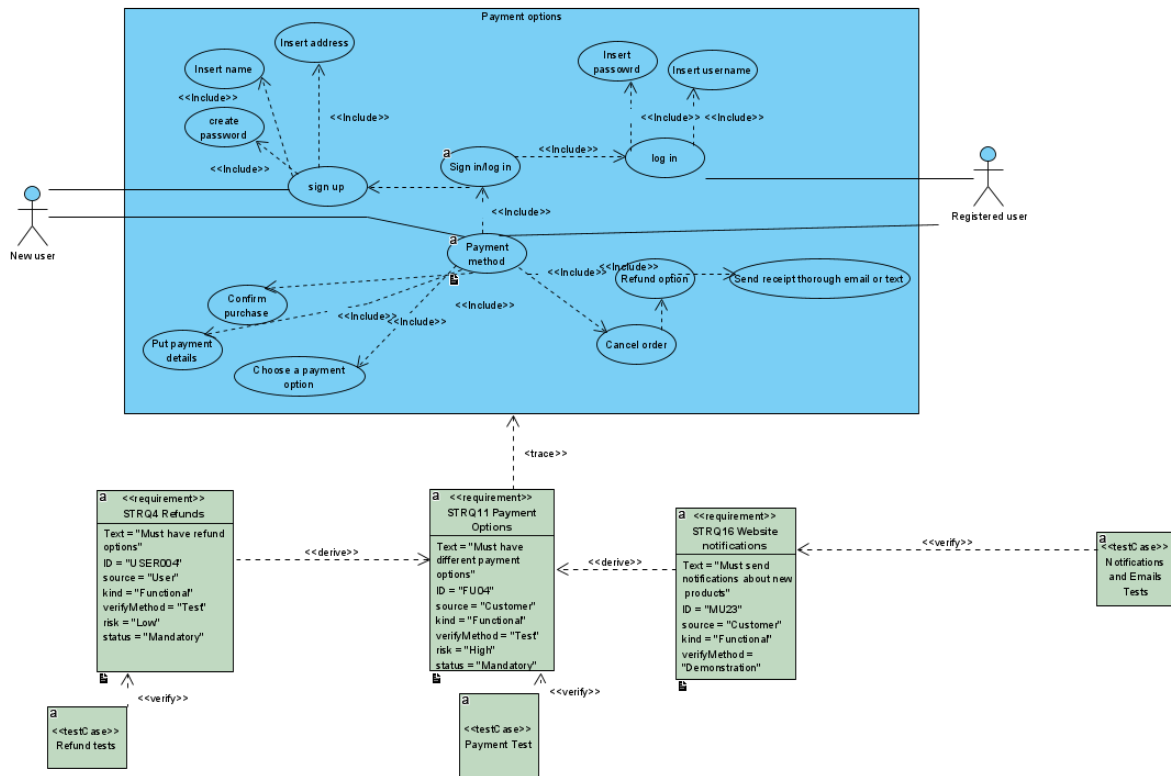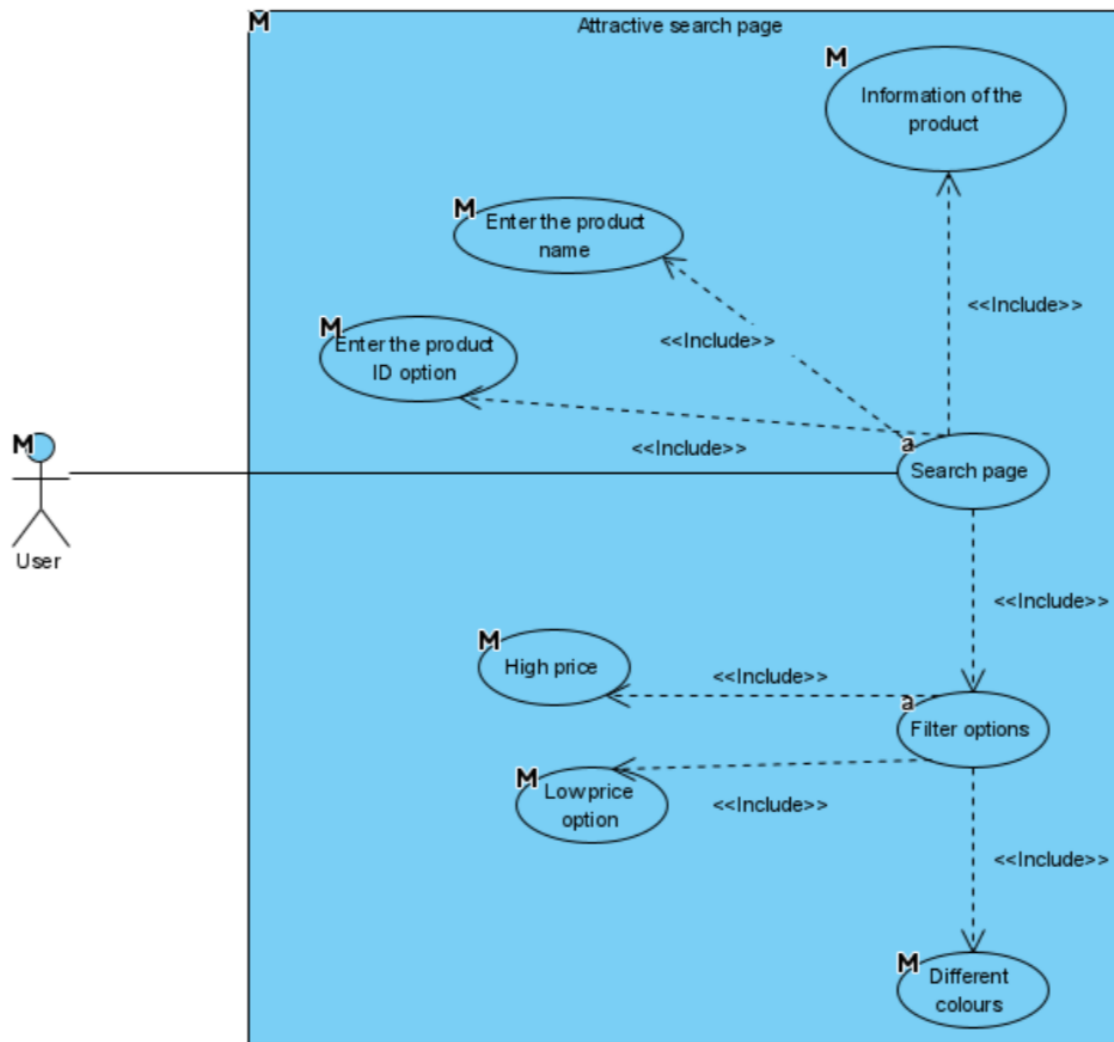# Use cases

Use case 1:

Use Case 1:

- Payment options and sign in : The users would like to have options on how to pay, varying from signing into their existing accounts or creating one. Users would like to purchase items quickly and have the option of cancelling confirmed orders then getting their refunds.
Exception: Users would not be able to access this feature without steady and fast internet;
Alternative:an alternative way of paying is going in store to purchase.

Use case 2:

Attractive search page

Information of the
product

Enter the product
name

Enter the product
ID option

<<Include>>

<<Include>>

<<Include>>

Search page

User

<<Include>>

High price

<<Include>>

Filter options

Low price
option

<<Include>>

<<Include>>

Different
colours

Attractive search page

- Information of the product
- Enter the product name
- Enter the product ID option
- Search page
- High price
- Filter options
- Low price option
- Different colours

User

<<trace>>
<<Include>>
<<Include>>
<<Include>>

a <<requirement>>
STRQ7 Inventory Count
Text = "Must have a count of the products available"
ID = "INT09"
source = "Customer"
kind = "Interface"
verifyMethod = "Test"
risk = "Low"

a <<requirement>>
STRQ3 Interface Appearance
Text = "Must be user friendly and attractive"
ID = "USER002"
source = "User"
kind = "Functional"
verifyMethod = "Test"
risk = "Low"
status = "Approved"

a <<testCase>>
Count of inventory test

a <<testCase>>
Display Test

<<derive>>
<<verify>>
<<verify>>
<<derive>>

a <<requirement>>
STRQ10 Brand Variety
Text = "Must have different brands"
ID = "FU 11"
source = "Customer"
kind = "Functional"
verifyMethod = "Test"
risk = "Low"
status = "Mandatory"

a <<testCase>>
Comparison Tool Test

a <<requirement>>
STRQ2 Comparison Tool
Text = "Must have easy comparison tool"
ID = "USER021"
source = "User"
kind = "Functional"
verifyMethod = "Test"
risk = "Low"

<<verify>>
<<derive>>

a <<testCase>>
Brand Database

a <<requirement>>
STRQ9 Price Database
Text = "Must have all databases for prices"
ID = "INT15"
source = "Customer"
kind = "Interface"
verifyMethod = "Test"
risk = "Low"
status = "Mandatory"

<<derive>>
<<verify>>

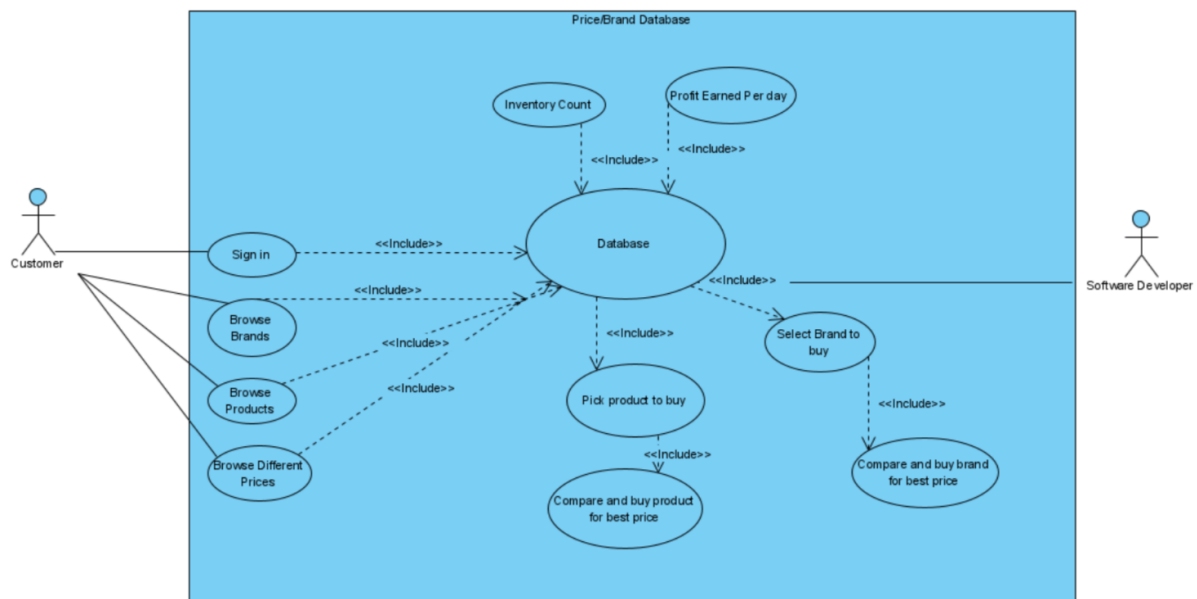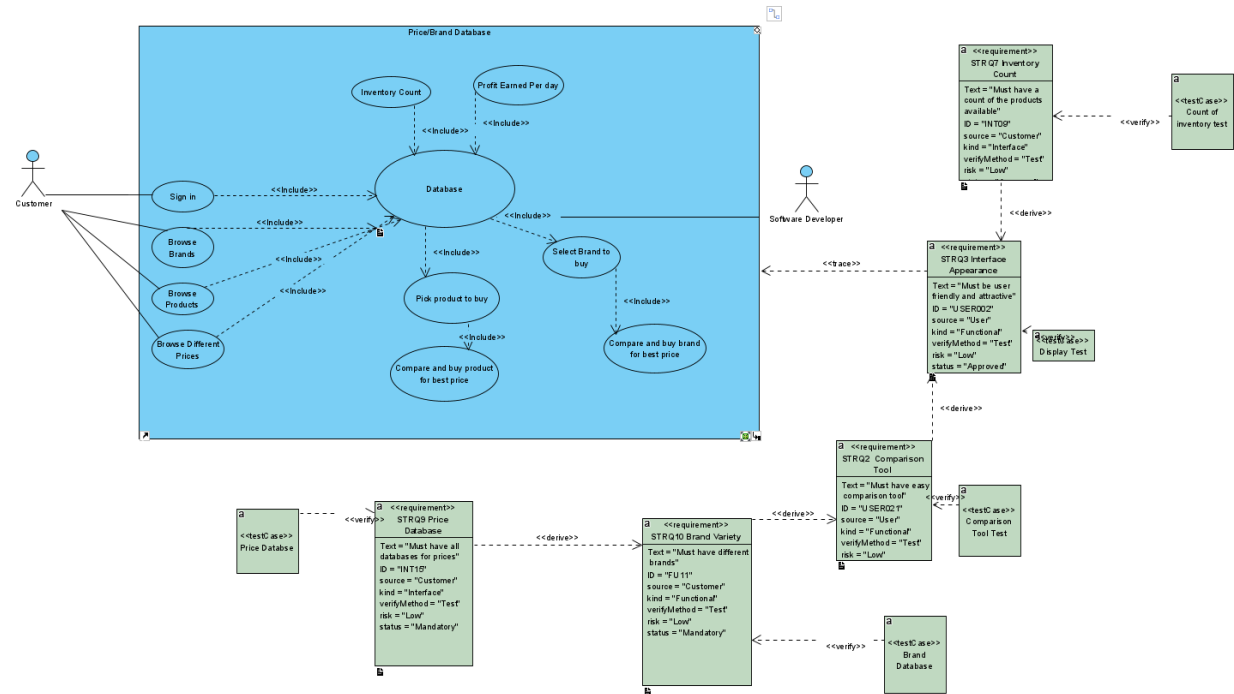a <<testCase>>
Price Databse

Description

- Attractive search page: The users want an engaging search page where they can easily look for products they desire by inserting the product ID or product name and also be able to filter the products by price(high or low); also by color. This way, they get better information about their desired product.
  Exception: Every user has different choices on what they see as attractive; this might make it difficult to know if the pages are attractive or not.
  Alternative: Using a plain search page to avoid conflict of ideas

Use case 3

Price/Brand Database (use case diagram)

Customer — Sign in, Browse Brands, Browse Products, Browse Different Prices
Use cases: Database, Inventory Count, Profit Earned Per day, Select Brand to buy, Pick product to buy, Compare and buy brand for best price, Compare and buy product for best price — connected via <<Include>>
Software Developer

Requirement / test case boxes:

<<requirement>> STRQ7 Inventory Count
Text = "Must have a count of the products available"
ID = "INT09"
source = "Customer"
kind = "Interface"
verifyMethod = "Test"
risk = "Low"

<<testCase>> Count of inventory test   (<<verify>>)

<<requirement>> STRQ3 Interface Appearance
Text = "Must be user friendly and attractive"
ID = "USER002"
source = "User"
kind = "Functional"
verifyMethod = "Test"
risk = "Low"
status = "Approved"

<<testCase>> Display Test   (<<verify>>)

<<requirement>> STRQ2 Comparison Tool
Text = "Must have easy comparison tool"
ID = "USER021"
source = "User"
kind = "Functional"
verifyMethod = "Test"
risk = "Low"

<<testCase>> Comparison Tool Test   (<<verify>>)

<<requirement>> STRQ9 Price Database
Text = "Must have all databases for prices"
ID = "INT15"
source = "Customer"
kind = "Interface"
verifyMethod = "Test"
risk = "Low"
status = "Mandatory"

<<testCase>> Price Databse   (<<verify>>)

<<requirement>> STRQ10 Brand Variety
Text = "Must have different brands"
ID = "FU11"
source = "Customer"
kind = "Functional"
verifyMethod = "Test"
risk = "Low"
status = "Mandatory"

<<testCase>> Brand Database   (<<verify>>)

Relationships between requirements: <<trace>>, <<derive>>

Price/Brand Database (use case diagram)

Customer — Sign in, Browse Brands, Browse Products, Browse Different Prices
Use cases connected via <<Include>>: Database, Inventory Count, Profit Earned Per day, Select Brand to buy, Pick product to buy, Compare and buy product for best price, Compare and buy brand for best price
Software Developer

Description:

- Price/Brand Database: The customers have the option to browse through an inventory of products they want; at the same time comparing the prices of one product with another.
  Exception: Sometimes users don't know what product to buy, so they spend time searching for their choice .
  Alternative: Instead of a search page, users could use a live chat option, where one of our employees can assist them and help them in finding their choice of product.