



## Defter-Beyan Sistemi: Açık Kaynak Temelli Geliştirilen Büyük Ölçekli Bir Kamu Projesi

Yasin Şahin<sup>a1,\*</sup>, İsmail Arık<sup>b2</sup>, Kürşat Sakallı<sup>b3</sup>

<sup>a</sup> Medyasoft, Üniversiteler Mah. 1596. Cad. Hacettepe Teknokent 6. Ar-Ge C Blok Kat:6 No:66-67, Ankara, Türkiye

<sup>b</sup> Medyasoft, Altunizade Mahallesi, Ord. Prof. Fahrettin Kerim Gökay Caddesi, Eşref Çakmak Plaza, No:32 Kat:2 34662 Üsküdar/Istanbul, Türkiye

Istanbul Sabahattin Zaim Üniversitesi Fen Bilimleri Enstitüsü Dergisi (2021) 3 (1): 60-65

<https://doi.org/10.47769/izufbed.868873>

ORCID <sup>1</sup> 0000-0001-9586-9822; <sup>2</sup> 0000-0001-8327-8881; <sup>3</sup> 0000-0002-1808-3735

### YAYIN BİLGİSİ

Yayın geçmişi:

Gönderilen tarih: 01 Şubat 2021

Kabul tarihi: 18 Şubat 2021

### Anahtar kelimeler:

Kamuda Açık Kaynak

Büyük Ölçekli Web

Uygulamaları

### ÖZET

Açık kaynak ve özgür yazılımın yüksek standartları dünya çapında yaygın olarak kullanılmaktayken ülkemizde yükseliş trendi biraz daha geriden gelmektedir. Buna rağmen kamuda da kullanımı yaygınlaşmakta, iyi örnekler her geçen gün artmaktadır. Bu çalışmada, Gelir İdaresi Başkanlığı tarafından hizmete sunulan Defter-Beyan Sistemi'nin açık kaynak kapsamı iyi bir mühendislik örneği olarak anlatılmaktadır. Yalnızca geliştirme sırasında kullanılan bileşenler değil, sistemin hayata geçirilmesinden sonra sağlık taraması ve izleme amaçlı kullanılan araçlar ile ilgili bilgilendirmelere de yer verilmiştir. Yapılan çalışmalardan elde edilen sonuçlar gösteriyor ki, tamamen açık kaynak ile geliştirilmiş ve sürdürülen bir kamu projesi doğru mühendislik uygulamaları ile birleştiğinde ticari herhangi bir alternatifine ihtiyaç duymadan sürdürülebilmektedir.

## Defter-Beyan Sistemi: A Large-Scale Public Project Developed Based on Open Source

### ARTICLE INFO

Article history:

Received: 01 February 2021

Accepted: 18 February 2021

### Key words:

Open Source in The Public

Large-Scale Web Applications

### ABSTRACT

While the high standards of open source and free software are widely used around the world, the upward trend in our country is a little behind. Despite this, its use is becoming widespread in public, and good examples are increasing day by day. In this study, the open source scope of Defter Beyan Sistem put into service by the Revenue Administration is described as a good example of engineering. Not only the components used during the development, but also the information about the tools used for health screening and monitoring after the implementation of the system. The results obtained from the studies show that a public project developed and maintained with open source, combined with the right engineering practices, can be sustained without the need for any commercial alternative.

### 1. Giriş

Açık kaynak kodlu yazılımlar, doğrudan son kullanıcıya hizmet eden ürünler olduğu gibi (Hall vd., 2009; Momjian, 2001), büyük çoğunluğu bir yazılım projesinde mimari bir bileşeni (Johns, 2013; Kiczales vd., 2001) ya da işlevsel bir gereksinimi karşılamak (Friesen, 2016; Reese, 2000) için kullanılan güvenilir parçalar olarak yer alabilmektedir. Çok nadiren açık kaynak bir yazılım başka bir açık kaynak projeye içerisinde yer vermez. Yani, açık kaynak projelerin büyük çoğunluğu başka açık kaynak projelerin yeteneklerini alıp

yeni yetenekler sunarlar. Bu da özgür yazılım dünyasının giderek büyüyen, kendini besleyen bir anlayış olduğunu gösteriyor bize.

Kapalı kaynak kodlu sistemlerin yerini çok hızlı bir şekilde alan açık kaynak kodlu projeler tarihsel olarak birbirine güvenmesini beklemeyeceğimiz ülkelerin stratejik firmalarını sponsorluklarla bir araya getirebilmektedir. Örneğin Postgresql aynı anda Amerika, Rusya ve pek çok ülkeden sponsorları aynı anda bir araya getirebilmiştir (Postgresql, 2020).

Açık kaynak yazılımlara olan devlet desteği ve teşvikleri uzun

\* Sorumlu yazar.

E-mail adresi: [yasinhun@gmail.com](mailto:yasinhun@gmail.com) (Yasin Şahin)

yıllardır uluslararası düzeyde karşımıza çıkmaktadır (Comino & Manenti, 2005; Evans & Reddy, 2002; Lewis, 2010). Ülkemiz de bu raporlarda ve çalışmalarda Pardus projesi ile zaman zaman yer alabilmektedir (Pardus, n.d.). Son dönemlerde hem ilgili bakanlık tarafından yönetilen Türkiye Kamu Açık Kaynak Platformu (Türkiye Açık Kaynak Platformu, n.d.) hem de Cumhurbaşkanlığı tarafından yayınlanan yıllık planlarda açık kaynak dönüşümünün yer alması ulusal anlayışımızın açık kaynak yazılımlar lehine geliştiğini göstermektedir (Türkiye Cumhuriyeti Cumhurbaşkanlığı Strateji ve Bütçe Başkanlığı, 2019). Bu çalışmanın geri kalanında sırası ile proje ve müşteri hakkında bilgilendirme yapılacak, yazılım ve sisteme dair mimari açıklamalara yer verilecek, son bölümde de açık kaynak yazılımlar ile yürütülen projemizin hangi büyüklüğe ve ölçüğe ulaştığı anlatılacaktır.

## 2. Materyal ve Yöntem

### 2.1 Güncel Çalışma

Gelir İdaresi Başkanlığı bünyesinde 2017 yılında geliştirmeye başlanan, 2018 yılında mükelleflerin hizmetine sunulan bir sistem olan Defter-Beyan Sistemi; serbest meslek erbabı, işletme hesabı esasına göre defter tutan mükellefler ile basit usule tabi olan mükelleflerin kayıtlarının elektronik ortamda tutulmasına, bu kayıtlardan serbest meslek erbabı ile işletme hesabı esasına göre defter tutanların defterlerinin elektronik olarak oluşturulması ve saklanmasına, vergi beyannameleri, bildirim ve dilekçelerin elektronik olarak verilebilmesine, elektronik ortamda belge düzenlenebilmesine imkan tanıyan ve kapsamda yer alan mükelleflerin Gelir İdaresi Başkanlığı veri tabanında yer alan diğer bilgilerini mükelleflerin bilgisine sunan sistemdir.

Defter-Beyan Sistemi'nin bir bilgisayar sistemi olarak en önemli özelliklerinden birisi, belki de en önemlisi, uygulamanın sürüm yönetimi, sunucu işletim sistemleri, yazılım teknolojileri ve bileşenleri, veri tabanı yönetim sistemi, canlı sistem izleme araçları gibi akla gelebilecek bütün bileşenlerinin açık kaynak yazılımlardan yararlanılarak geliştirilmiş, kurgulanmış ve yaşamını sürdürüyor olmasıdır.

### 2.2 Kullanılan Teknolojiler ve Mimari

#### 2.2.1. Yazılım Teknolojileri

2.2.1.1 *Java*: İlk olarak 1995'te Sun Microsystems tarafından piyasaya sürülen bir programlama dili ve bilgi işlem platformudur (Oracle, n.d.). Java Runtime Environment (JRE) sayesinde geliştirilen uygulamaların işletim sisteminden bağımsız olarak çalışmasını sağladığı için geliştiricilere esneklik sağlar.

2.2.1.2 *Spring Framework*: Modern kurumsal uygulamalar için kapsamlı bir geliştirme altyapısı sağlar. Java dünyasında çok popülerdir. Geliştiricilere düşük maliyetlerle (Transaction yönetimi, Dependency Injection, oturum yönetimi, kimlik doğrulama, yetkilendirme gibi çoğu kurumsal uygulamada yer alan işlevlerden soyutlayarak) geliştirdikleri uygulamanın iş mantıklarına odaklanmalarını sağlar (Spring, 2021).

2.2.1.3 *Maven*: Java uygulamalarının geliştirilmesinde

harcanan eforları en aza indirmeyi hedefleyen bir Apache kütüphanesidir. Maven uygulamanın derlenmesini kolaylaştırır, tekil bir geliştirme ortamı sunar, kütüphane bağımlılıklarını yönetir, proje kalite araçlarının kullanılmasına olanak sağlar ve tüm bu işlevler için eklenti altyapısı sunar (Apache Maven Project, 2021).

2.2.1.4 *Nexus Repository (Sonatype, 2008)* : Java, NuGet, Docker, npm vs. ürün formatlarının çıktılarını sürümler olarak saklayan, yerel ve genel repolarda bulunan kütüphaneleri derlenecek uygulamalara sağlayan açık kaynak sistemdir (IBM, n.d.).

2.2.1.5 *Hazelcast IMDG*: Java, C#, Node.js, Python gibi pek çok dil ile çalışabilen, çoğu veri tipi için saklama ve mesajlaşma kuyruk yapıları özelliği sağlayan, buluta hazır, kolayca ölçeklendirilebilen, sunucular arası otomatik veri taşıma, bölümlenme işlerini yapan açık kaynak bir bellek içi (in-memory) veri tabanı sistemidir (Hazelcast, n.d.).

2.2.1.6 *Apache Commons DBUtils*: JDBC (Kiczales vd., 2001) kütüphaneleri ile çalışmayı kolaylaştırmak amacı ile yazılmış bir açık kaynak kütüphanedir. Bu kütüphane sayesinde JDBC'de hatalı kaynak kullanımlarının önüne geçilmektedir. Sorgu sonuçlarının nesnelere dönüştürülmesini kolaylaştırır (Apache, n.d.).

2.2.1.7 *Spring Boot & Tomcat (Gömülü)*: Spring projelerini tek başına kolayca çalışır şekilde paketleyebilmek amacıyla yazılmış bir kütüphanedir. Bunun için gömülü Tomcat uygulama sunucusunu kullanır. Bu kütüphaneyi kullanan Spring uygulamalarının ayarlanması ve başlatılması çok kolay ve hızlıdır (Spring, 2021b).

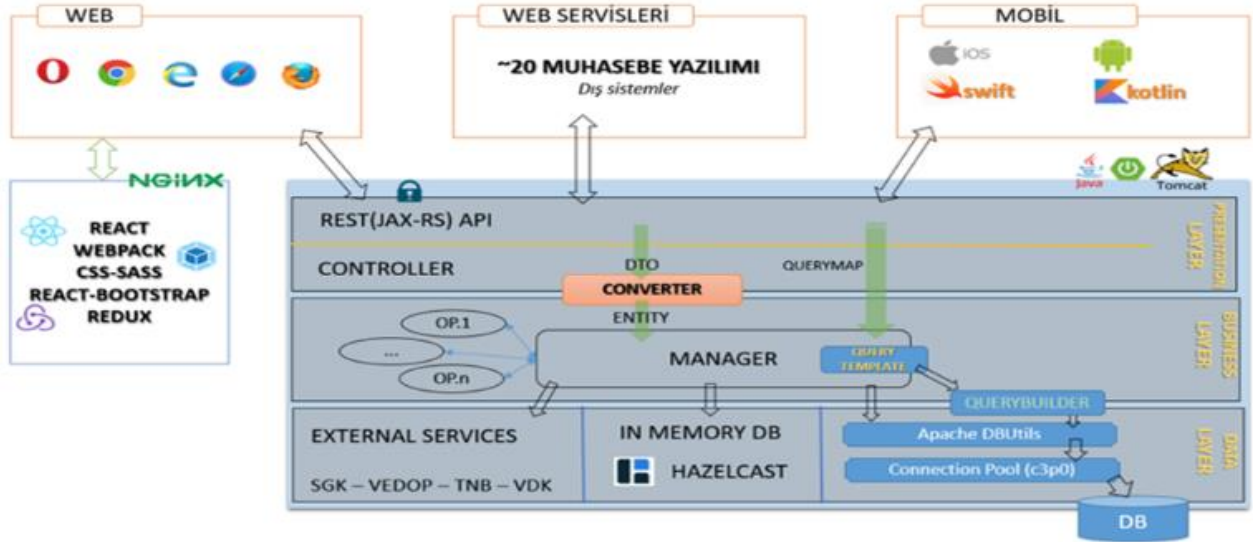
2.2.1.8 *React*: Facebook tarafından geliştirilen, bildirim dayalı, bileşen temelli, bir kere yaz her yerde kullan mantığı ile geliştirilmiş bir Javascript kütüphanesidir (Eisenman, 2015). React ile etkileşimli bir şekilde, yalnızca veri değişimi ile ilgili bileşenlerin etkileneceği şekilde geliştirildiği için yüksek başarılı, öngörülebilir ve test edilebilir bileşenler yazılmasına olanak sağlar.

2.2.1.9 *Redux*: React ile birlikte bileşenler arası veri iletişimine kolaylaştıran bir kütüphanedir. Uygulamaların daha derli toplu olabilmesi için Flux tasarım örüntüsü üzerine inşa edilmiştir. Redux sayesinde tahmin edilebilir, test edilebilir, verinin merkezileştiği, debug edilebilir ve esnek kodlar yazmak mümkündür (Redux, 2021)

Bunun dışında Junit (Junit, 2021) , Mockito (Mockito, 2021) , Gson (Google, 2020), Jackson (Reese, 2000), Aspectj (Comino, & Manenti, 2005), Log4j (Apache, 2020) gibi pek çok kütüphane proje içerisinde farklı seviyelerde ve yoğunluklarda kullanılmıştır.

#### 2.2.2. Teknoloji Mimarisi

Defter-Beyan Sistemi'nin üst düzey teknoloji mimarisi Şekil 1'de yazılım mimarisi ayrıntılarına girilmeden gösterilmiştir. Çok katmanlı mimari ile geliştirilen Defter-Beyan Sistemi'nin farklı istemci çeşitlerine hizmet verecek şekilde modellenmiş gösterimi ve kullanılan teknolojiler Şekil 1'de konumlandırılmıştır.



Şekil 1. Teknoloji Mimarisi

### 2.2.3. Yazılım Sürüm Yönetimi

2.2.3.1 *Git&Gitlab*: Git ücretsiz ve açık kaynak kodlu, küçükten büyüğe her ölçekte proje için paralel dallanmalara ve ayrı yerel çalışma ortamlarına izin veren bir sürüm yönetim sistemidir. Gitlab ise CI/CD süreçlerinin oluşturulmasını sağlayan, Git üzerinde sürümleme yapan, kaynak kod yönetiminin ve güvenliğinin sağlandığı bir sistemdir (Gitlab, n.d.).

2.2.3.2 *Liquibase*: Yazılım uygulamaları genelde bir ya da birkaç veri tabanına bağlı çalışır ve yaygın olarak karşılaşılan sorunlardan birisi geliştirme ortamlarında yapılan değişikliklerin yansması için veri tabanında işletilmesi gereken betiklerin (script) program sürümü ile birlikte canlı sisteme ya da ilgili test sistemlerine eş zamanlanamamasıdır. Liquibase çeşitli veri tabanı yönetim sistemleri ile çalışabilen, veri tabanı ile kod uyumunu sağlayan açık kaynak bir kütüphanedir (Liquibase, 2021).

### 2.2.4. Sistem Mimarisi ve Donanım

Defter-Beyan Sistemi uygulama sunucuları, Intel(R) Xeon(R) Platinum 8168 CPU @ 2.70GHz (258.62 GHz, 1,023.44 GB, 20.54 TB) yapılandırması ile DELL EMC marka fiziksel sunucularında sanal olarak bölümlenmiş şekilde kurgulanmıştır. Her bir sanal makine Centos 7.4 üzerinde çalışmaktadır. JRE sürümü 1.8'in güncel paketidir. Arka-uç uygulama sunucusu olarak hazırlanan makinelerin her biri 25GB disk, 8 CPU ve 24GB RAM ile sanallaştırılmıştır. Defter-Beyan Sistemi bağlamında kullanılan sistem sunucuları Tablo 1'de gösterilmiştir.

Şekil 2'de test ortamları, intranet sistemler ile entegrasyonlar, izleme ortamları ve pek çok DBS içi ağ etkileşiminin topolojisini görebilirsiniz.

### 2.2.5. Veri Tabanı Yönetim Sistemi ve Veri Tabanı Sistem Mimarisi

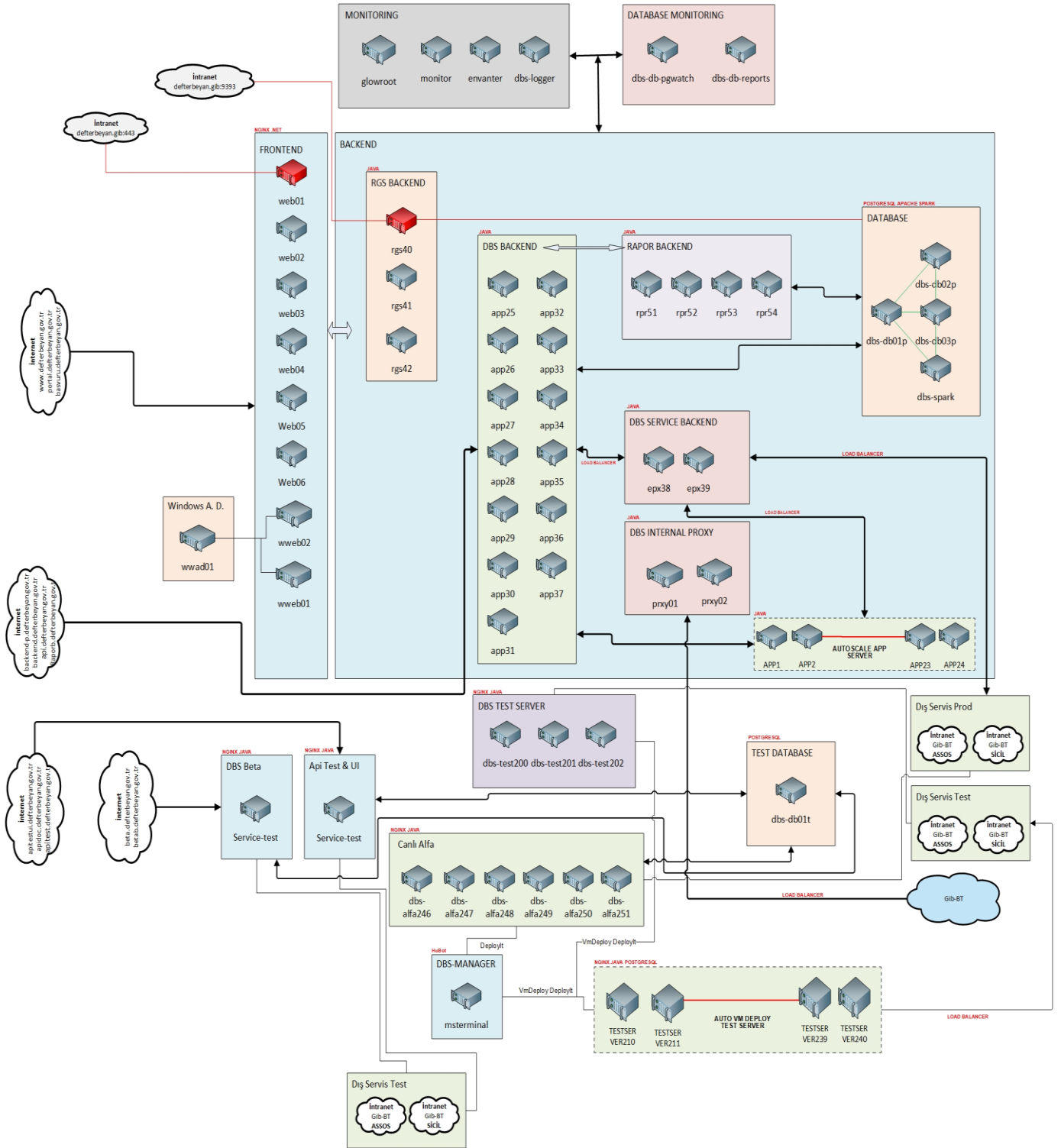
Bu projede veri tabanı yönetim sistemi olarak Postgresql kullanılmaktadır. Postgresql (Postgresql, 2020b),

Berkeley'deki California Üniversitesi'nde geliştirilen POSTGRES (Stonebraker, Rowe ve Hirohama, 1990) paketinin üzerine geliştirilmiş bir ilişkisel veri tabanı yönetim sistemidir. Bu yazı kaleme alındığı an itibarı ile 10.3 sürümü kullanılmaktadır. Üzerinde koştuğu işletim sistemi ise Centos 7.4'tür.

Tablo 1. Sunucu donanımları

Sayı	Kısa Kod	Açıklama	İşleyici Çekirdek	Anabellek
6	web	Ön-uç web sunucusu	6	8 GB
13	app	Uygulama sunucusu	8	24 GB
2	epx	Test sunucusu	8	24 GB
2	proxy	Temsilci sunucu	4	8 GB
3	rgs	Başvuru uygulaması sunucusu	8	24 GB
1	wad	Yönetici uygulama sunucuları	4	4 GB
2	wweb	Yönetici uygulama web sunucuları	4	4 GB
24	app	Autoscale için bekletilen uygulama sunucuları	8	24 GB

Veri tabanının üzerinde çalıştığı bir ana sunucu, bir gerçek zamanlı stand-by sunucusu, mantıksal kesintilerde devreye alınmak üzere 1 saat geriden gelen yedek sunucu ve iki adet uzak (felaket kurtarma amaçlı) stand-by sunucudan



Şekil 2. Sistem ağ topolojisi

oluşmaktadır. Söz konusu mimari Şekil 3’de görülmektedir.

Veri Tabanı Bakımı ve İzleme: Sistem üzerinde yapılan rutin çalışmalar şu şekildedir:

- i. Her gün 03:00’te vakumlama yapılmaktadır.
- ii. Her gün saat 07:00’de ve 12:00’de durum analizi çalıştırılmaktadır.
- iii. Her gün saat 00:05’te base-backup alınmaktadır

iv. Arşiv günlükleri bu yedek üzerine alınarak geri dönüş senaryoları bu arşivler üzerinden test edilmektedir.

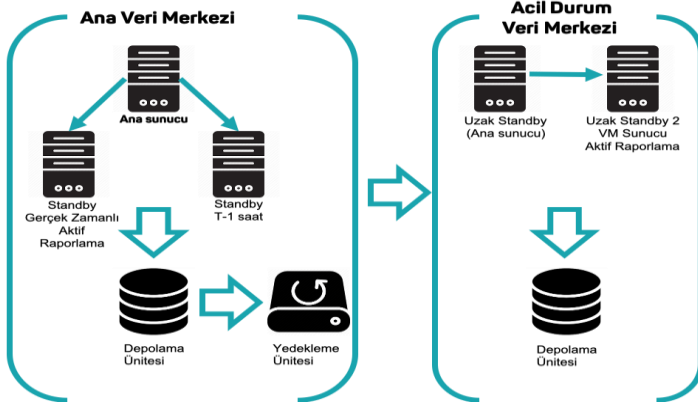
Veri tabanı yöneticileri tarafından izleme amaçlı kullanılan araçlar:

- i. PgWatch (PgWatch., n.d.): Ana sunucuda Docker üzerinde çalışan PgWatch, Grafana’da takip edilmek üzere belirli aralıklarla veri toplayarak yine Docker üzerinde çalışan bir veri tabanına bu verileri yazmaktadır.
- ii. Grafana: Farklı kaynaklardan verileri toplayarak tek bir

izleme aracı üzerinden görselleştirerek veri tabanı ve sunucu üzerindeki hareketleri görselleştiren açık kaynak kodlu bir uygulamadır (Grafana, n.d.).

- iii. Zabbix: Sunucu ve ağ üzerindeki servisleri, donanımları izlemek amacıyla geliştirilmiş olan açık kaynak kodlu üründür (Zabbix, 2020).
- iv. PgBadger: PostgreSQL servisinin günlüklerinin çıktısını alan ve onu bir HTML dosyasına işleyen Perl’de yazılmış, açık kaynak kodlu bir günlük analiz programıdır (Pgbadger, 2021).

Son olarak Defter-Beyan Sistemi veri tabanı önemli yapılandırma ayarları Tablo 2’de verilmiştir.



Şekil 3. Veritabanı sistem mimarisini

Tablo 2. Veritabanı yapılandırması

Parametre	Değer
shared_buffers	256GB
huge_pages	on
work_mem	1398kB
maintenance_work_mem	2GB
wal_buffers	256MB
max_wal_size	8GB
min_wal_size	6GB

### 3. Bulgular ve Tartışma

2018 yılında yaklaşık 150.000 mükellef ile ilk kez kullanıma açılan Defter-Beyan Sistemi, kapsamı planlı şekilde genişletilerek 2019 yılında 2.2 milyon, 2020 yılında ise 2.5 milyon mükellefe hizmet verir hale gelmiştir. Her ay vergi doğuran hareketlerini kayıt altına alan mükelleflerin takip eden dönemde beyannamelerini vermeleri beklenmektedir. Kullanıcı davranışı olarak beyannamenin son 3 günü yüksek bir kullanım hacmi oluşmaktadır. Aylık ortalama 3 milyonun üzerinde beyanname Defter-Beyan Sistemi üzerinden verilmektedir. Kullanıcılar tarayıcı üzerinden sağlanan kullanıcı arayüzleri aracılığı ile ya da üçüncü taraf uygulamalara sağlanan Defter-Beyan Sistemine ait API (Application Programming Interface) üzerinden kayıt yapılabilmektedir. Aynı yoğun geçen bu üç gününde kullanıcılar hem kayıtlarını sisteme yüklüyorlar hem de bu kayıtlardan otomatik vergi beyannameleri oluşturularak Gelir İdaresi Başkanlığı iç sistemleri ile gerekli bilgiler paylaşılıyor.

Defter-Beyan Sistemi veri tabanında en büyük boyutlu beş tablonun istatistikleri Tablo 3’te verilmiştir.

Görüldüğü üzere PostgreSQL bir veri tabanında yaklaşık 250

milyar satırlık bir tabloyu ya da başka bir ölçüm değeriyle 220 GB’lık bir tabloyu güvenle kullanabiliyoruz. Bu veriler içerisinde medya gibi hiçbir ikili(binary) formatta tutulan sütun olmadığını vurgulamak uygun olacaktır.

Tablo 3. Veritabanı tablolarına ait fiziksel özellikler

	Disk Boyutu	Table Size
#1	220 GB	236,444,524,544
#2	214 GB	229,775,286,272
#3	134 GB	143,438,307,328
#4	129 GB	138,138,206,208
#5	101 GB	108,740,902,912

Toplam veri hacminin 3,718 GB olduğu veri tabanında TPS (transaction per sec.) 4,000-6,000 arasında olurken, beyanname dönemlerinde 10,000 ile 20,000 arasında değişmektedir.

Sistem en yoğun anında toplam 13 arka-uç uygulama sunucusu (gömülü Tomcat) ile yaklaşık CPU %45 - 50 dolulukla çalışmaktadır ve aynı anda 8,000 kullanıcıya dakikada 75,000 işlemle cevap vermektedir. CPU kullanımı %60’ın üzerinde çıktığında otomatik ölçekleme yapılarak (auto-scale) her defasında yeni bir sunucu sisteme dahil olmakta ve CPU kullanımı %40’ın altına düştüğünde açılan uygulama sunucularının en sonuncusu silerek sistemden çıkarılmaktadır. Bu işlem ile sunuculara 24 ayı nitelikte sunucu eklenerek toplam 37’ye kadar çıkarılabilmektedir. Ön-uç web sunucusu olarak Nginx kullanılmaktadır ve 6 adet Web sunucusu ile hizmet vermektedir. Web sunucuları üzerinde herhangi bir otomatik ölçekleme ihtiyacı olmadığı için kurguda yer verilmemiştir.

### 4. Sonuç

Defter-Beyan Sistemi, devletin vergi başta olmak üzere tahsilatlarından sorumlu kritik derecede önemli bir kurumu olan Gelir İdaresi Başkanlığı gibi stratejik bir kurumda dahi tamamen açık kaynak yazılımlar üzerinde geniş ölçekli projelerin geliştirilebileceğini, güvenli ve tasarruflu bir şekilde idame ettirilebileceğini gösteren değerli bir örnek vaka olarak karşımıza çıkmaktadır. Defter-Beyan Sistemi yalnızca veri tabanı lisansı 5 yıllık sahipliği tutarının bile çok altında bir bütçe ile kamu yararına sunulmuş, açıldığı günden bu yana teknoloji kaynaklı hiçbir sorun yaşamamıştır. Bu avantajı sayesinde hizmet kalitesini artıracak donanım yatırımları daha kolay gerçekleşmiştir. 2.5 milyon mükellef, 20’den fazla entegratör ve 3 milyona yakın kullanıcıya hizmet veren bir sistemi anlatan bu çalışma ile açık kaynak destekli projelerin teşvik edilmesi, karar vericileri cesaretlendirmesi amaçlanmıştır.

### Kaynaklar

- Apache. (n.d.). DbUtils. <https://commons.apache.org/proper/commons-dbutils/Spring>. (2021).
- Apache. (2020, Kasım). Log4j. <https://logging.apache.org/log4j/2.x/>.
- Apache Maven Project. (2021, Ocak 15). What is maven?. <https://maven.apache.org/what-is-maven.html>.
- Comino, S., & Manenti, F. M. (2005). Government policies supporting open source software for the mass market. *Review of industrial organization*, 26(2), 217-240.
- Eisenman, B. (2015). *Learning react native: Building native*

- mobile apps with JavaScript. "O'Reilly Media, Inc."
- Evans, D. S., & Reddy, B. J. (2002). Government preferences for promoting open-source software: a solution in search of a problem. *Michigan Telecommunications and Technology Law Review*, 9, 313.
- Friesen, J. (2016). *Java XML and JSON*. New York, NY, USA: Apress.
- Git. (2020, Aralık). Git version control. <https://git-scm.com/>.
- Gitlab. (n.d.). The complete DevOps platform. <https://about.gitlab.com/>.
- Google. (2020, Mayıs). Gson. <https://github.com/google/gson>.
- Grafana. (n.d.). Grafana monitoring tool. <https://grafana.com/>.
- Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., & Witten, I. H. (2009). The WEKA data mining software: an update. *ACM SIGKDD Explorations Newsletter*, 11(1), 10-18.
- Hazelcast. (n.d.). Overview. <https://hazelcast.org/imdg/why/>.
- Junit. (2021). Junit. <https://junit.org/>.
- IBM. (n.d.). Nexus: An open source repository for build artifacts. [https://www.ibm.com/garage/method/practices/deliver/tool\\_nexus/](https://www.ibm.com/garage/method/practices/deliver/tool_nexus/).
- Johns, M. (2013). *Getting Started with Hazelcast*. Packt Publishing Ltd.
- Kiczales, G., Hilsdale, E., Hugunin, J., Kersten, M., Palm, J., & Griswold, W. G. (2001, June). An overview of AspectJ. In *European Conference on Object-Oriented Programming* (pp. 327-354). Springer, Berlin, Heidelberg.
- Lewis, J. A. (2010). *Government open source policies*. Center for Strategic and International Studies.
- Liquibase. (2021). Liquibase. <https://www.liquibase.org/>.
- Mockito. (2021). Tasty mocking framework for unit tests in Java. <https://site.mockito.org/>.
- Momjian, B. (2001). *PostgreSQL: introduction and concepts* (Vol. 192). New York: Addison-Wesley.
- Oracle. (n.d.). Java teknolojisi nedir?. [https://java.com/tr/download/help/whatis\\_java.html](https://java.com/tr/download/help/whatis_java.html).
- Pardus. (n.d.). Pardus. <https://www.pardus.org.tr/>.
- Pgbadger. (2021, Ocak). Pgbadger. <https://github.com/darold/pgbadger>.
- PgWatch. (n.d.). PgWatch. <https://pgwatch.com/>.
- Postgresql. (2020, Kasım). Sponsors. <https://www.postgresql.org/about/sponsors/>.
- Postgresql. (2020b, Kasım). PostgreSQL: The World's Most Advanced Open Source Relational Database. <https://www.postgresql.org/>.
- Reese, G. (2000). *Database Programming with JDBC and JAVA*. "O'Reilly Media, Inc."
- Redux. (2021). Redux. <https://redux.js.org/>.
- Sonatype. (2008). Nexus repository pro. <https://www.sonatype.com/nexus/repository-pro>
- Spring framework. <https://spring.io/projects/spring-framework>.
- Spring. (2021b). Spring boot. <https://spring.io/projects/spring-boot>.
- Stonebraker, M., Rowe, L. A., & Hirohama, M. (1990). The implementation of POSTGRES. *IEEE transactions on knowledge and data engineering*, 2(1), 125-142.
- Türkiye Açık Kaynak Platformu. (n.d.). Türkiye'de açık kaynaklı yazılım geliştirme kültürü, yazılım ekosistemi. <https://www.turkiyeacikkaynakplatformu.com/>
- Türkiye Cumhuriyeti Cumhurbaşkanlığı Strateji ve Bütçe Başkanlığı. (2019, Kasım). 2020 Yılı Cumhurbaşkanlığı Yıllık Programı. [https://sbb.gov.tr/wp-content/uploads/2019/11/2020\\_Yili\\_Cumhurbaşkanligi\\_Yillik\\_Programi.pdf](https://sbb.gov.tr/wp-content/uploads/2019/11/2020_Yili_Cumhurbaşkanligi_Yillik_Programi.pdf).
- Zabbix. (2020). Zabbix. <https://www.zabbix.com/>.