Test, ürünün beklenilen seviyede olduğunu belirlemek, ya da istenilen ölçüye gelmesini sağlamak için kullanılan, belirli birtakım kurallar dahilinde işletilen bir süreçtir. Yazılım dünyasında ise test, bir sistem ya da uygulamanın belirlenen koşullar altında çalıştırılabilmesini sağlamaya yarayan bir araçtır. Peki ya Test Yönetimi?

Test Yönetimi:

Geliştirilen yazılım projesinde öngörülen ya da öngörülemeyen hataları en aza indirmek için testleri belirleme, onları organize etme, farklı test teknolojileri kullanılarak test sonuçlarını toplama ve ilerlemeyi ölçme süreçlerinden oluşur. Tabi ki bir proje geliştirilirken her şeyden önce ilk dikkat edilmesi gereken “maliyet” olduğundan bu süreçte maliyeti en az indirecek şekilde plan yapmak gerekmektedir. Bir testin maliyeti de teste verilen efor miktarından kaynaklanmaktadır. Yapılan araştırmalar neticesinde;

* Sistem testi, projenin büyüklüğüne bağlı olarak toplam çabanın yüzde 16’sı ila 29'unu gerektirir.
* Testlerin belirlenmesi, projenin niteliğine, karmaşıklığına, kapsamlılığına ve tabi ki de sürece bağlı olarak değişiklik gösterir.

Testin başarılı olup olamayacağını belirlemek için Test Ölçümü yapmamız gerekmektedir.

Test Ölçümü:

Test durumları, bir ürünün kalitesini artırmaya yardımcı olmanın yanı sıra, test sonuçlarını içeren bir dizi işlemi ölçmek için kullanıldığından, geliştirme sürecini iyileştirmede değerlidir. Test Ölçümü yapmak için bazı formüller bulunmuştur:

* Defect Removal Efficiency (DRE), or test effectiveness, bir test sırasında bulunan hata sayısının toplam hata sayısına oranıdır.
* Pass rate, başarılı testlerin toplam test sayısına oranıdır.
* Passes versus failures, başarısız testlerin başarılı testlere oranıdır.

Tüm bu formüllerin ışığında zamana bağlı olarak, belirlenen, başarılı olan ve olmayan testler bir grafiğe dökülerek sonuçlar listelenmiş olur.

Bir test durumunun yararlı olması için bazı koşulların belirlenmesi gerekmektedir;

* Testin neyi amaçladığı ve önemi kısaca tanımlanır,
* Testin çalışması için bir ön koşul varsa bu tanımlanır.
* Hangi adımlarla uygulanması gerektiği belirlenir.
* Testin sonucunda ne olması beklendiği yazılır.
* Test sürecinin ne zaman ve hangi sıklıkta olacağı belirlenir.
* Yapılan test, öngörülenden fazla hata bulduysa bunların bir listesi çıkartılır.

Bu koşulların belirlenmesi basit gibi görünse de, yapılan testlerin bir tablo haline getirilmesi, kaydının tutulması oldukça önemlidir.

Açık ya da Kapalı kaynaklı olanların yanı sıra, Test Yönetimini kolaylaştırmaya yönelik oldukça fazla Test Aracı mevcuttur. Hangi aracın daha verimli olduğuna karar vermek için şu adımlar takip edilebilir;

* Online olarak birlikte çalışılabilirlik,
* Bir Test Senaryosu Yönetim Aracı, fonksiyonel gereksinimler, test senaryoları ve hatalar arasında üçlü bir bağlantı kurmalıdır. Bu şekilde, hangi test durumlarının hangi gereksinimlere karşılık geldiğini bilebilir, test koşullarına sahip olmayan gereksinimleri belirleyebilir veya belirli gereksinimlere yönelik hataları listeleyebiliriz.
* Kimlik doğrulama sistemleri ile birlikte çalışabilirlik. İdeal olarak, her bir proje katılımcısı tek bir kimliğe sahip olmalı ve bu kimliği her şey için kullanmalı, kodlama, hata bildirme, deneme testleri yapmalıdır.
* Maliyet ve lisans.

Bir aracın en önemli özelliği teknik kabiliyeti değil, ne kadar etkili kullanıldığı ve geliştirme sürecine ne kadar katkı sağladığıdır.