Software Requirements Specification

Point Of Sales System

Minimarket Retail

By :

Komang Arinanda

Nikolaus Aldo Halim

Blibli FUTURE Program Batch 2.0

**August 2017**

**Table Of Contents**

Table Of Contents..........................................................................................................................2

System Description........................................................................................................................3

Use Case Description.....................................................................................................................5

A. Account Management .................................................................................................5

1. Login................................................................................................................5

2. Account Management.....................................................................................6

3. Change Password............................................................................................7

B. Store Information Management..................................................................................8

C. Staff Management.......................................................................................................9

1. Show Staff.......................................................................................................9

2. Add Staff........................................................................................................10

3. Change Staff..................................................................................................11

4. Change Staff..................................................................................................12

D. Product Management................................................................................................13

1. Category Management..................................................................................13

a. Show Category..................................................................................13

b. Add Category….................................................................................14

c. Change Category...............................................................................15

d. Delete Category................................................................................16

2. Product Management....................................................................................17

a. Show Product....................................................................................17

b. Add Product......................................................................................18

c. Change Product.................................................................................19

d. Delete Product..................................................................................20

E. Stock Management.....................................................................................................21

1. Show Stock Product.......................................................................................21

2. Add Stock.......................................................................................................22

F. Transaction.................................................................................................................23

1. Input Transaction Data..................................................................................23

2. Delete Transaction Record.............................................................................24

3. CancelTransaction..........................................................................................25

4. Payment.........................................................................................................26

G. Promotion Management............................................................................................27

1. Show Promotions…………………………………………………….............…………………….27

2. Add Promotions .............................................................................................28

a. Add Product Discount Promotion......................................................28

b. Add Total Discount Promotion..........................................................29

c. Add Buy x Get Y Promotion...............................................................30

3. Change Promotions........................................................................................31

4. Delete Promotions.........................................................................................32

H. Reports.......................................................................................................................33

1. Transaction Report........................................................................................33

2. Sales Statistic Report.....................................................................................34

System Description

Point Of Sales System is a system that is used to handle every transaction like item transaction, accounting, and stock management.

There are two roles of user such like administrator and Cashier.

Every function that can be used in our system are descripted in this image below:



First there is the Login function, if the user login as a Cashier, they could use some function such as:

1. Account Management

User could change their photo, name, e-mail, and password.

1. Sales Management

User could input selling data, doing the payment process, and print the receipt.

The Cashier is a generalisation from the Administrator, hence the Administrator could use every single function the Cashier could. These are every function that exclusively the Administrator could use:

1. Store Information Management

User could change every information of the store, such as name, address, telephone number, e-mail, profit margin, and receipt information (if necessary).

1. Employee Management

User could add, change, remove, show and search every employees in the store.

1. Supplier Management

User could add, change, remove, show and search every supplier that supply the items in the store.

1. Stock Management

User could add, change, remove, show and search every item in the store.

1. Discount Management

User could add, change, remove, show and search every discount.

1. Supply Management

User could buy supplies from supplier.

1. Sales Report

User could take a look at the daily, monthly and annual report.

Use Case Description

**A. Account Management**

1. Login

|  |  |
| --- | --- |
| **Use Case** | Login |
| **User** | Admin atau Kasir |
| **Description** | Use case describes how the user log into the system |
| **Pre Condition** | User must already have their ID and Password on the database |
| **Flow Of Event** | 1. Started when the user accessed the system login page |
| 2. The system shows the login page and ask for the user ID and the Password. |
| 3. User input the ID and password |
| 4. User clicks the login button |
| 5. The system checks the ID and the Password |
| 6. The system pops up a logged in message |
| 7. The system checks the user’s role |
| 8. The system shows the main page as the user’s role |
| 9. Use case finished |
| **Post Condition** | The system shows the main page for the user based on their role |
| **Alternative Flow** | At number 3, if the user inputs the wrong ID/Password, the system will asks for another one. |

1. Account Management

|  |  |
| --- | --- |
| **Use Case** | Account Management |
| **Actor** | User |
| **Description** | Use case shows how user change their photo, name and email. |
| **Pre Condition** | User has been authorized, and have some information to be changed |
| **Flow Of Event** | 1. Use case beginss when user clicks on the “Account Management” button. 2. System shows the main page that consisted of their photo, name, and email 3. User can renew their photo or name with a new one. 4. User can input a new email 5. The system validates the email 6. User clicks the “save” button 7. The system shows a confirmation message 8. User clicks the “yes” button. 9. The system renew the information 10. The system shows a “success message” 11. Use case finished |
| **Post Condition** | Account information like photo, name or email is changed. |
| **Alternative Flow** | On the 4th step, if the user inputs a non-valid email, the system asks for a new one  On the 8th step, if the user clicks the “no” button, the process will be aborted and go back to step 1. |

1. Change password

|  |  |
| --- | --- |
| **Use Case** | Change password |
| **Actor** | User |
| **Description** | Use case shows how user change their account password |
| **Pre Condition** | User access the account management page, and clicks the “change password” button, the user have the old and the new password |
| **Flow Of Event** | 1. Use case beginss when the user access the “account management” page and clicks the “change password” button. 2. The system shows the form 3. User inputs the old and the new password and re-type the new password 4. User clicks the “save” button 5. The system validates the old password 6. The system validates the new password and the re-type 7. The system shows the confirmation message 8. User clicks “yes” button 9. The system renew password 10. The system shows the success message 11. Use case finished |
| **Post Condition** | Password renewed |
| **Alternative Flow** | At step 3, if user inputs the wrong password yang salah, the system will asks for another one.  At step 3, if user inputs a different password at the re-type, the system will asks for another re-type  At step 8 if user clicks the “no” button, the system aborts a the process and go to step 1 |

B. Store Information Management

|  |  |
| --- | --- |
| **Use Case** | Store Information Management |
| **Actor** | Admin |
| **Description** | Use case shows how Admin minimarket manages the information of the store such as name, address, phone number, email and receipt information. |
| **Pre Condition** | Admin should have been authorized by the system |
| **Flow Of Event** | 1. Use case beginss when the admin enters the “Store Information Management” page.  2. The system shows the old logo, name, address, phone number, email and receipt information and asks for the new one.  3. Admin inputs the new logo, name, address, phone number, email and receipt information and clicks the “save” button  4. The system shows the confirmation message  5. Admin confirms the update by clicking the ”yes” button  6. The system renews the changed information  7. Use case finished |
| **Post Condition** | The system renews the information of the store such as name, address, phone number, email and receipt information. |
| **Alternative Flow** | At step 5, if the Admin chose “no” the system aborts the process and the use case finished |

C. Staff Management

1. Show Staff

|  |  |
| --- | --- |
| **Use Case** | Show Staff |
| **Actor** | Admin |
| **Description** | Use case shows how Admin find and show the staffs |
| **Pre Condition** | Admin access the Staff Management page and owns the ID and the name of the staff they want to find |
| **Flow Of Event** | 1. Use case begins when the Admin wants to see, change, or delete a staff and access the Staff Management page.  2. The system asks for the ID and the name  3. Admin inputs the staff’s ID and the name  4. The system searches the database for the information  5. The system shows the staff’s ID, Name, Role, and email  6. Use case finished |
| **Post Condition** | The system shows the staff’s ID, Name, Role, and email |
| **Alternative Flow** | Admin can export the result into a document file by clicking the “”export” button, and the system will exports it.  At step 4, if The system doesn’t find the ID or the name, it will shows an error message |

2. Add Staff

|  |  |
| --- | --- |
| **Use Case** | Add Staff |
| **Actor** | Admin |
| **Description** | Use case shows how Admin add a new staff |
| **Pre Condition** | Admin access the Staff Management page and have the new information of the new staff. |
| **Flow Of Event** | 1. Use case begins when Admin choose the “add” button at the Staff Management page.  2. The system asks for the new information  3. The system new checks the staff’s code.  4. Admin inputs the name  5. Admin inputs the role  6. The system “password” sets the password with “password”  7. Admin inputs the email  8. Admin clicks the “add” button  9. The system shows the confirmation m  essage  10. Admin clicks the “yes” button  11. The system save the information of the new staff  12. Use case finished |
| **Post Condition** | The system save the information of the new staff |
| **Alternative Flow** | At step 3, if Admin inputs a wrong ID, the system will show an error message and asks for a new one  At step nomor 9, if Admin clicks “no”, the system aborts the process and go to step 1 |

3. Change Staff Information

|  |  |
| --- | --- |
| **Use Case** | Change Staff Information |
| **Actor** | Admin |
| **Description** | Use case shows how Admin changes a staff’s information |
| **Pre Condition** | Admin access the Staff Management Page and have the new ID or name of the staff and the new information. |
| **Flow Of Event** | * + - 1. Use case begins when the Admin select the staff to change the indormation.  1. The system searches the staff’s dataAdmin clicks button Ubah 2. The system shows the form 3. Admin inputs the new name 4. Admin inputs the new role 5. Admin inputs the new email 6. The system shows the confirmation message 7. Admin clicks the “Yes” button 8. The system renews the name, role and email. 9. Use case finished |
| **Post Condition** | The system renews the name, role, and email |
| **Alternative Flow** | At step 9, if the Admin selects “No” , the system aborts the process and go to step 1. |

4. Delete Staff

|  |  |
| --- | --- |
| **Use Case** | Delete Staff |
| **Actor** | Admin |
| **Description** | Use case shows how the Admin deletes a staff’s data |
| **Pre Condition** | Admin access the Staff Management Page and have the ID and the name of the staff to be deleted |
| **Flow Of Event** | 1. Use case begins when the Admin clicks the “delete” button  2. Admin selects the information of the staff to be deleted  3. The system searches the staff’s information  4. The system shows the “confirmation” message  5. Admin clicks the “Yes” button  6. The system deletes the information  7. Use case finished |
| **Post Condition** | The system deletes the information |
| **Alternative Flow** | At step 5, if the Admin selects “No” , the system aborts the process and go to step 1. |

D. Product Management

1. Category Management

a. Show Categories

|  |  |
| --- | --- |
| **Use Case** | Show Categories |
| **Actor** | Admin |
| **Description** | Use case shows how the Admin searches and the system shows the categories available |
| **Pre Condition** | The admin choose the Category Management page, and have the name or the code of the desired category to be searched |
| **Flow Of Event** | 1. Use case begins when Admin wants to see, edit, or delete a category.  2. The system meminta inputan berupa Kode atau Name Category yang dicari asks the user to input the code or the name of the desired category.  3. The admin inputs the code or the name of the desired category  4. The system Category searches the database for the category  5. The system shows code, name, and the description  6. Use case finished |
| **Post Condition** | The system shows code, name, and the description |
| **Alternative Flow** | The admin can Export the output to a document file by clicking the “export” button, then the system will export the data to a document file.  At step 4, if The system cannot find the code or the name of the category, the system will show an “error” message. |

b. Add Category

|  |  |
| --- | --- |
| **Use Case** | Add Category |
| **Actor** | Admin |
| **Description** | Use case shows how the admin add a new category |
| **Pre Condition** | The admin choose the Category Management page and have the data of the new category |
| **Flow Of Event** | 1. Use case begins when the admin choose the “add category” button  2. The system asks for a new code for the category  3. the admin inputs a new category code  4. The system checks if the code is unique or not  5. the admin inputs a new category name  6. the admin inputs the category description  7. The system shows the confirmation message  8. User clicks the “yes” button  9. The system save the data of the new category  10. Use case finished |
| **Post Condition** | The system save the data of the new category |
| **Alternative Flow** | At step 4, if the code is not unique, the system will show an “error” message and then asks for a new code.  At step 8, if the admin choose “no” , the system will abort the process and go back to step 1. |

c. Change Category

|  |  |
| --- | --- |
| **Use Case** | Change Category |
| **Actor** | Admin |
| **Description** | Use case shows how the admin changes a category |
| **Pre Condition** | Admin choose page Category Management and memiliki kode atau name Category yang akan diubah serta name atau deskripsi new |
| **Flow Of Event** | 1. Use case begins when the admin choose a category to be changed, also they could search for it.  2. The system searches the category  3.Admin clicks the “change” button  4. The system shows the form  5. Admin inputs the new category name  6. Admin inputs the new description  7. The system shows the “confirmation” message  8. The admin clicks the “yes” button  9. The system renew the name and the description  10. Use case finished |
| **Post Condition** | The system renew the name and the description |
| **Alternative Flow** | At step 6, if the admin choose “no”, the stystem will abort the process and go back to step 1 |

d. Delete Category

|  |  |
| --- | --- |
| **Use Case** | Delete Category |
| **Actor** | Admin |
| **Description** | Use case shows how the admin deletes a category |
| **Pre Condition** | The admin choose the Category Management page and have the code or the name of the category to be deleted, and the category does not have any product on it. |
| **Flow Of Event** | 1. Use case begins when the admin inputs the code or the name of the category to be deleted  2. The system searches the category  3. The system validates if the category doesn’t have any product  4. The system shows the confirmation message  5. The admin clicks the “yes” button  6. The system deletes the category  7. Use case finished |
| **Post Condition** | The system deletes the category |
| **Alternative Flow** | At step 3, if there is a product on the category, the process is failed and the system shows an “error” message and go to step 1  At step 5, if the admin choose “no”, the system aborts the process and go back to step 1 |

2. Product Management

a. Show Product

|  |  |
| --- | --- |
| **Use Case** | Show Product |
| **Actor** | Admin |
| **Description** | Use case shows how the admin searches and see products |
| **Pre Condition** | The admin access the Product Management page and have the code or the name of the desired product to be shown |
| **Flow Of Event** | 1. Use case begins when the admin wants to see, change, or delete a product  2. The system ask for the code or the name of the category  3. Admin inputs the code or the name of the desired product  4. The system searches the product  5. The system shows the code, the category, the product name, description, stock, and the price  6. Use case finished |
| **Post Condition** | The system shows the code, the category, the product name, description, stock, and the price |
| **Alternative Flow** | The admin can export the results into a document file by clicking the “export” button, the system will export the data into a document file  At step 4, if the system cannot find the code or the name, it will aborts the process and ask for another code or name |

b. Add Product

|  |  |
| --- | --- |
| **Use Case** | Add Product |
| **Actor** | Admin |
| **Description** | Use case shows how the admin adds a new product |
| **Pre Condition** | Admin choose the “Product Management” page and have the data of the desired product |
| **Flow Of Event** | 1. Use case begins when the system asks for the code of the desired product to be added  2. The system checks if the code is unique  3. Admin inputs the category of the added product  4. Admin inputs the product name  5. Admin inputs the product description  6. The system initialize the stock with 0  7. The system initialize the price with 0  8. The system shows the “confirmation” message  9. Admin choose “yes”  10. The system add the new product to the database  11. Use case finished |
| **Post Condition** | The system add the new product to the database |
| **Alternative Flow** | At step 2, if the code is not unique, or already taken by other product, the system will show an “error” message and ask for another code  At step 9, if the admin choose “no”, the system will abort the process and go back to step 1 |

c. Change Product

|  |  |
| --- | --- |
| **Use Case** | Change Product |
| **Actor** | Admin |
| **Description** | Use case shows the admin change a product information |
| **Pre Condition** | Admin the Product Management Page, and have the code or name of the desired product to be changed, and the new information such as category, name, and description |
| **Flow Of Event** | 1. Use case begins when the admin choose the product to be changed which can be searched first.  2. The system searches the product data  3. Admin clicks the “change” button  4. The system shows the form  4. Admin inputs the new category name  5. Admin inputs the new name  6. Admin inputs the new description  8. The system shows the confirmation message  9. The admin clicks the “yes” button  10. The system renew the name, category, and the description of the product  11. Use case finished |
| **Post Condition** | The system renew the name, category, and the description of the product |
| **Alternative Flow** | At step 6, if the admin choose “no”, the system aborts the process and go back to step 1 |

d. Delete Product

|  |  |
| --- | --- |
| **Use Case** | Delete Product |
| **Actor** | Admin |
| **Description** | Use case shows how the admin deletes a product |
| **Pre Condition** | The admin choose the Product Management Page and have the code or the name of the desired product to be deleted |
| **Flow Of Event** | 1. Use case begins when the admin inputs the code or the name of the desired product to be deleted  2. The system searches the product data  3. The system checks if the stock is empty  4. The system shows the “confirmation” message  5. The admin clicks the “yes” button  6. The system deletes the product  7. Use case finished |
| **Post Condition** | The system deletes the product |
| **Alternative Flow** | At step 3, if the stock is not empty, the process will be aborted and go back to step 1  At step 5, if the admin “no”, the process will be aborted and go back to step 1 |

E. Mengelola Stock

1. Show Product Stock

|  |  |
| --- | --- |
| **Use Case** | Show Product Stock |
| **Actor** | Admin |
| **Description** | Use case shows how the admin search and the system shows the product stock |
| **Pre Condition** | Admin choose the Stock Management page and have the code or the name of the product to be shown |
| **Flow Of Event** | * + - 1. Use case begins when the admin inputs the code, name, or the category of the product to be shown  1. The system searches the product 2. The system shows the code, name, category, and the stock 3. Use case finished |
| **Post Condition** | The system shows the code, category, name, and the stock |
| **Alternative Flow** | The admin could export the results into a document file by clicking the “Export” button. The system will export the data into a document file  At step 2, if the system can’t find the code, name, or the product to be searched, the system will show an “error” message. |

2. Add Stock

|  |  |
| --- | --- |
| **Use Case** | Add Stock |
| **Actor** | Admin |
| **Description** | Use case shows how the admin adds the stock for a product |
| **Pre Condition** | The admin access the Stock Management page and have the code or the name of the desired product to be added |
| **Flow Of Event** | * + - 1. Use case begins when the admin inputs the code, or the product name of the desired product to be added  1. Sistem searches the database 2. The admin inputs the product quantity 3. The system validates the quantity and shows the “confirmation” message 4. The admin choose “yes” 5. The system add the quantity into the stock 6. Use case finished |
| **Post Condition** | The system add the quantity into the stock |
| **Alternative Flow** | At step 1, if the code is not listed on the database, the process will be aborted and it will ask for a listed one  At step 3, if the quantity is not valid, the system will ask for a valid one  At step 5, if the admin choose “no”, the system will abort the process and go to step 1 |

F. Transaction

1. Input transaction data

|  |  |
| --- | --- |
| **Use Case** | Input transaction data |
| **Actor** | User(Cashier + Admin) |
| **Description** | Use case shows how the user adds the transaction data |
| **Pre Condition** | The user selects the Transaction page and have the code or the name of the desired product to be added |
| **Flow Of Event** | * + - 1. Use case begins when the user choose the Transaction page  1. The system generates the transaction number 2. The system sets the time, the name, and the ID of the user who do the process 3. The user inputs the product code 4. The system searched the product and shows the name 5. The user inputs the quantity 6. The system validates the quantity by checking the stock and shows the “confirmation” message 7. The user clicks the “+” button 8. The system calculates the promotion, and the subtotal 9. The system adds the product data 10. The system shows the items on a table, calculates tax, and the grand total 11. The system shows the grand total and asks the user if they want to add another product 12. User choose “no” 13. Use case finished |
| **Post Condition** | Sistem menambahkan stok produk |
| **Alternative Flow** | At step 3, if the time is wrong, the user could change it  At step 4, if the system cannot find the data, the system will ask for another one  At step 7, if the quantity is not valid, the system will ask for a valid one  At step 13, if the user choose “yes”, the process go to step 4 |

2. Hapus Record Transaksi

|  |  |
| --- | --- |
| **Nama Use Case** | Menghapus Record Transaksi |
| **Aktor** | User(Kasir/Admin) |
| **Deskripsi Singkat** | Use case menggambarkan bagaimana User menghapus record transaksi tertentu |
| **Pre Condition** | User telah menginputkan data transaksi sebelumnya dan mengetahui kode/ nama produk yang akan dihapus dari daftar transaksi |
| **Flow Of Event** | * + - 1. Use case dimulai ketika memilih data dengan mengklik pada record data yang akan dihapus kemudian mengklik button hapus atau menekan – pada keyboard       2. Sistem mendapatkan kode produk kemudian melakukan pencarian record produk tersebut  1. Sistem menampilkan pesan konfirmasi hapus 2. User memilih “Ya” pada pilihan yang muncul 3. Sistem menghapus record transaksi 4. Use case selesai |
| **Post Condition** | Sistem menghapus record transaksi |
| **Alternative Flow** | Pada langkah 4, jika User memilih “Tidak” pada pilihan yang muncul maka proses menghapus record transaksi dibatalkan dan proses kembali ke langkah 1 |

3. Batal Transaksi

|  |  |
| --- | --- |
| **Nama Use Case** | Batal Transaksi |
| **Aktor** | User(Kasir/Admin) |
| **Deskripsi Singkat** | Use case menggambarkan bagaimana User membatalkan transaksi yang sedang berjalan |
| **Pre Condition** | User telah menginputkan data transaksi sebelumnya |
| **Flow Of Event** | 1. Use case dimulai ketika User mengklik button Batal atau menekan ESC pada keyboard 2. Sistem menampilkan pesan konfirmasi pembatalan transaksi 3. User memilih “Ya” pada pilihan yang muncul 4. Sistem menghapus semua record transaksi, mengosongkan tabel, mengosongkan field kode, nama, dan jumlah produk 5. Use case selesai |
| **Post Condition** | Sistem menghapus semua record transaksi, mengosongkan tabel, mengosongkan field kode, nama, dan jumlah produk |
| **Alternative Flow** | Pada langkah 3, jika User memilih “Tidak” pada pilihan yang muncul maka proses menghapus record transaksi dibatalkan dan proses kembali ke langkah 1 |

4. Bayar

|  |  |
| --- | --- |
| **Nama Use Case** | Bayar |
| **Aktor** | User(Kasir/Admin) |
| **Deskripsi Singkat** | Use case menggambarkan bagaimana User melakukan proses pembayaran |
| **Pre Condition** | User telah menginputkan data-data transaksi |
| **Flow Of Event** | 1. Use case dimulai ketika User mengklik button Bayar atau menekan \* pada keyboard 2. Sistem meminta inputan jenis metode pembayaran(Tunai/ Debit/Kredit) 3. User menginputkan jenis metode Tunai 4. User menginputkan nominal bayar 5. Sistem memastikan nominal bayar tidak lebih kecil dari total akhir 6. Sistem menghitung dan menampilkan nominal kembalian 7. User mengklik button Bayar atau menekan tombol \* pada keyboard 8. Sistem menampilkan pesan konfirmasi Bayar 9. User memilih “Ya” pada pilihan yang muncul 10. Sistem menyimpan data transaksi dan mengurangi stok-stok produk terjual 11. Sistem mengenerate dan mencetak receipt 12. Use case selesai |
| **Post Condition** | Sistem menyimpan data transaksi dan mengurangi stok-stok produk |
| **Alternative Flow** | Pada langkah 3, jika user memilih jenis metode Debit atau Kredit maka Sistem meminta inputan nomor kartu dan proses langsung ke langkah 7  Pada langkah 9, jika User memilih “Tidak” pada pilihan yang muncul maka proses menghapus record transaksi dibatalkan dan proses kembali ke langkah 1 |

G. Mengelola Promo

1. Show Promo

|  |  |
| --- | --- |
| **Use Case** | Show Promo |
| **Actor** | Admin |
| **Description** | Use case shows how Admin melakukan proses pencarian and shows promo ataupun export hasil |
| **Pre Condition** | Admin choose page promo and memiliki kode promo yang akan dicari and Showkan |
| **Flow Of Event** | 1. Use case begins when Admin akan melihat, Change, atau Delete promo tertentu  2. The system meminta inputan berupa Kode and jenis promo  3. Admin inputs Kode promo  4. Admin choose jenis promo Diskon Barang  5. The system melakukan pencarian pencarian  6. The system shows Kode, name barang, tanggal mulai, berakhir, diskon, and status  7. Use case finished |
| **Post Condition** | The system shows Kode, name barang, tanggal mulai, berakhir, diskon, and status |
| **Alternative Flow** | Admin dapat melakukan Export hasil kedalam bentuk file dokumen dengan clicks buttn Export. The system akan mengeksport data Category kedalam file dokumen.  At step 5, if The system tidak menemukan kode atau name barang yang dicari maka The system akan shows pesan kesalahan bahwa kode atau name barang yang dicari tidak ditemukan  At step 4 if admin choose jenis Diskon Total Belanja maka data yang diShowkan adalah kode, tanggal mulai, berakhir, belanja minimal, diskon, and status.  At step 4 if admin choose jenis Diskon Buy x get y maka data yang diShowkan adalah kode, tanggal mulai, berakhir, name barang x, jumlah min, name barang y, jumlah, and status |

2. Add Promo

a. Add promo diskon barang

|  |  |
| --- | --- |
| **Use Case** | Add Promo diskon barang |
| **Actor** | Admin |
| **Description** | Use case shows how Admin Addkan promo diskon barang |
| **Pre Condition** | Admin choose page promo and clicks button Add, Admin memiliki data diskon yang akan diAddkan |
| **Flow Of Event** | 1. Use case begins when The system shows form Add diskon 2. Admin inputs Kode Promo 3. The system validates keunikan kode 4. Admin inputs tanggal mulainya promo 5. Admin inputs tanggal berakhir promo 6. The system validates tanggal mulai and berakhir 7. Admin choose jenis promo Diskon Barang 8. Admin menginput kode barang promo 9. The system shows name barang promo 10. Admin menginput persentase diskon 11. The system validates persentase diskon 12. The system shows pesan konfirmasi Add diskon 13. Admin choose Ya at pilihan yang muncul 14. The system mencatat data diskon 15. Use case finished |
| **Post Condition** | The system mencatat data promo Diskon Barang |
| **Alternative Flow** | At step 3, 6, and 11 if The system menilai bahwa inputan tidak valid mana The system shows pesan kesalahan and meminta input ulang  At step nomor 13, if Admin choose “Tidak” at pilihan yang muncul maka The system membatalkan penambahan data promo and proses kembali ke step 1 |

b. Add Promo Diskon Total Belanja

|  |  |
| --- | --- |
| **Use Case** | Add Promo diskon total belanja |
| **Actor** | Admin |
| **Description** | Use case shows how Admin Addkan promo total belanja |
| **Pre Condition** | Admin choose page promo and clicks button Add, Admin memiliki data diskon yang akan diAddkan |
| **Flow Of Event** | 1. Use case begins when The system shows form Add diskon 2. Admin inputs Kode Promo 3. The system validates keunikan kode 4. Admin inputs tanggal mulainya promo 5. Admin inputs tanggal berakhir promo 6. The system validates tanggal mulai and berakhir 7. Admin choose jenis promo Diskon Total Belanja 8. Admin menginput total belanja minimal 9. Admin menginput persentase diskon 10. The system validates persentase diskon 11. The system shows pesan konfirmasi Add diskon 12. Admin choose Ya at pilihan yang muncul 13. The system mencatat data diskon 14. Use case finished |
| **Post Condition** | The system mencatat data promo Diskon Barang |
| **Alternative Flow** | At step 3, 6, and 10 if The system menilai bahwa inputan tidak valid mana The system shows pesan kesalahan and meminta input ulang  At step nomor 13, if Admin choose “Tidak” at pilihan yang muncul maka The system membatalkan penambahan data promo and proses kembali ke step 1 |

c. Add Promo Buy x Get y

|  |  |
| --- | --- |
| **Use Case** | Add Promo Buy x Get y |
| **Actor** | Admin |
| **Description** | Use case shows how Admin Addkan promo Buy x Get y |
| **Pre Condition** | Admin choose page promo and clicks button Add, Admin memiliki data diskon yang akan diAddkan |
| **Flow Of Event** | 1. Use case begins when The system shows form Add diskon 2. Admin inputs Kode Promo 3. The system validates keunikan kode 4. Admin inputs tanggal mulainya promo 5. Admin inputs tanggal berakhir promo 6. The system validates tanggal mulai and berakhir 7. Admin choose jenis promo Buy x Get y 8. Admin menginput kode barang x 9. The system shows name barang x 10. Admin inputs jumlah barang minimum 11. The system validates jumlah 12. Admin inputs kode barang y 13. The system shows name barang y 14. Admin inputs jumlah barang y 15. The system melakukan validasi 16. The system shows pesan konfirmasi Add diskon 17. Admin choose Ya at pilihan yang muncul 18. The system mencatat data diskon 19. Use case finished |
| **Post Condition** | The system mencatat data promo Buy x Get y |
| **Alternative Flow** | At step 3, 6, 11, and 15 if The system menilai bahwa inputan tidak valid mana The system shows pesan kesalahan and meminta input ulang  At step nomor 17, if Admin choose “Tidak” at pilihan yang muncul maka The system membatalkan penambahan data promo and proses kembali ke step 1 |

3. Change Promo

|  |  |
| --- | --- |
| **Use Case** | Change Promo |
| **Actor** | Admin |
| **Description** | Use case shows how Admin Change data promo tertentu |
| **Pre Condition** | Admin berada at page promo memiliki kode promo yang akan diubah |
| **Flow Of Event** | 1. Use case begins when Admin choose promo yang akan diubah yang dapat dilakukan dengan melakukan pencarian terlebih dahulu(misal promo Diskon Barang) 2. Admin clicks button Ubah 3. The system shows form ubah promo beserta data lama 4. Admin dapat Change tanggal berakhir, and persentase diskon 5. The system shows pesan konfirmasi update 6. The admin clicks the “yes” button 7. The system mempernewi tanggal berakhir and atau persentase diskon and shows pesan berhasil 8. Use case finished |
| **Post Condition** | The system mempernewi tanggal berakhir and atau persentase diskon and shows pesan berhasil |
| **Alternative Flow** | At step 6, if Admin choose “Tidak” at pilihan yang muncul maka proses Change barang dibatalkan and proses kembali ke step 1  Apabila data yang diubah adalah jenis promo Diskon total belanja maka data yang dapat diubah adalah tanggal berakhir, total belanja minimal, and persentase diskon  Apabila daya yang diubah adalah jenis promo Buy x get y maka data yang dapat diubah adalah tanggal berakhir, jumlah minumum barang x, jumlah barang y |

4. Delete Promo

|  |  |
| --- | --- |
| **Use Case** | Delete Promo |
| **Actor** | Admin |
| **Description** | Use case shows how Admin Delete data promo tertentu |
| **Pre Condition** | Admin berada at page promo and memiliki kode promo yang akan dihapus |
| **Flow Of Event** | 1. Use case begins when Admin choose data promo yang akan dihapus 2. Admin clicks button Hapus 3. The system memastikan bahwa status promo tidak berlaku 4. The system shows pesan konfirmasi hapus 5. The admin clicks the “yes” button 6. The system Delete data promo 7. Use case finished |
| **Post Condition** | The system Delete data barang |
| **Alternative Flow** | At step 3, if status promo yang akan dihapus adalah berlaku maka proses hapus promo gagal and proses kembali ke step 1  At step 5, if Admin choose “Tidak” at pilihan yang muncul maka proses Delete promo dibatalkan and proses kembali ke step 1 |