

**MARMARA ÜNİVERSİTESİ**

**TEKNOLOJİ FAKÜLTESİ**

**BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**

**BİTİRME PROJESİ**

Blok Zinciri Tabanlı   
Öğrenci Bilgi Sistemi

**PROJE YAZARLARI**

Ahmet Faruk ALKAN

Kasım Selimhan BALTAŞ

**DANIŞMAN**

Dr. Öğr. Üyesi Eyüp Emre ÜLKÜ

**İSTANBUL, 2024**

**MARMARA ÜNİVERSİTESİ**

**TEKNOLOJİ FAKÜLTESİ**

**BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**

Marmara Üniversitesi Teknoloji Fakültesi Bilgisayar Mühendisliği Öğrencisi ……………………… nın “…………………………” başlıklı bitirme projesi çalışması, …./…./….. tarihinde sunulmuş ve jüri üyeleri tarafından başarılı bulunmuştur.

**Jüri Üyeleri**

Prof. Dr. Adı SOYADI (Danışman)

Marmara Üniversitesi ......................................................................... (İMZA) ..................

Doç. Dr. Adı SOYADI (Üye)

Marmara Üniversitesi ......................................................................... (İMZA) ..................

Dr. Öğr. Üyesi Adı SOYADI (Üye)

Marmara Üniversitesi ......................................................................... (İMZA) ..................

İÇİNDEKİLER

Sayfa

[ÖZET 7](#_Toc168236559)

[Bölüm 1- GİRİŞ 9](#_Toc168236560)

[1.1 Literatür Taraması ve İlişkili Çalışmalar 10](#_Toc168236561)

[1.2 Blockchain Teknolojisi 11](#_Toc168236562)

[1.3 Ethereum 12](#_Toc168236563)

[1.4. Problem Tanımı 13](#_Toc168236564)

[Bölüm 2- MATERYAL VE YÖNTEMLER 15](#_Toc168236565)

[2.1 Blockchain 15](#_Toc168236566)

[2.1.1 Ethereum 16](#_Toc168236567)

[2.1.2 Solidity ile Akıllı Kontrat Geliştirme 16](#_Toc168236568)

[2.1.3 Remix IDE 17](#_Toc168236569)

[2.1.4 Kontrat Geliştirme 18](#_Toc168236570)

[2.2 Ganache Yerel Ethereum Ağı 20](#_Toc168236571)

[2.3 ExpressJS 21](#_Toc168236572)

[2.3.1 ExpressJS ABI Bağlantısı 21](#_Toc168236573)

[2.3.2 AuthorizeRole ve Bearer Token 22](#_Toc168236574)

[2.3.3 Cüzdan sistemi 24](#_Toc168236575)

[2.4 React 25](#_Toc168236576)

[2.4.1 React Express bağlantısı 25](#_Toc168236577)

[2.4.2 RTK Query 25](#_Toc168236578)

[2.4.3 Tailwind 26](#_Toc168236579)

[2.5 Öğrenci Bilgi Sistemi 26](#_Toc168236580)

[2.5.1 Projenin İşlevleri 27](#_Toc168236581)

[2.5.1.1 Üye Kaydı ve Roller 27](#_Toc168236582)

[2.5.1.3 Ders Seçimi ve Derse Kayıtlanma 28](#_Toc168236583)

[2.5.1.4 Transkript 30](#_Toc168236584)

[2.5.2.6 Notlandırma 31](#_Toc168236585)

[2.5.2.7 Ders Programı 31](#_Toc168236586)

[2.5.2.8 Sınıf Paylaşım Planı 32](#_Toc168236587)

[Bölüm 3- Bulgular ve Tartışmalar 34](#_Toc168236588)

[3.1 Güvenlik 34](#_Toc168236589)

[3.2 Performans ve Ölçeklenebilirlik 34](#_Toc168236590)

[3.3 Merkezi ve Dağıtık Sistemlerde DDoS Saldırıları ve Gas Maliyetleri 35](#_Toc168236591)

[Bölüm 4- Sonuçlar 37](#_Toc168236592)

[4.1 İleri Çalışmalar 37](#_Toc168236593)

[KAYNAKLAR 38](#_Toc168236594)

[EK 1 41](#_Toc168236595)

**KISALTMALAR/ABBREVIATIONS**

**ABI:** Application Binary Interface (Uygulama İkili Arayüzü)

**BYS:** Bilgi Yönetim Sistemi

**CRUD:** Create, Read, Update, Delete

**dAPP:** Decentralized App (Merkezi olmayan uygulama)

**DDoS:** Distributed Denial of Services (Dağıtılmış Hizmet Engelleme)

**EVM:** Ethereum Virtual Machine (Ethereum Sanal Makinesi)

**JS:** Javascript

**ŞEKİL LİSTESİ**

Sayfa

[**Şekil 1:** Blockchain düğüm yapısı [20] 16](#_Toc168236275)

[**Şekil 2:** Remix IDE’de solidity ile akıllı kontrat geliştirme arayüzü 17](#_Toc168236276)

[**Şekil 3:** Akıllı kontrat ilişki şeması 18](#_Toc168236277)

[**Şekil 4:** Ganache arayüzü 21](#_Toc168236278)

[**Şekil 5:** Rol yetkinlik diyagramı 23](#_Toc168236279)

[**Şekil 6:** Proje mimarisi 27](#_Toc168236280)

[**Şekil 7:** Yönetici, yeni kaydolmuş üyelerin listesini görüntüler 28](#_Toc168236281)

[**Şekil 8:** Departman sayfası: Ders listesinden ders açma 29](#_Toc168236282)

[**Şekil 9:** Açılan dersin düzenlenmesi 29](#_Toc168236283)

[**Şekil 10:** Öğrenci ders seçim arayüzü 30](#_Toc168236284)

[**Şekil 11:** Danışman öğretmen ders seçimi incelemesi 30](#_Toc168236285)

[**Şekil 12:** Öğretim görevlisi verdiği derslerin notlandırmasını yapar 31](#_Toc168236286)

[**Şekil 13:** Ders programı 32](#_Toc168236287)

[**Şekil 14:** Paylaşım alanı için ders seçimi 32](#_Toc168236288)

[**Şekil 15:** Paylaşım alanı 33](#_Toc168236289)

[**Şekil 16:** Dağıtılmış cüzdan sistemi 41](#_Toc168236290)

[**Şekil 17:** Yöneticilerin kullanıcıya rol ataması 41](#_Toc168236291)

[**Şekil 18:** Öğrencilerin görüntüledikleri transkript belgesi. 42](#_Toc168236292)

## ÖZET

Blok zinciri teknolojisinin verilerin güvenliğini sağlamadaki önemi yadsınamaz. Günümüzde öğrenci bilgi yönetim sistemi gibi, üniversitede eğitim veren ve eğitim alan tüm bireylerin hassas verilerini güvenle depolayıp yönetimini sağlayan sistemlerde genellikle merkezileştirilmiş tasarım kullanılmaktadır. Eğitim sektöründeki sertifika ve diploma gibi önemli dökümanların yanında öğrencilerin notları ve tüm kullanıcıların kişisel bilgileri gibi verilerin güvenliği de sağlanmalıdır. Bu sistemlerin dağıtık sistemli uygulama (dApp: decentralized application) olarak tasarlanması güvenliğin arttırılmasında önemli rol oynayacaktır. Bu çalışma kapsamında, öğrenci bilgi yönetim sistemi güvenliğini arttırmak amacıyla dağıtılmış sistem tasarımı ile gerçekleştirilmiştir. Sistem, öğrenci kaydı, not girişi, devamsızlık takibi, ders programı yönetimi ve transkript oluşturma gibi temel işlemleri kapsamaktadır. Bu işlemler, blok zinciri teknolojisi ve kriptoloji teknikleri ile güvenli ve şeffaf bir şekilde gerçekleştirilmiştir. Kullanıcılar, web arayüzü aracılığıyla sisteme erişebilir ve kendi verilerini yönetebilirler. Blok zinciri, verilerin merkezi olmayan bir yapıda güvenli bir şekilde saklanmasını sağlar, böylece veri manipülasyonu ve yetkisiz erişim riskleri azaltılmaktadır. Sistem, kimlik bilgileri, akademik bilgiler, ders programları ve transkriptler gibi çeşitli veri türlerini güvenli bir şekilde depolamaktadır. Bu proje, öğrenci bilgilerini güvenli, şeffaf ve değişmez bir şekilde saklamanın yanı sıra, veri gizliliği, güvenliği ve kullanıcı dostu bir deneyim sunmayı amaçlamaktadır. Eğitim yönetimi ve bilgi teknolojileri alanlarındaki uzmanlar için bir referans kaynağı olarak hizmet etmeyi amaçlayan bu çalışma, eğitim kurumlarının bilgi yönetimi süreçlerini iyileştirmeye katkıda bulunmayı hedeflemektedir.

**ABSTRACT**

The importance of blockchain technology in ensuring data security is undeniable. Today, centralized designs are generally used in systems that securely store and manage sensitive data of all individuals who teach and receive education at universities, such as student information management systems. In addition to important documents such as certificates and diplomas in the education sector, the security of data such as students' grades and personal information of all users should also be ensured. Designing these systems as a distributed system application (dApp) will play an important role in increasing security. Within the scope of this study, a distributed system design was implemented to increase the security of the student information management system. The system covers basic operations such as student registration, grade entry, absence tracking, course program management and transcript creation. These operations are carried out securely and transparently with blockchain technology and cryptology techniques. Users can access the system and manage their own data via the web interface. Blockchain ensures that data is stored securely in a decentralized structure, thus reducing the risks of data manipulation and unauthorized access. The system securely stores various types of data such as identity information, academic information, course programs and transcripts. This project aims to provide data privacy, security and a user-friendly experience, as well as to store student information in a secure, transparent and immutable manner. This study aims to serve as a reference source for experts in the fields of education management and information technologies, and contributes to improving the information management processes of educational institutions.

# Bölüm 1- GİRİŞ

Teknoloji sürekli ve hızla gelişmekte olup, bu yeniliklere uyum sağlayabilmek, günümüzün çeşitli sektörlerindeki rekabetin kilit bir unsuru haline gelmiştir. Blockchain teknolojisi günümüzde yalnızca kripto varlık işlemleri için değil, güvenli bir altyapı sağladığı için birçok sektörde de tercih edilmektedir [1-4]. Bunlardan biri de eğitim sektörüdür. Eğitim kurumlarındaki öğrenciler, öğretmenler ve diğer görevlilerin hassas kişisel bilgilerinin, eğitim verileri ve sahtecilik yapılma olasılığı yüksek olan sertifika veya diploma gibi belgelerin güvenli bir altyapıda depolanması oldukça büyük önem arz etmektedir [5].

Öğrenci bilgi sistemlerinin sistem tasarımları yıllar içerisinde çeşitli teknolojik gelişmeler ve yeniliklerle şekillenmiştir. Geleneksel Bilgi Yönetim Sistemi (BYS), manuel kayıt tutmadan, veri doğruluğunu geliştiren, maliyetleri azaltan, verimliliği artıran ve karar verme yeteneklerini geliştiren kompleks veritabanı sistemlerine doğru gelişmiştir.

Ancak mevcut bu sistemlerin de bazı sınırlamaları vardır. Örneğin, öğrenci kimliklerini doğrulamak ve akademik ilerlemeyi takip etmek için aracılara güvenerek genellikle merkezileştirilirler; bu da güvenlik riskleri ve şeffaflığa sınırlamalar getirebilir. Ek olarak, merkezi veritabanlarının kullanımı, eğitim kurumları için ek altyapı yatırımları gerektirebilir ve kullanıcılar, yararları ve riskleri konusunda net bir rehberlik olmadan yeni teknolojileri benimsemekte tereddüt edebilirler.

Bu sistemlerin mevcut durumu, şeffaflık eksikliği, güvenlik kaygıları ve sınırlı erişilebilirlik ile karakterize edilmektedir. Öğrenci verilerini depolamanın ve yönetmenin geleneksel yöntemleri büyük ölçüde merkezi veritabanlarına dayanır ve bu veritabanları siber saldırılara, veri ihlallerine ve yetkisiz erişime karşı savunmasız olabilir. Eğitim için dijital platformlara olan bağımlılığın artması, oluşturulan öğrenci bilgilerinin hacminde ciddi bir artışa yol açarak eğitim kurumlarının bu verileri etkili bir şekilde yönetmesini zorlaştırdı. Üstelik geleneksel yöntemler çoğu zaman öğrencilerin akademik performansı, katılımı ve genel ilerlemesi hakkında kapsamlı bir görünüm sağlamada başarısız olur.

Blockchain teknolojisinin çeşitli sektörlerde yaygın olarak benimsenmesi, eğitimdeki potansiyel blockchain uygulamalarına olan ilgiyi artırdı. Blockchain'in merkezi olmayan yapısı, değişmezliği ve şeffaflık özellikleri, onu güvenli veri yönetimi için çekici bir çözüm haline getiriyor. Eğitim sektöründe Blockchain’in uygulanması, güvenlik, değiştirilemezlik, kurumdan bağımsızlık, resmi kayıtların ve sertifikaların değiştirilemezliği dahil ancak bunlarla sınırlı olmayan bir dizi işlevsel olmayan gereksinim için yeni bir ufuk sağlar. Bu nedenler doğrultusunda bu çalışma blockchain tabanlı bir BYS hazırlamayı amaçlamaktadır.

## 1.1 Literatür Taraması ve İlişkili Çalışmalar

Han M. ve arkadaşları ortaya koydukları çalışmada Jakarta'daki yirmi dört üniversitenin kullandığı mevcut McRhys modelini tartışmışlardır [6]. McRhys, Blockchain teknolojisini kullanarak üniversitedeki tüm etkinlikleri kaydeden ve entegre eden bir modeldir. Kaydedilen notlar değiştirilemez olmalı ve defter tabanlı bir sisteme kaydedilmelidir. Bir dizi sınav işareti, değişmez Blockchain defterine(ledger) kaydedildiğinde (işlem eklendiğinde), hiç kimse bunu değiştiremez. Akademik alandaki bir konu olarak Turkanovic M. ve arkadaşlarının çalışmasında, Blockchain'e, eğitim kayıtlarına ve sisteme eklenen bileşenlere dayalı yeni bir çerçeve önermişler ve ardından bunu tüm ilgili taraflar için yayınlamak üzere Blockchain'de dağıtmışlardır [7].

Dong Z. ve arkadaşlarının çalışmasında, EduCTX önerilen küresel bir yüksek öğrenim kredi platformudur [8]. Bu platform Avrupa Kredi Transfer ve Biriktirme Sistemi (ECTS) fikri üzerine inşa edilmiştir. Öğrenciler ve yükseköğretim kurumlarının yanı sıra işletmeler, kurumlar ve organizasyonlar gibi diğer potansiyel paydaşlara küresel olarak birleşik bir bakış açısı sağlayabilecek, dünya çapında güvenilir, merkezi olmayan bir yükseköğretim kredi ve derecelendirme sistemidir. Konseptin bir kanıtı olarak açık kaynaklı Ark Blockchain Platformunu temel alan ekosistemin prototip uygulamasını göstermişler. Bu çalışmaya göre, Blockchain'in geleceğin şebekesi için güvenli, dağıtılmış bir siber altyapı olarak kullanımına ilişkin bilgiler sağlanmıştır.

Seebacher S. ve arkadaşlarının çalışmasında, Blockchain teknolojisinin, çok çeşitli olası uygulama sektörlerini etkileyerek işlemlerin yapılma şeklini dönüştüreceği tahmin ediliyordu [9]. Blockchain teknolojisi, farklı tarafların iş birliği yapmasına olanak tanıyan eşler arası bir ağa dayandığından, hizmet sistemi, potansiyel katkısını değerlendirmek için bir birim analizi olarak seçilmiştir. Güveni ve merkezi olmayan yönetimi teşvik eden, bir hizmet sisteminin geliştirilmesini ve koordinasyonunu kolaylaştıran bir dizi nitelik keşfettiler.

P. Haveri ve arkadaşları,çalışmalarında Ethereum blockchain'i ve Solidity programlama dili kullanılarak "EduBlock" adında bir öğrenme yönetim sistemi (LMS) geliştirilmiştir [10]. Sistem, öğrenci kayıtları, kurs içerikleri, notlar ve ödevler gibi eğitim verilerini depolamak için blockchain teknolojisini kullanır. Ethereum blockchain'i, Solidity programlama dili, IPFS (InterPlanetary File System), Akıllı sözleşmeler. BlockEDU'nun geleneksel LMS'lerden daha güvenli, şeffaf ve veriye erişimi kontrol eden bir sistem olduğu kanıtlanmıştır.

S. K. Ambast ve arkadaşlarının çalışmasında ise, öğrencilerin akademik kimlik bilgilerini işlemek, depolamak ve doğrulamak için güvenli, merkezi olmayan bir arayüz sağlamak üzere blockchain'in gücünü kullanan, kimlik bilgileri formatı ve boyutundan bağımsız olarak blockchain tabanlı bir eğitimsel kimlik doğrulama sistemi önerilmektedir [11].

Z. Changhua ve arkadaşlarının çalışmasında ise, öğrencilerin kişisel bilgileri, dersleri ve ilgili sertifikalarının öğrenme derinliği ile öğrenme veya öğretim yöntemleri hakkında detaylı bilgi sağlayamaması sorununu ele almaktadır. Öğrencilerin öğrenme bilgilerini, öğrenme kayıtlarını ve Ethereum blok zinciri ile ilgili teknolojiyi araştırma nesnesi olarak kullanmayı hedeflemektedir. Ethereum blok zinciri ve akıllı sözleşmeleri kullanarak öğrenenlerin öğrenme kayıtlarını bağlamayı amaçlamaktadır [12].

## 1.2 Blockchain Teknolojisi

Blockchain kullanımı, taraflar arasında değiştirilen verilerin/işlemlerin güvenliğini ve gizliliğini akıllı kontratlar aracılığıyla sağlar. Blok zincirindeki eşler arası topolojinin doğası, eğitim alanında da güvenlik risklerinin azaltılmasına yardımcı olur. Farah ve arkadaşlarının ortaya koydukları çalışmada mutabakat protokolü işlemlerin sırasını korumak ve teminatsız riskleri azaltmak için kullanılmış olup, amaçlanan taraflar defter(ledger) içeriğini doğrulayabilmektedir [13]. Kriptografi ve imzaların kullanılması, işlemlerin güvenilirliğini arttırmıştır. Blockchain'in güvenilirliği, verinin kendisinin güvence altına alınmasına ve tahrif edilmediğinin doğrulanmasına dayanır. Bdiwi ve arkadaşlarının çalışmasındaki nesnelerin interneti ile gerçekleştirilen uygulamada, sensörler ve akıllı cihazlardan alınan eğitim verilerini imzalamak ve doğrulamak için blockchain kullanıldı [14]. Ayrıca eğitim alanı, maliyetleri azaltmak için blockchain'in kullanılmasından büyük fayda sağlayabilir. Maliyet, depolama maliyetini, ilgili işlem maliyetlerini ve eğitim kayıtlarını yönetme ve sürdürme maliyetini içerir. Her yerden erişilebilen genel/özel dağıtılmış bir ağ kullanmak, geleneksel bulut tabanlı depolamanın maliyetini önemli ölçüde azaltır [6]. Genel olarak akademik sertifikaların doğrulanması ve işlenmesi ek maliyetler gerektirirken, blockchain bu tür maliyetlerin azaltılmasına yardımcı olur [15].

Blockchain'in en büyük özelliklerinden ve faydalarından biri, saklanan kayıtlara erişimin kısıtlanması/kontrol edilmesidir. Eğitim kayıtları transkriptleri, diplomaları veya kişisel öğrenci/öğretmen kayıtlarını içerir. Arenas ve arkadaşlarının çalışması iyi bir örnek olabilir. Akademik kimlik bilgilerine erişimi kısıtlamak ve bunu yalnızca amaçlanan katılımcılarla sınırlamak için izinli bir blockchain platformu tasarlanmıştır [16]. Blockchain platformu, yalnızca belirli kurallara tabi sertifikalı kurumların saklanan verilere erişmesine ve bunları değiştirmesine izin verir [6]. Dahası, hesap verebilirliğin ve şeffaflığın arttırılması, blockchain teknolojisi kullanıldığında elde edilen iki avantajdır.

Güvenilirlik, blockchain teknolojisini kullanmanın getirdiği bir başka avantajdır. Ağa blok ekleme işlemine yalnızca güvenilir taraflar katılabilir ve yalnızca güvenilir taraflar erişim sağlayabilir. Farklı bölgelerden farklı yetkililerle iş yaparken güven büyük bir endişe kaynağıdır. Üniversiteler veya herhangi bir eğitim kurumu, güvenli ve güvenilir blockchain tabanlı sistemler uygulayarak güvenilir bir topluluk oluşturabilir. Kredi ve derecelendirme blockchain tabanlı bir platform olan EduCTX tanıtılmıştır [7]. Jetonları güvenilir taraflara aktarır. Bu jetonlar öğrencilerin kayıtlarında kazanılan tamamlanmış kredilere bağlıdır. Bu nedenle yükseköğretim organizasyonu için örnek bir küresel, güvenilir ve birleşik sistem oluşturulmuştur.

## 1.3 Ethereum

Ethereum, akıllı sözleşme işlevlerine sahip açık kaynaklı, halka açık bir Blockchain platformudur [17]. Özel kripto para birimi Ether(eth) aracılığıyla eşler arası kontratları işlemek için merkezi olmayan bir Ethereum Sanal Makinesi (EVM: Ethereum Virtual Machine) sağlar.[18]. Blockchain, üretilen veriyi belirli bir zaman aralığında kaydedilmek üzere veri bloklarına bölerek, daha sonra bunları veri bloklarının sırasına göre seri olarak bağlamaktır. EVM kod yürütme ortamı, Ethereum Blockchain üzerinde uygulanabilir. Zincir üzerinde yazma işlemleri gerçekleştirirken ağdaki her tam düğüm aynı hesaplamayı yapacak ve aynı değeri depolayacaktır. Ethereum ağında her bir yazma işlemi, "gas" adı verilen bir işlem ücreti gerektirir. Bu ücret, işlemi gerçekleştiren madencilere ödeme olarak verilir ve aynı zamanda gereksiz işlemleri caydırmaya yarar. Gas ücreti, işlemin karmaşıklığına ve ağdaki yoğunluğa göre değişir. Yoğunluk arttıkça gaz ücretleri de artar. Bu nedenle Ethereum'da işlem yapmak, özellikle ağ yoğun olduğunda, pahalı olabilir.

Bu durum, Ethereum kullanıcılarını işlemlerin bir kısmını zincir dışında gerçekleştirmeye teşvik eder. Ancak, bu işlemlerin güvenliğini sağlamak için genellikle ek güvenlik önlemleri alınması gerekir. Geliştiricilerin akıllı sözleşmelerini tasarlarken gaz tüketimini optimize etmeleri gerekmektedir [19].

## 1.4. Problem Tanımı

Literatür taramasından elde ettiğimiz veriler, Blockchain’in eğitim uygulamalarında öğrencilerin eğitim verilerinin yönetiminde kullanılmasının gerekliliği ve önemini ortaya koymaktadır. Bununla ilgili sistem tasarım önerileri sayıca fazla olmasına rağmen uygulanmış örnek sistem sayısı çok azdır. Mevcut öğrenci bilgi yönetimi sistemlerinin etkinliğini engelleyen çeşitli sınırlamaları vardır. İlk olarak, genellikle hatalara, gecikmelere ve verimsizliklere yol açabilecek manuel süreçlere güveniyorlar. İkinci olarak, veri depolama ve paylaşımında şeffaflık ve hesap verebilirliğin olmayışı öğrencilerin mahremiyeti ve güvenliğine ilişkin kaygıları artırmaktadır. Üçüncüsü, geleneksel yöntemler çoğu zaman esnek değildir, bu da diğer sistemlerle entegrasyonu veya değişen eğitim gereksinimlerine uyum sağlamayı zorlaştırır.

Üstelik mevcut sistemler öğrenci performansı, katılımı ve ilerlemesine ilişkin kapsamlı bir görünüm sağlayamayabilir, bu da eğitimcilerin iyileştirilmesi gereken alanları belirlemesini zorlaştırabilir. Gerçek zamanlı veri erişiminin olmaması, öğrencilerin akademik gidişatları hakkında bilinçli kararlar verme yeteneğini de engelleyebilir. Dahası, geleneksel yöntemler çoğu zaman engelli öğrencilerin veya yeterince temsil edilmeyen gruplardan gelenlerin çeşitli ihtiyaçlarını hesaba katmakta başarısız olmaktadır.

**1.5. Hedefler**

Bu çalışmada gerçekleştirdiğimiz öğrenci bilgi sisteminin bu konudaki örnek çalışmaları arttırarak eğitim sektöründe blockchain kullanımının yaygınlaştırılmasını hedefliyoruz.Bu projenin temel hedefleri: Güvenli, şeffaf ve erişilebilir veri yönetimi sağlayan blockchain tabanlı öğrenci bilgi sistemi tasarlamak ve geliştirmek; bu blockchain arka-ucunu tam teşekküllü bir web projesi olarak gerçekleştirmektir.

**1.6. Bu çalışmanın önemi**

Bu çalışmanın önemi, eğitim kurumlarının öğrenci bilgilerini yönetme biçiminde devrim yaratma potansiyeline sahip olmasıdır. Bu proje, blockchain teknolojisinden yararlanarak öğrenci verilerinin saklanması ve paylaşılması için güvenli ve şeffaf bir platform sağlamayı amaçlamaktadır. Dahası, blockchain'in eğitimle entegrasyonunun üniversite BYS’sini kullanan tüm kullanıcıların mahremiyeti ve güvenliği için önemlidir. Eğitim kurumları dijital platformlara giderek daha fazla bağımlı hale geldikçe şeffaflığı, güvenliği ve erişilebilirliği ön planda tutan yenilikçi çözümleri keşfetmek hayati önem taşıyor.

**1.7. Kapsam ve Sınırlamalar**

Bu projenin kapsamı aşağıda belirtilen teknolojileri içermektedir. Çalışma bu teknolojiler ve platformlar kullanılarak geliştirilmiştir:

Ganache (Ethereum), Solidity, Express.js, Remix IDE, React ve Tailwind kullanarak blockchain tabanlı öğrenci bilgi sisteminin tasarlanması ve geliştirilmesi; geleneksel öğrenci bilgi yönetimi sistemlerinde mevcut sınırlamaların kapsamlı bir analizinin yapılması.

**1.8. Çıktılar**

Bu çalışmada Öğrenci Bilgi Sistemi hazırlanırken Solidity ile sistem verilerinde kaydetme, okuma, güncelleme ve silme gibi tüm veri manipülasyonu işlemlerinin gerçekleştirileceği akıllı kontratlar hazırlanarak, Ganache’nin sağladığı yerel test Ethereum ağında dağıtılarak, bu kontratların ABI’lerine(Application Binary Interface) Express.js arka-uç çerçevesinden(framework) erişim sağlanmıştır. Yerel test ağı kullanmanın amacı, Ethereum ağında kontratları dağıtmanın ve bu kontratlarla işlem yapmanın bir gas bedeli olması ve kontrat geliştirme sürecinde bu gas bedelini ödememek için aynı ortamı simüle eden yazılımların tercih edilmesidir. Bu kontratlarda yapılacak işlemlerin kullanıcı arayüzleri ise React kullanılarak gerçekleştirilmiştir. Ön-yüz’de React ve Tailwind teknolojileri kullanılarak kullanıcı dostu, açık ve anlaşılır kullanıcı arayüzleri tasarlanmıştır. Eğitim alanında blockchain kullanımının literatür taramasına göre en kapsamlı uygulaması gerçekleştirilmiştir.

# Bölüm 2- MATERYAL VE YÖNTEMLER

Bu bölümde, bu çalışmada kullanılan teknolojiler, çalışmanın teknik detayları ve uygulama sürecinin nasıl gerçekleştirildiği açıklanmıştır. Uygulama sürecinde karşılaşılan bazı zorluklar ve bunların nasıl çözüldüğü hakkında bilgi verildi. Ganache Ethereum test ağında dağıtılmış akıllı kontratlardan oluşan arka-uç mimari tasarımından detaylıca bahsedildi.

## 2.1 Blockchain

Blockchain teknolojisi, tahrifata, sansüre veya silinmeye karşı dayanıklı dijital bir defter (ledger) oluşturulmasına olanak tanır ve böylece merkezi olmayan sistemler ve uygulamalar için güvenli bir temel sağlar.

Şekil 2.1’de görülen, blockchain'deki ilk blok olan Genesis Bloğu'nun hash değeri, blockchain'in başlangıç noktasını temsil eder. Her yeni blok, kendinden önceki bloğun hash değerini içerir. Bu, blokların birbirine güvenli bir şekilde bağlanmasını sağlar. Bir bloktaki verilerin herhangi bir şekilde değiştirilmesi, o bloğun hash değerini değiştirir ve bu değişiklik takip eden blokların hash değerlerini de etkiler. Bu nedenle, blockchain üzerindeki veriler değiştirilemez ve güvenli bir şekilde saklanır.

metin, ekran görüntüsü, yazı tipi, sayı, numara içeren bir resim

Açıklama otomatik olarak oluşturuldu

**Şekil 1:** Blockchain düğüm yapısı [20]

Hash fonksiyonları, herhangi bir boyuttaki veriyi sabit uzunlukta bir çıktıya dönüştüren matematiksel fonksiyonlardır. Blockchain'de kullanılan hash fonksiyonları, verilerin bütünlüğünü korumak ve blokları birbirine bağlamak için kullanılır. Bu yapı aynı zamanda bağlı liste (linked list) yapısına da benzer.

Blockchain üzerinde gerçekleştirilen her işlem, bir bloğa kaydedilir. Yeni bir blok oluşturulduğunda, ağdaki diğer düğümler bloğun bu bloğun hash değerinin doğruluğunu kontrol ederek doğrular. Doğrulanan bloklar, blockchain'e eklenir ve kalıcı hale gelir.

Blockchain, merkezi olmayan bir yapıya sahiptir, yani tek bir otorite tarafından kontrol edilmez. Bunun yerine, ağdaki birçok düğüm tarafından yönetilir. Bu dağıtık yapı, blockchain'i güvenli, şeffaf ve sansüre karşı dirençli hale getirir.

## 2.1.1 Ethereum

Ethereum, akıllı sözleşmelerin oluşturulmasını kolaylaştırmak ve merkezi olmayan uygulamaları (dApps) yürütmek için blockchain teknolojisini kullanan açık kaynaklı, merkezi olmayan bir blockchain yapısıdır. Geliştiricilerin Solidity dilinde kendi kendine çalışan kod yazmasına olanak tanıyan Ethereum Sanal Makinesi (EVM) üzerine inşa edilmiştir. Ethereum ağı, Bitcoin'inkine benzer, ancak ölçeklenebilirlik ve esnekliğe daha fazla odaklanır.

Temelde Ethereum, akıllı sözleşmeler aracılığıyla merkezi olmayan uygulamaların (dApp'ler) oluşturulmasına olanak tanıyan, taraflar arasında güvenli, şeffaf ve sansüre dayanıklı işlemler ve etkileşimler sağlayan, blockchain tabanlı bir platformdur.

## 2.1.2 Solidity ile Akıllı Kontrat Geliştirme

Solidity, Ethereum blok zincirinde akıllı sözleşmeler geliştirmek için kullanılan yüksek seviyeli bir programlama dilidir. Geliştiricilerin, Ethereum ağı üzerinde dağıtılabilen, kendi kendini çalıştıran akıllı kontratları yazmasına olanak tanıyarak, merkezi olmayan uygulamaların (dApp'ler) hazırlanmasını mümkün kılar.

Akıllı sözleşmeler blockchain üzerinde çalışır; sansür, kesinti, dolandırıcılık veya üçüncü taraf müdahalesi olasılığı olmaksızın tam olarak programlandığı gibi çalışırlar [21].

Solidity ile akıllı sözleşme geliştirmek için solidity compiler kullanabilecek bir ide’ye ihtiyaç vardır. Visual Studio Code[22] bunun için kullanılabilir fakat bu proje geliştirilirken Remix IDE kullanıldı. Remix IDE’nin solidity’e odaklanmış olması ve çok daha fazla geliştirici seçeneği sunması, seçimde ön planda tutulan etkenler arasındadır [23].

## 2.1.3 Remix IDE

Remix IDE (Entegre Geliştirme Ortamı), geliştiricilerin Solidity programlama dilini kullanarak Ethereum blok zincirinde akıllı sözleşmeler oluşturmasına, dağıtmasına ve yönetmesine olanak tanıyan açık kaynaklı, web tabanlı bir geliştirme ortamıdır. Şekil 2’de arayüzü verilen Remix IDE, sözdizimi vurgulama, kod tamamlama, hata ayıklama araçları ve akıllı sözleşmeleri test etmek ve dağıtmak için yerleşik bir derleyici gibi özellikler sağlar.

metin, ekran görüntüsü, yazılım, sayı, numara içeren bir resim

Açıklama otomatik olarak oluşturuldu

**Şekil 2:** Remix IDE’de solidity ile akıllı kontrat geliştirme arayüzü

Remix IDE, özünde, Ethereum ağı üzerinde merkezi olmayan uygulamalar (dApp'ler) geliştirmek ve yönetmek için kullanıcı dostu bir platformdur ve geliştiricilerin düşük seviyeli blockchain ayrıntıları hakkında endişelenmeden güvenli ve verimli akıllı sözleşme mantığı yazmaya odaklanmasını sağlar.

Akıllı kontrat kodları yazıldıktan sonra, derleyici kısmından bir derleyici sürümü seçilir. Bu projede “0.8.0+commit.c7dfd78e” sürümü kullanıldı. “Ctrl + S” veya derleme butonuna basılarak “.sol” uzantılı dosyalar derlenir, derlenen akıllı sözleşmeler, “Deploy” sekmesinden dağıtılabilir. Bu sekmede ilk olarak deploy edilecek ortam(environment) seçilir. Daha önce de bahsedildiği üzere bu proje için Ganache servisi kullanıldı, bu nedenle ortam olarak “Dev – Ganache Provider” seçildi. Ethereum ağında akıllı sözleşmelerde işlem yapmanın yanında, bu akıllı sözleşmeleri dağıtmak için de bir gas bedeli ödemek gerekir, bu nedenle bu kısımda bir cüzdan adresi belirtilmesi gerekiyor. Bu adımdan sonra da derlenen akıllı sözleşmeler seçilerek dağıtılır.

## 2.1.4 Kontrat Geliştirme

Bu proje için Remix IDE’de Solidity 0.8.0 sürümü ile akıllı kontratlar geliştirildi. Toplamda 12 adet akıllı kontrat hazırlandı. Akıllı kontratların birbirleriyle ilişki şeması şekil 3’te verilmiştir.

diyagram, metin, çizgi, plan içeren bir resim

Açıklama otomatik olarak oluşturuldu

**Şekil 3:** Akıllı kontrat ilişki şeması

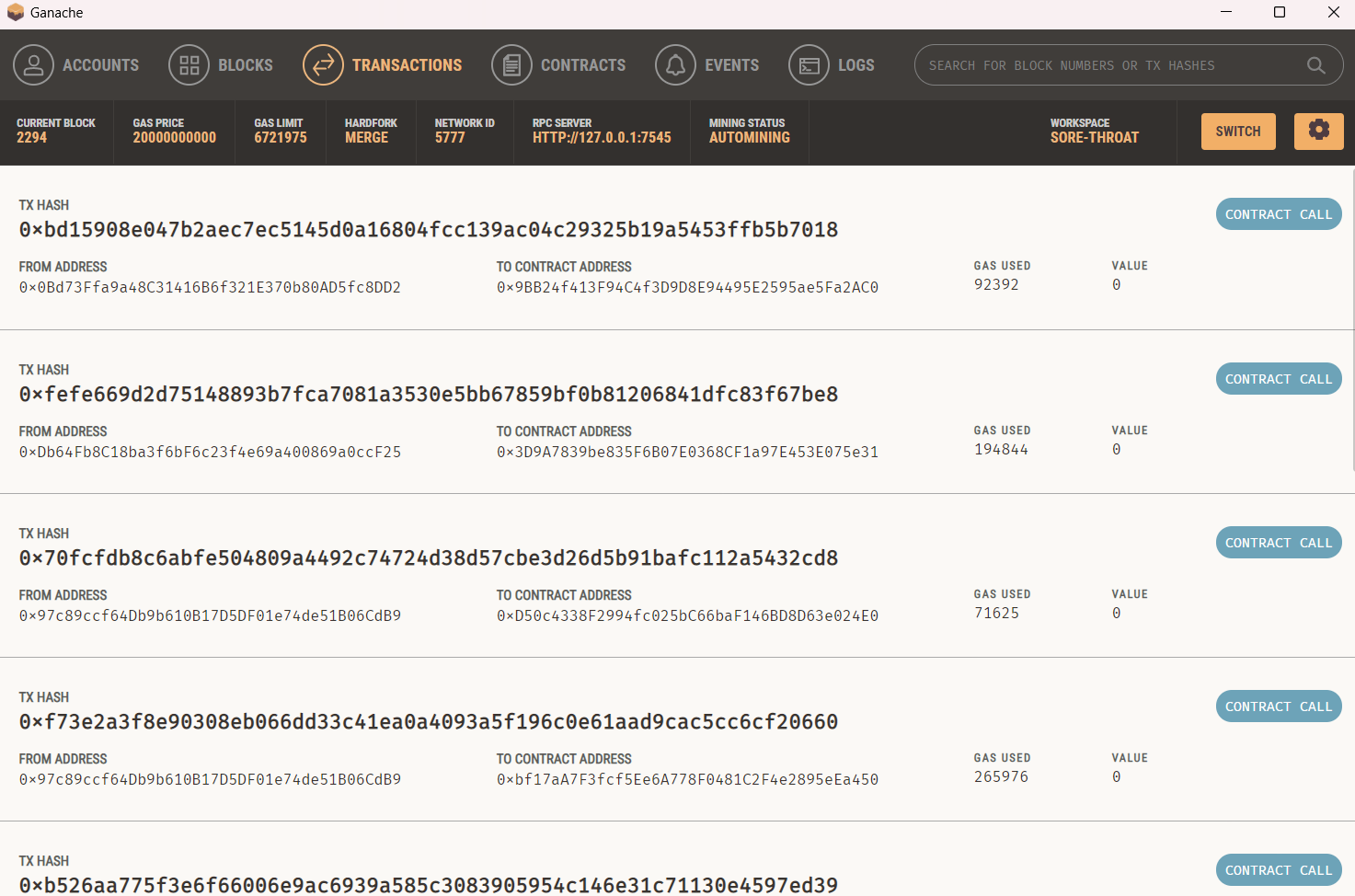
Şekil 3’te gösterilen akıllı kontrat ilişkileri, kontratların birbirleri ile veri alışverişini temsil eder. Akıllı sözleşmeler blockchainde ilk dağıtılırken bu ilişkili diğer sözleşmelerin ağdaki adresleri parametre olarak girilerek dağıtılır. Böylece akıllı sözleşmeler birbirleriyle iletişim kurabilir. BYS uygulaması için geliştirilen akıllı sözleşmeler:

1. SenderChecker: Tüm akıllı kontratlarda ekleme, güncelleme ve silme gibi işlemleri gerçekleştirebilecek cüzdan adreslerinin tanımlandığı kontrattır. Diğer tüm kontratlarda bu adreslerin kontrolü, SenderChecker kontratındaki “isAllowedSender” metodu ile gerçekleştirilir.
2. UserContract: Kullanıcıların bilgileri ve rolleri, öğrencilerin danışman öğretmenleri, kullanıcıların kayıtlı oldukları departmanların kaydedildiği yapılar(struct); bunlar üzerinde “Create, Read, Update, Delete” (CRUD) işlemleri yapan fonksiyonları içermektedir.
3. FacultyContract: Fakülte bilgilerini içeren kontrattır. CRUD işlemlerini içerir.
4. DepartmentContract: Departman bilgilerinin ve departmana özgü tanımlanacak cüzdan adresi özelliğinin de kaydedildiği struct yapısını içeren kontrattır. Departmanın eğitim derecesi (ön lisans, yüksek lisans vb.) “degreeLevel” özelliğinde yer alır. Bu departmanlar ile ilgili CRUD metotlarını barındırır.
5. CourseContract: Uygulamada depratmanda yeni bir ders oluşturulacağı zaman temel alınan; ders adı, kapasitesi ve kodu gibi bilgileri içeren ders objesi bu kontratta yer alır. CRUD işlemleri yapan fonksiyonları içermektedir.
6. OpenCourseContract: Bu kontrat, abstract ders yapısını baz alarak oluşturulan yeni derslerin yönetildiği kontrattır. CourseContract’taki yapıyı kullanarak eklenen derslerin: Ders zamanı, yılı, dönemi, öğretmenleri, sınavları, harf notu kuralları, dersin yapılacağı konum bilgisi ve dersin aktiflik durumu bilgileri yer alır. CRUD işlemlerini gerçekleştiren metotları barındırır.
7. DepartmentOpenCourseContract: Departmana özel açılan derslerin fonksiyonlarını içerir.
8. StudentCourseInfoContract: Öğrencilerin, departmanda oluşturulan ve aktif edilen derslerdeki kayıtlarının kaydedildiği kontrattır. Öğrencilerin devamsızlık bilgileri; öğrencinin dersteki kayıtlanma durumu (danışman incelemesi, kesin kayıt vb.); vize, final ve bütünleme notlarını içerir. Bu kontrat; ders seçme döneminde dersin kayıtlanabilirlik durumunu belirten, öğrencilerin notlarının girilmesi, kaydolduğu derslerin listesinin döndürülmesi ve akademik bilgilerinin döndürülmesi gibi işlevleri ve diğer CRUD metotlarını barındırır.
9. StudentContract: Öğrencilerin kaydoldukları dersler, devamsızlık bilgileri, ders notları ve kayıtlı oldukları departmanlar gibi bilgileri içerir.
10. TeacherContract: Öğretmenlerin eğitim verdikleri dersler, danışmanı oldukları öğrencilerin listesi, kayıtlı oldukları departmanlar gibi bilgileri içerir.
11. ExamContract: Sınav dersinin bilgileri, sınav tipi, sınavın zamanı ve yeri bilgilerini içerir. Departmana ait olan sınavların, öğrencinin aldığı derslerin sınavlarının ve öğretmenin verdiği derslerin döndürüldüğü ve diğer CRUD işlemlerini de barındırır.
12. PostContract: Derse ait, öğrencilerin ve öğretmenlerin yazı olarak paylaşım yapmalarına olanak sağlar. Bu kontrat gönderilerin içeriğini, hangi derse ait olduklarını ve paylaşım zamanı gibi bilgileri içerir.

## 2.2 Ganache Yerel Ethereum Ağı

Ganache, yazılım geliştiricilerin dApp merkezi olmayan projelerini henüz geliştirme veya test aşamalarındayken genel Ethereum ağına dağıtım yapmadan, kontrollü bir ortamda test etmesine ve geliştirmesine olanak tanıyan açık kaynaklı, yerel bir Ethereum blockchain simülatörüdür. Kendi hesapları, cüzdanları, kontratları ve işlemleriyle simüle edilmiş bir blok zinciri sağlar ve hızlı prototip oluşturma, test ve debug etmeye olanak tanır.

Ganache servisi, yeni bir proje oluşturulup, Uygulama Programlama Arayüzü (API) bağlantı adresini ve portunu düzenledikten sonra kullanıma başlanabilir. Cüzdan sayısı, cüzdanlardaki Ethereum miktarı gibi özellikler, Şekil 4’teki arayüzde ayarlar kısmından düzenlenebilir.



**Şekil 4:** Ganache arayüzü

## 2.3 ExpressJS

Express.js, güçlü özelliklere ve ölçeklenebilirliğe sahip web uygulamaları oluşturmak için kullanılan hafif, açık kaynaklı bir JavaScript çerçevesidir. İstekleri ve yanıtları verimli bir şekilde işleyebilen http tabanlı API’ler (Uygulama Programlama Arayüzleri) oluşturmak için esnek bir temel sağlar. Node.js tabanlı web uygulamaları oluştururken sıklıkla tercih edilmektedir. http isteklerini işlemek için API’leri oluşturmak, modüler tasarımı ve üçüncü taraf kitaplıklardan oluşan kapsamlı ekosistemi ile kolayca sunucu hazırlamak gibi avantajları vardır.

Express.js, geliştiricilerin http yöntemleri (ör. GET, POST, PUT, DELETE) için özel yollar ve işleyiciler tanımlamasına olanak tanıyan güçlü bir yönlendirme sistemi sağlar. Express.js hafif olacak şekilde tasarlanmıştır; bu da onu küçük ölçekli uygulamalara veya minimum düzeyde ek yük gerektiren uygulamalara uygun hale getirir.

## 2.3.1 ExpressJS ABI Bağlantısı

Uygulama İkili Arayüzü (ABI), Solidity akıllı sözleşmesinin arayüz tanımını içeren bir JSON dosyasıdır. Sözleşmenin işlevlerinin, olaylarının ve değişkenlerinin insan tarafından okunabilir bir temsilini sağlayarak geliştiricilerin sözleşmeyle programlı olarak etkileşimde bulunmasına olanak tanır.

Bir ABI, akıllı sözleşmenin sunduğu işlevleri, olayları ve değişkenleri tanımlayarak geliştiricilerin sözleşmeyle programlı olarak etkileşimde bulunmasını mümkün kılar [24].

## 2.3.2 AuthorizeRole ve Bearer Token

AuthorizeRole, kullanıcıların BYS içerisindeki belirli kaynaklara ve işlevlere erişimini kontrol eden bir yetkilendirme mekanizmasıdır. Üniversite ortamında farklı kullanıcı grupları (roller) vardır ve her grubun belirli izinleri ve sorumlulukları bulunur. Örneğin:

1. **Yönetici**:
   * Tüm sistem ayarlarına erişebilir.
   * Kullanıcı hesaplarını yönetebilir.
   * Fakülte, bölüm ve ders bilgilerini güncelleyebilir.
2. **Öğretim Görevlisi**:
   * Öğrenci notlarını sisteme girebilir ve güncelleyebilir.
   * Devamsızlık kaydı girebilir ve güncelleyebilir.
   * Danışman öğretmen ise, danışmanı olduğu öğrencilerin ders kayıtlarını düzenleyebilir ve akademik bilgilerini görüntüleyebilir.
   * Verdiği derslerin öğrencilerini, haftalık ders programını ve sınav programını görüntüleyebilir.
3. **Öğrenci**:
   * Ders seçimi yapar.
   * Kendi notlarını ve ders materyallerini görüntüleyebilir.
   * Aldığı derslerin sınav sonuçlarını, haftalık ders programını, sınav programını görüntüleyebilir.

metin, diyagram, çizgi, ekran görüntüsü içeren bir resim

Açıklama otomatik olarak oluşturuldu

**Şekil 5:** Rol yetkinlik diyagramı

BYS'de, her kullanıcının sadece kendi rolüne uygun işlevlere erişimini sağlamak için AuthorizeRole kullanılır. BYS'nin farklı bölümlerine sadece yetkili kullanıcıların erişimini sağlayarak, veri güvenliğini ve sistem bütünlüğü korunmuş olunur [25].

Bearer token (JWT: Javascript Web Token), kimlik doğrulama ve yetkilendirme için kullanılan basit bir kimlik doğrulama yöntemidir. Kullanıcılara kullanıcı adı ve parola gibi bazı kimlik bilgilerinin yerine erişimi sağlayan bir karakter dizisi sunulur. Bu token, yetkili sunucu tarafından oluşturulur ve genellikle bir süre geçene kadar geçerlidir. Bearer token’in kullanımı ve uygulanması basittir, oturum durumu bilgisi yoktur. Bu, ölçeklenebilir ve dağıtılmış sistemlerde kullanımı kolaylaştırır. Ayrıca farklı kimlik doğrulama ve yetkilendirme senaryolarında da kullanılabilir. Bir Bearer token, bir API'ye erişmek için kullanılabilir. Sunucu, token'ı doğrular ve istemciye API'yi kullanma izni verir. Tek sayfa uygulamalarında (SPAs: Single Page Applications) da kimlik doğrulama ve yetkilendirme için bir Bearer token kullanabilir.

Kullanıcı, giriş kimlik bilgilerini (kullanıcı adı ve parola) girerek BYS sistemine giriş yapar. Sistem bu kimlik bilgilerini doğrular ve doğruysa, o kullanıcıya bir bearer token verir. Kullanıcı, sistem içinde bir şey yapmak istediğinde (örneğin, bir kursa kaydolmak veya notları görüntülemek), bu isteğe(request’e) bearer token'ı ekler. İstek başlığı şu şekilde olacaktır: Authorization: Bearer <token>. Sistem, istekle birlikte gelen bearer token'ı kontrol eder. Eğer bu token geçerli, süresi dolmamış ve kullanıcıda gereken tüm yetkilendirmeler varsa, istek işlenir. Değilse, sistem herhangi bir erişime izin vermez; kullanıcıyı yeniden giriş yapması için yönlendirir.

Bearer token'lar genellikle kısa süreler için kullanılır. Süresi dolunca, kullanıcı yeniden giriş yapmalı ve yeni bir token almalıdır. Bu, güvenliği artırmak ve yetkisiz erişimi önlemek açısından önemlidir [25-27]. BYS uygulaması için ise 1 saatlik bearer token kullanıldı.

## 2.3.3 Cüzdan sistemi

Uygulama öğrenci bilgi sistemi olup bu sistem merkezi olmayan bir uygulama olarak tasarlanmıştır. Bu yaklaşımın, güvenlik ve öğelerin dağıtımı açısından en iyi şekilde görülen birçok faydası vardır. Ancak bu sistemde blockchain üzerinde akıllı sözleşmelerle yapılan tüm işlemler, dağıtık sistem açısından güvensiz ve mantıksız kabul edilen bir cüzdan tarafından başlatılıyordu. Böylece, arka uç için tek bir cüzdan yerine, daha fazla cüzdan örneğine sahip olunarak Ek 1 Şekil 16’daki gibi merkezi olmayan bir yapı oluşturulması amaçlandı. Öğrenci bilgi sistemlerinde kullanıcı sayısının farklılık gösterebileceği ve her üniversiteye kabul edilen öğrenci sayısına göre her kullanıcıya cüzdan verilmesinin fazla olacağı varsayılmaktadır. Bu durumda kullanıcılar için cüzdan oluşturmaktan farklı olarak departmanlar için cüzdan oluşturuldu. Yeni oluşturulan departmanlar için, önceden oluşturulup kaydedilmiş belirli cüzdan adresleri kullanıldı. Tanımlanan bu adres bilgisi daha sonra akıllı sözleşmelere kaydedildi. Bu konum, departmanlar arasındaki işlemlerin daha resmi ve güvenli bir şekilde yapılması anlamına geliyor. Örneğin bir departmana ders ekleme, not yazma, sınav tarihi oluşturma gibi işlemlerde o departmanın cüzdan adresi alınır ve alınan işlem ücretleri o bölümün cüzdanından ödenir. Bu durum sistemin diğer sistemlere göre daha verimli olmasını sağlamakta ve öğrenci bilgi sistemi üzerindeki trafik yükü zaman diliminde cüzdanların yoğun kullanımından dolayı oluşabilecek yoğunluğu bölüştürmektedir. İşlemler farklı cüzdan adreslerinden gerçekleştiği için sistem daha merkeziyetsiz ve verimli çalışmaktadır. Bu kullanım aynı zamanda akıllı sözleşmelerde ek bir güvenlik katmanı görevi de görmektedir. Verileri değiştirme olanağı yalnızca belirli cüzdanlarda mevcuttur, dolayısıyla akıllı sözleşmenin kullanılabilirliği çeşitli harici manipülasyonlara karşı korunur. Böylece öğrenci bilgi sistemi işleyişinde daha güvenilir ve verimli hale gelir.

## 2.4 React

Web geliştiriciliği günümüzde hızla gelişmektedir. Bu gelişmede en önemli faktörlerden biri ön uç çerçevelerin varlığıdır. Ön uç çerçeveler veya JavaScript çerçeveleri, geliştiricilerin kullanıcı erişilebilirliğini geliştirmesine yardımcı olan, etkileşimli, kullanıcı merkezli web sitelerinin geliştirilmesini kolaylaştıran, klasör ve dosyalarda önceden yazılmış, homojen kodlardan oluşan paketler olarak kabul edilir.

## 2.4.1 React Express bağlantısı

BYS sistemi temelde bir web projesi olduğundan bu projey kaba taslak ön uç ve arka uç olarak ayrılabilir. Bu projede ön uç olarak bir JS çerçevesi olarak React, arka uç olarak da yine bir JS çerçevesi olarak ExpressJS kullanılmıştır. Bu iki JS çerçevesinin veri alışverişinin, isteklerin yönetiminin geliştirici açısından daha kolay gerçekleşmesini sağlamaktadır.

Bu kombinasyon, Node.js'nin hızı ve React'in sanal DOM'u ile performansı artırırken, React'in bileşen tabanlı mimarisi, yeniden kullanılabilirliği ve modülerliğin gücünü kullanır. Geliştirme deneyimi, React ve Express.js'in ara yazılım(middleware) sistemindeki sıcak yeniden yükleme (hot reload) ile daha da kolaylaştırılır ve bu ikiliyi gerçek zamanlı platformlarda, çeşitli web uygulamaları oluşturmak için mantıklı bir seçimdir.

## 2.4.2 RTK Query

RTK Sorgusu, modern React uygulamalarında verileri yönetmenin daha basit ve daha hızlı bir çözümünü sağlayan Redux Toolkit kullanılarak geliştirilen bir kütüphanedir [28]. Redux Toolkit, Redux kullanımıyla ilişkili kalıpları azaltmaya odaklanan Redux tabanlı uygulamalar oluşturmak için bir araç setidir. RTK Sorgusu, veri alma veya API istemci kütüphanelerinde genellikle elle yazılması gereken temel işlemlerin bazılarını otomatikleştirir. Geliştiricilerin iş yükünü azaltarak enerjilerini temel mantığa ve arayüzlere odaklamalarını sağlar.

RTK Sorgusu, Veri almayla ilgili tüm işlemleri soyutlayarak kapsamlı kodlama ve karmaşıklık ihtiyacını azaltır. Verileri otomatik olarak önbelleğe alarak uygulama optimizasyonunu etkinleştirir. Tür denetimi yoluyla hataları önlemek için TypeScript ile entegre edilmiştir. API çağrısı bir sonuç döndürmeden önce bile kullanıcı arayüzünü güncelleyerek daha hızlı ve daha sorunsuz kullanıcı arayüzü oluşturma sağlar. API uç noktaları için gerekli sorgu parametrelerini otomatik olarak oluşturur. API yanıtlarını kolayca kullanılabilir hale getirmek için önceden işler.

## 2.4.3 Tailwind

Tailwind CSS, geliştiricilerin yardımcı program odaklı bir yaklaşım sunarak daha verimli ve sürdürülebilir kod yazmalarına olanak tanıyan popüler, açık kaynaklı bir CSS çerçevesidir. 2017 yılında Adam Wathan ve Steve Schoger tarafından oluşturulmuştur. Bootstrap gibi geleneksel CSS çerçevelerinde, bazı temel stil seçenekleriyle birlikte önceden tasarlanmış bir dizi kullanıcı arayüzü bileşeni (ör. butonlar, formlar) verilir. “tailwind.config.js” dosyasında kişisel tasarım tercihleri (ör. renkler, tipografi, aralık) tanımlanabilir; bu da çerçevenin davranışının özelleştirilmesine ve onu belirli proje ihtiyaçlarınıza göre uyarlamaya olanak tanır.

Tailwind CSS, HTML öğelerini biçimlendirmek için çok sayıda yardımcı sınıf barındırır, örneğin: “Boşluklar: margin, padding, width, height”, “Renkler: background colors, text colors, borders” vb.

Tailwind ayrıca ekran boyutuna duyarlılık (responsivity) yaklaşımını da benimseyerek geliştiricilere kolaylık sağlar. Geliştiriciler örneğin “sm:p-0” gibi ifadelerle, “ekran boyutu küçük olduğunda dolgulama özelliğini 0 olarak güncelle” gibi duyarlılık kodları yazabilirler.

Tailwind, React, Vue.js ve Angular gibi popüler JavaScript çerçeveleriyle kullanılabilir.

## 2.5 Öğrenci Bilgi Sistemi

Bu bölümde, geliştirilen Bilgi Yönetim Sistemi'nin (BYS) projeye özgü mimarisi ve çeşitli işlevleri detaylı bir şekilde ele alınmaktadır. Sistemin yapısı, entegre edilen teknolojiler ve sağladığı çözümler hakkında ayrıntılı bilgiler verilerek, nasıl bir yenilik ortaya koyduğu açıklanmaktadır.

Şekil 5’te verilen proje mimarisine göre akış şu şekildedir: Ganache Yerel Ethereum Ağında dağıtılmış olan 12 akıllı sözleşme bir bakıma API olmakla birlikte aynı zamanda veritabanı görevi görmektedir. Ganache kullanılarak yerel bir Ethereum ağı oluşturulur ve Solidity ile yazılan akıllı sözleşmeler burada test edilip çalıştırılır. Bu akıllı sözleşmeler, verilerin güvenli ve merkeziyetsiz bir şekilde saklanmasını sağlar. Derlenen akıllı sözleşmelerden oluşturulan ABI’ler kullanılarak, Express.js kullanılarak bir API sunucusu oluşturulur. Bu sunucu, istemciden gelen istekleri alır, ilgili akıllı sözleşmelerle etkileşime geçer ve sonuçları döner. Veri alışverişi yalnızca tanımlı cüzdan adresleri üzerinden yapılabilir. ExpessJS’de işlenen veriler React aracılığıyla da kullanıcılara dinamik ve etkileşimli bir arayüz ile sunulur. React, kullanıcıların blockchain verilerine erişmesini ve işlem yapmasını sağlar, aynı zamanda RTK Query ile Express.js sunucusuna istekler gönderir ve yanıtları alır.

metin, ekran görüntüsü, diyagram, tasarım içeren bir resim

Açıklama otomatik olarak oluşturuldu

**Şekil 6:** Proje mimarisi

## 2.5.1 Projenin İşlevleri

Proje tasarlanırken, güncel BYS’lerde kullanıcıların en çok ihtiyaç duydukları işlevlerden mümkün olduğunca çoğunluğu seçilerek uyarlandı. Roller; üye kaydı ve girişi; ders oluşturma, aktif etme, ders seçimi, danışman öğretmenlerin ders seçimlerini inceleyip onaylaması; not girişi ve düzenleme; transkript oluşturma; devamsızlık işleme; alınan derslere göre otomatik ders programı oluşturma; sınıf paylaşım alanı gibi işlevler uygulamaya eklenmiştir.

## 2.5.1.1 Üye Kaydı ve Roller

BYS sistemine giriş yapmak istendiğinde, “Giriş yap” ve “Kaydol” butonlarından, kullanıcı zaten kayıtlı ise giriş yapar, kayıtlı değil ise yeni üyelik oluşturur. Kaydolmak için adı, soyadı, e-mail adesi, telefon numarasını ve şifresini girer ve onaylar. Bu işlemden sonra, Şekil 6’da görüleceği üzere yeni kullanıcı, yöneticinin “Pasif Kullanıcılar” sayfasında yeni kaydolmuş kullanıcılara eklenir. Bu adımda yöneticiler, daha önce de bahsedildiği üzere “Yönetici”, “Öğrenci” veya “Öğretim Üyesi” rollerinden birini seçerek yeni kullanıcının kaydını tamamlarlar. Bu sayede, kullanıcıların sisteme kaydolduktan sonra yetkisiz işlem yapmaları engellenmiş ve kontrollü bir kayıt süreci gerçekleşmiş olur.

metin, yazılım, yazı tipi, ekran görüntüsü içeren bir resim

Açıklama otomatik olarak oluşturuldu

**Şekil 7:** Yönetici, yeni kaydolmuş üyelerin listesini görüntüler

Öğrenci ve öğretim üyesi kayıtlanırken, yönetici tarafından Ek 1 Şekil 17’de görüldüğü gibi rol, fakülte ve departman seçimleri atanır. Böylece hangi öğrencinin veya eğitim görevlisinin hangi departmanda görev aldığı tanımlanır.

## 2.5.1.3 Ders Seçimi ve Derse Kayıtlanma

CourseContract’a yazılan abstract ders kayıtlarını kullanarak, yeni dersler açılır. Bunun için Şekil 7’de görülen “Bölümler/Dersler” kısmından, abstract dersleri seçerek, aktif dersler “Dersi aç” butonu ile oluşturulur. Aynı dersten birden fazla açılabilir, bu sayede eğer ki bir dersi birden fazla eğitim görevlisi veriyorsa, çakışmalar olmadan kayıtlar yönetilebilir.

metin, ekran görüntüsü, yazılım, sayı, numara içeren bir resim

Açıklama otomatik olarak oluşturuldu

**Şekil 8:** Departman sayfası: Ders listesinden ders açma

Yeni ders kaydı açıldıktan sonra, ilgili eğitim görevlisi “Açılan Dersler” sayfasından Şekil 8’deki arayüzden düzenlenmelidir.

metin, ekran görüntüsü, sayı, numara, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu

**Şekil 9:** Açılan dersin düzenlenmesi

Şekil 8’de görüleceği üzere, dersin yapıldığı sınıf, dersin aktiflik durumu, öğretim görevlisi ve harf notu aralıkları bu arayüzden güncellenebilir. Öğrenciler ise, yönetici ders seçimi dönemini başlattıktan sonra, Şekil 9’daki arayüzden ders seçimlerini yaparlar. Bunun için “Ders İşlemleri/Ders Seçimi” sayfasına giriş yaparlar. Açılan derslerden seçim yaparlar ve danışman onayına gönderirler.

metin, ekran görüntüsü, yazılım, sayı, numara içeren bir resim

Açıklama otomatik olarak oluşturuldu

**Şekil 10:** Öğrenci ders seçim arayüzü

Öğrenci ders seçimi yapıp danışman onayına gönderdikten sonra, danışman öğretim görevlisinin “Ders İşlemleri/Ders Seçimlerini İncele” sayfasında, ilgili öğrencinin öğrencinin ders seçimi bilgileri güncellenir. Danışman öğretim görevlisi bu bilgilere erişerek, öğrencinin seçtiği derslerin uygunluk durumuna göre dersleri onaylayabilir veya silebilir.

metin, sayı, numara, yazı tipi, çizgi içeren bir resim

Açıklama otomatik olarak oluşturuldu

**Şekil 11:** Danışman öğretmen ders seçimi incelemesi

## 2.5.1.4 Transkript

Transkript, öğrencilerin kayıtlandıkları zamandan beri aldıkları dersler ve bu derslerdeki başarı durumlarını özetleyen bir dökümdür. Bu belge öğrencinin ders kayıtlarına göre otomatik olarak oluşturulur. Tasarlanan BYS sisteminde Ek 1 Şekil 18 örneğindeki gibi transkriptleri öğrenciler ve öğretim görevlileri görüntüleyebilirler.

## 2.5.2.6 Notlandırma

Şekil 11’deki gibi her bir dersin notlandırması için ayrı ayrı harf aralıkları düzenlenebilir. Bu seçim öğretim görevlileri ve yöneticilere bırakılmıştır.

Öğrencinin devam durumu “Devamlı” veya “Devamsız” olarak seçilebilir. “Devamsız” seçilirse, daha sonra girilen vize, final veya büt notlarının bir etkisi olmadan “DZ” harf notu verilmiş olur. “Devamlı” seçilirse, vize %50 ve final %50 hesaplanarak, daha önce belirtilen harf notu aralığına göre notlandırma yapılır. Bütünleme notu girilmişse, final notu dikkate alınmaz.

metin, ekran görüntüsü, sayı, numara, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu

**Şekil 12:** Öğretim görevlisi verdiği derslerin notlandırmasını yapar

## 2.5.2.7 Ders Programı

Ders programı, öğrencileri seçtikleri derslerin, ders vakitlerine göre otomatik olarak Şekil 12’deki gibi oluşturulur. Ders programında çakışma olmaması, ders seçiminde onaylama yapan öğretim görevlilerinin ve yöneticilerin sorumluluğundadır. Ders programı tablosu Pazartesi’den Pazar gününe kadar ve saat aralıkları da bir ders saati yani 50 dakika olacak şekilde düzenlendi. Öğrenciler, eğer birden fazla departmanda eğitim görüyorlarsa dahi bölüm seçeneğinden ilgili departmanı seçerek, güncel haftalık ders programlarına erişirler.

metin, ekran görüntüsü, sayı, numara, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu

**Şekil 13:** Ders programı

## 2.5.2.8 Sınıf Paylaşım Planı

Öğrenciler ve öğretim görevlilerinin, ders hakkında fikirlerini paylaşabilecekleri bir paylaşım alanı oluşturuldu. Şekil 13’te görüldüğü gibi ana sayfadaki ders kartlarıyla etkileşime girerek, Şekil 14’teki ders içi paylaşım alanına ulaşılır. Söz konusu derse kayıtlı tüm kullanıcılar yazı paylaşabilirler.

metin, ekran görüntüsü, yazı tipi, sayı, numara içeren bir resim

Açıklama otomatik olarak oluşturuldu

**Şekil 14:** Paylaşım alanı için ders seçimi

Şekil 14’teki arayüzde, dersi veren öğretmen ve öğrencileri, yazı formatında tüm sınıf üyelerinin göreceği paylaşımları oluşturabilirler. Bu paylaşım alanlar her bir ders için özeldir.

metin, ekran görüntüsü, yazı tipi, sayı, numara içeren bir resim

Açıklama otomatik olarak oluşturuldu

**Şekil 15:** Paylaşım alanı

# Bölüm 3- Bulgular ve Tartışmalar

Bu bölümde, Bölüm 2’de anlatılan blockchain tabanlı BYS test edildi ve değerlendirildi.  Blockchain tabanlı Öğrenci Bilgi Sistemi'nin (BYS) güvenliği ve şeffaflığı kapsamlı testlerden geçirildi. Bu testler, sistemin veri bütünlüğünü koruyabilme yeteneğini, yetkisiz erişimlere karşı direncini ve tüm işlemlerin şeffaf bir şekilde kaydedilip doğrulanabilmesini değerlendirmeyi amaçladı.

## 3.1 Güvenlik

Blok zincirinin amacı doğrultusunda geliştirilen şeffaf ve değiştirilemez (immutable) doğası gereği, BYS üzerindeki verilerin yetkilendirme dışında değiştirilmesi veya silinmesi teorik olarak mümkün değildir. Bu durum, öğrenci kayıtları, notlar, devamsızlık bilgileri ve diğer kritik verilerin güvenliğini sağlar. Bu uygulamada yapılmamasına karşın öğrenci diplomaları ve sertifikaları için de evrakta sahtecilik yapılmasının önüne geçilebilir [29].

BYS, kullanıcıların kontrollü üye kaydı ve rol ataması yaparak yetkisiz erişimleri engeller. Sistemin yetkilendirme mekanizması, her kullanıcının sadece kendi rolüne uygun verilere erişebilmesini sağlar. Yapılan testlerde, farklı kullanıcı rolleriyle (öğrenci, öğretmen, yönetici) sisteme yetkisiz erişim denemeleri yapılmış ve sistemin bu denemeleri başarıyla engellediği görülmüştür.

Blockchain'in şeffaflık özelliği sayesinde, BYS üzerinde gerçekleştirilen tüm işlemler herkese açık bir şekilde kaydedilir ve doğrulanabilir. Bu durum, sistemin hesap verebilirliğini artırır ve veri manipülasyonu riskini azaltır. Testler sırasında, sistemdeki tüm işlemlerin blok zincirinde doğru bir şekilde kaydedildiği ve bu kayıtların herkes tarafından erişilebilir olduğu doğrulanmıştır.

## 3.2 Performans ve Ölçeklenebilirlik

BYS'nin performansı ve ölçeklenebilirliği, sistemin gerçek dünya koşullarında ne kadar iyi çalışabileceğini belirlemek için önemlidir. Bu nedenle, farklı kullanıcı yükleri ve veri hacimleri altında sistemin işlem hızı, gecikme süresi ve kaynak kullanımı analiz edildi. Sistem, düşük ve orta boyutlu kullanıcı yükleri altında hızlı ve verimli bir şekilde çalıştı. Ancak, yüksek kullanıcı yüklerinde işlem hızında bir miktar düşüş gözlendi. Örneğin arka uçtaki RefactoreController metodunda, sistem başlatılırken testleri gerçekleştirebilmek adına örnek kayıtlar eklendi.

Bu kayıt sayısı 20’den 50’ye arttırılınca işlem (transaction) süresinde çok net bir gecikme olduğu gözlemlendi. Bu durum, blockchain teknolojisinin doğasında olan işlem onaylama süreçlerinden kaynaklanmaktadır. İşlem gecikme süresi, özellikle yüksek kullanıcı yüklerinde artış gösterdi. Bu durum, blockchain ağındaki işlem onaylama mekanizmalarından kaynaklanmaktadır.

BYS'nin ölçeklenebilirliği, sistemin büyüyen veri miktarı ve artan kullanıcı sayısını yönetebilme becerisini ifade eder. Yapılan testlerde, sistemin mevcut haliyle belirli bir ölçeklenebilirlik sınırına sahip olduğu görüldü. Ancak, farklı blockchain ölçeklendirme çözümlerinin (sharding, sidechain'ler vb.) entegre edilmesiyle bu sınır aşılabilir [8].

## 3.3 Merkezi ve Dağıtık Sistemlerde DDoS Saldırıları ve Gas Maliyetleri

Dağıtık Olmayan Hizmet Engelleme (DDoS) saldırıları, bir sistemi veya ağı, trafiği aşırı yükleyerek hizmet veremez hale getirmeyi amaçlayan kötü niyetli saldırılardır. Geleneksel merkezi sistemlerde, bu tür saldırılar genellikle tek bir noktaya odaklanır ve başarılı olursa tüm sistemi çökertebilir. Ancak, blockchain tabanlı dağıtık sistemlerde durum farklıdır.

Merkezi Sistemler, genellikle daha basit bir yapıya sahiptirler ve yönetimi daha kolaydır. Tek bir hata noktası vardır, bu nedenle DDoS saldırılarına karşı savunmasızdırlar. Saldırı başarılı olursa tüm sistem etkilenir.

Dağıtık Sistemler (Blockchain), merkezi olmayan yapıları sayesinde tek bir hata noktası yoktur. DDoS saldırıları tek bir düğümü etkileyebilir, ancak tüm sistemi devre dışı bırakması zordur. Bunlara karşın işlem maliyetleri (gas ücretleri) yüksek olabilir. DDoS saldırıları sırasında, saldırganlar çok sayıda gereksiz işlem başlatarak gas ücretlerini artırabilir. Bu durum, sistemin normal kullanıcıları için maddi bir yük oluşturabilir.

Örnek olarak bu çalışmada geliştirilen BYS uygulaması DDoS saldırısına uğradığında, saldırganlar çok sayıda sahte işlem yapmaya çalışarak sistemi yavaşlatabilir veya durdurabilir. Yapılan uygulamada işlem gerçekleştirebilmek için SenderChecker akıllı kontratında tanımlanmış cüzdanların kullanılması gerekmektedir fakat bunun kontrolü için de bir işlem maliyeti vardır. Sistemin dağıtık doğası gereği, saldırının tüm sistemi etkilemesi zordur. Yine de bu tür bir saldırı, Ethereum gibi bir blockchain platformunda gas ücretlerinde artışa neden olabilir. Gas ücretleri, işlemlerin onaylanması için ödenen ücretlerdir ve ağdaki yoğunluğa bağlı olarak değişkenlik gösterebilir. Bir DDoS saldırısı sırasında, ağdaki yoğunluk arttığı için gas ücretleri de önemli ölçüde yükselebilir. Uygulamada cüzdanların ve bakiyelerin sistem sahipleri tarafından karşılandığı düşünülürse, bu durum maddi bir yük anlamına gelebilir.

Blockchain tabanlı BYS'nin merkezi olmayan yapısı, DDoS saldırılarına karşı daha dirençli olmasını sağlar. Ancak, bu tür saldırılar gas ücretlerinde artışa neden olarak kullanıcılar için maddi bir yük oluşturabilir. Bu nedenle, BYS geliştiricilerinin, gas ücretlerini optimize etmek ve kullanıcıları bu tür maliyetlerden korumak için uygun önlemler alması önemlidir.

# Bölüm 4- Sonuçlar

Bu bitirme projesi kapsamında geliştirilen Blockchain tabanlı Öğrenci Bilgi Sistemi (BYS), güvenli, şeffaf ve merkezi olmayan bir veri yönetimi için umut verici bir çözüm ve fonksiyonel bir örnek uygulama sunmaktadır. Geleneksel öğrenci bilgi sistemlerinin karşılaştığı güvenlik, şeffaflık ve veri manipülasyonu gibi sorunlara potansiyel çözümler getiren BYS, eğitim kurumları için önemli faydalar sağlayabilir.

Blok zincirindeki tüm işlemler herkese açık ve doğrulanabilir olduğu için, sistemdeki her türlü değişiklik izlenebilir ve denetlenebilir. Bu durum hem eğitim kurumları hem de öğrenciler için güvenilir bir ortam oluşturur. Öğrenciler, kendi verilerinin nasıl kullanıldığını ve paylaşıldığını takip edebilirler. BYS, öğrenci kayıtları, notlar, devamsızlık bilgileri, ders programları, transkriptler ve diğer akademik verileri güvenli bir şekilde yönetir. Akıllı sözleşmeler sayesinde, öğrenci kayıt işlemleri otomatik ve güvenli bir şekilde gerçekleştirilir. Not girişleri ve güncellemeleri de akıllı sözleşmeler aracılığıyla gerçekleştiği için, notlarda herhangi bir değişiklik yapılması durumunda bu değişiklikler blok zincirinde kaydedilir ve kolayca takip edilebilir.

BYS, sadece öğrenci verilerini güvenli bir şekilde saklamakla kalmaz, aynı zamanda eğitim süreçlerini de kolaylaştırır. Öğrenciler, ders seçimlerini online olarak yapabilir, notlarını ve devamsızlık bilgilerini görüntüleyebilir, ders paylaşım alanına erişebilir ve diğer öğrencilerle etkileşim kurabilirler. Öğretmenler ise, not girişlerini kolayca yapabilir, öğrenci performansını takip edebilir ve ders hakkında fikirlerini ve duyurularını paylaşabilirler.

## 4.1 İleri Çalışmalar

Sistemin daha yüksek kullanıcı yükleri ve veri hacimleri altında daha iyi performans göstermesi için sharding veya sidechain'ler gibi ölçeklenebilirlik çözümleri entegre edilebilir. Sistemin kullanıcı arayüzü, daha kullanıcı dostu ve sezgisel hale getirilebilir. Farklı kullanıcı rolleri için özelleştirilmiş arayüzler tasarlanabilir. BYS'ye, mevcut paylaşım alanı geliştirilerek öğrenci-öğretmen iletişimi, ödev yönetimi, online sınavlar gibi ek özellikler eklenebilir. Ethereum dışında, Hyperledger Fabric gibi farklı blockchain platformları üzerinde BYS'nin uygulanabilirliği araştırılabilir [9]. Bu çalışma, eğitim alanında blockchain teknolojisinin kullanımına yönelik önemli bir adım olup, gelecekteki araştırmalar için bir temel oluşturabilir.

Sonuç olarak, bu bitirme projesi kapsamında geliştirilen Blockchain tabanlı Öğrenci Bilgi Sistemi (BYS), eğitim alanında blok zinciri teknolojisinin potansiyelini ortaya koymaktadır. Güvenli, şeffaf ve merkeziyetsiz yapısıyla öğrenci verilerinin yönetimi için yeni bir paradigma sunan BYS, gelecekteki araştırmalar ve uygulamalar için önemli bir temel oluşturacaktır.

# KAYNAKLAR

[1] Liang X, Shetty S, Tosh D, Kamhoua C, Kwiat K, Njilla L. ProvChain: A Blockchain-Based Data Provenance Architecture in Cloud Environment with Enhanced Privacy and Availability, in 2017 17th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGRID), Madrid, Spain, May 2017, pp. 468–477. 10.1109/CCGRID.2017.8.

[2] Jha S, Koul S. Application of block chain technology in higher education. 16th AIMS Int’l Conference on Management (AIMS-16), 2019. Accessed: Oct. 29, 2021. [Online]. Available: <http://dspace.jgu.edu.in:8080/jspui/handle/10739/2065>

[3] Lacity MC. Blockchain Foundations: for the internet of value; Imprint of the University of Arkansas Press, 2020.

[4] Tapscott D, Tapscott A. Blockchain revolution: how the technology behind bitcoin is changing money, business, and the world. New York: Portfolio / Penguin; 2016.

[5] C. Johnson, “Credentialism and the proliferation of fake degrees: The employer pretends to need a degree; the employee pretends to have one,” Hofstra Lab. & Emp. LJ, vol. 23, p. 269, 2005.

[6] Han M, Li Z, He J (Selena), Wu D, Xie Y, Baba A. A Novel Blockchain-based Education Records Verification Solution. In: Proceedings of the 19th Annual SIG Conference on Information Technology Education, Fort Lauderdale Florida USA, Sep. 2018, pp. 178–183. 10.1145/3241815.3241870.

[7] Turkanovic M, Holbl M, Kosic K, Hericko M, Kamisalic A. EduCTX: a blockchainbased higher education credit platform. IEEE Access 2018;6:5112–27. doi: <https://doi.org/10.1109/ACCESS.2018.2789929>.

[8] Dong Z, Luo F, Liang G. Blockchain: a secure, decentralized, trusted cyber infrastructure solution for future energy systems. J Mod Power Syst Clean Energy Sep. 2018;6(5):958–67. doi: <https://doi.org/10.1007/s40565-018-> 0418-0.

[9] Seebacher S, Schüritz R. Blockchain Technology as an enabler of service systems: a structured literature review. In: Exploring Services Science, Cham, 2017, pp. 12–23. 10.1007/978-3-319-56925-3\_2.

[10] P. Haveri, R. U. B, N. D. G, N. K, and S. K, "EduBlock: Securing Educational Documents using Blockchain Technology," in 2020 11th International Conference on Computing, Communication and Networking Technologies (ICCCNT), Kharagpur, India, 2020, pp. 1-7. doi: 10.1109/ICCCNT49239.2020.9225258.

**[11]** S. K. Ambast and T. A. Sumesh, "A Blockchain Based Credential Verification System using IPFS," in 2022 IEEE 19th India Council International Conference (INDICON), Kharagpur, India, 2022, pp. 1-5. doi: 10.1109/INDICON56171.2022.10039743.

[12]Z. Changhua and M. Chunqiao, "Research on Personalized Learning Path Management Based on Blockchain Technology," in 2021 16th International Conference on Computer Science & Education (ICCSE), 2021, pp. 339-344. doi: 10.1109/ICCSE51940.2021.9569565.

[13] Farah, J.C.; Vozniuk, A.; Rodríguez-Triana, M.J.; Gillet, D. A Blueprint for a Blockchain-Based Architecture to Power a Distributed Network of Tamper-Evident Learning Trace Repositories. In Proceedings of the 2018 IEEE 18th International Conference on Advanced Learning Technologies (ICALT), Mumbai, India, 9–13 July 2018; pp. 218–222.

[14] Bdiwi, R.; De Runz, C.; Faiz, S.; Cherif, A.A. A Blockchain Based Decentralized Platform for Ubiquitous Learning Environment. In Proceedings of the 2018 IEEE 18th International Conference on Advanced Learning Technologies (ICALT), Mumbai, India, 9–13 July 2018; pp. 90–92

[15] Hölbl, M.; Kamisali´c, A.; Turkanovi´c, M.; Kompara, M.; Podgorelec, B.; Heri´cko, M. EduCTX: An Ecosystem for Managing Digital Micro-Credentials. In Proceedings of the 2018 28th EAEEIE Annual Conference (EAEEIE), Hafnarfjordur, Iceland, 26–28 September 2018; pp. 1–9.

[16] Arenas, R.; Fernandez, P. CredenceLedger: A Permissioned Blockchain for Verifiable Academic Credentials. In Proceedings of the 2018 IEEE International Conference on Engineering, Technology and Innovation (ICE/ITMC), Stuttgart, Germany, 17–20 June 2018; pp. 1–6.

[17] V. Buterin, “Ethereum white paper: a next generation smart contract & decentralized application platform,” [Online]. Available: https://github.com/ethereum/wiki/wiki/White-Paper, 2013

[18] Wu Keco, Peng Bo, Xie Hua, et al. Quantitative evaluation method of decentralization degree of Blockchain network based on entropy theory. Computer and Modernization, 2019.

[19] Jiao Tong, Tong J, Shen Derong, et al. Blockchain database: a queryable and tamper-resistant database. Journal of Software, 2019, Vol. 30Issue (9): 2671-2685.

[20] Sura I. Mohammed Ali a , Haitham Farouk b , Hussien Sharaf c A blockchain-based models for student information systems Egyptian Informatics Journal Volume 23, Issue 2, July 2022, Pages 187-196

[21] Ethereum Project. "Ethereum." ethereum.org, (Erişim tarihi: 02/06/2024)

[22] Remix Project. "Remix." remix.ethereum.org, (Erişim tarihi: 02/06/2024)

[23] Microsoft. "Visual Studio Code." code.visualstudio.com, (Erişim tarihi: 02/06/2024)

[24] X. Sun, X. Lin, and Z. Liao, "An ABI-based Classification Approach for Ethereum Smart Contracts," in 2021 IEEE Intl Conf on Dependable, Autonomic and Secure Computing, Intl Conf on Pervasive Intelligence and Computing, Intl Conf on Cloud and Big Data Computing, Intl Conf on Cyber Science and Technology Congress, 2021, pp. 99-104. doi: 10.1109/DASC-PICom-CBDCom-CyberSciTech52372.2021.00029.

[25] Eradicating Bearer Tokens for Session Management Philippe De Ryck, Lieven Desmet, Frank Piessens, Wouter Joosen iMinds-DistriNet, KU Leuven, 3001 Leuven, Belgium

[26] Hammer-Lahav, E. (2010). "The OAuth 2.0 Authorization Framework." RFC 6749. https://datatracker.ietf.org/doc/html/rfc6749 (Erişim tarihi: 29.05.2024).

[27] Jones, M. B., Hardt, D., & Jones, W. (2012). "OAuth 2.0 Bearer Token Usage." RFC 6750. https://datatracker.ietf.org/doc/html/rfc6750 (Erişim tarihi: 29.05.2024).

[28] RTK Query Resmi Dokümantasyonu: <https://redux-toolkit.js.org/rtk-query/overview>, (Erişim tarihi: 02/06/2024)

[29] G. Sweeney, Transparency International, Global Corruption Report: Education. Abingdon, Oxon: Routledge, 2013.

# EK 1

metin, ekran görüntüsü, diyagram, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu

**Şekil 16:** Dağıtılmış cüzdan sistemi

metin, ekran görüntüsü, sayı, numara, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu

**Şekil 17:** Yöneticilerin kullanıcıya rol ataması

metin, ekran görüntüsü, sayı, numara, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu

**Şekil 18:** Öğrencilerin görüntüledikleri transkript belgesi.