**Use Case Template**

**Project Name:**

**Project ID:**

**Executive Sponsor:**

**Project Manager:**

**Business Analyst:**

Date: March 28, 2019

**Table of Contents**

[Approvals 3](#_Toc221528079)

[Use Case List 3](#_Toc221528080)

[1 Feature Name (Example: ATM Transaction) 3](#_Toc221528081)

[1.1 Feature Process Flow / Use Case Model 3](#_Toc221528082)

[1.2 Use Case(s) 3](#_Toc221528083)

[2 Feature Name (Repeat for each Use Case) 3](#_Toc221528084)

[2.1 Feature Process Flow / Use Case Model 3](#_Toc221528085)

[2.2 Use Case(s) 3](#_Toc221528086)

Revision History

| **Version** | **Date** | **Revision Description** |
| --- | --- | --- |
| .01 |  |  |
| .02 |  |  |
| .03 |  |  |
| .04 |  |  |
| 1.0 |  | Approved Use Case |
|  |  |  |
|  |  |  |

# Approvals

We have carefully assessed the Use Cases for this project. This document has been completed in accordance with the requirements of the System Development Methodology.

MANAGEMENT CERTIFICATION - Please check the appropriate statement.

\_\_\_\_\_\_ the document is accepted.

\_\_\_\_\_\_ the document is accepted pending the changes noted.

\_\_\_\_\_\_ the document is not accepted.

We fully accept the changes as needed improvements and authorize initiation of work to proceed. Based on our authority and judgment, the continued operation of this system is authorized.

(\*=Required \*\*= Submit for Review Approval Not Required)

Executive Sponsor\*\* DATE

Project Sponsor\* DATE

Quality Assurance Manager / Team Lead\* DATE

Business Analyst Manager / Team Lead\* DATE

Project Manager DATE

# Use Case List

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | **Primary Actor** | **Use Cases** |
|  |  |  |
|  |  |  |
|  |  |  |

# Feature Name (Example: ATM Transaction)

## Feature Process Flow / Use Case Model

## Use Case(s)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | Enter a unique numeric identifier for the Use Case. e.g. UC-1.2.1 | | | |
| **Use Case Name:** | Enter a short name for the Use Case using an active verb phrase. e.g. Withdraw Cash | | | |
| **Created By:** |  | | **Last Updated By:** |  |
| **Date Created:** |  | | **Last Revision Date:** |  |
| **Actors:** | | [An actor is a person or other entity external to the software system being specified who interacts with the system and performs use cases to accomplish tasks. Different actors often correspond to different user classes, or roles, identified from the customer community that will use the product. Name the actor that will be initiating this use case (primary) and any other actors who will participate in completing the use case (secondary).] | | |
| **Description:** | | [Provide a brief description of the reason for and outcome of this use case.] | | |
| **Trigger:** | | [Identify the event that initiates the use case. This could be an external business event or system event that causes the use case to begin, or it could be the first step in the normal flow.] | | |
| **Preconditions:** | | [List any activities that must take place, or any conditions that must be true, before the use case can be started. Number each pre-condition. e.g.   1. Customer has active deposit account with ATM privileges 2. Customer has an activated ATM card.] | | |
| **Postconditions:** | | [Describe the state of the system at the conclusion of the use case execution. Should include both *minimal guarantees* (what must happen even if the actor’s goal is not achieved) and the *success guarantees* (what happens when the actor’s goal is achieved. Number each post-condition. e.g.   1. Customer receives cash 2. Customer account balance is reduced by the amount of the withdrawal and transaction fees] | | |
| **Normal Flow:** | | [Provide a detailed description of the user actions and system responses that will take place during execution of the use case under **normal, expected** conditions. This dialog sequence will ultimately lead to accomplishing the goal stated in the use case name and description.   1. Customer inserts ATM card 2. Customer enters PIN 3. System prompts customer to enter language performance English or Spanish 4. System validates if customer is in the bank network 5. System prompts user to select transaction type 6. Customer selects Withdrawal From Checking 7. System prompts user to enter withdrawal amount 8. … 9. System ejects ATM card] | | |
| **Alternative Flows:**  **[Alternative Flow 1 – Not in Network]** | | [Document **legitimate** branches from the main flow to handle special conditions (also known as extensions). For each alternative flow reference the branching step number of the normal flow and the condition which must be true in order for this extension to be executed. e.g. Alternative flows in the *Withdraw Cash* transaction:  4a. In step 4 of the normal flow, if the customer is not in the bank network   1. System will prompt customer to accept network fee 2. Customer accepts 3. Use Case resumes on step 5   4b. In step 4 of the normal flow, if the customer is not in the bank network   1. System will prompt customer to accept network fee 2. Customer declines 3. Transaction is terminated 4. Use Case resumes on step 9 of normal flow   Note: Insert a new row for each distinctive alternative flow. ] | | |
| **Exceptions:** | | [Describe any anticipated **error conditions** that could occur during execution of the use case, and define how the system is to respond to those conditions.  e.g. Exceptions to the Withdraw Case transaction  2a. In step 2 of the normal flow, if the customer enters and invalid PIN   1. Transaction is disapproved 2. Message to customer to re-enter PIN 3. Customer enters correct PIN 4. Use Case resumes on step 3 of normal flow] | | |
| **Includes:** | | [List any other use cases that are included (“called”) by this use case. Common functionality that appears in multiple use cases can be split out into a separate use case that is included by the ones that need that common functionality. e.g. steps 1-4 in the normal flow would be required for all types of ATM transactions- a Use Case could be written for these steps and “included” in all ATM Use Cases.] | | |
| **Frequency of Use:** | | [How often will this Use Case be executed. This information is primarily useful for designers. e.g. enter values such as 50 per hour, 200 per day, once a week, once a year, on demand etc.] | | |
| **Special Requirements:** | | [Identify any additional requirements, such as nonfunctional requirements, for the use case that may need to be addressed during design or implementation. These may include performance requirements or other quality attributes.] | | |
| **Assumptions:** | | [List any assumptions that were made in the analysis that led to accepting this use case into the product description and writing the use case description.  e.g. For the *Withdraw Cash* Use Case, an assumption could be:  The Bank Customer understands either English or Spanish language.] | | |
| **Notes and Issues:** | | [List any additional comments about this use case or any remaining open issues or TBDs (To Be Determined) that must be resolved. e.g.   1. What is the maximum size of the PIN that a use can have?] | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | [Repeat for multiple use cases] | | | |
| **Use Case Name:** | Generate Coupon | | | |
| **Created By:** |  | | **Last Updated By:** |  |
| **Date Created:** |  | | **Last Revision Date:** |  |
| **Actors:** | | Admin() | | |
| **Description:** | | A coupon is generated when Admin clicks on generate coupons and  Is directed to coupon generation jsp, where admin will fill details of coupon. | | |
| **Trigger:** | | When Admin wants to generate the coupon. | | |
| **Preconditions:** | | **Admin should give coupon name , discount and duration.** | | |
| **Postconditions:** | | coupon details is saved in file corresponding to customer name.and also be saved to database.After expiry of time coupon will be deleted. | | |
| **Normal Flow:** | | Admin calls generate coupons method .  Then coupon will be generated | | |
| **Alternative Flows:** | | Call methods by direct object. | | |
| **Exceptions:** | | If admin enters same coupon number again coupon will not be generated. | | |
| **Includes:** | | Would be used by apply coupon module. | | |
| **Frequency of Use:** | | On demand. | | |
| **Special Requirements:** | | All customer list should be present. | | |
| **Assumptions:** | | Database of list of customer is present. | | |
| **Notes and Issues:** | | Size of string 5 alphabet. | | |

# Feature Name (Repeat for multiple features)

## Feature Process Flow / Use Case Model

## Use Case(s)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | Return Goods | | | |
| **Use Case Name:** |  | | | |
| **Created By:** |  | | **Last Updated By:** |  |
| **Date Created:** |  | | **Last Revision Date:** |  |
| **Actors:** | | Customer, Admin(),Shipping/Receiving department, Admin of third party merchant. | | |
| **Description:** | | Customer wants to return an item(s) purchased through e-Store system by requesting from Return Goods Authorization(RGA) | | |
| **Trigger:** | | When customer wants to return the item(s) | | |
| **Preconditions:** | | **The customer has purchased an item and wishes to return it.** | | |
| **Postconditions:** | | The customer has received either approval to return the item(s) and it will be updated in the inventory and in case of third party merchant, it will be updated in the inventory of third party admin. | | |
| **Normal Flow:** | | 1. The customer logs in to the e-Store system 2. Use search for order list. 3. The system will display the selected order in the Return Goods Screen. 4. The customer will select the products to return. 5. The customer will select submit. 6. The e-Store system notifies the customer of the request approval and sends further instructions via e-mail. 7. The customer ships the item(s) back to CapStore according to the instructions. 8. The receiving department at CapStore acknowledges the receipt in the e-Store system. 9. The e-Store system refunds the customer’s payment. 10. The e-Store notifies the customer of the receipt and refund via e-mail. | | |
| **Alternative Flows:** | | 1. The customer logs in to the e-Store system. 2. Use search for order. 3. The system will display the selected order in the Return Goods Screen. 4. The customer will select products to return. 5. The customer will select Submit. 6. The Receiving department of e-Store system deny the customer requests. 7. The e-Store system notifies the customer of the requests denial via e-mail. | | |
| **Exceptions:** | | Customer should return goods within 14 days of purchase otherwise the payment will not be refunded back. | | |
| **Includes:** | | Would be used by refund money module. | | |
| **Frequency of Use:** | | N/A | | |
| **Special Requirements:** | | None | | |
| **Assumptions:** | | Database of lists of goods present. | | |
| **Notes and Issues:** | | 1. Order id not found in system 2. Customer not found in the system. 3. Account no longer valid.   I | | |