

Jenkins

Jenkins, Sürekli Entegrasyon (CI: Continuous Integration) yöntemi için kullanılan java ile yazılmış açık kaynak kodlu bir otomasyon sunucusudur. Yazılım geliştirme süreçlerini otomatize etmemize yarar. Jenkins, belirli bir sunucuda ve portta çalışır, belirlenen kaynaktan projeye ulaşır ve istenen işlemleri gerçekleştirir. Sonuçlarını belirlenen kişilere iletir. Bu sayede projemiz sürekli test edildiği için hatalar hızlıca tespit edilir ve çalışır bir vaziyette tutulur. Jenkins, manuel olarak yapılan build, test ve deploy gibi işlemleri uygular ve bu süreçte yaşanabilecek tüm aksaklık ve iletişim eksikliğini en aza indirir.

Örnek olarak; Jenkins yazılım geliştirme sürecinde belirlenen kaynaktan projeye ulaşır ve belirlenen şartlarda çalışıp projeyi derler. Derleme başarılı ise testleri çalıştırır. Bu iki aşamada herhangi bir sorunla karşılaşmazsa 3. aşama olan deploy işlemini gerçekleştirir. Bu durum Sürekli Teslimat (CD: Continuous Delivery) işlemi olarak nitelendirilebilir. Sürekli teslimatta amaç projeyi sürekli güncel ve çalışır bir vaziyette tutmak ve canlı ortama benzer bir ortama aktarmaktır. Jenkins eğer build veya test aşamasında herhangi bir hata ile karşılaşırsa deploy işlemini gerçekleştirmez ve geliştiriciyi bilgilendirir. Bu süreç sürekli tekrar eder. Jenkins bunlara ek olarak statik analiz, versiyon kontrol sistemi, dağıtım gibi birçok geliştirme sürecini entegre eder.

Jenkins, projemizi Git'ten çeker, build eder, yazılan test adımlarını belirlenen node'larda gerçekleştirir, sonuçlarını belirlenen kişilere iletir. Bunu sürekli hale getirir ve DevOps kültürünü oluşturur.

Jenkins Kullanmanın Avantajları

- Jenkins, çok açık olan topluluk tarafından yönetiliyor. Her ay halka açık toplantılar düzenlerler ve Jenkins projesinin geliştirilmesi için halktan girdiler alırlar.
- Şimdiye kadar yaklaşık 280 bilet kapatıldı ve proje her üç ayda bir kararlı sürüm yayınlıyor.
- Teknoloji büyüdükçe Jenkins de büyüyor. Şimdiye kadar Jenkins'in eklenti veritabanında yayınlanan yaklaşık 320 eklentisi var. Eklentiler sayesinde Jenkins daha da güçlü ve zengin özelliklere sahip olmaktadır.
- Jenkins aracı, bulut tabanlı platformlarda Jenkins'i dağıtabilmeniz için bulut tabanlı mimariyi destekler.
- Jenkins'in popüler olmasının nedeni, geliştiriciler için bir geliştirici tarafından oluşturulmuş olmasıdır.
- Büyük topluluk desteğine sahip açık kaynaklı bir araçtır.
- Ücretsizdir.
- Java ile oluşturulmuştur ve bu nedenle tüm büyük platformlara taşınabilir.

Jenkins Kullanmanın Dezavantajları

- Jenkins çok güçlü bir araç olmasına rağmen, zorlukları da vardır.
- Arayüzü güncel değil ve mevcut UI trendlerine kıyasla kullanıcı dostu değildir.
- **Jenkins** birçok geliştirici tarafından sevilse de, bakımı o kadar kolay değil çünkü Jenkins bir sunucu üzerinde çalışıyor ve etkinliğini izlemek için sunucu yöneticisi olarak bazı beceriler gerektirir.
- Birçok insanın Jenkins'i uygulamamasının nedenlerinden biri, Jenkins'i kurma ve yapılandırmadaki zorluğundan kaynaklanmaktadır.
- Bazı küçük ayar değişiklikleri nedeniyle sürekli entegrasyonlar düzenli olarak bozulabilmektedir. Buda özel bir ilgi gerektirir.

Devops

Devops, BT personeli ve yazılım geliştiriciler arasındaki boşluğu kapatmayı amaçlayan bir yazılım geliştirme stratejisidir.

DevOps, Development ve Operations kelimelerinin birleştirilmesi ile oluşturulan bir kısaltma olarak ifade edilebilir

DevOps, tasarım başlangıcından, geliştirme süreçlerine kadar tüm programlama işlemlerinin yürütüldüğü ekip çalışmasını belirtmektedir. DevOps aynı zamanda, **IT** (Information Technology) Türkçe karşılığı ile Bilgi Teknolojileri alanında çalışmalarını yürüten ekiplerin, en üst seviyedeki verimliliklerini, sonuçlandırma yeteneklerini yükselten araçların ve felsefelerin tamamını ifade etmektedir.

DevOps, kurumların ürünleri geleneksel yazılım geliştirme ve altyapı yönetim süreçlerini kullanan kurumlara göre daha hızlı geliştirmesini ve iyileştirmesini sağlayarak, uygulama ve hizmetleri yüksek hızda sunma becerisini artıran kültürel felsefelerin, yöntemlerin ve araçların birleşimidir. Bu hız, kurumların müşterilerine daha iyi hizmet sunmasına ve piyasada daha etkili bir şekilde rekabet etmesine imkan tanır.

Bir DevOps modelinde geliştirme ve operasyon ekipleri artık birbirinden kopuk değildir. Bazen bu iki ekip tek bir ekip haline getirilir ve ekipteki mühendisler geliştirme ve testten dağıtım ve operasyona kadar bir uygulamanın yaşam döngüsünün tamamında çalışırken tek bir işleyle kısıtlı olmayan bir dizi beceri edinir.

DevOps'un Avantajları

- Hız
- Hızlı Teslim
- Hızlı Teslim
- Ölçek
- Geliştirilmiş İş Birliği
- Güvenlik

DevSecOps

DevSecOps, geliştirme, güvenlik ve operasyonlar kelimelerinin kısaltmasıdır. Amacı, güvenlik karar ve eylemlerinin, geliştirme ve operasyon karar ve eylemleriyle aynı ölçek ve hızda uygulanması için herkesin güvenlikten sorumlu olmasını sağlamaktır.

DevOps “sistem geliştirme yaşam döngüsünü kısaltmayı ve yüksek yazılım kalitesiyle sürekli teslimat sağlamayı amaçlayan yazılım geliştirme ve bilgi teknolojisi işlemlerini birleştiren bir dizi uygulama]” olarak tanımlanmıştır.

DevSecOps, uygulama ve altyapı güvenliğini Çevik ve DevOps süreçleri ve araçları ile sorunsuz bir şekilde bütünleştirir. Güvenlik sorunlarını ortaya çıktıkları zamanda (ve üretime alınmadan önce) ele alır; bu zaman düzeltmenin de daha kolay, hızlı ve ucuz olduğu zamandır. Ayrıca DevSecOps, uygulama ve altyapı güvenliğini; bir güvenlik silosunun tek sorumluluğu olmaktan çok, geliştirme, güvenlik ve BT operasyon ekiplerinin ortak sorumluluğu haline getirir. Güvenli yazılımın, yazılım geliştirme döngüsünü yavaşlatmadan teslim edilmesini otomatikleştirerek, "daha güvenli, daha hızlı yazılıma" —DevSecOps sloganı— olarak tanır.

DevSecOps'un avantajları

- Hızlı, uygun maliyetli yazılım teslimi
- Geliştirilmiş, proaktif güvenlik
- Hızlandırılmış güvenlik açığı yamaları
- Modern geliştirmeyle uyumlu otomasyon
- Yinelenebilir ve uyarlanabilir bir süreç

Dockerfile

Docker, uygulamalarınızı hızla derlemenize, test etmenize ve dağıtmanıza imkan tanıyan bir yazılım platformudur. Docker, yazılımları kitaplıklar, sistem araçları, kod ve çalışma zamanı dahil olmak üzere yazılımların çalışması için gerekli her şeyi içeren container adlı standartlaştırılmış birimler halinde paketler.

Docker, kodunuzu çalıştırmanın standart bir yolunu sağlayarak çalışır. Docker, container'lara yönelik bir işletim sistemidir. Bir sanal makinenin sunucu donanımını sanallaştırmasına (doğrudan yönetme gereksinimini ortadan kaldırma) benzer şekilde container'lar da bir sunucunun işletim sistemini sanallaştırır. Docker her sunucuya yüklenir ve container'ları oluşturmak, başlatmak veya durdurmak için kullanabileceğiniz basit komutlar sağlar.

Bir Container'ın çalışması için,

- İlk olarak Dockerfile oluşturulur.
- İkinci aşamada, Dockerfile üzerinden Image oluşturulur.
- Son olarak da bu Image'den Container'ı çalıştırırız.

Dockerfile Avantajları

- Sunucu yapılandırmaya gerek yoktur.
- Uygulama bağımlılıklardan izole ediliriz.
- Docker, Hypervisor kullanmaz ve tam yüklü bir işletim sistemi barındırmaz bu sayede saniyeler içinde çalışır ve kullanıma hazır hale gelir.
- Docker, yazılımlarınızın tüm altyapı gereksinimlerini kod olarak saklar. (versiyonlama) Bu Docker'ın en önemli özelliklerinden biridir. Bu özellik sayesinde yazılımınızı farklı servis sağlayıcıları üzerinde kolaylıkla gezindirebilir, çoğaltabilir veya paylaşabilirsiniz.
- Docker, çok az kaynakla büyük işler yapabilmektedir. Bunun ana sebebi kullandığı konteyner teknolojisidir.
- Uygulamalarınıza beklenmedik bir şekilde yüksek trafik geldiğinde saniyeler içerisinde 1000'lerce konteyner hazır hale gelerek yükünüzü omuzlar.
- Docker, uygulamalarınızı standart bir zemine oturatarak her platformda aynı şekilde çalışmasını sağlar.

Tomcat

Apache Tomcat veya Tomcat Java tabanlı web uygulamalarını yayınlamak için kullanılan web sunucusudur.

Apache Tomcat, etkili bir Java sunucu ortamını tanıtmak için Java Sunucu Uygulamalarının ve JavaServer Sayfalarının (JSP) uygulanmasına izin verir. Kullanıcılar ayrıca yapılandırma için kaynaklara erişebilir ve projeleri yapılandırmak için genişletilebilir biçimlendirme dili (XML) kullanabilirler.

Örneğin Java ile bir war dosyası oluşturduğunuzda bu war dosyasını Apache Tomcat'in içine yükleyebilir ve war dosyası olarak derlediğiniz Java uygulamanızın çalışmasını sağlayabilirsiniz.

Web server

Web server ya da ağ sunucusu, internet üzerinde bir web sitesinin yayınından sorumlu olan sunucudur. Web server, Hosting ya da "barındırma" işlemini internet protokolü üzerinden sunan bir sunucudur. Barındırma ya da hosting, Web sayfalarını internette yayınlamak için gerekli alanın kiralınmasıdır. Diğer bir ifade ile hosting, bir Web sitesinde yayınlanmak istenen sayfaların, resimlerin veya dokümanların internet kullanıcıları tarafından erişilebileceği bir bilgisayarda tutulmasıdır.

Web Sunucusu (*Web Server*), HTTP protokolünü kullanarak internet siteleri üzerinde yer almakta olan dosyaları kullanıcılara sunan, kullanıcıların her bir talep isteğini yanıtlayan ve bilgisayarların otomatik mekanizmaları tarafından yürütülen yazılımlardır. Web, sunucuları normal bir bilgisayarın sahip olduğu birçok donanıma sahiptir ancak ekran yerine komut sistemi üzerinden ve uzaktan yazılımlar yardımıyla yönetilirler. Paylaşımlı bilgisayarlar ve sadece web sunucusu yazılımını çalıştırmak için dizayn edilen bilgisayarlar da, web sunucusu olarak adlandırılabilirler.

Bir web sunucusu seçimi yaparken kullanıcıların değerlendirmesi gereken en önemli Web sunucusunun çalışmak için ihtiyaç duyduğu platform ortamı (Linux veya Windows, gibi)

- Sunucu taraflı programlama yetenekleri
- Güvenlik karakteristiği
- Çalışma performansı
- Kontrol edilebilirliği
- Bölünebilirliği

Application server

Uygulama **sunucusu** , platform ara katman yazılımının modern bir biçimidir. Bir tarafta işletim sistemi (OS), diğer tarafta harici kaynaklar (veritabanı yönetim sistemi [DBMS], iletişim ve İnternet hizmetleri gibi) ve üçüncü tarafta kullanıcıların uygulamaları arasında bulunan sistem yazılımıdır. Uygulama sunucusunun işlevi, iş uygulamasına erişimi ve iş uygulamasının performansını kolaylaştırırken, kullanıcının iş mantığı için ana bilgisayar (veya kapsayıcı) olarak hareket etmektir. Uygulama sunucusu, istemci isteklerinin değişken ve rekabet eden trafiğine, donanım ve yazılım hatalarına, daha büyük ölçekli uygulamaların dağıtık doğasına ve uygulamaların iş gereksinimlerini karşılamak için gereken veri ve işleme kaynaklarının potansiyel heterojenliğine rağmen performans göstermelidir.

Uygulama sunucusunun temel amacı aşağıda verilmiştir:

- İstemci programlarının boyutunu ve karmaşıklığını azaltmak için bir mekanizma.
- Daha iyi performans için veri akışını önbelleğe alma ve kontrol etme ihtiyacı için.
- Verilerin yanı sıra son kullanıcı trafiği için güvenliği uygulamaya yönelik bir mekanizma.