CEVAPLAR

11- RDS servisinin temel özelliklerini belirtiniz ve gerçek dünyadan bir kullanım örneği veriniz.

Cvp:

1-RDS Amazon Database Relational Database Service dir. Bu servis bulut üzerinde ilişkisel veritabanımızı kolayca kurar, ölçekler ve çalıştırır.

2-Çokça zaman harcadığımız database yönetim işleriyle uğraşacağımıza, RDS bize düşük maliyetli ve yeniden boyutlanabilir alanlar sunar ve DB yönetim işlerini elimizden alır. Bu da bizim uygulamamıza ve domainimize daha çok odaklanmamızı sağlar.

3- Yeni bir DB server satın aldığımızda bu server ile beraber onun belleğini, diskini, I/O kapasitesini yani tüm özelliklerini bir bütün olarak almak zorunda kalırız. Fakat RDS bunların hepsini bağımsız olarak ayrı ayrı ölçekleyebilir. Örneğin daha çok cpu yada disk alanına ihtiyacınız oldu yada daha az I/O kapasitesi istiyorsun bunların hepsini RDS ile istediğin anda tahsis eder ve hemen kullanmaya başlarsın. Böylece server bakımı yükseltmesi gibi işlerle uğraşmaz ve tasarruf sağlarsın.

4-RDS yedeklerinizi yönetir. Hataları otomatik olarak belirler ve kurtarır.

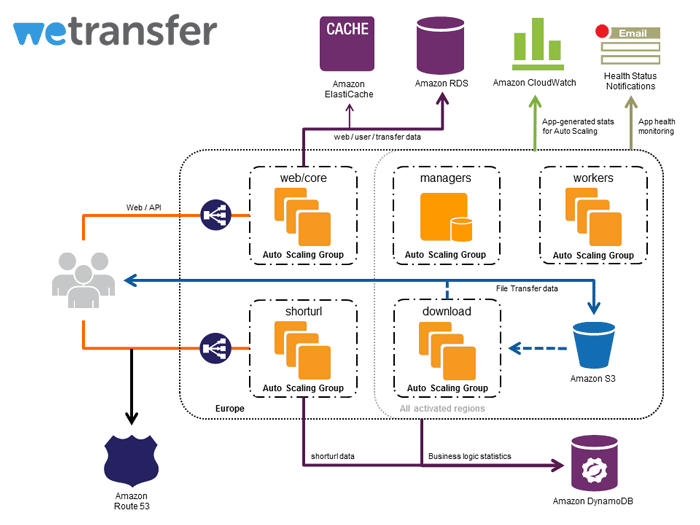
5- Şu ana kadar aşina olduğunuz MySQL, PostgreSQL, Oracle, Microsoft SQL Server dan istediğiniz ürünü kullanabilirsiniz. Bunların hepsi için RDS servisi mevcuttur.

6- RDS in güvenlik olaylarına ek olarak, sizin RDS DB nize kimin ne zaman eriştiğini yada erişeceğini ne izinlerine sahip olduğunu yada olacağını AWS IAM kullanarak tanımlayabilirsiniz.

7-Native call tanımladığınız bir çağrı server 2010 da çalışabilir fakat 2015 sürümünde bu çağrı tanınmayabilir. Servis API leri bu problemleri halleder.

Örneğin okulumuzun ebys sisteminde AWS RDS hizmetini satın alarak ilişkisel tablolarımızı bunun üzerinde tutabiliriz. Böylece yukarıdaki özelliklerin ve avantajların tamamına uygulamamız sahip olur. Fakat burada unutmamamız gerekir ki öğrenci verileri Amazon datacenterlarında tutulacak. Bunun yasal bir probleme yol açmamasına özen göstermeliyiz. Bu sebeptendir ki bankacılık sektörüde AWS RDS ini kullanamaz.

12- Çoklu veri servisi mimarisinin getirdiği avantajları ve dezavantajları örnek bir uygulama mimarisi kapsamında tartışınız.



Cvp: Yukarıdaki gibi bir mimariye sahip uygulamamız olduğunu düşünelim. Bu mimaride uygulamamız AWS S3, DynomoDB, RDS ve Elastic Cache data mimarilerini kullanmaktadır. Bu durum artık günümüz uygulamaları için bir lüks olmaktan çıkmış çoğu uygulama için bu senoryada olduğu gibi çoklu data mimarisi kullanılması zorunlu olmuştur.

**Avantajlar**

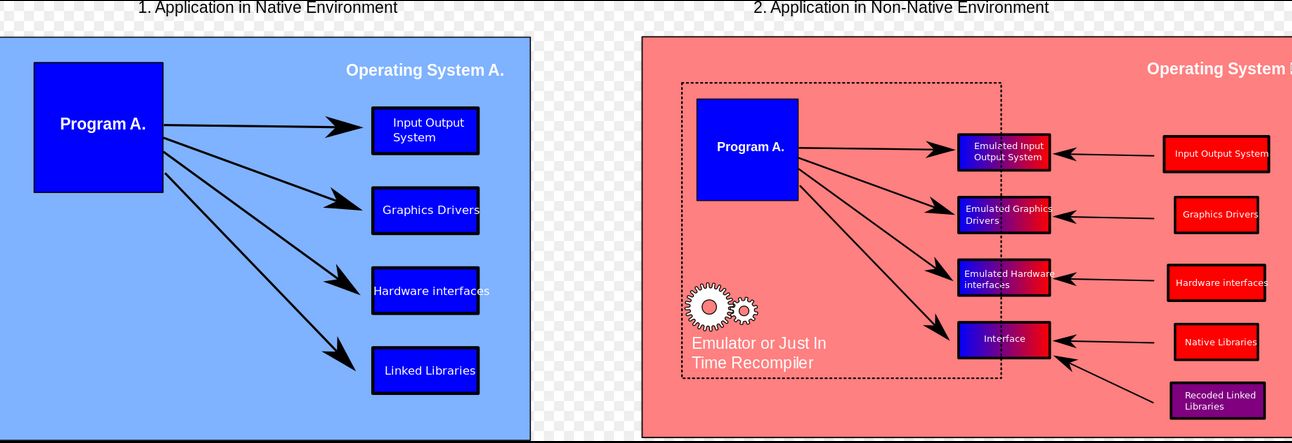
1. Bu mimarideki en büyük avantaj bu kadar çeşitliliğe sahip data mimarisinin her birini yönetmek için ayrı ayrı data serverler almamıza gerek kalmayacak olmasıdır. Bu gerçekten çok büyük bütçeyle projeye başlamak zorunda kalmamıza sebeb olurdu.
2. Ayrı ayrı server almış olsaydık hepsini uygun yapılandırıp, yönetmemiz gerekecekti fakat bu data mimarilerini birer Amazon Servisi olarak alarak bunlardan kurtuluruz.
3. Diğer en büyük avantajı eğer ayrı ayrı bu serverları alsaydık, uygulamamızın içindeki business layerımızla data layerımız arasındaki persistence Layerimiz çok karışık olacak, Sql için Entity framework, diğerleri için onlara uygun frameworkları kullanmak zorunda kalacaktık, buda katmanımızı karmaşıklaştıracak framework sorunları ile uğraşmamız gereken bir çok sorunu ortaya çıkaracaktı. Amazon servisleri bu problemlerden bizi kurtaracaktır.

**Dezavantajlar**

1. Her nekadar AWS ler işimizi kolaylaştırsada çoklu data servisleri uygulamayı karışık hale getirecektir.
2. Uygulamayı yönetmek gitgide zorlaşacaktır.
3. Developer her nekadar bu servisleri kullansada kodunda bu servisler arasındaki ilişkileri kendi takip etmek zorunda kalacaktır.
4. Developer açısından yazdığı kod miktarı arttığı için karmaşıklık artacak ve bu karmaşıklığıda yönetmesi gerekecektir.

Fakat sonuç olarak bu karmaşıklık data servislerini kullanmadığımız zaman çok daha fazla olacağından mimariyi bu şekilde yapılandırmak faydalı ve zorunlu olacaktır.

**13­)** Sanallaştırmanın uygulandığı temel alanı ifade edip, aynı fikrin uygulama geliştirme ortamlarına uygulanması konusundaki görüşünüzü açıklayınız.

Cvp: 

Sanallaştırma esas olarak IaaS larda, sunucu hizmetlerinde sanal makine oluşturma işlemlerinde kullanılır bu sayede sistem kaynakları tek olmasına rağmen bunlar sanal olarak ayrıştırılıp her biri ayrı makine olarak kendine has OS yüklü olarak bir platform olarak çalışır.

Bu fikir uygulama geliştirme ortamlarından Visual Studio’ya uygulandığını düşünelim. Böylece bu uygulamanın kaynakları gerçek ortamda tek olmasına rağmen vereceğimiz servisler ile kullanıcılara bu uygulamanın kaynaklarını ve özelliklerini kendi istedikleri şekilde yapılandırmalarına imkan verebiliriz. İsterlerse uygulamayı daha iyi cpu, disk, I/O imkanları ile kullanabilirler istedikleri bu imkanlara göre bir lisans ücreti tahsis edebilir. Daha sonrada uygulama içindeki istedikleri özellikler için onlara API ler sağlayabiliriz. Örneğin Visual studio içinden mobile uygulama, Mvc projesi oluşturmayan birini bu menüleri görmesine gerek yoktur ve kendi ürünü için bu hizmetleri istemeyebilir ve lisans ücreti istediği hizmetlere göre yeniden yapılandırılabilir. Böylece kullanmadığı hizmetlere para ödememiş ve programı kullanmadığı özelliklerin karmaşıklığından kurtulmuş olur. Sonradan kullanmaya karar verirse gerekli API yi satın alıp IDE arayüzüne bunu ekleyebilir. Yada istemediği özelliği IDE arayüzünden çıkarabilir. Bu işlemleri yukarıdaki sağ taraftaki şeklin ilk 3 katmanı ile donanımsal özellikleri belirleyebilir. Interface katmanı ilede istediği özelliklerin API lerini alabilir.

14­) Alana özel PaaS ile alandan bağımsız PaaS arasındaki farkı açıklayınız.

Cvp: Bu soruyu en güzel açıklayan örnek salesforce.com ile force.com dur. İlk olarak bir SaaS olarak uygulama geliştirilmeye başlandı. Bu süreçte bir çok tecrübe elde edinildi. Bu uygulamanın Platformu bu yeni tecrübelere göre configire edildi ve gitgide Platformun configrasyonu bu uygulama açısından optimum düzeye gelmeye başladı. Bu sırada uygulamanın Use Caseleri, SSD leri, System function catalogları, Domain modeli, Design modeli olgunlaştı bunlar RUP dökümanlarına girdi ve bunlarla ilgili API ler yazıldı ve geliştirildi her karşılaşılan problem bu şekilde çözüldükten ve sistem olgunlaştıktan sonra sistemi yapanlar ellerindeki bu birikimden diğer bu domainde uygulama geliştirmek isteyenler için ücret karşılığında ellerindeki bu API leri kullanan IDE benzeri bir ortam sundular bu sundukları ortamda force.com’du. Bu ortam domain dependent bir ortamdır yani alana özel ve bağımlı bir PaaS örneğidir. Bu domainde uygulama geliştirmek isteyenler force.com u kullanarak kendilerine özel CRM uygulamaları üretebilirler.

Alandan bağımsız Domain independent PaaS ‘lar ise belli bir domain için oluşturulmuş değillerdir her uygulamanın ihtiyaç duyabileceği her domainde kullanılabilen özellikler örneğin Searching, Depolama, VT işlemleri vb. gibi servislerdir. Bunlara örnek olarak AWS ler gösterilebilir, örneğin S3 servisi belli bir domain için değil her domainde kullanılabilecek bir servistir yani diğer AWS ler gibi Alandan bağımsızdır. Diğer bir örnekte Google Cloud servisleri verilebilir.

15­) Bulut bilişim ile uygulamaların sahip olacağı yeni iş modellerinin temel özelliklerini açıklayınız.

Cvp: Bu soruya en güzel yanıt Oracle firmasının bu firmaya girmek isteyen arkadaşımıza araştırması için verdiği konudur. Bu konuda arkadaşımızın Güvenlik-Bulut Bilişim-Performans konularını açıklanması istenmiş. Bu da gösteriyor ki yeni iş modelleri bu kavramlar üzerine kurulacak yani bir uygulamanın avantajları göz önünde tutularak bulut ta çalışması istenecek. Uygulamanın güvenliğinin bulut üzerinde sağlanması bu iş modelinde yine çok önemli olacak. Ve de performans gibi fonksiyonel olmayan gereksinimlerin response time, I/O miktarı, testler vb. non fonksiyonel özellikler bu iş modelinde çok daha fazla öneme sahip olacak. Ve bu özelliklerin test edilebildiği hızlı servislere ihtiyaç duyulacak.

Ayrıca yeni iş modellerinde müşteriler kapsamlı uygulamaların demo larını kısıtlı sürelerde kullanıp fayda zarar analizleri ile stres testlerini yapmak isteyecekler. Gerçek verilerini bu sistem üzerinde test etmek isteyecekler. Bu istekler ve bu isteklere cevaplar freemium economics modelini uygulamamızı gerektirecek ve bu olanakları bize en etkin sağlayan Serrvisleri kullanabilmemiz yeni iş modellerinde çok önemli olacak.