Pair Programming

Pair programming, bir yazılım geliştirme yöntemidir. Bu yöntemde iki yazılımcı tek bir kod bloğu üzerinde çalışır. Yazılımcılardan biri sürücü (driver) rolünü üstlenirken diğeri ise gözlemci (observer) rölünü üstlenir. Sürücü kodu yazar ve yön gösterici sürekli kodu takip eder.

Yararları

* İnsanlar hep aynı mantıkta düşünür farklı bir kişinin olaya farklı bir mantık ve açıdan bakması gözden kaçan detayların görülmesini sağlar.
* 2 yazılımcının aynı kod üzerinde çalışması daha temiz bir kod çıkmasını sağlar.
* İki kişinin tek bir kod üzerinde çalışması kodun daha hızlı bitmesini sağlar.
* Kodu birlikte yazan kişiler arasında bilgi alışverişi olur ve kendilerinin daha iyi programcı olmasını sağlar.
* Takım çalışması ve iletişimi artırır.

Zararları

* Çalışanlardan biri daha tecrübeli olduğu için efendilik ilan ettiği durumlarda diğer çalışanın motivasyonu düşer ve bu durum uzaklaşma ile sonuçlanabilir.
* İki kişi arasında sessizlik olabilir ve sessizlik oluyorsa birlikte çalışmadıklarını gösterir
* Bir kodu nasıl yazacakları konusunda fikir ayrışıklığı olabilir.

Virtualization

Virtualization (Sanallaştırma) tek bir bilgisayarının donanımının işlemcisini, ramini, depolaması ve çok daha fazlasını yazılım ile çok sayıda sanal bilgisayarlara böler. Genellikle biz bunlara Virtual Machines (Sanal Makineler VMs) deriz. Her bir VM kendi işletim sistemi altında çalışır ve sadece host (ana makine)nin altında çalışsa bile tamamen hostdan bağımsız bir bilgisayar olarak çalışır.

Yararları

* Kaynak verimliliği: VMs’den önce çoğu uygulama kendine ait bir fiziksel CPU’su olan bir server istiyordu şimdi ise VMs sayesinde her uygulama VM’i altında diğer VMs’e bulaşmadan kolaylıkla çalışabilir ve her VM için ayrı ve ona özel kaynak ayrılabilir.
* Yönetme Kolaylığı: Makinede başka bir VM’de çıkan sorun diğer VMs’i etkilemez ve VM’de bir sorun çıktığında önceden alınan bir image’a kolaylıkla dönülebilinir. Bu yarar aynı zamanda downtime (aksama süresi)ı yüksek derecede düşürebilir.

Containerization

Containerization (Kapsayıcılaştırma)ı Virtualization olarak düşünebiliriz fakat Virtualization’dan çok önemli farkları vardır. bunlardan en önemlisi Virtualization’da bütün işletim sistemi ve onun getirdiği ağır yükler var iken Containerization’da sadece uygulamayı çalıştırmak için gereken kütüphaneler ve diğer bağımlılıklar vardır herhangi bir ek yük yoktur.

Bir container imajı, çalıştırmak için gereken her şeyi içeren bir yazılım parçasının hafif, bağımsız, yürütülebilir bir paketidir. Kod, çalışma zamanı, sistem araçları, sistem kütüphaneleri, konfigurasyonlar bulundurur. Hem Linux hem de Windows tabanlı uygulamalar için mevcut olan konteynerli yazılımlar, çevreye bakılmaksızın her zaman aynı şekilde çalışır.

Containerization’da sadece uygulamayı çalıştırmak için gerekler olduğu için Virtualization’dan çok daha verimli ve hızlıdır.



