T.C.
İSTANBUL ÜNİVERSİTESİ-CERRAHPAŞA
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ

YÜKSEK LİSANS TEZİ

YAZILIM TEST OTOMASYON ARAÇLARININ KARŞILAŞTIRILMASI

Ekrem EROL

DANIŞMAN
Dr. Öğr. Üyesi Sibel SENAN

Bilgisayar Mühendisliği Anabilim Dalı

Bilgisayar Mühendisliği Programı

İSTANBUL-2019

Bu çalışma 12.06.2019 Tarihinde aşağıdaki jüri tarafından Bilgisayar Mühendisliği Anabilim Dalı, Bilgisayar Mühendisliği Tezli Yüksek Lisans Programı Yüksek Lisans Tezi olarak kabul edilmiştir.

TEZ JÜRİSİ

Dr.Öğr.Üyesi Sibel SENAN (Danışman)
İstanbul Üniversitesi-Cerrahpaşa
Fakülte

Doç. Dr. Eylem YÜCEL DEMİREL
İstanbul Üniversitesi-Cerrahpaşa
Mühendislik Fakültesi

Dr.Öğr.Üyesi Mohammed Ahmed SHAH
İstanbul Arel Üniversitesi
Mühendislik-Mimarlık Fakültesi

20.04.2016 tarihli Resmi Gazete'de yayımlanan Lisansüstü Eğitim ve Öğretim Yönetmeliğinin 9/2 ve 22/2 maddeleri gereğince; Bu Lisansüstü teze, İstanbul Üniversitesi-Cerrahpaşa'nın abonesi olduğu intihal yazılım programı kullanılarak Lisansüstü Eğitim Enstitüsü'nün belirlemiş olduğu ölçütlere uygun rapor alınmıştır.

ÖNSÖZ

Bu tez çalışmasında bazı test otomasyon araçları test edilerek, Türkiye yazılım sektöründe önemli bir yer edinmeye başlayan yazılım testi alanında test otomasyon aracı tercihlerine yardımcı olunması amaçlanmıştır.

Öncelikle bu tezi hazırlarken isteklerimi ve ihtiyaçlarımı göz önünde bulundurup bana yardımcı olan ve ihtiyaç duyduğum her zaman bana kapılarını açan değerli danışmanım Dr.Öğr.Üyesi Sibel SENAN'a ve lisanstan beri beni yüreklendiren ve destekleyen değerli hocam Doç.Dr. Rüya ŞAMLI'ya teşekkürlerimi sunarım. Bu zorlu tez sürecinde beni her daim manevi olarak destekleyen saygıdeğer çalışma ve okul arkadaşlarıma ve tüm eğitim hayatım boyunca benden maddi ve manevi desteklerini bir an olsun esirgemeyen, her durumda tercihlerime saygı duyan ve beni hedeflerim doğrultusunda cessaretlendirerek her zaman yanımda olan sevgili aileme teşekkürlerimi bir borç bilirim.

Haziran 2019 Ekrem EROL

İÇİNDEKİLER

Sayfa No

ÖNSÖZ ..... **iv**
İÇİNDEKİLER ..... **v**
ŞEKİL LİSTESİ ..... **vi**
TABLO LİSTESİ ..... **vii**
KISALTMA LİSTESİ ..... **x**
ÖZET ..... **x**
SUMMARY ..... **xiii**
1. GİRİŞ **1**
2. GENEL KISIMLAR **4**
2.1. TEST TÜRLERİ 6
2.1.1. Fonksiyonel Testler 6
2.1.2. Fonksiyonel Olmayan Testler 7
2.1.3. Beyaz Kutu Testi 7
2.1.4. Kara Kutu Testi 7
2.1.5. Manuel/Otomatik Testler 8
2.2 TEST SEVİYELERİ 10
2.2.1. Birim Testi 10
2.2.2. Entegrasyon Testi 10
2.2.3 Sistem Testi 10
2.2.4 Kabul Testi 10
3. MALZEME VE YÖNTEM **11**
4. BULGULAR **17**
5. TARTIŞMA VE SONUÇ **44**
KAYNAKLAR **45**
EKLER **48**
ÖZGEÇMİŞ **49**

ŞEKİL LİSTESİ

Sayfa No

Şekil 1: V-Model....4
Şekil 2: Bir Test Durumu (Test Case) Örneği....6
Şekil 3: Beyaz Kutu Testi'nin Gösterimi....7
Şekil 4: Kara Kutu Testi'nin Gösterimi....7
Şekil 5: Manuel ve Otomatik Test Süreçlerindeki Test Durum Sayısı ve Maliyet Oranı....9
Şekil 6: Örnek Windows 10 Görev Yöneticisi Performans Grafiği....18

TABLO LİSTESİ

Sayfa No

Tablo 1: Elle ve Otomasyonla Test'in Avantaj ve Dezavantajları 8
Tablo 2: Çalışmada Test Edilen Web Siteleri 11
Tablo 3: Test Edilen Sitelerde İzlenen Yollar 12
Tablo 4: Çalışmada Kullanılan Test Otomasyon Araçları.....13
Tablo 5: Çalışmada Kullanılan Cihazlar ve Teknik Özellikleri 14
Tablo 6: Çalışmada Kullanılan Tarayıcılar ve Sürümleri.....15
Tablo 7: Çalışmada Kullanılan Test Otomasyon Araçlarının Kod Gereksinimleri.....17
Tablo 8: Selenium WebDriver Test Aracının CPU ve RAM Kullanımı 19
Tablo 9: Katalon Studio Test Aracının CPU ve RAM Kullanımı 20
Tablo 10: Telerik Test Studio Test Aracının CPU ve RAM Kullanımı 21
Tablo 11: Ranorex Studio Test Aracının CPU ve RAM Kullanımı 22
Tablo 12: Squish Test Aracının CPU ve RAM Kullanımı 23
Tablo 13: TestingWhiz Test Aracının CPU ve RAM Kullanımı 24
Tablo 14: Çalışmada Kullanılan Araçların Ortalama CPU ve RAM Kullanımları 24
Tablo 15: Test Otomasyon Araçlarının Güncel Ücretlendirmeleri (Eylül-2018).....25
Tablo 16: Selenium WebDriver İçin Kullanım Değerlendirmesi.....26
Tablo 17: Selenium WebDriver İçin Kod Gereksinim Değerlendirmesi 27
Tablo 18: Selenium WebDriver İçin Raporlama Değerlendirmesi 27
Tablo 19: Selenium WebDriver İçin Fiyatlandırma Değerlendirmesi 28
Tablo 20: Selenium WebDriver İçin Destek Kanalları Değerlendirmesi.....28
Tablo 21: Selenium WebDriver İçin Dokümantasyon ve Eğitim Değerlendirmesi.....29
Tablo 22: Katalon Studio İçin Kullanım Değerlendirmesi.....29
Tablo 23: Katalon Studio İçin Kod Gereksinim Değerlendirmesi 30
Tablo 24: Katalon Studio İçin Raporlama Değerlendirmesi 30
Tablo 25: Katalon Studio İçin Fiyatlandırma Değerlendirmesi 31
Tablo 26: Katalon Studio İçin Destek Kanalları Değerlendirmesi.....31
Tablo 27: Katalon Studio İçin Dokümantasyon ve Eğitim Değerlendirmesi.....31
Tablo 28: Telerik Test Studio İçin Kullanım Değerlendirmesi.....32
Tablo 29: Telerik Test Studio Kod Gereksinim Değerlendirmesi.....32
Tablo 30: Telerik Test Studio İçin Raporlama Değerlendirmesi.....33

Tablo 31: Telerik Test Studio İçin Fiyatlandırma Değerlendirmesi.....33
Tablo 32: Telerik Test Studio İçin Destek Kanalları Değerlendirmesi 34
Tablo 33: Telerik Test Studio İçin Dokümantasyon ve Eğitim Değerlendirmesi 34
Tablo 34: Ranorex Studio İçin Kullanım Değerlendirmesi.....34
Tablo 35: Ranorex Studio İçin Kod Gereksinim Değerlendirmesi 35
Tablo 36: Ranorex Studio İçin Raporlama Değerlendirmesi 35
Tablo 37: Ranorex Studio İçin Fiyatlandırma Değerlendirmesi.....36
Tablo 38: Ranorex Studio İçin Destek Kanalları Değerlendirmesi 36
Tablo 39: Ranorex Studio İçin Dokümantasyon ve Eğitim Değerlendirmesi 37
Tablo 40: Squish İçin Kullanım Değerlendirmesi 37
Tablo 41: Squish İçin Kod Gereksinim Değerlendirmesi.....38
Tablo 42: Squish İçin Raporlama Değerlendirmesi.....38
Tablo 43: Squish İçin Fiyatlandırma Değerlendirmesi.....38
Tablo 44: Squish İçin Destek Kanalları Değerlendirmesi 39
Tablo 45: Squish İçin Dokümantasyon ve Eğitim Değerlendirmesi 39
Tablo 46: TestingWhiz İçin Kullanım Değerlendirmesi 40
Tablo 47: TestingWhiz İçin Kod Gereksinim Değerlendirmesi.....40
Tablo 48: TestingWhiz İçin Raporlama Değerlendirmesi.....41
Tablo 49: TestingWhiz İçin Fiyatlandırma Değerlendirmesi.....41
Tablo 50: TestingWhiz İçin Destek Kanalları Değerlendirmesi 41
Tablo 51: TestingWhiz İçin Dokümantasyon ve Eğitim Değerlendirmesi 42
Tablo 52: Kullanılan Araçların Genel Kullanım Değerlendirmesi.....42

KISALTMA LİSTESİ

Kısaltmalar	Açıklama
AUT	: Application Under Test – Test Edilen Uygulama
CPU	: Central Processing Unit – Merkezi İşlem Birimi, kısaca İşlemci
IDE	: Integrated Development Environment – Tümleşik Geliştirme Ortamı
IEEE	: The Institute of Electrical and Electronics Engineers – Elektrik Elektronik Mühendisleri Enstitüsü
ISTQB	: International Software Testing and Quality Board - Uluslararası Yazılım Test ve Kalite Kurulu
KKT	: Kullanıcı Kabul Testi – User Acceptance Test
OYT	: Otomatik Yazılım Testi – Automated Software Test
RAM	: Random Access Memory – Rastgele Erişimli Hafıza, kısaca Bellek
T.A.T.	: Test Automation Tools – Test Otomasyon Araçları

ÖZET

YAZILIM TEST OTOMASYON ARAÇLARININ KARŞILAŞTIRILMASI

YÜKSEK LİSANS TEZİ

Ekrem EROL

İstanbul Üniversitesi-Cerrahpaşa

Lisansüstü Eğitim Enstitüsü

Bilgisayar Mühendisliği Anabilim Dalı

Danışman : Dr. Öğr. Üyesi Sibel SENAN

Günümüzde, yaşamın her alanında bilgisayar yazılımları kullanılmaktadır. Kullanım amaçları veya alanlarına göre geliştirilmiş çok sayıda bilgisayar yazılımı bulunmaktadır. Bu nedenle, bilgisayar yazılımlarını güvenlik, hız, performans, maliyet, yük, kullanılabilirlik gibi farklı kriterler üzerinden test ederek değerlendirme gereği doğmuştur. Bu amaçla çeşitli yazılım test araçları üretilmiştir. Yazılım test araçlarının sayısının artması ile bilgisayar yazılımları arasındaki rekabet ve geliştirilen yazılımların kalitesine verilen önem de artmaktadır.

Yazılım testi, yazılım geliştirme yaşam döngüsündeki ana aşamalardan biri olarak kabul edilir. Yazılım geliştirme yaşam döngüsü içerisindeki test sürecine ayrılan zaman ve önem artmakla birlikte, hangi yazılım testi aracının hangi tür yazılımlarda daha etkin çalıştığını gösteren yeterli sayıda kaynak bulunmamaktadır. Bu tez çalışmasının amacı, bu kaynak açığına destek olacak şekilde farklı test otomasyon araçlarını, performans, maliyet, kullanılabilirlik gibi kategoriler altında incelemek ve karşılaştırmaktır. Bu amaçla, son zamanlarda en çok tercih edilen yazılım test otomasyon araçları, farklı donanım özelliklerine sahip bilgisayarlarda ve farklı web siteleri üzerinde denenerek analiz edilmiştir. Ele alınan yazılım test araçlarının aynı şartlar altındaki çalışma durumları incelenmiş ve

karşılaştırılmıştır. Bu sayede, kullanıcılara test otomasyon aracı tercihi yapmaları esnasında yardımcı olacak bir kaynak oluşturulması hedeflenmiştir.

Haziran 2019, 62 sayfa.

Anahtar kelimeler: Yazılım Testi, Yazılım Geliştirme Yaşam Döngüsü, Yazılım Test Otomasyon Araçları, Otomatik Yazılım Test Etme

SUMMARY

A COMPARISON OF SOFTWARE TEST AUTOMATION TOOLS

M.Sc. THESIS

Ekrem EROL

Istanbul University-Cerrahpasa

Institute of Graduate Studies

Department of Computer Engineering

Supervisor : Assist. Prof. Dr. Sibel SENAN

Nowadays, computer software are used in every area of life. There are a lot of software that have been developed for many purposes and areas. Therefore, it has become necessary to evaluate computer software by using different criteria such as security, speed, performance, cost, load, usability, etc. For this purpose, various software testing tools have been produced. The increasing number of software testing tools increases the competition between computer software and the importance of the quality of developed software.

Software testing is considered to be one of the main stages in the software development life cycle. While the time and importance of testing in the software development lifecycle increases, there is not a sufficient number of resources to show which software testing tool is more efficient in which software. The purpose of this thesis is to examine and compare different test automation tools by categories such as performance, cost, usability, to support this resource gap. For this purpose, recently the most preferred software test automation tools have been analyzed by experimenting on computers with different hardware specifications and on different websites. The working conditions of the software testing tools under the

same conditions were examined and compared. In this way, it is aimed to create a resource that will help the users to choose the test automation tool.

June 2019, 62 pages.

Keywords: Software Testing, Software Development Life Cycle, Software Test Automation Tools, Automated Software Testing

1. GİRİŞ

Bilgisayar yazılımları günlük yaşamın her alanı için gelişmeye ve yayılmaya devam etmektedir. Yazılımın günlük yaşam rutinindeki etkileri çok önemlidir, ancak bu etkiler artık hissedilmeyecek kadar içselleşmiştir. Örneğin; insanlar genellikle bir ATM'den para çekerken sadece çalışmasını beklerler, bu esnada arka planda çalışan sistemi düşünmezler. Bu örnekte olduğu gibi, yazılım bugün insan yaşamına görünmez bir temel oluşturmaktadır ve genellikle insanlar sadece çalışmayı bıraktığında yazılımın farkına varmaktadır.

Bilgisayar yazılımlarının kullanım alanlarının artmasıyla birlikte yazılım kalitesinin önemine dair farkındalık da artmıştır. Yazılım testi, yazılım mühendisliğinin teslim edilen ürünün kalitesiyle ilgili bir parçasıdır. Test, yazılım ürünlerinin doğrulanmasında, onaylanmasında ve güvenilirliğinin tahmininde yazılım kalitesini sağlamak için kullanılır. Çünkü, yazılım sistemlerinin kalitesi başarısızlıkla sonuçlandığında feci sonuçlar doğabilir. Tüm bu hususlar neticesinde, 1970'lerin sonlarından itibaren yazılım testi sürecine ayrılan kişi ve zaman miktarı çoğaltılmıştır. Yazılım ürünlerinin, her geçen gün artan yaşam gereksinimlerine bağlı olarak günlük yaşamdan, kritik görev sistemlerine kadar geniş uygulanma alanlarına yayılması nedeni ile, yazılım testi süreci şirketler, kuruluşlar, araştırmacılar için en zor ve kaçınılmaz konulardan biri haline gelmiştir [1]. Yazılım testi ile ilgili göz önünde bulundurulması gereken birçok husus vardır. Seçilen test tekniklerinin belirli bir yazılım için doğru olup olmadığı, hangi test tiplerinin / yöntemlerinin kullanılması gerektiği, yazılım gereksinimlerinin neler olduğu, doğrulama ve onaylama için hangi tekniğin kullanılması gerektiği, kod kapsam testinin nasıl yapılacağı, test işlemlerinin belgelenmesi vb., yazılım testi için gerekli temel hususlardır.

Yazılım testleri manuel veya otomatik olarak yapılabilir. Manuel Yazılım Testinde süreç, insan girişi, analiz ve değerlendirme gerektirir. Manuel testler insan müdahalesiyle ilgilidir ve doğal olarak hatalara açıktır. Çünkü çoğu zaman insanlar test işlemini tekrar tekrar yapmaktan yorulur. Manuel test sürecinde, yazılımın doğruluğunu kontrol etmek için test senaryoları yazmak gerekmektedir ve bu işlem çok zaman ve sabır gerektirir. Otomatik Yazılım Testi ise, test senaryolarının geliştirilmesini, test senaryolarının yürütülmesini, doğrulanmasını ve otomatik araçların kullanımını içeren, test faaliyetlerinin otomatikleştirildiği bir işlemdir.

Otomatik Yazılım Testinde, tüm testler otomatik olabileceği gibi sadece bazı test durumları da otomatikleştirilebilir. Yazılım şirketlerinin çoğu, kalite iyileştirme, pazara çıkma süresi ve daha az insan çabası gibi faydalar elde etmek için otomatik yazılım testi kullanmaktadır. Verimli otomatik yazılım testinin elde edilmesi, testlerin daha kısa sürede ve daha az çabayla gerçekleştirilmesine bağlıdır. Bu da kullanılacak "yazılım otomasyon test aracı"nın seçimi ile ilişkilidir. Otomatik test sürecinde kullanılmak üzere geliştirilen "yazılım otomasyon test araçları" kendi aralarında kullanım alanlarına göre farklı özellikler göstermektedir. Örneğin; bir kısmı mobil uygulamalar için kullanılırken, bir kısmı web sitelerinin test edilmesi için kullanılmaktadır. Ayrıca, bu test otomasyon araçları kullanılan programlama diline ve test yöntemlerine bağlı olarak da farklılaşmaktadır.

Yıllar boyunca geliştirilen birçok otomatik test aracı olmuştur. Ancak bu durum kullanıcıların hangi test aracını seçmeleri gerektiğine karar vermelerini zorlaştırmaktadır. Literatürde, otomatik test araçlarının incelendiği ve değerlendirildiği çeşitli çalışmalar bulunmaktadır [2]-[13]. Otomatik Yazılım Test Etme araçlarının kategorize edilmesine ilişkin olarak, Sergey Uspenskiy [9] tarafından yayınlanan çalışmada test araçlarını sınıflandırmak için otomatik bir test modeli öne sürülmüştür. Berner [8], yazılım hatalarının büyük çoğunluğunun manuel yazılım testi sürecinde tespit edildiğini ortaya koymuştur. Bunun nedeninin, yazılım buglarının %60'ının otomatik test araçları ile, %80'inin ise yazılım testi geliştirilirken bulunduğu olarak belirtmiştir. Ayrıca bazı çalışmalarda; otomatik yazılım test araçlarının yararları ve zorlukları, bu testlerin tutarlılığı ve tekrarlanabilirliği, testlerin tekrar kullanılabilirliği, otomatik test araçlarından beklentilerin neler olduğu, ekonomik pazardaki yeri gibi konulara yer verilmiştir [2], [10]-[13]. Bu makaleler otomatik yazılım test araçları konusunda çok değerli olsa da, bu araçların test seviyeleri, kullanılan arayüzler, performans analizleri, kullanım kolaylıkları, kod gereksinimleri gibi yönlerinin incelenerek değerlendirilmesi hususlarını kapsamamaktadır ve bu araştırma boşluğunun doldurulmasına ihtiyaç duyulmaktadır.

Bu tez çalışmasının amacı; bu boşluğun doldurulması ve bu sayede doğru otomatik test araçlarının seçimi sırasında kişilerin ve kurumların tercih yapmasını kolaylaştıracak bir kaynak oluşturulmasıdır. Bu amaç ile, piyasadaki kullanıcı görüşlerine göre en çok tercih edilen test otomasyon araçları ele alınmıştır. Belirlenen test otomasyon araçları, web sitelerini test etmede kullanılan araçlardır.

Bu tezin içeriği aşağıdaki şekilde organize edilmiştir:

Tezin Giriş bölümünde, yazılım testlerinin önemi ve Manuel/Otomatik Yazılım Testleri ile ilgili bir açıklama sunulmaktadır. Tezin temel amacı ve akışı burada yer almaktadır.

Tezin Genel Kısımlar bölümünde, Otomatik Yazılım Testi kavramı ve yazılım testinin otomasyonuna olan ihtiyacı anlamak için gerekli temel bilgilere yer verilecektir. Ayrıca, test otomasyon araçlarına ilişkin ilgili çalışmalardan bahsedilecektir. Daha sonra, tez çalışmasında sunulacak olan mevcut araştırma açığının ayrıntılı bir açıklaması verilecektir.

Tezin Malzeme ve Yöntem bölümünde, kullanılan test otomasyon araçlarının ayrıntılarına değinilecektir. Tez çalışmasında kullanılan platformlar, belirlenen web siteleri, test sürecinde izlenen yollar ve kullanılan donanımların özellikleri bu bölümde verilecektir.

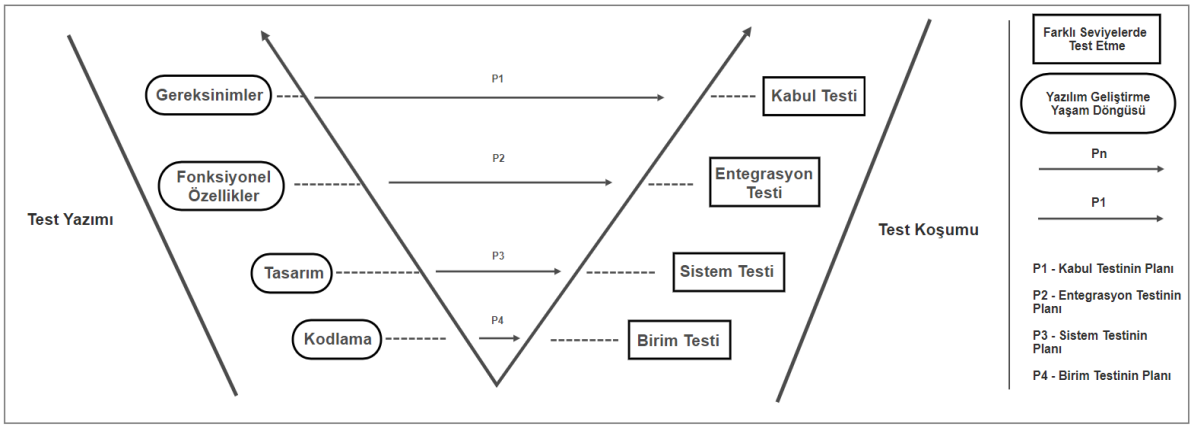
Tezin Bulgular bölümünde, incelenen test otomasyon araçlarının değerlendirilmesi ve maliyet, kullanım kolaylığı, performans, kod gereksinimi gibi kategoriler üzerinden analizleri sunulacaktır.

Tartışma ve Sonuç bölümünde ise, tez çalışması neticesinde elde edilen bulgulara dayanarak çıkarılan sonuçlar sunulacaktır.

2. GENEL KISIMLAR

Bilgisayar yazılımları, sosyal hayatımızda önemli bir yere sahiptir. Bu nedenle, yazılım geliştiricilerin ürünlerini kalite açısından test etmeleri gereklidir [1]. Kalite bir sorun haline geldiğinde, yazılım testleri, ürün kalitesini arttırmak için kullanılan bir araç haline gelmektedir [2]. Yazılım testi, istenilen sonuçlara ulaştığını belirlemek için bir ürünü, programı veya sistemi değerlendiren bir tekniktir [4]. Özetle test, hata bulma amacıyla program yürütme işlemidir [4].

Yazılım geliştirme yaşam döngüsünde, yazılım testi önemli bir aşamadır; ancak testin yazılım geliştirme yaşam döngüsünde geç olduğu düşünüldüğünde problemler ortaya çıkmaktadır. John Watkins [6], her geliştirme aktivitesinin alternatif bir test aktivitesine sahip olduğu bir 'V-Modeli' önermiştir. Şekil 1, geliştirmenin dört aşamasını ve karşılık gelen test aktivitesini gösterir.



Şekil 1: V-Model [6].

Yazılım testi, yazılım kalite güvencesinin kritik bir unsurudur ve tasarım ve kodlamanın nihai incelemesini temsil eder. Yazılım testi, yazılımın işlevselliğini ve doğruluğunu çalıştırarak test etme işlemidir. Yazılım testi genellikle iki nedenden ötürü gerçekleştirilir: Hata tespiti ve güvenilirlik tahmini [23].

Yazılım Testi Yaşam Döngüsü, kalite hedeflerine ulaşıldığından emin olmak için belirli bir sırada gerçekleştirilmesi gereken belirli adımlara sahip bir test sürecini ifade eder. Yazılım Testi Yaşam döngüsü içerisinde, her aktivite planlı ve sistematik bir şekilde gerçekleştirilir.

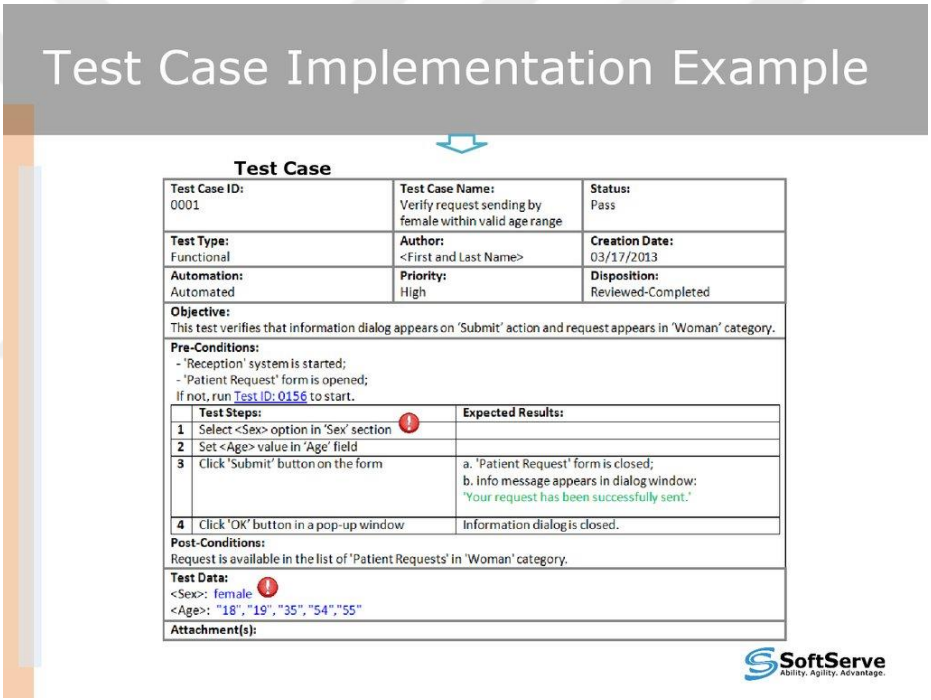
Her aşamada farklı hedefler ve teslim edilebilir çıktılar vardır. Test yaşam döngüsünde farklı organizasyonların farklı aşamaları vardır; ancak temel aşamalar aynı kalır [24].

Yazılım Testi Yaşam Döngüsü aşamaları şu şekilde verilebilir:

1. Gereksinimler (Requirements) Aşaması: Test gereksinimlerinin analiz edilerek incelendiği aşamadır.
2. Planlama (Planning) Aşaması: Pratikte test sürecinin ilk aşamasıdır. Test stratejisi ve risk analizi/yönetimi bu aşamada gerçekleştirilir.
3. Analiz (Analysis) Aşaması: "Ne test edilecek?" sorusunun cevabının verildiği aşamadır. Ürün karmaşıklığı, ürün ve projenin riskleri, test yönetimi, takımın becerileri ve bilgileri bu aşamada analiz edilerek izlenebilir olması sağlanır. Ayrıca test koşulları ve testin çıkış kriterleri bu aşamada belirlenmelidir.
4. Tasarım (Design) Aşaması: "Nasıl test edilecek?" sorusunun cevabı, bu aşamada verilmektedir. Test koşullarının detaylandırılması, test verilerinin ve test ortamının tanımlanması bu aşamada gerçekleştirilmelidir.
5. Uygulama (Implementation) Aşaması: Test yaşam döngüsü içindeki test durumlarının (test cases) oluşturulduğu aşama, Uygulama Aşaması'dır. Bu aşamada test senaryolarının önceliklendirmesi yapılmalıdır.
6. Yürütme (Execution) Aşaması: Test durumlarındaki giriş kriterlerinin kontrol edilerek test durumlarının koşulduğu aşamadır. Test sürecinin nihai eyleme dönüştüğü aşamada, Yürütme Aşaması'dır. Her durumun, test koşulmasıyla, kontrol edilmesinin ardından özellikle olumsuzlukların ve tutarsızlıkların kayıt altına alınması (log) gerekmektedir.
7. Sonuç (Conclusion) Aşaması: Test yaşam döngüsü içerisinde koşulan testlerin çıkış kriterlerine ve raporlamaya ağırlık verildiği aşama, Sonuç Aşaması'dır. Bu aşamada projeye ve paydaşların tercihine göre Günlük Durum Raporu (Daily Status Report – DSR) veya Haftalık Durum Raporu (Weekly Status Report - WSR) hazırlanabilir. Hazırlanan raporun içeriği (teknik detaylar, okunabilirlik vs.) raporun sunulacağı kişi veya kişilere bağlıdır. Proje yönetici veya takım liderine, teknik detayların yoğun olduğu bir rapor sunulurken; üst düzey yönetici ve paydaşlar teknik kısımları es geçerek girdi ve çıktılara odaklanabilmektedir.

8. Kapama (Closure) Aşaması: Test yaşam döngüsü, genel olarak tüm yazılım geliştirme süreci boyunca devam etse de, test durumlarının koşulması bir noktadan sonra son bulur. Bu, genelde test durumlarının tamamının kontrol edilip raporlanmasıyla tamamlanır. Kapama Aşaması'nda test durumlarının tamamlanıp tamamlanmadığı kontrol edilerek test yaşam döngüsü sonlandırılır veya test koşulma eylemi indirgenir.

Yazılım testi koşulurken "Test Durumları"ndan yararlanılır. Test Durumları, test koşullarına dayalı olarak geliştirilen bir dizi önkoşul, girdi, eylem (uygun olduğunda), beklenen sonuçlar ve son koşulları içeren dokümanlardır [15]. Şekil 2.de bir "Test Durumu" örneği gösterilmektedir.



Şekil 2: Bir Test Durumu Örneği [25].

2.1. TEST TÜRLERİ

Farklı özelliklere bağlı olarak ayrılmış çeşitli test türleri vardır.

2.1.1 Fonksiyonel Testler

Fonksiyonel uygulamasının her bir işlevinin gereksinim belirtimine uygun çalıştığını doğrulayan bir tür testtir [27]. Fonksiyonel testlere örnek olarak, kullanıcı kabul testi (KKT) (User Acceptance Test – UAT) verilebilir. Kullanıcı kabul testi, ilgili sistemin ürün olarak sunulmadan önce yapılan son testidir ve sistemin müşterinin ihtiyaçları doğrultusunda çalışıp

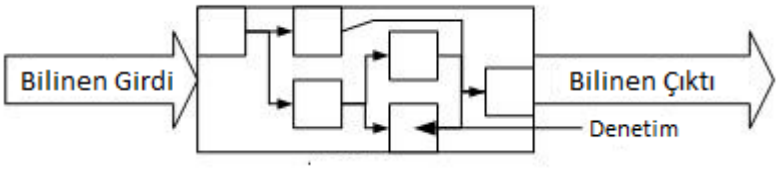
çalışmadığını kontrol etmeye yarar [29]. KKT sürecini genişleten Beta Testi (Beta Testing) bulunur. Bir yazılımın Beta Testi, yazılım uygulamasının gerçek kullanıcıları tarafından gerçek ortamda gerçekleştirilir ve harici bir Kullanıcı Kabul Testi olarak kabul edilebilir [30]. Burada gerçek kullanıcı tanımı, son kullanıcı ve yazılımın hedef kitlesi anlamını taşırken; gerçek ortam ise son kullanıcının ilgili yazılımı kullandığı ortam ve ortamları ifade etmektedir.

2.1.2. Fonksiyonel Olmayan Testler

Fonksiyonel olmayan test, bir yazılım uygulamasının işlevsel olmayan yönlerini (performans, kullanılabilirlik, güvenilirlik vb.) kontrol etmek için yapılan bir test türüdür [27]. Fonksiyonel olmayan testlere örnek olarak, performans testi verilebilir. Performans Testi, yazılım uygulamalarının beklenen iş yükü altında iyi performans göstermelerini sağlamak için bir tür yazılım testi olarak tanımlanır [28].

2.1.3. Beyaz Kutu Testi

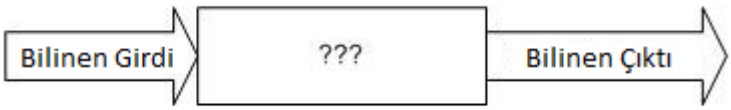
Beyaz Kutu Testi ayrıca yapısal test veya cam kutu testi olarak da bilinir. Beyaz Kutu Testinde yazılım mühendisi, yazılımın iç yapısı ile ilgili bilgileri kullanarak test durumlarını türetir. Şekil 3.te Beyaz Kutu testinin gösterimi verilmiştir.



Şekil 3: Beyaz Kutu Testi'nin Gösterimi.

2.1.4. Kara Kutu Testi

Kara Kutu Testi'nde yazılım mühendisi, yazılımın harici kısmını (özellik/arayüz) iç kısım olarak görmektedir. Genellikle, prosedür ve fonksiyon başlıkları gibi program arayüzünün özelliklerine dayanır. Ayrıca program girişini ve beklenen program çıkışını belirtmesi gerekir [3]. Şekil 4.te Kara Kutu testinin gösterimi verilmiştir.



Şekil 4: Kara Kutu Testi'nin Gösterimi.

2.1.4. Manuel/Otomatik Testler

Yazılım testi, testin koşulma yöntemine göre temelde ikiye ayrılır: Manuel ve Otomatik Testler. Manual Testler, test durumlarının elle (bir test elemanı veya test mühendisi tarafından) koşulması yöntemiyken; Otomatik Testler, test durumlarının bir otomasyon aracı ile koşulması yöntemidir.

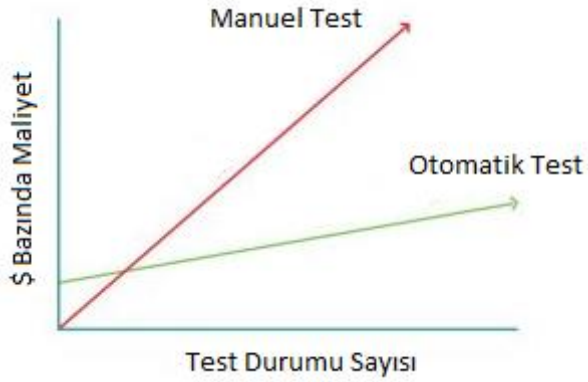
Otomatik Yazılım Testi, günümüzün gerçek zaman sınırlarıyla yüzleşen yazılım uzmanları için bir meydan okumadır ve bir proje üzerinde otomatik testler sunması, yönetmesi ve gerçekleştirmesi gereken zorlukları ele almaktadır [31].

Her iki yöntemin de kendi içerisinde avantaj ve dezavantajları mevcuttur.

Tablo 1: Manuel ve Otomasyonla Test'in Avantaj ve Dezavantajları.

	Manuel Test	**Otomasyonla Test**
Avantajlar	Test elemanın cihaz/yazılım kullanım tecrübesi işleri kolaylaştırabilir ve kullanıcı arayüzü daha güvenli kontrol edilebilir.	Birden fazla test durumunun aynı anda koşularak, özellikle tekrarlı test durumlarının daha hızlı koşulmasını sağlar.
Dezavantajlar	Test durumlarının tekrarlı bir şekilde koşulması ve yazılımdaki değişikliklerin sık aralıklarla yapılması durumlarında manuel test zaman ve maliyet açısından olumsuz sonuçlar doğurabilir.	Araç ve sistem kullanımı genellikle daha masraflıdır ve otomasyondaki bir problemi çözebilmek için anında elle müdahale gerekebilir.

Otomatik yazılım testi, yazılım mühendisliği alanında hayati bir rol oynar. Bu süreç, bir kuruluştaki standartlara göre yazılım kalitesini artırmak için mükemmel bir fikirdir. Ayrıca ürünün verimliliğini de arttırmaktadır. Yazılım test otomasyonunun, karmaşık yazılım testlerinin karmaşık alanındaki en güçlü silah olduğu kanıtlanmıştır [23]. Şekil 5.te Manuel ve Otomatik Test Süreçlerindeki Test Durum Sayısı ve Maliyet Oranı verilmiştir.



Şekil 5: Manuel ve Otomatik Test Süreçlerindeki Test Durum Sayısı ve Maliyet Oranı [16].

Fewster [2], Kaner ve ark. [17], Rice [18] ve Pettichord [8]'a göre; Otomatik Yazılım Testinin sık görülen avantajları ve zorlukları şu şekildedir:

Avantajları:

- Daha fazla ve daha sık test koşmak.
- Manuel olarak yapılması zor veya imkânsız olan testleri yapmak.
- Kaynakların daha iyi kullanılması.
- Testlerin tutarlılığı ve tekrarlanabilirliği.
- Testlerin tekrar kullanımı.
- Daha erken pazara çıkış.
- Artan güven.

Zorlukları:

- Gerçekçi olmayan beklentiler: Yöneticiler, Otomatik Yazılım Testinin (OYT) sorunlarını çözebileceğine ve kaliteyi artırabileceğine inanabilir.
- Kötü/zayıf test uygulaması.
- Otomatize edilmiş testlerin birçok yeni kusur bulacağı beklentisi.
- Yanlış güvenlik duygusu.
- Bakım.
- Teknik problemler.
- Örgütsel sorunlar.

2.1. TEST SEVİYELERİ

Genel olarak, en yaygın test türleri [1] [2], Birim Testi, Entegrasyon Testi, Sistem Testi ve Kabul Testidir.

2.2.1. Birim Testi

Birim testi, yazılımdaki hataları bulmak için temel seviye testidir. Birim testinde, yazılımdaki hataları bulmak için yazılım küçük birimlere bölünmektedir [19].

2.2.2. Entegrasyon Testi

Entegrasyon testinin amacı, yazılım modüllerinin doğru ve istikrarlı bir şekilde birlikte etkileşimde bulunmalarını belirlemektir [19].

2.2.1. Sistem Testi

Bu aşamada, geliştiriciler sistemin işlevselliğinin ve dengesinin yanısıra performans ve güvenilirlik gibi işlevsel olmayan gereksinimlerini test etmektedir [19].

2.2.1. Kabul Testi

Sistem testinde bulunan hatalar düzeltilir ve sistem müşteri tarafından kabul için test edilir. Kabul testinde müşterinin test edilip edilmediği test edilir. Kabul testinde müşteri, sistemin gereksinimleri karşılayıp karşılamadığını ve doğru çalışıp çalışmadığını test etmektedir [19].

3. MALZEME VE YÖNTEM

Test otomasyon araçları; gerektiğinde yardımcı yazılımlarla (örneğin bir IDE – Integrated Development Environment (Tümleşik Geliştirme Ortamı) birlikte kullanılarak veya tek başına bir arayüz kullanılarak test durumlarının koşulmasını sağlayan yazılımlardır. Bu çalışma kapsamında test edilen uygulamalar, çalışma sırasında İnternet üzerinde aktif olan web siteleridir. Farklı amaçlar için kullanılan web siteleri önceden belirlenen test otomasyon araçları kullanılarak, yine önceden belirlenen yollar izlenecek şekilde test edilmiş ve sonuçları değerlendirilmiştir.

Belirlenen web siteleri Tablo 2.de verilmiştir. Test senaryoları Tablo 3.te verildiği gibi oluşturulmuş ve sitenin hizmet ettiği amaca uygun olan bu senaryolar üzerinden test otomasyon araçları çalıştırılmıştır. Seçilen test otomasyon araçları Tablo 4.te verilmiştir.

Web sitelerinin belirlenmesinde öncelikli bir kriter belirlenmemiş olup; metinsel veya görsel içeriklerin yoğun olduğu siteler tercih edilmiştir.

Tablo 2: Çalışmada Test Edilen Web Siteleri.

Test Edilen Web Sitesi	Web Sitesi Türü	Web Adresi
TRT World	Haber	https://www.trtworld.com/
BBC World	Haber	https://www.bbc.com/
The New York Times	Haber	https://www.nytimes.com/
Tom’s Hardware	Forum	https://forums.tomshardware.com/
Linus Tech Tips	Forum	https://linustechtips.com/main/
xda Developers	Forum	https://forum.xda-developers.com/
Twitter	Sosyal Medya	https://twitter.com
Deviantart	Görsel Paylaşım	https://www.deviantart.com/
YouTube	Video Paylaşım	https://www.youtube.com/
Amazon	Alışveriş	https://www.amazon.com/
HepsiBurada	Alışveriş	https://www.hepsiburada.com/

Tablo 3: Test Edilen Sitelerde İzlenen Yollar.

Test Edilen Web Sitesi	**İzlenen Yol**
TRT World	Siteye gir > Opinion linkine tıkla > Must See linkine tıkla > Doğru sayfa olup olmadığını kontrol et
BBC World	Siteye gir > Weather linkine tıkla > Arama çubuğunda İstanbul'u ara > Doğru sonucun bulunup bulunmadığını kontrol et
The New York Times	Siteye gir > Sports linkine tıkla > Tennis linkine tıkla > Doğru sayfa olup olmadığını kontrol et
Tom's Hardware	Siteye gir > Mevcut hesapla giriş yap > Ana sayfaya dön > Ana sayfada, doğru kullanıcı ile giriş yapılıp yapılmadığını kontrol et
Linus Tech Tips	Siteye gir > Mevcut hesapla giriş yap > Ana sayfaya dön > Ana sayfada, doğru kullanıcı ile giriş yapılıp yapılmadığını kontrol et
xda Developers	Siteye gir > Mevcut hesapla giriş yap > Ana sayfaya dön > Ana sayfada, doğru kullanıcı ile giriş yapılıp yapılmadığını kontrol et
Twitter	Siteye gir > Mevcut hesapla giriş yap > Belirli bir hashtag kullanarak Tweet yaz > Ana sayfayı güncelle > Hashtag'i bul ve tıkla > İlgili hashtag ile ilgili sayfanın açılıp açılmadığını kontrol et
Deviantart	Siteye gir > Belirli bir kelime ile görsel ara > Açılan arama sonucu sayfasındaki üçüncü görseli aç ve cihaza kaydet > İlgili görselin kaydedilip kaydedilmediğini manuel olarak kontrol et
YouTube	Siteye gir > Belirli bir videoyu ifade eden arama yap > İlgili videoyu aç > Videoyu durdur > Doğru videonun açıldığını ve durdurulduğunu kontrol et
Amazon	Siteye gir > Belirli bir ürünü ara > İlgili ürünün sayfasına gir > Ürünü istek listesine ekle > İstek listesini aç > İlgili ürünün istek listesinde bulunup bulunmadığını kontrol et
HepsiBurada	Siteye gir > Belirli bir ürünü ara > İlgili ürünün sayfasına gir > Ürünü alışveriş sepetine ekle > Alışveriş sepetini aç > İlgili ürünün alışveriş sepetinde bulunup bulunmadığını kontrol et

Kullanılacak olan test otomasyon araçları belirlenmesi, en çok tercih edilen test otomasyon araçları arasından seçilerek yapılmıştır.[33][34][35][36] Bu bağlamda toplam 6(altı) tane test otomasyon aracı seçilmiş ve uygulama sırasında en son sürümleri kullanılarak sürece başlanmıştır.

Tablo 4: Çalışmada Kullanılan Test Otomasyon Araçları.

Kullanılan Test Otomasyon Aracı	Versiyon/Sürüm
Selenium WebDriver (IntelliJ Java IDE ile birlikte)	WebDriver 3.14.0
Katalon Studio	5.7.1
Telerik Test Studio	2018.2.606.0
Ranorex Studio	8.2.1
Squish	IDE 6.4.1
TestingWhiz	6.1.1: Saturn

Selenium, farklı tarayıcı ve platformlardaki web uygulamaları için, farklı yazılım araçları içeren ücretsiz (açık kaynaklı) bir otomasyon test paketidir. Bu araçlardan biri de Selenium-WebDriver'dır. Selenium-WebDriver, belli tarayıcıları ve tarayıcıların uyumlu olduğu işletim sistemleriyle birlikte birçok tarayıcıyı da destekler [37] [38]. Selenium Java, C#, Perl, Python, JavaScript, Ruby, PHP gibi birçok dili desteklemekle birlikte, birçok web tarayıcısını da desteklemektedir.

Bu çalışmada Selenium WebDriver ile birlikte IntelliJ IDEA kullanılmıştır. IntelliJ IDEA, bilgisayar yazılımı geliştirmek için Java ile entegre bir geliştirme ortamıdır (IDE). JetBrains (eski adıyla IntelliJ olarak bilinir) tarafından geliştirilmiştir ve Apache 2 Lisanslı bir topluluk baskısı olarak ve tescilli bir ticari sürümde mevcuttur [39].

Katalon Studio; API, Web ve mobil uygulamaları test etmek için güçlü ve kapsamlı bir otomasyon çözümüdür. Bu tür testler için ayarlanmış zengin bir özelliğe sahiptir ve Windows, macOS ve Linux dahil olmak üzere birçok platformda çalışabilir [30]. Katalon Studio Java/Groovy ve JavaScript dillerini desteklemektedir.

Telerik Test Studio, Telerik tarafından geliştirilen en yeni otomasyon aracıdır. HTML5, Angular, AJAX, JavaScript, Silverlight, WPF, MVC, Ruby ve PHP, iOS, Android gibi uygulamaların otomasyon testlerinin koşulmasını sağlar. C# ve VB.Net dillerini desteklemektedir [33].

Ranorex Studio; web tabanlı, masaüstü ve mobil uygulamaların testi için kullanılan bir GUI testi otomasyon aracıdır. Ranorex Studio'nun uygulamayı otomatize etmesi için kendi dili yoktur; VB.NET ve C # gibi standart programlama dilleri kullanır. Ranorex Studio herhangi

bir masaüstü, web tabanlı uygulamayı veya mobil yazılımı otomatize edebilir. Silverlight, .NET, Winforms, Java, SAP, WPF, HTML5, Flash, Flex, Windows Uygulamaları (Yerel / Karma) ve iOS, Android gibi birçok teknolojiyi destekler [40].

Squish; birden fazla platformu aynı anda destekleyen bir otomasyon test aracıdır. Tüm ana GUI teknolojileri için derinlemesine destek sağlar ve tüm masaüstü, mobil, web ve gömülü platformlar için tam destek sunar [34]. JavaScript, Perl, Python, Ruby gibi temel script dillerini desteklemektedir.

TestingWhiz; yazılım, web, mobil, veri tabanı, bulut, web servisleri ve API testlerini sağlayabilen Kodsuz Otomasyon Test Aracıdır. TestingWhiz web uygulamalarının kritik işleyişini doğrulamak için kullanıcı arayüzünü ve işlevselliğin otomatikleştirilmesin olanak sağlamaktadır [41].

Tüm bu test otomasyon araçlarının kurulumu, Tablo 5.te belirtilen cihazlara yapılmıştır. Kullanılan cihazlar, tüm araçların minimum sistem gereksinimlerini karşılamaktadır. Test ortamı farklı donanım özelliklerine sahip iki bilgisayarda gerçeklenmiştir. Her bilgisayarda iki tarayıcı üzerinden test edilmiştir. Bu tarayıcılar Tablo 6.da verilmiştir. Bu bilgisayarlar çalışma içerisinde PC-1 ve PC-2 olarak adlandırılmıştır.

Tablo 5: Çalışmada Kullanılan Cihazlar ve Teknik Özellikleri.

	PC-1	**PC-2**
İşlemci (CPU)	Intel Core i5 - 4590 (3.30 GHz)	Intel Core m3 7Y30 (1.00 GHz)
Bellek (RAM)	16GB DDR3	4GB LPDDR3
Disk	SSD (Tarayıcılar) HDD (Test Otomasyon Araçları)	SSD (Tarayıcılar) SSD (Test Otomasyon Araçları)
Çözünürlük	1920x1080	1920x1280
İşletim Sistemi	Windows 10 Education	Windows 10 Home

Intel Core i5, çift çekirdekli veya dört çekirdekli olarak kullanılabilen bir bilgisayar işlemcisidir. Hem masaüstü hem de dizüstü bilgisayarlarda kullanılabilir (bu çalışma için kullanılan bilgisayar türü, masaüstü bilgisayardır) ve "i" serisindeki üç işlemciden biridir. Core i5 işlemciler sıklıkla dört çekirdekli olarak bulunur. Core i5 işlemciler genellikle

masaüstü bilgisayarlarda günlük kullanımda ve oyun oynama gibi bazı yüksek performans gerektiren eylemlerde kullanılabilir [42].

Intel Core m3; ince ve hafif 2'si 1 arada laptoplarda ve diğer laptoplarda hız, güç ve 4K HD grafikler sunabilen işlemci serisidir. Cihaz üzerinde dokunma, ses ve dijital kalemle etkileşime olanak sağlar. Düşük güç tüketimi sayesinde kullanıcılara uzun pil ömrü vaat etmektedir [43].

Her iki bilgisayarda da aynı web tarayıcı sürümleri kullanılmıştır ve internet bağlantısı -aynı anda olmamak şartıyla- aynı ağ üzerinden sağlanmıştır.

Firefox web tarayıcısı ilk olarak 23 Eylül 2002'de beta sürümünde "Mozilla Tarayıcısı" olarak yayınlanmıştır, ancak dahili olarak "Phoenix" olarak adlandırılmıştır. Firefox, sekme başına bireysel işlemlerle sekmeli tarama, açılır pencere engelleme, özel tarama, kimlik avı algılama ve sanal alan güvenlik modeli gibi özelliklere sahiptir [44]. Linux, Mac, Windows, avuçiçi cihazlar ve 70'den fazla dilde kullanılabilir [45].

Chrome, Google tarafından 11 Aralık 2008'de resmen yayınlanmış ücretsiz bir İnternet tarayıcısıdır. Özellikleri arasında Google hizmetleri ve hesaplarıyla senkronize etme, sekmeli tarama ve web sayfalarının otomatik çeviri ve yazım denetimi bulunur. Ayrıca, çok amaçlı adres çubuğu (omnibox) adı verilen tümleşik bir adres çubuğu / arama çubuğuna da sahiptir [46].

Tablo 6: Çalışmada Kullanılan Tarayıcılar ve Sürümleri.

	Mozilla Firefox	**Google Chrome**
Sürüm	61-62 (64-bit)	69 (64.bit)

Belirlenen test otomasyon araçlarında Record and Playback (Kaydet ve Yeniden Oynat) adı verilen özellik bulunmaktadır. Record and Playback, çoğunlukla bir test aracının web sitesiyle (veya web uygulamasıyla) etkileşimini kaydeden bir tarayıcı uzantısı/eklentisidir. Test komut dosyaları, test için yürütülebilmesi adına (oynatma) otomatik olarak oluşturulur. Aynı testleri defalarca yapmak için idealdir [47]. Kısaca Record and Playback, belirlenen test senaryosunun web sitesi üzerinde canlı olarak kayıt edilmesi ve gerekli düzenlemeler yaparak veya yapmaksızın olduğu gibi tekrar test edilmesine olanak sağlayan bir özelliktir. Bu özellik;

web sitesi henüz canlıya geçmeyen (yayınlanmadan) yerelde test senaryosu kaydı yapılarak, canlıya geçtiğinde aynı senaryonun test edilmesine imkân sağlar.

Kullanılan araçlarda Record and Playback özelliği bulunsa da, daha büyük proje ve test durumları için kod bilgisi gerekebilmektedir veya aracı kullanan teknik kişinin kodla müdahale ederek verimi arttırmasına olanak sağlanabilmektedir.

4. BULGULAR

Bu bölümde, ele alınan test otomasyon araçlarına ilişkin elde edilmiş analiz sonuçları verilecektir.

Kod bilgileri 5(beş) üzerinden değerlendirilerek, 1 hiç kod bilgisi gerektirmeden basit UI testlerinin yapılabileceği, 5 gerekli script dilinin veya HTML bilgisinin orta ve üstü derecede bilinmesinin büyük avantajları sağlayacağı durumlar için kullanılmıştır. Tablo 7.de çalışmada Kullanılan Test Otomasyon Araçlarının Kod Gereksinimleri verilmiştir.

Tablo 7: Çalışmada Kullanılan Test Otomasyon Araçlarının Kod Gereksinimleri.

	Kod Bilgisi	**Script Dili**	**Record and Playback**
Selenium WebDriver	5/5	Java, C#, Perl, Python, JavaScript, Ruby, PHP	Evet
Katalon Studio	3/5	Java/Groovy, JavaScript	Evet
Telerik Test Studio	1/5	C#, VB.NET	Evet
Ranorex Studio	2/5	C#, VB.NET	Evet
Squish	4/5	JavaScript, Perl, Python, Ruby	Evet
TestingWhiz	2/5	Java	Evet

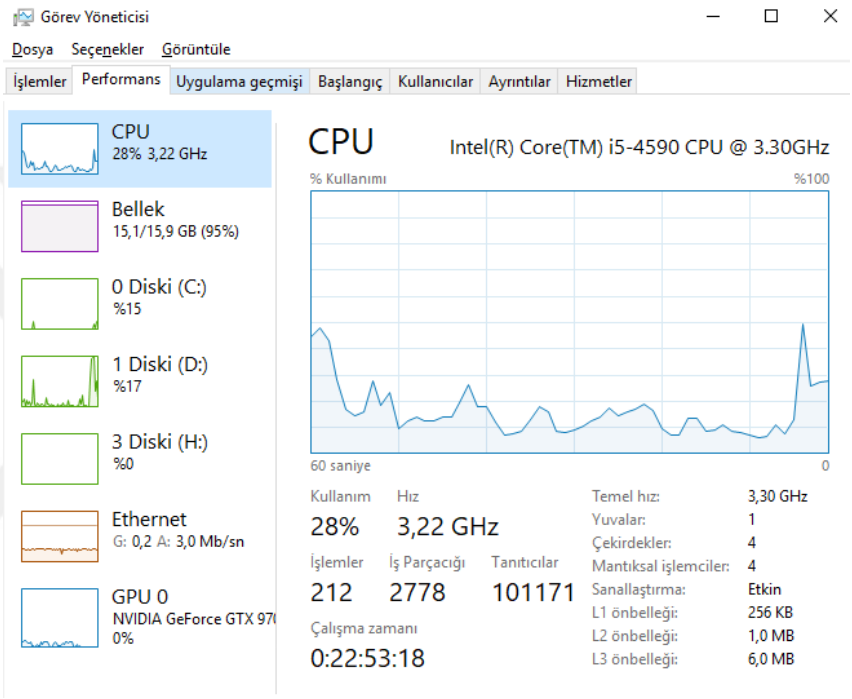
Çalışma sırasında belirlenen test otomasyon araçları, bilgisayarlarda her iki tarayıcıda da (Tablo 6) çalıştırıldıktan sonra toplam süreç boyunca ilgili bilgisayardaki (Tablo 5) işlemci (CPU) ve bellek (RAM) kullanımları ortalama değerleri hesaplanmıştır.

CPU ve RAM kullanım oranları, Windows 10 Görev Yöneticisi üzerinden video kaydı ile kayıt alınarak hesaplanmıştır.

Bu süreç içerisinde değerlendirilen her aracın sadece kendi kullanımı katılmış olup (Selenium için IntelliJ IDEA ile birlikte), tarayıcının sistem üzerindeki yükü hesaba katılmamıştır. Çalışma sırasında arka planda bilgisayarlara yük bindirecek veya araçların çalışmalarına engel olacak başka herhangi bir uygulama çalıştırılmamıştır. Web tarayıcıları her koşumdan önce sıfırlanarak çerezlerin ve geçici dosyaların etkisi azaltılmaya çalışılmıştır.

Her bir araç için aynı web sitesinin test senaryosu toplamda 10(on) defa koşularak ortalama değerler alınmış (Tablo 8, 9, 10, 11, 12 ve 13) ve her test otomasyon aracı için saptanan değerlerin ortalamaları hesaplanmıştır (Tablo 14).

Belirlenen web siteleri ve test otomasyon araçları ile, önceden hazırlanan senaryolar, son kullanıcı deneyimlerine dayanarak (User Acceptance Test mantığıyla) koşulmuştur.



Şekil 6: Örnek Windows 10 Görev Yöneticisi Performans Grafiği.

Test senaryolarının koşulması sırasında Selenium için Record and Playback özelliği kullanılmamıştır. WebDriver için gerekli kod IntelliJ IDEA ortamında Java diliyle yazılıp her iki web tarayıcısında çalıştırılmak üzere derlenmiştir.

Selenium WebDriver'ın, IntelliJ IDE yardımıyla çalıştırıldığında düşük donanımlı bilgisayarda, özellikle tek sayfada fazla öğe içeren web sitelerinde CPU ve RAM'i yorduğu, fakat yine de işlevselliğini kaybetmediği gözlemlenmiştir. Selenium'un her web sayfası için koşulan senaryolar sonrası gözlemlenen CPU ve RAM kullanımları Tablo 8.de verilmiştir.

Test senaryolarının koşulması sırasında Katalon Studio için Record and Playback özelliği kullanılmış, kodlama kullanılmamıştır.

Katalon Studio'nun, web sitesi içeriğinden bağımsız bir şekilde tüm senaryolarda benzer performans sergilediği gözlemlenmiştir. Aracın yüksek donanımlı sistemi de, düşük donanımlı sistem kadar yorduğu ve özellikle CPU kullanımı bakımından oldukça ağır yük bindirdiği gözlemlenmiştir. Katalon Studio'nun her web sayfası için koşulan senaryolar sonrası gözlemlenen CPU ve RAM kullanımları Tablo 9.da verilmiştir.

Tablo 8: Selenium WebDriver Test Aracının CPU ve RAM Kullanımı.

Test Edilen Web Sitesi	**Web Sitesi Türü**	**Performans**		
		Bilgisayar	**CPU(%)**	**RAM(%)**
TRT World	Haber	PC-1	10,3	4,9
		PC-2	14,2	21,6
BBC World	Haber	PC-1	7,1	2,1
		PC-2	21,6	30,4
The New York Times	Haber	PC-1	5,7	6,2
		PC-2	8,5	14,1
Tom's Hardware	Forum	PC-1	5,3	6,8
		PC-2	15,5	23,7
Linus Tech Tips	Forum	PC-1	7,8	6,3
		PC-2	5,2	9,4
xda Developers	Forum	PC-1	5,1	4,0
		PC-2	9,1	22,0
Twitter	Sosyal Medya	PC-1	6,5	5,1
		PC-2	3,8	22,8
Deviantart	Görsel Paylaşım	PC-1	9,1	4,4
		PC-2	14,5	23,3
YouTube	Video Paylaşım	PC-1	5,8	4,2
		PC-2	16,1	19,3
Amazon	Alışveriş	PC-1	6,2	3,2
		PC-2	12,1	34,6
HepsiBurada	Alışveriş	PC-1	6,6	4,8
		PC-2	14,4	18,9

Tablo 9: Katalon Studio Test Aracının CPU ve RAM Kullanımı.

Test Edilen Web Sitesi	**Web Sitesi Türü**	**Performans**		
		Bilgisayar	**CPU(%)**	**RAM(%)**
TRT World	Haber	PC-1	14,2	6,5
		PC-2	29,3	23,6
BBC World	Haber	PC-1	11,2	5,8
		PC-2	29,6	18,1
The New York Times	Haber	PC-1	11,5	8,4
		PC-2	27,0	26,0
Tom's Hardware	Forum	PC-1	12,4	3,4
		PC-2	29,9	21,0
Linus Tech Tips	Forum	PC-1	13,6	5,5
		PC-2	41,0	15,3
xda Developers	Forum	PC-1	11,4	6,4
		PC-2	35,1	27,8
Twitter	Sosyal Medya	PC-1	13,0	4,6
		PC-2	31,3	30,0
Deviantart	Görsel Paylaşım	PC-1	13,8	5,5
		PC-2	32,8	27,4
YouTube	Video Paylaşım	PC-1	14,5	3,6
		PC-2	38,6	25,4
Amazon	Alışveriş	PC-1	13,3	7,5
		PC-2	41,6	19,8
HepsiBurada	Alışveriş	PC-1	14,1	3,8
		PC-2	32,4	17,5

Test senaryolarının koşulması sırasında Telerik Test Studio için Record and Playback özelliği kullanılmış, kodlama kullanılmamıştır. Telerik Test Studio'nun, web sitesi içeriğinden bağımsız bir şekilde tüm senaryolarda benzer ve etkili bir performans sergilediği gözlemlenmiştir. Aracın CPU ve RAM kullanımları, iki cihazda da birbirine yakın değerleri vermekte ve sistemi yormadan, minimum düzeyde kullanım sağlamaktadır. Telerik Test Studio'nun her web sayfası için koşulan senaryolar sonrası gözlemlenen CPU ve RAM kullanımları Tablo 10.da verilmiştir.

Tablo 10: Telerik Test Studio Test Aracının CPU ve RAM Kullanımı.

Test Edilen Web Sitesi	Web Sitesi Türü	Performans		
		Bilgisayar	CPU(%)	RAM(%)
TRT World	Haber	PC-1	1,0	1,1
		PC-2	1,0	2,0
BBC World	Haber	PC-1	0,9	1,0
		PC-2	1,2	2,2
The New York Times	Haber	PC-1	0,9	1,1
		PC-2	0,6	2,3
Tom's Hardware	Forum	PC-1	1,1	0,9
		PC-2	0,7	2,1
Linus Tech Tips	Forum	PC-1	1,0	1,0
		PC-2	1,1	1,8
xda Developers	Forum	PC-1	1,0	1,1
		PC-2	0,8	1,4
Twitter	Sosyal Medya	PC-1	0,9	1,1
		PC-2	0,7	1,7
Deviantart	Görsel Paylaşım	PC-1	1,1	1,0
		PC-2	1,2	2,1
YouTube	Video Paylaşım	PC-1	0,9	1,2
		PC-2	1,0	2,0
Amazon	Alışveriş	PC-1	1,0	1,0
		PC-2	1,2	2,2
HepsiBurada	Alışveriş	PC-1	1,1	0,9
		PC-2	0,9	2,1

Test senaryolarının koşulması sırasında Ranorex Studio için Record and Playback özelliği kullanılmış, kodlama kullanılmamıştır.

Ranorex Studio'nun, metin yoğunluğu bulunan web sitelerinde RAM kullanım değerlerinin arttığı gözlemlenmiştir. Bu fark, özellikle düşük donanımlı bilgisayarda daha belirgin bir şekilde görülmektedir. Ayrıca Ranorex Studio'nun görsel içerikli sitelerde CPU kullanımını arttırdığı gözlemlenmiştir. Bunlara rağmen araç, her iki cihazda da sistemi yormadan işlevselliğini devam ettirmiştir. Ranorex Studio'nun her web sayfası için koşulan senaryolar sonrası gözlemlenen CPU ve RAM kullanımları Tablo 11.de verilmiştir.

Tablo 11: Ranorex Studio Test Aracının CPU ve RAM Kullanımı.

Test Edilen Web Sitesi	Web Sitesi Türü	Performans		
		Bilgisayar	CPU(%)	RAM(%)
TRT World	Haber	PC-1	0,8	1,0
		PC-2	1,2	2,3
BBC World	Haber	PC-1	1,2	0,7
		PC-2	1,6	2,4
The New York Times	Haber	PC-1	0,6	0,7
		PC-2	2,0	3,2
Tom's Hardware	Forum	PC-1	0,9	0,8
		PC-2	1,7	2,6
Linus Tech Tips	Forum	PC-1	1,1	1,0
		PC-2	1,9	3,0
xda Developers	Forum	PC-1	0,6	0,9
		PC-2	1,9	3,4
Twitter	Sosyal Medya	PC-1	0,9	0,5
		PC-2	1,4	1,9
Deviantart	Görsel Paylaşım	PC-1	1,2	1,0
		PC-2	1,1	2,7
YouTube	Video Paylaşım	PC-1	1,0	1,2
		PC-2	2,2	2,9
Amazon	Alışveriş	PC-1	0,9	0,8
		PC-2	2,5	2,5
HepsiBurada	Alışveriş	PC-1	1,0	0,9
		PC-2	0,8	2,3

Test senaryolarının koşulması sırasında Squish için Record and Playback özelliği kullanılmış, kodlama kullanılmamıştır. Squish'in, özellikle düşük donanımlı bilgisayarda RAM kullanımı arttırdığı ve sistemi RAM bakımından yormaya başladığı gözlemlenmiştir. Araç, web sitesi içeriğinden bağımsız bir şekilde, herhangi bir şartı gözetmeksizin değişken RAM kullanımında bulunmuştur. CPU kullanımları her ne kadar birbirine yakın olsa da RAM kullanımındaki değişikliklerin sebebi net bir şekilde belirlenememiştir. Squish'in her web sayfası için koşulan senaryolar sonrası gözlemlenen CPU ve RAM kullanımları Tablo 12.de verilmiştir.

Tablo 12: Squish Test Aracının CPU ve RAM Kullanımı.

Test Edilen Web Sitesi	Web Sitesi Türü	Performans		
		Bilgisayar	CPU(%)	RAM(%)
TRT World	Haber	PC-1	0,4	3,1
		PC-2	1,8	12,6
BBC World	Haber	PC-1	2,4	2,5
		PC-2	4,3	9,8
The New York Times	Haber	PC-1	0,8	2,9
		PC-2	4,4	13,9
Tom's Hardware	Forum	PC-1	2,5	1,5
		PC-2	5,3	14,8
Linus Tech Tips	Forum	PC-1	1,9	3,3
		PC-2	6,3	6,8
xda Developers	Forum	PC-1	2,4	4,5
		PC-2	5,6	7,8
Twitter	Sosyal Medya	PC-1	1,2	2,9
		PC-2	3,7	12,2
Deviantart	Görsel Paylaşım	PC-1	2,0	2,7
		PC-2	4,6	11,1
YouTube	Video Paylaşım	PC-1	0,6	3,8
		PC-2	4,2	1,6
Amazon	Alışveriş	PC-1	2,0	3,2
		PC-2	2,1	23,5
HepsiBurada	Alışveriş	PC-1	1,8	2,4
		PC-2	1,5	13,7

Test senaryolarının koşulması sırasında TestingWhiz için Record and Playback özelliği kullanılmış, kodlama kullanılmamıştır. TestingWhiz'in özellikle metin ağırlıklı web sitelerinde CPU'yu zorladığı ve RAM'i hafif kullanımda bıraktığı gözlemlenmiştir. Özellikle düşük donanımlı bilgisayarda CPU kullanımı %20 seviyelerine çıkmış, bu da sisteme ağır yük bindirmiştir. Yüksek donanımlı bilgisayarda ise CPU kullanımları birbirine yakın olup, sistemi çok fazla zorlamadığı gözlemlenmiştir. TestingWhiz'in her web sayfası için koşulan senaryolar sonrası gözlemlenen CPU ve RAM kullanımları Tablo 13.te verilmiştir.

Tablo 13: TestingWhiz Test Aracının CPU ve RAM Kullanımı.

Test Edilen Web Sitesi	Web Sitesi Türü	Performans		
		Bilgisayar	CPU	RAM
TRT World	Haber	PC-1	7,5	0,6
		PC-2	20,8	9,1
BBC World	Haber	PC-1	4,0	1,3
		PC-2	13,7	5,3
The New York Times	Haber	PC-1	6,7	0,9
		PC-2	18,6	4,2
Tom's Hardware	Forum	PC-1	6,9	1,0
		PC-2	7,2	8,5
Linus Tech Tips	Forum	PC-1	4,5	1,1
		PC-2	20,3	5,9
xda Developers	Forum	PC-1	7,6	0,9
		PC-2	22,3	3,8
Twitter	Sosyal Medya	PC-1	5,8	0,9
		PC-2	12,0	4,3
Deviantart	Görsel Paylaşım	PC-1	1,5	1,0
		PC-2	17,5	7,3
YouTube	Video Paylaşım	PC-1	5,4	0,6
		PC-2	9,0	7,4
Amazon	Alışveriş	PC-1	6,1	1,6
		PC-2	15,4	6,1
HepsiBurada	Alışveriş	PC-1	6,4	1,1
		PC-2	14,9	8,1

Sonuç olarak, çalışmada kullanılan tüm test otomasyon araçlarının ortalama CPU ve RAM kullanımı Tablo 14.te gösterilmiştir.

Tablo 14: Çalışmada Kullanılan Araçların Ortalama CPU ve RAM Kullanımları.

Test Otomasyon Aracı	Kullanılan Bilgisayar	CPU Kullanımı	RAM Kullanımı
Selenium WebDriver	PC-1	%7	%5
	PC-2	%12	%22
Katalon Studio	PC-1	%13	%6
	PC-2	%35	%20

Tablo 14 (Devam): Çalışmada Kullanılan Araçların Ortalama CPU ve RAM Kullanımları.

Telerik Test Studio	PC-1	%1	%1
	PC-2	%1	%2
Ranorex Studio	PC-1	%1	%1
	PC-2	%2	%3
Squish	PC-1	%2	%3
	PC-2	%3	%12
TestingWhiz	PC-1	%5	%1
	PC-2	%14	%6

Bu değerlerden yola çıkarak; Katalon Studio'nun düşük donanımlı bilgisayarda sisteme çok fazla yük bindirdiği, Ranorex Studio'nun her iki bilgisayarda da donanımı yormadan çalışmaya devam ettiği ve Selenium'un IDE ile kullanılmasına rağmen sisteme bindirdiği yük bakımından çok da kötü bir durumda olmadığı gözlemlenmiştir. Sonuç olarak ise Telerik Test Studio'nun en az CPU ve RAM kullandığı gözlemlenerek, performans açısından çalışmadaki diğer test otomasyon araçlarından daha etkili olduğu görülmüştür.

Kullanılan test araçları, sağlayıcı firmalardan akademik çalışma adı altında talep edilmiştir. Araçların ticari amaçla kullanım için belirlenen ücretlendirme Tablo 15.te sunulmuştur.

Tablo 15: Test Otomasyon Araçlarının Güncel Ücretlendirmeleri (Eylül-2018).

Test Otomasyon Aracı	**Minimum Ücret**	**Maksimum Kullanıcı/Cihaz Sayısı**	**Destek**
Selenium WebDriver	Ücretsiz	-	Kullanıcı topluluğu tarafından sağlanan ücretsiz destek
Katalon Studio	Ücretsiz	-	Dokümantasyon, forum, eğitici videolar ücretsiz. Helpdesk 7/24 yardım aylık $249
Telerik Test Studio	$2,499/Yıl	1 Cihaz	12 ay boyunca haftalık iş günleri içerisinde 24 saat öncelikli destek
Ranorex Studio	€2,290/Tek sefer. 1 yıllık güncelleme ve destek dahil. €490/Yıllık yenileme	1 Kullanıcı	İsteğe bağlı uzatılabilir güncelleme ve yardım

Tablo 15 (Devam): Test Otomasyon Araçlarının Güncel Ücretlendirmeleri (Eylül-2018).

Squish	+€1,300/Yıl	Kullanıcı sayısına göre esnetilen fiyat	12 ay boyunca güncelleme ve yardım
TestingWhiz	Esnek. Projenin bütçesine ve süresine göre şirket tarafından teklif sunulmaktadır.	Proje sayısına bağlı olarak değişmektedir.	Teklife göre değişiklik göstermektedir.

Tablo 16'da Selenium WebDriver aracını ilk defa kullanacak kullanıcılar için kullanım deneyimleri sunulmuştur. Tabloda Selenium WebDriver'ın, bir cihaza (referans olarak PC1 ve PC2 alınmıştır) yüklenmesinin kolaylığına değinilmiştir. Arayüze sahip olmayan Selenium WebDriver'ın yerine birlikte kullanılan IDE'nin arayüz etkili olacağından, bu araç için 'Arayüz Tasarımı' kriteri göz ardı edilmiştir. Araç içerisinde test kurullumu ve çalıştırılması kriterinde ise kod bilgisi gerekliliği belirtilmiş ve birkaç eklentiye ihtiyaç duyulduğu vurgulanmıştır. Sonuç olarak yüzdelik (%) bir oranla aracın, çalışmadaki diğer araçlara göre kullanım kolaylığı puanlanmıştır.

Tablo 16: Selenium WebDriver İçin Kullanım Değerlendirmesi.

Yükleme ve Çalıştırma	Diğer IDE yazılımlarla entegre çalıştığından dolayı; çalışmadaki diğer araçlarla karşılaştırıldığında kurulum açısından çok zahmetli ve uğraştırıcı kalmaktadır.
Arayüz Tasarımı	WebDriver'ın herhangi bir arayüzü olmadığından, kullanılan IDE'nin arayüzü göz önünde bulundurulmalıdır.
Test Kurulumu ve Çalıştırma	Test kurulumu ve çalıştırılması, web tarayıcısı üzerine küçük eklentiler kurularak, kodun derlenmesi şeklindedir. Bu da kod bilgisi gerektirmektedir.
Sonuç	%40 ; Aracın kurulum ve çalıştırması belli oranda bilgisayar ve programlama bilgisi gerektirdiğinden, teknik elemanlar dışında kullanımı kolay değildir.

Tablo 17.de Selenium WebDriver aracını kullanmak ve bu araç yardımıyla test koşmak için gerekli kod bilgisi değerlendirilmiştir. Desteklenen diller çalışma içerisinde daha önceden verildiği için (Tablo 7), bu kısımda sadece birçok dilde kullanım sağlamakta olduğu değerlendirmesi yapılmıştır. Desteklenen dillerin hangi seviyede bilinmesi gerektiği detaylandırılmadan belirtilmiş ve çalışmadaki diğer araçlara göre ne kadar seviyede kod kullanılması gerektiği üzerine değerlendirme yapılmıştır. Çalışmadaki diğer araçlara göre kod gereksiniminin fazla olduğu gözlenmiş ve sonuç puanı da bu bilgiler ışığında belirlenmiştir.

Tablo 17. Selenium WebDriver İçin Kod Gereksinim Değerlendirmesi.

Desteklenen Dil	Birçok dilde kullanım sağlamaktadır.
Dilin Ne Seviyede Kullanıldığı	Desteklenen dilin en az başlangıç-orta seviyede bilinmesi gerekmektedir.
Kod Kullanım Oranı	WebDriver'ın IDE üzerinden kodlanarak derlenmesi, testin koşulmasını sağladığı için kod kullanım oranı diğer araçlara göre bir hayli fazladır.
Sonuç	%25 ; Birçok dilde destek veriyor olmasına rağmen; çalışmadaki diğer test otomasyon araçlarına kıyasla çok fazla kod gereksinimi olması, son kullanıcıdan kod birikimi ve tecrübesi gerektirmektedir.

Tablo 18.de Selenium WebDriver aracı ile koşulan test sonucunda aracın sonuç raporları değerlendirilmiştir. Selenium WebDriver, arayüze sahip olmadığı için rapor detayları kısmında IDE üzerindeki sonuçlar baz alınmıştır. Raporun okunabilirliği de yine aynı şekilde IDE üzerinden değerlendirilmiştir. Raporun sundukları ise IDE'nin rapor içeriği olarak göz önüne alınmış ve en sonunda sonuç olarak Seleinum WebDriver'ın raporlama puanı verilmiştir.

Tablo 18: Selenium WebDriver İçin Raporlama Değerlendirmesi Gereksinim Değerlendirmesi.

Rapor Detayları	WebDriver'ın rapor olarak sundukları, IDE üzerindeki kod derleme sonuçlarıdır.
Rapor Okunabilirliği	Rapor, derleme sonucu olduğundan dolayı raporun okunması programlama bilgisi gerektirmektedir.
Raporun Sundukları	Teknik hata kodları şeklinde sonuç vermektedir.
Sonuç	%15 ; Okunabilirliği kolay, detaylı bir rapor sunulmamaktadır. Hatanın nerede olduğunu bulmak için daha fazla zaman harcanması gerekmektedir.

Tablo 19'da Selenium WebDriver aracının fiyatlandırmasıyla ilgili değerlendirme yapılmıştır. Dönemlik ücretler, çalışmada daha önce fiyat tablosu verildiği için (Tablo 15) bu kısımda kullanıcı yorumu olarak değerlendirilmiştir. Fiyat/performans analizi, dönemlik ücret karşılığında cihazlar üzerindeki donanımsal performansı karşılaştırılarak değerlendirmeye alınmıştır. Fiyata dahil olan hizmetler belirtilmiş ve sonuç olarak da verilen ücrete karşılık alınan hizmet ve performans yorumlanarak puanlandırma yapılmıştır.

Tablo 19: Selenium WebDriver İçin Fiyatlandırma Değerlendirmesi.

Aylık/Yıllık/Dönemlik Fiyat	Tamamen ücretsiz bir kullanım sağlamaktadır.
Fiyat/Performans Analizi	Ücretsiz olması ve performansının tatmin edici olması; Selenium'u özellikle orta ölçekli projeler için tercih edilebilir kılmaktadır.
Fiyata Dahil Hizmetler	Tamamen ücretsiz kullanımın yanında ücretsiz dokümantasyon ve ücretsiz teknik destek mevcuttur.
Sonuç	%100

Tablo 20.de Selenium WebDriver aracının teknik destek kriterleri değerlendirilmiştir. Hangi canlı destek kanallarını kullandığı belirtilmiş ve resmi destek kanallarına ulaşılabilecek zamanlar, kendi web sayfalarından alınan bilgileri doğrultusunda, tabloya eklenmiştir. Topluluk (topluluk) başlığında ise aracın dünya çapındaki kullanıcılarının birbirleriyle ne derece ve hangi platformlarda yardımlaştığı ve iletişim içinde olduğu araştırılarak değerlendirilmiştir. Sonuç olarak resmi destek kanalları, destek saatleri ve topluluğun yaygınlığı ve yardımcı olması göz önünde bulundurularak puanlandırma yapılmıştır.

Tablo 20: Selenium WebDriver İçin Destek Kanalları Değerlendirmesi.

Destek Kanalları	IRC ChatRoom ve Slack üzerinden canlı teknik destek vermektedir.
Destek Zamanları	Belli bir destek saati vaat edilmemektedir.
Topluluk	Oldukça geniş bir kullanıcı/geliştirici topluluğuna sahiptir ve bu sayede resmi kanallar dışında da destek alınabilmektedir.
Sonuç	%50 ; Kesin bir teknik destek olmaması ve topluluğun çok geniş bir şekilde yayılmış olması, alınacak yardımın kesinliği ve süresi konusunda kullanıcıyı şüpheye düşürebilmektedir.

Tablo 21.de Selenium WebDriver aracını kullanan/kullanmak isteyen kullanıcıların bu araçla ilgili dokümantasyon ve eğitimlere nasıl ve ne şekilde ulaşabilecekleri üzerine değinilmiştir.

Öğretici Kaynak başlığında, ilgili aracın resmi olarak video tutorial sunmadığını belirlenmiştir. Eğitim Dokümanları, resmi olarak herkese açık bir şekilde sunulmaktadır ve dünya çapında bağımsız olarak birçok Selenium WebDriver kurslarının mevcudiyeti araştırılmıştır. Sonuç olarak eğitim dokümanlarına erişim kolaylığı ve doküman içerikleri göz önünde bulundurularak puanlandırma yapılmıştır.

Tablo 21: Selenium WebDriver İçin Dokümantasyon ve Eğitim Değerlendirmesi.

Eğitim Kaynağı	Resmi bir eğitim kanalı bulunmamaktadır.
Eğitim Dokümanları	Ücretsiz eğitim dokümanları herkese açık bir şekilde paylaşılmaktadır.
Kurslar	Dünya çapında birçok sertifikasyon ve eğitim kursları bulunmaktadır.
Sonuç	%80 ; Eğitim kaynaklarına erişim oldukça kolay olmaktadır.

Tablo 22.de Katalon Studio aracına, ilk kullanıcı gözüyle bakılarak değerlendirme yapılmıştır. Yükleme ve çalıştırma başlığında ilgili aracın cihazlara (PC1 ve PC2 referans alınarak) yüklenmesinin kolaylığı kısaca belirtilmiştir. Aracın arayüz tasarımı değerlendirilirken kullanıcıyı yorup yormadığı göz önünde bulundurulmuş ve çalışmadaki diğer araçların arayüz tasarımıyla karşılaştırılmıştır. Test kurulum ve çalıştırma kısmında ise arayüzün bu aşamaları nasıl etkilediği göz önünde bulundurularak Record and Playback özelliği vurgulanmıştır. Sonuç olarak arayüz sadeliği ve kurulumun hızı/kolaylığı değerlendirilerek puanlama yapılmıştır.

Tablo 22: Katalon Studio İçin Kullanım Değerlendirmesi.

Yükleme ve Çalıştırma	Yazılımın kurulumu son derece basit ve zahmetsizdir.
Arayüz Tasarımı	Kullanıcıyı yormayan basit bir arayüze sahiptir.
Test Kurulumu ve Çalıştırma	Basit arayüzü ve Record and Playback özelliği sayesinde test kurulumu kolay olmakla birlikte test gereksinimlerinin belirlenmesi gibi durumlarda kullanıcıyı uğraştırabilmektedir.
Sonuç	%70 ; Sade arayüzü ve hızlı kurulumu büyük kolaylık sağlamaktadır.

Tablo 23.te Katalon Studio aracını kullanırken kullanıcıdan beklenen kod bilgisi değerlendirilmiştir. Desteklenen dil sayısının azlığı vurgulanmış; ama bu dillerin de sık tercih edilen diller olması değerlendirme sırasında göz önünde bulundurulmuştur. Hiçbir koda ihtiyaç duymaması , ilk kullanıcı gözüyle bakılarak değerlendirme yapılmıştır. Katalon Studio'nun kod gerektirmeden çalışmasının mümkün olduğu vurgulanarak, kod yardımıyla

test kalitesinin arttırılabileceği gözlemlenmiştir. Sonuç olarak desteklenen dillerin miktarı ve test kalitesi artırımına imkân saplanması göz önünde bulundurularak puanlama yapılmıştır.

Tablo 23: Katalon Studio İçin Kod Gereksinim Değerlendirmesi.

Desteklenen Dil	Desteklenen dil sayısı oldukça kısıtlıdır.
Dilin Ne Seviyede Kullanıldığı	Az sayıda dil desteklenmesine ve kod kullanımı düşük seviyede olmasına rağmen, desteklenen dilin orta seviyede bilinmesi gereklidir.
Kod Kullanım Oranı	Basit scriptlerle test kalitesi arttırılabildiği gibi, hiç kod kullanmadan da test senaryosu koşulabilmektedir.
Sonuç	%60 ; Desteklenen dil sayısının az olması, müdahale için fazladan bilgi gerektirebilmektedir.

Tablo 24.te Katalon Studio aracının test raporları değerlendirilmiştir. Raporlardaki detayların yeterliliği yorumlanarak; raporların okunabilirliği için gerekli bilgilere değinilmiştir. Raporun sundukları basitçe belirtilmiş ve sonuç olarak rapor detaylarının belirsizliği vurgulanarak, çalışmadaki diğer araçlara kıyasla raporlama konusunda geri kaldığı gözlenmiştir ve buna göre puanlama yapılmıştır.

Tablo 24: Katalon Studio İçin Raporlama Değerlendirmesi.

Rapor Detayları	Rapor detayları, bazen sorunun çözümünü bulmak için yeterli detaya sahip olmamaktadır.
Rapor Okunabilirliği	Rapor okunabilirliği temel seviyede HTML veya sitenin teknik içeriği hakkında bilgi sahibi olunmasını gerektirmektedir.
Raporun Sundukları	Rapor her adımın kabul/ret durumunu göstermektedir.
Sonuç	%40 ; Rapor detaylarının çok belirgin olmaması ve rapor okunabilirliğinin zor olması, aracı raporlama konusunda geride bırakmaktadır.

Tablo 25.te Katalon Studio aracının fiyatlandırmasıyla ilgili değerlendirme yapılmıştır. Dönemlik ücretler, çalışmada daha önce fiyat tablosu verildiği için (Tablo 15) bu kısımda kullanıcı yorumu olarak değerlendirilmiştir. Tabloda, aracın kullanımının ücretsiz fakat şartlı olduğu belirtilmiştir. Performans olarak, çalışmadaki diğer araçların geride kalması gözlenmiştir. Ücretin sadece destek hizmetini kapsadığı belirtilip, sonuç olarak puanlama yapılırken ücret şartı ve performans göz önünde bulundurulmuştur.

Tablo 25: Katalon Studio İçin Fiyatlandırma Değerlendirmesi.

Aylık/Yıllık/Dönemlik Fiyat	Tamamen ücretsiz kullanım sağlamaktadır. Teknik destek için aylık ücret ödenmesi gerekmektedir.
Fiyat/Performans Analizi	Ücretsiz olmasına rağmen test edilen cihazları diğer araçlara göre fazla yorması, fiyat performans açısından dengede kalmasını sağlamaktadır.
Fiyata Dahil Hizmetler	Teknik destek için verilen ücret karşılığında 7/24 teknik destek sağlanmaktadır.
Sonuç	%80 ; Kullanımın ücretsiz, fakat teknik desteğin ücretli olması sonuç puanını düşürmektedir.

Tablo 26'da Katalon Studio aracının teknik destek kriterleri değerlendirilmiştir. Resmi destek kanallarının varlığına değinilmiş ve bu kanalların hizmet verdiği zamanlar, kendi web sitesinden alınan verilere göre, tabloya eklenmiştir. Kullanıcı topluğunun yetersiz olması vurgulanarak sonuç kısmında, çalışmadaki diğer araçlara göre teknik destek konusunda Katalon Studio'nun yetersiz olduğu belirtilmiş ve buna göre puanlama yapılmıştır.

Tablo 26: Katalon Studio İçin Destek Kanalları Değerlendirmesi.

Destek Kanalları	Resmi teknik destek kanalı mevcuttur.
Destek Zamanları	Resmi teknik destek 7/24 hizmet vermektedir.
Topluluk	Çok geniş bir topluluğu olmaması, desteği yetersiz kılmaktadır.
Sonuç	%20 ; Resmi teknik destek dışında fazla seçenek olmaması ve resmi teknik desteğin ücretli olması destek puanını düşürmektedir.

Tablo 27.de Katalon Studio aracının dokümantasyon ve eğitim olanakları değerlendirilmiştir. Resmi web sitesinde öğretici kaynak ve eğitim dokümanlarının sunulduğu gözlemlenmiş ve resmi olarak -ücret karşılığında- kurslar verildiği gözlenmiştir. Sonuç olarak eğitim kaynaklarının ücretsiz olması göz önünde bulundurulmuş ve bu kaynakların özellikle yeni kullanıcılar için yeterli olması sonucuna varılarak puanlama yapılmıştır.

Tablo 27: Katalon Studio İçin Dokümantasyon ve Eğitim Değerlendirmesi.

Öğretici Kaynak	Ücretsiz olarak resmi web sitesinde sunulmaktadır.

Eğitim Dokümanları	Ücretsiz olarak resmi web sitesinde sunulmaktadır.
Kurslar	Resmi olarak eğitim hizmeti vermektedir.
Sonuç	%100 ; Tamamen ücretsiz eğitim kaynakları, aracın kullanımı için oldukça yeterlidir.

Tablo 28.de Telerik Test Studio aracının çalışmadaki cihazlar (PC1 ve PC2) referans alınarak yüklenmesinin kolaylığına değinilmiş ve arayüz tasarımının sadeliği vurgulanmıştır. Test kurulumu ve çalıştırma sırasında arayüzün sadeliği göz önünde bulundurularak değerlendirme yapılmış ve sonuç olarak çalışmadaki diğer araçlara kıyasla kolay bir kurulum sunması göz önüne alınarak puanlama yapılmıştır.

Tablo 28: Telerik Test Studio İçin Kullanım Değerlendirmesi.

Yükleme ve Çalıştırma	Kurulumu son derece basit ve hızlıdır.
Arayüz Tasarımı	Basit arayüzü, kullanım kolaylığı sağlamaktadır.
Test Kurulumu ve Çalıştırma	Son derece basit bir şekilde senaryolar oluşturularak testlerin koşulması sağlanmaktadır.
Sonuç	%80 ; Kolay kurulum ve test hazırlığı sunmaktadır.

Tablo 29'da Telerik Test Studio aracını kullanmak için gerekli kod bilgisi değerlendirilmiştir. Değerlendirme, desteklenen dillerin popülerliği göz önünde bulundurulmuş ve bu kullanımın da verim amaçlı olduğu gözlenmiştir. Kullanıcının kod müdahalesinde bulunabileceği görülmüş ve puanlamada koda ihtiyaç duyulmaması ve desteklenen dillerin popülerliği hesaba katılmıştır.

Tablo 29: Telerik Test Studio İçin Kod Gereksinim Değerlendirmesi.

Desteklenen Dil	Kısıtlı sayıda ama popüler dilleri desteklemektedir.
Dilin Ne Seviyede Kullanıldığı	Orta seviye kod bilgisi, verim arttırımı sağlamak amacıyla yeterlidir.
Kod Kullanım Oranı	Kullanıcının isteğine bağlı olarak kod müdahalesi yapılabilmekle beraber, kod ihtiyacı yoktur.
Sonuç	%100 ; Kod ihtiyacının olmaması ve C# gibi popüler dilleri desteklemesi, aracın puanını olumlu yönde etkilemektedir.

Tablo 30'da Telerik Test Studio aracı ile koşulan testler sonucu sunduğu raporlar değerlendirilmiştir. Telerik Test Studio'da rapor detaylarının yetersiz olduğu; buna rağmen okunabilirliğinin yüksek olduğu gözlenmiştir. Sunulan raporların görsel içerik olarak zengin olması olumlu değerlendirilmiş ve buna göre puanlama yapılmıştır.

Tablo 30: Telerik Test Studio İçin Raporlama Değerlendirmesi.

Rapor Detayları	Raporlardaki detay oldukça kısıtlıdır.
Rapor Okunabilirliği	Raporlar teknik olmayan cümlelerle sunulmakta olduğundan dolayı, okunabilirliği gayet basittir.
Raporun Sundukları	Renkli grafiklerle, görsel bir raporlandırma yapmaktadır.
Sonuç	%70 ; Rapor okunabilirliği ve raporun grafiklerle sunulması, kullanıcıya kolaylık sağlamaktadır.

Tablo 31.de Telerik Test Studio aracının fiyatlandırmasıyla ilgili değerlendirme yapılmıştır. Dönemlik ücretler, çalışmada daha önce fiyat tablosu verildiği için (Tablo 15) bu kısımda kullanıcı yorumu olarak değerlendirilmiştir. Tabloda, ücretlendirmenin dönemi ve şekli (kullanıcı/cihaz) belirtilmiştir. Fiyat/performans değerlendirmesi sırasında iki kriter de çalışmadaki diğer uygulamalar kıyaslandığında farklı sıralamaları verdiği için bu alanda ortalamada kaldığı sonucuna varılmıştır. Fiyata Dahil Hizmetlerde sadece teknik desteğin olması puanlamayı olumsuz etkilese de asıl olumsuz etkiyi fiyatın çalışmadaki diğer araçlardan yüksek olması ve kullanıcı yerine cihaz bazlı bir ücretlendirilmesi yapmıştır. Puanlama da tüm bunlar göz önünde bulundurularak yapılmıştır.

Tablo 31: Telerik Test Studio İçin Fiyatlandırma Değerlendirmesi.

Aylık/Yıllık/Dönemlik Fiyat	Yıllık tek cihaz üzerinden fiyatlandırma yapılmaktadır.
Fiyat/Performans analizi	Performans olarak, çalışmadaki diğer araçlardan çok daha üstün performans sergilemesine rağmen, fiyatlandırma olarak diğer araçların gerisinde kalmaktadır.
Fiyata Dahil Hizmetler	Bir yıl boyunca geçerli teknik destek sunulmaktadır.
Sonuç	%20 ; Ücretlendirmenin, diğerlerine kıyasla yüksek olması ve bir cihaz üzerinden yapılması

Tablo 32.de Telerik Test Studio aracının teknik destek kriterleri değerlendirilmiştir. Resmi destek kanalları olduğu belirtilmiş fakat bu kanalların kullanım ücretine dahil olduğu ve destek zamanlarının oldukça kısıtlı olduğu gözlenmiştir. Topluluk olarak aktif ve kullanıcı sayısı yüksek denilebilecek resmi forum incelenmiştir. Sonuç olarak ise teknik desteğin zamanının kısıtlı olması büyük bir olumsuzluk olarak değerlendirilerek puanlama yapılmıştır.

Tablo 32: Telerik Test Studio İçin Destek Kanalları Değerlendirmesi.

Destek Kanalları	Resmi teknik destek mevcuttur.
Destek Zamanları	Kullanım ücret karşılığında yıl boyunca çalışma saatleri içerisinde destek sunulmaktadır.
Topluluk	Resim forum aktiftir ve forumda iki buçuk milyonun üzerinde kullanıcı bulunmaktadır
Sonuç	%60 ; Ücret alınmasına rağmen resmi desteğin kısıtlı zamanlarının olması, aracı destek konusunda diğer araçlardan geride bırakmaktadır.

Tablo 33.te Telerik Test Studio aracının sağladığı eğitim kaynakları değerlendirilmiştir. Resmi olarak sunulan ücretsiz eğitimlerin içerikleri incelenerek, oldukça detaylı oldukları gözlenmiştir. Aynı zamanda ücret karşılığında resmi olarak eğitim imkanları sunulmaktadır. Tüm bunlar göz önünde bulundurulduğunda Telerik Test Studio'nun dokümantasyon ve eğitim konusunda başarılı olduğu kanısına varılmış ve buna göre puanlama yapılmıştır.

Tablo 33: Telerik Test Studio İçin Dokümantasyon ve Eğitim Değerlendirmesi.

Öğretici Kaynak	Resmî sitesinde eğitim videoları yer almaktadır.
Eğitim Dokümanları	Ücretsiz eğitim dokümanları sunulmaktadır.
Kurslar	Belirli ücret karşılığında resmi site üzerinden, çeşitli başlıklar altında online eğitimler verilmektedir.
Sonuç	%100 ; Detaylı ve ücretsiz eğitimler, aktif forum ve çeşitli blog yazıları ücretsiz olarak sunulmaktadır.

Tablo 34.te Ranorex Studio'nun kullanım deneyimleri sunulmuştur. Kurulumun basit olduğu belirtilmiş, arayüzün ise çalışmadaki diğer araçlara göre biraz daha karışık olduğu gözlenmiştir. Fakat bu karışık arayüz, IDE arayüzlerine benzediği için özellikle kod lama tecrübesi olan kullanıcıları yormayacağı düşünülerek buna göre puanlama yapılmıştır.

Tablo 34: Ranorex Studio İçin Kullanım Değerlendirmesi.

Yükleme ve Çalıştırma	Yazılımın kurulumu son derece basittir.
Arayüz Tasarımı	Sade olmayan, ama bir IDE'yi andıran arayüze sahiptir.
Test Kurulumu ve Çalıştırma	Testlerin kurulumu ve koşulması basit birkaç komutla sağlanabilmektedir.
Sonuç	%80 ; Programlama tecrübesi olan ve kullanıcılar için oldukça kolay bir

	kullanım sağlamaktadır.

Tablo 35.te Ranorex Studio aracının etkili kullanımı için gerekli kod bilgisi değerlendirilmiştir. Etkili ve ayrıntılı test koşulması için, desteklenen dillerin bir miktar kod bilgisine ihtiyaç olduğu gözlenmiş ve popüler dilleri desteklemesi e göz önünde bulundurularak; kod müdahalesiyle detaylı test sonuçları oluşturulabileceği belirtilerek puanlama yapılmıştır.

Tablo 35: Ranorex Studio İçin Kod Gereksinim Değerlendirmesi.

Desteklenen Dil	Kısıtlı sayıda ama popüler dilleri desteklemektedir.
Dilin Ne Seviyede Kullanıldığı	Başlangıç-orta seviye kod bilgisi, etkili test senaryoları yazmak için yeterlidir.
Kod Kullanım Oranı	Kodsuz bir kullanım sağlamakla birlikte; ince ve önemli detaylar için kod kullanımı gerekebilmektedir.
Sonuç	%80 ; Popüler dilleri desteklemek ve kodsuz kullanım sağlamakla birlikte; detaylı bir test senaryosu koşulması için kod ile müdahale edilmesi gerekmektedir.

Tablo 36'da Ranorex Studio'nun sunduğu test raporları değerlendirilmiştir. Aracın sunduğu raporların detayları tatmin edici bulunmuş olup, raporların adım adım olduğu gözlenmiştir. Test raporlarının kategorilere ayrılmış olmasının yorumu kullanıcıya göre değişiklik gösterecek olsa da, bu durumun raporun bir bütün olarak görülmesine engel olduğu sonucuna varılmıştır. Tüm bu durumlar göz önünde bulundurularak son puanlama yapılmıştır.

Tablo 36: Ranorex Studio İçin Raporlama Değerlendirmesi.

Rapor Detayları	Test koşusu sonunda sunulan rapor, çıkan hataların sebebini gösterecek kadar detaylıdır.
Rapor Okunabilirliği	Raporların kendi içlerinde kategorilere ayrılması, okunabilirliği kolaylaştırıyor gibi gözükse de, sonucu bir bütün olarak görmeyi zorlaştırmaktadır.
Raporun Sundukları	Hangi eylemin hangi aşamada ne şekilde yapıldığı gibi sonuçlar gösterilmektedir.
Sonuç	%80 ; Okunabilirlik konusunda sonuç bütününü görmekte kullanıcıya kolaylık sağlamamaktadır.

Tablo 37.de Ranorex Studio'nun fiyatlandırmasıyla ilgili değerlendirme yapılmıştır. Dönemlik ücretler, çalışmada daha önce fiyat tablosu verildiği için (Tablo 15) bu kısımda kullanıcı yorumu olarak değerlendirilmiştir. Dönemlik ücretin, çalışmadaki diğer araçlara göre farklı bir sistemle işlenmesi olumlu bir durum olarak değerlendirilmiştir. Fakat ilk seferde talep edilen kullanım ücretinin, dönemlik ücretlere kıyasla fazla olması ve ücretle sunulan hizmette sadece destek ve güncelleme olması olumsuz olarak değerlendirilmiştir. Tüm bunlara rağmen aracın performansı göz önünde bulundurulduğunda, çalışmadaki diğer araçlara kıyasla daha iyi bir sonuç sunduğundan dolayı puanlandırma yapılırken fiyat/performans değerlendirmesi, puanlamayı olumlu etkilemiştir.

Tablo 37: Ranorex Studio İçin Fiyatlandırma Değerlendirmesi.

Aylık/Yıllık/Dönemlik Fiyat	Tek kullanıcı için tek seferlik ücret karşılığında güncelleme ve destek dahil kullanım açılmakta olup, yıllık yenileme daha düşük fiyatlandırma üzerinden yapılmaktadır.
Fiyat/Performans analizi	Sunduğu performans göz önünde bulundurulduğunda, çalışmadaki diğer otomasyon araçlarının önüne geçebilmektedir.
Fiyata Dahil Hizmetler	Tek seferlik ücrette bir yıllık güncelleme ve destek, yıllık yenilemede ise kullanım devamı ve uzatılabilir destek sağlanmaktadır.
Sonuç	%60 ; Yıllık yenileme ücreti, diğer araçlara kıyasla daha düşük olsa da, yüksek seviyede tek seferlik ücret alınması ve desteğin bu ücrete dahil edilme durumu mevcuttur.

Tablo 38.de Ranorex Studio'nun teknik destek kriterleri değerlendirilmiştir. Ücrete dahil teknik destek, canlı destek sağlamamasına rağmen e-mail veya form yoluyla 24 saat içerisinde dönüş alındığı tecrübe edilmiştir. Forum ve blogların aktif olması, kullanıcı kitlesinin etkileşim içinde olduğunu göstermektedir. Bu durumlar göz önünde bulundurularak puanlama yapılmıştır.

Tablo 38: Ranorex Studio İçin Destek Kanalları Değerlendirmesi.

Destek Kanalları	Resmi destek kanalı, canlı destek sağlamamaktadır.
Destek Zamanları	E-mail veya form yoluyla 24 saat içerisinde destek sunulmaktadır.
Topluluk	Aktif forum ve bloglar.
Sonuç	%90 ; Canlı destek sunulmaması, desteğin gecikmesine sebep olabilmektedir.

Tablo 39'da Ranorex Studio'nun eğitim kanalları değerlendirilmiştir. Çeşitli kaynak ve dokümanların resmi olarak ücretsiz sunulduğu gözlenmiştir. Ayrıca araçla ilgili kursların yine resmi kanallar aracılığıyla online olarak sunulduğu görülmüştür. Tüm bunlar, Ranorex Studio'nun dokümantasyon ve eğitim konusunda son derece başarılı olduğunu göstermektedir.

Tablo 39: Ranorex Studio İçin Dokümantasyon ve Eğitim Değerlendirmesi.

Öğretici Kaynak	Resmi eğitim videoları ücretsiz olarak sunulmaktadır.
Eğitim Dokümanları	Resmi eğitim dokümanları ve webinar, ücretsiz sunulmaktadır.
Kurslar	Resmi kanallarda online kurslar yer almaktadır.
Sonuç	%100 ; Tüm eğitim dokümanları resmi kanallar aracılığıyla sunulmaktadır.

Tablo 40'ta Squish aracının kullanımının ilk kullanıcı gözüyle değerlendirilmesine yer verilmiştir. Aracın yüklenmesi ve çalıştırılmasının, çalışmadaki diğer araçlardan daha zor olmadığı tecrübe edilmiştir. Arayüz olarak Squish de Ranorex Studio gibi IDE'ye benzer bir tasarım sunsa da programlama tecrübesi olanlara hitap eden, daha basit bir arayüze sahiptir. Bu durumun, programlama tecrübesi olmayan kullanıcılar için alışma sürecini uzatacağı düşünülerek puanlama yapılmıştır.

Tablo 40: Squish İçin Kullanım Değerlendirmesi.

Yükleme ve Çalıştırma	Hızlı ve basit bir kurulum sağlamaktadır.
Arayüz Tasarımı	Basit bir IDE arayüzüne benzer bir arayüze sahiptir.
Test Kurulumu ve Çalıştırma	Programlama tecrübesi olan kullanıcılar için kod derleme kolaylığında test koşulmasını sağlar.
Sonuç	%80 ; Basit bir arayüze sahip olmasına rağmen, IDE benzeri arayüz, programlama tecrübesi olmayan kullanıcılar için alışma süreci gerektirebilmektedir.

Tablo 41.de Squish aracının test koşulması için ihtiyaç duyduğu kod bilgisi değerlendirilmiştir. kullanmak ve bu araç yardımıyla test koşmak için gerekli kod bilgisi değerlendirilmiştir. Test koşulması sırasında göz önünde sürekli kod bulunması ve anlık müdahaleler için en az orta seviyede kod bilgisine ihtiyaç duyulması, aracın kod gereksinimi konusunda çalışmadaki diğer araçlardan geride kalmasına sebep olduğu sonucuna varılmıştır.

Tablo 41: Squish İçin Kod Gereksinim Değerlendirmesi.

Desteklenen Dil	Popüler script dillerini desteklemektedir.
Dilin Ne Seviyede Kullanıldığı	Desteklenen dilin orta-iyi derecede bilinmesi gerekmektedir.
Kod Kullanım Oranı	Sık bir şekilde kod kullanma ihtiyacı ve kod bilgisi gerektirmektedir.
Sonuç	%40 ; Record and Playback özelliği olmasına rağmen göz önünde sürekli kodların olması ve anlık müdahale için yeterli kod bilgisine sahip olunması gerekmektedir.

Tablo 42.de Squish'in test sonuç raporlarının değerlendirmesi yer almaktadır. Squish'in en ufak sonuçları dahi kategorilendirerek bunları renkli tablolar hâlinde sunması, detaylı rapor analizi açısından olumlu olsa da; raporların okunabilirliğini zorlaştırdığı sonucuna varılmıştır. Fakat rapor detayları ve raporların adım adım sonuç sunmasının bu açığı biraz da olsa kapatabildiği sonucuna varılmıştır.

Tablo 42: Squish İçin Raporlama Değerlendirmesi.

Rapor Detayları	Her senaryonun, her durumun sonucu ayrı ayrı kategorilendirilerek, renkli tablolar aracılığıyla kullanıcıya sunulmaktadır.
Rapor Okunabilirliği	Test sonucunu bir bütün olarak basit bir şekilde sunmamaktadır.
Raporun Sundukları	Senaryo, durum ve adım sayıları dahil olmak üzere tüm adımları tek tek raporlamaktadır.
Sonuç	%80 ; Çok fazla ayrıntıya gidilmesi, okunabilirlik açısından olumsuzluğa sebep olabilmektedir.

Tablo 43.te Squish'in fiyatlandırma kriterleri değerlendirilmiştir. Sunduğu performans, çalışmadaki diğer araçlara göre daha düşük olduğundan dolayı fiyat/performans değerlendirmesinde olumsuz etki gözlenmiştir. Ücretlendirmenin net olmadığı araştırılmış ve bu durumun, büyüyen ekipler için fazladan maliyet olduğu sonucuna varılmıştır. Yenileme ücretinin, ilk ücretle aynı olması da göz önünde bulundurularak puanlama yapılmıştır.

Tablo 43: Squish İçin Fiyatlandırma Değerlendirmesi.

Aylık/Yıllık/Dönemlik Fiyat	Belli bir alt sınırdan başlayan ve ekipteki kullanıcı sayısına göre değişen

	fiyat seçenekleri sunulmaktadır.
Fiyat/Performans analizi	Sunduğu performans ve ücretlendirmeye bakıldığında, çalışmada değerlendirilen diğer araçların gerisine kalmaktadır.

Tablo 43: Squish İçin Fiyatlandırma Değerlendirmesi.

Fiyata Dahil Hizmetler	Bir yıl boyunca ücrete dahil destek ve güncelleme desteği sunulmaktadır.
Sonuç	%40 ; Yenileme ücretinin ilk fiyat ile aynı olması, büyüyen ekipler için maliyeti arttırmaktadır.

Tablo 44.te Squish'in destek kriterleri değerlendirilmiştir. Canlı destek olmadığı ve net bir destek zamanı belirtilmediği gözlenmiştir. Resmi kanallar araçlığıyla form veya e-mail yoluyla destek alınabildiği görülmüş ve topluluğun çeşitli etkinlikler düzenlediği bilgileri edinilmiştir. Canlı desteğin olmaması, destek için bir zaman sözü verilmemesi ve çeşitli etkinliklerin düzenlenmesi göz önünde bulundurularak puanlama yapılmıştır.

Tablo 44: Squish İçin İçin Destek Kanalları Değerlendirmesi.

Destek Kanalları	Resmi kanallarda form veya e-mail yoluyla destek alınabilmektedir.
Destek Zamanları	Canlı destek hattı olmadığından dolayı net bir destek zamanı belirtilmemiştir.
Topluluk	Yazılımın dağıtımcısı şirketin forumu mevcuttur. Ayrıca çeşitli zamanlarda çeşitli etkinlikler düzenlenmektedir.
Sonuç	%60 ; Canlı destek sunulmamaktadır ve destek kanallarına diğer araçların destek kanallarına kıyasla daha zahmetli bir şekilde ulaşılmaktadır.

Tablo 45.te Squish aracının sunduğu eğitimler değerlendirilmiştir. Çeşitli formatlarda eğitim materyallerinin resmi olarak sunulduğu görülmüştür. Ücretsiz eğitim dokümanlarının resmi olarak sunulduğu gözlenmiş; eğitim ve danışmanlık hizmetlerinin deneyimli elemanlar önderliğinde verildiği görülmüştür. Sonuç olarak, düzenlenen çeşitli etkinlikler de değerlendirmeye katılarak puanlama yapılmıştır.

Tablo 45: Squish İçin Dokümantasyon ve Eğitim Değerlendirmesi.

Öğretici Kaynak	Screencast ve webinar formatında eğitim materyalleri mevcuttur.
Eğitim Dokümanları	Resmi kanallardan ücretsiz eğitim dokümanları sağlanmaktadır.
Kurslar	Tecrübeli elemanlar önderliğinde kısa süreli eğitim ve danışmanlık hizmetleri verilmektedir.

Sonuç	%80 ; Ücretsiz eğitim dokümanları sunulmakta ve etkinlikler düzenlenmektedir.

Tablo 46'da TestingWhiz aracının kullanım değerlendirmesi verilmiştir. Aracın kurulumunun basit olması ve arayüzününü IDE benzerliği gözlenmiştir. Test kurulumu ve çalıştırması, sıfırdan bir yazılım projesi oluşturmaya benzediği için, alışılması zor ve zahmetli bir test kurulumu olduğu sonucuna varılarak buna göre puanlama yapılmıştır.

Tablo 46: TestingWhiz İçin Kullanım Değerlendirmesi.

Yükleme ve Çalıştırma	Kurulumu basittir.
Arayüz Tasarımı	Arayüz tasarımı IDE arayüzünü andırmaktadır.
Test Kurulumu ve Çalıştırma	Kurulum aşaması, sıfırdan bir yazılım projesi oluşturmaya benzemektedir. Bu nedenle alışması zaman alabilmektedir.
Sonuç	%60 ; Test kurulumu kolay olmamaktadır.

Tablo 47.de TestingWhiz'in kod gereksinimleri değerlendirilmiştir. Araç kullanıldığında amaçları olan 'daha az kod' durumunu karşıladığı gözlenmiştir. Neredeyse hiç kod ihtiyacına duymadan, kullanıcının test senaryolarına odaklanması amaçlandığı tecrübe edilmiş ve bu durum olumlu bir şekilde göz önünde bulundurularak puanlama yapılmıştır.

Tablo 47: TestingWhiz İçin Kod Gereksinim Değerlendirmesi.

Desteklenen Dil	Çok kısıtlı sayıda dil desteklemektedir.
Dilin Ne Seviyede Kullanıldığı	Başlangıç seviyesinde bilinmesi yeterlidir.
Kod Kullanım Oranı	Kod kullanımına neredeyse hiç ihtiyaç duyulmamaktadır.
Sonuç	%90 ; Koda ihtiyaç minimum indirgenmiştir ve kullanıcının kod yerine test senaryolarına odaklanması amaçlanmaktadır.

Tablo 48.de TestingWhiz test aracının sunduğu raporlar değerlendirilmiştir. TestingWhiz'in raporlarda ayrıntılara girmediği ve sunduğu raporların etkili okunabilmesi programlama tecrübesine ihtiyaç duyduğu gözlenmiştir. Bu durumun, hata çözümlerinde zaman kaybettireceği varsayılarak değerlendirmeye alınmıştır. TestingWhiz'in raporlarda test koşulması sırasındaki eylemleri adım adım belirtmesi ise olumlu bir sonuç olarak görülmüş, buna göre puanlama yapılmıştır.

Tablo 48: TestingWhiz İçin Raporlama Değerlendirmesi.

Rapor Detayları	Ayrıntılara çok girmeden yüzeysel rapor sunulmaktadır.
Rapor Okunabilirliği	Rapor okunabilirliği, programlama tecrübesi gerekmektedir.
Raporun Sundukları	Testin koşulması sırasındaki eylemler adım adım belirtilmektedir.
Sonuç	%60 ; Raporun yüzeysel olması ve bununla birlikte raporun okunabilirliğinin zayıf olması; bazı hataların çözümünde zaman kaybettirebilmektedir.

Tablo 49'da TestingWhiz test aracının fiyat performans değerlendirilmesi yapılmak istenmiş, fakat mevcut durumlarda böyle bir değerlendirmenin mümkün olmadığı vurgulanmıştır.

Tablo 49: TestingWhiz İçin Fiyatlandırma Değerlendirmesi.

Aylık/Yıllık/Dönemlik Fiyat	Proje bütçesi, proje süresi ve şirketin büyüklüğüne göre teklifler sunulmaktadır. Bu nedenle çalışma sürecinde TestingWhiz için fiyatlandırma değerlendirmesi yapılamamıştır.
Fiyat/Performans analizi	
Fiyata Dahil Hizmetler	
Sonuç	%10 ; Net bir fiyat etiketi olmaması, kullanıcının karar vermesini zorlaştırabilir.

Tablo 50.de TestingWhiz aracının teknik destek kriterleri değerlendirilmiştir. TestingWhiz'in, çalışmadaki diğer araçlardan farklı olarak telefonla destek sunduğu gözlenmiş, bu durumun acil destek ihtiyaçlarında kolaylık sağlayacağı öngörülmüştür. Topluluk olarak, çalışmadaki diğer araçlara göre daha zayıf bir topluluğa sahip olmasının forumların aktiflik durumunu etkilediği gözlenmiştir. Telefon dışında e-mail ve resmi kanallardaki form araçlığıyla da destek alınabildiği görülerek puanlama yapılmıştır.

Tablo 50: TestingWhiz İçin Destek Kanalları Değerlendirmesi.

Destek Kanalları	Telefon, e-mail ve resmi kanallardaki form yoluyla destek imkânı sunulmaktadır.
Destek Zamanları	Telefonla destek, acil durumlarda kolaylık sağlayabilmektedir.
Topluluk	Aktif ve yoğun bir foruma sahip değildir.
Sonuç	%80 ; Geniş ve aktif bir topluluğunun olmaması, kullanıcıyı resmi teknik desteğe zorlamaktadır.

Tablo 51.de TestingWhiz'in sunduğu eğitim materyalleri değerlendirilmiştir. Resmi kanallarda özet niteliğinde eğitim videoları olduğu ve yazılı materyallerin de ayrıntıya girmeden özet olarak sunulduğu gözlenmiştir. Araç eğitim materyallerinin tek metin altında toplanması, kullanıcının istediği başlıktaki eğitime erişiminin kısıtlanması olarak öngörülmüştür. Metin dışında çeşitli görsel eğitim materyallerinin sunulduğu gözlenerek puanlama yapılmıştır.

Tablo 51: TestingWhiz İçin Dokümantasyon ve Eğitim Değerlendirmesi.

Öğretici Kaynak	Kısa ve özet niteliğinde eğitim videoları bulunmaktadır.
Eğitim Dokümanları	Doküman şeklinde olmamasına rağmen, tek metin altında araç hakkında özet bilgiler sunulmaktadır. Bunun dışında çeşitli infografikler, örnek videolar ve webinar mevcuttur.
Kurslar	Webinar aracılığıyla eğitim materyalleri sunulmaktadır.
Sonuç	%40 ; Eğitim materyalleri, diğer araçlara kıyasla yetersiz kalmaktadır.

Tablo 16 – 51 arasındaki değerlendirmelerin ortalamaları alınarak Tablo 52.deki kullanım değerlendirmesi oluşturulmuştur. Bu genel değerlendirmede ilgili test otomasyon aracının sıfırdan kullanımı varsayılarak değerlendirilmiştir.

Tablo 52: Test Otomasyon Araçlarının Ortalama Değerlendirmesi.

Test Otomasyon Aracı	**A**	**B**	**C**	**D**	**E**	**F**	**TOPLAM**
Selenium WebDriver	4	3	2	10	5	8	32/60
Katalon Studio	6	6	4	8	2	10	36/60
Telerik Test Studio	8	10	6	2	6	10	42/60
Ranorex Studio	8	8	8	6	10	10	50/60
Squish	10	4	8	4	10	6	42/60
TestingWhiz	6	8	6	1	8	4	33/60

A: Kullanım Kolaylığı

B: Kod Gereksinimi

C: Raporlama

D: Fiyatlandırma

E: Destek

F: Dokümantasyon ve Eğitim

Bu bağlamda 1-60 arasında derecelendirme aşağıdaki dağılımla yapılmıştır:

- Kullanım Kolaylığı (Arayüz Sadeliği) (1-10 arası; 1 çok zor, 10 çok kolay)
- Kod Gereksinimi (1-10 arası; 1 maksimum ihtiyaç, 10 minimum ihtiyaç)
- Raporlama (1-10 arası; 1 anlaşılması zor, 10 anlaşılması kolay)
- Fiyatlandırma (1-10 arası; 1 yüksek, 10 düşük)
- Destek (1-10 arası; 1 ulaşılması zor, 10 ulaşılması kolay)
- Dokümantasyon ve Eğitim (1-10 Arası; 1 erişimi zor, 10 erişimi kolay)

Puanlama sonucuna göre en yüksek skoru Ranorex Studio test otomasyon aracı almış ve böylece çalışmadaki değerlendirme kriterlerine göre, çalışmada kullanılan diğer test otomasyon araçlarından daha çok tercih edilebilir olduğu gözlemlenmiştir.

5. TARTIŞMA VE SONUÇ

Son ve ortalama değerlendirmede gözlemlendiği üzere, günümüzde test otomasyon araçları neredeyse tüm test sürecini otomatize etmeye yaklaşmış durumdadır. Elbette elle müdahale gerektiren durumlar ve sınır durumları (edge-case) yaşanacaktır.

Öncelikle önemli testlerin otomatikleştirilmesi, daha fazla geri besleme ile daha iyi sonuçlar verecektir. Testlerin otomasyonu, test mühendisinin iş yükünü azaltabilir, yazılım kalitesini iyileştirebilir ve test süresini kısaltabilir. Otomatik Yazılım Testini kullanmanın birçok yararı vardır, ancak bunu başarmak kolay bir iş değildir.

Özellikle web sitesi geliştirilirken test sürecine önem verilmemekte ve kullanıcıdan alınan geri dönüş üzerine bakım yapılmaktadır. Bu da hem geliştiriciyi hem de kullanıcıyı olumsuz yönde etkilemektedir.

Test sürecinin temelini oluşturan ve sürecin ilerleyen aşamalarını önemli noktada etkileyen ve ilk adım olan gereksinim analizi aşamasında hangi test aracının, ne sebeple seçileceği sorusu zaman kaybına yol açabilir. Her aracın tek tek denenerek uygun aracın seçilmeye çalışılması ise ayrı bir zaman ve emek harcanması anlamına gelmektedir. Bu durum da doğrudan test sürecinin ve dolaylı yoldan da yazılım geliştirme sürecinin özellikle zaman bakımından büyük bir aksamaya uğramasına sebep olacaktır.

Zaman ve maliyetin kritik noktalar olduğu yazılım geliştirme sürecinde böylesine bir zaman ve maliyet kaybı riskine girmek ise, projenin büyüklüğüyle orantılı olarak sıkıntılar doğuracak ve belki de projedeki elemanların motivasyonunda düşüşe sebebiyet verecektir.

Bu çalışmada son zamanlarda en çok tercih edilen yazılım test otomasyon araçlarının farklı donanım özelliklerine sahip bilgisayarlarda web sitelerini basitçe test etme süreçleri test edilerek, popüler araçların aynı şartlar altındaki durumları incelenmiş ve karşılaştırma yapılarak test otomasyon tercihi yapılmasında yardımcı olunmaya çalışılmıştır. Ayrıca; uygulamaları test eden araçların da teste tabi tutulması gerektiğini, böylece bu araçların kullanılırken kullanıcı tarafından nasıl bir izlenim bıraktığı yansıtılmaya çalışılmıştır.

KAYNAKLAR

[1] I. Burnstein, *Practical Software Testing: process oriented approach*, Springer Professional Computing, 2003.

[2] M. Fewster and D. Graham, *Software Test Automation: Effective Use of Test Execution Tools*, ACM Press/Addison-Wesley Publishing Co., 1999.

[3] T. Maneela and D. Gaurav, "A Research Study on importance of Testing and Quality Assurance in Software Development Life Cycle (SDLC) Models", International Journal of Soft Computing and Engineering (IJSCE) ISSN: 2231-2307, Volume-2, Issue-3, July 2012.

[4] R.Torkar, "Towards automated software Testing Techniques, classifications and frame work", Blekinge Institute of Technology, 2006.

[5] Hetzel, William C., The Complete Guide to Software Testing, 2nd ed. Publication info: Wellesley, Mass: QED Information Sciences, 1988.

[6] E. Dijkstra, "Notes on Structured Programming", Technical Report 70-WSK-03, Dept. of Mathematics, Technological University of Eindhoven, Netherlands, April 1970.

[7] Khaled M. Mustafa et.al, "Classification of Software Testing Tools Based on the Software Testing Methods", second International Conference on Computer and Electrical Engineering, Al-Zaytoonah University of Jordan, 2009.

[8] Keller, R.K., Weber, R., & Berner. S, "Observations and lessons learned from Automated testing". On Proceedings of the 27th International Conference on Software Engineering, 571-579.

[9] Sergey Uspenskiy, "A survey and classification of software testing tools", Lappeenranta University of Technology, 2010.

[10] Brain Marick, ―When Should a Test Be Automated‖, Software Testing Analysis & Review conference (STAR EAST), Orlando, FL, 1999.

[11] S.Berner, R.Weber, and R. K. Keller, ―Observations and lessons learned from automated testing‖. In Proceedings of the 27th international Conference on Software Engineering, 2005.

[12] R. Ramler and K. Wolfmaier, ―Economic Perspectives in Test Automation: Balancing Automated and Manual Testing with Opportunity Cost‖.AST'06, May 23, 2006, Shanghai, China.

[13] K.Karhu, et.al, ―Empirical Observations on Software Testing Automation‖, Lappeenranta University of Technology, International Conference on Software Testing Verification and Validation, 2009.

[14] O'Broin, Caoimh, "The Impact of Test Automation on Software Testers" (2015).

[15] IEEE 83a, 610.12-1990 - IEEE Standard Glossary of Software Engineering Terminology

[16] ISTQB Glossary, Online, https://glossary.istqb.org

[17] C.Kaner, "Pitfalls and strategies in automated software testing", Volume: 30, Issue: 4, April 1997.

[18] R. W. Rice, "Surviving the top ten challenges of software test automation". Software Testing, Analysis & Review Conference (STAR) East, Software Quality Engineering, 2003.

[19] John Watkins, Testing IT, Cambridge University Press, 2001.

[20] Glenford J. Myers, The Art of Software Testing, Third Edition, Pg.6

[21] R.Chopra, Software Testing: A Self-Teaching Introduction, Mercury Learning and Information, Pg.2

[22] R.Chopra, Software Testing: A Self-Teaching Introduction, Mercury Learning and Information, Pg.7

[23] Ahamed, S.S.Riaz, "Studying the Feasibility and Importance of Software Testing: An Analysis", IJEST Volume 1 Issue 3 2009 119-128.

[24] Software Testing Help, Online, https://www.softwaretestinghelp.com/what-is-software-testing-life-cycle-stlc/

[25] SoftServe Inc, Online, https://en.ppt-online.org/52766

[26] Chitirala, Sai Charan Raj, "Comparing the effectiveness of automated test generation tools "EVOSUITE" and "Tpalus"" (2015).

[27] Guru99, Functional vs Non-Functional Testing, Online, https://www.guru99.com/functional-testing-vs-non-functional-testing.html

[28] Guru99, Performance Testing, Online, https://www.guru99.com/performance-testing.html

[29] Guru99, User Acceptance Testing, Online, https://www.guru99.com/user-acceptance-testing.html

[30] Guru99, Alpha Testing vs Beta Testing, Online https://www.guru99.com/alpha-beta-testing-demystified.html

[31] Elfriede Dustin, Jeff Rashka, John Paul, Automated Software Testing: Introduction, Management, and Performance, Addison-Wesley Professional, 1999.

[32] Krify, Advantages of Manual Testing, Online, https://krify.co/tag/advantages-of-manual-testing/

[33] Medium, Best Automation Testing Tools For 2018 , Online, https://medium.com/@briananderson2209/best-automation-testing-tools-for-2018-top-10-reviews-8a4a19f664d2

[34] Software Testing Help, Top 20 Automation Testing Tools, Online, https://www.softwaretestinghelp.com/top-20-automation-testing-tools/

[35] DZone, Top 10 Automated Software Testing Tools, Online, https://dzone.com/articles/top-10-automated-software-testing-tools

[36] Guru99, Top 10 Automation Testing Tools, Online, https://www.guru99.com/automated-testing-tools.html

[37] Guru99, Introduction to Selenium, Online, https://www.guru99.com/introduction-to-selenium.html

[38] Selenium HQ, Documents, Online, https://www.seleniumhq.org/docs/01_introducing_selenium.jsp#introducing-selenium

[39] Wikipedia, IntelliJ IDEA, Online, https://en.wikipedia.org/wiki/IntelliJ_IDEA

[40] The Economic Times, Definition of 'Ranorex Tool', Online, https://economictimes.indiatimes.com/definition/ranorex-tool

[41] TestingWhiz, Online, https://www.testing-whiz.com/

[42] Computer Hope, Core i5, Online, https://www.computerhope.com/jargon/c/core-i5.htm

[43] Intel Core m3 İşlemciler, Online, https://www.intel.com.tr/content/www/tr/tr/products/processors/core/m3-processors.html

[44] Computer Hope, Firefox, Online, https://www.computerhope.com/jargon/f/firefox.htm

[45] Access Firefox, What is Firefox?, Online, https://www.accessfirefox.org/What_is_Firefox.php

[46] Computer Hope, Chrome, Online, https://www.computerhope.com/jargon/c/chrome.htm

[47] CloudQA, Online, https://cloudqa.io/cloudqa-advanced-record-playback-testing-tools/

EKLER

ÖZGEÇMİŞ



Kişisel Bilgiler	
Adı Soyadı	Ekrem EROL
Doğum Yeri	Küçükçekmece
Doğum Tarihi	10.02.1993
Uyruğu	☑ T.C. ❒ Diğer:
Telefon	05358132415
E-Posta Adresi	ekremerol12@gmail.com
Web Adresi	www.EkremErol.com

Eğitim Bilgileri Lisans	
Üniversite	İstanbul Üniversitesi
Fakülte	Mühendislik Fakültesi
Bölümü	Bilgisayar Mühendisliği
Mezuniyet Yılı	2016

Yüksek Lisans	
Üniversite	İstanbul Üniversitesi-Cerrahpaşa
Enstitü Adı	Lisansüstü Eğitim Enstitüsü
Anabilim Dalı	Bilgisayar Mühendisliği Anabilim Dalı
Programı	Bilgisayar Mühendisliği – Tezli Yüksek Lisans

Makale ve Bildiriler