Assignment 2: Use-cases for Microservices in E-commerce Company

# Objective: To understand the requirements of the e-commerce company and create microservices to fulfill the same.

Task: As a Cloud Architect, you have to identify the different use cases and requirements of an e-commerce company and design microservices that can handle these use cases. The use cases should include alternate courses and exceptions to be taken care of in the microservices design.

Checklist:

1. Identify the use cases of the e-commerce company and document them.
2. Identify the requirements for each use case and document them.
3. Identify alternate courses of action that may arise in each use case and document them.
4. Identify exceptions that may arise in each use case and document them.
5. Design microservices that can handle the use cases, alternate courses, and exceptions.
6. Ensure that the microservices are scalable, reliable, and secure.
7. Test the microservices to ensure that they meet the requirements, alternate courses, and exceptions identified.

## Deliverables:

1. A document outlining the use cases, requirements, alternate courses, and exceptions.
2. A design document for the microservices that can handle the use cases, alternate courses, and exceptions.
3. Test reports showing that the microservices meet the requirements, alternate courses, and exceptions identified.

Note: You can use any cloud platform for designing and deploying the microservices, but AWS would be preferred.

## Questions to ask:

1. What are the primary use cases for the e-commerce platform?
2. What are the different alternate courses and exceptions that need to be considered for each use case?
3. What are the specific requirements for each microservice needed to support each use case?
4. How can the microservices be designed to be scalable and fault-tolerant?
5. How will data be shared between the microservices?
6. How will communication between the microservices be handled?
7. What security and compliance requirements need to be considered for each microservice?
8. How will the deployment and management of the microservices be handled?

## Checklist:

1. Identify and prioritize the use cases for the e-commerce platform.
2. Map out the alternate courses and exceptions for each use case.
3. Identify the microservices needed to support each use case.
4. Define the requirements for each microservice, including scalability, fault tolerance, and data sharing.
5. Determine the communication protocols between the microservices.
6. Ensure that security and compliance requirements are considered for each microservice.
7. Define a deployment and management strategy for the microservices.
8. Test and validate the microservices to ensure they meet the requirements and perform as expected.

# Solution:

## Use Case: Product Search and Recommendation Service

Requirements:

* High availability and scalability to handle millions of requests per second
* Real-time data processing and analysis for personalized recommendations
* Integration with existing e-commerce platform and customer data
* Secure data storage and processing to protect customer privacy
* Cost optimization to manage resource usage and minimize expenses

Alternate Courses:

* Use a third-party recommendation service instead of building in-house
* Implement a hybrid solution with some components on-premises and some in the cloud
* Implement a federated search and recommendation service to include products from third-party sellers

Exceptions:

* Service downtime or slow response time impacting customer experience
* Data breaches or unauthorized access to customer data
* Increased costs due to resource usage or unexpected traffic spikes

## Use case: Product Recommendation Engine

Requirements:

* Ability to provide personalized product recommendations to customers based on their browsing and purchase history
* Real-time processing of customer data to provide relevant recommendations
* Ability to handle a high volume of requests during peak traffic times

Alternate Courses:

* Use pre-built recommendation engine services from AWS like Amazon Personalize
* Build a custom recommendation engine using microservices architecture
* Use a combination of pre-built and custom recommendation engines

Exceptions:

* Lack of customer data to provide relevant recommendations
* Privacy concerns around the use of customer data
* Technical limitations in processing real-time data for recommendations

## Roadmap to create resilient architecture

1. Understand the requirements: The first step is to understand the requirements of the e-commerce company and identify the use cases for which microservices are required. This can be done by:

* Analyzing the existing monolithic application and identifying the different functionalities.
* Identifying the pain points in the current system and the areas where scalability is required.
* Understanding the business goals and objectives of the company.

1. Identify alternate courses: Once the requirements are identified, the next step is to identify the alternate courses that can be taken to fulfill those requirements. This can be done by:

* Identifying the different microservices that can be developed to fulfill the requirements.
* Evaluating the pros and cons of each microservice and its impact on the overall system.
* Identifying the dependencies between different microservices and how they can be integrated.

1. Identify exceptions: While identifying the alternate courses, it is also important to identify the exceptions that may arise in the system. This can be done by:

* Identifying the different scenarios where the microservices may fail or behave unexpectedly.
* Identifying the impact of these failures on the overall system and the users.
* Developing a plan to handle these exceptions and minimize their impact.

## Checklist before starting Development phase:

1. Have the requirements been identified and documented?
2. Have the alternate courses for fulfilling the requirements been identified and evaluated?
3. Have the dependencies between different microservices been identified and documented?
4. Have the exceptions that may arise in the system been identified and documented?
5. Have plans been developed to handle the exceptions and minimize their impact?
6. Have the scalability and performance requirements been considered while designing the microservices?
7. Have the security and compliance requirements been considered while designing the microservices?
8. Have the testing and deployment strategies been defined for the microservices?
9. Have the monitoring and logging strategies been defined for the microservices?

By following this assignment, the cloud architect can effectively design and implement microservices for the e-commerce company that meet the business goals and objectives while addressing scalability, performance, security, compliance, and exception handling requirements.