**Test Plan for Inventory Management System.**

**1 Introduction**

This test plan is designed to validate the integration between the company's internal inventory management system and the third-party logistics providers (3PLs) to ensure accurate inventory adjustments across all systems.

## Objectives

* Verify that inventory adjustments made in one system are updated correctly across all internal and third-party warehouses.
* Ensure that inventory changes made are auditable, and can be identified.
* Ensure the inventory can handle both positive and negative inventory across all the systems.
* Verify the Validate error received from 3PL systems are having proper error handling.

1. **Risks and Solutions**

Integrating inventory management with third-party logistics providers has various risks. These risks need to be carefully assessed and resolved to ensure smooth and reliable operations. Here are the key risks of integrating third-party systems:

* 1. Inconsistent Data Sync Between Systems:

Risk: Due to unknown Factors like network issues or system downtime, inventory changes may not reach the 3PL and the synced between internal and third-party systems, leading to discrepancies in inventory levels.

Solution: Implement periodic reconciliation processes to compare inventory levels between systems and 3PL. Ensure API responses are recorded and maintained so if the inventory isn’t updated, proper error messages are shown.

2.2 Performance issue during large inventory adjustments

Risk: During large inventory adjustments or periods of high traffic, the system may experience performance issues. This could result in slower response times, system crashes, or delays in processing inventory changes.

Solution: It’s essential to conduct load testing before and during periods of high traffic. Load testing simulates high volumes of simultaneous inventory adjustments and user traffic to identify potential issues in the system.

1. **Test Approach**

The project uses an agile approach. The requirements identified for that iteration will be delivered to the team and tested.

## Test Automation

Automated scripts are used to perform repetitive testing tasks, ensuring the inventory system functions as expected without manual intervention.

* 1. Functional Testing

Testing to ensure that the inventory management system’s features work as intended, meeting specified business requirements.

* 1. Regression Testing,

Testing the inventory system after changes or updates to ensure that new code doesn't negatively impact existing functionality, such as accurate stock tracking or warehouse communication.

* 1. Integration Testing

Testing how well the inventory system integrates with other systems, such as third-party logistics (3PL) providers, to ensure smooth data exchange and inventory updates across all platforms.

* 1. Load Testing

Testing the inventory system under high traffic or large data volumes to ensure it can handle peak load, or bulk inventory adjustments, without performance degradation or crashes.

1. **Test Phases**

## Functional Test Cases:

| **Test Case ID** | **Test Scenario** | **Scenario Type** | **Test Objective** | **Action** | **Expected Result** |
| --- | --- | --- | --- | --- | --- |
| TC001 | Verify that the new inventory received is uploaded correctly across all systems. | Positive | To ensure that new inventory is uploaded correctly in both internal and third-party logistics systems. | Enter the new inventory received into the system with their valid item id, dates and batch details | The new inventory should reflect correctly in all systems, including third-party logistics, with accurate data. |
| TC002 | Verify that the inventory which has crossed its expiry date is removed across all systems. | Positive | To ensure that expired inventory is removed across all systems, preventing its sale or storage. | Identify items that have passed their expiry date.  Find the same item with the same batch number and date across all the warehouse  Remove the expired items from all the systems | The that expired items should be removed from the all the warehouses. |
| TC003 | Verify that the inventory that was recalled is removed across all systems. | Positive | To verify that recalled inventory is properly removed from the system to prevent sale or use. | Mark inventory items as recalled.  Find the same item with the same batch number and date across all the warehouse  Remove recalled items from the internal system. | Ensure that recalled items are also removed all the warehouses. |
| TC004 | Verify that a new item received from the supplier is added in all warehouses and recorded with a unique ID for easy identification. | Positive | To ensure that a new item is added across all warehouses with a unique identifier for easy tracking. | Add new item to system, assign it to multiple warehouses. | New item is added to inventory in all warehouses with a unique ID for easy tracking. |
| TC005 | Verify that the inventory received by a selective warehouse is not updated across other warehouses. | Negative | To ensure that inventory adjustments made in one warehouse are not incorrectly reflected in other warehouses. | Adjust inventory in one warehouse only. | The inventory is updated only in the selected warehouse and not in others. |
| TC006 | Verify that validation errors are displayed when attempting to add inventory to an invalid item. | Negative | To ensure that invalid items are not added to the inventory, and appropriate error messages are displayed. | Try to add inventory to a non-existent or invalid item. | Validation error is displayed indicating the item is invalid or doesn't exist in the system. |
| TC007 | Verify that an error is displayed when attempting to update a warehouse that is no longer in use. | Negative | To ensure that attempts to update a warehouse no longer in use are prevented with an error message. | Attempt to make an update to a warehouse marked as no longer in use. | An error message is displayed indicating that the warehouse is no longer in use and cannot be updated. |
| TC008 | Verify proper validation errors are shown when trying to adjust more than the available stock. | Negative | To ensure validations are shown when tring to remove more items than are available in stock. | Adjust inventory to remove more items than what's available | An error message is displayed indicating that the item requested is more than the available items. |
| TC009 | Verify a duplication error is shown when submitting the same inventory details twice | Negative | To ensure the system detects and prevents the submission of duplicate inventory details. | Remove an item from the inventory and submit the same request again | An error message is displayed indicating that the item already exists in the system. |

## Integration Test Cases

| **Test Case ID** | **Test Scenario** | **Scenario Type** | **Test Objective** | **Action** | **Expected Result** |
| --- | --- | --- | --- | --- | --- |
| TC001 | Verify the API inegration between Internal and 3PL Systems are working as expected | Positive | To verify that the API integration between internal and 3PL systems works correctly for data synchronization. | Initiate an inventory change in the internal system and check for correct data reflection in the 3PL system. | The inventory data is successfully synchronized between the internal system and the 3PL system without errors. |
| TC002 | Verify proper validation errors are shown when there are any API failues from the 3PL System | Negative | To ensure that proper validation errors are displayed when the API fails to communicate with the 3PL system. | Simulate an API failure from the 3PL system and attempt to update inventory. | An error message is displayed, indicating the failure to communicate with the 3PL system. |
| TC003 | Verify the data changes made in the system is reflecting correctly in the 3PL system | Positive | To confirm that changes made in the internal system are reflected correctly in the 3PL system. | Modify inventory data in the internal system and verify that the changes are reflected in the 3PL system. | The changes made in the internal system are reflected accurately in the 3PL system. |
| TC004 | Verify the 3PL system is able to handle multiple adjustments at the same time | Positive | To verify that the 3PL system can process multiple inventory adjustments simultaneously without errors. | Perform multiple inventory adjustments at the same time in the internal system and ensure the 3PL system processes them correctly. | The 3PL system successfully handles and processes multiple inventory adjustments simultaneously. |
| TC005 | Verify a timeout error is shown when there is connection issues with the 3PL System | Negative | To check if a timeout error is displayed when there are connection issues with the 3PL system. | Simulate a connection issue between the internal system and the 3PL system and attempt an inventory update. | A timeout error message is displayed, indicating a connection issue with the 3PL system. |
| TC006 | Verify a concurrency error is shown when trying to update the same inventory from multiple devices | Negative | To ensure that the system correctly handles concurrency issues when the same inventory is updated from multiple devices simultaneously. | Attempt to update the same inventory record from two different devices at the same time. | A concurrency error is displayed, indicating that the same inventory cannot be updated simultaneously from multiple devices. |
| TC007 | Verify proper validation is shown when invalid data is entered | Negative | To ensure that proper validation is triggered when invalid data is entered into the system. | Enter invalid inventory data (e.g., incorrect quantity or item ID) and submit the changes. | A validation error message is displayed, indicating the invalid data entered. |
| TC008 | Verify only authorized users can access and modify the inventory via API. | Positive | To ensure users without the access aren't able to update the system | Send API requests without an authentication token. | Requests without authentication or with invalid tokens should be rejected with a 401 Unauthorized status. |
| TC009 | verify that the API provides meaningful error messages for different types of errors. | Negative | Ensure that API calls return appropriate error codes and messages for different error conditions (e.g., invalid input, system failures). | Trigger an API request with missing or invalid data. | The API should return appropriate error codes (e.g., 400 for bad request, 404 for not found) and error messages that help users identify the issue. |

* 1. Performance Test Caes

| **Test Case ID** | **Test Scenario** | **Test Objective** | **Action** | **Expected Result** |
| --- | --- | --- | --- | --- |
| TC001 | Verify the API response time with load testing | To ensure that the API can handle a high volume of requests within an acceptable time frame. | Send 1000 API requests per minut with mix of GET and POST operations | The API should process all requests within an acceptable response time without errors or delays. |
| TC002 | Verify the performance of the system during maximum expected workload | To evaluate how the system performs under maximum expected workload conditions. | Send 1000 inventory adjustments within 1 hour | The system should process all inventory adjustments successfully, with no crashes or performance degradation. |
| TC003 | Verify system stability long operation periods | To ensure the data is synchronized properly and the system's performance is not affected during prolonged usage | Run the system for an extended period. with continuous API calls and user activity. | The system should maintain stable performance, with accurate data synchronization |
| TC004 | Verify the systemm is able to perform with maximum users working on the system | To evaluate how the system performs under maximum user requests | Send request to adjust inventory from 100 users are the same time. | The system should process requests from all 100 users simultaneously without crashes, slowdowns, or errors. |

* 1. Regression Test Cases

| **Test Case ID** | **Test Scenario** | **Test Objective** | **Action** | **Expected Result** |
| --- | --- | --- | --- | --- |
| TC001 | Verify inventory adjustments are saved | Ensure that inventory adjustments made by users are saved correctly across all systems | Adjust inventory by adding 50 items to an item ID | The inventory of that particular item should increase by 50. |
| TC002 | Verify negative inventory adjustments | Ensure that inventory adjustments made by users are saved correctly across all systems | Adjust inventory by reducing 50 items | The inventory of that particular item should decrease by 50 items. |
| TC003 | Verify inventory update for multiple warehouses | Ensure inventory adjustments are reflected correctly based on the warehouses | Adjust inventory in one warehouse and verify other warehouses | The inventory of the specified warehouse should change, while the other warehouses remain the same. |
| TC004 | Verify addition of new item in warehouse | Ensure new item is added correctly across all warehouses | Add a new item to inventory and check all warehouses | The item should appear in all warehouse systems. |
| TC005 | Verify system handles invalid item IDs | Ensure the system returns an error for invalid item ID | Try to adjust inventory with an invalid item ID | The system should display an error message, and the existing inventory shouldn't change. |
| TC006 | Verify error message is shown when invalid quantity is entered | Ensure the system returns an error when an invalid quantity is entered | Try to adjust inventory with a quantity of 0, a decimal, or a string | The system should display an error message, and the existing inventory shouldn't change. |
| TC007 | Verify inventory adjustment is disabled for unused warehouse | Ensure inventory adjustment doesn't occur if the warehouse is no longer in use | Try to adjust inventory with a warehouse that is no longer in use | The system should display an error and not adjust inventory. |
| TC008 | Verify that maximum item limits are enforced | Ensure that the system limits the quantity that can be adjusted at once | Try to adjust inventory by an amount greater than the system limit | The system should display an error for exceeding the limit. |
| TC009 | Verify user permissions for inventory adjustments | Ensure users can only adjust inventory within their allowed permissions | Attempt inventory adjustment with a user without the required permissions | The system should display an error message. |
| TC010 | Verify inventory adjustments are reversible | Ensure that inventory adjustments can be undone if needed | Undo an inventory adjustment and verify the previous level | The previous inventory level should be restored after undoing. |
| TC011 | Verify proper error handling for empty inventory adjustments | Ensure the system handles empty or null values in inventory adjustments | Try submitting an empty inventory adjustment | The system should return an error indicating missing data. |
| TC012 | Verify user is able to adjust the warehouse details | Ensure the system is able to update the warehouse details entered | Update the name or address of the warehouse | The system should display the updated warehouse details. |
| TC013 | Verify that comprehensive audit logging is implemented for inventory adjustments | Ensure the actions are logged accurately with correct user details and timestamps | Perform various inventory adjustments and check the audit logs | All actions should be logged, including the user's details, action, and timestamp. |
| TC014 | Verify the search functionality is working as expected | Ensure the search functionality is working as expected | Search for an item in the warehouse | All the items with the related details should be shown to the user. |
| TC015 | Verify notification is triggered when adding inventory | Ensure a notification is triggered when inventory is added to either an internal warehouse or a 3PL warehouse | Add inventory to an internal warehouse and 3PL warehouse | A notification is sent for both internal warehouse and 3PL warehouse inventory additions. |
| TC016 | Verify notification is triggered when inventory is added by the internal warehouse or a 3PL warehouse | Ensure a notification is triggered when inventory is added by an internal warehouse or a 3PL warehouse | Add inventory from the internal warehouse and 3PL warehouse | A notification should be received by the system to adjust the database. |