Assignment 5: Define Bounded Contexts for an E-commerce Website

Instructions:

1. Identify the main functional areas of an E-commerce website, such as product catalog, shopping cart, checkout, order management, etc.
2. For each functional area, identify the **main concepts and sub-concepts**. For example, for the product catalog, the main concepts could be products, categories, and attributes.
3. Group the concepts and sub-concepts into logical domains, which represent distinct business capabilities or processes. These domains should have well-defined boundaries and should be cohesive and loosely coupled with other domains.
4. Define the bounded contexts for each domain, which represent a distinct and independent part of the system that has its own language, models, and rules. These bounded contexts should have a clear boundary and should communicate with other bounded contexts through explicit interfaces and APIs.

Example:

Functional area: Product catalog

Concepts and sub-concepts:

* Products
* Categories
* Attributes
* Reviews
* Inventory

Logical domains:

* Product management
* Category management
* Attribute management
* Review management
* Inventory management

Bounded contexts:

* Product catalog API
* Category API
* Attribute API
* Review API
* Inventory API

Assignment: Design APIs using Bounded Context approach

Instructions:

1. Choose an industry/domain (such as healthcare, finance, or retail) for which you want to design APIs.
2. Identify the different bounded contexts within the industry/domain that will require APIs to be designed.
3. For each bounded context, identify the key entities and operations that will be needed to create the APIs.
4. Create a list of APIs that need to be designed, along with their purpose, input parameters, output parameters, and any special considerations that may be required.
5. Consider the interdependence between different bounded contexts and ensure that the APIs are designed in a way that facilitates their interaction.
6. Write a brief description of the overall design approach you took, highlighting any specific challenges you faced and how you overcame them.

Example solution:

Industry/domain: E-commerce

Bounded contexts:

1. Customer Management
2. Order Management
3. Product Catalog Management
4. Cart Management
5. Payment Management
6. Shipping Management
7. Review Management

APIs:

* Customer Management
  + Create new customer - API calls
  + Get customer details
  + Update customer details
  + Delete customer
* Order Management
  + Create new order
  + Get order details
  + Update order status
  + Cancel order
* Product Catalog Management
  + Create new product
  + Get product details
  + Update product details
  + Delete product
  + Cart Management
  + Add item to cart
  + Get cart details
  + Update cart
  + Delete item from cart
* Payment Management
  + Process payment
  + Get payment status
  + Refund payment
* Shipping Management
  + Create new shipment
  + Get shipment details
  + Update shipment status
  + Cancel shipment
* Review Management
  + Create new review
  + Get review details
  + Update review details
  + Delete review

Interdependence between bounded contexts:

* **Customer Management and Order Management: The customer ID is required to create a new order.**
* **Order Management and Payment Management: The order ID is required to process payment. CRM -> Payment -> ERP**
* Order Management and Shipping Management: The order ID is required to create a new shipment.