance in handling rare situations.

An additional level of explanation of the specifications included in the report is provided by the inclusion of non-functional requirements grouped under FURPS (Functions, Usability, Reliability, Performance, and Security). A user-friendly interface, discount computations, code limits, and code validity periods are just a few examples of the needs that are included. They show an understanding of the capabilities of the system's operation and point out the significance of components other than its fundamental functionality.

The activity diagram that depicts how a client uses a promotional code gives a clear picture of the interaction flow. The activity diagram makes it easier to understand each step, including entering the code, calculating discounts, and getting to the intended outcome. The modified domain model class diagram accurately illustrates the connections between the brand-new "Promotion" class and the pre-existing

classes, including "Order" and "Product." To entirely understand the data structure of the system, attributes must be added, and relationships must be made clear as stated in the case study given.

Lastly, the system sequence diagram demonstrates the interaction and data flow between the customer and the system when applying a promo code to an order. The inputs and outputs are clearly identified, providing a clear overview of the process.

Overall, the report does a good job of outlining the specifications and design features of the suggested system. The content is clearly organised and simple to follow, making it easy for readers to understand the project's essential components. The system's operation and data flow are further understood through the inclusion of diagrams, tables, and a data dictionary.

References

- Gaskin ,J. (2021) Everything You Need To Know About Use-Case Diagrams. Available at: https://venngage.com/blog/use-case-diagram/ (Accessed:24 April 2023).
- Altexsoft.(2022) Non-functional Requirements: Examples, Types, How to Approach. Available at: https://www.altexsoft.com/blog/non-functional-requirements/#:~:text=Non%2Dfunctional%20requirements%20or%20NFRs,reliability%2C%20data%20integrity%2C%20etc. (Accessed:15 April 2023).