

T.C.
KIRIKKALE ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI
YÜKSEK LİSANS TEZİ

ENTİTY FRAMEWORK'ÜN FARKLI VERİ TABANLARINDAKİ
PERFORMANS ANALİZİ

Tahsin BALCI

Haziran 2018

Bilgisayar Mühendisliđi Anabilim Dalında Tahsin BALCI tarafından hazırlanan ENTİTY FRAMEWORK'ÜN FARKLI VERİ TABANLARINDAKİ PERFORMANS ANALİZİ adlı Yüksek Lisans Tezinin Anabilim Dalı standartlarına uygun olduğunu onaylarım.

Prof. Dr. Hasan ERBAY

Anabilim Dalı Başkanı

Bu tezi okuduğumu ve tezin **Yüksek Lisans Tezi** olarak bütün gereklilikleri yerine getirdiğini onaylarım.

Doç..Dr Erman YÜKSELTÜRK
Danışman

Jüri Üyeleri

Başkan (Danışman) : Doç. Dr. Erman YÜKSELTÜRK _____
Üye : Dr. Öğr. Üyesi Bülent Gürsel EMİROĞLU _____
Üye : Dr. Öğr. Üyesi Hakan KÖR _____

...../...../.....

Bu tez ile Kırıkkale Üniversitesi Fen Bilimleri Enstitüsü Yönetim Kurulu Yüksek Lisans derecesini onaylamıştır.

Prof. Dr. Mustafa YİĞİTOĞLU
Fen Bilimleri Enstitüsü Müdürü



ÖZET

ENTITY FRAMEWORK'ÜN FARKLI VERİ TABANLARINDAKİ PERFORMANS ANALİZİ

BALCI, Tahsin

Kırıkkale Üniversitesi

Fen Bilimleri Enstitüsü

Bilgisayar Anabilim Dalı, Yüksek Lisans Tezi

Danışman: Doç. Dr. Erman YÜKSELTÜRK

HAZİRAN 2018, 71 sayfa

Yazılım geliştiriciler tarafından sıkça kullanılan Entity Framework'ün farklı veri tabanları ile kullanılarak performans analizlerinin yapılması ve böylelikle ilgili framework için veri tabanı seçimine yardımcı olacak öneriler çıkarılması amaçlanmaktadır. Bu amaca ulaşmak için çalışma sürecinde bir yazılım geliştirilmiştir. Geliştirilen yazılım, profesyonel bir yazılım geliştirme süreci takip edilerek tamamlanmıştır. Yazılım geliştirme süreçlerini yönetmek için şelale modeli tercih edilmiştir. Yazılım, Entity Framework'ün farklı veri tabanlarında (MYSQL, MSSQL, ORACLE, POSTGRESQL gibi) performansını test etmek için kullanılmıştır. Bunun için yazılımda farklı veri kümeleriyle kayıt ekleme, silme, güncelleme arama ve sıralama gibi temel veri tabanı işlemleri yapılırken İŞLEM SÜRESİ, RAM, CPU gibi kaynakların kullanımı karşılaştırmalı olarak analiz edilmektedir. Karşılaştırma sonuçları grafiksel olarak bulgular ve değerlendirme bölümünde detaylı olarak verilmiş ve yorumlanmıştır.

Anahtar kelimeler: Entity Framework, Mysql, Mssql, Orm, Oracle, Postgresql

ABSTRACT

PERFORMANCE ANALYSIS OF ENTITY FRAMEWORK IN DIFFERENT DATABASES

BALCI, Tahsin

Kırıkkale University

Graduate School of Natural And Applied Sciences

Department of Computer Engineering, M. Sc. Thesis

Supervisor: Doç. Dr. Erman YÜKSELTÜRK

June 2018, 71 pages

This research intends to conduct the performance analysis of The Entity Framework, which is frequently used by software developers, using different databases and thus to make suggestions that will help select a database for the relevant framework. In order to achieve this goal, a software was developed during the study period. The developed software has been completed by following a professional software development process. The waterfall model was chosen to manage the software development processes. The software was used to test the performance of the Entity Framework in different databases (such as MYSQL, MSSQL, ORACLE, POSTGRESQL). For this purpose, the use of resources such as TIME, RAM, CPU were analysed comparatively while basic database operations such as creating, deleting, updating, searching and sorting are done in the software with different data clusters. The comparison results are presented and interpreted graphically in the Findings and Evaluation section in detail.

Keywords: Entity Framework, Mysql, Mssql, Orm, Oracle, Postgresql

TEŐEKKÜR

Yüksek lisans tezimi hazırlarken yaptığım arařtırmalar ve deneyler sırasında yardımlarını esirgemeyen, bilgi, tecrübe ve görüşlerinden yararlandığım değerli danışman hocam, Sayın Doç. Dr. Erman YÜKSELTÜRK' e, tezimin birçok aşamasında yardım gördüğüm Ersin KORKMAZ' a teşekkür ederim.



İÇİNDEKİLER DİZİNİ

Sayfa

ÖZET	iv
ABSTRACT	v
TEŞEKKÜR	vi
İÇİNDEKİLER DİZİNİ	vii
ŞEKİLLER DİZİNİ	ix
ÇİZELGELER DİZİNİ	x
SİMGELER DİZİNİ	xi
KISALTMALAR DİZİNİ	xii
1. GİRİŞ	1
1.1. Amaç.....	3
2. OBJECT RELATION MAP (ORM)	4
2.1. ORM.....	4
2.2. Entity Framework.....	5
3. VERİ TABANLARI	9
3.1. Veri Tabanı.....	9
3.2. ADO.NET.....	9
3.3. MSSQL.....	9
3.4. MYSQL.....	10
3.5. ORACLE.....	10
3.6. POSTGRESQL	10
4. METODOLOJİ	11
4.1. Entity Framework Performans Test (EFPT) Yazılımı	11
4.1.1. Planlama	13
4.1.2. Analiz.....	13
4.1.3. Tasarım	14
4.1.4. Gerçekleştirime	14
4.1.5. Test.....	15
4.1.6. Bakım.....	15
4.2. Entity Framework Performans Testi Uygulaması.....	15
4.2.1. Ekran Tasarımlarına Genel Bakış	15
4.2.1.1. Giriş-Ana Ekran	16
4.2.1.2. Kayıt Ekleme Test Ekranı.....	17
4.2.1.3. Arama Testi Ekranı	20
4.2.1.4. Kayıt Silme Ekleme Test Ekranı	22
4.2.1.5. Güncelleme Testi Ekranı	23
4.2.1.6. Sırlama Testi Ekranı	25
4.2.2. Analiz Testlerin Uygulanışı	27
4.2.2.1. Kayıt Ekleme Testi.....	28
4.2.2.2. Kayıt Arama Testi.....	29

4.2.2.3. Kayıt Silme Testi.....	30
4.2.2.4. Kayıt Güncelleme Testi	32
4.2.2.5. Kayıt Sıralama Testi.....	33
4.3. Performans Kayıt Uygulaması.....	34
5. BULGULAR VE DEĞERLENDİRME	35
5.1. Testlere Genel Bakış	35
5.2. Kayıt Ekleme Testi'nin İncelenmesi	40
5.2.1. Kayıt Ekleme Testi Uygulama Cpu Analizi.....	40
5.2.2. Kayıt Ekleme Testi Uygulama Ram Analizi	42
5.2.3. Kayıt Ekleme Testi Veri Tabanı Engine Cpu Analizi.....	43
5.2.4. Kayıt Ekleme Testi Veri Tabanı Engine Ram Analizi.....	45
5.3. Kayıt Arama Testi'nin İncelenmesi.....	46
5.4. Kayıt Silme Testi'nin İncelenmesi	46
5.4.1. Kayıt Silme Testi Uygulama Cpu Analizi.....	47
5.4.2. Kayıt Silme Testi Uygulama Ram Analizi	48
5.4.3. Kayıt Silme Testi Veri Tabanı Engine Cpu Analizi.....	50
5.4.4. Kayıt Silme Testi Veri Tabanı Engine Ram Analizi.....	51
5.5. Kayıt Güncelleme Testi'nin İncelenmesi	53
5.5.1. Kayıt Güncelleme Testi Uygulama Cpu Analizi.....	53
5.5.2. Kayıt Güncelleme Testi Uygulama Ram Analizi.....	54
5.5.3. Kayıt Güncelleme Testi Veri Tabanı Engine Cpu Analizi	56
5.5.4. Kayıt Güncelleme Testi Veri Tabanı Engine Ram Analizi.....	57
5.6. Kayıt Sıralama Testi'nin İncelenmesi	59
5.6.1. Kayıt Sıralama Testi Uygulama Cpu Analizi.....	59
5.6.2. Kayıt Sıralama Testi Uygulama Ram Analizi	60
5.6.3. Kayıt Sıralama Testi Veri Tabanı Engine Cpu Analizi.....	62
5.6.4. Kayıt Sıralama Testi Veri Tabanı Engine Ram Analizi.....	63
5.7. Tüm Testlerin Ortak İncelenmesi	65
5.7.1. Tüm Testler İşlem Süresi Analizi	65
5.7.2. Tüm Testler İşlem Cpu Analizi	66
5.7.3. Tüm Testler İşlem Ram Analizi.....	67
6. SONUÇLAR.....	68
KAYNAKÇA.....	70

ŞEKİLLER DİZİNİ

<u>Şekil</u>	<u>Sayfa</u>
Şekil 2.1 Net Yapısı	6
Şekil 4.1 Şelale Modeli Aşamaları	12
Şekil 4.2 Genel Uygulama Yapısı	14
Şekil 4.3 Ekran Yerleşim Haritası	16
Şekil 4.4 Giriş Ekran Görüntüsü	17
Şekil 4.5 Kayıt Ekleme Testi Ekranı	17
Şekil 4.6 Kayıt Ekleme Testi Örnek Veriler Ekranı	19
Şekil 4.7 Analiz Rapor Ekranı	19
Şekil 4.8 Model Özellik Listesi	21
Şekil 4.9 Arama Kriter Listesi	22
Şekil 4.10 Arama Sonuç Listesi	22
Şekil 4.11 Güncelleme Model Alan Listesi	25
Şekil 4.12 Sıralama Model Alan Listesi	26
Şekil 4.13 Sıralama Yönü Seçenek Listesi	26
Şekil 5.1 Kayıt Ekleme Testi Uygulama Cpu Analizi	41
Şekil 5.2 Kayıt Ekleme Testi Uygulama Ram Analizi	42
Şekil 5.3 Kayıt Ekleme Testi Veri Tabanı Engine Cpu Analizi	44
Şekil 5.4 Kayıt Ekleme Testi Veri Tabanı Engine Ram Analizi	45
Şekil 5.5 Kayıt Silme Testi Uygulama Cpu Analizi	47
Şekil 5.6 Kayıt Silme Testi Uygulama Ram Analizi	49
Şekil 5.7 Kayıt Silme Testi Veri Tabanı Engine Cpu Analizi	50
Şekil 5.8 Kayıt Silme Veri Tabanı Engine Ram Analizi	52
Şekil 5.9 Kayıt Güncelleme Testi Uygulama Cpu Analizi	53
Şekil 5.10 Kayıt Güncelleme Testi Uygulama Ram Analizi	55
Şekil 5.11 Kayıt Güncelleme Testi Veri Tabanı Engine Cpu Analizi	56
Şekil 5.12 Kayıt Güncelleme Testi Veri Tabanı Engine Ram Analizi	58
Şekil 5.13 Kayıt Sıralama Testi Uygulama Cpu Analizi	59
Şekil 5.14 Kayıt Sıralama Testi Uygulama Ram Analizi	61
Şekil 5.15 Kayıt Sıralama Testi Veri Tabanı Engine Cpu Analizi	62
Şekil 5.16 Kayıt Sıralama Testi Veri Tabanı Engine Ram Analizi	64
Şekil 5.17 Tüm Test Ortalamaları İşlem Süresi Analizi	65
Şekil 5.18 Tüm Test Ortalamaları Cpu Analizi	66
Şekil 5.19 Tüm Test Ortalamaları Ram Analizi	67

ÇİZELGELER DİZİNİ

<u>Çizelge</u>	<u>Sayfa</u>
Çizelge 2.1 ORM Araçları	5
Çizelge 4.1 Kayıt Ekleme Testi Ekranı Kontrol Listesi	18
Çizelge 4.2 Kayıt Arama Testi Ekranı Kontrol Listesi	20
Çizelge 4.3 Kayıt Silme Testi Ekranı Kontrol Listesi	23
Çizelge 4.4 Test İş İstasyonu Özellik Listesi	27
Çizelge 5.1 Test Listesi	35
Çizelge 5.2 Kayıt Ekleme İşlem Süresi ve Ortalama Cpu Değeri	41
Çizelge 5.3 Kayıt Ekleme İşlem Süresi ve Ortalama Ram Değeri	43
Çizelge 5.4 Kayıt Ekleme İşlem Süresi ve Veri Tabanı Engine Cpu Değeri	44
Çizelge 5.5 Kayıt Ekleme İşlem Süresi ve Veri Tabanı Engine Ram Değeri	46
Çizelge 5.6 Kayıt Silme Uygulama Cpu Değeri	48
Çizelge 5.7 Kayıt Silme İşlem Süresi ve Ortalama Ram Değeri	49
Çizelge 5.8 Kayıt Silme İşlem Süresi ve Veri Tabanı Engine Cpu Değeri	51
Çizelge 5.9 Kayıt Silme İşlem Süresi ve Veri Tabanı Engine Ram Değeri	52
Çizelge 5.10 Kayıt Güncelleme Uygulama Cpu Değeri	54
Çizelge 5.11 Kayıt Güncelleme İşlem Süresi ve Ortalama Ram Değeri	55
Çizelge 5.12 Kayıt Güncelleme İşlem Süresi ve Veri Tabanı Engine Cpu Değeri	57
Çizelge 5.13 Kayıt Güncelleme İşlem Süresi ve Veri Tabanı Engine Ram Değeri	58
Çizelge 5.14 Kayıt Sıralama Uygulama Cpu Değeri	60
Çizelge 5.15 Kayıt Sıralama İşlem Süresi ve Ortalama Ram Değeri	61
Çizelge 5.16 Kayıt Sıralama İşlem Süresi ve Veri Tabanı Engine Cpu Değeri	63
Çizelge 5.17 Kayıt Sıralama İşlem Süresi ve Veri Tabanı Engine Ram Değeri	64

SİMGELER DİZİNİ

sn	Saniye
kb	Kilobyte
mb	MegaByte
mhz	MegaHertz
ghz	Gigahertz



KISALTMALAR DİZİNİ

EF

Entity Framework

ORM

Object Relation Mapping



1. GİRİŞ

Bilişim teknolojilerinde yaşanan hızlı gelişim her geçen gün daha fazla kamu ve özel sektördeki organizasyonları etkileyerek farklı çözümler üretmeye zorlamaktadır. Başka bir ifadeyle, belli başlı bir amaca ulaşmak için veri veya ham bilginin işlenerek ilgililere yarar sağlayacak biçime dönüştürülmüş hali olan bilgi, organizasyonlar tarafından sürekli daha kısa sürede erişilmek istenen en etkili faktör haline gelmiştir. Bu nedenle, günümüzde verilerin saklanması ve erişimi için en küçük organizasyonlarda bile artık veri tabanları yaygın olarak kullanılmaktadır.

Veri tabanı bilgilerin özel bir düzen içerisinde kayıt edilmesini sağlar. İstenildiğinde verilere hızlı ve güvenli erişimi sağlar [1, p. 30].

Bir başka ifade ile veri tabanı, kullanım amacına uygun olarak düzenlenmiş veriler topluluğudur. Birbirleriyle ilişkileri olan verilerin tutulduğu, mantıksal ve fiziksel olarak tanımlarının olduğu bilgi depolarıdır. Veri tabanları gerçekte var olan ve birbirleriyle ilişkisi olan nesnelere ve ilişkileri modellerdir. Veri tabanı yönetim sistemleri (VTYS) ise, verilere aynı anda birden çok bağlantı sağlayabilme özelliği sağlar. Bu sistemler, veri tabanı yönetiminin bir parçası olarak, verinin nasıl depolanacağı, kullanılacağı ve erişileceğini mantıksal olarak yönlendiren bir kurallar sistemidir [2] [3]

Veri tabanı yönetim sisteminin sahip olduğu özellikler bakımında kullanıcılarına farklı avantajlar sağlamaktadır, bunların bazıları şöyle sıralanabilir [3]

1. **Veri bağımsızlığı:** Uygulama programları veri depolanan ve temsil edilen yerden mümkün olduğunca bağımsız olmalıdır. Veri tabanı yönetim sistemleri, uygulama kodları gibi detaylardan ayırmak için verinin genel bir özetini sağlayabilir.
2. **Veri bütünlüğü:** Veri tabanı içindeki bir veri, bütün veri tablolarında birbiri ile bağlantılıdır. Bu verideki herhangi bir değişiklik bütün tablolarda

uygulanmalıdır. Bu veri kaydı bir tablodan silindi ise bütün tablolardan silinmelidir.

3. **Veri güvenliği:** Veri tabanı içindeki bir verinin, kullanıcı tarafından isteyerek veya yanlışlıkla bozulmasını önlemek için veri tabanı yönetim sistemlerinde güvenlik mekanizması oluşturulabilir. İstenildiği takdirde veri tabanı kullanıcı adı ve şifre ile korunabilmektedir. Bunlara ek olarak veri tabanı, kullanıcılara sınırlandırılabilir. Örneğin kullanıcı veri tabanı içindeki tablolardan sadece müsaade edilen tablo içindeki verileri görebilir.
4. **Veri tekrarı:** Aynı veri değişik kullanıcılar tarafından tekrar tekrar kaydedilmez. Böylece hem karmaşıklığın önüne geçilir hem de veri tabanında bir düzen ve hafıza olarak daha az bir kullanım sağlanmaktadır.
5. **Veri tutarlığı:** Aynı verinin değişik yerlerde birkaç veri tabanında bulunması durumunda, veride yapılan herhangi bir değişiklik diğer veri tabanlarında yapılmaz ise karmaşaya sebep olabilir. Veri tabanı yönetim sistemlerinde bunun önüne geçilmiş olur.

Veri tabanı sistemleri incelendiğinde, 1960'lı yıllardan itibaren çeşitli gelişmeler göstererek günümüze kadar gelmiştir. İlk yıllarda veri tabanları yerine, verileri muhafaza etmek için dosyalar kullanılmıştır. Veri tabanı mimarileri, 1960'lı yıllarda ilk olarak Ağ Modeli ile geliştirilmiştir. 1970'li yıllara gelince İlişkisel Model olarak adlandırılan diğer bir model geliştirilmiştir. Bu modeli temel alan 80'li yılların başında ticari ürünler olarak piyasaya çıkmıştır. Bu yıllarda bu model, DB2, FILEMAKER, ORACLE, SQL Server, SYBASE, INFORMIX gibi veri tabanı yönetim sistemleri tarafından kullanılmıştır. Günümüzde, bu modellerin üzerine Nesneye Yönelik Veri Modeli ve diğer farklı yaklaşımları kullanan birçok model geliştirilmiş olmasına rağmen, ilişkisel veri modeli halen yaygın bir kullanıma sahip kalmayı başarmıştır [4].

Nesneye yönelik programlama ve nesneye yönelik veri modellerinin yaygınlaşması veri tabanı programlama ve modellemesine yenilikler getirmiştir. Özellikle sağladığı avantajlardan dolayı artık yazılımcılar ve tasarımcılar frameworkleri kullanmaya başlamıştır. Microsoft firmasında .Net platformları için Entity Framework piyasaya

sürmüştür. Entity Framework, ORM (Object Relational Mapping) araçlarından biridir. ORM (Object Relational Mapping) ise veri tabanı ile nesneye yönelik programlama arasındaki ilişkiyi kuran teknolojidir. Başka bir ifadeyle Entity Framework, nesne tabanlı programlamada veri tabanındaki tablolara uygun nesnelere oluşturma tekniğidir denilebilir.

ORM ve Entity Framework teknolojisi veri tabanından bağımsız teknolojilerdir. Farklı veri tabanlarıyla kullanılabilir. Son yıllarda farklı projelerde kullanıldığı gözlenmektedir. Fakat hangi veri tabanında nasıl performans sergilediği ile ilgili yeterince çalışma olmadığı görülmektedir. Entity Framework kullanılarak yapılacak testlere ve bu test sonuçlarının değerlendirilmesine ihtiyaç vardır. Böylelikle yazılımcılara bu konuda önerilerde bulunulabilecektir. Bu çalışmada Entity Framework'ün farklı veri tabanları kullanılması analiz edilmesi ile ilgili bir araştırma yapılması planlanmaktadır.

1.1. Amaç

Araştırmada, son yıllarda yazılım geliştiriciler tarafından sıkça kullanılan Entity Framework'ün farklı veri tabanları ile kullanılarak performans analizlerinin yapılması ve böylelikle ilgili framework için veri tabanı seçimine yardımcı olacak öneriler çıkarılması amaçlanmaktadır. Bu amaca ulaşmak için çalışma sürecinde bir yazılım geliştirilmiştir. Geliştirilen yazılım, profesyonel bir yazılım geliştirme süreci takip edilerek tamamlanmıştır. Yazılım geliştirme süreçlerini yönetmek için şelale modeli tercih edilmiştir. Yazılım, Entity Framework'ün farklı veri tabanlarında (MySQL, MsSql, Oracle, PostgreSQL gibi) performansını test etmek için kullanılmıştır. Bunun için yazılımda farklı veri kümeleriyle kayıt ekleme, silme, güncelleme arama ve sıralama gibi temel veri tabanı işlemleri yapılırken İŞLEM SÜRESİ, RAM, CPU gibi kaynakların kullanımı karşılaştırmalı olarak analiz edilmektedir. Bu çalışmada, geliştirilen yazılım ve bu yazılım kullanılarak yapılan testler ve sonuçları detaylı olarak verilecektir.

2. OBJECT RELATION MAP (ORM)

2.1. ORM

ORM (Object Relational Mapping), en temel tanımıyla veri tabanı ile nesneye yönelik programlama (OOP) arasındaki ilişkiyi kuran teknolojidir. Başka bir ifadeyle, nesnelere veri tabanındaki tablolara bağlayan ve veri alış-verişini yapan bir tekniktir. Son zamanlarda yaygın olarak kullanılan bu tekniğin bazı avantaj ve dezavantajları şöyle sıranabilir [5]:

Avantajları :

- Nesne yönelimli programlama olarak kod geliştirmeye imkan sağlamaktadır.
- Sql bilmeden bir veri tabanına bağlanabilir ve sql ile ilgili bir çok işlemi rahatlıkla yapabilmek.
- Herhangi bir database bağımlılığı yoktur.
- Oldukça esnek bir yapı sağlamaktadır.

Dezavantajları :

- Performans (ADO.NET kadar hızlı çalışmamaktadır)
- Veri tabanından bilgi alış-verişi sırasında kontrol %100 bizde değildir. Bazen oluşturulan sql ifadelerinde sorun çıkabilir veya biz daha güzel ve hızlı çalışacak sql ifadeleri yazabileceğimiz durumlar olabilir
- Veri tabanı bağımsızdır. Fakat uygulama tarafındaki nesnelere veri tabanındaki nesnelere birbirine map edildiği için nesne bağımlılığı vardır

ORM tekniği belli bir programlama diline bağlı değildir ve her nesneye yönelik programlama dilinde yazılabilir veya kullanılabilir. Günümüzde kullanılan popüler ORM kütüphaneleri Çizelge 2.1'de sunulmuştur.

Çizelge 2.1 ORM Araçları

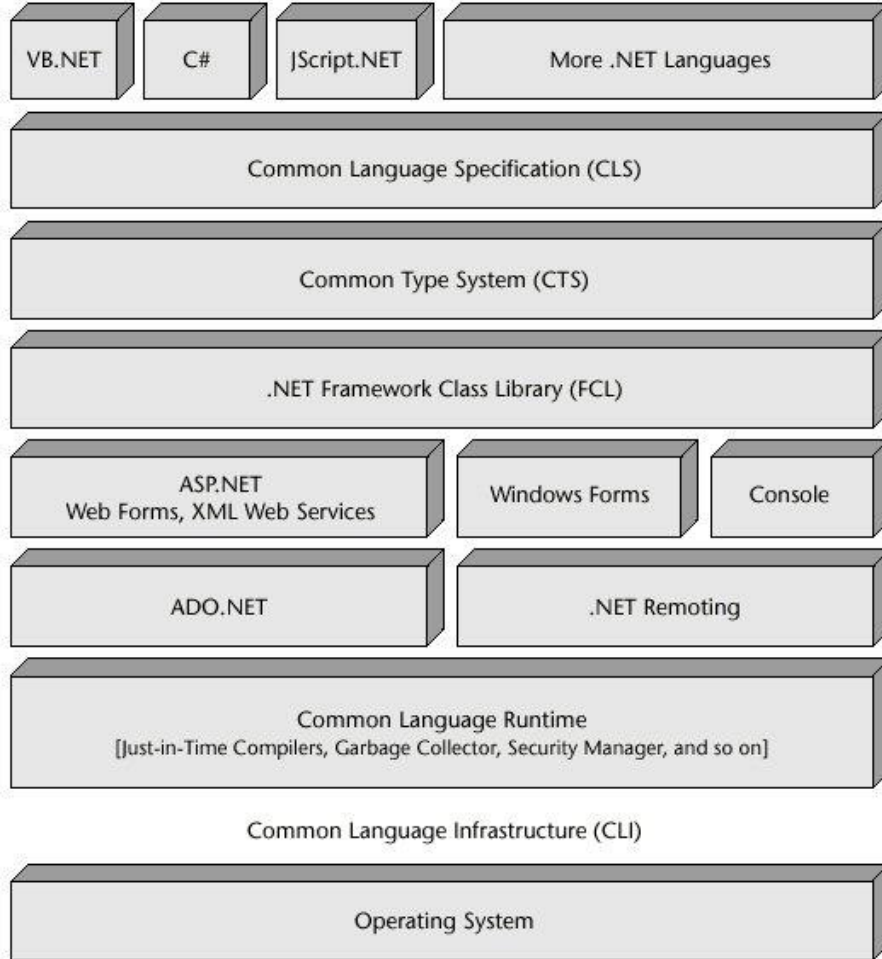
Java için	.NET için	PHP için
- Hibernate - JPA - OpenJPA - Toplink - EclipseLink - Apache Cayenne - MyBattis	- Entity Framework - Nhibernate - .Net Persistence - BBADataObjects- - DataObjects.NET - DotNorm - FastObjects.NET - Norm - OJB.NET	- Propel - Doctrine - PHP-Activerecord - PdoMap - RedBean

2.2. Entity Framework

Entitiy framework'den bahsetmeden önce .NET framework yapısını da kısaca incelemek gerekmektedir. .NET Framework alt seviye dillerden (Assembly, C, Fortran) farklı olarak Windows ile ara katman aracılığıyla iletişim kurarak, bizi detaylarla uğraştırmadan bir kabuk üzerinde geliştirme yapmamızı sağlayan bir yapı olarak görülmektedir. Son dönemde çok performans isteyen, makineye özel uygulamalar ve gömülü programlama haricinde,alt seviye diller kullanmak gerekmemektedir, bu nedenle bu tür frameworklerin performans kayıpları gözardı edildiğinden popülerliği daha artmıştır. Başka bir ifadeyle, bu frameworkle alt seviye olarak nitelendirilen dillerin bir çok detayından kurtulup daha hızlı ve hata oranı daha az olan geliştirmeler yapılabilen ortamlara olanak sağlanmıştır.

Şekil 2.1.'de görüldüğü üzere .NET ile Windows üzerine ara katman olarak CLR (Common Language Runtime) gelmektedir. Bu katman hata ayıklama, tip kontrolü, çalışma zamanı derleyicileri gibi kodun üzerinde çalışabileceği bir altyapı sunmaktadır, Framework'ün dosya, kelime, güvenlik, ağ, eş zamanlı işlemler, diziler gibi kolaylık

sağlayan altyapısı bir üst katmanda yazılımcıların kullanımına açıklamaktadır. Bunları kullanarak, geliştirilmiş veri ve XML altyapısı da yine yazılımcılara sunulmaktadır [6].



Şekil 2.1 Net Yapısı [5]

.NET platformunun özelliklerini destekleyen XML standardı üzerine kurulmuş bir veri erişim teknolojileri için ADO.NET karşımıza çıkmaktadır. ADO kısaltmasının açılımı ActiveX Data Object olup, veri tabanlarında verileri almak, işlemek, düzenlemek ve veri kaynağına yeni veriler kaydetmek için kullanılan sınıflardır. Bu teknoloji yazılım projelerinde kullanılmaya devam etmektedir. Diğer yandan son yıllarda verilere erişimde daha efektif sorgular yazma isteğinin artmasıyla bu teknoloji yerine nesnelere veri

tabanlarında çalışacak platformlara geçiş olmuştur. Microsoft'ta ilk kez .NET Framework 3.5 sürümüyle Entity Framework teknolojisini tanıtmıştır.

Entity Framework, Microsoftun .NET nesneleri kullanarak veri tabanıyla çalışmasına imkan tanıyan bir nesne ilişkisel eşleyicisidir. Özetle .Net tabanlı bir ORM aracıdır.

Bu yapı ile yazılımcıların veri tabanına erişim ve değişiklik için yazması gereken fazla kodu önleyen ilişkili verilerle onlar ile alana özgü nesneymiş gibi çalışmaya fırsat veren bir frameworktür. En basit şekilde şöyle özetlenebilir: veri tabanındaki tabloları sınıf yapılarına, kolonları özellikler, kayıtları ise nesnelere dönüştürerek uygulamanın veri tabanına erişmesine gerek kalmadan tüm veri tabanı işlemlerini gerçekleştirmesine yarayan yapıların oluşturulmasıdır. Böylece veri tabanı işlemlerinin Sql kodları yazmadan nesnelere üzerinden kolayca yapılmasının sağlanması amaç edinilmiştir [5]. Microsoft'un ilk kez .Net Framework 3.5 ile tanıttığı Entity Framework sürekli olarak gelişmeler göstererek daha kullanışlı ve verimli hale gelmiştir.Çizelge 2.2' de Entity Framework'ün sürüm özellikleri gösterilmektedir.

Çizelge 2.2 Entity Framework Sürüm Geliştirilen Özellik Listesi

Entity Framework Sürümü	Geliştirilen Özellik
EF 3.5	Database first yaklaşımı üzerinden temel ORM/m desteği
EF 4.0	POCO desteği, Lazy Loading, Model First yaklaşımı
EF 4.1	İlk kez NuGet üzerinden yüklenebilme, basit DbContext API ile objectcontext temelli geliştirmeye alternatif geliştirme olanağı – Bu sürümü 4.0 sürümüne iyileştirme olarak duyurulmuştur.
EF 4.3	Code First yaklaşımındaki iyileştirmeler. Code First yaklaşımı ile geliştirilen yapının üzerinden database oluşturabilme özelliği
EF 5.0	Enum Type kullanımı, DBGeography ve DBGeometry tipleri, DbContext ile code generation, Model Diyagramlarının bölünebilmesi, Design ekranında Entitylerin renkelendirilebilmesi, Birden Fazla Stored ProcEDURE seçerek ekleme, Table valued Function
EF 6.0	Task-based async, Code First Yaklaşımı ile stored procedure ve Function geliştirebilme

3. VERİ TABANLARI

3.1. Veri Tabanı

Veri tabanı en genel tanımıyla, kullanım amacına uygun olarak düzenlenmiş veriler topluluğudur. Birbirleriyle ilişkileri olan verilerin tutulduğu, mantıksal ve fiziksel olarak tanımlarının olduğu bilgi depolarıdır. Veri tabanları gerçekte var olan ve birbirleriyle ilişkisi olan nesnelere ve ilişkileri modellerdir. Veri tabanı yönetim sistemleri (VTYS) ise, verilere aynı anda birden çok bağlantı sağlayabilme özelliği sağlar. Bu sistemler, veri tabanı yönetiminin bir parçası olarak, verinin nasıl depolanacağı, kullanılacağı ve erişileceğini mantıksal olarak yönlendiren bir kurallar sistemidir [2] [3]

3.2. ADO.NET

ADO.NET veri erişim hizmetleri .NET Framework programcıları için kullanıma sınıflar kümesidir. ADO.NET, dağıtılmış, veri paylaşımı uygulamaları oluşturmak için zengin bir bileşen kümesi sağlar. İlişkisel erişimi, XML ve uygulama verileri sağlayan, .NET Framework ayrılmaz bir parçasıdır. ADO.NET geliştirme gereksinimi, ön uç veri tabanı istemciler ve uygulamalar, araçlar, dilleri veya Internet tarayıcıları tarafından kullanılan orta katman iş nesnelere oluşturulmasını da dahil olmak üzere destekler [7].

3.3. MSSQL

Microsoft firması tarafından yayınlanan ticari bir üründür. Farklı verisyonları bulunmaktadır. Farklı ölçekteki işletmelere ve projelere uygun olarak geliştirilmeye devam edilmektedir. Şu an için son sürümü Microsoft SQL Server 2017 dir. Windows işletim sistemi üzerinde çalışmaktadır. Microsoft SQL Server 2017 sürümünün Linux işletim sistemi desteği bulunmaktadır. [8]

3.4. MYSQL

MySql, 1995 yılında oluşturulmuştur. Açık kaynaklı ve ücretsiz olması sebebiyle firmalar ve yazılım geliştiriler tarafından kısa zamanda kullanımı yaygınlaşmıştır. PHP dili ile yazılım geliştiren geliştiriciler ve üniversiteler tarafından yaygın kullanılmaktadır. Windows işletim sistemi üzerinde kurulumu da bulunmaktadır. Fakat Linux işletim sistemlerinde daha performanslı çalışmaktadır. Birden fazla kullanıcı desteği, parçacıklı iş yapabilme özelliği ve alt sorgu desteği bulunmaktadır [9].

3.5. ORACLE

İlişkisel veri tabanları alanında en gelişmiş özelliklere sahip veri tabanıdır. Büyük veri merkezleri tarafından ve büyük şirketler tarafından tercih edilmektedir [10]. Son sürümü "Oracle Database 12c Release" dir. Windows ve Linux sistemleri üzerinde çalışabilmektedir.

Oracle veri tabanı SQL dilinin dışında PL/SQL dili kullanılmaktadır ve sadece Oracle veri tabanı için kullanılabilir. Özellikle yoğun veri işlemlerinin veri tabanında gerçekleştirilmesi uygulamaların daha performanslı çalışmasını sağlamaktadır. [11, p. 8]

3.6. POSTGRESQL

PostgreSQL'in geliştirilmesi; 1977 de Berkeley üniversitesinde Ingres adındaki ilişkisel veri tabanı geliştirilmeye başlanmıştır.1996 yılında SQL standartlarını desteklemesi nedeniyle kullanımı yaygınlaşmıştır. Açık kaynak kodlu ücretsiz bir veri tabanıdır [12]. Windows ve Linux sistemleri üzerinde çalışabilmektedir. PostgreSQL ilk sürümlerinde Windows desteği olmaması nedeniyle tercih edilmiyordu, Şubat 2003' de Windows versiyonun yayınlanması ile birlikte kullanımı yaygınlaşmaya başladı. [13]

4. METODOLOJİ

4.1. Entity Framework Performans Test (EFPT) Yazılımı

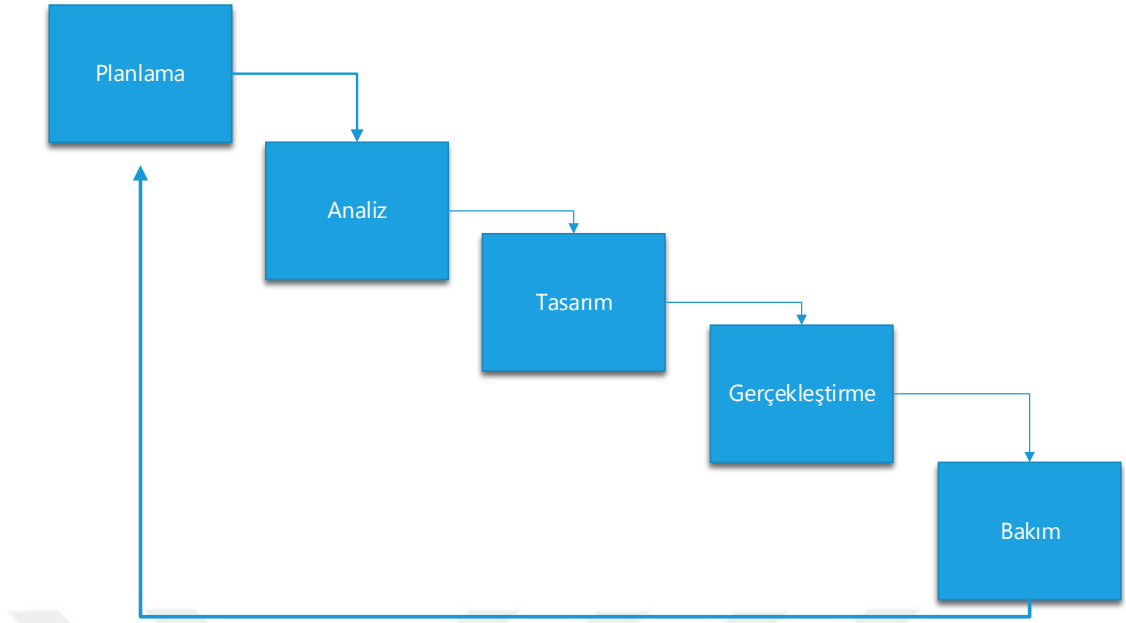
Çalışmada geliştirilen yazılım (EFPT), profesyonel bir yazılım geliştirme süreci takip edilerek tamamlanmıştır. Bu nedenle, yazılım geliştirme süreçlerini yönetmek için literatürde de sıkça bahsedilen Şelale Modeli tercih edilmiştir.

[14, p. 128] Tarafından Şelale Modeli şöyle tanımlanmaktadır:

“Klasik çevrim yöntemi çağlayan(waterfall), şelale modeli, büyük tasarım modeli(grand design) ya da geleneksel model olarak adlandırılır ve sistematik olarak ilerleyen ardışık bir yaklaşımla yazılım geliştirilmesini sağlar. Bu modelde, tüm kullanıcı gereksinimleri belirlenir, isterler tanımlanır, buna göre tasarım yapılır, gerçekleştirim sonunda tümleştirilir. Daha sonra, sistem sınanarak teslim edilir ve bakım aşaması başlatılır.”

Yazılım gibi esnek bir yapıda Şelale modelinin geri dönüşe izin vermeyen yapısı nedeniyle tasarım ve analiz sonrasında da değişikliklere ihtiyaç duyulması durumunda yetersiz olmaktadır. Bu durum yeni modellerin oluşmasını neden olmuştur [15].

Bu yazılım geliştirme modelinin 6 temel aşamasının gösterildiği grafik Şekil 4.1’de verilmektedir.



Şekil 4.1 Şelale Modeli Aşamaları

Bu aşamaları ve bu aşamalardaki yapılacak olan temel işlemler kısaca özetlersek;

Planlama

Planlama aşaması proje planının oluşturulması ve plan için gerekli fizibilitenin çıkarılması ve buna bağlı olarak personel ve donanım gereksinimlerin belirlendiği aşamadır. [16]

Analiz

Analiz aşamasında var olan işlemler ve süreçler incelenip sistem gereksinimlerin ayrıntılarının ve temel sorunların belirlendiği aşamadır [16]

Tasarım

Bu aşama da planlama ve analiz aşamasında belirlenen gereksinimlere uygun olarak temel yazılım sistemin oluşturulduğu aşamadır. Bu aşamada yazılımı oluşturan bileşenler ve bunların ayrıntıları ayrıca sistemin yapısı oluşturulur. [16]

Gerçekleştirime

Tasarıma uygun olarak uygulamanın kodlanması aşamasıdır. Bu aşamada kodlama yapılırken eş zamanlı olarak test işlemleri yapılır. Kodlamanın tamamlanması ile genel olarak sistem tümüyle test edilir. Bu testlerin başarıyla sonuçlanmasının ardından kurulum çalışmaları yapılır [16]

Bakım

Uygulamanın kurulumu ve teslimi sonrasında süreç içerisinde dönemselsel olarak oluşan ve belirlenen yeni ihtiyaçlara uygun olarak geliştirme yapıldığı aşamadır [16].

Bu araştırmada da EFPT Yazılımı Şelale Modeli yazılım süreci modeline uygun olarak geliştirilmiştir. Geliştirme süreci aşağıdaki aşamalardan oluşmaktadır.

4.1.1. Planlama

Bu aşamada uygulamanın amaçlarına uygun olarak ihtiyaç duyulan testler belirlenmiştir. Bu testler için ihtiyaç duyulan yazılım mimarisine karar verilmiştir. Yazılım mimarisi göz önüne alınarak yazılım dili ve kodlama araçlarına karar verilmiştir. Testlerin başarıyla uygulanabilmesi için yazılım süreci planlanmıştır.

4.1.2. Analiz

Bu aşamada uygulanacak testler incelendi. Testlerin gerçekleştirilebilmesi amacıyla öncelikli olarak donanımsal ihtiyaçlar belirlendi ve donanımsal ihtiyaç listesi oluşturuldu.

Ayrıca uygulama amacına yönelik olarak, istenen özellikler ve bu özellikler için ekran ara yüzlerinin ihtiyaçları oluşturuldu. Her test için farklı iş akışları belirlenerek

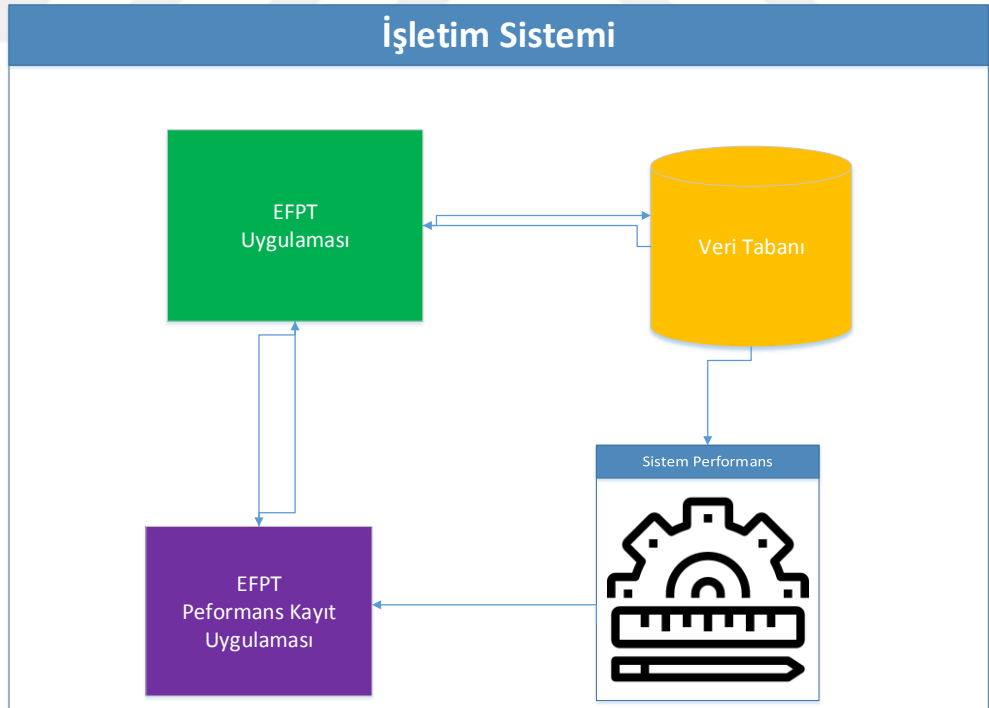
uygulama için gerekli tüm gereksinimler netleştirildi. Yazılım mimarisi göz önünde tutularak yazılım alt bileşen ve yazılım kütüphaneleri belirlendi.

4.1.3. Tasarım

Analiz aşamasında belirlenen test ekranları için tasarımlar yapıldı. Bu ekranlar test özelliğine göre farklılıklar içermektedir. Bazı test ekranları çok az bileşen ihtiyaç duyarken bazı ekranların birçok bileşene ihtiyaç duyduğu gözlemlendi. Ekran tasarımlarının kullanıcı dostu olması sağlandı.

4.1.4. Gerçekleştirme

Planlanan geliştirme sürecine göre analiz ve tasarım aşamasından sonra ekran tasarımlarına uygun olarak geliştirme ve iş akış süreçleri çıkarıldı, İş akışları göz önünde bulundurularak mantıksal aşamalar kodlandı.



Şekil 4.2 Genel Uygulama Yapısı

4.1.5. Test

Geliştirme süreci tamamlanan ekranlar ve iş akışları kontrol edildi. Test sonuçlarına uygun olmayan, sorun görülen yerler için test analizleri oluşturuldu.

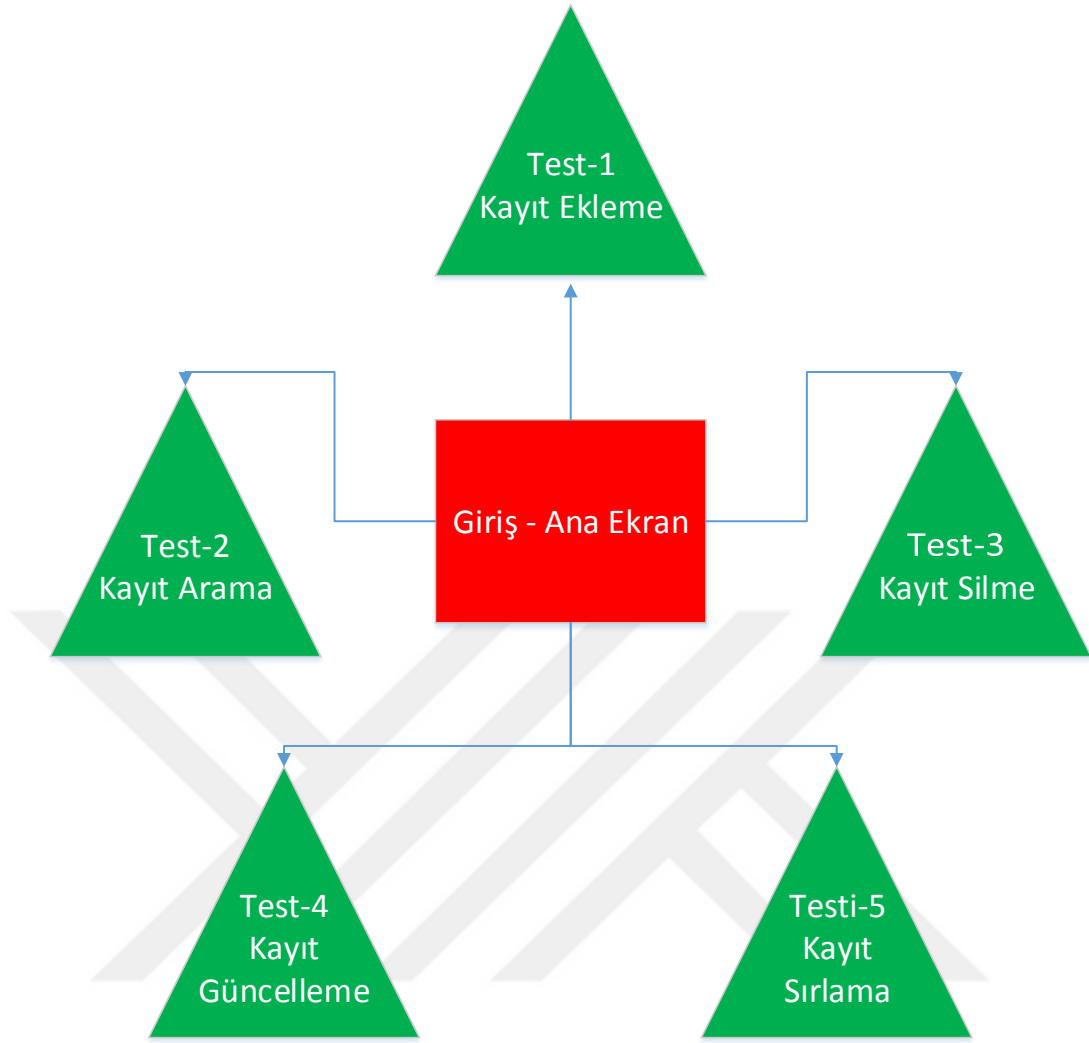
4.1.6. Bakım

Uygulama gerçekleştirme esnasında ihtiyaç duyulan geliştirmeler yapıldı.

4.2. Entity Framework Performans Testi Uygulaması

4.2.1. Ekran Tasarımlarına Genel Bakış

Uygulama bir (1) giriş ekranı ve beş (5) tane test ekranından olmak üzere toplam altı (6) adet ekrandan oluşmaktadır. Bu ekranların işlevleri ve yapılış amaçları aşağıda açıklanmaktadır. Ekran tasarım ve yönlendirme haritası Şekil 4.3'de gösterilmektedir.



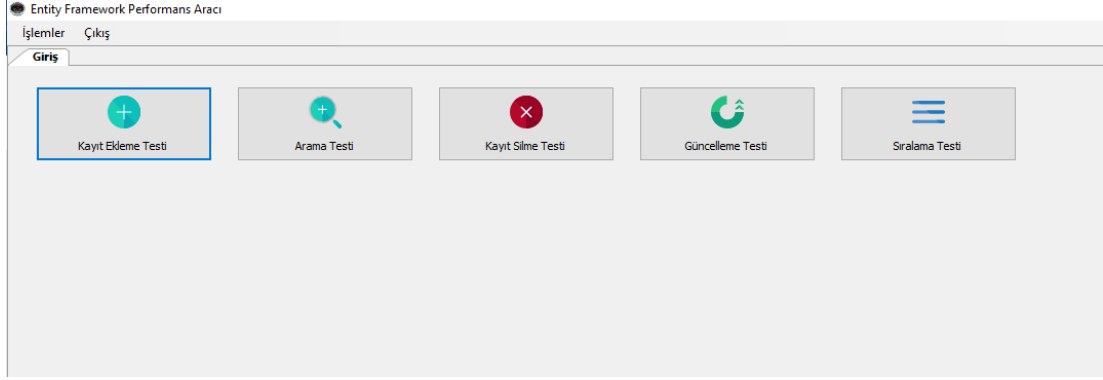
Şekil 4.3 Ekran Yerleşim Haritası

4.2.1.1. Giriş-Ana Ekran

Uygulama ilk açılışında kullanıcının karşılaştığı ekrandır. Kullanıcının yapılacak testlere ulaşabilmesi amacıyla tasarlanmıştır. Bu giriş ekranı ile sistem üzerinde uygulanan aşağıdaki testlere ulaşılmaktadır.

1. *Kayıt Ekleme Testi*
2. *Kayıt Arama Testi*
3. *Kayıt Silme Testi*
4. *Kayıt Güncelleme Testi*
5. *Kayıt Sıralama Testi*

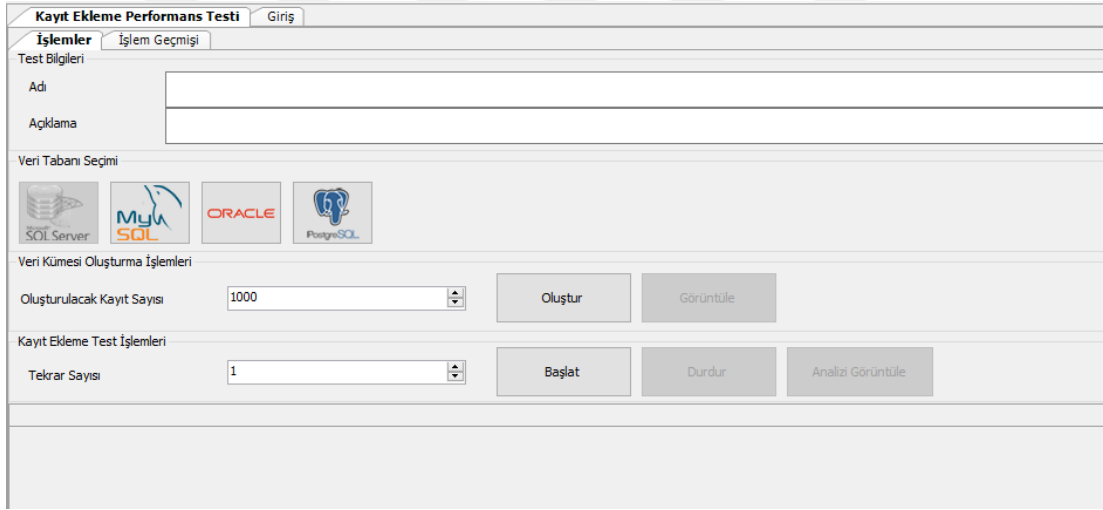
Ekranın tasarım görüntüsü Şekil 4.4’ de gösterilmektedir.



Şekil 4.4 Giriş Ekran Görüntüsü

4.2.1.2. Kayıt Ekleme Test Ekranı

Bu ekran; kayıt ekleme testinin gerçekleştirebilmek amacıyla tasarlanmıştır. Kayıt Ekleme Testi Ekranı Şekil 4.5’de gösterilmektedir.



Şekil 4.5 Kayıt Ekleme Testi Ekranı

Ekran üzerinde yapılacak testi tanımlayıcı bilgilerin girildiği bilgi alanları ve test kontrol nesneleri bulunmaktadır. Ekran kontrolleri ve tanım bilgileri Çizelge 4.1’de açıklanmaktadır.

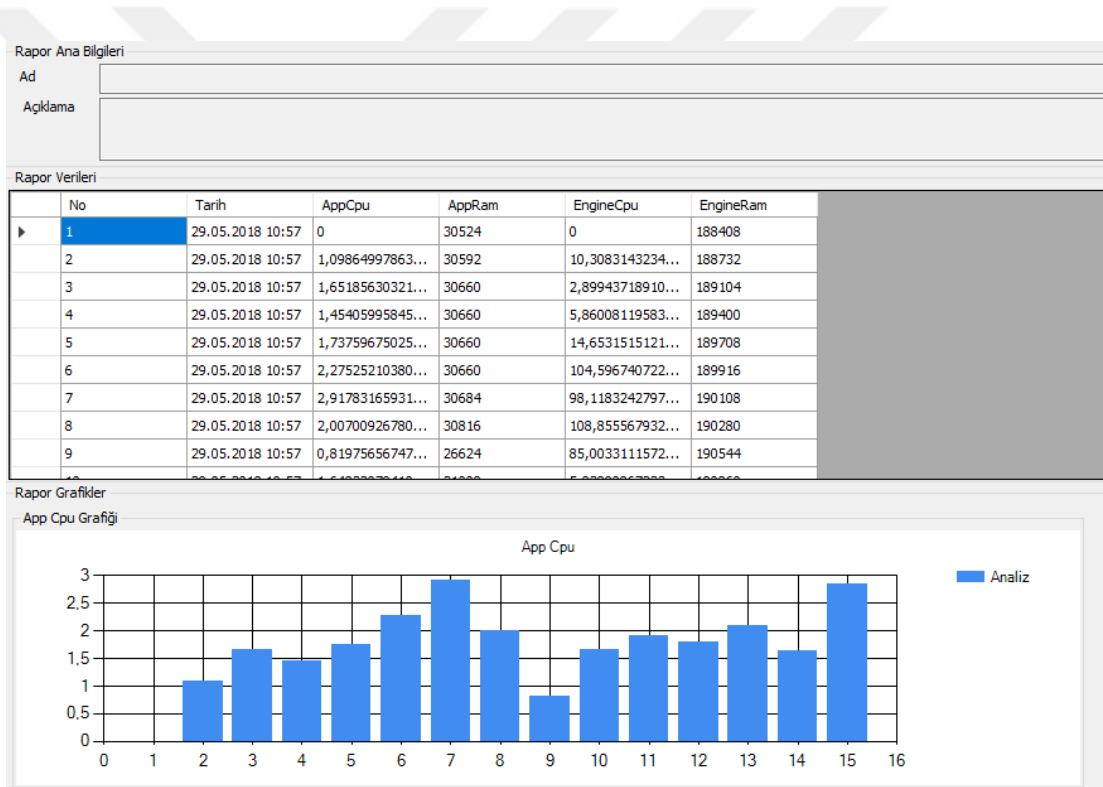
Çizelge 4.1 Kayıt Ekleme Testi Ekranı Kontrol Listesi

Kontrol Adı	Türü	Amaç-Görev
Ad	Veri Giriş Alanı	Testin tanımlanabilmesi amacıyla kullanılmaktadır. Testin isim bilgisini içermektedir.
Açıklama	Veri Giriş Alanı	Testin açıklama bilgisini ve hangi amaçla oluşturulduğu gibi bilgileri tutulması ve testin ayrıntılı bilgilerini içermektedir.
Veri Tabanı Seçimi	Seçim grup düğme Listesi	Testin hangi veri tabanına uygulanacağını belirlemek.
Kayıt Sayısı	Veri Giriş Alanı	Veri kümesi oluşturma işlemleri altında eklenecek kayıt sayısının belirlemek.
Oluştur	Düğme	Eklenecek kayıtların oluşturulmasını sağlar.
Görüntüle	Düğme	Oluşturulan eklenecek kayıtların ekranda listelenmesini sağlar. Oluşturulan örnek verilerin ekran görüntüsü Şekil 4.6'da gösterilmektedir.
Tekrar Sayısı	Veri Giriş Alanı	İşlemin kaç kez tekrar edileceği belirlenebilmesini sağlar.
<u>Başlat</u>	Düğme	Testin başlamasını sağlar ve test başladığında pasif olur. Test bitiminde tekrar aktif olacaktır.
<u>Durdur</u>	Düğme	Test bitmeden testi durdurmaya sağlamaktadır.
<u>Analizi Görüntüle</u>	Düğme	Test Sonrasında oluşan analizin görüntülenmesini sağlar. Analiz ekran görüntüsü Şekil 4.7'de gösterilmektedir.

Test Verileri

	Ad	Soyad	Mail
▶	Ad 0	Soyad 0	mail0@mail.com
	Ad 1	Soyad 1	mail1@mail.com
	Ad 2	Soyad 2	mail2@mail.com
	Ad 3	Soyad 3	mail3@mail.com
	Ad 4	Soyad 4	mail4@mail.com
	Ad 5	Soyad 5	mail5@mail.com
	Ad 6	Soyad 6	mail6@mail.com
	Ad 7	Soyad 7	mail7@mail.com

Şekil 4.6 Kayıt Ekleme Testi Örnek Veriler Ekranı



Şekil 4.7 Analiz Rapor Ekranı

4.2.1.3. Arama Testi Ekranı

Bu ekran kayıt arama testinin gerçekleştirebilmek amacıyla tasarlanmıştır. Ekran üzerinde yapılacak testi tanımlayıcı bilgilerin girildiği bilgi alanları ve test kontrol nesnelere bulunmaktadır. Ekran kontrolleri ve tanım bilgileri Çizelge 4.2’de açıklanmaktadır.

Çizelge 4.2 Kayıt Arama Testi Ekranı Kontrol Listesi

Kontrol Adı	Türü	Amaç-Görev
Ad	Veri Giriş Alanı	Testin tanımlanabilmesi amacıyla kullanılmaktadır. Testin isim bilgisini içermektedir.
Açıklama	Veri Giriş Alanı	Testin açıklama bilgisini ve hangi amaçla oluşturulduğu gibi bilgileri tutulması ve testin ayrıntılı bilgilerini içermektedir.
Veri Tabanı Seçimi	Seçim grup düğme Listesi	Testin hangi veri tabanına uygulanacağını belirlemek.
Mevcut Verileri Görüntüle	Düğme	Belirlenen veri tabanı üzerindeki mevcut verilerin listelenmesini sağlar.
Aranacak Alan	Liste Seçim Kutusu	Aramanın yapılacağı alanın belirlenmesini sağlar. Bu alanlar model nesnesinin tüm özelliklerini içermektedir. Model özellik listesi Şekil 4.8’de gösterilmektedir.
Arama Şekli	Liste Seçim Kutusu	Aramanın yapılaş kriteri belirlenmektedir. Bu alanlar tüm arama türlerini içermektedir. Arama özellik listesi Şekil 4.9’da gösterilmektedir.
<u>Arama Metni</u>	Veri Giriş Alanı	Aranacak ifadenin girişini sağlar.

Kontrol Adı	Türü	Amaç-Görev
<u>Başlat</u>	Düğme	Testin başlamasını sağlar ve test başladığında pasif olur. Test bitiminde tekrar aktif olacaktır.
<u>Durdur</u>	Düğme	Test bitmeden testi durdurmayı sağlamaktadır.
<u>Arama Sonuç Listesi</u>	Liste	Aramanın sonuçların görüntülenmesini sağlar. Arama sonuç ekran görüntüsü Şekil 4.10'da gösterilmektedir.
<u>Analizi Görüntüle</u>	Düğme	Test Sonrasında oluşan analizin görüntülenmesini sağlar. Analiz ekran görüntüsü Şekil 4.7'de gösterilmektedir.

Kayıt Arama Test İşlemleri

Aranacak Alan	Ad
Arama Şekli	Ad
Arama Metni	Soyad
	Mail
	Telefon
	Adres
Tekrar Sayısı	İl
	İlçe
	No
Arama Sonuç	Tarih

Şekil 4.8 Model Özellik Listesi

Kayıt Arama Test İşlemleri

Aranacak Alan Ad

Arama Şekli Eşittir

Arama Metni Eşittir
Eşit Değil
Büyüktür
Küçüktür
Küçük ve Eşittir
Büyük ve Eşittir
İle Başlayan
İle Biten
İçeren
İçermeyen

Tekrar Sayısı

Arama Sonuç

Şekil 4.9 Arama Kriter Listesi

Arama Sonuçları									
	Ad	Soyad	Mail	Telefon	Adres	İl	İlçe	No	Tarih
▶	Ad 0	Soyad 0	mail0@mail.com					1	29.05.2018 10:56
	Ad 1	Soyad 1	mail1@mail.com					2	29.05.2018 10:56
	Ad 2	Soyad 2	mail2@mail.com					3	29.05.2018 10:56
	Ad 3	Soyad 3	mail3@mail.com					4	29.05.2018 10:56
	Ad 4	Soyad 4	mail4@mail.com					5	29.05.2018 10:56
	Ad 5	Soyad 5	mail5@mail.com					6	29.05.2018 10:56

Şekil 4.10 Arama Sonuç Listesi

4.2.1.4. Kayıt Silme Test Ekranı

Bu ekran; kayıt silme testinin gerçekleştirilebilmesi amacıyla tasarlanmıştır. Ekran üzerinde yapılacak testi tanımlayıcı bilgilerin girildiği bilgi alanları ve test kontrol nesnelere bulunmaktadır. Ekran kontrolleri ve tanım bilgileri Çizelge 4.3’de açıklanmaktadır.

Çizelge 4.3 Kayıt Silme Testi Ekranı Kontrol Listesi

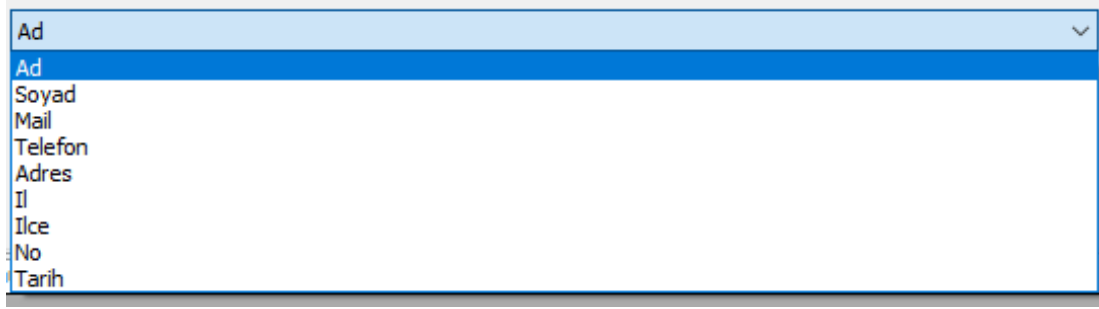
Kontrol Adı	Türü	Amaç-Görev
Ad	Veri Giriş Alanı	Testin tanımlanabilmesi amacıyla kullanılmaktadır. Testin isim bilgisini içermektedir.
Açıklama	Veri Giriş Alanı	Testin açıklama bilgisini ve hangi amaçla oluşturulduğu gibi bilgileri tutulması ve testin ayrıntılı bilgilerini içermektedir.
Veri Tabanı Seçimi	Seçim grup düğme Listesi	Testin hangi veri tabanına uygulanacağını belirlemek.
Mevcut Verileri Görüntüle	Düğme	Belirlenen veri tabanı üzerindeki mevcut verilerin listelenmesini sağlar.
<u>Başlat</u>	Düğme	Testin başlamasını sağlar ve test başladığında pasif olur. Test bitiminde tekrar aktif olacaktır.
<u>Durdur</u>	Düğme	Test bitmeden testi durdurmaya sağlamaktadır.
<u>Analizi Görüntüle</u>	Düğme	Test Sonrasında oluşan analizin görüntülenmesini sağlar.

4.2.1.5. Güncelleme Testi Ekranı

Bu ekran; kayıt güncelleme testini gerçekleştirebilmek amacıyla tasarlanmıştır. Ekran üzerinde yapılacak testi tanımlayıcı bilgilerin girildiği bilgi alanları ve test kontrol nesnelere bulunmaktadır. Ekran kontrolleri ve tanım bilgileri Çizelge 4.4'de açıklanmaktadır.

Çizelge 4.4 Kayıt Güncelleme Testi Ekranı Kontrol Listesi

Kontrol Adı	Türü	Amaç-Görev
Ad	Veri Giriş Alanı	Testin tanımlanabilmesi amacıyla kullanılmaktadır. Testin isim bilgisini içermektedir.
Açıklama	Veri Giriş Alanı	Testin açıklama bilgisini ve hangi amaçla oluşturulduğu gibi bilgileri tutulması ve testin ayrıntılı bilgilerini içermektedir.
Veri Tabanı Seçimi	Seçim grup düğme Listesi	Testin hangi veri tabanına uygulanacağını belirlemek.
Mevcut Verileri Görüntüle	Düğme	Belirlenen veri tabanı üzerindeki mevcut verilerin listelenmesini sağlar.
Güncellenecek Alan	Seçim Listesi	Güncelleme işleminde kullanılacak alan bilgisinin belirlenmesini sağlar. Seçim için kullanılacak model alan listesi Şekil 4.11’ de verilmiştir.
Güncelleme Metni	Veri Giriş Alanı	Güncellenecek alan için yeni metnin belirlenmesini sağlar.
Başlat	Düğme	Testin başlamasını sağlar ve test başladığında pasif olur. Test bitiminde tekrar aktif olacaktır.
Durdur	Düğme	Test bitmeden testi durdurmayı sağlamaktadır.
Analizi Görüntüle	Düğme	Test Sonrasında oluşan analizin görüntülenmesini sağlar. Analiz ekran görüntüsü Şekil 4.7’da gösterilmektedir.



Şekil 4.11 Güncelleme Model Alan Listesi

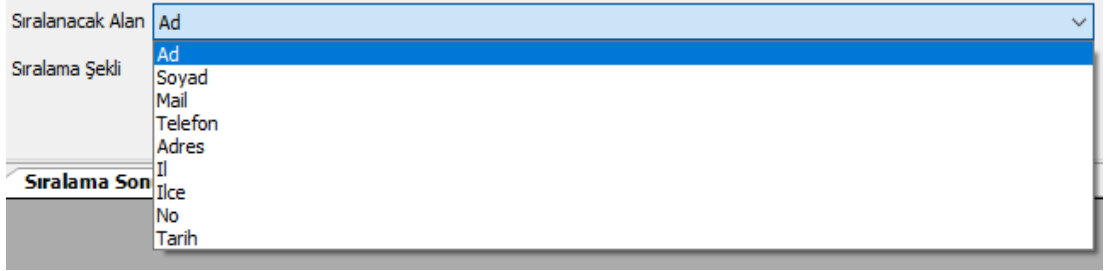
4.2.1.6. Sırlama Testi Ekranı

Bu ekran; kayıt sıralama testini gerçekleştirebilmek amacıyla tasarlanmıştır. Ekran üzerinde yapılacak testi tanımlayıcı bilgilerin girildiği bilgi alanları ve test kontrol nesnelere bulunmaktadır. Ekran kontrolleri ve tanım bilgileri Çizelge 4.6'da açıklanmaktadır.

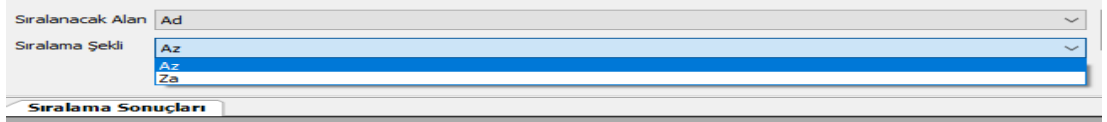
Çizelge 4.6 Kayıt Sıralama Testi Ekranı Kontrol Listesi

Kontrol Adı	Türü	Amaç-Görev
Ad	Veri Giriş Alanı	Testin tanımlanabilmesi amacıyla kullanılmaktadır. Testin isim bilgisini içermektedir.
Açıklama	Veri Giriş Alanı	Testin açıklama bilgisini ve hangi amaçla oluşturulduğu gibi bilgileri tutulması ve testin ayrıntılı bilgilerini içermektedir.
Veri Tabanı Seçimi	Seçim grup düğme Listesi	Testin hangi veri tabanına uygulanacağını belirlemek.
Mevcut Verileri Görüntüle	Düğme	Belirlenen veri tabanı üzerindeki mevcut verilerin listelenmesini sağlar

Kontrol Adı	Türü	Amaç-Görev
Sıralanacak Alan	Seçim Listesi	Sırlamanın veri modelindeki hangi alan üzerinde yapılacağını belirler. Model alan listesi Şekil 4.12’de verilmiştir.
Sıralama Şekli	Seçim Listesi	Sıralamanın yönünü belirlenmesini sağlar. Sırlama yönü seçenek listesi Şekil 4.13’de verilmiştir.
Başlat	Düğme	Testin başlamasını sağlar ve test başladığında pasif olur. Test bitiminde tekrar aktif olacaktır.
Durdur	Düğme	Test bitmeden testi durdurmayı sağlamaktadır.
Analizi Görüntüle	Düğme	Test Sonrasında oluşan analizin görüntülenmesini sağlar.



Şekil 4.12 Sıralama Model Alan Listesi



Şekil 4.13 Sıralama Yönü Seçenek Listesi

4.2.2. Analiz Testlerin Uygulanışı

Analiz testleri hazırlanan Entity Framework Performans Testi uygulaması ile test edilmiştir. Testler Çizelge 4.4’de belirtilen iş istasyonu bilgisayar üzerinde yapılmıştır. Testlerin doğru sonuçlara ulaşması amacıyla, test uygulamasından bağımsız olarak Performans Kayıt Uygulaması geliştirilmiştir. Bu uygulama ile test uygulaması sadece test başlamadan önce ve test bitiminde iletişime geçerek verilerin doğru olarak toplanmasını garanti etmektedir.

Çizelge 4.4 Test İş İstasyonu Özellik Listesi

Özellik	Değer
Bellek Hızı (Mhz):	2400
Bellek Kapasitesi:	32 GB
Bellek Tipi:	DDR4
Ekran Kartı Bellek Kapasitesi (GB):	4 GB
Güç Kaynağı (Watt):	650
Harddisk RPM:	7200
Harddisk Tipi:	PCIe SSD + SATA
SATA Harddisk Kapasitesi (GB):	2 TB
SSD Harddisk Kapasitesi (GB):	256 GB
Ürün Tipi:	Masaüstü İş İstasyonu
İşlemci Hızı (Ghz):	2.1
İşlemci Markası:	INTEL
İşlemci Modeli:	Xeon E5
İşletim Sistemi:	Windows 10 Professional

4.2.2.1. Kayıt Ekleme Testi

Amaç:

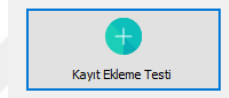
Bu test ile veri tabanlarının kayıt ekleme esnasındaki performansları analiz edilerek kayıt altına alınmıştır.

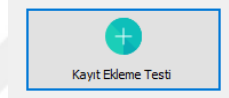
Uygulanışı:

25.000 kayıt ile kayıt ekleme testi yapılmıştır. Uygulanacak test verileri uygulanacak veri tabanlarının tümü için aynı olacak şekilde oluşturulmuştur. Veriler users model nesnesi örneğinden anlık rastgele oluşturulmuştur. Testin doğruluğunu garanti etmek amacıyla veri tabanı seçimi dışında diğer etkenlerin aynı olması sağlanmıştır.

Uygulanış adımları aşağıda listelenmiştir.

1. Uygulama başlatıldı.



2. Giriş ekranı üzerinden  düğmesine basılarak Kayıt ekleme Testi Ekranı açıldı.
3. Ekran üzerindeki “**Adı**” bilgi giriş alanına testi açıklayıcı ad tanımlama bilgisi girildi.
4. Ekran üzerindeki “**Açıklama**” alanına testin ne amaçla yapıldığını bildiren bilgi amaçlı bir açıklama bilgisi girildi.
5. Ekran üzerindeki “**Veri Tabanı Seçimi düğme seçim grubu**” butonlarından testin uygulanmasını istediğimiz veri tabanı belirlendi. Bu adım testin en önemli adımını oluşturmaktadır. Bu nedenle dikkatli olunması gerekmektedir.
6. Veri kümesi oluşturma işlemleri menüsü altından test için kullanılacak örnek veriler oluşturuldu. Örnek veriler oluşturulması için oluşturulacak “**kayıt sayısı alanına**” 25.000 yazıldı ve “**Oluştur**” düğmesi ile veriler oluşması sağlandı. Görüntüle düğmesine basılarak verilerin oluştuğu kontrol edildi. Bu veriler Ram üzerinde oluşturuldu.
7. Kayıt Ekleme Test İşlemleri menüsü altındaki “**Başlat**” düğmesine basılarak test başlatıldı.

8. Testin bitmesi beklendi.
9. Test bitimi ile “**Analiz Görüntüle**” düğmesine basılarak test sonuçları rapor ekranına ulaşıldı.
10. Bu ekran üzerindeki “**Excel’ Aktar**” düğmesi ile raporun Microsoft Excel ortamına veri ve grafik olarak aktarılması sağlandı.

Sonuç:

Testi Microsoft Sql Server, Mysql, Oracle, Postgresql veri tabanları üzerinde uygulanarak performans analiz veri ve grafikleri oluşturuldu.

4.2.2.2. Kayıt Arama Testi

Amaç:

Bu test ile veri tabanı üzerinde kayıt arama anındaki performansları analiz edilerek kayıt altına alınmıştır.

Uygulanışı:

Kayıt arama testi daha önce mevcut veriler üzerinde gerçekleştirilmiştir. Test 1.000.000 kayıt üzerinde yapılmıştır. Testin arama kriteri olarak users modeli üzerindeki ad ve soyad özellikleri seçilmiştir. Arama şekli olarak, eşittir, eşit değildir, içerir, içermez olarak testler yapılmıştır. Testler Microsoft Sql Server, Mysql, Oracle ve PostgreSQL veri tabanları üzerinde yapılmıştır.

Uygulanış adımları aşağıda listelenmiştir.

1. Uygulama başlatıldı.
2. Giriş ekranı üzerinden “Arama Testi” düğmesine basılarak Kayıt Arama Testi Ekranı açıldı.

3. Ekran üzerindeki “**Adı**” bilgi giriş alanına testi açıklayıcı ad tanımlama bilgisi girildi.
4. Ekran üzerindeki “**Açıklama**” alanına testin ne amaçla yapıldığını bildiren bilgi amaçlı bir açıklama bilgisi girildi.
5. Ekran üzerindeki “**Veri Tabanı Seçimi düğme seçim grubu**” butonlarından testin uygulanmasını istediğimiz veri tabanı belirlendi. Bu adım testin en önemli adımını oluşturmaktadır. Bu nedenle dikkatli olunması gerekmektedir.
6. Aranacak Alan seçim listesinden aramanın yapılacağı alan olan ad alanı seçildi. Ad alanı için test bitimi ile bu adımda soyad alanı seçilerek test tekrar uygulandı.
7. Arama şekli seçim kutusundan aramanın şekli eşittir seçildi. Eşittir şeklindeki aramanın bitimi ile arama şekli seçimi eşittir, eşit değildir, içerir, içermez, arama şekilleri içinde tekrarlandı.
8. Arama Metni bilgi girişi alanına aranacak metin girişi yapıldı.
9. “**Başlat**” düğmesine basılarak test başlatıldı.
10. Test bitimi ile “**Analiz Görüntüle**” düğmesine basılarak test sonuçları rapor ekranına ulaşıldı.
11. Bu ekran üzerindeki “**Excel’ Aktar**” düğmesi ile raporun Microsoft Excel ortamına veri ve grafik olarak aktarılması sağlandı.

Sonuç:

Testi Microsoft Sql Server, Mysql, Oracle, Postgresql veri tabanları üzerinde uygulanarak performans analiz veri ve grafikleri oluşturuldu.

4.2.2.3. Kayıt Silme Testi

Amaç:

Bu test ile veri tabanı üzerinde kayıt silme anındaki performansları analiz edilerek kayıt altına alınmıştır.

Uygulanışı:

Kayıt silme testi daha önce mevcut veriler üzerinde gerçekleştirilmiştir. Mevcut veriler daha önceki kayıt ekleme testinde eklenen kayıtlardır. Test 25.000 kayıt üzerinde yapılmıştır. Tüm kayıtların tek tek silinmesi şeklinde uygulanmıştır. Testler Microsoft Sql Server, Mysql, Oracle ve PostgreSQL veri tabanları üzerinde yapılmıştır.

Uygulanış adımları aşağıda listelenmiştir.

1. Uygulama başlatıldı.
2. Giriş ekranı üzerinden “Silme Testi” düğmesine basılarak Kayıt Silme Testi Ekranı açıldı.
3. Ekran üzerindeki “**Adı**” bilgi giriş alanına testi açıklayıcı ad tanımlama bilgisi girildi.
4. Ekran üzerindeki “**Açıklama**” alanına testin ne amaçla yapıldığını bildiren bilgi amaçlı bir açıklama bilgisi girildi.
5. Ekran üzerindeki “**Veri Tabanı Seçimi düğme seçim grubu**” butonlarından testin uygulanmasını istediğimiz veri tabanı belirlendi. Bu adım testin en önemli adımını oluşturmaktadır. Bu nedenle dikkatli olunması gerekmektedir.
6. Mevcut Verileri Görüntüle düğmesi ile ilgili veri tabanındaki tüm kayıtlar listelenerek kayıt sayısı ve silinecek veriler doğrulandı.
7. “**Başlat**” düğmesine basılarak test başlatıldı.
8. Test bitimi ile “**Analiz Görüntüle**” düğmesine basılarak test sonuçları rapor ekranına ulaşıldı.
9. Bu ekran üzerindeki “**Excel’ Aktar**” düğmesi ile raporun Microsoft Excel ortamına veri ve grafik olarak aktarılması sağlandı.

Sonuç:

Testi Microsoft Sql Server, Mysql, Oracle, Postgresql veri tabanları üzerinde uygulanarak performans analiz veri ve grafikleri oluşturuldu.

4.2.2.4. Kayıt Güncelleme Testi

Amaç:

Bu test ile veri tabanı üzerinde kayıt güncelleme anındaki performansları analiz edilerek kayıt altına alınmıştır.

Uygulanışı:

Kayıt güncelleme testi daha önce mevcut veriler üzerinde gerçekleştirilmiştir. Mevcut veriler daha önceki kayıt ekleme testinde eklenen kayıtlardır. Test 25.000 kayıt üzerinde yapılmıştır. Testin güncelleme kriteri olarak users modeli üzerindeki ad ve soyad özellikleri seçilmiştir. Testler Microsoft Sql Server, Mysql, Oracle ve PostgreSQL veri tabanları üzerinde yapılmıştır.

Uygulanış adımları aşağıda listelenmiştir.

1. Uygulama başlatıldı.
2. Giriş ekranı üzerinden “Güncelleme Testi” düğmesine basılarak Kayıt Güncelleme Testi Ekranı açıldı.
3. Ekran üzerindeki “**Adı**” bilgi giriş alanına testi açıklayıcı ad tanımlama bilgisi girildi.
4. Ekran üzerindeki “**Açıklama**” alanına testin ne amaçla yapıldığını bildiren bilgi amaçlı bir açıklama bilgisi girildi.
5. Ekran üzerindeki “**Veri Tabanı Seçimi düğme seçim grubu**” butonlarından testin uygulanmasını istediğimiz veri tabanı belirlendi. Bu adım testin en önemli adımını oluşturmaktadır. Bu nedenle dikkatli olunması gerekmektedir.
6. Güncellenecek Alan seçim listesinden güncellemenin yapılacağı alan olan ad alanı seçildi.
7. Güncelleme Metni bilgi girişi alanına metin girişi yapıldı.
8. “**Başlat**” düğmesine basılarak test başlatıldı.
9. Test bitimi ile “**Analiz Görüntüle**” düğmesine basılarak test sonuçları rapor ekranına ulaşıldı.
10. Bu ekran üzerindeki “**Excel’ Aktar**” düğmesi ile raporun Microsoft Excel ortamına veri ve grafik olarak aktarılması sağlandı.

Sonuç:

Testi Microsoft Sql Server, Mysql, Oracle, Postgresql veri tabanları üzerinde uygulanarak performans analiz veri ve grafikleri oluşturuldu.

4.2.2.5. Kayıt Sıralama Testi

Amaç:

Bu test ile veri tabanı üzerinde kayıt sıralama ve listeleme anındaki performansları analiz edilerek kayıt altına alınmıştır.

Uygulanışı:

Kayıt sıralama testi daha önce mevcut veriler üzerinde gerçekleştirilmiştir. Test 1.000.000 kayıt üzerinde yapılmıştır. Testin sıralama özelliği olarak users modeli üzerindeki ad ve soyad özellikleri seçilmiştir. Sıralama şekli olarak, AZ, ZA olarak testler yapılmıştır. Testler Microsoft Sql Server, Mysql, Oracle ve PostgreSQL veri tabanları üzerinde yapılmıştır.

Uygulanış adımları aşağıda listelenmiştir.

1. Uygulama başlatıldı.
2. Giriş ekranı üzerinden “Sıralama Testi” düğmesine basılarak Kayıt Sıralama Testi Ekranı açıldı.
3. Ekran üzerindeki “**Adı**” bilgi giriş alanına testi açıklayıcı ad tanımlama bilgisi girildi.
4. Ekran üzerindeki “**Açıklama**” alanına testin ne amaçla yapıldığını bildiren bilgi amaçlı bir açıklama bilgisi girildi.
5. Ekran üzerindeki “**Veri Tabanı Seçimi düğme seçim grubu**” butonlarından testin uygulanmasını istediğimiz veri tabanı belirlendi. Bu adım testin en önemli adımını oluşturmaktadır. Bu nedenle dikkatli olunması gerekmektedir.
6. Sıralanacak Alan seçim listesinden Sıralamanın yapılacağı alan olan ad alanı seçildi. Ad alanı için test bitimi ile bu adımda soyad alanı seçilerek test tekrar uygulandı.

7. Sıralama şekli seçim kutusundan sıralamanın şekli “Az” seçildi. A’dan Z’ye şeklindeki aramanın bitimi ile sıralamanın şekli “Za” sıralama şekilleri içinde tekrarlandı.
8. “**Başlat**” düğmesine basılarak test başlatıldı.
9. Test bitimi ile “**Analiz Görüntüle**” düğmesine basılarak test sonuçları rapor ekranına ulaşıldı.
10. Bu ekran üzerindeki “**Excel’ Aktar**” düğmesi ile raporun Microsoft Excel ortamına veri ve grafik olarak aktarılması sağlandı.

Sonuç:

Testi Microsoft Sql Server, Mysql, Oracle, Postgresql veri tabanları üzerinde uygulanarak performans analiz veri ve grafikleri oluşturuldu.

4.3. Performans Kayıt Uygulaması

Bu uygulama Entity Framework Performans Test Uygulaması’nda uygulanan testleri gerçek zamanlı olarak ölçme, verileri kayıt altına alma ve raporlama amacıyla geliştirilmiştir. Testlerin doğru olarak anlamlandırılması önem arz etmektedir. Bu uygulama ile test ölçmede ve anlamlandırmada önem arz eden çalışma zamanında oluşan uygulama ram, uygulama işlemci, veri tabanı işlemci ve veri tabanı ram değerlerini kayıt altına alınması sağlanmaktadır.

Ölçüm işleminin sadece ölçme testi esnasında başlaması ve test bitimi ile sonlandırılmasını sağlamak ölçmenin doğru olması için önemlidir.

Ayrıca alınan kayıtların ana uygulamaya gönderilmesi ve bu görevin başarıyla sonuçlandırılmasından sorumludur.

Bu uygulamaya, Performans Analiz Uygulamasının çalışma aşamasındaki değerlerinin testleri etkilemesi, test sonuçları için doğru olmayan sonuçlara neden olmasından dolayı ihtiyaç duyulmuş ve bu sorunu gidermek amacıyla hazırlanmıştır.

5. BULGULAR VE DEĞERLENDİRME

5.1. Testlere Genel Bakış

Oluşturulan test ortamı üzerinde testler başarıyla tamamlanmıştır. Toplam olarak 5 kategori altında 20 adet test yapılmıştır. Uygulanan testlerin listesi Çizelge 5.1’de verilmiştir.

Çizelge 5.1 Test Listesi

Kategori	Veri Tabanı	Kayıt Sayısı	Adı	Açıklama
Kayıt Ekleme Testi	MSSQL	25.000	MSSQL Kayıt Ekleme Testi	Bu test MSSQL veri tabanı üzerinde 25.000 kayıt ile yapılmıştır.
Kayıt Ekleme Testi	MYSQL	25.000	MYSQL Kayıt Ekleme Testi	Bu test MYSQL veri tabanı üzerinde 25.000 kayıt ile yapılmıştır.
Kayıt Ekleme Testi	ORACLE	25.000	ORACLE Kayıt Ekleme Testi	Bu test ORACLE veri tabanı üzerinde 25.000 kayıt ile yapılmıştır.
Kayıt Ekleme Testi	POSTGRESQL	25.000	POSTGRESQL Kayıt Ekleme Testi	Bu test POSTGRESQL veri tabanı üzerinde 25.000

Kategori	Veri Tabanı	Kayıt Sayısı	Adı	Açıklama
				kayıt ile yapılmıştır.
Kayıt Sıralama Testi	MSSQL	1.000.000	MSSQL Kayıt Sıralama Testi Ad Alanı Az	Bu test MSSQL veri tabanı üzerinde 1.000.000 kayıt ile Ad Alanı üzerinde Az şeklinde sıralanarak yapılmıştır.
Kayıt Sıralama Testi	MYSQL	1.000.000	MYSQL Kayıt Sıralama Testi Ad Alanı Az	Bu test MYSQL veri tabanı üzerinde 1.000.000 kayıt ile Ad Alanı üzerinde Az şeklinde sıralanarak yapılmıştır.
Kayıt Sıralama Testi	ORACLE	1.000.000	ORACLE Kayıt Sıralama Testi Ad Alanı Az	Bu test ORACLE veri tabanı üzerinde 1.000.000 kayıt ile Ad Alanı üzerinde Az şeklinde sıralanarak yapılmıştır.

Kategori	Veri Tabanı	Kayıt Sayısı	Adı	Açıklama
Kayıt Sıralama Testi	POSTGRESQL	1.000.000	POSTGRESQL Kayıt Sıralama Testi Ad Alanı Az	Bu test POSTGRESQL veri tabanı üzerinde 1.000.000 kayıt ile Ad Alanı üzerinde Az şeklinde sıralanarak yapılmıştır.
Kayıt Güncelleme Testi	MSSQL	25.000	MSSQL Kayıt Güncelleme Testi Ad Alanı Yeni Ad Ad	Bu test MSSQL veri tabanı üzerinde 25.000 kayıt ile Ad Alanı üzerinde Yeni Ad şeklinde güncellenerek yapılmıştır.
Kayıt Güncelleme Testi	MYSQL	25.000	MYSQL Kayıt Güncelleme Testi Ad Alanı Yeni Ad Ad	Bu test MYSQL veri tabanı üzerinde 25.000 kayıt ile Ad Alanı üzerinde Yeni Ad şeklinde güncellenerek yapılmıştır.
Kayıt Güncelleme Testi	ORACLE	25.000	ORACLE Kayıt Güncelleme Testi Ad Alanı Yeni Ad Ad	Bu test ORACLE veri tabanı üzerinde 25.000 kayıt ile Ad Alanı

Kategori	Veri Tabanı	Kayıt Sayısı	Adı	Açıklama
				üzerinde Yeni Ad şeklinde güncellenerek yapılmıştır.
Kayıt Güncelleme Testi	POSTGRESQL	25.000	POSTGRESQL Kayıt Güncelleme Testi Ad Alanı Yeni Ad Ad	Bu test POSTGRESQL veri tabanı üzerinde 25.000 kayıt ile Ad Alanı üzerinde Yeni Ad şeklinde güncellenerek yapılmıştır.
Kayıt Silme Testi	MSSQL	25.000	MSSQL Kayıt Silme Testi	Bu test MSSQL veri tabanı üzerinde 25.000 kayıt ile yapılmıştır.
Kayıt Silme Testi	MYSQL	25.000	MYSQL Kayıt Silme Testi	Bu test MYSQL veri tabanı üzerinde 25.000 kayıt ile yapılmıştır.
Kayıt Silme Testi	ORACLE	25.000	ORACLE Kayıt Silme Testi	Bu test ORACLE veri tabanı üzerinde 25.000 kayıt ile yapılmıştır.
Kayıt Silme Testi	POSTGRESQL	25.000	POSTGRESQL Kayıt Silme Testi	Bu test POSTGRESQL veri tabanı

Kategori	Veri Tabanı	Kayıt Sayısı	Adı	Açıklama
				üzerinde 25.000 kayıt ile yapılmıştır.
Kayıt Arama Testleri	MSSQL	1.000.000	MSSQL Kayıt Arama Testi Ad Alanı Eşittir Ad 100	Bu test MSSQL veri tabanı üzerinde 1.000.000 kayıt ile Ad Alanı üzerinde Ad 100 verisi Eşittir şeklinde aranarak yapılmıştır.
Kayıt Arama Testleri	MYSQL	1.000.000	MYSQL Kayıt Arama Testi Ad Alanı Eşittir Ad 100	Bu test MYSQL veri tabanı üzerinde 1.000.000 kayıt ile Ad Alanı üzerinde Ad 100 verisi Eşittir şeklinde aranarak yapılmıştır.
Kayıt Arama Testleri	ORACLE	1.000.000	ORACLE Kayıt Arama Testi Ad Alanı Eşittir Ad 100	Bu test ORACLE veri tabanı üzerinde 1.000.000 kayıt ile Ad Alanı üzerinde Ad 100 verisi Eşittir şeklinde

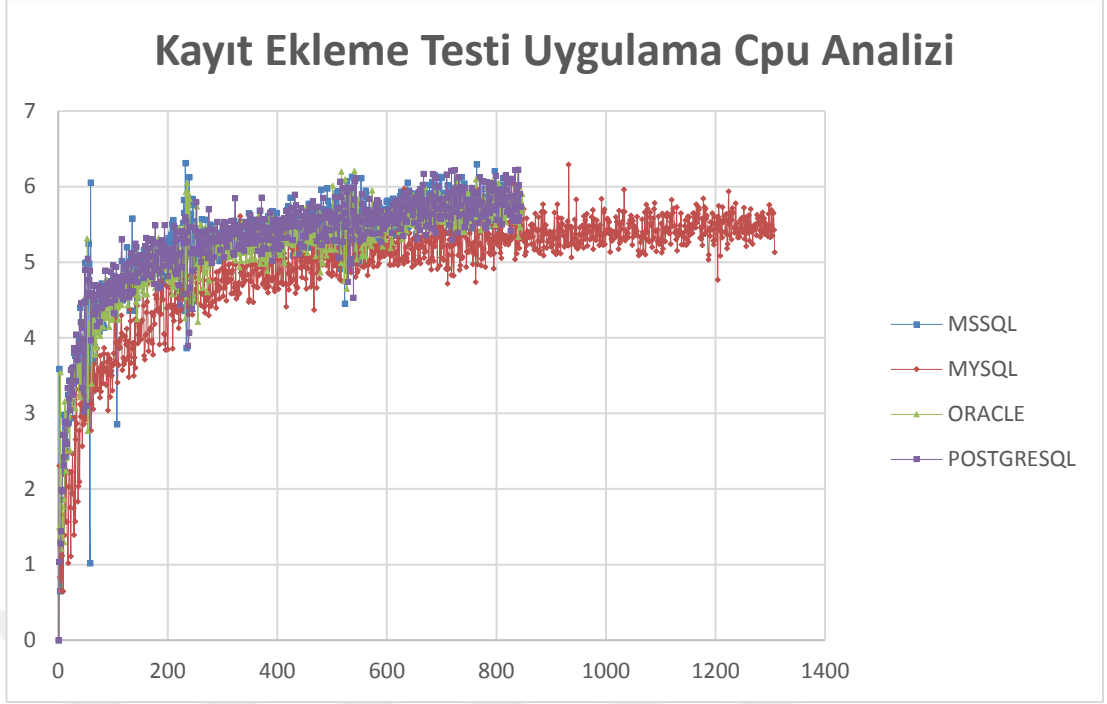
Kategori	Veri Tabanı	Kayıt Sayısı	Adı	Açıklama
				aranarak yapılmıştır.
Kayıt Arama Testleri	POSTGRESQL	1.000.000	POSTGRESQL Kayıt Arama Testi Ad Alanı Eşittir Ad 100	Bu test POSTGRESQL veri tabanı üzerinde 1.000.000 kayıt ile Ad Alanı üzerinde Ad 100 verisi Eşittir şeklinde aranarak yapılmıştır.

5.2. Kayıt Ekleme Testi'nin İncelenmesi

Kayıt ekleme testi 4 farklı veri tabanı üzerinde 25.000 kayıt ile uygulanmıştır. Uygulanan bu test ile İşlem süresi, Uygulama Cpu Kullanımı, Uygulama Ram Kullanımı, Veri Tabanı Engine Cpu Kullanımı ve Veri Tabanı Engine Ram Kullanım değerleri kayıt altına alınmıştır.

5.2.1. Kayıt Ekleme Testi Uygulama Cpu Analizi

Her bir veri tabanı ile uygulama çalışma durumuna ait cpu kullanım oranları grafiksel olarak ifade edilmiş olup, aynı grafik üzerinden karşılaştırılması yapılmıştır. Kayıt ekleme testinde veri tabanları test uygulama cpu kullanım değerleri Şekil 5.1'de gösterilmiştir. Kayıt ekleme testindeki işlem süreleri ve ortalama CPU değerleri Çizelge 5.2'de verilmiştir.



Şekil 5.1 Kayıt Ekleme Testi Uygulama Cpu Analizi

Çizelge 5.2 Kayıt Ekleme İşlem Süresi ve Ortalama Cpu Değeri

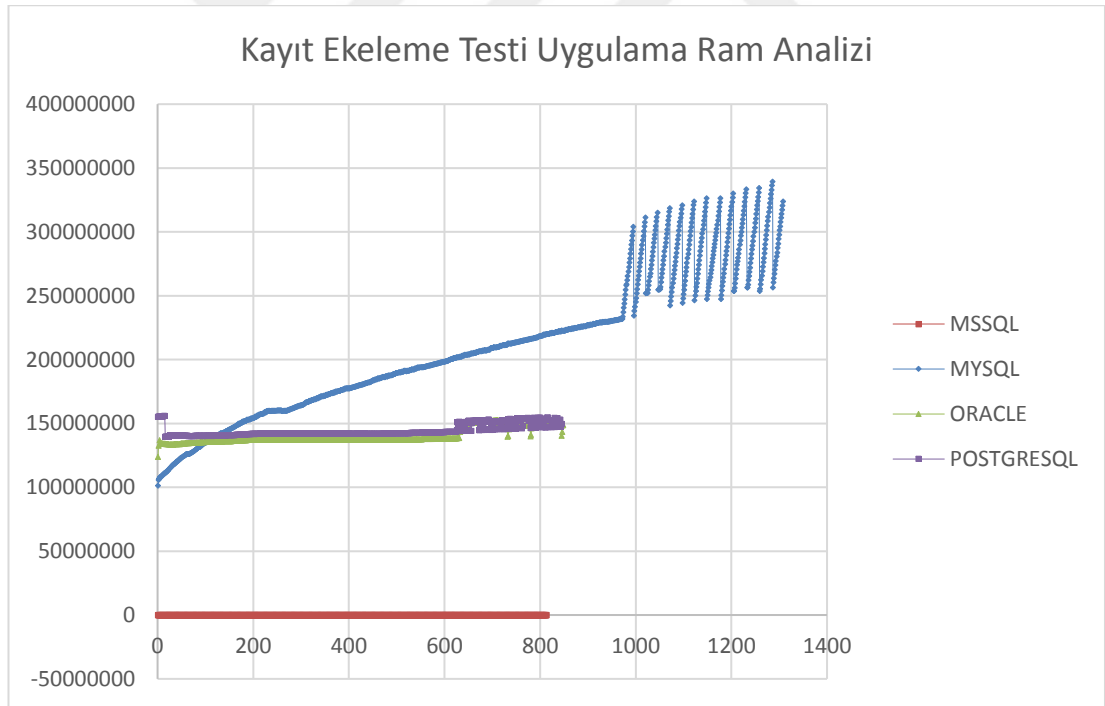
Veri Tabanı Adı	İşlem Süresi (Sn)	Ortalama (%)
MSSQL	813	5,20
MYSQL	1308	4,92
ORACLE	848	5,14
POSTGRESQL	845	5,26

Şekil 5.1 ve Çizelge 5.2 incelendiğinde MYSQL veri tabanının en fazla işlem süresinde kayıt eklediği ve ortalama 4.92 CPU değerinde çalıştığı görülmüştür. Diğer veri tabanlarına göre en düşük CPU değerini göstermiş olmasına rağmen en uzun işlem süresine sahip olmuştur. MSSQL veri tabanı en kısa işlem süresi olan 813 sn'de kayıt eklemeyi tamamlamış ve MYSQL'e göre yüksek CPU ortalama değeri olan 5.20 CPU değeri kullanmıştır. Diğer iki veri tabanı ise işlem süresi ve CPU ortalama değeri

bakımından ORACLE ve POSTGRESQL veri tabanlarına göre ortalama performans göstermiştir. Kayıt eklemede her ne kadar CPU değerleri önem arz etsede birbirlerine yakın sonuçlar vermiş olup, işlem süresi bakımından MSSQL veri tabanının kullanımı daha uygun olmaktadır.

5.2.2. Kayıt Ekleme Testi Uygulama Ram Analizi

Her bir veri tabanı ile uygulama çalışma durumuna ait ram kullanım oranları grafiksel olarak ifade edilmiş olup, aynı grafik üzerinden karşılaştırılması yapılmıştır. Kayıt ekleme testinde veri tabanları test uygulama ram kullanım değerleri Şekil 5.2’de gösterilmiştir. Kayıt ekleme testindeki işlem süreleri ve ortalama RAM değerleri Çizelge 5.3’de verilmiştir.



Şekil 5.2 Kayıt Ekleme Testi Uygulama Ram Analizi

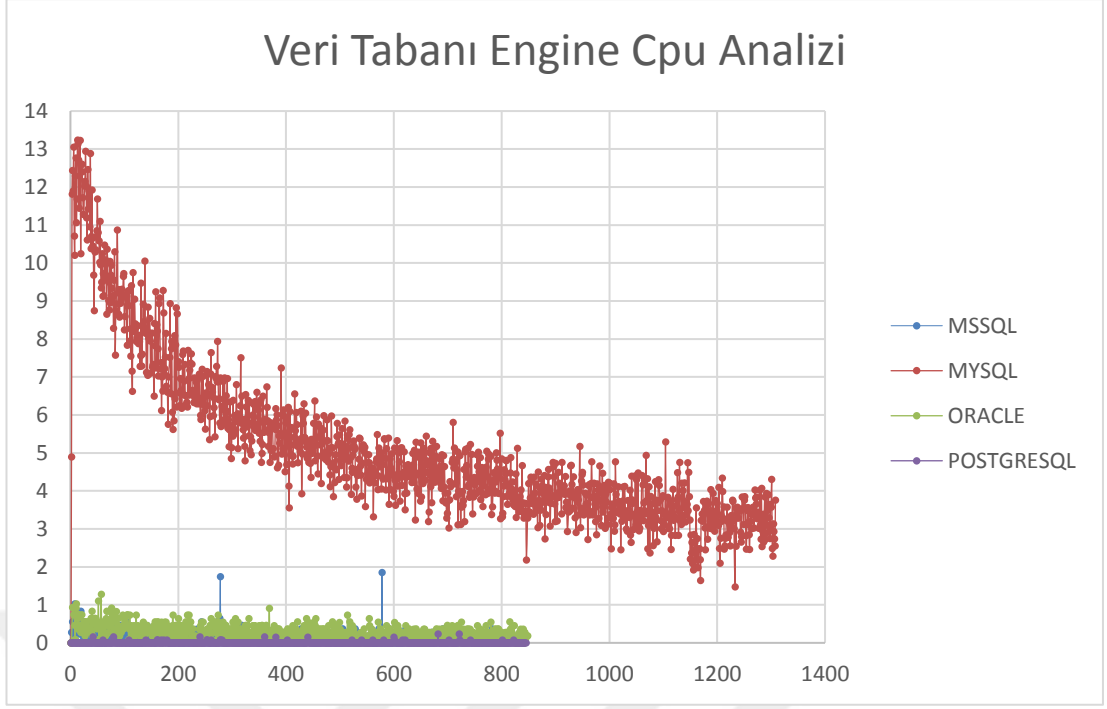
Çizelge 5.3 Kayıt Ekleme İşlem Süresi ve Ortalama Ram Değeri

Veri Tabanı Adı	İşlem Süresi (sn)	Ortalama (mb)
MSSQL	813	85,90
MYSQL	1308	199,60
ORACLE	848	133,54
POSTGRESQL	845	137,66

Şekil 5.2 ve Çizelge 5.3 incelendiğinde MYSQL veri tabanının en fazla işlem süresinde kayıt eklediği ve ortalama 199,60 mb ram kullandığı görülmüştür. Hem işlem süresi hemde ram kullanım ortalaması olarak en kötü değerlere sahiptir. MSSQL veri tabanı en kısa işlem süresi olan 813 sn’de kayıt eklemeyi tamamlamış ve en az ram olan 85.90 mb kullanmıştır. Diğer iki veri tabanı ise işlem süresi ve ram değeri bakımından MYSQL ve MSSQL veri tabanlarına göre ortalama performans göstermiştir. Kayıt eklemede her ne kadar ram değerleri önem arz etsede birbirlerine yakın sonuçlar vermiş olup, işlem süresi ve ram kullanımı bakımından MSSQL veri tabanının kullanımı daha uygun olmaktadır.

5.2.3. Kayıt Ekleme Testi Veri Tabanı Engine Cpu Analizi

Her bir veri tabanı engine çalışma durumuna ait cpu kullanım oranları grafiksel olarak ifade edilmiş olup, aynı grafik üzerinden karşılaştırılması yapılmıştır. Kayıt ekleme testinde veri tabanları engine çalışma cpu kullanım değerleri Şekil 5.3’de gösterilmiştir. Kayıt ekleme testindeki işlem süreleri ve ortalama CPU değerleri Çizelge 5.4’de verilmiştir.



Şekil 5.3 Kayıt Ekleme Testi Veri Tabanı Engine Cpu Analizi

Çizelge 5.4 Kayıt Ekleme İşlem Süresi ve Veri Tabanı Engine Cpu Değeri

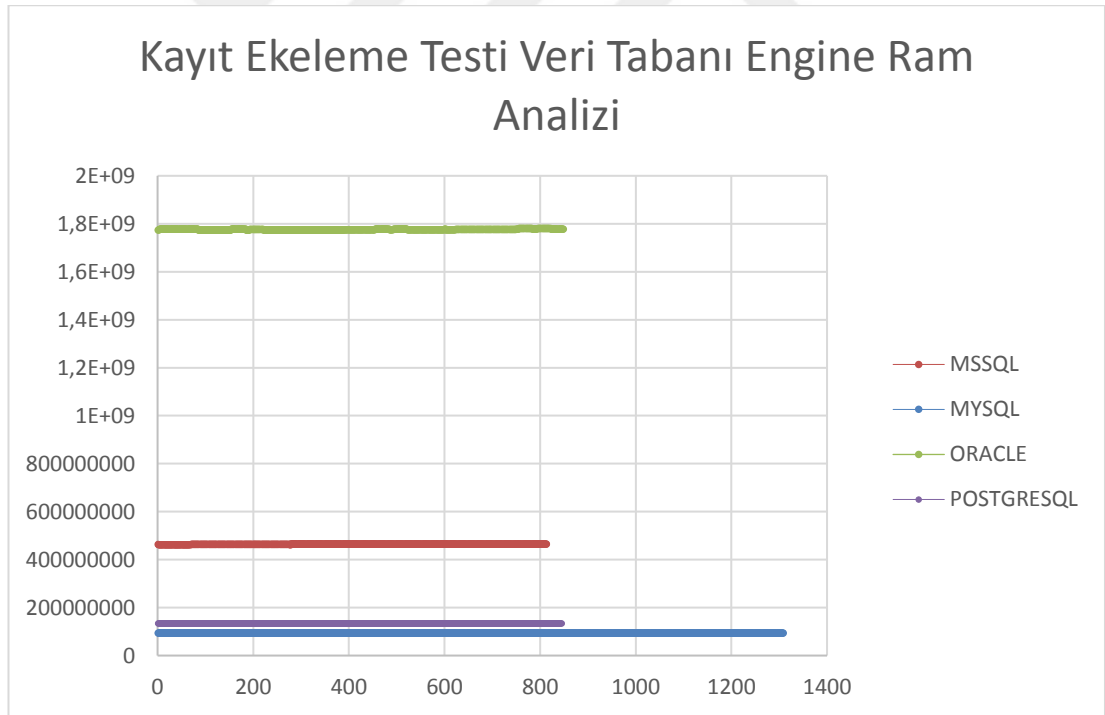
Veri Tabanı Adı	İşlem Süresi (sn)	Ortalama (%)
MSSQL	813	0,134
MYSQL	1308	5,13
ORACLE	848	0,28
POSTGRESQL	845	0,01

Şekil 5.3 ve Çizelge 5.4 incelendiğinde MYSQL veri tabanının en fazla işlem süresinde kayıt eklediği ve ortalama 5.13 cpu kullandığı görülmüştür. Hem işlem süresi hemde cpu kullanım ortalaması olarak en kötü değerlere sahiptir. MSSQL veri tabanı en kısa işlem süresi olan 813sn’de kayıt eklemeyi tamamlamış ve 0,134 cpu değeri kullanmıştır. ORACLE ve POSTGRESQL ise işlem süresi bakımından MYSQL ve MSSQL veri tabanlarına göre ortalama performans göstermiştir. Fakat POSTGRESQL en düşük ortalama cpu değerine sahip olmuştur. Kayıt eklemede her ne kadar cpu değerleri önem

arz etsede birbirlerine yakın sonuçlar vermiş olup, işlem süresi ve cpu kullanımı bakımından MSSQL ve POSTGRESQL veri tabanının kullanımı daha uygun olmaktadır.

5.2.4. Kayıt Ekleme Testi Veri Tabanı Engine Ram Analizi

Her bir veri tabanı engine çalışma durumuna ait ram kullanım oranları grafiksel olarak ifade edilmiş olup, aynı grafik üzerinden karşılaştırılması yapılmıştır. Kayıt ekleme testinde veri tabanları test engine çalışma durumuna ait ram kullanım değerleri Şekil 5.4'de gösterilmiştir. Kayıt ekleme testindeki işlem süreleri ve ortalama RAM değerleri Çizelge 5.5'de verilmiştir.



Şekil 5.4 Kayıt Ekleme Testi Veri Tabanı Engine Ram Analizi

Çizelge 5.5 Kayıt Ekleme İşlem Süresi ve Veri Tabanı Engine Ram Değeri

Veri Tabanı Adı	İşlem Süresi (sn)	Ortalama (mb)
MSSQL	813	443,52
MYSQL	1308	90,68
ORACLE	848	1694,55
POSTGRESQL	845	128,23

Şekil 5.4 ve Çizelge 5.5 incelendiğinde ORACLE veri tabanının 1694,55 mb olarak en fazla ram kullandığı görülmüştür. Ram kullanım ortalaması olarak en kötü değerlere sahiptir. MYSQL en az ram olan 90,68 mb ram kullanılmasına rağmen işlem süresi olarak en kötü değere sahiptir. Hem işlem süresi hemde ram kullanım oranı göz önüne alındığında veri tabanı engine ram analizine göre en iyi performansı POSTGRESQL veri tabanı göstermiştir.

5.3. Kayıt Arama Testi'nin İncelenmesi

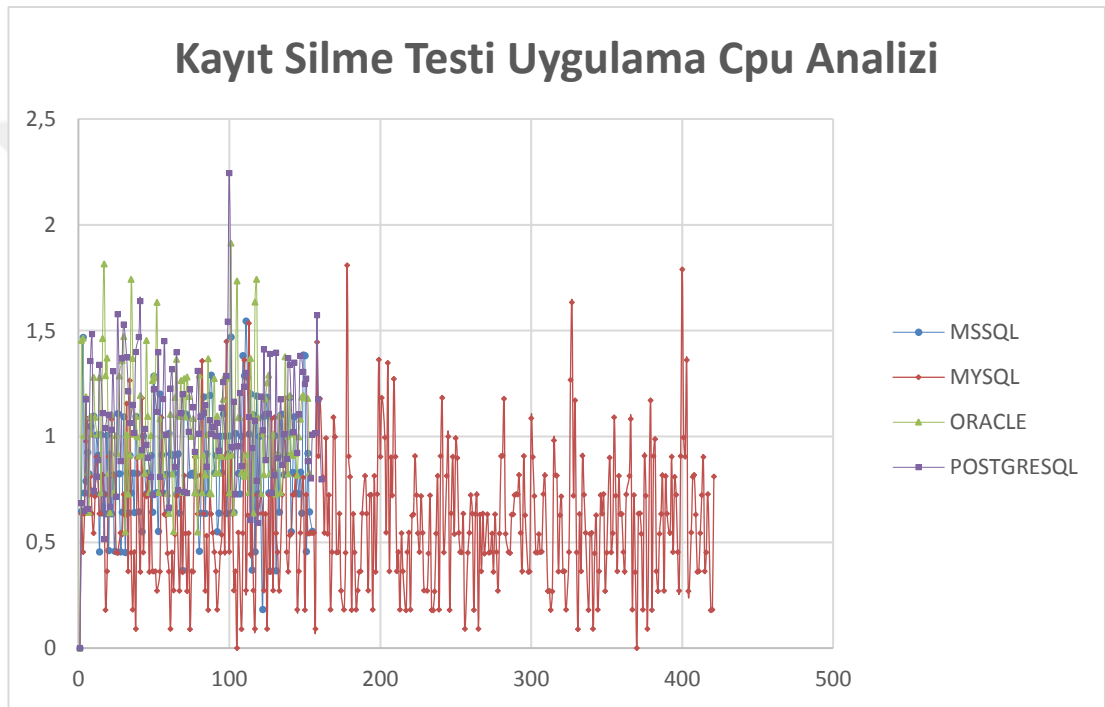
Kayıt arama testi 1.000.0000 kayıt üzerinde gerçekleştirilmiştir. Fakat işlemin çok kısa sürede sonuçlanması nedeniyle analiz edilebilecek cpu ve ram değerleri kayıt altına alınamamıştır. Kayıt sıralama testi ile benzer özellikler göstermesi sebebiyle bu değerler değerlendirmeye alınmamıştır. Bu test kayıt sıralama testi içerisinde değerlendirilmiştir.

5.4. Kayıt Silme Testi'nin İncelenmesi

Kayıt silme testi 25.000 kayıt ile 4 veri tabanı üzerinde başarıyla tamamlanmıştır. Uygulanan bu test ile İşlem süresi, Uygulama Cpu Kullanımı, Uygulama Ram Kullanımı, Veri Tabanı Engine Cpu Kullanımı ve Veri Tabanı Engine Ram Kullanım değerleri kayıt altına alınmıştır.

5.4.1. Kayıt Silme Testi Uygulama Cpu Analizi

Her bir veri tabanı ile uygulama çalışma durumuna ait cpu kullanım oranları grafiksel olarak ifade edilmiş olup, aynı grafik üzerinden karşılaştırılması yapılmıştır. Kayıt ekleme testinde veri tabanları test uygulama cpu kullanım değerleri Şekil 5.5'de gösterilmiştir. Kayıt ekleme testindeki işlem süreleri ve ortalama CPU değerleri Çizelge 5.6'da verilmiştir.



Şekil 5.5 Kayıt Silme Testi Uygulama Cpu Analizi

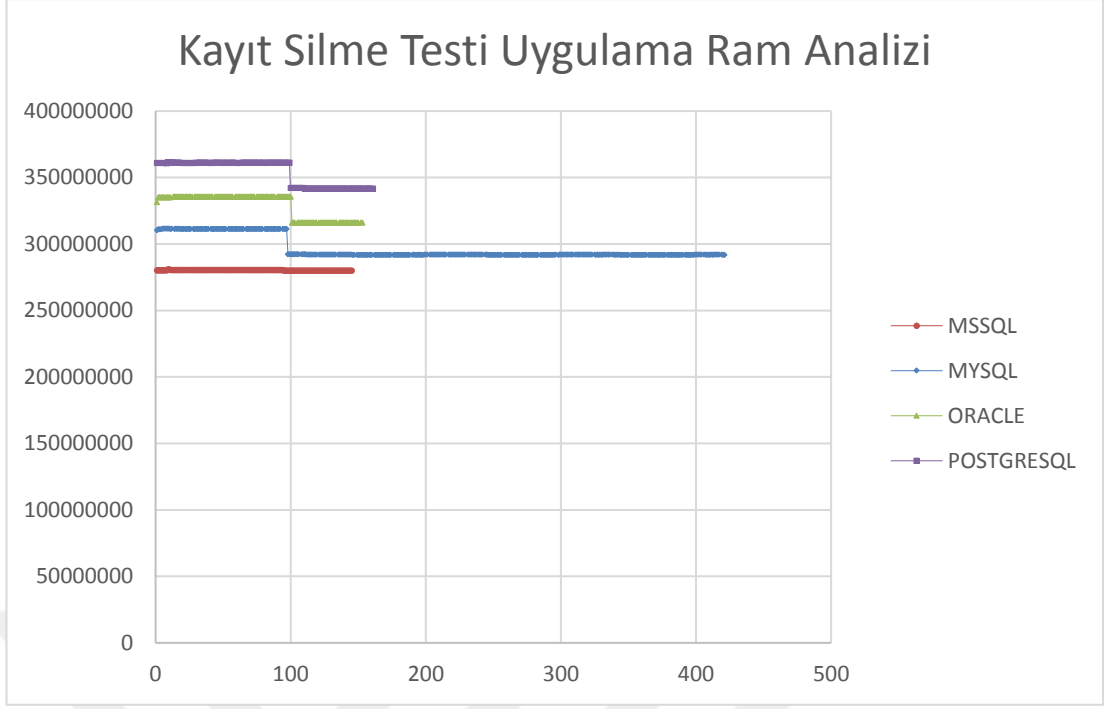
Çizelge 5.6 Kayıt Silme Uygulama Cpu Değeri

Veri Tabanı Adı	İşlem Süresi (sn)	Ortalama (%)
MSSQL	155	0,85
MYSQL	421	0,60
ORACLE	153	1,04
POSTGRESQL	161	1,07

Şekil 5.5 ve Çizelge 5.6 incelendiğinde MYSQL veri tabanının en fazla işlem süresinde kayıt sildiği ve ortalama 0,60 cpu değerinde çalıştığı görülmüştür. Diğer veri tabanlarına göre en düşük CPU değerini göstermiş olmasına rağmen en uzun işlem süresine sahip olmuştur. ORACLE veri tabanı en kısa işlem süresi olan 153 sn'de kayıt silmeyi tamamlamış fakat ortalama değeri 1,04 cpu değeri kullanmıştır. Diğer iki veri tabanı ise işlem süresi ve CPU ortalama değeri bakımından MYSQL ve ORACLE veri tabanlarına göre ortalama performans göstermiştir. Kayıt silmede her ne kadar CPU değerleri önem arz etsede birbirlerine yakın sonuçlar vermiş olup, işlem süresi bakımından MSSQL ve ORACLE veri tabanının kullanımı daha uygun olmaktadır.

5.4.2. Kayıt Silme Testi Uygulama Ram Analizi

Her bir veri tabanı ile uygulama çalışma durumuna ait ram kullanım oranları grafiksel olarak ifade edilmiş olup, aynı grafik üzerinden karşılaştırılması yapılmıştır. Kayıt silme testinde veri tabanları test uygulama ram kullanım değerleri Şekil 5.6'da gösterilmiştir. Kayıt ekleme testindeki işlem süreleri ve ortalama RAM değerleri Çizelge 5.7'de verilmiştir.



Şekil 5.6 Kayıt Silme Testi Uygulama Ram Analizi

Çizelge 5.7 Kayıt Silme İşlem Süresi ve Ortalama Ram Değeri

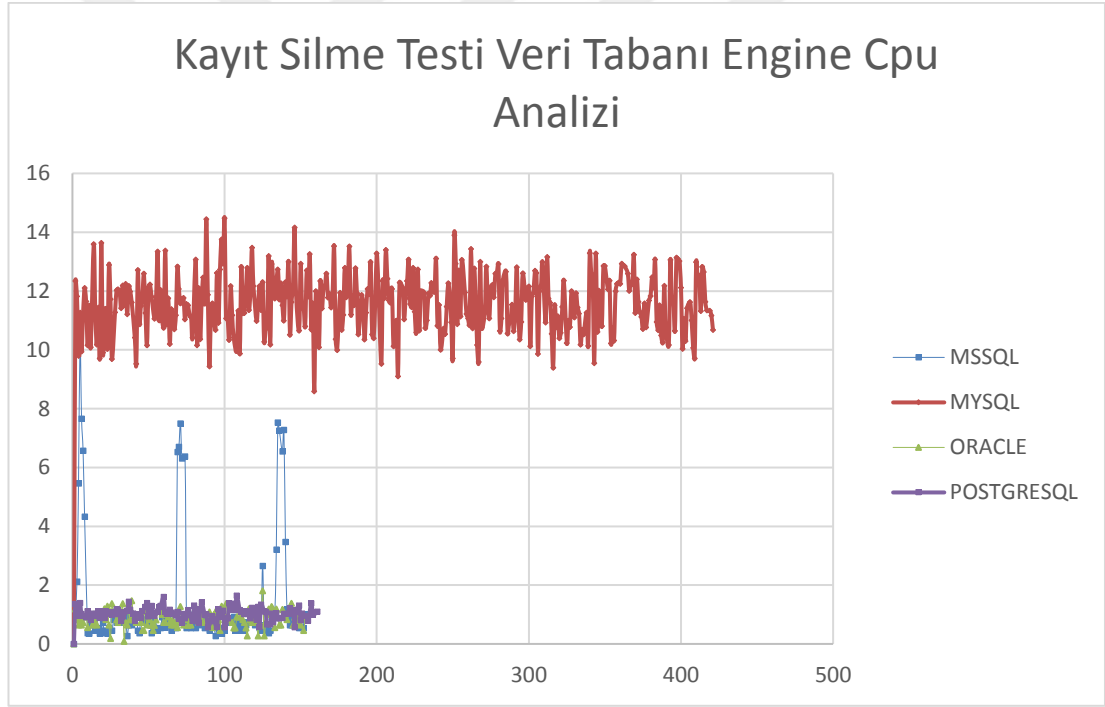
Veri Tabanı Adı	İşlem Süresi (sn)	Ortalama (mb)
MSSQL	155	267,31
MYSQL	421	282,71
ORACLE	153	313,63
POSTGRESQL	161	337,24

Şekil 5.6 ve Çizelge 5.7 incelendiğinde MYSQL veri tabanının en fazla işlem süresinde kayıt eklediği ve ortalama 282,71 mb ram kullandığı görülmüştür. ORACLE veri tabanı en kısa işlem süresi olan 153 sn’de kayıt silmeyi tamamlamış fakat en fazla ikinci ram değeri olan 313,63 mb kullanmıştır. POSTGRESQL en yüksek ram kullanımı ile 161 sn’de işlemini tamamlamış, MSSQL veri tabanı ise en düşük ram kullanımı ile 155 sn’de işlemi bitirmiştir. Kayıt silmede her ne kadar ram değerleri önem arz etsede birbirlerine

yakın sonuçlar vermiş olup, işlem süresi ve ram kullanımı bakımından MSSQL veri tabanının kullanımı daha uygun olmaktadır.

5.4.3. Kayıt Silme Testi Veri Tabanı Engine Cpu Analizi

Her bir veri tabanı engine çalışma durumuna ait cpu kullanım oranları grafiksel olarak ifade edilmiş olup, aynı grafik üzerinden karşılaştırılması yapılmıştır. Kayıt silme testinde veri tabanları engine çalışma cpu kullanım değerleri Şekil 5.7’de gösterilmiştir. Kayıt ekleme testindeki işlem süreleri ve ortalama CPU değerleri Çizelge 5.8’de verilmiştir.



Şekil 5.7 Kayıt Silme Testi Veri Tabanı Engine Cpu Analizi

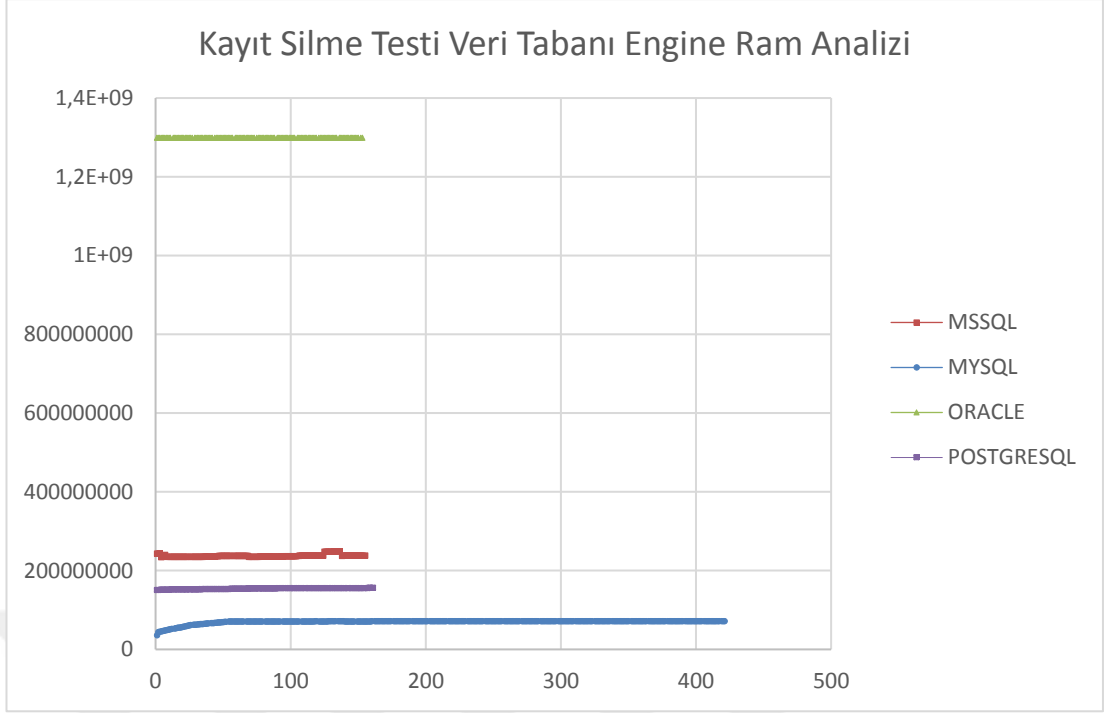
Çizelge 5.8 Kayıt Silme İşlem Süresi ve Veri Tabanı Engine Cpu Değeri

Veri Tabanı Adı	İşlem Süresi (sn)	Ortalama (%)
MSSQL	155	1,34
MYSQL	421	11,57
ORACLE	153	0,85
POSTGRESQL	161	1,03

Şekil 5.7 ve Çizelge 5.8 incelendiğinde MYSQL veri tabanının en fazla işlem süresinde kayıt eklediği ve ortalama 11,57 cpu kullandığı görülmüştür. Hem işlem süresi hemde cpu kullanım ortalaması olarak en kötü değerlere sahiptir. ORACLE veri tabanı en kısa işlem süresi olan 153 sn’de kayıt silmeyi tamamlamış aynı zamanda az cpu olan 0,85 değeri kullanmıştır. Diğer iki veri tabanı ise işlem süresi ve ram değeri bakımından MYSQL ve ORACLE veri tabanlarına göre ortalama performans göstermiştir. Kayıt silmede her ne kadar cpu değerleri önem arz etsede birbirlerine yakın sonuçlar vermiş olup, işlem süresi ve cpu kullanımını bakımından ORACLE veri tabanının kullanımı daha uygun olmaktadır.

5.4.4. Kayıt Silme Testi Veri Tabanı Engine Ram Analizi

Her bir veri tabanı engine çalışma durumuna ait ram kullanım oranları grafiksel olarak ifade edilmiş olup, aynı grafik üzerinden karşılaştırılması yapılmıştır. Kayıt silme testinde veri tabanları test engine çalışma durumuna ait ram kullanım değerleri Şekil 5.8’de gösterilmiştir. Kayıt ekleme testindeki işlem süreleri ve ortalama RAM değerleri Çizelge 5. 9’da verilmiştir.



Şekil 5.8 Kayıt Silme Veri Tabanı Engine Ram Analizi

Çizelge 5.9 Kayıt Silme İşlem Süresi ve Veri Tabanı Engine Ram Değeri

Veri Tabanı Adı	İşlem Süresi (sn)	Ortalama (mb)
MSSQL	155	227,03
MYSQL	421	66,79
ORACLE	153	1239,11
POSTGRESQL	161	147,25

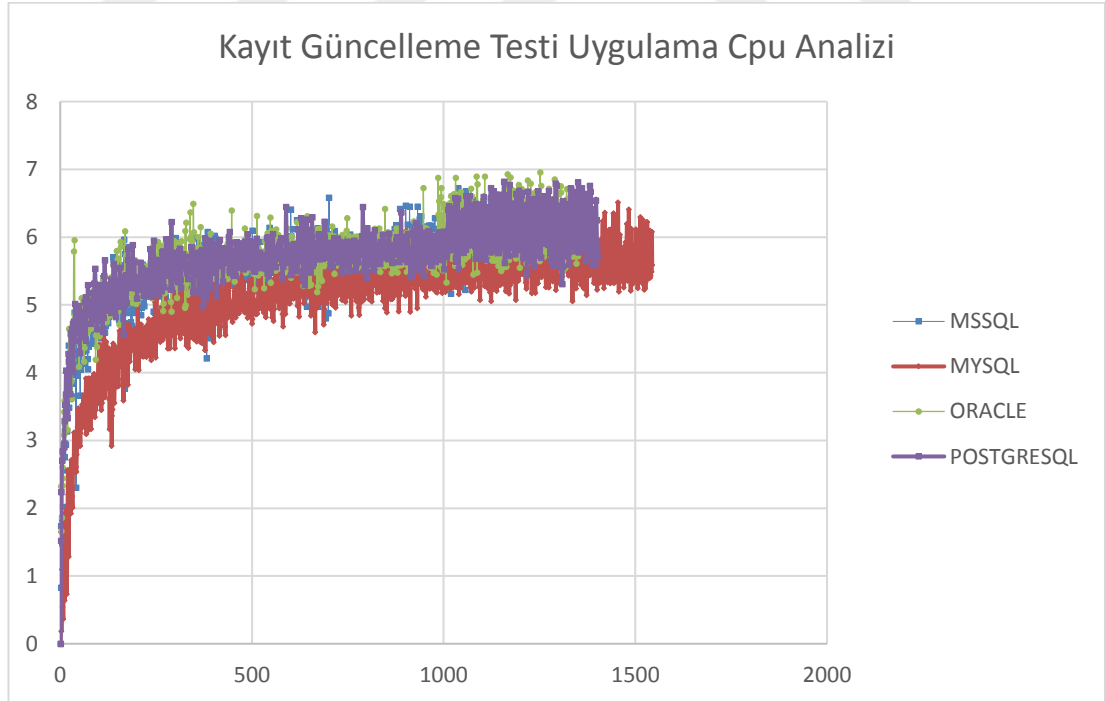
Şekil 5.8 ve Çizelge 5.9 incelendiğinde ORACLE veri tabanının 1239,11 mb olarak en fazla ram kullanarak en düşük işlem süresine sahip olduğu görülmüştür. MYSQL en az ram olan 66,79 mb ram kullanılmasına rağmen işlem süresi olarak en kötü değere sahiptir. Hem işlem süresi hemde ram kullanım oranı göz önüne alındığında veri tabanı engine ram analizine göre en iyi performansı POSTGRESQL veri tabanı göstermiştir.

5.5. Kayıt Güncelleme Testi'nin İncelenmesi

Kayıt güncelleme testi 25.000 kayıt ile 4 veri tabanı üzerinde başarıyla tamamlanmıştır. Uygulanan bu test ile İşlem süresi, Uygulama Cpu Kullanımı, Uygulama Ram Kullanımı, Veri Tabanı Engine Cpu Kullanımı ve Veri Tabanı Engine Ram Kullanım değerleri kayıt altına alınmıştır.

5.5.1. Kayıt Güncelleme Testi Uygulama Cpu Analizi

Her bir veri tabanı ile uygulama çalışma durumuna ait cpu kullanım oranları grafiksel olarak ifade edilmiş olup, aynı grafik üzerinden karşılaştırılması yapılmıştır. Kayıt güncelleme testinde veri tabanları test uygulama cpu kullanım değerleri Şekil 5.9'da gösterilmiştir. Kayıt güncelleme testindeki işlem süreleri ve ortalama CPU değerleri Çizelge 5.10'de verilmiştir.



Şekil 5.9 Kayıt Güncelleme Testi Uygulama Cpu Analizi

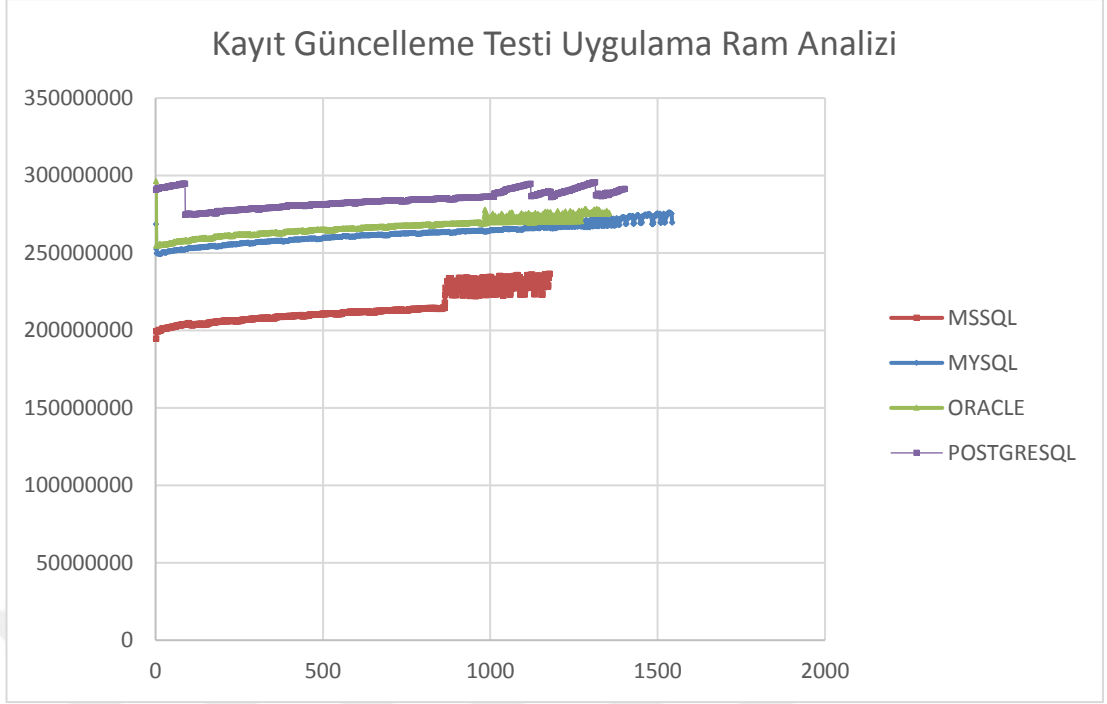
Çizelge 5.10 Kayıt Güncelleme Uygulama Cpu Değeri

Veri Tabanı Adı	İşlem Süresi (Sn)	Ortalama (%)
MSSQL	1176	5,48
MYSQL	1544	5,09
ORACLE	1355	5,68
POSTGRESQL	1401	5,70

Şekil 5.9 ve Çizelge 5.10 incelendiğinde bütün veri tabanlarında cpu değerleri birbirine yakın iken işlem sürelerine bakıldığında MSSQL en iyi performansı göstermektedir. MYSQL ise en kötü performansı göstermektedir. Diğer performanslar bu değerler arasında bulunmaktadır.

5.5.2. Kayıt Güncelleme Testi Uygulama Ram Analizi

Her bir veri tabanı ile uygulama çalışma durumuna ait ram kullanım oranları grafiksel olarak ifade edilmiş olup, aynı grafik üzerinden karşılaştırılması yapılmıştır. Kayıt güncelleme testinde veri tabanları test uygulama ram kullanım değerleri Şekil 5.10'de gösterilmiştir. Kayıt güncelleme testindeki işlem süreleri ve ortalama RAM değerleri Çizelge 5.11'de verilmiştir.



Şekil 5.10 Kayıt Güncelleme Testi Uygulama Ram Analizi

Çizelge 5.11 Kayıt Güncelleme İşlem Süresi ve Ortalama Ram Değeri

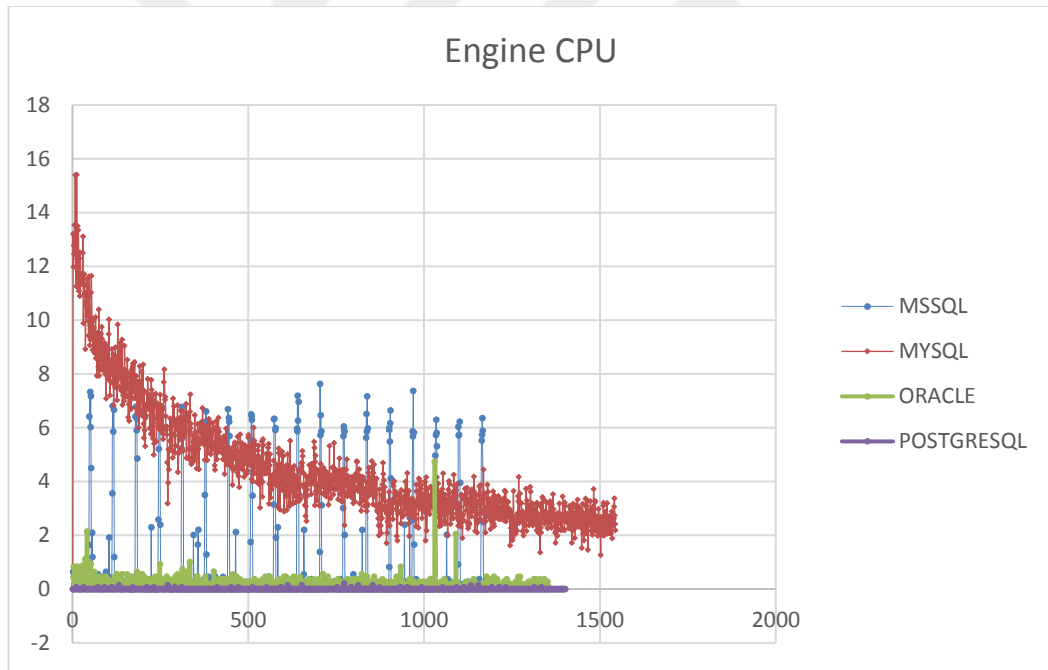
Veri Tabanı Adı	İşlem Süresi (sn)	Ortalama (mb)
MSSQL	1176	204,41
MYSQL	1544	250,26
ORACLE	1355	254,18
POSTGRESQL	1401	271,48

Şekil 5.10 ve Çizelge 5.11 incelendiğinde MYSQL veri tabanının en fazla işlem süresinde kayıt güncellediği ve ortalama 250,26 mb ram kullandığı görülmüştür. MSSQL veri tabanı en kısa işlem süresi olan 1176 sn'de kayıt güncellemeyi tamamlamış ve en düşük ram olan 204,41 mb kullanmıştır. Diğer iki veri tabanı ise işlem süresi ve ram değeri bakımından MYSQL ve MSSQL veri tabanlarına göre ortalama performans göstermiştir. POSTGRESQL veri tabanı ortalama bir performans göstermiş olmasına

rağmen en yüksek ram olan 271,48 mb kullanımına sahiptir. Kayıt güncellemede her ne kadar ram değerleri önem arz etsede birbirlerine yakın sonuçlar vermiş olup, işlem süresi ve ram kullanımı bakımından MSSQL veri tabanının kullanımı daha uygun olmaktadır.

5.5.3. Kayıt Güncelleme Testi Veri Tabanı Engine Cpu Analizi

Her bir veri tabanı engine çalışma durumuna ait cpu kullanım oranları grafiksel olarak ifade edilmiş olup, aynı grafik üzerinden karşılaştırılması yapılmıştır. Kayıt güncelleme testinde veri tabanları engine çalışma cpu kullanım değerleri Şekil 5.11’de gösterilmiştir. Kayıt güncelleme testindeki işlem süreleri ve ortalama CPU değerleri Çizelge 5.12’de verilmiştir.



Şekil 5.11 Kayıt Güncelleme Testi Veri Tabanı Engine Cpu Analizi

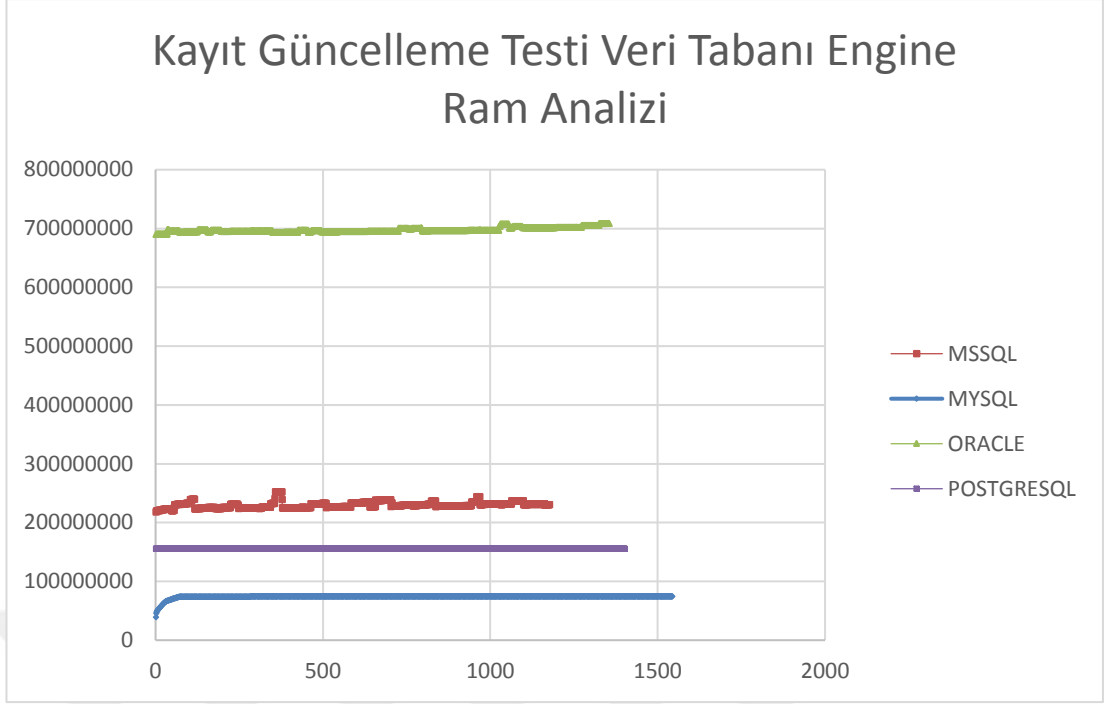
Çizelge 5.12 Kayıt Güncelleme İşlem Süresi ve Veri Tabanı Engine Cpu Değeri

Veri Tabanı Adı	İşlem Süresi (sn)	Ortalama (%)
MSSQL	1176	0,63
MYSQL	1544	4,51
ORACLE	1355	0,17
POSTGRESQL	1401	0,003

Şekil 5.11 ve Çizelge 5.12 incelendiğinde MYSQL veri tabanının en fazla işlem süresinde kayıt güncellediği ve ortalama 4,51 cpu kullandığı görülmüştür. Hem işlem süresi hemde cpu kullanım ortalaması olarak en kötü değerlere sahiptir. MSSQL veri tabanı en kısa işlem süresi olan 1176 sn'de kayıt eklemeyi tamamlayarak 0,63 cpu ortalamasına sahiptir. POSTGRESQL veri tabanı en düşük ortalama cpu değeri olan 0,003 ile işlemi bitirmiş olmasına rağmen işlem süresi bakımından düşük bir performans göstermiştir. ORACLE ise işlem süresi ve ortalama cpu bakımından MSSQL'den sonra en iyi performansa sahiptir. Kayıt güncellemede her ne kadar cpu değerleri önem arz etsede birbirlerine yakın sonuçlar vermiş olup, işlem süresi ve cpu kullanımı bakımından MSSQL ya da ORACLE veri tabanının kullanımı daha uygun olmaktadır.

5.5.4. Kayıt Güncelleme Testi Veri Tabanı Engine Ram Analizi

Her bir veri tabanı engine çalışma durumuna ait ram kullanım oranları grafiksel olarak ifade edilmiş olup, aynı grafik üzerinden karşılaştırılması yapılmıştır. Kayıt güncelleme testinde veri tabanları test engine çalışma durumuna ait ram kullanım değerleri Şekil 5.12'de gösterilmiştir. Kayıt güncelleme testindeki işlem süreleri ve ortalama RAM değerleri Çizelge 5.13'de verilmiştir.



Şekil 5.12 Kayıt Güncelleme Testi Veri Tabanı Engine Ram Analizi

Çizelge 5.13 Kayıt Güncelleme İşlem Süresi ve Veri Tabanı Engine Ram Değeri

Veri Tabanı Adı	İşlem Süresi (sn)	Ortalama (mb)
MSSQL	1176	219,19
MYSQL	1544	70,36
ORACLE	1355	665,28
POSTGRESQL	1401	148,07

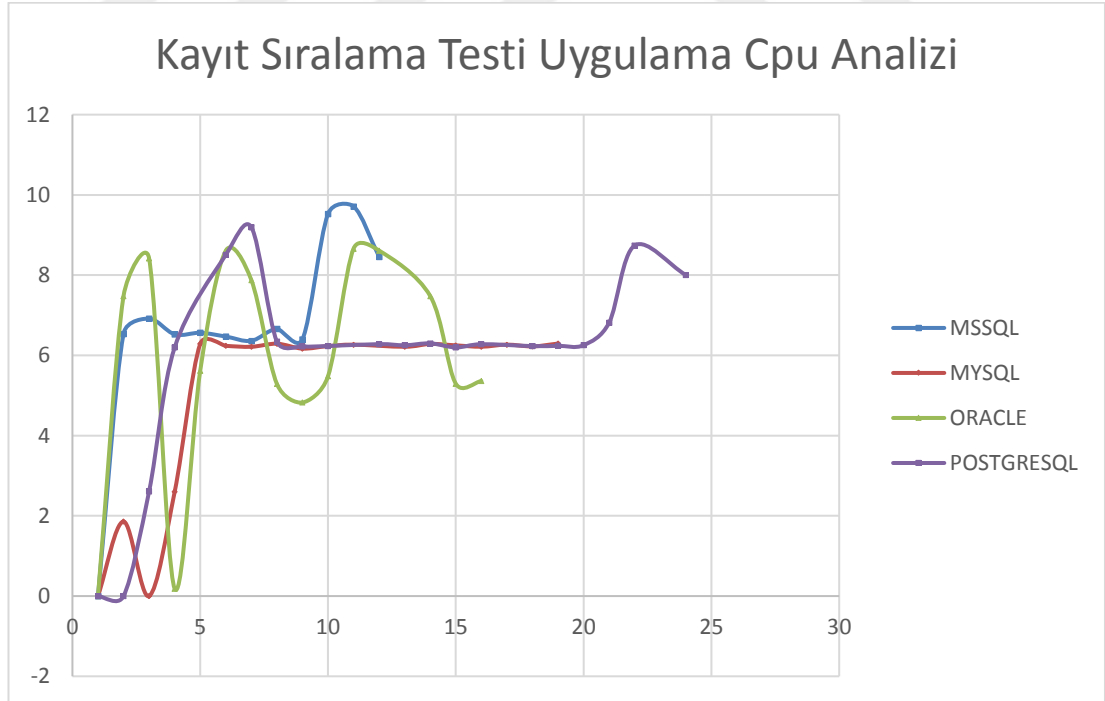
Şekil 5.12 ve Çizelge 5.13 incelendiğinde ORACLE veri tabanının 665,28 mb olarak en fazla ram kullandığı görülmüştür. Ram kullanım ortalaması olarak en kötü değerlere sahiptir. MYSQL en az ram olan 70,36 mb ram kullanılmasına rağmen işlem süresi olarak en kötü değere sahiptir. Hem işlem süresi hemde ram kullanım oranı göz önüne alındığında veri tabanı engine ram analizine göre en iyi performansı MSSQL veri tabanı göstermiştir.

5.6. Kayıt Sıralama Testi'nin İncelenmesi

Kayıt sıralama testi 25.000 kayıt ile 4 veri tabanı üzerinde başarıyla tamamlanmıştır. Uygulanan bu test ile İşlem süresi, Uygulama Cpu Kullanımı, Uygulama Ram Kullanımı, Veri Tabanı Engine Cpu Kullanımı ve Veri Tabanı Engine Ram Kullanım değerleri kayıt altına alınmıştır.

5.6.1. Kayıt Sıralama Testi Uygulama Cpu Analizi

Her bir veri tabanı ile uygulama çalışma durumuna ait cpu kullanım oranları grafiksel olarak ifade edilmiş olup, aynı grafik üzerinden karşılaştırılması yapılmıştır. Kayıt sıralama testinde veri tabanları test uygulama cpu kullanım değerleri Şekil 5.13'de gösterilmiştir. Kayıt sıralama testindeki işlem süreleri ve ortalama CPU değerleri Çizelge 5.14'de verilmiştir.



Şekil 5.13 Kayıt Sıralama Testi Uygulama Cpu Analizi

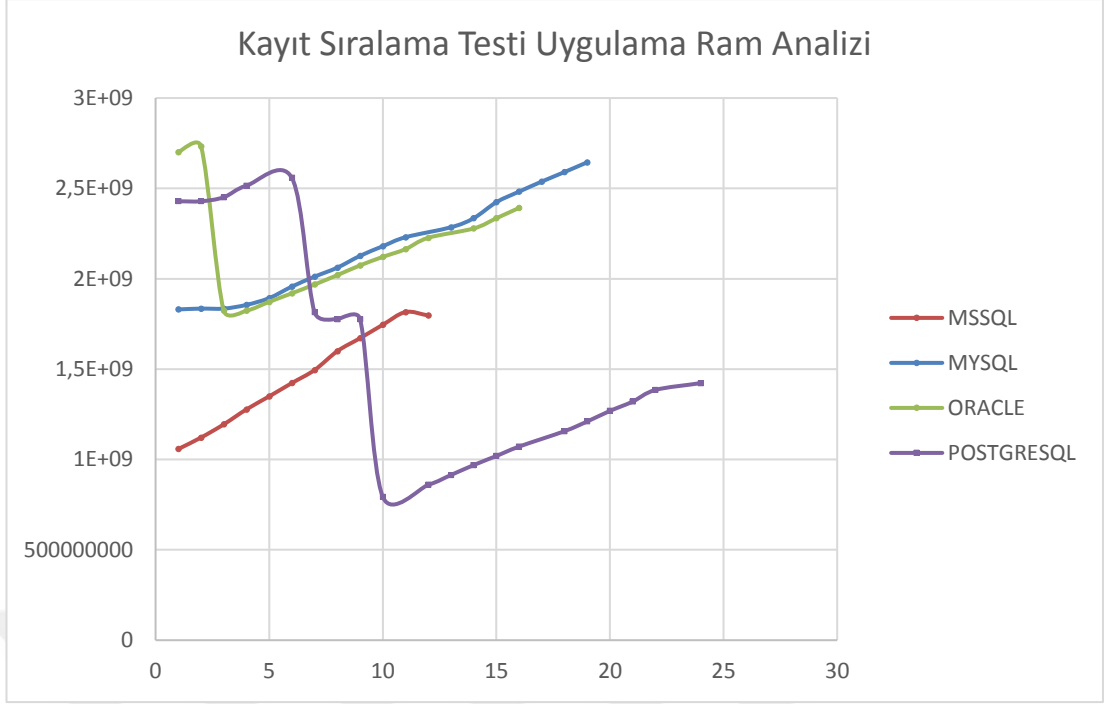
Çizelge 5.14 Kayıt Sıralama Uygulama Cpu Değeri

Veri Tabanı Adı	İşlem Süresi (Sn)	Ortalama (%)
MSSQL	12	6,68
MYSQL	19	5,10
ORACLE	16	5,94
POSTGRESQL	24	5,94

Şekil 5.13 ve Çizelge 5.14 incelendiğinde bütün veri tabanlarında cpu değerleri birbirine yakın iken işlem sürelerine bakıldığında POSTGRESQL en kötü performansı göstermektedir. MSSQL ise en iyi performansı göstermektedir. Diğer performanslar bu değerler arasında bulunmaktadır.

5.6.2. Kayıt Sıralama Testi Uygulama Ram Analizi

Her bir veri tabanı ile uygulama çalışma durumuna ait ram kullanım oranları grafiksel olarak ifade edilmiş olup, aynı grafik üzerinden karşılaştırılması yapılmıştır. Kayıt sıralama testinde veri tabanları test uygulama ram kullanım değerleri Şekil 5.14'de gösterilmiştir. Kayıt sıralama testindeki işlem süreleri ve ortalama RAM değerleri Çizelge 5.15'de verilmiştir.



Şekil 5.14 Kayıt Sıralama Testi Uygulama Ram Analizi

Çizelge 5.15 Kayıt Sıralama İşlem Süresi ve Ortalama Ram Değeri

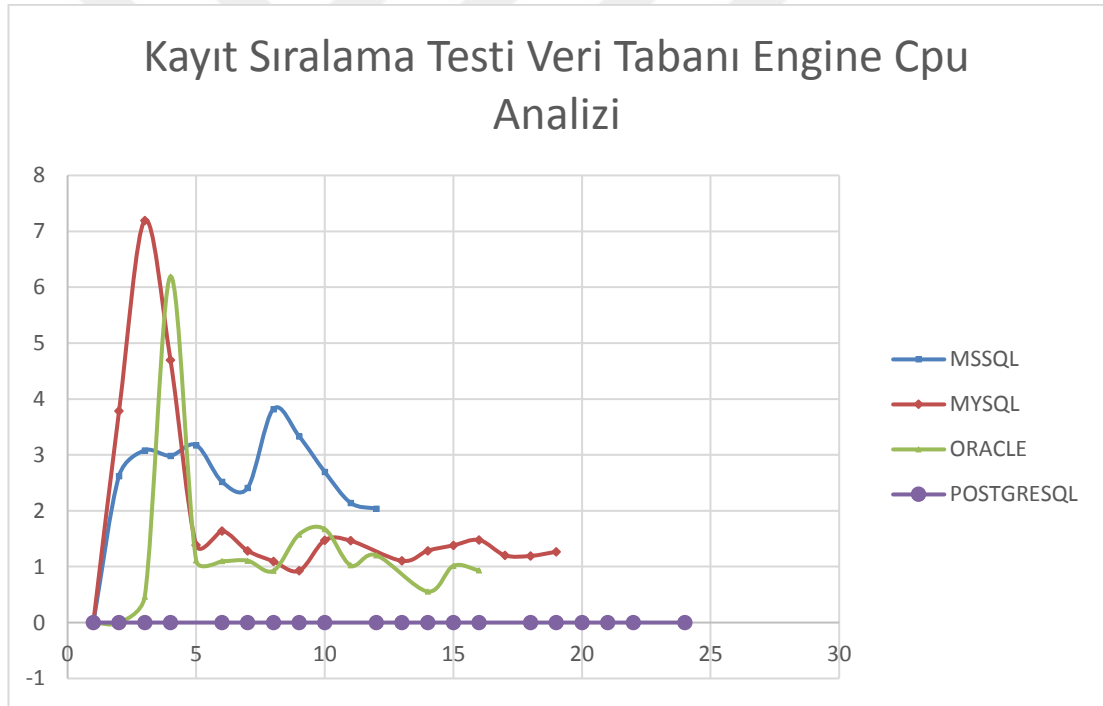
Veri Tabanı Adı	İşlem Süresi (sn)	Ortalama (mb)
MSSQL	12	1394,50
MYSQL	19	2072,38
ORACLE	16	2063,29
POSTGRESQL	24	1484,87

Şekil 5.14 ve Çizelge 5.15 incelendiğinde POSTGRESQL veri tabanının en fazla işlem süresinde kayıt sıraladığı ve ortalama 1484,87 mb ram kullandığı görülmüştür. MSSQL veri tabanı en kısa işlem süresi olan 12 sn'de kayıt sıralamayı tamamlamış hemde en az ram olan 1394,50 mb kullanmıştır. Diğer iki veri tabanı ise işlem süresi ve ram değeri bakımından POSTGRESQL ve MSSQL veri tabanlarına göre ortalama performans göstermiştir. Kayıt sıralamada her ne kadar ram değerleri önem arz etsede birbirlerine

yakın sonuçlar vermiş olup, işlem süresi ve ram kullanımı bakımından MSSQL veri tabanının kullanımı daha uygun olmaktadır.

5.6.3. Kayıt Sıralama Testi Veri Tabanı Engine Cpu Analizi

Her bir veri tabanı engine çalışma durumuna ait cpu kullanım oranları grafiksel olarak ifade edilmiş olup, aynı grafik üzerinden karşılaştırılması yapılmıştır. Kayıt sıralama testinde veri tabanları engine çalışma cpu kullanım değerleri Şekil 5.15’de gösterilmiştir. Kayıt sıralama testindeki işlem süreleri ve ortalama CPU değerleri Çizelge 5.16’da verilmiştir.



Şekil 5.15 Kayıt Sıralama Testi Veri Tabanı Engine Cpu Analizi

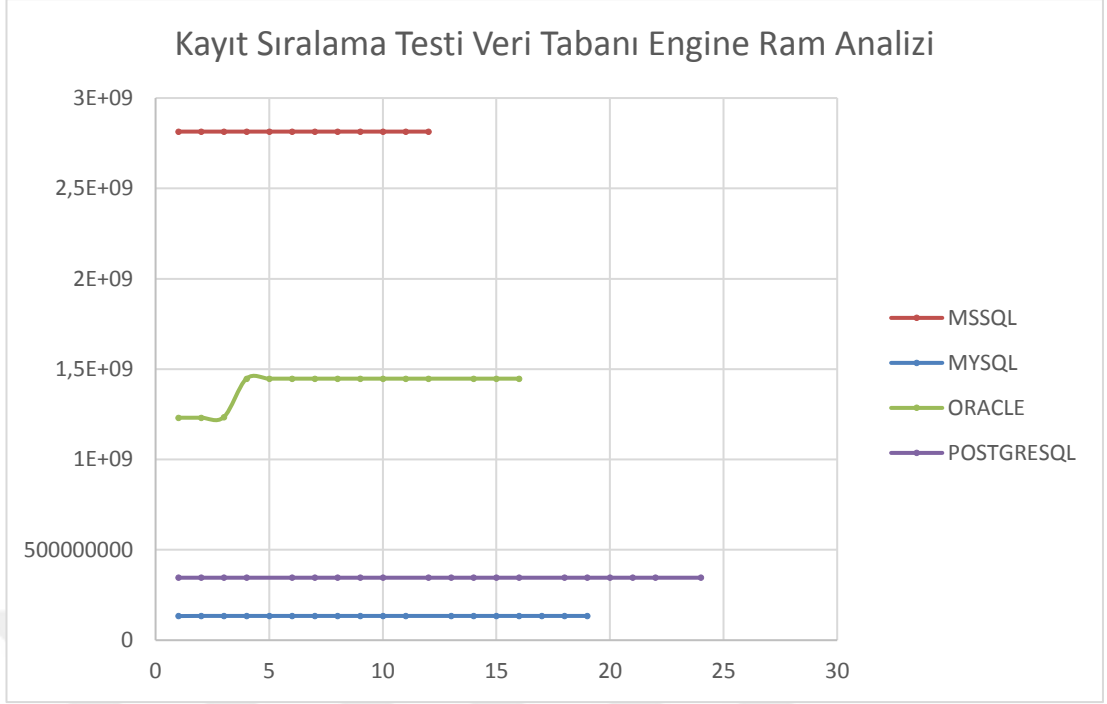
Çizelge 5.16 Kayıt Sıralama İşlem Süresi ve Veri Tabanı Engine Cpu Değeri

Veri Tabanı Adı	İşlem Süresi (sn)	Ortalama (%)
MSSQL	12	2,57
MYSQL	19	1,88
ORACLE	16	0,93
POSTGRESQL	24	0,00

Şekil 5.15 ve Çizelge 5.16 incelendiğinde POSTGRESQL veri tabanının en fazla işlem süresinde kayıt güncellediği ve ortalama 0,00 cpu kullandığı görülmüştür. MSSQL veri tabanı en kısa işlem süresi olan 12 sn’de kayıt sıralamayı tamamlamıştır. Cpu kullanımını ve işlem süreleri ortak değerlendirildiğinde ORACLE veri tabanı en iyi performansı göstermiştir.

5.6.4. Kayıt Sıralama Testi Veri Tabanı Engine Ram Analizi

Her bir veri tabanı engine çalışma durumuna ait ram kullanım oranları grafiksel olarak ifade edilmiş olup, aynı grafik üzerinden karşılaştırılması yapılmıştır. Kayıt sıralama testinde veri tabanları test engine çalışma durumuna ait ram kullanım değerleri Şekil 5.16’de gösterilmiştir. Kayıt sıralama testindeki işlem süreleri ve ortalama RAM değerleri Çizelge 5.17’de verilmiştir.



Şekil 5.16 Kayıt Sıralama Testi Veri Tabanı Engine Ram Analizi

Çizelge 5.17 Kayıt Sıralama İşlem Süresi ve Veri Tabanı Engine Ram Değeri

Veri Tabanı Adı	İşlem Süresi (sn)	Ortalama (mb)
MSSQL	12	2683,99
MYSQL	19	128,10
ORACLE	16	1339,25
POSTGRESQL	24	329,84

Şekil 5.16 ve Çizelge 5.17 incelendiğinde MSSQL veri tabanının 2683,99 mb olarak en fazla ram kullanarak en kısa sürede işlemi tamamladığı görülmüştür. POSTGRESQL veri tabanı en kötü işlem süresi olan 24 sn'de 329,84 mb ram kullanımı ile kayıt sıralama işlemi gerçekleştirmiştir. MYSQL en az ram olan 128,10 mb ram kullanılmasına rağmen işlem süresi olarak POSTGRESQL'den sonra en kötü değere sahiptir. Hem

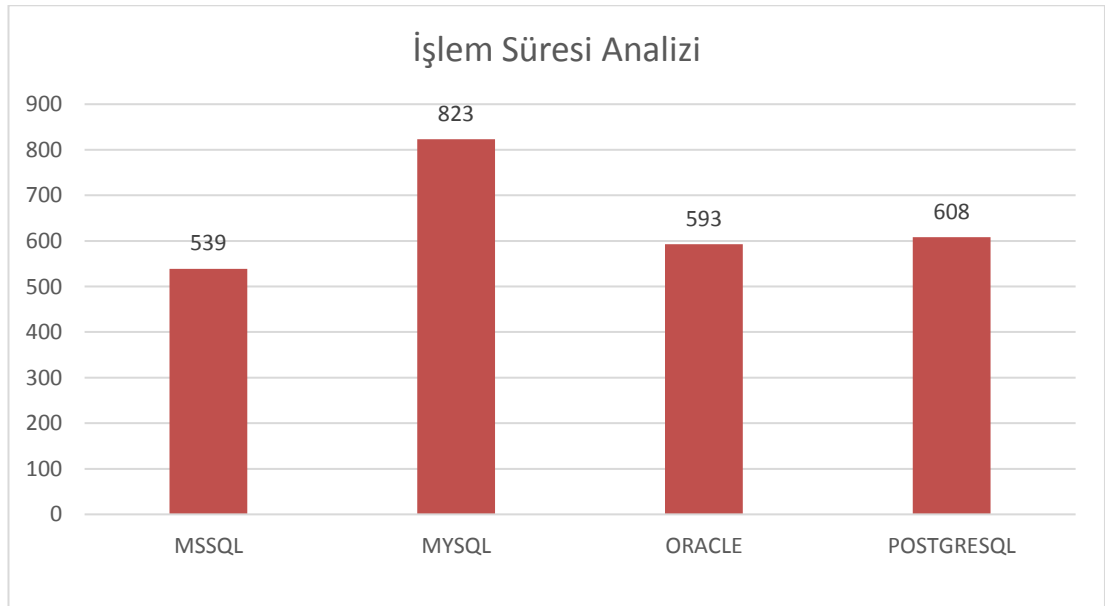
işlem süresi hemde ram kullanım oranı göz önüne alındığında veri tabanı engine ram analizine göre en iyi performansı ORACLE veri tabanı göstermiştir.

5.7. Tüm Testlerin Ortak İncelenmesi

Kayıt ekleme, silme, güncelleme ve sıralama işlemleri için kullanılan veri tabanlarının işlem süreleri, ortalama cpu ve ram değerleri analiz edilmiştir. Analiz sonuçlarının detayları grafik üzerinden açıklanmıştır.

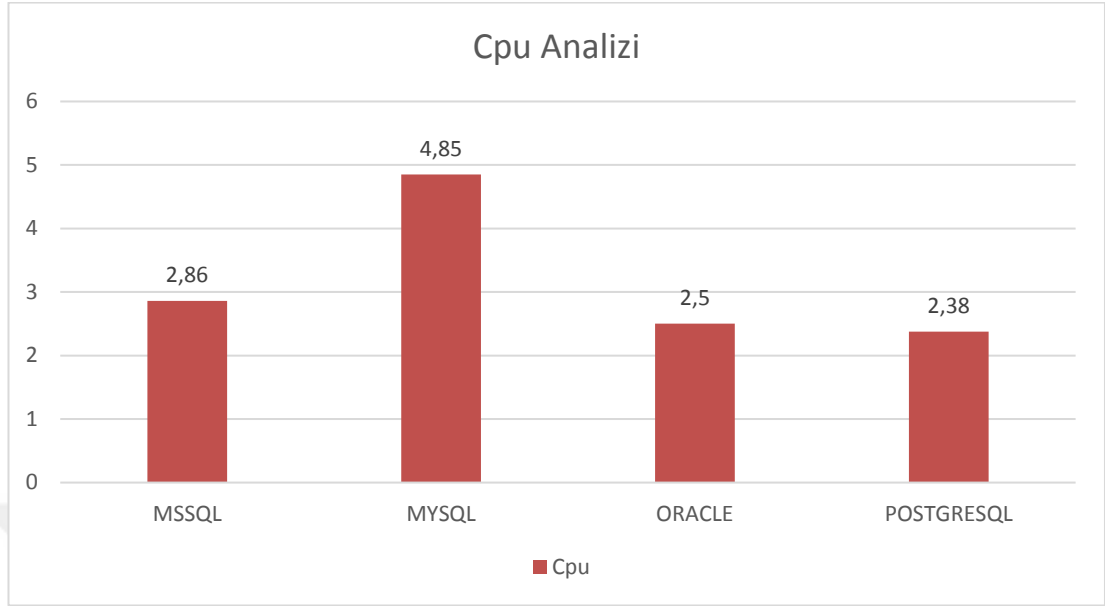
5.7.1. Tüm Testler İşlem Süresi Analizi

Şekil 5.17'deki ortalama işlem süreleri analizine göre MSSQL veri tabanı kayıt ekleme, silme, güncelleme işlemlerinde en iyi performansı göstermiştir. MYSQL veri tabanı ekleme, silme, güncelleme işlemlerinde en kötü performansı göstererek en uzun sürede işlemi tamamlamıştır. ORACLE veri tabanı bütün işlemlerde MSSQL'den sonra iyi bir performans göstermiştir. POSTGRESQL veri tabanı ise MYSQL'den iyi olmasına karşın diğer veri tabanlarından kötü bir performans göstermiştir.



Şekil 5.17 Tüm Test Ortalamaları İşlem Süresi Analizi

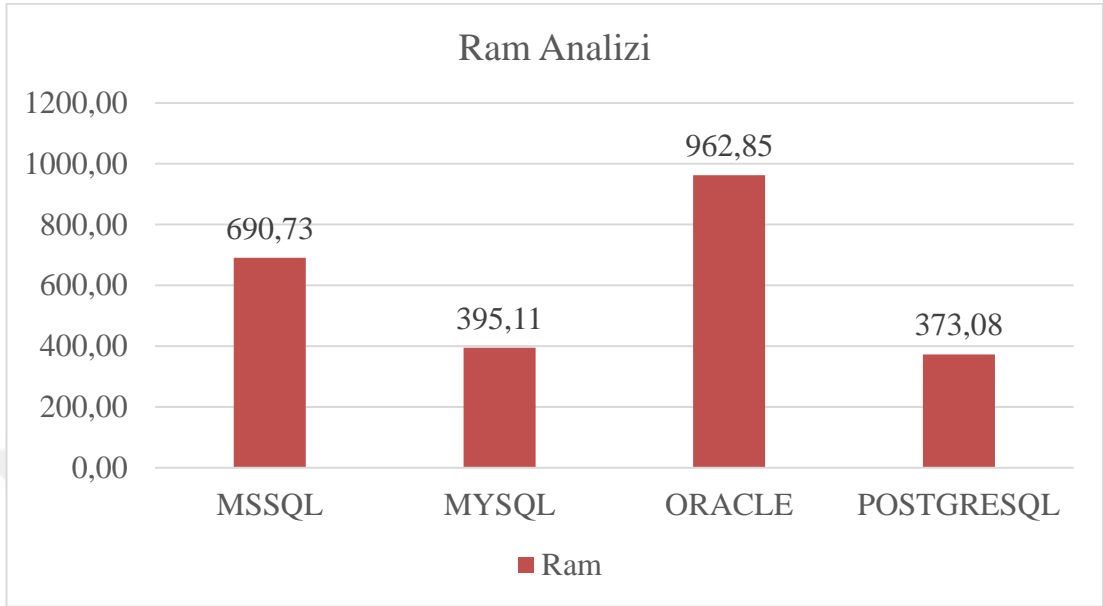
5.7.2. Tüm Testler İşlem Cpu Analizi



Şekil 5.18 Tüm Test Ortalamaları Cpu Analizi

Şekil 5.18'deki ortalama cpu değerlendirmelerinde ise karşılaştırma işlemi uygulama ve veri tabanı sistem cpu değerlerinin ortalaması üzerinden yapılmıştır. Buna göre POSTGRESQL veri tabanı en az cpu kullanarak işlemi gerçekleştirirken MYSQL en fazla cpu kullanmıştır. MSSQL ve ORACLE veri tabanları birbirine yakın değerlerde olup POSTGRESQL ve MYSQL değerleri arasında cpu kullanmıştır. Bu değerlere bakıldığında, en düşükten en yükseğe doğru POSTGRESQL (%2,38), ORACLE (%2,50), MSSQL (%2,86), MYSQL (%4,85) olarak bulunmuştur.

5.7.3. Tüm Testler İşlem Ram Analizi



Şekil 5.19 Tüm Test Ortalamaları Ram Analizi

Şekil 5.19'daki ortalama ram değerlendirmelerine bakıldığında uygulama ve veri tabanı sistemi ram ortalama değerlerine göre POSTGRESQL veri tabanı en düşük ram kullanımını gerçekleştirmiştir. Ram bakımından en yüksek kullanıma sahip olan ORACLE veri tabanıdır. MYSQL ve MSSQL veri tabanları ise ORACLE veri tabanına göre daha az ram kullanımı ile işlemleri tamamlamışlardır.

6. SONUÇLAR

Entity Framework'ün farklı veri tabanları ile çalışma performansı bu çalışma kapsamında analiz edilmiştir. Bu analiz için oluşturulan test uygulaması ile 5 kategori altında 20 farklı test yapılmıştır. Diğer bir uygulama ile bu test sonuçları kayıt altına alınarak analiz edilmiştir. Bu testler MSSQL, MYSQL, ORACLE, POSTGRESQL veri tabanları üzerinde yapılmış olup, kayıtların eklenmesi, silinmesi, güncellenmesi, aranması ve sıralanması işlemleri üzerinden işlem süresi, cpu ve ram değerlerine göre karşılaştırması gerçekleştirilmiştir.

Yapılan 5 farklı işlem üzerinde zamansal karşılaştırma yapıldığında MSSQL veri tabanı kayıt ekleme, silme, güncelleme ve sıralamada en iyi performansı göstermiş olup en kısa sürede işlemleri gerçekleştirmiştir. MYSQL veri tabanı ekleme, silme, güncelleme işlemlerinde en kötü performansı göstererek en uzun sürede işlemi tamamlamıştır. Sıralama işleminde ise POSTGRESQL'den daha iyi diğer veri tabanlarından daha kötü bir performans göstermiştir. ORACLE veri tabanı kayıt sıralama işleminde en iyi süreye sahip olup, diğer işlemlerde MSSQL'den sonra en iyi performansı göstermiştir.

Cpu değerlendirmelerinde ise karşılaştırma işlemi uygulama ve veri tabanı sistem cpu değerlerinin ortalaması üzerinden yapılmıştır. Buna göre POSTGRESQL veri tabanı en az cpu kullanarak işlemi gerçekleştirirken MYSQL en fazla cpu kullanmıştır. MSSQL ve ORACLE veri tabanları birbirine yakın değerlerde olup POSTGRESQL ve MYSQL değerleri arasında cpu kullanmıştır. Bu değerlere bakıldığında, en düşüğe doğru POSTGRESQL (%2,38), ORACLE (%2,50), MSSQL (%2,86), MYSQL (%4,85) olarak bulunmuştur.

Ram değerlendirmelerine bakıldığında uygulama ve veri tabanı sistemi ram ortalama değerlerine göre POSTGRESQL veri tabanı 373.08 mb ile en düşük ram kullanımını gerçekleştirmiştir. Ram bakımından en yüksek kullanıma sahip olan ORACLE veri tabanı 962,85 mb ile performans ortaya koymuştur. MYSQL ve MSSQL veri tabanları ise sırasıyla 395.11 mb ve 690.73 mb ram kullanımları ile ORACLE veri tabanına göre daha az ram kullanımı ile işlemleri tamamlamışlardır.

İşlem Süresi, Cpu ve Ram değerlerine bakıldığında MSSQL veri tabanı diğer veri tabanlarına göre ortalamada daha fazla ram ve cpu kullanmış olsa da işlem süresinin daha etkin bir parametre olmasından dolayı daha iyi performans göstermiştir. MYSQL veri tabanı ise en düşük sonuçlara sahip olarak diğer veri tabanlarına göre en düşük performans ortaya koymuştur. ORACLE en yüksek ram kullanıma sahip olmakla birlikte MSSQL'den sonra iyi bir performans ortaya koymuştur. POSTGRESQL ise cpu ve ram bakımından en iyi değerlere sahip olmasına karşın, işlem süresi bakımından MYSQL'den daha iyi performans göstermiştir.

Bu araştırmada Entity Framework'un farklı veri tabanları ile performans analizi yapılmıştır. Bu amaca ulaşmak için çalışma sürecinde bir yazılım geliştirilmiştir. Geliştirilen yazılım, profesyonel bir yazılım geliştirme süreci takip edilerek tamamlanmıştır. Yazılım geliştirme süreçlerini yönetmek için şelale modeli tercih edilmiştir. Geliştirilen yazılım Çizelge 4.4'de belirtilen donanımlara sahip iş istasyonu bilgisayar üzerinde kayıt ekleme, silme, güncelleme arama ve sıralama gibi temel veri tabanı işlemleri yapılırken İŞLEM SÜRESİ, RAM, CPU gibi kaynakların kullanımı karşılaştırmalı olarak analiz edilebilmiştir. Benzer çalışmaları yapmak isteyen araştırmalar aşağıdaki önerileri dikkate alabilir:

- Farklı ORM araçları üzerinde performans analizi yapılabilir
- Entity Framework'ün ve veri tabanlarının yeni versiyonlarında benzeri performans analizleri yapılabilir.
- Yazılımın çalıştırıldığı donanımlar ve işletim sistemleri değiştirilerek benzer analizler yapılabilir.

7. KAYNAKÇA

- [1] M. ERTEN, «Bölüm 1 Başlarken Temel Kavramlar,» *Bilgisayar Mühendisliğine Giriş*, R. ÇÖLKESEN, Dü., İstanbul, Papatya Yayıncılık, 2016.
- [2] S. ÖZTÜRK ve H. E. ATMACA, «İlişkisel ve İlişkisel Olmayan (NoSQL) Veri Tabanı Sistemleri Mimari Performansının Yönetim Bilişim Sistemleri Kapsamında İncelenmesi,» *Bilişim Teknolojileri Dergisi*, cilt 10, no. 2, pp. 199-209, Nisan 2017.
- [3] R. Ramakrishnan ve J. Gehrke, *DATABASE MANAGEMENT SYSTEMS*, 2003.
- [4] T. Kellegöz ve T. Çetinyokuş, *Veritabanı Sistemleri: Teoriden Uygulamaya*, Ankara: Nobel Akademik Yayıncılık, 2017.
- [5] Microsoft, «Entity Framework Belgeleri,» 2018. [Çevrimiçi]. Available: <https://docs.microsoft.com/tr-tr/ef/>. [Erişildi: 05 06 2018].
- [6] S. Berksoy, *C# Tam Hakimiyet Herkes İçin Uygulamalı Adım Adım .Net 4.0 ile Programlama.*, Ankara: Dikeyksen Yayınları, 2014.
- [7] «ADO.NET,» 20 06 2018. [Çevrimiçi]. Available: <https://docs.microsoft.com/tr-tr/dotnet/framework/data/adonet/>. [Erişildi: 12 06 2018].
- [8] «Micsoroft Sql Server,» 15 06 2018. [Çevrimiçi]. Available: <https://www.microsoft.com/tr-tr/sql-server/sql-server-2017>. [Erişildi: 15 06 2018].
- [9] A. ÇETİN, *Kitabevi Bayisi Oline Sok/Sparış Tkip Sstemi Yüksek Lisans Tezi*, İstanbul: Haliç Üniversitesi Fen Bllimleri Enstitüsü, 2017.
- [10] B. DEMİR, *Oracle veritabanı güvenliği Yüksek Lisans Tezi*, KOCAELİ: Kocaeli Üniversitesi Fen Bilimleri Enstitüsü, 2010.
- [11] T. H. ÖZTÜRK, *Veritabanı Yönetimine Giriş Oracle Database 11g R2*, İstanbul: Abaküs Yayınları, 2015.
- [12] «wikipedia PostgreSQL,» 20 6 2018. [Çevrimiçi]. Available: <https://tr.wikipedia.org/wiki/PostgreSQL>. [Erişildi: 10 06 2018].

- [13] M. ŞAMLI, Mysql ve Postgresql ile Veritabanı Programlama, İSTANBUL: Pusula Yayıncılık, 2005.
- [14] M. E. SARIDOĞAN, İstanbul: Papatya Yayıncılık, 2017.
- [15] Ş. E. ŞEKER, «MISSözlük Şelale Modeli (Waterfall Model),» [Çevrimiçi]. Available: <http://mis.sadievrenseker.com/2015/01/selale-modeli-waterfall-model/>. [Erişildi: 26 06 2018].
- [16] G. YILMAZ, «Yazılım Geliştirme Yaşam Döngüsü,» [Çevrimiçi]. Available: web.iku.edu.tr/~gyilmaz/Notes/YazilimMuhendisligiYonetimi/Bolum-02.ppt. [Erişildi: 24 05 2018].
- [17] N. E. ŞAVKLI, Her Yönüyle Entity Framework 5.0 ve Uygulamalar, İSTANBUL: Pusula Yayıncılık, 2014.
- [18] V. ÖKCÜ, *PHP ve MYSQL tabanlı uygulamalarda performans yönetimi Yüksek Lisans Tezi*, İstanbul: Bahçeşehir Üniversitesi Fen Bilimleri Enstitüsü, 2016.
- [19] N. DİKER, G. GİRAY ve M. O. ÜNALIR, «Servis Tabanlı Bir Melez Veri Erişim Mimarisi Önerisi,» *10. Ulusal Yazılım Mühendisliği Sempozyumu*, Çanakkale, 2016.