Uses cases description

## Log in (Admin - Customer)

**Description :**

Allows administrator and customer to use their username and password to connect to application. Allows user to have extra settings compared to a unregistered user..

**Risks :** 2

**Business value :** 0

## Logout (Admin - Customer)

**Description :**

Allows administrator and customer to secure their account and not allow anyone else access.

**Risks :** 2

**Business value :** 0

## Search product (Admin, Customer)

**Description :**

Allow customer or adminisrator to enquire about details of a certain product. By supplying the application with the UPC(Universal Packing Code) the application will return all relevant details.

**Risks :** 1

**Business value :** 4

## Stock Enquiry (Admin)

**Description :**

Allow administrator to check the current stock level of a certain product. Just like the ‘Search Product’ function the application will ask for the UPC and will then display the stock level to the administrator.

**Risks :** 4

**Business value :** 4

## Add product to basket (Customer)

**Description :**

Allows a customer to add many products to the basket and allow them to continue shopping without have to pay for anything. It will allow the customer to complete their transaction will all their desired products at the same. Effectively giving the customer a easy to use purchasing system and allowing them to purchase multiple products in one transaction.

**Risks :** 2

**Business value :** 3

## Remove product from basket (Customer)

**Description :**

Allows a customer to make modifications to their basket by removing a product they no longer wish to buy or have added to the basket accidentally. Another function that will give a user more control over their transaction.

**Risks :** 2

**Business value :** 3

## Check Out (Customer)

**Description :**

Allows a customer to complete their transaction by checking out. This could be with a basket of products or with a single product. User will process the transaction the same way and will be required to fill out their payment details exactly the same way.

Details of user will be saved for future transactions and for details about the payment.

**Risks :** 4

**Business value :** 4

## Add new product (Admin)

**Description :**

Allow administrator to add a new product for purchase to the product database. Product will be given a product ID and all necessary data before it can be entered into the table.

**Risks :** 2

**Business value :** 4

## Remove product (Admin)

**Description :**

Allow administrator to simply remove a product from the system without reason. Reasons for removing the products could be if the product was not physically in stock but showed stock on the database. ‘Remove Product’ would be a way or correcting the count in the database.

**Risks :** 2

**Business value :** 4

## Waste product (Admin)

**Description :**

Allows an administrator to waste a damaged or date expired product. Allows administrator to give reason for wasting the product

**Risks :** 2

**Business value :** 2

# Casual Description

## Use case number 1 : Log in (admin, customer)

User must have an account in the system before starting this use case.

The administrator or the customer must enter his login and his password and confirm.

The system check if the login and password already exists.

If they exists, the user is connected into the application and he arrives in application’s home.

If login and password doesn’t exists, the system will display an appropriate error message and the user can try again to log in with good login and password.

The system searchs to find if there is a good login and password, it must not exceed 10 seconds, beyond this time, an appropriate error message will display to warn the user.

## Use case number 4 : Search product (admin, customer)

User must have an account in the system before starting this use case.

The user logs onto the system and may search products they want. The user searches by name, price, category, origin. The system will search a correspondence with the criteria previously entered.

If the system finds at least one correspondence, the system will display the products found.

If no elements fit the search, the system displays an appropriate message to warn the user. If the user entered bad characters in the search bar, an appropriate error message will be displayed.

The search must not exceed 10 seconds, beyond this time, an appropriate error message will be displayed to warn the user.

**Fully Dressed Use Case**

**Use case number 7: Check Out (Customer)**

**Description:**

Allows a customer to complete their transaction by checking out. This could be with a basket of products or with a single product. User will process the transaction the same way and will be required to fill out their payment details exactly the same way.

Details of user will be saved for future transactions and for details about the payment.

**Level:**

User-Level goal (EBP level)

**Primary Actor:**

Customer

**Stakeholders and interests:**

Adminstrator: Wants customer to use this function and purchase products to generate income

Customer: Wants a fast and easy to use application to buy their desired products

Accountant: Wants a record of each transaction that occurs

**Success guarantee:**

* User's payment method had sufficient funds
* Products stock level must be altered

**Main Success Scenario:**

User has used the ‘Check Out’ function to purchase all the items in their basket. The transaction has been completed without error and all parties are happy

**Extensions:**

* User entered wrong payment details and was displayed with an error message when they tried to finish the transaction.
* Products that customer tried to buy did not have stock
  + Products could have been bought after customer added them to basket
* Customer decided they did not want to buy certain product so had to use the ‘Remove Product From Basket’ function.

**Preconditions:**

* In order for this use case to work the user must enter the correct details required of them.
* The product or products they wish to buy must be in stock.
* The payment details given must be valid

**Frequency of Occurrence:**

This function should be used very frequently. This is the only way for a customer to purchase products so ideally every customer that uses the application should use this function

**Complexity:**

Function should be moderate enough to develop. No aspect is too difficult. With a good design there should not be any problems. Easiest solution would be to identify all the requirements before coding begins.

2 functions chosen are ‘Waste Product’ and ‘Check Out’