|  |  |
| --- | --- |
| **Use Case Name** | Create Detailed Sales Order with Wire Specifications and Make-to-Order/Make-to-Stock Options |
| **Use Case ID** | UC-002 |
| **Actor** | Sales Representative |
| **Pre-Condition** | The Sales Representative is logged into ERPNext. The Sales Order form has been customized to include detailed wire specifications and options for make-to-order and make-to-stock, with a table format for multiple items. |
| **Post-Condition** | A detailed Sales Order is created with wire specifications and make-to-order/make-to-stock options, and it is saved, its status is submitted and transferred to Pending Order List. |
| **Basic Flow** | 1. Sales Representative logs into ERPNext.  * **System Response:** ERPNext displays the dashboard.  1. Sales Representative navigates to Selling > Sales Order > New.  * **System Response:** ERPNext displays the Sales Order form.  1. Sales Representative selects the order type: make-to-order or make-to-stock.  * **System Response:** If make-to-order is selected, the customer section is enabled. If make-to-stock is selected, the customer section is disabled.  1. Sales Representative (if make-to-order) selects the customer from the customer list.  * **System Response:** ERPNext associates customer to that Sales Order.  1. Sales Representative enters the details in the fields (e.g., wire specifications: gauge, number of wires, PVC color, length, packaging, quantity) in a table format, with each row representing a different order/item.  * **System Response:** ERPNext displays fields for entering wire specifications in a tabular format.  1. Sales Representative adds more rows to the table for additional orders/items as needed.  * **System Response:** ERPNext allows additional rows to be added to the table.  1. Sales Representative enters the quantity and price for each item in the table.  * **System Response:** ERPNext calculates the total amount for each row and the overall total.  1. Sales Representative reviews the Sales Order details and clicks "Save" to save the Sales Order as a draft.  * **System Response:** ERPNext saves the Sales Order as a draft.  1. Sales Representative clicks "Submit" to confirm the Sales Order.  * **System Response:** ERPNext submits the Sales Order and updates the status to "Submitted." |
| **Alternate Flow** | * **4a)** Customer not found in the customer list (Step 4):   - **System Response:** ERPNext displays a message: "Customer not found."  - **Sales Representative** clicks "New Customer"  - **System Response:** ERPNext navigates to the new Customer Creation form.   * **5a)** Missing or invalid wire specifications (Step 5):   - **System Response:** ERPNext displays an error message: "Wire specifications are required."  - **Sales Representative** enters the missing or correct wire specifications.  - **System Response:** ERPNext validates the specifications and updates the form. |

|  |  |
| --- | --- |
| **Use Case Name** | Material Availability Check for Production Order |
| **Use Case ID** | UC003 |
| **Actor** | User (Production Planner) |
| **Pre-Condition** | Pending orders exist in the system, materials are recorded in the inventory. |
| **Post-Condition** | User is notified of material shortages if any, and production order cannot be marked complete if there is a shortage. |
| **Basic Flow** | 1. User navigates to the Pending Orders list.  * **System Response:** ERPNext displays the Pending Orders list.  1. User selects pending orders to transfer to the Production Order list.  * **System Response:** ERPNext marks the selected orders for transfer to the Production Order list.  1. System automatically calculates the required materials for the selected orders by checking current inventory levels for required materials and calculating the material requirements for the production orders. 2. If sufficient materials are available, the system allows the transfer to the Production Order list.  * **System Response:** Confirms sufficient materials and marks the orders as ready for production.  1. User marks the production order as complete.  * **System Response:** Updates the material consumption in the inventory.  1. System updates the material consumption in the inventory. |
| **Alternate Flow** | * **4a)** If sufficient materials are not available (after Step 4):   - **System Response:** Notifies the user of the material shortage and prevents the production order from being marked as complete until materials are available. |

|  |  |
| --- | --- |
| **Use Case Name** | Automated BOM Creation and Material Calculation |
| **Use Case ID** | UC004 |
| **Actor** | User (Production Planner) |
| **Pre-Condition** | Pending orders exist in the system. |
| **Post-Condition** | BOM is automatically created, and material requirements are calculated based on pending orders. |
| **Basic Flow** | 1. User navigates to the Pending Orders list.  * **System Response:** ERPNext displays the Pending Orders list.  1. User selects pending orders to Create BOM.  * **System Response:** ERPNext marks the selected orders for BOM creation  1. User clicks Create BOM option.  * **System Response:** System calculates the required materials for the production orders. Creates and displays the BOMs for user review.  1. User clicks the print button to print the BOM for the selected pending orders.  * **System Response:** Prompts confirmation message to print.  1. User confirms.  * **System Response:** Print out BOM report |
| **Alternate Flow** | * **3a)** If BOM template does not exist (after Step 3):   - **System Response:** Prompts the user to create a new BOM template. User creates and saves the new BOM template. Saves the new BOM template. Proceeds with the material calculation and BOM creation. |

|  |  |
| --- | --- |
| **Use Case Name** | Automated Cost Calculation and Recording |
| **Use Case ID** | UC005 |
| **Actor** | User (Finance Manager) |
| **Pre-Condition** | Completed orders exist in the system, cost formula is defined. |
| **Post-Condition** | Costs are automatically calculated and recorded against completed orders. |
| **Basic Flow** | 1. User navigates to the Completed Orders list.  * **System Response:** ERPNext displays the Completed Orders list.  1. User selects a completed order.  * **System Response:** ERPNext opens the selected order details.  1. System retrieves the cost formula (Total cost = Cost + x% of Cost) and calculates the total cost for the order. 2. System records the calculated cost against the completed order. 3. User generates sales invoice against completed order and cost is included in Sales Invoice. |
| **Alternate Flow** | None |

|  |  |
| --- | --- |
| **Use Case Name** | Vendor Creation |
| **Use Case ID** | UC006 |
| **Actor** | Procurement Officer |
| **Pre-Condition** | The Procurement Officer is logged into the ERPNext system. |
| **Post-Condition** | Vendor details are saved and submitted in the system. |
| **Basic Flow** | 1. Procurement Officer navigates to Buying > Supplier > New.  * **System Response:** Displays the Supplier creation form.  1. Procurement Officer enters supplier name.  * **System Response:** Validates the name.  1. Procurement Officer enters contact details (phone, email).  * **System Response:** Validates the contact details.  1. Procurement Officer enters the address (street, city, state, zip code).  * **System Response:** Validates the address.  1. Procurement Officer enters Payables and Receivables.  * **System Response:** Validates the address.  1. Procurement Officer Clicks "Save."  * **System Response:** Saves the supplier details but does not submit them.  1. Procurement Officer clicks "Submit."  * **System Response:** Submits the supplier details, finalizing the creation process. |
| **Alternate Flow** | * **2a, 3a, 4a, 5a)** If the entered information is invalid at any step (Step 2, 3, 4, 5), the system displays an error message with details on what needs to be corrected. * Procurement Officer corrects the invalid information and proceeds with the next steps. |

|  |  |
| --- | --- |
| **Use Case Name** | Purchase Order Creation with Material Destination |
| **Use Case ID** | UC007 |
| **Actor** | Procurement Officer |
| **Pre-Condition** | The Procurement Officer is logged into the ERPNext system. |
| **Post-Condition** | Purchase order is created and submitted in the system. |
| **Basic Flow** | 1. Procurement Officer navigates to Buying > Purchase Order > New.  * **System Response:** Displays the Purchase Order creation form.  1. Procurement Officer selects the supplier from a dropdown list.  * **System Response:** Populates supplier details in the form.  1. Procurement Officer enters the order date.  * **System Response:** Validates the date format.  1. Procurement Officer selects the destination (Subcontractor or Owner’s Store).  * **System Response:** Updates form fields accordingly.  1. Procurement Officer clicks "Add Row" to add items.  * **System Response:** Displays fields for product, quantity, and price.  1. Procurement Officer selects the product from a list.  * **System Response:** Displays product details. * If Product is "Copper Rods" System Disables “Color” field.  1. Procurement Officer enters the quantity and price.  * **System Response:** Calculates the total price for the item.  1. Procurement Officer repeats steps 5-7 for all items in the order.  * **System Response:** Updates the total order amount.  1. Procurement Officer Clicks "Save."  * **System Response:** Saves the purchase order but does not submit it.  1. Procurement Officer clicks "Submit."  * **System Response:** Submits the purchase order, finalizing the creation process. |
| **Alternate Flow** | * **3a, 7a)** If any information is invalid (Step 3 and 7), the system displays an error message with details on what needs to be corrected. * **4a)** Subcontractor not found in the destination list (Step 4), Procurement Officer clicks "New Subcontractor" * **System Response:** ERPNext navigates to the new Subcontractor Creation form. * Procurement Officer corrects the invalid information and proceeds with the next steps. |

|  |  |
| --- | --- |
| **Use Case Name** | Marking Incomplete Purchase Orders as Complete |
| **Use Case ID** | UC008 |
| **Actor** | User (Inventory Manager) |
| **Pre-Condition** | Incomplete purchase orders exist in the system. |
| **Post-Condition** | Stock levels are updated based on the completion of purchase orders. |
| **Basic Flow** | 1. User navigates to the Incomplete Purchase Orders list.  * **System Response:** ERPNext displays the Incomplete Purchase Orders list.  1. User selects an incomplete purchase order.  * **System Response:** ERPNext opens the selected purchase order details.  1. User marks the purchase order as complete.  * **System Response:** ERPNext updates the status of the purchase order to complete.  1. System updates the stock levels at the specified location (owner's store or subcontractor).  * **System Response:** Adjusts the inventory levels based on the completed purchase order. Confirms the update of stock levels. |
| **Alternate Flow** | * **4a)** If stock levels are not correctly updated (after Step 4): * **System Response:** Notifies the user of discrepancies in stock levels. User manually adjusts the stock levels. Updates the inventory accordingly. Confirms the correction of stock levels. |

### Subcontractor Management

|  |  |
| --- | --- |
| **Use Case Name** | Create Subcontractor |
| **Use Case ID** | SC-002 |
| **Actor** | Procurement Manager |
| **Pre-Condition** | * The Procurement Manager is logged into the ERPNext system. * 2. The Procurement Manager has the necessary permissions to create a subcontractor. |
| **Post-Condition** | * The subcontractor is created and listed in the subcontractor list. * 2. Any errors in data entry are communicated to the user for correction. |
| **Basic Flow** | 1. Procurement Manager navigates to Subcontractor Creation Form:  * System displays the subcontractor creation form.  1. Procurement Manager enters subcontractor details:  * System accepts the input data.  1. Procurement Manager submits the subcontractor form:  * System validates the entered data for completeness and correctness. * If validation is successful, the system creates the subcontractor and updates the subcontractor list to include the new entry. |
| **Alternate Flow** | * **3a)** If the user enters incorrect or missing required details: * System displays an error message indicating the specific fields that need correction or completion. * Procurement Manager corrects the highlighted errors and resubmits the form. * System revalidates the form data and proceeds according to Basic Flow if valid. |

|  |  |
| --- | --- |
| **Use Case Name** | View Subcontractor Stock Balance |
| **Use Case ID** | SC-003 |
| **Actor** | Procurement Manager |
| **Pre-Condition** | * The Procurement Manager is logged into the ERPNext system. * 2. Subcontractors are already created and have associated stock balance data. |
| **Post-Condition** | The subcontractor's stock balance statistics are displayed to the user. |
| **Basic Flow** | 1. Procurement Manager navigates to Subcontractors List:  * System displays the list of subcontractors.  1. Procurement Manager selects a subcontractor:  * System opens the selected subcontractor's details page.  1. Procurement Manager opens the Stock Balance tab:  * System displays the Stock Balance Stats for the selected subcontractor. |
| **Alternate Flow** | * **2a)** If the selected subcontractor has no stock balance data: * System displays a message indicating no stock balance data is available for the selected subcontractor. |

|  |  |
| --- | --- |
| **Use Case Name** | View Subcontractor Transaction History |
| **Use Case ID** | SC-005 |
| **Actor** | Procurement Manager |
| **Pre-Condition** | * The Procurement Manager is logged into the ERPNext system. * Subcontractors are already created and have associated transaction history data. |
| **Post-Condition** | The subcontractor's transaction history is displayed to the user. |
| **Basic Flow** | 1. Procurement Manager navigates to Subcontractors List:  * System displays the list of subcontractors.  1. Procurement Manager selects a subcontractor:  * System opens the selected subcontractor's details page.  1. Procurement Manager opens the Transactions tab:  * System displays the transaction history for the selected subcontractor. |
| **Alternate Flow** | * **3a)** If the selected subcontractor has no transaction history: * System displays a message indicating no transaction history is available for the selected subcontractor. |

|  |  |
| --- | --- |
| **Use Case Name** | View and Filter Drawing Orders |
| **Use Case ID** | SC-006 |
| **Actor** | Procurement Manager |
| **Pre-Condition** | 1. The Procurement Manager is logged into the ERPNext system. 2. Drawing orders are already created and have associated statuses (In-progress and completed). |
| **Post-Condition** | The list of drawing orders, filtered by status (In-progress or completed), is displayed to the user. |
| **Basic Flow** | 1. Procurement Manager navigates to Drawing Orders List:  * System displays the list of drawing orders.  1. Procurement Manager filters orders by status (In-progress):  * System updates the list to display only In-progress drawing orders.  1. Procurement Manager filters orders by status (Completed):  * System updates the list to display only Completed drawing orders. |
| **Alternate Flow** | * **2a)** If there are no In-progress drawing orders: * System displays a message indicating no In-progress drawing orders are available. * **3a)** If there are no Completed drawing orders: * System displays a message indicating no Completed drawing orders are available. |

|  |  |
| --- | --- |
| **Use Case Name** | Custom Packaging Management |
| **Use Case ID** | UC010 |
| **Actor** | User (Warehouse Manager) |
| **Pre-Condition** | Items are ready for packaging; packaging specifications are available. |
| **Post-Condition** | Custom packaging management is integrated into the ERPNext system. |
| **Basic Flow** | 1. User navigates to Stock > Packaging > New.  * **System Response:** ERPNext displays the New Packaging form.  1. User enters packaging specifications (type, dimensions, weight).  * **System Response:** ERPNext populates the entered packaging specifications in the form.  1. User selects items to be packaged.  * **System Response:** ERPNext displays fields for selecting items.  1. User assigns packaging to the selected items.  * **System Response:** ERPNext links the packaging details to the selected items.  1. User saves and submits the packaging details.  * **System Response:** ERPNext saves and submits the packaging details.  1. System integrates custom packaging management into the ERPNext system.  * **System Response:** Confirms the packaging details. Updates the inventory with packaging information. |
| **Alternate Flow** | * **2a)** If packaging specifications are not available (after Step 2): * **System Response:** * Allows the user to save the packaging details without specifications. * User saves the packaging details without specifications. * Saves the packaging details. * Prompts the user to update packaging specifications later when available. |

|  |  |
| --- | --- |
| **Use Case Name** | Automated Inventory Updates from Subcontractors |
| **Use Case ID** | UC011 |
| **Actor** | User (Inventory Manager) |
| **Pre-Condition** | Materials have been received from subcontractors, and inventory records need updating. |
| **Post-Condition** | Inventory levels are updated automatically, and received quantities can be edited. |
| **Basic Flow** | 1. User navigates to Stock > Stock Entry > New.  * **System Response:** ERPNext displays the New Stock Entry form.  1. User selects the stock entry type as Material Receipt.  * **System Response:** ERPNext displays fields specific to Material Receipt.  1. User selects the subcontractor from whom the materials were received.  * **System Response:** ERPNext populates the subcontractor details in the form.  1. User enters the details of the received materials (items, quantities, etc.).  * **System Response:** ERPNext displays fields for entering received material details. Automatically updates inventory levels with the received quantities.  1. User reviews the entered details and makes any necessary edits to the received quantities.  * **System Response:** Allows editing of received quantities.  1. User saves and submits the stock entry.  * **System Response:** ERPNext saves and submits the stock entry. Updates the inventory records with the final received quantities.  1. System updates the inventory records and adjusts stock levels accordingly.  * **System Response:** Confirms the update of inventory levels. Reflects the changes in the stock ledger. |
| **Alternate Flow** | * **4a)** If there is a discrepancy in the received quantities (after Step 4): * **System Response:** * Notifies the user of the discrepancy. * User adjusts the quantities to match the actual received amounts. * Allows the user to edit the quantities. * Updates the inventory records accordingly. * User saves and submits the corrected stock entry. * Saves and submits the corrected stock entry. * Updates the inventory records with the corrected quantities. |