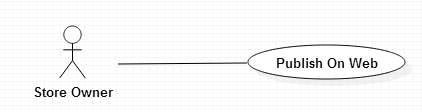
1. <Store Owner> Publish On Web

**Use Case diagram**

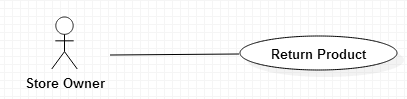


**Use Case Specification**

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – DQT01** | | | |
| **Use Case No.** | **DQT01** | Use Case Version | 2.0 |
| **Use Case Name** | Publish On Web | | |
| **Author** | DanQT | | |
| **Date** | May 26, 2015 | Priority | High |
| **Actor**:   * Store Owner   **Summary**:   * This use case allows store owner chooses one product to raise on website.   **Goal**:   * Successfully publish one product on web.   **Triggers**:   * Store owner click on link “In Inventory”.   **Preconditions**:   * Product status is “In Inventory”.   **Post Conditions**:   * **Success**: Product is raised on web and can be found by search function. Product status changes from “In Inventory” to “On Web”. * **Fail**: Product status still is “In Inventory”.   **Main Success Scenario**:   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Store owner input product’s information to fields. | System checks input values. [Exception 1] | | 2 | Store owner clicks on “Publish” button. [Alternative 1] | System change product status to “On Web” and notify store owner. [Exception 2] |   **Alternative Scenario**:   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Store owner clicks on “Cancel” button. | System forwards to “Management Page”. No information is recorded. |   **Exceptions**:   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | If inputted price is not valid. | System show notification and request store owner to enter a new one. | | 2 | Store owner confirms error and clicks on “OK” button. | Change product status unsuccessfully, system show error message on screen.  System forwards to “Management Page”. |   **Relationships**: N/A  **Business Rules**:   * Only product in “In Inventory” status can be published on web. * Value inputted in “Price” field must be positive number or “Call For Price”. | | | |
|  | | | |

1. <Store Owner> Return Product

**Use Case diagram**

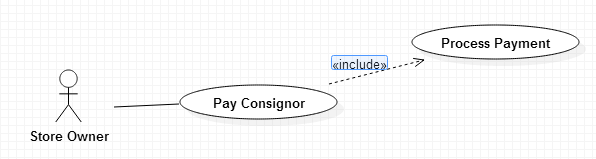


**Use Case Specification**

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – DQT02** | | | |
| **Use Case No.** | **DQT02** | Use Case Version | 2.0 |
| **Use Case Name** | Return Product | | |
| **Author** | DanQT | | |
| **Date** | May 26, 2015 | Priority | High |
| **Actor**:   * Store Owner   **Summary**:   * This use case allows store owner to cancel product consignment.   **Goal**:   * Successfully remove product from website.   **Triggers**:   * When store owner click on “Cancel” button.   **Preconditions**:   * Product status must not be “Sold” nor “Completed”.   **Post Conditions**:   * **Success**: Product status is changed to “Cancel” and product can not be found by Search function. * **Fail**: Product status keeps its current status (Can be “On Web” or “In Inventory”).   **Main Success Scenario**:   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Store owner clicks on “Cancel” button. | System shows pop-up to confirm cancel request. | | 2 | Store owner clicks on “Yes” button. [Alternative 1] | System cancel consignment and show notification on screen.[Exception 1]  System forwards to “Management Page” |   **Alternative Scenario**:   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Store owner clicks on “No” button. | System forwards to “Management Page”. No information is recorded. |   **Exceptions**:   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Store owner confirms error and clicks on “OK” button. | Change product status unsuccessfully, system show error message on screen.  System forwards to “Management Page”. |   **Relationships**: N/A  **Business Rules**:   * Product status must not be “Sold” nor “Completed”. * Consignor must make a cancel request so store owner can cancel this consignment. | | | |
|  | | | |

1. <Store Owner> Pay Consignor

**Use Case diagram**

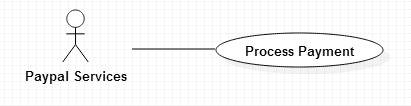


**Use Case Specification**

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – DQT03** | | | |
| **Use Case No.** | **DQT03** | Use Case Version | 2.0 |
| **Use Case Name** | Pay Consignor | | |
| **Author** | DanQT | | |
| **Date** | May 26, 2015 | Priority | High |
| **Actor**:   * Store Owner   **Summary**:   * This use case allows store owner to pay consignor when product is sold.   **Goal**:   * Pay consignor when their product was sold.   **Triggers**:   * When store owner click on “Sold” link.   **Preconditions**:   * Product status must be “Sold”.   **Post Conditions**:   * **Success**: Product status is changed to “Completed”. * **Fail**: Product status still remains is “Sold”. No payment is completed.   **Main Success Scenario**:   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Store owner enter needed information. [Exception 1].  Store owner clicks on “Next” button. [Alternative 1]. | System shows payment information. | | 2 | Store owner clicks on “Transfer” button. [Alternative 2] | System calls payment services from Paypal Services.  [Exception 2].  System shows confirm form. | | 3 | Store owner enter code given by Paypal to form and click “Submit”.  [Alternative 3].  [Exception 3] | System processes payment. [Exception 4]  System shows notification when done and forward to “Management Page”. |   **Alternative Scenario**:   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Store owner clicks on “Cancel” button. | System forwards to “Management Page”. No information is recorded. | | 2 | Store owner clicks on “Cancel” button. | System forwards to “Management Page”. No information is recorded. | | 3 | Store owner clicks on “Cancel” button. | System forwards to “Management Page”. No information is recorded. |   **Exceptions**:   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Details information is empty. | System shows notification and ask store owner to enter information again. | | 2 | Store owner clicks on “OK” button to confirm. | Process payment unsuccessfully, system shows error message on screen.  System forwards to “Management Page”. | | 3 | Given code it not matched. | System shows notification and ask store owner to enter code again. | | 4 |  | Process payment unsuccessfully, system shows error message on screen.  System forwards to “Management Page”. |   **Relationships**: include use case “Process Payment”.  **Business Rules**:   * Given code is automatically generated by Paypal service. * Payment information must be in correct format and system will get those information from database. If store owner want to change his payment information, he must edit his profile before process payment. * Store owner can not change receiver’s information. | | | |
|  | | | |

1. <Paypal Services> Process Payment

**Use Case diagram**

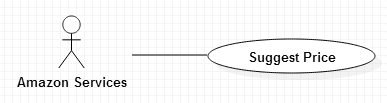


**Use Case Specification**

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – DQT04** | | | |
| **Use Case No.** | **DQT04** | Use Case Version | 2.0 |
| **Use Case Name** | Process Payment | | |
| **Author** | DanQT | | |
| **Date** | May 26, 2015 | Priority | High |
| **Actor**:   * Paypal Services   **Summary**:   * This use case helps store owner process “Pay Consignor” use case.   **Goal**:   * Transfer money from store owner account to consignor account.   **Triggers**:   * When “Pay Consignor” use case is processing.   **Preconditions**:   * Transferor and receiver information must be provided.   **Post Conditions**:   * **Success**: Deposit receiver bank account and withdraw transferor bank account. * **Fail**: No transaction is completed.   **Main Success Scenario**:   * System request Paypal services to complete payment process with specified information. [Exception 1] * Notify to system so system can continue Pay Consignor use case.   **Alternative Scenario**: N/A  **Exceptions**:   * Given information is not right. System notify to terminate process. * Transferor bank   **Relationships**: N/A  **Business Rules**:   * Transferor balance must be greater than the amount. * Transferor and receiver information will be taken from database. | | | |
|  | | | |

1. <Amazon Services> Suggest Price

**Use Case diagram**



**Use Case Specification**

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – DQT05** | | | |
| **Use Case No.** | **DQT05** | Use Case Version | 2.0 |
| **Use Case Name** | Suggest Price | | |
| **Author** | DanQT | | |
| **Date** | May 26, 2015 | Priority | High |
| **Actor**:   * Store Owner   **Summary**:   * This use case allows system to suggest price to store owner and show suggested price to consignor.   **Goal**:   * Estimate product price using information provided by consignor.   **Triggers**:   * When consignor clicks on “Next”.   **Preconditions**:   * Product information must be provided.   **Post Conditions**:   * **Success**: Show original price of the product and give suggested price given by specify formula. * **Fail**: Can not found product original price.   **Main Success Scenario**:   * System calls Amazon services to find product with specified information. * System return price of product if found or “null” if it can not be found. * Notify to system to continue “Consign Product” use case.   **Alternative Scenario**: N/A  **Exceptions**:   * Error occurs when call Amazon services, notify system to terminate “Consign Product”. Show error message on screen.   **Relationships**: N/A  **Business Rules**:   * The reliability of this method depends on user’s given information. * System will get the lowest price from the result and use specified formula to calculate suggest price. This formula depends on each store owner decision. * If the result from Amazon API is empty, system will notify user with specified message:” We could not find your product. Store owner will check and price your product later” | | | |
|  | | | |