Order to Cash

# Order Placement (Somya)

Customers place an order.

Task

1. Create the web portal where customer can place an order. (Order creation will include the following information like – Shipping address, Billing address, customer details, payment details (card details), Total amount details, order status (while creating an order assume initial state is draft), Items details etc.)
2. Once order is placed customer will get the immediate notification that your order is placed successfully.
3. All the information is store into the database (Use any relational database – MySQL, Postgres etc.)
4. Create one dashboard page where customer will see the dashboard information’s (How many orders they have placed, total product they buy, total amount etc.) and create filter option to change the dashboard information by day, week, month, year, and custom date filter.
5. Validation:
6. Nonzero quantity
7. Product selection based on region
8. Voucher validation check
9. Discount validation
10. Cards expire validation
11. Create the flow diagram which explain the Order placement journey.
12. Create the documentation for Customer Order placement.

# Order Fulfillment (Deepak)

Task

1. Create the web portal where Inventory counts should be updated on the sales side in real time to avoid accepting orders that cannot be completed.
2. When an out-of-stock order does make it to fulfillment, it must be flagged immediately. Then you need to alert the customer and cancel the order, which can help avoid billing issues.
3. All the information is store into the database (Use any relational database – MySQL, Postgres etc.)
4. Create one admin and one (Floor manager) dashboard page where order fulfillment will see the dashboard information’s (How many orders are going for out-of-stock, total fulfillment etc.) and create filter option to change the dashboard information by day, week, month, year, and custom date filter.
5. Validation:
6. Inventory and Stock validation (Region wise)
   1. In case of privileged customer, inventory/stock across all regions to be checked.
   2. Notification alert to supplier in case of inventory going below threshold value.
7. Create the flow diagram which explain the Order Fulfillment journey.
8. Create the documentation for Customer Order Fulfillment.

# Order Shipping (Vicky)

Task

1. Create a web portal where shipping teams plan shipments, design pickup schedules based on region and get orders delivered to customers on time.
2. All the information is stored into the database (Use any relational database – MySQL, Postgres etc.)
3. Create one admin and one (Shipping manager) dashboard page where order shipping will see the dashboard information (How many orders shipped, Number of orders shipped on region etc.) and create filter option to change the dashboard information by day, week, month, year, and custom date filter.
4. Validation:
5. Package details to be verified with the order details
6. Update shipping details for an order based on normal/urgent delivery for a region.
7. Maintain a correlation between the shipment details, vehicle details, and type of delivery.
8. Maintain a tabular dataset to display delivery attempts and change in delivery details (address, region) in case of non-receipt or address not found.
9. Create the flow diagram which explains the Order Shipment journey.
10. Create the documentation for Customer Order Shipment.

# Customer Invoice (Anmol)

Task

1. Create the web portal where finance team maintains details of incoming/pending/cleared invoice.
2. All the information is store into the database (Use any relational database – MySQL, Postgres etc.)
3. Create one dashboard page where invoice details will be available (How many invoices, pending, cleared etc.) and create filter option to change the dashboard information by day, week, month, year, and custom date filter.
4. Validation:
5. Validate product pricing and discount (if any) with the order details.
6. In case of return/replacement, invoice to be re-generated accordingly.
7. Create the flow diagram which explain the Customer Invoice journey.
8. Create the documentation for Customer Invoice.

# Payment Collection

1. When a payment is over-due, and the time for clearance is missed, it must be flagged immediately. Alert the customer about penalty charges, which can help avoid billing issues.
2. Create the web portal where finance team maintains details of incoming/pending/cleared payment.
3. All the information is store into the database (Use any relational database – MySQL, Postgres etc.)
4. Create one dashboard page where payment details will be available (How many payments are pending, cleared etc.) and create filter option to change the dashboard information by day, week, month, year, and custom date filter.
5. Create the flow diagram which explain the Payment Collection journey.
6. Create the documentation for Payment Collection.

# Report and Analysis (Admin and Support Team Portal)