

Test Design for Discount Calculator

- List of requirements :
- Discounts are available only to buyers who made their first purchase at least 1 year ago.
 - All members of the loyalty program have a 50 percent discount on their birthday.
 - If the product is re-ordered by the same buyer after the return, no discount is applied.
 - When paying with a MasterCard, all members of the loyalty program receive a 10% discount.
 - Discounts do not add up.

1. Equivalence Partitioning

To create an optimal test coverage for the discount calculator using the Equivalence Partitioning test design technique, we can divide the input conditions into equivalent classes where the system should behave similarly for each class.

Years of Loyalty

Class	Values	Discount
Class 1	years >= 1	Eligible for discounts
Class 2	years < 1	Expect no discount

Payment Method

Class	Values	Discount
Class 1	MasterCard	Expect a 10% discount (if other conditions are met)
Class 2	Visa, cash, etc.	No specific discount for payment method

2. Boundary Value Analysis

To create an optimal test coverage for the discount calculator function using the Boundary Value Analysis technique, we should consider the boundary conditions based on the customer requirements

Years of Loyalty

- Just below the boundary: 1 year-1 day – Expect no discount.
- On the boundary: 1 year – Eligible for discounts.
- Just above the boundary: 1 year+1 day – Eligible for discounts.

Birthday

- Just below the boundary: Birthday-1 day – Expect no discount.
- On the boundary: Birthday – Eligible for discounts.
- Just above the boundary: Birthday+1 day – Expect no discount.

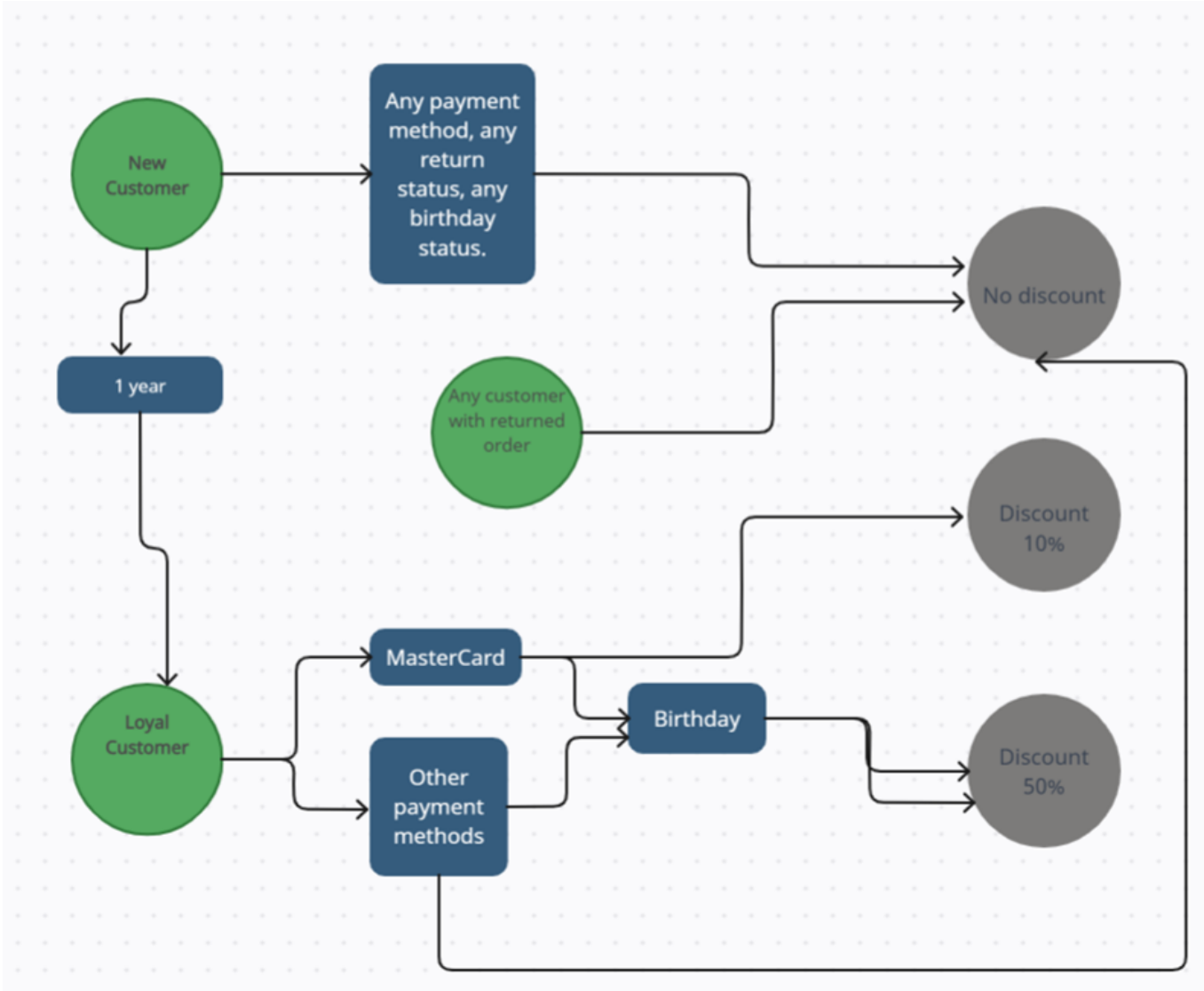
3. Decision Table Testing

Decision Table Testing is a technique used to test combinations of input conditions. In this case, the input conditions are the years, Payment Method, Birthday and Returned Order, while the output condition is the discount. We can create a decision table to cover various combinations of these input conditions based on the given requirements.

Years >= 1	Payment Method is MasterCard	Returned Order	Birthday	Expected Action
Yes	Yes	No	Yes	Apply 10% Discount
Yes	No	No	Yes	Apply No Discount
Yes	Yes	Yes	Yes	Apply No Discount
Yes	No	Yes	Yes	Apply No Discount
Yes	Yes	No	No	Apply 50% Discount
Yes	No	No	No	Apply 50% Discount
Yes	Yes	Yes	No	Apply No Discount
Yes	No	Yes	No	Apply No Discount
No	Yes	No	Yes	Apply No Discount
No	No	No	Yes	Apply No Discount
No	Yes	Yes	Yes	Apply No Discount
No	No	Yes	Yes	Apply No Discount
No	Yes	No	No	Apply No Discount
No	No	No	No	Apply No Discount
No	Yes	Yes	No	Apply No Discount
No	No	Yes	No	Apply No Discount

4. Cause-Effect

To create an optimal test coverage for the given discount calculator based on the Cause-Effect Analysis technique, we need to consider various scenarios and conditions.



5. Error Guessing

To effectively test the discount calculator function for the loyalty program using the Error Guessing test design technique, we should consider various scenarios and edge cases based on the requirements and the function's logic. The aim is to identify potential errors or unexpected behavior.

- Scenarios:
- Customer's birthday changes
 - Leap year
 - Discount cumulation

6. Pairwise Testing

In pairwise testing we focus on creating test cases that cover all possible pairs of input combinations. This is particularly useful in this context to ensure that every possible pair of conditions is tested at least once.

Years of Purchase History (>=1 year)	Payment Method	Returned Order	Is Birthday Today	Expected Discount
Yes	MasterCard	No	No	10%
Yes	MasterCard	No	Yes	50%
Yes	MasterCard	Yes	No	0%
Yes	MasterCard	Yes	Yes	0%
Yes	Other	No	No	0%
Yes	Other	No	Yes	50%
Yes	Other	Yes	No	0%
Yes	Other	Yes	Yes	0%
No	MasterCard	No	No	0%
No	Other	Yes	Yes	0%

7. State Transition Testing

To effectively design test cases using State Transition Testing for the discount calculator based on the loyalty program, we need to consider the different states and transitions that affect the discount calculation. The primary states are based on the years of being a customer, the payment method, whether an order was returned, and if it's the customer's birthday.

- New client -> Loyal client
- MasterCard Payment Method -> Other Payment Method -> MasterCard Payment Method
- No returned order -> Returned order -> No returned order
- No Birthday -> Is Birthday Today -> No Birthday