

1- SAP Commerce (Hybris) nedir? Hangi amaçlarla kullanılır? Kullandığı teknolojiler nelerdir? Kısaca açıklayınız.

SAP Commerce Cloud, eski adıyla Hybris, bulut tabanlı bir omni-channel e-ticaret çözümüdür. Tüm müşterilerin özel ihtiyaçlarına göre uyarlanabilen; online, mobil, mağaza, depo, sosyal medya, çağrı merkezi gibi müşteriye temas edilen noktaları birleştirerek; tekil bir müşteri deneyimi yaşanmasına olanak kılar. Tüm müşteri deneyim verisinin bütünsel olarak değerlendirilmesini sağlar. Ayrıca müşterilerinin hem B2B hemde B2C pazarlarında etkili bir şekilde satış yapmalarını sağlayan bir e-ticaret platformudur. Bu özellikler online, mobil ve mağaza içi üzerinden satış yapabileceğiniz bir sistem anlamına gelir. Paketin en önemli parçalarından biri ise Ürün İçerik Yönetimi (PCM) modülüdür. Hybris, Java tabanlı bir uygulamadır. Web, uygulama ve veritabanı katmanlarından oluşan 3 katmanlı bir mimarinin parçası olarak çalışır. Web sunucuları, genellikle trafiği uygulama sunucularına yönlendirir. Veritabanı platformu herhangi bir SQL tabanlı çözümden oluşabilir ancak MySQL küçük ve orta ölçekli işletmeler için kullanılır ve daha fazla kurumsal dağıtım için Oracle ve benzerleri de kullanılır.

SAP Hybris bu teknolojileri kullanır:

Frameworks: Spring Framework features/modules like Spring AOP, Spring MVC, Spring RestFul, Spring Security - caching, Persistence like (ZK, Hibernate, JPA, etc), Search, Import / Export, Internationalization, Solr

FrontEnd: JSF and other UI technology

Databases: You can use any database like Mysql, Oracle, SQL server, etc...

App Server: You can use any Hybris Server, JBoss, IBM Websphere, Oracle BEA, SpringSource Server, etc...

User Interfaces: Product Cockpit, CMS Cockpit, Print Cockpit, Admin, HMC, Multi-Channel Cockpit
Using SAP Hybris, we can develop B2B, B2C

2- Birbirinden bağımsız iki platformun birbiriyle haberleşmesi nasıl sağlanabilir? Örneğin, X platformu Java ile yazılmış olsun, Y platformu C# ile. Bu iki platformun bir biri ile iletişim halinde request-response ilişkisi kurması gerekiyor. Bu yapıyı nasıl sağlarız? Bu iletişim sırasında güvenlik nasıl sağlanır?

Bu soruyu cevaplamak için benim en uygun bulduğum yöntem Mikroservislerinde desteklediği HTTP Service ve RESTFUL Service kaynaklarını kullanarak birbirleri ile haberleşmelerini sağlamak. Bu bağlantı servislerini seçmemdeki amaç bir web sitesi ile mobil uygulama aynı verileri kullanıyor olsalarda eşit şekilde müşteri yoğunluğu ve sunucu baskısı göstermeyerek sistemi yorabilirler. Bu yüzden de REST API çağrıları, herhangi bir önceki çağrıya bağlı olmadan yeni işlemler gerçekleştirmeye izin verir, böylece her çağrı bağımsız ve ilgisiz diğerlerinden ayrı olarak ele alınır. Bu, yüksek performans ve daha az sunucu kaynağı kullanımına olanak tanır. İki arasındaki bağlantıyı sağladıktan sonrada REST API tasarımında HATEOAS (Hypermedia as the Engine of Application State) prensibi ile OAuth, JWT gibi token tabanlı sistemler için kullanılan API erişimi yetkilendirme yöntemi ile güvenlik sorununa çözüm getirir. Eğer uygulamanın bu bağlantı ile güvenlik açığı gibi durum olursa burada Postman, Cucumber, and k6. gibi servisler ile test senaryosu gerçekleştirebilir.

3- SOLR Nedir? Kullanım alanlarını araştırınız. Kurumsal bir projede kullanılabilecek iki farklı kullanım alanı örneği veriniz.

SOLR en basit ifadeyle tanımlamak gerekirse Google gibi bir arama motorudur. Web sayfaları, xml belgeleri, düz dosyalar, rdbms vb. gibi herhangi bir ortamdan içerik indekslemenize yardımcı olur, Kullandığı yerlerden bazıları , web sitesi araması, yani Wordpress veya Drupal e-ticaretiniz için arama motoru kullanımı, mağazanız için arama ve analiz yapma , zamana dayalı veri analizi yapma , günlük analizi, tıklama analizi, trend analizi vb. gibi alanlarda kullanılabilir.

Kurumsal bir proje kullanım örneği:

Çok fazla kullanıcıya sahip bir E-ticaret sitesinin kendi arama motorunu kullanmak yerine SOLR getirdiği atama kutularını kullanarak arama sonuçlarını milisaniye seviyelerine indirebilir. Solr, metin içeriğini tam metin araması ile indeksleyebilir ve arayabilir. Apache Lucene tabanlı olduğu için hızlı arama özellikleri sunar.

Yabancı kaynaklı bir haber sitesi için arama motorları büyük miktarda veri ile uğraşmak zorundadır ve bu nedenle arama motoru iyi geliştirilmelidir. Arama motorunu makalelerin tüm içeriğini çok hızlı ve doğru bir şekilde arayabiliriz. Arama motoru, dilin zorlu çekim biçimlerini ve diğer ilgili faktörleri de dikkate alarak aramayı kolaylaştırır.

4-Java'da 100 adet random sayıya sahip bir liste oluşturun.

- Daha sonra bu listenin bir kopyasını oluşturun.
- 0 ile 100 arasında rastgele bir sayı üretin.
- Kopya listedeki bu random sayının olduğu indisteski değeri silin
- Şimdi elinizde iki adet liste var ve kopya listede orjinal listeye göre bir eleman eksik.
- Hangi elemanın eksik olduğunu bulan bir metot oluşturun.

```
import java.util.ArrayList;
import java.util.List;
import java.util.Random;

public class Main {
    public static void main(String[] args) {

        List<Integer> randomList = generateRandomList(100);

        List<Integer> kopyaList = new ArrayList<>(List.copyOf(randomList));

        Random random = new Random();
        int randomSay = random.nextInt(101);

        int index = kopyaList.indexOf(randomSay);
        if(index != -1){
            kopyaList.remove(index);
        }

        int kayakSaying = join(randomList,kopyaList);
        System.out.println("Kayıp sayı {}: " + kayakSaying);
    }

    public static List<Integer> generateRandomList(int n){ 1 usage
        List<Integer>list = new ArrayList<>();
        Random random = new Random();
        for (int i = 0; i<n;i++){
            list.add(random.nextInt(101));
        }
        return list;
    }

    public static int join(List<Integer> list1, List<Integer> list2){ 1 usage
        int xor = 0;
        for(int num : list1){
            xor ^=num;
        }for(int num :list2){
            xor ^=num;
        }
        return xor;
    }
}
```