

Blok Zincir Ekonomisinde Yeni Uygulamalar: E-Fatura Üzerine Bir İnceleme¹

New Applications in Blockchain Economy: A Review on E-Invoice

Mansur BEŞTAŞ, Siirt Üniversitesi, Türkiye, mansur@siirt.edu.tr

Orcid No: 0000-0002-8192-2044

Öz: Blok zincir ve akıllı sözleşmeler birçok iş yapış şeklinde yerini bulmaya başlamıştır. Blok zincir teknolojisi kimi zaman verinin değiştirilmemesi amaçlı olarak kullanılmakta iken kimi zaman da merkezi yazılım kullanımına ait yatırım harcamalarını azaltmaya yönelik olmaktadır. Muhasebe alanında bulunan iş süreçleri dijitalleşmenin getirdiği faydalardan gittikçe daha fazla faydalanmaktadır. Ancak muhasebe ile ilgili ortaya çıkan çözümler merkezi yazılım mimarisine ait çözümler şeklinde olmaktadır. Muhasebe süreçlerinde oluşturulan faturaların yasal zorunluluktan dolayı belli süre boyunca saklanması ve bu faturaların kesinlikle değiştirilmemiş olması, garanti altına alınması bir gerekliliktir. Bu noktada Blok zincir teknolojisinin sağladığı akıllı sözleşmeler sayesinde süreçler yönetilmektedir. Verilerin saklanması amacıyla ek olarak merkezi olmayan depolama teknolojileri bir çözüm olarak kullanılmaktadır. Bu çalışmada e-faturanın merkezi olmayan uygulamalar ile nasıl işleneceği ve saklanacağına yönelik çözüm önerisi sunulmuş olup uygulama olarak hayata geçirilmiştir. Önerilen çözüm yöntemi blok zincir teknolojisi aracılığı ile muhasebe süreçlerinde ihtiyaç duyulan onaylama işlemlerini üçlü onay seviyesine taşımaya, merkezi olmayan uygulama mimarisinde e-fatura'nın süreçlerini ortaya koymasına, veri güvenliğinin yönetilmesine yardımcı olmaktadır.

Anahtar Kelime: Blok Zincir, Muhasebe, Fatura, Finans, Akıllı Sözleşme

JEL Sınıflandırması: M41, O31, M15

Abstract: Blockchain and smart contracts have begun to find their place in many ways of doing business. While the blockchain technology is sometimes used for the purpose of not changing the data, sometimes it is aimed at reducing the investment expenditures of the use of central software. Business processes in the field of accounting are increasingly benefiting from the benefits of digitalization. However, the solutions related to accounting are in the form of solutions belonging to the central software architecture. It is a requirement that the invoices created in the accounting processes are kept for a certain period of time due to legal obligations and that these invoices are not changed and guaranteed. At this point, processes are managed thanks to smart contracts provided by Blockchain technology. In addition, decentralized storage technologies are used as a solution for data storage. In this study, a solution proposal has been presented on how to process and store e-invoice with decentralized applications and has been implemented as an application. The proposed solution method helps to carry the approval processes needed in accounting processes to triple approval level, to reveal the processes of e-invoice in a decentralized application architecture, and to manage data security through blockchain technology.

Keywords: Blockchain, Accounting, Invoice, Finance, Smart Contract

JEL Classification: M410, O310, M150

¹ Bu çalışmada geliştirilmiş tüm kodlar <https://github.com/digifelis/efatura> adresinde paylaşılmıştır.

Makale Geçmişi / Article History

Başvuru Tarihi / Date of Application : 1 Şubat / February 2022

Kabul Tarihi / Acceptance Date : 11 Şubat / February 2022

© 2022 Journal of Yaşar University. Published by Yaşar University. Journal of Yaşar University is an open access journal.

1.Giriş

Küreselleşen bir dünyada işletmeler ve tüketiciler ülke sınırları olmaksızın birbirinden etkilenir hale gelmektedir. Bir ülkede gerçekleşen olumlu ya da olumsuz bir durum ister istemez diğer ülkeleri de etkilemektedir. Özellikle teknolojinin gelişmesi bu etkileşimi arttırmaktadır. Etkileşim entegrasyon olarak da kendini göstermektedir. İşletmeler daha önceden sınırlı coğrafyalarda faaliyet gösterirken şimdilerde birden fazla coğrafyada faaliyet göstermektedir. İşletmelerin faaliyet alanlarının değişimi ister istemez muhasebe alanında ihtiyaçları beraberinde getirmiştir. Teknolojinin gelişimi muhasebe sistemlerinde ihtiyaç duyulan yeniliklerin karşılanmasını sağlamaktadır. Muhasebe alanında karşılanan ilk teknolojik ihtiyaç muhasebe yazılımlarının ortaya çıkışıdır. Muhasebe yazılımlarının ortaya çıkışı, işletmelerin muhasebede en sık kullandığı belgelerin dijital ortama taşınmasını da beraberinde getirmektedir. Özellikle son kullanıcı tarafında fatura işlemleri, en sık kullanılan muhasebe evrakıdır (Atabey ve Parlakkaya, 2004; Demirdöven, 2017).

E-fatura süreçlerinde var olması gereken asgari gereklilikler; e-faturanın değiştirilmemiş olması ve kalıcı olarak saklanmasıdır. E-fatura açısından veri güvenliği ve kalıcılık, blok zincir teknolojisi ile sağlanabilmektedir. Blok zincir teknolojisi merkezi olmayan bir mimariye sahiptir. Merkezi olmayan bu mimari birbirine bağlı ağ yapısına sahiptir. Bu mimari; ağ içerisinde bulunan, verileri saklayan ve sorgulanan verinin doğruluğunu kontrol eden düğüm (Node) ögelerinden oluşmaktadır. Bu nedenle zincir içerisine eklenen bir veri ivedilikle tüm ağda bulunan düğümlere dağıtılmaktadır. Verinin tüm ağ üzerinde bulunan düğüm noktalarına gönderilmiş olması verinin birden fazla kopyasının olması anlamına gelmektedir. Bu durum, e-fatura bilgisinin blok zincir ağının çalışmayı sürdürdüğü sürece kalıcı olmasını sağlamaktadır. Ayrıca merkezi olmayan depolama çözümleri verinin birden fazla kopyasını sağladığından dolayı verinin kalıcılığını mümkün hale getirmektedir. Blok zincir teknolojisi isminden de anlaşılacağı üzere bloklardan oluşan bir yapıya sahiptir. Bir blok üzerinde kaydedilen bilgi onaylandıktan sonra artık değiştirilemez. Bu nedenle e-fatura süreçlerinde ihtiyaç duyulan verinin değiştirilmemiş olması sağlanmaktadır. Bu çalışmada, Blok zincir teknolojisi ile e-fatura'ların depolanabileceği ve sorgulanabileceği bir çözüm geliştirilmiştir. Bu çalışmanın 2. bölümünde, bu alanda yapılmış çalışmalar hakkında bilgi verilmiştir. Bölüm 3'te problemin çözümünün daha açık bir şekilde anlaşılabilmesi için temel tanımlar yapılmıştır. Bölüm 4'de problemin tanımı ve çözüm önerisi anlatılmıştır. Bölüm 5'te önerilen çözüm yöntemi ile ilgili bulgular sonuç bölümünde sunulmuştur.

2. Literatür Özeti

Brainstorm web sitesinde konuk yazar olarak bir makale yayımlayan Lazonis, Bitcoin'in arkasındaki teknolojinin muhasebeyi ne şekilde dönüştürülebileceğini sorgulamıştır. Bu çalışma blok zincir ve muhasebe ilişkisinin nasıl kurulduğu ve bu yeni teknolojinin etkilerinin sorgulandığı ilk çalışmalardan biridir (Lazanis, 2015).

Blok zincir temelinde önerilen muhasebe çözümlerinde araştırmacılar tarafından farklı kavramlar kullanılmıştır. Ancak kullanılan kavramların incelenmesi ve ortak kavram kabulü ile ilgili çalışmalar gerçekleştirilmiştir. Bu çalışmalar sonucunda blok zincir temelinde oluşturulan muhasebe sistemlerine “üç taraflı muhasebe sistemi” kavramı kullanılmasına karar verilmiştir (Dai ve Vasarhelyi, 2017; Hambiralovic ve Karlsson, 2018). Birçok araştırmacı muhasebe ve blok zincir ilişkisinin 3. taraf onay ve kontrolü yönüyle ele almış, bu bağlamda çalışmalarını gerçekleştirmiştir (Baev ve diğerleri, 2020; Cai, 2019; Fullana ve Ruiz, 2021; Hambiralovic ve Karlsson, 2018; Pascual Pedreño, Gelashvili ve Pascual Nebreda, 2021).

Özdoğan ve Karğın yaptıkları çalışmada blok zincirinin muhasebe ve finansman alanında olası etkilerini, beklentilerini incelemiştir ve blok zincir teknolojisinin sunduğu değer karmasını oluşturmuşlardır (Özdoğan ve Karğın, 2018). Akıllı sözleşmeler ve blok zincir uygulamalarının muhasebe ve vergi uygulamalarına yönelik çalışmalar yapılmıştır. Bu çalışmalarda blok zincir teknolojisinin muhasebe denetiminde olumlu etki yaratacağı yönünde bulgular elde edilmiştir (Asrav, 2019; Atçı, 2019). Özkul ve Alkan yaptıkları çalışmada blok zinciri temelinde oluşturulmuş üç taraflı muhasebe sistemini teorik olarak önermişlerdir (Özkul ve Alkan, 2020).

Narayanam ve ark., uluslararası ticaret için blok zincir teknolojisi ile geliştirilmiş e-fatura platformu önerisinde bulunmuşlardır (Narayanam ve diğerleri, 2020). Guerar ve ark., blok zincir teknolojisinin fatura işlemlerinde ve finasta dolandırıcılığı azaltacağı yönünde çalışma yapmışlardır (Guerar, Merlo, Migliardi, Palmieri ve Verderame, 2020).

Literatür özetinde belirtildiği üzere muhasebe yönünden blok zincir teknolojisi birçok fırsat barındırmaktadır. Bu fırsatların neler olduğunu, muhasebe süreçlerinde katkıları birçok araştırmacı ortaya koymuştur. Ancak yapılan çalışmaların tümü teorik olarak konunun değerlendirilmesi, çözüm önerilerin kavramsal seviyede belirtilmesi ile sınırlı kalmıştır. Bu çalışmada konu sadece teorik değil uygulamalı olarak değerlendirmeye alınmıştır.

3. Tanımlar

Muhasebe işlemlerinde en çok kullanılan belge faturadır. Faturanın tanımı ve yasal dayanağı hakkında bilgi verilmiştir. Teknolojik imkanların getirdiği fırsatlar çerçevesinde faturanın dijital olarak saklanması mümkün hale gelmiştir. Dijital ortamda saklanan faturalar e-fatura olarak isimlendirilmektedir. Bu çalışmada e-faturanın tanımı ve ilgili alanda yasal düzenlemeler açıklanmıştır. Çalışmanın devamında blok zincir teknolojisi, akıllı kontrat kavramları açıklanmıştır.

3.1. Faturanın Tanımı

İşletmelerde muhasebe kayıtları gerçekleştirilirken poliçe, çek, fatura, gider pusulası, sevk irsaliyesi vb. birçok belge düzenlenmektedir. Muhasebe belgesi genel olarak ticari faaliyeti, işlem veyahut olayı kişi, zaman, nitelik ve nicelik bilgileri yönünden kayda geçiren ve kanıtlayan evrakların tümüdür (Atabey ve Parlakkaya, 2004). Fatura, işletmeler tarafından ve işletmeler arasında en fazla kullanılan belge türüdür. Vergi Usul Kanunu (VUK) 229. maddesinde faturanın tanımı “Fatura, satılan emtia veya yapılan iş karşılığında müşterinin borçlandığı meblağı göstermek üzere emtiayı satan veya işi yapan tüccar tarafından müşteriye verilen ticari vesikadır.” şeklinde belirtilmiştir. Faturanın ne şekilde düzenleneceği VUK 231. maddesinde faturaların sıralı olması, bir asıl ve bir örnek olarak düzenlenmesi ve imza yetkili kişi tarafından imzalanması gerekliliği belirtilmiştir. Birinci ve ikinci sınıf tüccarlar, serbest meslek erbabı, kazançları serbest usulde tespit olunan tüccarlar, defter tutmak zorunda olan çiftçiler ve vergiden muaf esnafın fatura kullanması gerekliliği VUK 232. maddesinde belirtilmiştir (Vergi Usul Kanunu, 1961).

3.2. E-fatura Tanımı

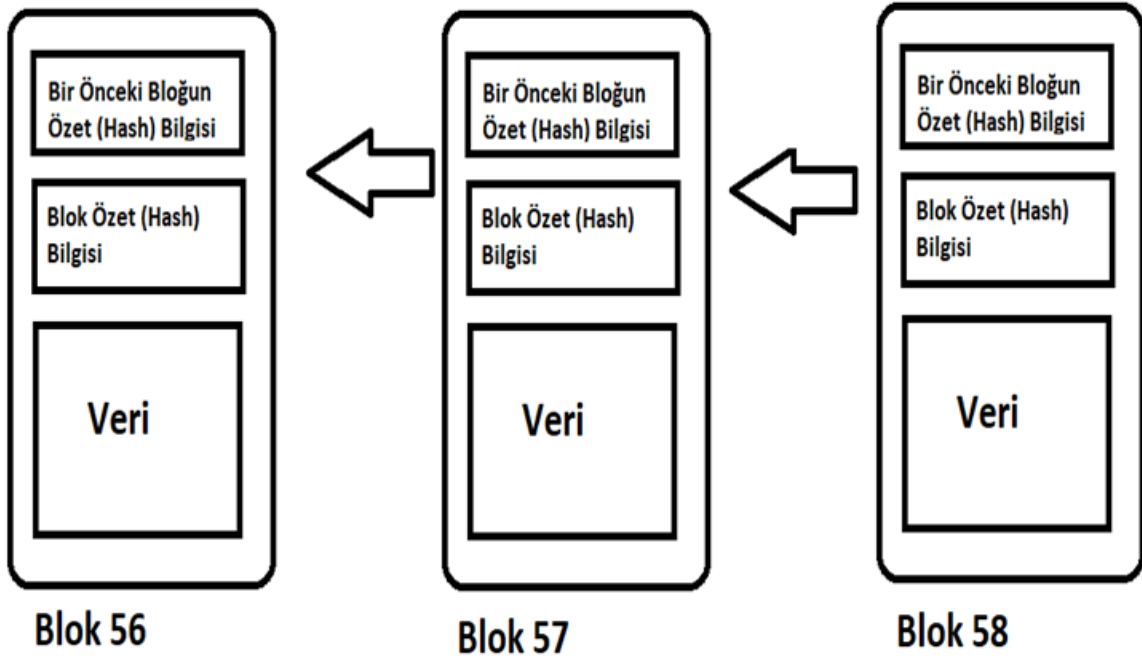
Türkiye’de e-devlet projesi ile birlikte muhasebe belgelerinin elektronik olarak oluşturulması ve elektronik ortamda saklanması işlemi, gerçekleştirilmeye başlanmıştır. Faturaların elektronik ortamda düzenlenmesi gerekliliği, faaliyet büyüklüğü ve işletmenin ticari alanına göre birçok işletmede zorunlu hale getirilmiştir. Dünyada diğer ülkelerde uygulanan e-fatura sistemi Türkiye’de Elektronik Fatura Kayıt Sistemi (EFKS) ismi ile hayata geçirilmiştir (Aykın, 2002; Demirdöven, 2017). Genel çerçevede elektronik fatura, VUK’da belirtilen faturanın gerekliliklerinin elektronik ortamda uygun karşılıklarının sağlanarak kaydedilmesi ve ihtiyaç halinde ilgili kişi ve kurumlara ibraz edilmesi için oluşturulan dijital varlıktır.

Gelir İdaresi Başkanlığı (GİB) tarafından elektronik ortamda oluşturulmuş faturaların kayıt altına alınması ve saklanması amacıyla EFKS oluşturulmuştur. GİB’in 2008 yılında yaptığı çalışma ile elektronik faturaya geçen ilk kurumlar Avea, TNet, Turkcell, Türk

Telekom olmuştur (Yanık ve Karadaş, 2013). Elektronik faturanın son kullanıcıya sunulmasında en çok tercih edilen dosya formatı PDF'dir. Maliye Bakanlığı 397 sıra numaralı VUK genel tebliği ile elektronik fatura tanımını ve esaslarını belirlemiştir (Resmi Gazete, 2010). Elektronik fatura uygulamasına daha önceden geçmiş ülkeler ile uyumluluğunun sağlanabilmesi için XML tabanlı Universal Business Language (UBL) standardında bir yapı kabul edilmiştir (Gelir İdaresi Başkanlığı, 2013). GİB tarafından e-fatura entegrasyon süreçlerinin sağlandığı e-fatura portalı oluşturulmuştur. GİB elektronik faturaların kendisine iletilmesinin yeterli olmadığını belirtmekte ve e-faturaların saklanması gerektiğini belirtmektedir. Bu amaçla işletmelerin isterse kendisi isterse GİB tarafından yetkilendirilmiş firmalar tarafından e-faturaların en az 10 yıl saklanması gerektiği belirtilmiştir (Elektronik Defter Genel Tebliği, 2019).

3.3. Blok zincir Tanımı

Blok zincir teknolojisinin temelini oluşturan temel çalışma Stuart Haber ve W. Scott Stornetta tarafından 1991 yılında yapılmıştır. Stuart Haber ve W. Scott Stornetta yaptıkları çalışmada blok zincir, kriptolojik yöntemler ile desteklenmiş dijital belgelerin zaman damgası ile değiştirilemeyeceği ve geriye dönük olarak bilginin doğruluğunun takibinin yapılabileceği zincir olarak tarif etmişlerdir. Daha sonra bu zincir yapısında güvenliğin artırılması amacıyla oluşturulmuş tasarıma Merkle ağaçları yöntemi eklenmiştir. Ancak bu tasarımda kullanılan teknoloji Bitcoin dijital paranın ortaya çıkışı ve temelinde yatan blok zincir teknolojisine kadar aktif bir şekilde kullanılmamıştır (Haber ve Stornetta, 1991; Merkle, 1987; Narayanan, Bonneau, Felten, Miller ve Goldfeder, 2016). Genel olarak blok zincir teknolojisinde; verilerin tutulduğu bir bilgi paketi bulunmaktadır ve bu bilgi paketi blok olarak isimlendirilmektedir. Her blok içerisinde bulunan verinin kriptolojik yöntemler ile oluşturulmuş özet bilgisi ortaya çıkarılır. Her bir bloğun kendinden önce oluşturulmuş bloğa ait özet bilgisini barındırması ile veri bloklarının bir zinciri oluşturulmaktadır (Şekil 1). Günümüzde kullanılan blok zincir yöntemini ve zincirin içerdiği verinin eşler arası bir yöntemle nasıl doğrulanacağını ilk ortaya koyan çalışma 2008 yılında Satoshi Nakamoto tarafından gerçekleştirilmiştir. Nakamoto ortaya koyduğu çalışmadaki fikirlerin hayata getirilmiş halini 2009 yılında yayına almıştır. Blok zincir teknolojisi hayata geçmiş olup bu teknoloji üzerine bina edilen dijital para birimi Bitcoin olarak adlandırılmıştır (Brito ve Castillo, 2013; Nakamoto, 2008).



Şekil 1. Blok Zincir Yapısı (Nakamoto, 2008)

3.4. Akıllı sözleşme Tanımı

Blok zincir teknolojisi tasarım ve yapı olarak her ne kadar bilgi saklama için güvenli olsa dahi tek bir görevi icra edecek biçimde tasarlanmıştır. Ancak blok zincir, teknolojinin getirdiği imkanlar ve Web 2.0'la birlikte insanların internet aracılığı ile teknoloji çözüm odaklı yeni ihtiyaçlarını karşılamakta yetersiz kalmıştır (Liu ve Fraser, 2018; Mittal, 2017). Bu nedenle 2014 yılında Ethereum projesi ortaya çıkmıştır. Bu projenin belirleyici özelliği Akıllı Sözleşmelerdir. Akıllı sözleşmeler temel olarak blok zincir'de tutulan verinin yerine kod betiği konulması ve bu kodun Ethereum Virtual Machine (EVM) üzerinde çalıştırılmasıdır. Akıllı kontratların belirli şartlar altında kod betiğinin etkileşim yaratması belli süreçlere başlayabilmesi yönüyle önemlidir (Clack, Bakshi ve Braine, 2016; Mittal, 2017). Akıllı sözleşmeler temel olarak kontrat owner, state, nonce ve amount bileşenlerine sahiptir. Akıllı kontratlar Ethereum ağında Solidity isminde nesne yönelimli bir programlama dili ile yazılmaktadır. Akıllı sözleşmeler çalıştırıldıklarında kullandıkları işlem ve depolama kaynağına göre GAS ismi verilen işlem ücreti ile çalıştırılmaktadırlar (Clack ve diğerleri, 2016).

Akıllı sözleşmeler, merkezi olmayan uygulamaların (Dapps) geliştirilmesinde önemli bir role sahiptir. Bu nedenle insan rolüne ihtiyaç duyulmayan ve merkezi bir işlem birime ihtiyaç olmayan uygulamalarda kullanılmaya başlanmıştır. Akıllı sözleşmelerin kullanıldığı alanlar; örnek olarak tedarik zinciri, sağlık, finans, sigortacılık, oy verme, enerji yönetimi, perakende,

sosyal medyadır (Allison, 2016; Jamil, Hang, Kim ve Kim, 2019; Sharma, 2018). Merkezi olmayan uygulamalar bağlam olarak blok zincir ağı ile akıllı sözleşmeler aracılığıyla iletişime geçilmesini ifade etmektedir. Ön yüz olarak Dapps normal bir web sitesi veyahut mobil uygulama arayüzüne sahip olabilmektedir. Ancak arka planda blok zincir ağı ile değişik şekillerde iletişim ve etkileşim kurabilme yetisine sahiptir. Dapps temel özellikleri olarak değişmezlik, kurcalamaya karşı dirençlilik, şeffaflık ve sürekli çalışabilirlik sayılabilir (Lau ve diğerleri, 2020).

Değişmezlik (Immutability): Bir veri, blok zincir ağı üzerine kaydedildikten sonra bir daha değiştirilemez (Lau ve diğerleri, 2020).

Kurcalamaya karşı dirençli (Tamper-proof): Akıllı sözleşme bir kere blok zincir ağı üzerinde yayımlandıktan sonra ağ üzerindeki tüm katılımcılar bilgilendirilmeden değişiklik gerçekleşmez (Lau ve diğerleri, 2020).

Şeffaflık (Transparent): Dapps etkileşime girdiği akıllı sözleşmeler denetime açıktır (Lau ve diğerleri, 2020).

Sürekli çalışabilirlik (Availability): Akıllı sözleşmenin barındırıldığı blok zincir ağı çalıştığı sürece çalışabilmektedir (Lau ve diğerleri, 2020).

Merkezi olmayan uygulamaların ortaya çıkışı beraberinde merkezi olmayan depolama çözümlerini de getirmiştir. Merkezi olmayan depolama projelerinden en kabul göreni Interplanetary File System (IPFS)'dir. IPFS hem bir protokol hem de dağıtık bir dosya sistemidir (Benet ve Greco, 2017; Finley, 2016).

4. Problemin Tanımı ve Çözüm Önerisi

E-faturaların merkezi olmayan bir yazılım mimarisi kullanılarak blok zincir üzerinde saklanması bir problemdir. Bu çalışmanın devamında bu problemin temel düzeyden gerçekleştirilmesi, süreçlerinin açıklanması ve örnek kod betiği verilmiştir.

4.1. Problemin Tanımı

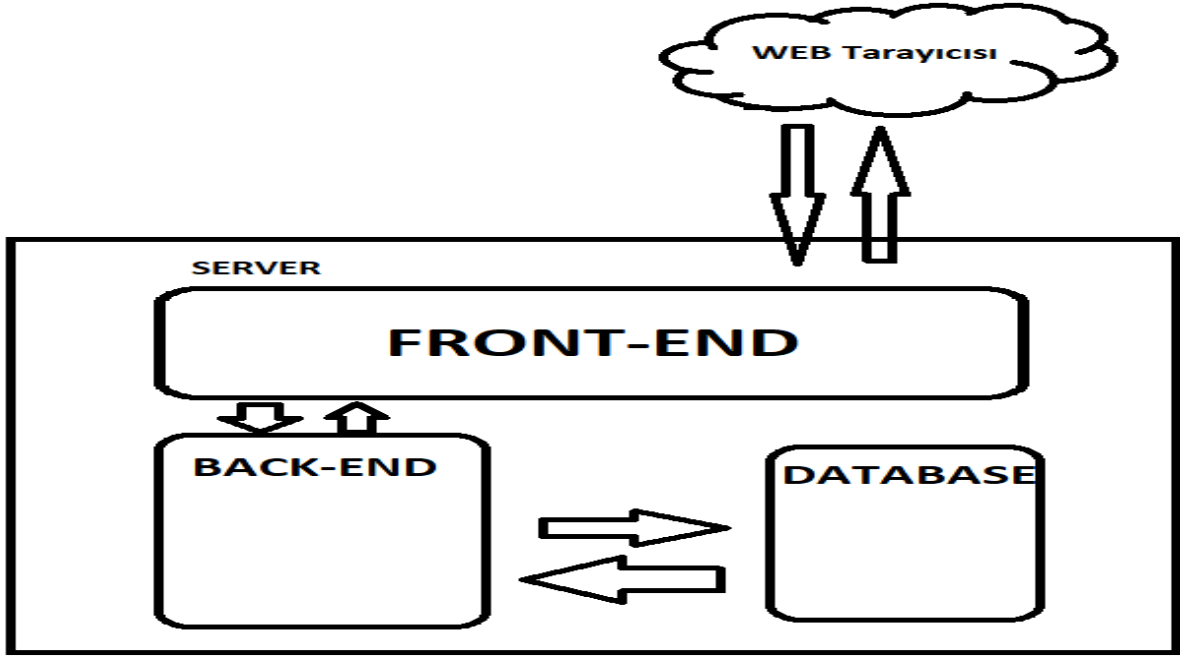
Bilişim teknolojileri bireylere olduğu kadar işletmelere de faydalar sağlamaktadır. Bu faydalar işletmelerin süreçlerinde verimliliğin artırılması açısından önemlidir. E-faturanın işletmelerin hayatına aktif olarak girmesi ile bu alandaki teknolojik ihtiyaçlar, gerek büyük gerekse küçük işletmelerde önem arz etmeye başlamıştır. İşletmeler oluşturdukları e-faturaları GİB tarafına göndermektedirler. Ancak yasal düzenlemelerle belirtilen belli bir süre boyunca oluşturulan e-faturaları saklama yükümlülükleri bulunmaktadır (Elektronik Defter Genel Tebliği, 2019). Saklanan faturaların hiçbir şekilde değiştirilmemesi ve değişikliklerden korunması işletmelerin bir diğer yükümlülüğüdür. Bu durum işletmeler için bir problemdir. Bu

ihtiyaçların giderilmesinde blok zincir teknolojisi, akıllı sözleşmeler ve merkezi olmayan depolama çözümleri bir cevap olabilmektedir.

Blok zincir teknolojisi kendi içinde veri tutabilme yetisine sahiptir. Ayrıca kendi üzerinde tuttuğu veri bir kere kaydedildikten sonra bu veri üzerinde geçmişe yönelik olarak değişikliğe izin vermemektedir. Blok zincir üzerinde tutulan veriler tek bir merkezde saklanmadığından verinin güvenliği ve değiştirilemezliği emniyet altına alınmaktadır. Akıllı sözleşmeler blok zincir üzerinde çalışan ve belli şartlar gerçekleştiğinde harekete geçen kod altyapısıdır. Bu nedenle kaydedilen veri daha önceden belirtilen şartlar gerçekleştiğinde istenilen bir şekilde kullanıma sunulabilmektedir. Örnek olarak sadece belli bir yetkiye sahip olan hesapların veriyi görebilmesi verilebilir. Akıllı sözleşmeler her ne kadar istenilen aksiyonu almak için iyi bir çözüm olsa da veri depolamak için etkili bir yöntem değildir. Kalıcı veri depolama için kullanıldığında sınırlara sahip olması ve GAS maliyetlerinin artması nedeniyle depolama alanı olarak bir başka çözümün kullanılması gerekliliği ortaya çıkmaktadır. Bu noktada depolama çözümü olarak IPFS devreye girmektedir. Dosyaların kalıcı olarak depolanması ve adreslerinin dosyanın özet bilgisi ile gerçekleştirilmesi, dağıtık depolama ile nerede saklandığının bilinmemesi sadece dosya adresi ile ulaşılabilir olması belli ölçüde dosyanın saklanmasında etkili bir çözüm olarak ortaya çıkmaktadır.

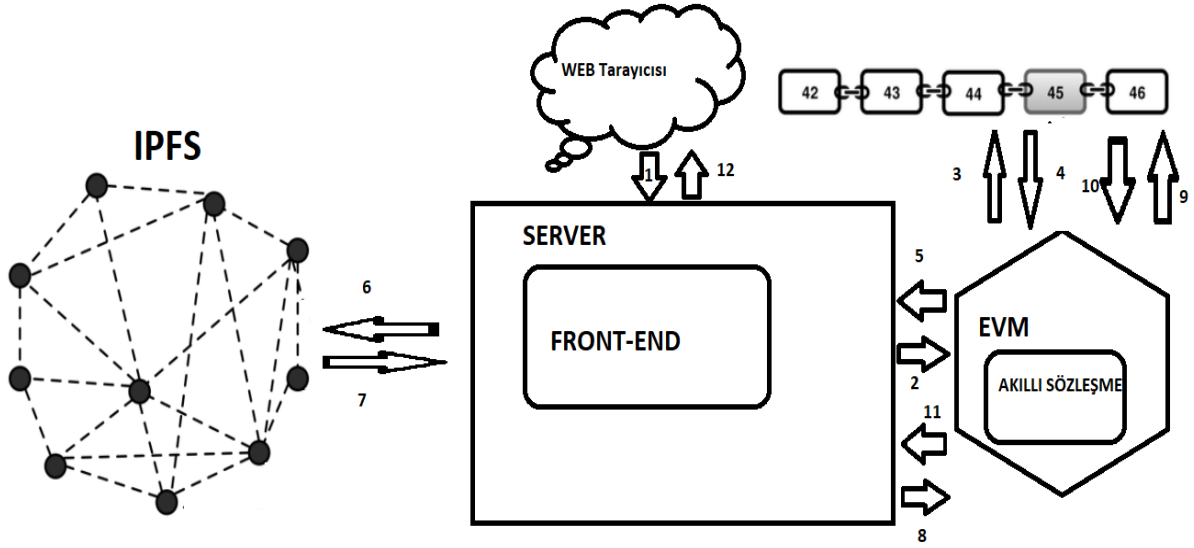
4.2. Önerilen Yöntem

Klasik merkezi mimaride geliştirilmiş uygulama ve veri saklama süreçlerinde, uygulama sunucusu ve veri tabanı bulunmaktadır. Şekil 2.'de gösterildiği üzere son kullanıcı web tarayıcısı ile uygulamanın Front-End katmanı ile iletişime geçmektedir. Front-End katmanında genel olarak HTML, CSS, Javascript teknolojileri kullanılmaktadır. Front-End katmanı gelen isteği karşıladıktan sonra Back-End katmanına göndermektedir. Back-End katmanına gelen istekler tanımlı kod betiklerini harekete geçirmektedir. Eğer gelen istek; harekete geçirdiği kod betiğinin veri tabanına istek yollamasını gerekli kılması durumunda, Back-End katmanı veritabanı ile iletişime geçmektedir. Back-End katmanından genel olarak PHP, ASP, Python, Java ve Ruby vb. programlama dili bulunmaktadır. Web tarayıcı tarafından başlatılan tüm bu süreç tamamlandığında yine Web tarayıcısına aynı yolu geri takip ederek cevap verilmektedir. İşlem sürecinde belirtilen tüm aşamalar sunucu tarafında tek bir işlem birimine güven üzerine kurulmuş olarak gerçekleşmektedir.



Şekil 2. Merkezi Yazılım Mimarisi

E-faturaların saklanması var olan probleme, blok zincir teknolojisi ve onun sağladığı merkezi olmayan uygulama ile bir çözüm önerisi sunulmuştur. Bu çözüm önerisi Şekil 3.'de verilmiştir. Şekil 3.'de verilen çözüm önerisi adım adım takip edildiğinde gerçekleşen aşamalar şunlardır. Son kullanıcı Web tarayıcısı aracılığı ile Front-End ile iletişime geçmektedir. Bu aşamada, web tarayıcısı ile görüntülenen Front-End aşamasında bulunan formda e-faturanın temel bileşenlerini oluşturan bilgilerin girişi yapılmaktadır. Yapılan giriş bilgilerine göre akıllı sözleşme tetiklenir ve fatura bilgisinin blok zincir ağı üzerinde daha önceden kayıtlı olup olmadığı kontrol edilir. Kayıtlı olmadığı bilgisine ulaşılması durumunda e-fatura bilgileri bir dosya içine kayıt edilerek IPFS üzerinde kayıt işlemi gerçekleştirilir. IPFS üzerinde yükleme sonucunda oluşan dosyanın adresi elde edilir. Bu adres bilgisi geri Front-End katmanına döndürülür. Fatura bilgisi, dosya adresi bilgileri ile tekrar akıllı sözleşmede ilgili fonksiyon tetiklenir ve blok zincir ağı üzerinde bilgilerin kaydedilmesi için süreç başlatılır. Sürecin başarılı olması durumunda Front-End katmanında kayıt işleminin başarılı bir şekilde gerçekleştiği bilgisi verilir. Tüm bu süreç e-fatura kaydının merkezi olmayan bir uygulamada mimari yapısının özetidir.



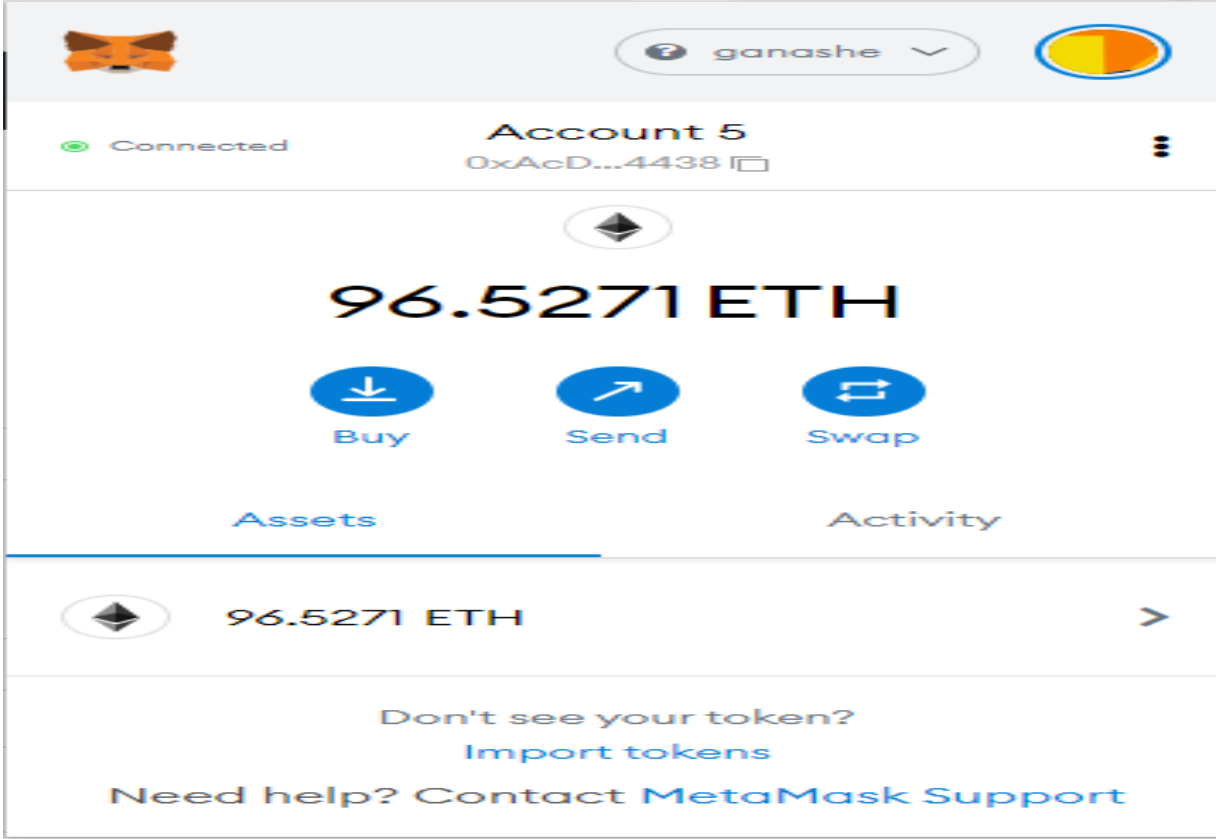
Şekil 3. Önerilen Yöntemin Uygulama Mimarisi

Bu çalışmada e-faturaların blok zincir ve merkezi olmayan depolama alanı üzerinde saklanması metot olarak sunulmasının yanında uygulama olarak da gerçekleştirilmiştir. Uygulamanın gerçekleştirilmesi sürecinde yerelde çalışan ve test işlemlerinin gerçekleştirilmesi amacıyla akıllı sözleşmeleri çalıştırabileceğiniz, kendi blok zincir ağınıza oluşturabileceğiniz Ganache yazılı kullanılmıştır (Şekil 4) (ConsenSys, 2021).

Ganache			
ACCOUNTS	BLOCKS	TRANSACTIONS	CONTRACTS
CURRENT BLOCK 95	GAS PRICE 2000000000	GAS LIMIT 6721975	HARDFORK MUIRGLACIER
NETWORK ID 5777	RPC SERVER HTTP://127.0.0.1:7545	MINING STATUS AUTOMINING	WORKSPACE SEDÁTE-FAMILY
MNEMONIC visit logic deposit maid service tomorrow fix acid legend connect venue tiger		HD PATH m/44'/60'/0'/0'/account_index	
ADDRESS 0xAcD605526778073CFFde463eb2a9963b3a034438	BALANCE 96.53 ETH	TX COUNT 88	INDEX 0
ADDRESS 0xb9f23BFba7eA38Efc29CbF39B069c8Be611B089d	BALANCE 103.16 ETH	TX COUNT 7	INDEX 1
ADDRESS 0x7c26066053Ce759126901401A5357bc5C3636C5B	BALANCE 100.02 ETH	TX COUNT 0	INDEX 2
ADDRESS 0xbFB46f9Ff25861649D61B122e218aaF4A167C6F4	BALANCE 100.00 ETH	TX COUNT 0	INDEX 3
ADDRESS 0xddc88f7495A163E17ae04b69dC8faFa119870fca	BALANCE 100.00 ETH	TX COUNT 0	INDEX 4
ADDRESS 0x27885B1671f7Ef477a9aC815E63C26f4177767b6	BALANCE 100.00 ETH	TX COUNT 0	INDEX 5
ADDRESS 0x888F0CCBF04d20313C6D2bb7AAF81C90cFAAC6d4	BALANCE 100.00 ETH	TX COUNT 0	INDEX 6

Şekil 4. Ganache Ekran Görüntüsü

Ganache aracı üzerinde oluşturulmuş hesapların kontrolü ve GAS ödemelerinin gerçekleştirilmesi amacıyla Metamask cüzdanı (Şekil 5) kullanılmıştır. Front-End katmanında son kullanıcı ile etkileşime geçmesi için HTML, Javascript dilinin Web3Js paketinden faydalanılmıştır (Web3Js, 2021).



Şekil 5. Metamask Cüzdan Uygulaması Ekran Görüntüsü

Uygulama Şekil 3.'de belirtildiği üzere aynı süreçlere sahiptir. Son kullanıcı web tarayıcısı ile Şekil 6.'te verilmiş olan web formuna ulaşmaktadır. Bu form doldurulduktan sonra Javascript kodları Ethereum ağı üzerinde tutulan akıllı kontrat ile iletişime geçmektedir.

Fatura Oluştur

Fatura Bilgileri

Fatura No :	Fatura Tutar :	Fatura Tarihi :
<input type="text" value="Fatura No"/>	<input type="text" value="Fatura Tutar"/>	<input type="text" value="gg.aa.yyyy --:--"/>

Fatura oluşturan Bilgileri

Ünvan :	İlçe :	İl :	Ülke :	
<input type="text" value="Ünvan"/>	<input type="text" value="İlçe"/>	<input type="text" value="İl"/>	<input type="text" value="Ülke"/>	
Adres :				
<input type="text" value="Adres"/>				
Telefon :	Vergi Dairesi :	Vergi Kayıt No :	Ticaret Sicil No :	Mersis No :
<input type="text" value="Telefon"/>	<input type="text" value="Vergi Dairesi"/>	<input type="text" value="Vergi Kayıt No"/>	<input type="text" value="Ticaret Sicil No"/>	<input type="text" value="Mersis No"/>

Fatura Sahibi Bilgileri

İsim :	Soyad :	TC :
<input type="text" value="İsim"/>	<input type="text" value="Soyad"/>	<input type="text" value="TC"/>
İlçe :	İl :	Ülke :
<input type="text" value="İlçe"/>	<input type="text" value="İl"/>	<input type="text" value="Ülke"/>
Adres :		
<input type="text" value="Adres"/>		
Eposta :	Telefon :	
<input type="text" value="Eposta"/>	<input type="text" value="Telefon"/>	
<input type="button" value="Create Fatura"/>		

atus: Ready!

Coded By Mansur BEŞTAŞ 2021

Şekil 6. Fatura Oluşturma Formu

Kontrat ile iletişim sağlayan Javascript kodları şekil 7.'da verilmiştir. Şekil 7'da görüleceği üzere dosya yükleme, fatura varlığının kontrolü, fatura kaydı vb. metotlar bulunmaktadır.

```

1  const testmi = 1;
2  if(testmi == 1){
3    var NODE_URL = "https://data-seed-prebsc-2-s2.binance.org:8545/"; // test
4    var CONTRACT_ADDRESS = "0xb85dA4455615Aa8699b99a10d79cCF5f2EfcA9E6"; // fatura
5  } else {=}
6
7
8
9
10 const WALLET_ADDRESS = "0xAcD605526778073CFFde463eb2a9963b3a034438";
11 const PRIVATE_KEY = "0xc18809495a9956a5622176bf542e5dfecd3aba9612a282c963e406210317fa35";
12 const chain_id = 1337;
13 > const abi = [
109  /* fonksiyonlar */
110 > async function loadWeb3() {=}
116
117 > async function loadContract() {=}
120
121 > async function getCurrentAccount() {=}
125
126 > async function load() {=}
132
133 > function updateStatus(status) {=}
138
139  ////////////////////////////////////////////////////
140 > async function createFatura(){=}
195
196 > async function contratcalistir(FaturaNo, OvergiKayitNo, DosyaHash, account){=}
206
207 const sleep = (ms) => new Promise(resolve => setTimeout(resolve, ms))
208
209 > async function getaccountdata(){=}
213
214 > async function fileuploadipfs(file_name){=}
233
234 > async function checkFaturaExist(){=}
263
264 > async function kaydet_dosya(FaturaNo, bilgiler_toplu){=}
274
275 > async function getFaturaNumber(){=}
293
294 > async function createFaturaPrivatekey(){=}
316
317 > async function sentTransaction(){=}
336
337 > async function testFatura(){=}
347
348 > async function getFaturaDetails(){=}
364
365 async function verifyFatura(){
366   let id = $('#FaturaHash').val();
367   getDetailsIpfs(id)
368 }
369 > async function getDetailsIpfs(DosyaHash){=}
405
406 > /*=
428 load();
429

```

Şekil 7. Javascript Kodları

Javascript kodları Solidity dilinde yazılmış akıllı sözleşme ile iletişime geçmektedir. Bu akıllı sözleşmenin kodları şekil 8.'de verilmiştir. Akıllı sözleşme kendisine gönderilmiş olan fatura bilgisinin blok zincir ağı üzerinde kayıtlı olup olmadığı kontrol etmektedir. Eğer kayıtlı değil ise IPFS üzerinde kaydı yapılmış olan dosyanın adres bilgisi ağ üzerinde kaydedilmektedir. Solidity kodları incelendiğinde getData fonksiyonu (sıra 39-44) ile faturanın ağ üzerinde kayıtlı olup olmadığı kontrolü gerçekleştirilmektedir. GenerateFatura fonksiyonu (sıra 27-37) ile fatura bilgisi ve dosya bilgisi ağ üzerine kaydedilmektedir.

```

1  pragma solidity ^0.4.24;
2
3  contract Fatura {
4      constructor() public {}
5
6      struct Fatura {
7          string candidate_name;
8          string org_name;
9          string course_name;
10         uint256 expiration_date;
11     }
12
13     mapping(bytes32 => Fatura) public faturalar;
14
15     event faturaGenerated(bytes32 _faturaId);
16
17 > function stringToBytes32(string memory source) private pure returns (bytes32 result) {=
25     }
26
27     function generateFatura(
28         string memory _id,
29         string memory _candidate_name,
30         string memory _org_name,
31         string memory _course_name,
32         uint256 _expiration_date) public {
33         bytes32 byte_id = stringToBytes32(_id);
34         require(faturalar[byte_id].expiration_date == 0, "Fatura zaten var");
35         faturalar[byte_id] = Fatura(_candidate_name, _org_name, _course_name, _expiration_date);
36         emit faturaGenerated(byte_id);
37     }
38
39     function getData(string memory _id) public view returns(string memory, string memory, string memory, uint256) {
40         bytes32 byte_id = stringToBytes32(_id);
41         Fatura memory temp = faturalar[byte_id];
42         require(temp.expiration_date != 0, "No data exists");
43         return (temp.candidate_name, temp.org_name, temp.course_name, temp.expiration_date);
44     }
45 }
46 |

```

Şekil 8. Akıllı Kontrat Kodları

Blok zincir ağı üzerinde kaydedilmiş olan bir fatura, hem fatura bilgisi hem de IPFS üzerinde kaydedilmiş dosya bilgisi ile sorgulama işlemi yapılabilir. IPFS üzerinde kaydedilen her dosyanın özet bilgisinden oluşan adres değeri alınarak benzer bir dosyanın olup olmadığı kontrol edilmektedir. Bu IPFS üzerinde dosya olarak kaydedilen faturaların tekliğini sağlamaktadır.

5. Sonuç

Sürekli gelişen teknoloji hayatın tüm alanlarında kendine yer edinmektedir. İşletmeler gelişen teknolojinin getirdiği kolaylık ve verimlilik fırsatlarını sürekli olarak süreçlerine entegre etmektedir. İşletmelerin muhasebe iş ve işlemleri, işletmede gerçekleştirilen standart faaliyetlerde önemli yer tutmaktadır. Dünyada ve Türkiye’de muhasebe işlemlerinin devlet eli ile teknolojik imkanlardan faydalanılarak kolaylaştırma sürecine geçilmiştir. Muhasebe işlemlerinde önemli işlerden biri de faturadır. Faturanın oluşturulması ve saklanması dijitalleşmenin getirdiği fırsatlar ile dönüşüme girmiştir. Dijitalleşmenin getirdiği dönüşüm e-

fatura kavramını ortaya ıkarmıřtır. Halihazırdaki e-fatura özümleri merkezi yazılım ve depolama birimleri ile gerekleřtirilmektedir.

Son yıllarda blok zincir ve merkezi olmayan depolama özümleri birok alanda kullanılmaya bařlanmıřtır. Blok zincir ve merkezi olmayan depolama özümleri, beraberlerinde merkezi olmayan uygulamaları ortaya ıkarmıřtır. Merkezi olmayan uygulamaların veri güvenlięi, deęiřtirilemezlik faydalarının yanısıra teknolojik yatırım ihtiyacını azaltması bir bařka avantaj olarak görölmektedir. Bu alıřmada merkezi olmayan bir uygulama geliřtirilmiř ve bu uygulama aracılıęı ile e-faturaların 3. taraf güvenine ihtiyaç duyulmadan kayıt altına alınması için özüm önerisi sunulmuřtur. Önerilen özüm yöntemi tamamen kendi kendine alıřan ve verinin deęiřikliğe karřı korumalı olarak alıřması yönüyle önemlidir. Gelecekte blok zincir ve akıllı kontratlar desteęi ile oluřturulmuř merkezi olmayan uygulamaların günlük hayatta birok alanda daha fazla yer alacaęı düşünölmektedir. Bu yönüyle bu alıřma muhasebe alanında önemli yere sahiptir.

KAYNAKÇA

- Allison, I. (2016). Skuchain: Here's how blockchain will save global trade a trillion dollars. *International Business Times*, 1–5. <https://www.ibtimes.co.uk/skuchain-heres-how-blockchain-will-save-global-trade-trillion-dollars-1540618>. Erişim tarihi: 28.12.2021
- Asrav, E. A. (2019). Blok Zincir Teknolojisi Ve Akıllı Sözleşmelerin Vergilendirme Usulleri İle Muhasebe Ve Denetim Uygulamalarına Muhtemel Etkileri. *Vergi Raporu*, (232), 245–268.
- Atabey, A. ve Parlakkaya, R. (2004). *Muhasebede Belge Düzeni*. Ankara: Nobel Yayın Dağıtım.
- Atçı, M. (2019). Blok Zincir, Akıllı Sözleşmeler ve Üçlü Kayıt Sistemi Vergi ve Muhasebe Uygulamalarına Katkı Sağlayabilir mi? *Vergi Sorunları Dergisi*, 42(370), 76–88.
- Aykın, H. (2002). Türk Vergi Hukukunda Elektronik Defter, Belge ve Kayıt Kavramları. *Yaklaşım Dergisi*, (110).
- Baev, A. A., Levina, V. S., Reut, A. V, Svidler, A. A., Kharitonov, I. A. ve Grigor'ev, V. V. (2020). Blockchain technology in accounting and auditing. *Accounting. Analysis. Auditing*, 7(1), 69–79.
- Benet, J. ve Greco, N. (2017). *Filecoin: A Decentralized Storage Network. Protocol Labs*. <https://filecoin.io>. Erişim tarihi: 28.12.2021
- Brito, J. ve Castillo, A. (2013). BITCOIN A Primer for Policymakers. <http://online.wsj.com/article/SB100>. Erişim tarihi: 28.12.2021
- Cai, C. (2019). Triple-entry accounting with blockchain: How far have we come? *Accounting & Finance*, 61. doi:10.1111/acfi.12556
- ConsenSys (2021). Truffle Suite. <https://trufflesuite.com/ganache/>. Erişim tarihi: 28.12.2021
- Clack, C. D., Bakshi, V. A. ve Braine, L. (2016). Smart contract templates: foundations, design landscape and research directions. *arXiv preprint arXiv:1608.00771*.
- Dai, J. ve Vasarhelyi, M. A. (2017). Toward blockchain-based accounting and assurance. *Journal of Information Systems*, 31(3), 5–21.
- Demirdöven, M. O. (2017). *Muhasebede E fatura ve E defter; Türkiye'de E fatura E defter Sistemine Geçen İşletmelere İlişkin Bir Araştırma*. <http://dspace.trakya.edu.tr/xmlui/bitstream/handle/trakya/2019/0142449.pdf?sequence=1>. Erişim tarihi: 20.12.2021
- Elektronik defter genel tebliği (2019). Elektronik defter genel tebliği (sıra no: 1)'nde değişiklik yapılmasına dair tebliğ. Resmi Gazete (Sayı: 30923). <https://www.resmigazete.gov.tr/eskiler/2019/10/20191019-4.htm>. Erişim tarihi: 20.12.2021
- Finley, K. (2016). The Inventors of the Internet Are Trying to Build a Truly Permanent Web. *Wired*. 20 Haziran. <https://www.wired.com/2016/06/inventors-internet-trying-build-truly-permanent-web/>. Erişim tarihi: 30.12.2021
- Fullana, O. ve Ruiz, J. (2021). Accounting information systems in the blockchain era. *International Journal of Intellectual Property Management*, 11(1), 63–80.
- Gelir İdaresi Başkanlığı. (2013). e-Fatura Portalı Kullanım Kılavuzu. *e-Fatura Portalı Kullanım Kılavuzu*. Ankara: Gelir İdaresi Başkanlığı.
- Guerar, M., Merlo, A., Migliardi, M., Palmieri, F. ve Verderame, L. (2020). A fraud-resilient blockchain-based solution for invoice financing. *IEEE Transactions on Engineering Management*, 67(4), 1086–1098.
- Haber, S. ve Stornetta, W. S. (1991). How to time-stamp a digital document. *Journal of Cryptology*, 3(2), 99–111. doi:10.1007/BF00196791
- Hambiralovic, M. ve Karlsson, R. (2018). Blockchain Accounting in a Tripple-Entry System.
- Jamil, F., Hang, L., Kim, K. ve Kim, D. (2019). A Novel Medical Blockchain Model for Drug Supply Chain Integrity Management in a Smart Hospital. *Electronics* . doi:10.3390/electronics8050505
- Lau, D., Lau, D., Jin, T. S., Kho, K., Azmi, E., Lee, T. M., & Ong, B. (2020). How to Defi (Vol. 1). <https://www.coingecko.com/account/rewards/how-to-defi-book?locale=tr>. Erişim tarihi: 02.01.2022
- Lazanis, R. (2015). How Technology Behind Bitcoin Could Transform Accounting As We Know It. *BrainStation*. 22 Ocak. <https://brainstation.io/magazine/how-technology-behind-bitcoin-could-transform-accounting-as-we-know-it>. Erişim tarihi: 01.01.2022
- Liu, M. ve Fraser, J. (2018). *Origin White Paper*. <https://www.originprotocol.com/en/whitepaper>. Erişim tarihi: 23.12.2021
- Merkle, R. C. (1987). A Digital Signature Based on a Conventional Encryption Function. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 293 LNCS, 369–378. doi:10.1007/3-540-48184-2_32
- Mittal, V. (2017). A survey of Blockchain Technologies for Open Innovation. *World Open Innovation Conference 2017*, (November), 1–27.
- Nakamoto, S. (2008). Bitcoin: A peer-to-peer electronic cash system. *Decentralized Business Review*, 21260.

- Narayanan, K., Goel, S., Singh, A., Shrinivasan, Y., Chakraborty, S., Selvam, P., Verma, M. (2020). Blockchain based e-invoicing platform for global trade. *2020 IEEE International Conference on Blockchain (Blockchain)*, ss. 385–392. IEEE.
- Narayanan, A., Bonneau, J., Felten, E., Miller, A. ve Goldfeder, S. (2016). *Bitcoin and cryptocurrency technologies: a comprehensive introduction*. Princeton University Press.
- Özdoğan, B. ve Karğın, S. (2018). Blok zinciri teknolojisinin muhasebe ve finans alanlarına yönelik yansımaları ve beklentiler. *Muhasebe ve Finansman Dergisi*, (80), 161–176.
- Özkul, F. ve Alkan, B. Ş. (2020). Dijital çağda muhasebenin dönüşümü:“Blockchain” teknolojisinde muhasebe ve mali kontroller. *Muhasebe Bilim Dünyası Dergisi*, 22(2), 218–236.
- Pascual Pedreño, E., Gelashvili, V. ve Pascual Nebreda, L. (2021). Blockchain and its application to accounting. *Intangible Capital*, 17(1), 1–16.
- Resmi Gazete. (2010). Vergi Usul Kanunu Genel Tebliği. *Resmi Gazete*. 5 Mart. <https://www.gib.gov.tr/node/88067>. Erişim tarihi: 28.12.2021
- Sharma, A. (2018, 24 Ağustos). 5 Trends Shows How Blockchain Is Changing Social Media | Hacker Noon. *Hackernoon*. 24 Ağustos. <https://hackernoon.com/5-trends-shows-how-blockchain-is-changing-social-media-ba50c975c041>. Erişim tarihi: 26.11.2021
- Vergi Usul Kanunu. (1961). Vergi Usul Kanunu. <https://www.mevzuat.gov.tr/mevzuatmetin/1.4.213.pdf>. Erişim tarihi: 20.01.2022
- Yanık, R. ve Karadaş, A. (2013). E-Faturanın Türkiye Muhasebe Standartları Uyum Sürecine Uygun Düzenlenmesine İlişkin Bir Öneri. *Ekev Akademi Dergisi*, 57(57), 133–142.
- Web3Js (2021). Ethereum JavaScript API. <https://web3js.readthedocs.io/en/v1.7.0/index.html>. Erişim tarihi: 02.12.2021