UNIVERSITATEA DE VEST DIN TIMIŞOARA
FACULTATEA DE MATEMATICĂ ŞI INFORMATICĂ
PROGRAMUL DE STUDII DE LICENŢĂ: Informatică

LUCRARE DE LICENŢĂ

COORDONATOR:
Lect. Dr. Cira Cristian

ABSOLVENT:
Negrea Cristian

TIMIŞOARA
2024

UNIVERSITATEA DE VEST DIN TIMIŞOARA
FACULTATEA DE MATEMATICĂ ŞI INFORMATICĂ
PROGRAMUL DE STUDII DE LICENŢĂ: Informatică

URMĂRIEA PRODUSELOR FOLOSIND REŢEAUA ETHEREUM

COORDONATOR:
Lect. Dr. Cira Cristian

ABSOLVENT:
Negrea Cristian

TIMIŞOARA 2024

Abstract

In today's increasingly interconnected and digitalised world, the demand for transparency, efficiency and security in supply chain management has never been greater. Traditional product tracking systems frequently face challenges such as lack of transparency, susceptibility to fraud and logistical inefficiencies. Because it is decentralised, immutable and transparent, blockchain technology offers a viable answer to these problems.

This thesis investigates the use of blockchain technology, primarily Ethereum, to create a reliable and transparent product tracking system. By providing an example of such a solution, this project hopes to demonstrate the potential benefits and practical application of blockchain in real-world supply chain settings.

The proposed Ethereum product tracking application takes advantage of Ethereum smart contracts to generate an immutable and transparent ledger for recording product information. This project demonstrates how blockchain technology can be used to improve transparency, reduce fraud and improve supply chain efficiency.

Implementing this application requires creating smart contracts, front-end interface design, and integration with current systems and off-chain storage.

Thanks to the use of smart contracts, essential information related to each product is kept secure on the Ethereum blockchain. Offering complete transparency, this decentralized method ensures that data remains trustworthy. A comparative analysis of existing product tracking systems, as well as an assessment of the reliability of the proposed solution, are all included in the thesis. The study also includes current blockchain applications for supply chain management.

The hope of the thesis is to highlight the benefits and potential applications of blockchain technology in real-world supply chain scenarios by providing this example of a blockchain-based product tracking system. The project's findings and outlook lay the foundation for future research and development in the field, adding insight into blockchain applications in supply chain management.

Rezumat

În lumea de astăzi, din ce în ce mai interconectată și digitalizată, cererea de transparență, eficiență și securitate în gestionarea lanțului de aprovizionare nu a fost niciodată mai mare. Sistemele tradiționale de urmărire a produselor se confruntă frecvent cu provocări precum lipsa de transparență, susceptibilitatea la fraudă și ineficiența logistică. Deoarece este descentralizată, imuabilă și transparentă, tehnologia blockchain oferă un răspuns viabil la aceste probleme.

Această lucrare investighează utilizarea tehnologiei blockchain, în principal Ethereum, pentru a crea un sistem fiabil și transparent de urmărire a produselor. Prin furnizarea unui exemplu de astfel de soluție, acest proiect speră să demonstreze beneficiile potențiale și aplicarea practică a tehnologiei blockchain în cadrul lanțului de aprovizionare din lumea reală.

Aplicația Ethereum propusă pentru urmărirea produselor profită de contractele inteligente Ethereum pentru a genera un registru imutabil și transparent pentru înregistrarea informațiilor despre produse. Acest proiect demonstrează modul în care tehnologia blockchain poate fi utilizată pentru îmbunătățirea transparenței, reducerea fraudei și îmbunătățirea eficienței lanțului de aprovizionare.

Implementarea acestei aplicații necesită crearea de contracte inteligente, proiectarea interfeței front-end și integrarea cu sistemele actuale și cu stocarea în afara lanțului.

Datorită utilizării contractelor inteligente, informațiile esențiale referitoare la fiecare produs sunt păstrate în siguranță pe blockchain-ul Ethereum. Oferind o transparență completă, această metodă descentralizată asigură faptul că datele rămân de încredere. Teza cuprinde o analiză comparativă a sistemelor existente de urmărire a produselor, precum și o evaluare a fiabilității soluției propuse. Studiul include, de asemenea, aplicații blockchain actuale pentru gestionarea lanțului de aprovizionare.

Speranța lucrarii este de a evidenția beneficiile și aplicațiile potențiale ale tehnologiei blockchain în scenariile lanțului de aprovizionare din lumea reală prin furnizarea acestui exemplu de sistem de urmărire a produselor bazat pe blockchain. Constatările și perspectivele proiectului pun bazele cercetării și dezvoltării viitoare în domeniu, adăugând o perspectivă asupra aplicațiilor blockchain în gestionarea lanțului de aprovizionare.

Cuprins

Capitolul 1

Introducere

În copilărie, am experimentat o serie de situații care m-au făcut să mă îndoiesc de credibilitatea și autenticitatea lucrurilor. Aceste evenimente au scos la iveală problemele răspândite ale fraudei și înșelăciunii de pe piață. Am o poveste deosebit de memorabilă din cei zece ani în care am jucat pentru echipa de fotbal a juniorilor din orașul meu natal. În această perioadă, am cumpărat o pereche de ghete de fotbal de care eram foarte mândru pentru că m-au făcut să mă simt ca un jucător profesionist de fotbal.

Cu toate acestea, bucuria mea a fost trecătoare. Am descoperit rapid că ele nu erau autentice, materialele au fost de calitate proastă și nu au fost așa cum s-a promis. Am fost supărat atât pe site-ul web de pe care le-am cumpărat, cât și pe sistemul mai mare care a permis ca acest lucru să se întâmple.

În primul rând, am crezut că respectiva companie și site-ul sunt responsabile, dar în curând am descoperit mai multe probleme. Problema erau cei care aveau capacitatea de a urmări, modifica sau de a schimba bunurile reale cu cele false în scopul de a câștiga bani. Această înțelegere m-a supărat mai ales pentru că știam că multe persoane apropiate ar putea fi ușor păcălite fără cunoștințe specifice.

Acesta nu a fost un caz izolat. De-a lungul anilor, am întâlnit multe cazuri de fraudă, bunuri false și informații false. Acest lucru a arătat că sunt necesare mecanisme mai bune de urmărire și verificare.

În liceu, am aflat despre tehnologia blockchain prin contextul financiar al criptomonedelor precum Ethereum[1] și Bitcoin[2]. Abordarea acestor monede digitale față de serviciile bancare digitale și potențialul lor ca oportunități de investiții au fost principalele teme de discuție la acea vreme. Cu toate acestea, tehnologia de bază blockchain, a fost aproape complet ignorată de persoanele cu care am interacționat.

[1]Este o platformă și sistem de operare open-source, distribuit, pe bază de blockchain, ce oferă posibilitatea implementării contractelor smart.

[2]Bitcoin este un sistem de plată electronică descentralizat și o monedă digitală (criptomonedă) opensource creată în 2009.

Interesul meu pentru domeniul informatic și expunerea la conceptele tehnologice inovatoare m-au dus spre explorarea tehnologiei blockchain. Fascinația mea pentru această tehnologie provine din capacitatea să de a crea o rețea descentralizată și imutabilă, în care fiecare tranzacție și fiecare informație înregistrată este sigură și transparentă.

Astfel, pasiunea mea pentru tehnologie și preocuparea pentru autenticitatea și proveniența produselor au convergat într-un proiect care explorează modul în care tehnologia blockchain poate aduce valoare și încredere în lumea consumabilelor.

1.1 Problema abordată

Problemele cu trasabilitatea și autenticitatea produselor sunt mai importante ca niciodată în economia modernă globalizată și digitalizată. De obicei, sistemele tradiționale de urmărire a produselor sunt netransparente, ineficiente și susceptibile la fraudă. Aceste probleme pot provoca pierderi mari de bani, riscuri pentru siguranța clienților și pierderea încrederii în produse și mărfuri.

Dificultatea de a obține informații precise și verificabile despre modul în care un produs trece de la producător la consumator este un element central al problemei. Aceste informații sunt vitale pentru asigurarea calității, autenticității și conformității cu legea. În plus, un sistem de urmărire eficient poate ajuta la identificarea produselor afectate în cazul în care este necesar, reducând riscurile pentru consumatori și îmbunătățind siguranță și încrederea companiilor.

Sistemele tradiționale de urmărire a produselor prezintă numeroase provocări și limitări, printre care:

Lipsa de claritate pentru clienți Pentru majoritatea clienților, nu este posibil să obțină informații detaliate și verificabile cu privire la itinerarul produsului. De exemplu, un client care cumpără un produs alimentar într-un lanț de magazine nu poate verifica întotdeauna exact de unde provine produsul și cum a fost transportat. Acest lucru provoacă neîncredere în calitatea și autenticitatea produsului.

Problemele pentru producători Producătorii se confruntă frecvent cu dificultăți în urmărirea întregului proces prin care produsele lor ajung la consumatori. De exemplu, un producător de îmbrăcăminte poate să nu fie pe deplin conștient de modificările sau manipulările care sunt efectuate asupra produselor sale pe parcursul lanțului de aprovizionare. Aceasta poate provoca probleme de calitate și poate afecta reputația mărcii.

Distribuitorii gestionează datele În diferite faze ale lanțului de aprovizionare, distribuitorii pot modifica sau suprascrie informațiile despre produse. Distribuitorii de produse electronice, de exemplu, pot modifica datele de producție sau certificările de calitate pentru a face produsele să pară mai noi sau mai conforme cu standardele decât sunt în realitate. Aceasta poate avea consecințe legale grave, pe lângă afectarea integrității produsului.

1.2 Scopul şi obiectivele aplicaţiei

Această aplicație își propune să ilustreze beneficiile utilizării tehnologiei blockchain, în special Ethereum, prin îmbunătățirea trasabilității, securității și transparenței datelor în sistemele de urmărire a produselor. Se dorește crearea unei interfețe web ușor de folosit atât de către utilizatori , producători cât și de verificatorii produselor.

Proiectul demonstrează modul în care tehnologia blockchain poate rezolva problemele cu sistemele tradiționale de gestionare a lanțului de aprovizionare prin crearea unei aplicații de urmărire a produselor. Aceasta nu este o soluție originală, este o ilustrare a modului în care blockchain-ul poate fi folosit în mod efectiv pentru a oferi beneficii reale atât producătorilor, cât și consumatorilor.

1.2.1 Obiective Principale

Obiectivele specifice ale acestei aplicații se concentrează pe trei aspecte principale: transparența datelor, securitatea datelor și integritatea datelor.

Transparența Datelor Obiectivul este de a se asigura că toate părțile interesate, de la producător la consumator, au acces la date clare și verificabile despre întregul proces de fabricare a unui produs. Clienții primesc mai multă încredere și sunt mai mulțumiți atunci când pot vedea dacă produsul provine dintr-un anumit loc. De exemplu, un cumpărător poate vedea informații despre producător, date de producție și traseu logistic printr-un cod QR[3] pe ambalajul unui produs. Fiecare etapă a lanțului de aprovizionare este înregistrată pe blockchain prin utilizarea contractelor inteligente în implementare. Acest lucru permite tuturor părților interesate să aibă acces rapid la datele stocate pe blockchain.

Securitatea Datelor Obiectivul este de a împiedica manipularea și accesul neautorizat la informații sensibile. Beneficiul este că protejează consumatorii și reputația companiilor, deoarece asigură că informații sensibile despre produse nu pot fi modificate sau falsificate de persoane neautorizate. De exemplu, intermediarii

[3]"Quick Response" in engleza, este o gamă de standarde de codare cu formă de bare bidimensionale.

sau distribuitorii nu au capacitatea de a modifica informațiile referitoare la certificările de calitate ale produselor alimentare. Implementarea utilizării tehnologiei de criptare și a caracteristicilor imutabile ale blockchain-ului pentru a asigura că informațiile pe care le păstrați sunt sigure și integre.

Integritatea Datelor Obiectivul este să se asigure că datele pe blockchain sunt exacte și nu pot fi modificate oricând. Implementarea implică utilizarea caracteristicilor imutabile ale blockchain-ului pentru a înregistra în mod constant toate tranzacțiile și modificările care au loc asupra unui produs.

1.2.2 Obiective Secundare

Autenticitatea Produselor Fiecare etapă a lanțului de aprovizionare este monitorizată și verificabilă cu blockchain. Un producător de vin, de exemplu, are capacitatea de a înregistra pe blockchain fiecare lot de vin, cuprinzând informații despre recoltare, producție și distribuție. Consumatorii pot verifica aceste detalii pentru a se asigura că vinul este de calitate superioară și autentic.

Reducerea Fraudelor Companiile pot reduce șansele de fraudă și contrafacere prin utilizarea blockchain-ului. Un producător de echipamente electronice, de exemplu, poate folosi blockchain-ul pentru a urmări componentele care sunt utilizate în fiecare produs, garantând că toate componentele sunt originale și îndeplinesc specificațiile.

Îmbunătățirea Încrederii Consumatorilor Atunci când blockchain-ul oferă acces la informații detaliate și verificabile despre produse, consumatorii pot avea mai multă încredere în ceea ce cumpără. De exemplu, un consumator care cumpără alimente organice ar putea folosi o aplicație bazată pe blockchain iar printr-un singur cod QR poate verifica dacă produsul provine dintr-o anumită țară și are toate certificările necesare.

Prin abordarea acestor obiective și exemplificarea acestor beneficii, această lucrare își propune să demonstreze modul în care tehnologia blockchain poate transforma sistemele de urmărire a produselor, oferind soluții eficiente și fiabile pentru problemele comune întâlnite în lanțurile de aprovizionare tradiționale.

1.3 Abordări existente

1.3.1 Gestiune a Lanţului de Aprovizionare

Sistemele de gestionare a lanțului de aprovizionare au fost implementate în diverse industrii pentru a urmări și gestiona fluxul de produse de la furnizori către consumatori [Sta14]. Aceste sisteme integrează tehnologii precum coduri de bare, RFID[4], sau IoT[5], permițând înregistrarea și monitorizarea detaliată a fiecărui pas al produselor pe parcursul lanțului de distribuție. Deși aceste soluții au îmbunătățit vizibilitatea asupra lanțului de aprovizionare, există încă provocări în ceea ce privește transparența completă și verificabilitatea în anumite etape ale procesului[LC00].

Amazon [6] Una dintre cele mai mari companii de comerț online din lume, a revoluționat industria folosind tehnologii avansate pentru a gestiona lanțul de aprovizionare într-un mod eficient. Incluzând roboți în depozite pentru gestionarea și sortarea eficientă a produselor, au redus semnificativ timpul necesar pentru procesarea comenzilor și livrarea produselor către clienți[CD05].

Cu toate acestea, Amazon se confruntă cu probleme specifice legate de monitorizarea lanțului de aprovizionare în sectorul alimentar. Monitorizarea lanțului de aprovizionare poate fi dificilă din cauza riscului de contaminare a alimentelor. Gestionarea produselor perisabile și a cerințelor stricte de stocare poate crea dificultăți în asigurarea unei urmăriri detaliate și a unei gestiuni eficiente a produselor pe durata transportului și depozitării[GRB19].

Spre exemplu in industria textilă, implementarea unui sistem de urmărire a materiilor prime prin intermediul unui lanț de aprovizionare global poate fi dificilă din cauza numeroșilor furnizori și proceselor intermediare implicate. Uneori, lipsa standardelor comune și a adopției tehnologiei în toate etapele poate crea lacune în urmărire și verificare.

1.3.2 Urmărirea Produselor bazate pe QR Code sau Cod de bare

Diverse aplicații mobile au fost dezvoltate pentru a permite consumatorilor să scaneze coduri QR sau Barcode[7] de pe produse și să obțină informații despre proveniența,

[4]RFID este prescurtarea termenului englez Radio-Frequency Identification (Identificare prin frecvență radio);

[5]Internetul obiectelor ,în engleză Internet of Things, este un concept ce presupune folosirea Internetului pentru a conecta între ele diferite dispozitive, servicii și sisteme automate;

[6]Este o companie multinațională americană care se concentrează pe e-commerce, cloud computing, reclame online, streaming digital, și inteligență artificială;

[7]Codul de bare este o reprezentare de date codificată, destinată a fi citită pe cale optică.

autenticitatea și alte detalii relevante. Aceste aplicații folosesc adesea informațiile furnizate de producători sau distribuitori pentru a oferi o perspectivă mai detaliată asupra istoricului produselor. Cu toate acestea, lipsa unui sistem robust de verificare a autenticității și a unui registru imutabil al informațiilor reprezintă încă provocări pentru aceste aplicații[GPJ07].

Open Food Facts [8] Aplicații precum Open Food Facts permit consumatorilor să scaneze codurile produselor pentru a accesa informații detaliate despre ingrediente, aditivi, proveniență și impactul lor asupra sănătății. Folosește datele colectate de la utilizatori, voluntari și alte surse pentru a furniza informații detaliate despre produsele alimentare, inclusiv ingrediente, valori nutriționale, origini, etichetări și alergeni. Ele oferă o transparență mai mare consumatorilor și îi ajută să facă alegeri informate. Aceasta oferă o modalitate simplă și accesibilă pentru consumatori să-și direcționeze achizițiile în concordanță cu valorile lor personale.

Unele aplicații bazate pe scanarea codurilor de bare pot fi influențate de informațile furnizate de producători și pot fi susceptibile la manipulare. În unele cazuri, informațiile pot fi incomplete sau inexacte, ceea ce ar putea duce la erori în identificarea provenienței sau caracteristicilor produselor. În contextul schimbărilor frecvente ale produselor alimentare (rețete noi, modificări ale ingredientelor), menținerea datelor actualizate și precise poate fi o provocare continuă pentru platformă[CDS+20].

1.3.3 Utilizarea Platformelor Blockchain

Tehnologia blockchain a fost explorată din ce în ce mai mult ca soluție pentru transparență și verificabilitate în urmărirea produselor. Prin intermediul blockchainului, fiecare etapă a lanțului de aprovizionare poate fi înregistrată și autentificată în mod imutabil, asigurându-se că informațiile nu pot fi modificate sau șterse ulterior. Aceasta permite consumatorilor să acceseze un istoric complet și verificabil al produsului, de la producător până la livrare. Cu toate acestea, implementarea extensivă a acestei tehnologii în lanțurile de aprovizionare globale necesită încă dezvoltare și adoptare largă pentru a fi aplicabilă la scară globală[AFC+19].

VeChain [9] Proiectul VeChain folosește tehnologia blockchain pentru a urmări autenticitatea și proveniența produselor de lux, precum articolele de îmbrăcăminte sau băuturile alcoolice scumpe. Acest sistem oferă o trasabilitate completă, începând de la producător și până la consumator, permițând acestuia din urmă să verifice

[8] Aplicație mobilă pentru propria baza de date;

[9] O platformă blockchain concepută pentru a îmbunătăți gestionarea lanțului de aprovizionare și procesele de afaceri.

autenticitatea produsului. Utilizarea unor identificatori unici și a unor dispozitive IoT, care înregistrează și transmit date în fiecare etapă a procesului de producție și distribuție, face posibilă această trasabilitate. Prin scanarea unui cod QR sau a unui alt tip de identificator unic, consumatorii pot accesa un istoric complet și clar al produsului. Implementarea VeChain în lanțurile de aprovizionare ajută la prevenirea contrafacerilor și la protejarea mărcilor de lux de produse false[VeC24].

În unele cazuri, tehnologiile blockchain pot avea costuri inițiale ridicate pentru implementare și pot necesita eforturi semnificative pentru a convinge toate părțile implicate să adere la această tehnologie. De asemenea, scalabilitatea și compatibilitatea cu alte sisteme existente pot reprezenta provocări în adoptarea extensivă a tehnologiei blockchain în lanțurile de aprovizionare globale[MMT22].

1.4 Tehnologii Web3

În lucrarea sa din 2008, *Bitcoin: A Peer-to-Peer Electronic Cash System*, Satoshi Nakamoto[10] a propus pentru prima dată tehnologia blockchain[Nak08]. Acesta a creat ideea de blockchain pentru a crea Bitcoin, o monedă digitală descentralizată care nu are nevoie de o administrare centrală. Blockchain-ul Bitcoin a demonstrat modul în care o rețea distribuită de noduri poate proteja și verifica tranzacțiile.

Gavin Wood[11] în 2014 a publicat un articol numit *Ethereum: A Secure Decentralized Generalized Transaction Ledger* prin care a extins conceptul de blockchain pentru a include contractele smart și a permis crearea de aplicații descentralizate [Woo14]. În consecință, Ethereum a ajutat la dezvoltarea unui ecosistem de aplicații blockchain și a devenit platforma standard pentru implementarea contractelor smart și a aplicaților descentralizate.

Flexibilitatea, suportul oferit pentru un limbaj de programare Turing[12] complet și a ecosistemului activ de dezvoltatori, au contribuit semnificativ la adoptarea Ethereum pe scară largă in industrie. Poziția sa de lider in spațiul blockchain a fost consolidată și de standardele ERC-721[13],ERC-20[14] ce asigură compatibilitatea între diferite platforme și aplicații.

Platformele blockchain oferă diferite caracteristici, fiecare fiind adecvat pentru diferite aplicații. De exemplu, Bitcoin folosește un algoritm de consens Proof-of-Work [15], care oferă securitate rețelei, dar este criticat pentru cantitatea mare de energie pe care o consumă [Nak08]. În schimb, Ethereum a trecut la Proof-of-Stake [16] cu Ethereum 2.0, deși inițial a fost bazat exclusiv pe PoW. Acest lucru a redus semnificativ consumul de energie și a îmbunătățit scalabilitatea [But14].

Platforme precum Hyperledger[17] oferă un nivel mai mare de control și confidențialitate prin utilizarea permisiunilor pentru a controla cine poate participa la rețea [Mou16].

Blockchain-ul este o tehnologie relativ nouă datorită existenței criptomonedelor precum Bitcoin și Ethereum. Cu toate acestea, potențialele blockchain-ului sunt mult

[10]Satoshi Nakamoto este pseudonimul persoanei sau grupului de persoane necunoscute care a creat moneda virtuală Bitcoin;

[11]Gavin James Wood este un informatician englez, co-fondator al Ethereum și creator al Polkadot și Kusama;

[12]Mașinile Turing sunt mecanisme extrem de elementare de dispozitive de prelucrare a simbolurilor care pot fi adaptate pentru a simula logica oricărui calculator;

[13]ERC-721 este un standard pentru crearea și gestionarea tokenilor non-fungibile pe blockchain-ul Ethereum;

[14]Asemanător cu ERC-721 dar are proprietatea de a fi la fel din punct de vedere al tipului și valorii cu un alt token;

[15](PoW) din engleza dovadă de lucru, este o măsură economică care descurajează atacuri de tipul denial of service sau alte tipuri de abuzuri;

[16]Este o clasă de mecanisme de consens pentru blockchain-uri care funcționează prin selectarea validatorilor proporțional cu cantitatea lor de deținere în criptomoneda asociată;

[17]Hyperledger este un proiect blockchain și o comunitate open source axată pe dezvoltarea de standarde, instrumente și aplicații conexe de contabiliate distribuită.

mai mari decât doar înregistrarea tranzacțiilor financiare. Blockchain-ul oferă un registru descentralizat, imuabil și transparent care se poate folosi pentru o varietate de scopuri, cum ar fi pentru a urmări produsele.

În afară de utilizarea sa pentru trasabilitatea produselor, blockchain-ul are multe utilizări promițătoare în diferite industrii. În industria financiară, blockchain-ul permite tranzacții rapide și sigure, reducând nevoia de intermediari și cheltuielile asociate. Contractele Inteligente[18] de asemenea pot ajuta la reducerea fraudei și la accelerarea răspunsului la cereri de despăgubire [CDP19].

Blockchain-ul poate îmbunătăți gestionarea datelor medicale în sectorul sănătății, menținând integritatea și confidențialitatea acestora. De exemplu, printr-un contract inteligent, un pacient poate permite unui medic acces la dosarul său medical. Acest contract garantează că informațiile sunt accesibile doar persoanelor autorizate [Tia16].

Definiție și Principii de Bază Blockchain-ul este o tehnologie de tip DLT[19] care înregistrează tranzacțiile într-un mod securizat și transparent. Cele mai importante caracteristici ale blockchain-ului sunt următoarele:

Descentralizare Nakamoto afirmă că o rețea de noduri distribuite deține controlul asupra blockchain-ului, mai degrabă decât o singură organizație[Nak08].

Imutabilitate Datorită faptului că blockchain-ul este înregistrat, informațiile nu pot fi modificate sau șterse[Mou16].

Transparența Deoarece toate tranzacțiile sunt vizibile pentru toți participanții din rețea, transparența este maximă[TT16].

Există mai multe tipuri de blockchain, fiecare având propriul scop și caracteristici. Satoshi Nakamoto descrie blockchain-urile publice ca rețele deschise în care oricine poate participa și contribui la validarea tranzacțiilor[Nak08]. Blockchain-urile private sunt mai bune pentru aplicații de afaceri, deoarece sunt controlate de o singură entitate sau un grup mic de entități [Mou16]. În domenii precum finanțele și logistica, consorțiile blockchain sunt administrate de un grup de organizații care lucrează împreună pentru a menține și valida blockchain-urile.

Blockchain-ul funcționează folosind blocuri de date care sunt conectate în ordine cronologică. Un hash criptografic al blocului anterior, un timestamp și datele tranzacțiilor sunt incluse în fiecare bloc. Nakamoto afirmă că hashingul criptografic este o operațiune matematică care transformă o intrare de date într-un șir unic de caractere, asigurând astfel integritatea datelor[Nak08]. Potrivit lui Mougayar, pentru

[18]Wiki: Un program pe calculator destinat să faciliteze, verifice, sau să aplice negocierea sau executarea unui contract;

[19]Registrul distribuit este un tip de contabilitate digitală distribuită în care conturile contabile sunt înregistrate în mai multe registre în locuri diferite în același timp.

a menține tranzacțiile în ordine cronologică, timbrul de timp indică exact momentul în care a fost creat blocul[Mou16]. Proof-of-Work, un algoritm de consens folosit de blockchain-uri publice precum Bitcoin, care necesită minerii să rezolve probleme criptografice complicate pentru a valida tranzacțiile și a crea noi blocuri.

În ceea ce privește trasabilitatea produselor, blockchain-ul oferă o serie de beneficii. Prin înregistrarea fiecărei mișcări a unui produs și a fiecărei tranzacții pe blockchain, se creează o sursă de adevăr care poate fi verificată de toate părțile implicate, ceea ce creează un grad ridicat de transparență și încredere. Deoarece datele înregistrate pe blockchain sunt sigure, există mai puține șanse ca informațiile despre produse să fie falsificate sau manipulate . În consecință, există mai puține fraude. Blockchain-ul permite identificarea rapidă și precisă a produselor afectate în cazul unui recall, reducând timpul și costurile procesului [Tia16].

În ciuda avantajelor evidente, utilizarea blockchain-ului în trasabilitatea produselor are și dezavantaje. Când se confruntă cu un volum mare de tranzacții, blockchain-urile publice, cum ar fi Ethereum, pot întâmpina probleme de scalabilitate. Chiar dacă transparența este un avantaj, aceasta poate aduce probleme de confidențialitate, în special atunci când vine vorba de date sensibile. Implementarea și menținerea unei soluții blockchain pot fi costisitoare, ceea ce poate împiedica adoptarea pe scară largă.

Scalabilitatea blockchain-ului este printre cele mai dificile probleme. Gestionarea unui număr mare de tranzacții pe secundă poate fi o provocare pentru rețelele blockchain publice precum Bitcoin și Ethereum. Implementarea soluțiilor de nivel 2 ar putea include soluții precum rețelele Lightning[20] pentru Bitcoin și sharding pentru Ethereum 2[But14].

O altă problemă semnificativă este cu privire la confidențialitate. Cu toate acestea, una dintre caracteristicile esențiale ale blockchain-ului este transparența, care poate fi o problemă în cazul datelor sensibile. Pentru a permite tranzacțiile private pe blockchain-uri publice, sunt studiate soluții precum ZK-SNARKs[21][Woo14].

1.4.1 Ethereum

Ethereum este un sistem blockchain cu sursă deschisă, descentralizat care are funcționalitățile contractelor inteligente. A fost propus de programatorul Vitalik Buterin[22] la sfârșitul anului 2013, iar proiectul a primit finanțare prin Crowdfunding [23] în 2014. Cu 72 de milioane de monede preminate, rețeaua a început să funcționeze la 30

[20]Lighting Network este un tip de contabilitate digitală distribuită în care conturile contabile sunt înregistrate în mai multe registre în locuri diferite în același timp;

[21]Este o dovadă criptografică care permite unei părți să demonstreze că deține anumite informații și să nu le dezvăluie;

[22]Vitalik Buterin , este un programator canadian și co-fondator al Ethereum;

[23]Crowdfundingul este o tehnică de finanțare a proiectelor folosind resurse online desprinsă din crowdsourcing.

iulie 2015. În ceea ce privește capitalizarea de piață, criptomoneda proprie Ethereum, Ether [24], este a doua după Bitcoin din 2021.

Ethereum extinde conceptul de blockchain inițiat de Bitcoin pentru a include o caracteristică pe care Bitcoin nu o oferă: contractele inteligente. Contractele inteligente sunt scripturi de cod care sunt stocate în blockchain și care se execută automat atunci când sunt îndeplinite anumite condiții. Aceste contracte inteligente permit crearea de aplicații descentralizate, care rulează pe rețeaua Ethereum și funcționează fără întreruperi, fraude, control sau interferențe din partea unei terțe părți. "Ethereum este o platformă descentralizată pe care rulează contracte inteligente: aplicații care funcționează exact așa cum au fost programate, fără nicio posibilitate de întrerupere, fraudă sau interferență din partea unor terți"[Fou24].

Mașina virtuală Ethereum [25] este mediul de execuție pentru contractele inteligente din Ethereum. Este o mașină virtuală Turing completă care permite oricui să ruleze orice program, indiferent de limbajul de programare, dacă dispune de suficient timp și memorie. EVM permite executarea codului exact așa cum a fost prevăzut, ceea ce face ca procesul de dezvoltare a aplicațiilor blockchain să fie mult mai simplu și mai eficient. *Mașina virtuală Ethereum este calculatorul global, descentralizat, care rulează codul pentru aplicațiile descentralizate*[Fou24].

Contractele inteligente sunt elementele de bază ale aplicațiilor Ethereum. Acestea sunt contracte care se execută singure, în care termenii acordului sau condițiile sunt scrise direct în linii de cod. Aceste contracte se află pe blockchain-ul Ethereum și se execută automat atunci când sunt declanșate de evenimente predefinite. Imutabilitatea și securitatea blockchain-ului asigură faptul că termenii contractelor inteligente sunt întotdeauna aplicați așa cum sunt scrise. "Contractele inteligente sunt programe informatice care controlează direct activele digitale. Acestea sunt stocate pe blockchain-ul Ethereum, asigurându-se că sunt inviolabile și funcționează exact așa cum au fost programate"[AW18].

Aplicațiile descentralizate [26] sunt aplicații care rulează pe o rețea descentralizată, spre deosebire de aplicațiile tradiționale care rulează pe un server centralizat. DApps valorifică contractele inteligente și blockchain-ul Ethereum pentru a furniza servicii direct utilizatorilor, fără a fi nevoie de intermediari. Aceste aplicații beneficiază de securitatea, transparența și reziliența blockchain-ului.

Ethereum a folosit inițial un mecanism de consens de tip Proof-of-Work, similar cu cel al Bitcoin, în care minerii concurau pentru a rezolva probleme matematice complexe pentru a adăuga noi blocuri la blockchain. Cu toate acestea, în 2022, Ethereum a trecut la un mecanism de consens Proof-of-Stake prin actualizarea Ethereum 2.0. Acesta

[24]ETH este criptomoneda a cărui blockchain este generat de platforma Ethereum;
[25]EVM este un mediu care rulează contracte inteligente Ethereum;
[26]Aplicațiile descentralizate bazate pe tehnologia de registru distribuit și blockchain.

îmbunătățește securitatea, scalabilitatea și eficiența energetică a rețelei Ethereum. "Ethereum a trecut de la Proof-of-Work la Proof-of-Stake în 2022, ceea ce face ca rețeaua să fie mai sigură, mai scalabilă și mai eficientă din punct de vedere energetic" [Fou24].

1.4.2 Arhitectura Aplicațiilor Web Descentralizate

O aplicație web descentralizată este o aplicație care funcționează pe blockchain sau un alt tip de rețea distribuită. O aplicație descentralizată este diferită de o aplicație web convențională prin faptul că nu depind de un server centralizat pentru a stoca logica de afaceri și datele. În schimb, acestea folosesc blockchain-ul pentru a asigura descentralizarea, securitatea și integritatea datelor lor. Componentele esențiale ale unui DApp includ diferite tehnlogii și concepte informatice.

Contracte intelgiente Codul care gestionează logica aplicației și funcționează pe blockchain. De obicei, acestea sunt scrise în limbaje specificate de programare, cum ar fi Solidity[27] pentru Ethereum[Woo14].

Interfața utilizatorului Reprezintă front-end-ul aplicației, care se poate crea folosind tehnologii web convenționale precum HTML[28], CSS[29] și JavaScript[30]. Interfața comunică cu blockchain-ul prin intermediul unei biblioteci sau a unui API[31][But14].

Stocare off-chain Soluțiile de stocare off-chain precum IPFS[32] sunt utilizate pentru a stoca date mari sau fișiere care nu pot fi stocate pe blockchain, cum ar fi imagini sau videoclipuri.

Biblioteci și API-uri Acestea ajută contractele smart și interfața utilizator să comunice mai bine. Biblioteca de tip Web3.js sau Ethers.js sunt esențiale pentru această funcționalitate [Woo14].

1.4.3 Contracte inteligente

Pentru a oferi mai multă automatizare și securitate diferitelor aplicații, contractele smart sunt o parte importantă a tehnologiei blockchain. Acestea sunt programe informatice bazate pe un blockchain și sunt capabile să execute automat anumite acțiuni

[27]Un limbaj de programare orientat obiect pentru scrierea de contracte inteligente;

[28]HyperText Markup Language este un limbaj de marcare utilizat pentru crearea paginilor web ce pot fi afișate într-un browser;

[29]CSS sau Cascading Style Sheets este un standard pentru formatarea elementelor unui document HTML;

[30]JavaScript este un limbaj de programare orientat obiect bazat pe conceptul prototipurilor;

[31]Application Programming Interface reprezintă un set de definiții de sub-programe, protocoale și unelte pentru programarea de aplicații și software;

[32]InterPlanetary File System este un protocol de comunicații și rețea descentralizată peer-to-peer.

atunci când sunt îndeplinite anumite condiții. În multe industrii, cum ar fi trasabilitatea produselor, utilizarea contractelor smart oferă numeroase beneficii în ceea ce privește eficiența, securitatea și transparența proceselor.

Un contract inteligent este un protocol informatizat care ajută, verifică sau execută automat respectarea termenilor unui contract. Conform lui Wood, funcționează pe o platformă blockchain și este alimentat de cod scris în anumite limbaje de programare, cum ar fi Solidity pentru Ethereum. Părțile care participă la un acord pot avea încredere că termenii acordului vor fi respectați automat și fără intervenția unui intermediar cu ajutorul contractelor smart[Woo14].

Utilizând limbaje de programare specifice, contractele smart sunt create și implementate pe blockchain-uri precum Ethereum. Procesul de operare constă din următoarele etape:

Scrierea codului Dezvoltatorii folosesc limbaje precum Solidity pentru a scrie codul inteligent al contractului. Codul descrie ce trebuie făcut și ce condiții trebuie îndeplinite [Woo14].

Implementarea pe blockchain Codul contractului smart este implementat pe blockchainul selectat. Contractul smart nu poate fi modificat și devine o componentă esențială a blockchain-ului odată ce este implementat.

Executarea automată Atunci când sunt îndeplinite condițiile specificate, contractul inteligent se execută automat. De exemplu, atunci când plata este confirmată, un contract inteligent pentru vânzarea unui produs va transfera automat banii și proprietatea asupra produsului.

Verificare și Securitate Blockchain-ul protejează și verifică contractele smart, reducând riscul de manipulare sau fraudă.

Când vine vorba de trasabilitatea produselor, contractele smart pot îmbunătăți transparența și securitatea lanțurilor de aprovizionare. Contractele smart pot automatiza și înregistra fiecare mișcare a unui produs pe blockchain, asigurând un flux de date continuu și securizat.

Un contract inteligent poate fi, de exemplu, utilizat într-un lanț de aprovizionare alimentar pentru a monitoriza modul în care bunurile ajung de la fermă până la rafturile magazinului. Fiecare proces, inclusiv procesarea și transportul, poate fi automatizat și verificat, ceea ce reduce riscul de fraudă și garantează integritatea produselor[Tia16].

În ciuda avantajelor evidente, utilizarea contractelor smart are și câteva dezavantaje.

Complexitatea progresului Scrierea și aplicarea contractelor smart necesită o înțelegere profundă a limbajelor de programare specifice, cum ar fi Solidity, precum și cunoștințe tehnice avansate[Woo14].

Imuabilitatea Odată ce un contract inteligent este implementat nu poate fi modificat. Aceasta poate duce la probleme dacă se întâmplă greșeli sau trebuie făcute ajustări[But14].

Scalabilitatea Contractele smart pot întâmpina probleme de scalabilitate, în special în cazul unui număr mare de tranzacții[Mou16].

1.4.4 Solidity

Solidity este un limbaj de programare orientat spre contracte, conceput pentru dezvoltarea de contracte inteligente care rulează pe Mașina Virtuala Ethereum. A fost dezvoltat de echipa Ethereum și este principalul limbaj pentru scrierea contractelor inteligente pe Ethereum și pe alte platforme blockchain care acceptă EVM.

`Solidity` a fost propus de Gavin Wood, unul dintre co-fondatorii Ethereum, în august 2014. Solidity este un limbaj de nivel înalt, orientat pe contracte, pentru implementarea contractelor inteligente. Acesta a fost influențat de C++[33], Python[34] și JavaScript[Woo14].

Solidity este conceput special pentru scrierea de contracte inteligente, care sunt contracte care se execută singure, cu termenii acordului direct în cod. Contractele inteligente de pe Ethereum pot fi utilizate pentru o gamă largă de aplicații, de la finanțe descentralizate la jocuri și gestionarea lanțului de aprovizionare.

1.4.5 Metamask

MetaMask este un portofel popular de criptomonede și o poartă de acces la aplicațiile blockchain. Acesta permite utilizatorilor să interacționeze cu blockchain-ul Ethereum fără a fi nevoie să descarce întregul blockchain. MetaMask oferă o interfață ușor de utilizat pentru gestionarea conturilor Ethereum, efectuarea de tranzacții și interacțiunea cu aplicațiile descentralizate.

Portofelul a fost creat de ConsenSys[35] în 2016 și este disponibil ca extensie de browser și ca aplicație mobilă. Aceasta acționează ca o punte de legătură între browserul web al utilizatorului și blockchain-ul Ethereum, permițând utilizatorilor să își gestioneze activele digitale și să interacționeze cu DApps fără probleme.„MetaMask este o

[33]C++ este un limbaj de programare de nivel înalt, cu scop general, creat de informaticianul danez Bjarne Stroustrup;

[34]Python este un limbaj de programare de nivel înalt, cu scop general;

[35]Consensys este o companie privată de tehnologie software blockchain fondată de Joseph Lubin și cu sediul în Fort Worth;

extensie de browser care permite utilizatorilor să gestioneze portofelele Ethereum și să interacționeze cu aplicațiile descentralizate"[Met24].

MetaMask oferă mai multe caracteristici cheie care îl fac un instrument valoros pentru utilizatorii blockchain-ului Ethereum:

Gestionarea portofelului Utilizatorii pot crea și gestiona mai multe conturi Ethereum, fiecare cu propria cheie privată.

Tranzacții sigure MetaMask oferă un mediu securizat pentru trimiterea și primirea de Ether și alte token-uri cum ar fi ERC-721.

Interacțiunea DApp Utilizatorii își pot conecta portofelul MetaMask la diverse DApps, permițând o interacțiune și semnarea tranzacțiilor fără probleme.

Personalizarea rețelei MetaMask suportă mai multe rețele Ethereum, inclusiv rețeaua principală , rețele de teste și rețele personalizate. „MetaMask permite utilizatorilor să ruleze dApps Ethereum direct în browserul lor fără a rula un nod Ethereum complet" [Met24].

`MetaMask` pune accentul pe securitate și confidențialitate, oferind caracteristici precum protecția prin parolă, backup-ul frazei de recuperare și integrarea portofelului hardware. Cheile private ale utilizatorilor sunt criptate și stocate la nivel local pe dispozitivele lor, asigurându-se că aceștia păstrează controlul asupra activelor lor. „MetaMask oferă utilizatorilor o modalitate sigură de a-și gestiona cheile private Ethereum și de a interacționa cu aplicațiile descentralizate"[Met24].

Portofelul digital este esențial pentru dezvoltatorii Ethereum, oferind o interfață simplă pentru a interacționa cu contractele inteligente și aplicațile descentralizate[AW18]. De asemenea, dezvoltatorii pot folosi MetaMask pentru a se conecta la rețele blockchain locale în scopuri de testare.

1.4.6 Javascript

`JavaScript` este un limbaj de programare versatil, de nivel înalt, care este una dintre tehnologiile de bază ale World Wide Web[36], alături de HTML și CSS. Acesta permite realizarea de pagini web interactive și este o parte esențială a aplicațiilor web. Marea majoritate a site-urilor web îl utilizează pentru comportamentul paginilor de pe partea clientului, iar toate browserele web importante au un motor JavaScript dedicat pentru a-l executa.

`JavaScript` a fost creat de Brendan Eich în 1995[37], în timpul activității sale la Netscape Communications[38]. Dezvoltat inițial sub numele de "Mocha", a fost redenumit

[36]WWW este totalitatea site-urilor / documentelor și informațiilor de tip hipertext legate între ele
[37]Brendan Eich este un programator american și creatorul limbajului de scripting JavaScript;
[38]Este o companie americană ce oferă servicii pentru calculatoare;

ulterior "LiveScript" și, în cele din urmă, `JavaScript`. Limbajul a evoluat semnificativ de la începuturile sale și este acum unul dintre cele mai populare limbaje de programare din lume. "JavaScript este un limbaj de programare care se conformează specificației ECMAScript. JavaScript este de nivel înalt, adesea compilat just-in-time și multiparadigmă"[(MD24]. JavaScript oferă câteva caracteristici cheie care îl fac o alegere atractivă pentru dezvoltarea web.

Tipare dinamică JavaScript este tipizat dinamic, ceea ce înseamnă că variabilele nu au nevoie de un tip predefinit și pot conține diferite tipuri de valori în momente diferite.

Prototipuri JavaScript utilizează moștenirea bazată pe prototipuri, ceea ce permite obiectelor să moștenească proprietăți și metode de la alte obiecte.

Funcții de primă clasă Funcțiile din JavaScript sunt obiecte de primă clasă, ceea ce înseamnă că pot fi stocate în variabile, transmise ca argumente altor funcții și returnate de funcții.

Programare bazată pe evenimente JavaScript este foarte potrivit pentru programarea bazată pe evenimente, permițând dezvoltatorilor să scrie cod care răspunde la interacțiunile cu utilizatorul și la alte evenimente.

JavaScript interacționează cu modelul Document Object Model[39] pentru a actualiza și manipula în mod dinamic conținutul web. DOM reprezintă structura unei pagini web, iar JavaScript poate fi utilizat pentru a adăuga, elimina și modifica elemente și atribute, pentru a gestiona evenimente și pentru a schimba stiluri în mod dinamic.

1.4.7 React

React este o bibliotecă JavaScript populară pentru crearea de interfețe utilizator, în special aplicații cu o singură pagină în care datele se pot schimba fără a fi necesară reîncărcarea paginii. A fost dezvoltată de Facebook[40] și este întreținută de Facebook și de o comunitate de dezvoltatori individuali și companii. React permite dezvoltatorilor să creeze aplicații web de mari dimensiuni care se pot actualiza și reda eficient ca răspuns la modificările de date.

A fost implementat pentru prima dată în fluxul de știri al Facebook în 2011 și mai târziu în Instagram în 2012. React simplifică procesul de construire a interfețelor interactive, permițând dezvoltatorilor să construiască componente încapsulate care își gestionează propria stare, iar apoi să le compună pentru a crea interfețe complexe.

[39]DOM este o interfață de programare a aplicațiilor inter-platformă și independentă de limbaj care tratează un document structură de arbore;

[40]Este un site web de tip rețea de socializare din Internet.

React oferă câteva caracteristici cheie care îl fac o alegere atractivă pentru dezvoltarea front-end.

Arhitectura bazată pe componente React încurajează dezvoltatorii să construiască interfețe de utilizare folosind componente reutilizabile, ceea ce ajută la întreținerea și scalarea aplicațiilor mari.

DOM virtual React utilizează un DOM virtual pentru a optimiza actualizările și redarea. Atunci când datele se modifică, DOM virtual este actualizat mai întâi, iar apoi actualizează eficient DOM real pentru a se potrivi.

JSX Este o extensie de sintaxă care permite scrierea HTML direct în JavaScript. Aceasta face codul mai ușor de citit și mai ușor de scris.

Flux de date unidirecțional React impune un flux de date unidirecțional, ceea ce face mai ușor de înțeles și de depanat aplicațiile.

"Componentele vă permit să împărțiți interfața de utilizare în bucăți independente, reutilizabile, și să vă gândiți la fiecare bucată în parte"[Rea24].

Hooks, care au fost introduse în React 16.8, sunt funcții care permit dezvoltatorilor să folosească stagiul și alte caracteristici React fără a scrie o clasă. UseState si UseEffect sunt printre cele mai folosite[Rea24].

1.4.8 Node.js

Node.js este un mediu de execuție JavaScript open-source și multi-platformă care permite dezvoltatorilor să ruleze JavaScript pe partea de server.A fost creat în 2009 de Ryan Dahl[41] cu scopul de a construi cu ușurință programe de rețea scalabile. "Node.js este un timp de execuție JavaScript construit pe motorul JavaScript V8 al Chrome. Folosește un model de I/O bazat pe evenimente, fără blocaj, care îl face ușor și eficient" [Nod24].

Node.js a câștigat o popularitate semnificativă datorită capacității sale de a gestiona un număr mare de conexiuni simultane cu un debit ridicat.

Este utilizat pe scară largă în dezvoltarea de diverse tipuri de aplicații, inclusiv servere web, aplicații de chat în timp real, API-uri și microservicii.Node.js a devenit o alegere populară pentru dezvoltarea de aplicații în timp real datorită capacității sale de a gestiona un număr mare de conexiuni simultane cu un debit ridicat.

[41]Este un inginer software american care este foarte bine cunoscut pentru crearea Node.js JavaScript.

1.4.9 Hardhat

Hardhat este un mediu de dezvoltare complet pentru Ethereum care facilitează compilarea, implementarea, testarea și depanarea contractelor inteligente. Acesta este conceput pentru a face procesul de dezvoltare mai ușor și mai eficient, oferind o varietate de instrumente și plugin-uri pentru a ajuta dezvoltatorii să construiască aplicații descentralizate robuste.

Acesta a fost dezvoltat pentru a aborda complexitatea și provocările legate de dezvoltarea contractelor inteligente și a aplicațiilor descentralizate pe blockchain-ul Ethereum. Acesta oferă un cadru flexibil și extensibil care se integrează perfect cu alte instrumente populare de dezvoltare Ethereum[Lab24].

Mediul de dezvoltare simplifică procesul de dezvoltare a contractelor inteligente prin furnizarea de instrumente pentru compilarea codului Solidity, rularea de teste automate și implementarea contractelor. Integrarea sa cu Hardhat Network permite depanarea și testarea detaliată a contractelor inteligente înainte de a le implementa în rețelele reale.

Capitolul 2

Arhitectura aplicației

Aplicația descentralizată pentru urmărirea produselor în Ethereum utilizează o interfață grafică web React. Aceasta permite producătorilor și verificatorilor să adauge și să valideze informațiile despre produse. Aplicația permite utilizatorilor să scaneze un cod QR pentru a vizualiza statusul unui produs (vezi figura 2.1), iar datele produselor sunt gestionate de către contractele inteligente.

Autentificarea producătorilor și verificatorilor se face prima dată prin JWT[1], iar mai apoi prin Metamask. Utilizatorii pot scana coduri QR pentru a accesa informațiile despre produse. Conexiunea `JWT` este utilizată pentru autentificarea și autorizarea producătorilor și verificatorilor, asigurând o comunicare securizată între client și server. Integrarea Metamask permite acestora să interacționeze cu blockchain-ul direct din browser.

2.1 Componentele Principale ale Aplicației

Frontend Este construit folosind React, un framework `JavaScript`, utilizat pentru o interfață dinamică și modulară. Prin intermediul interfeței, producătorii și verificatorii pot adăuga respectiv valida produsele, iar utilizatorii pot scana coduri QR pentru a accesa informațiile. Acesta dispune și de conexiunea între client și server, cu ajutorul JWT și Metamask.

Backend Serverul aplicației este construit cu Node.js și Express.js[2] pentru gestionarea solicitărilor HTTP[3], autentificării și interacțiunii cu blockchain-ul. Express este folosit în principal pentru gestionarea rutelor, JWT este folosit pentru autenti-

[1]JSON Web Token este un standard Internet propus pentru crearea de date cu semnătură opțională și/sau criptare opțională ;

[2]Express este un cadru ușor de utilizat care simplifică procesul de dezvoltare a aplicațiilor Node;

[3]Este un protocol de nivel aplicație în cadrul modelului suitei de protocoale Internet pentru sisteme de informații distribuite, colaborative și hipermedia;

ficare, Body-Parser[4] pentru parsarea corpurilor de solicitare în format JSON[5], CORS[6] și Multer[7] pentru gestionarea resurselor între domenii diferite și QRCode pentru generarea codurilor pentru produse.

Comunicarea cu blockchain Pentru testarea și dezvoltarea contractelor inteligente, aplicația utilizează Hardhat. Acesta oferă un mediu de testare local. `MetaMask` permite utilizatorilor să interacționeze cu contractele smart. Utilizând butoanele din interfață care apelează contractele smart, puteți adăuga și valida produsele. Contractele inteligente, odată implementate, supraveghează înregistrarea, validarea și actualizarea datelor legate de produse.

2.1.1 Componentele front-end ale aplicației

Acasa Pagina principală de unde fiecare actor poate incepe interacțiunea cu sistemul.

Scanare cod QR Funcționalitate care permite utilizatorilor să vizualizeze statusul produsului scanând un cod QR.

Status Produs După scanarea codului QR, utilizatorul poate vizualiza detaliile produsului.

Logare Permite logarea cu drepturi de producător sau verificator utilizând JWT.

Logare Metamask După logarea cu server-ul este necesară logarea cu Metamask.

Verificare cont Atât producătorul cât și verificatorul pot vizualiza detalile contului.

Vizualizare produse Producătorii pot vizualiza produsele deja adăugate și pot accesa detalile acestora.

Adăugare Produs Producătorii pot adăuga produse noi.

Verificare produs Funcționalitate care este accesibilă verificatorilor pentru a valida produsele.

Delogare Atât producătorii cât și verificatorii se pot deloga de la server.

[4]Body-parser este un middleware Node.js de comparare a corpurilor;

[5]JavaScript Object Notation este un format de fișier standard deschis și un format de schimb de date care utilizează text lizibil pentru a stoca și transmite obiecte de date constând în perechi atribut-valoare și matrici;

[6]Este un mecanism bazat pe antetul HTTP care permite unui server să indice orice origine , alta decât a sa, din care un browser ar trebui să permită încărcarea resurselor;

[7]Este un middleware node.js care este utilizat în principal pentru încărcarea fișierelor.

2.2 Perspectiva externa

2.2.1 Cazuri de Utilizare

Diagrama cazurilor de utilizare prezentată ilustrează principalele interacțiuni dintre utilizatori, producători, verificatori și sistemul de tracking de produse dezvoltat pe blockchain-ul Ethereum. Aplicația utilizează `QRCode` pentru scanarea produselor, contracte smart pentru gestionarea datelor despre produse și autentificare prin JWT și Metamask(vezi figura 2.1).

Actorii principali

Utilizator Persoana care scanează codul QR pentru a verifica statusul unui produs.

Producător Persoana sau entitatea care adaugă produse noi în sistem și generează coduri QR pentru acestea.

Verificator Persoana sau entitatea care validează produsele și adaugă informații suplimentare.

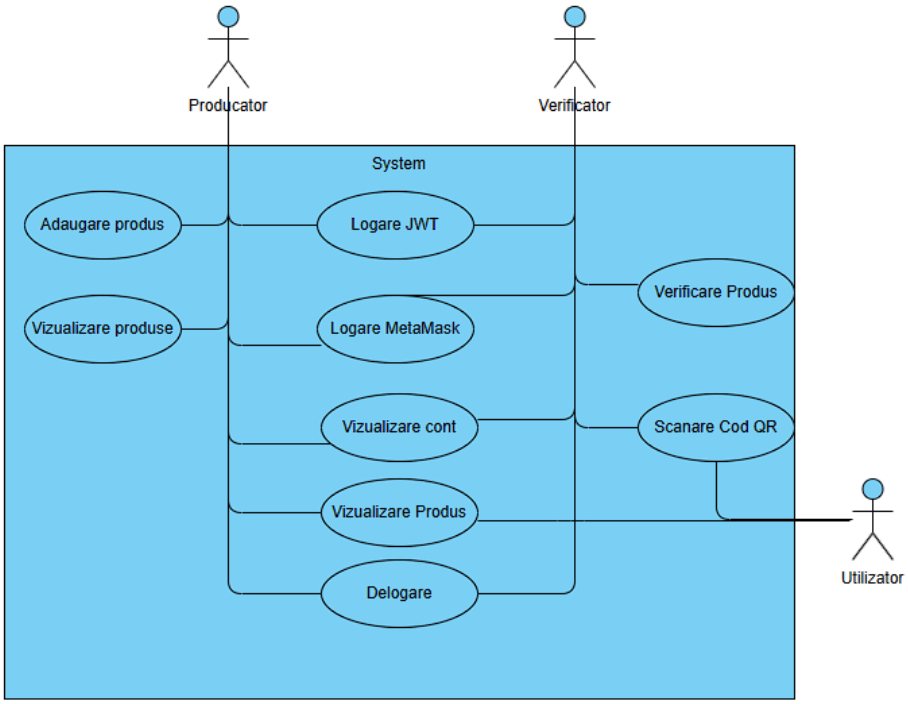

Figura 2.1: Cazuri de utilizare

Diagrama cazurilor de utilizare prezentată ilustrează principalele interacțiuni dintre actorii sistemului și funcționalitățile oferite de aplicația descentralizată pentru urmărirea produselor. Aceasta arată cum diferiți utilizatori pot interacționa cu sistemul în diferite moduri, executând multiple sarcini.

2.2.2 Diagrame secvențiale

Diagramele de secvență prezentate ilustrează principalele interacțiuni dintre actorii principali (utilizator, producător, verificator) și componentele sistemului (frontend, MetaMask, blockchain) pentru diferite scenarii din aplicația Web3 de urmărire a produselor.

Adăugare Produs de Către Producător

Producator

Flux Producătorul introduce datele produsului în interfața aplicației. Frontend-ul trimite datele către MetaMask pentru semnarea tranzacției de adăugare produs. MetaMask semnează tranzacția și o trimite către blockchain. Blockchain-ul confirmă tranzacția de adăugare a produsului. Confirmarea tranzacției este trimisă înapoi către MetaMask. MetaMask trimite confirmarea către frontend. Frontend-ul afișează codul QR generat pentru produs producătorului (vezi figura 2.2).

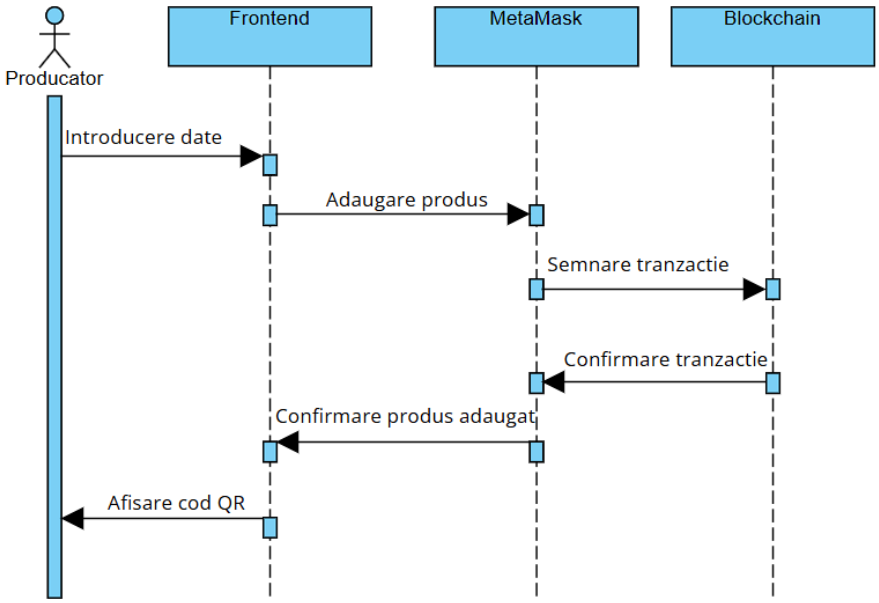

Figura 2.2: Diagrama Secvențe Producător

Validare Produs de Către Verificator

Verificator

Flux Verificatorul introduce datele de validare în interfața aplicației. Frontend-ul trimite datele către MetaMask pentru semnarea tranzacției de validare produs. MetaMask semnează tranzacția și o trimite către blockchain. Blockchain-ul confirmă

tranzacția de validare a produsului. Confirmarea tranzacției este trimisă înapoi către MetaMask. MetaMask trimite confirmarea către frontend. Frontend-ul afișează statusul actualizat al produsului verificatorului(vezi figura 2.3).

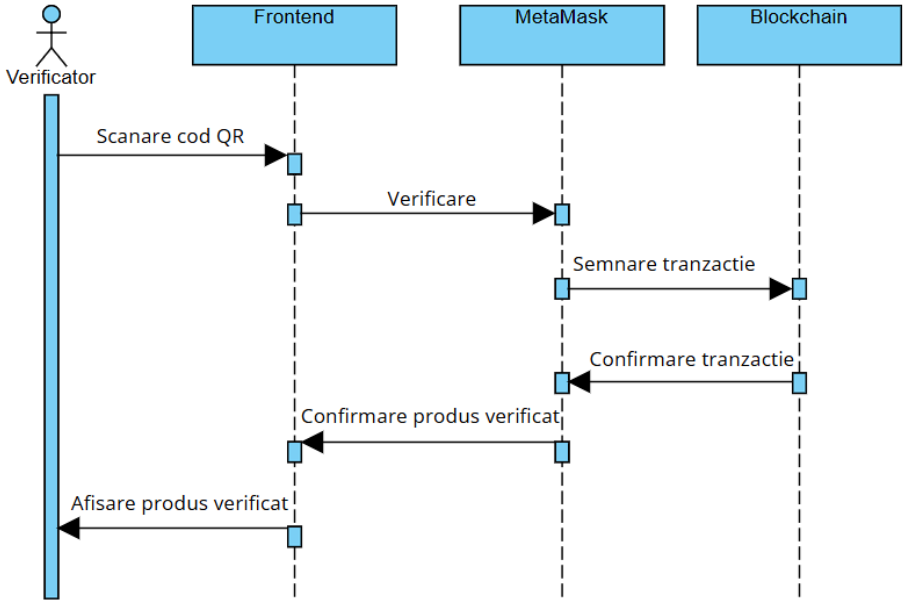

Figura 2.3: Diagramă de Secvențe Verficator

2.2.3 Diagrama de Interfață

Diagrama de mai jos ilustrează structura interfeței pentru aplicația web destinată urmării produselor pe blockchain-ul Ethereum. Fiecare nod din diagramă reprezintă o pagină sau o secțiune a aplicației, iar săgețile indică posibilele navigații între aceste secțiuni(vezi figura 2.4).

Prima pagină prezintă funcționalitățile aplicației și permite utilizatorilor să interacționeze cu ele. Utilizatorii pot accesa pagina de scanare a codului QR sau pagina de autentificare de aici. După autentificare, utilizatorii sunt redirecționați fie la ecranul Deployer, fie la ecranul Verifier, în funcție de rolul lor.

Utilizatorii pot accesa o pagină de profil specifică. Producătorul poate vedea produsele existente sau poate adăuga altele noi, în timp ce verificatorul poate verifica un produs prin scanarea codului QR, ceea ce îi permite să ajungă la pagina de informații despre produs.

2.3 Perspectiva interna

Putem împărți procesul în mai multe părți importante care lucrează împreună pentru a detalia arhitectura sistemului de urmărire a produselor pe Ethereum. Această

structură include interfața utilizator care este construită folosind React și JavaScript, precum și serverul backend Node.JS, blockchain-ul Ethereum folosit pentru contractele inteligente, unde folosim Solidity și Hardhat pentru a crea rețele de teste, precum și servicii suplimentare precum QRCode și Metamask.

Aplicația folosește React Router[8] pentru a gestiona rutele și navigația, iar Bootstrap[9] este utilizat pentru stilizarea interfeței. Librăria Axios[10] este utilizată pentru a gestiona solicitările HTTP către backend.

Node.js și Express.js sunt utilizate pentru a construi backend-ul aplicației. Acesta este responsabil pentru autentificarea utilizatorilor, încărcarea și gestionarea fișierelor, crearea codurilor QR și gestionarea datelor produselor. În timp ce JWT este folosit pentru autentificare și autorizare, Multer se ocupă de încărcarea fișierelor. Librăria QRCode este folosită pentru generarea codurilor QR. CORS și Body-Parser sunt utilizate pentru gestionarea cererilor HTTP și pentru parsarea datelor din corpul cererilor.

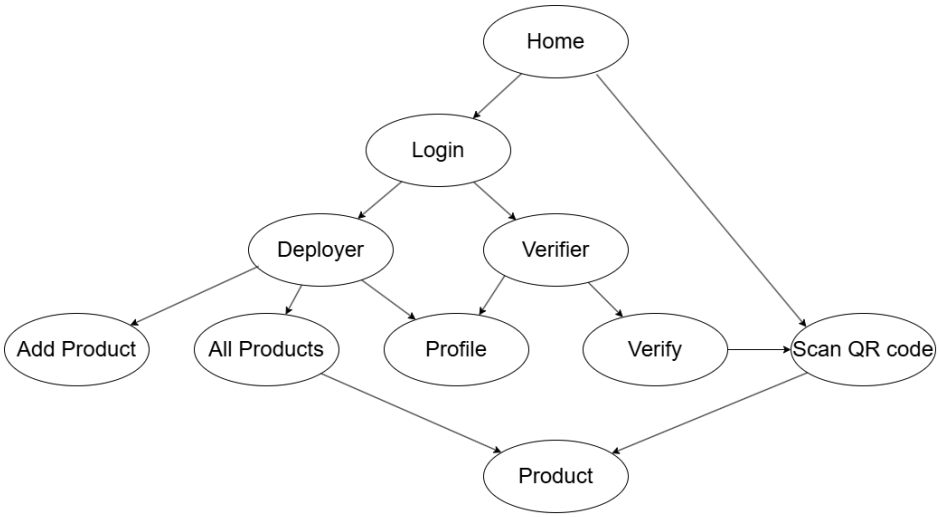

Figura 2.4: Diagrama de Interfață

Logica de tranzacționare pe blockchain-ul Ethereum este gestionată de contractul inteligent creat în Solidity. Acesta permite stocarea și verificarea datelor produselor într-un mod sigur și descentralizat. Hardhat este un instrument popular pentru dezvoltarea blockchain care ajută la dezvoltarea și testarea contractelor inteligente. Logica de urmărire a produselor, care include înregistrarea și verificarea datelor, este gestionată cu ajutorul contractelor inteligente.

Două instrumente esențiale pentru a interacționa cu blockchain-ul Ethereum din aplicație sunt MetaMask și Web3.js[11]. MetaMask este o extensie pentru browser care

[8]React Router DOM este un pachet care oferă legături pentru utilizarea aplicației web;

[9]Bootstrap este un cadru CSS gratuit și open-source destinat dezvoltării web front-end responsive, mobile-first;

[10]Axios este un client HTTP pentru node.js și browser.

[11]Este o colecție de biblioteci TypeScript și JavaScript care permite dezvoltatorilor să interacționeze cu nodurile Ethereum locale sau la distanță;

permite utilizatorilor să se autentifice cu un cont la o rețea specifică și să efectueze tranzacții ulterior. Web3.js este o librărie JavaScript care permite front-end-ului și MetaMask să comunice cu blockchain-ul Ethereum.

2.3.1 Diagrama de relație

Contractul inteligent este conceput pentru a urmări produsele pe blockchain-ul Ethereum. Acesta permite înregistrarea produselor și adăugarea istoricului în cazul verificării acestora. Diagrama de mai jos prezintă relațiile dintre entitățile Product și ProductHistory în contextul contractului inteligent. Relația dintre entități este one-to-many, fiecare produs poate avea mai multe înregistrări în istoricul său.

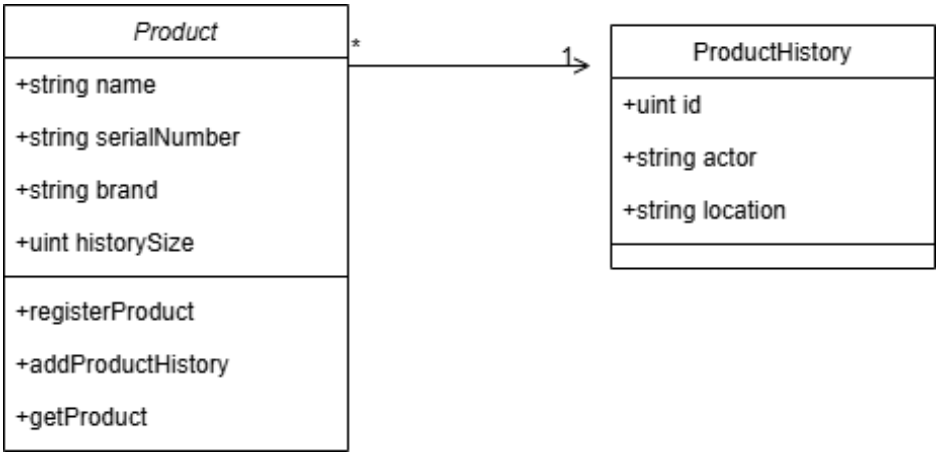

Figura 2.5: Diagrama de relație

Înregistrarea unui nou produs pe blockchain duce la inițializarea istoriei cu prima înregistrare compusă din Actor și locație. Verificarea produsului duce la adăugarea în istoric a unui nou Actor și a locației acestuia. La cererea detaliilor despre produs, funcția getProduct returnează toate înregistrările produsului.

2.3.2 Diagrama de activitate

Diagrama de activitate prezentată mai jos ilustrează fluxul de procese pentru adăugarea unui produs în rețeaua blockchain utilizând autentificare prin `JWT`, autentificare prin MetaMask și semnarea tranzacțiilor.

Procesul începe cu etapa de logare a utilizatorului, unde acesta introduce token-ul corespunzător producătorului și așteaptă confirmarea de logare în cont. Dacă token-ul nu este valid, procesul se oprește aici. Altfel, producătorul este conectat și urmează să se conecteze la `MetaMask`. Doar când acesta s-a conectat poate începe procesul de adăugare produs, completând un formular în pagina respectivă. Dacă datele sunt valide sau fondurile suficiente, acesta poate continua. La final, dacă producătorul confirmă tranzacția din extensia MetaMask, produsul este adăugat cu succes.

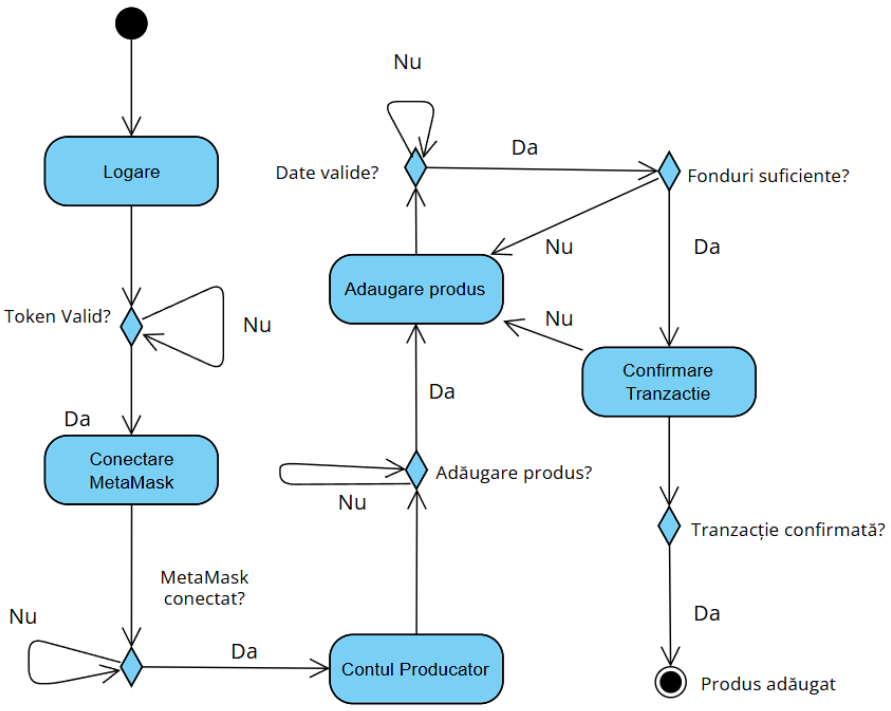

Figura 2.6: Diagrama de activitate

2.4 Perspectiva instalarii

Instalarea aplicației `Web3` pentru urmărirea produselor pe blockchain implică mai multe etape care acoperă configurarea și rularea componentelor frontend, backend și a contractelor inteligente.

Node.js si npm Pentru a rula aplicația React, este necesară instalarea `Node.js`. Acesta poate fi descărcat și instalat de pe site-ul oficial. Dupa instalarea Node.js am configurat proiectul React utilizand `npx create-react-app dapp-tracking full-stack-app`. Pentru rularea aplicației este nevoie de comanda `npm start` ce va deschide automat aplicația în browser la adresa `http://localhost:3000`.

Backend În directorul backend am inițializat un nou proiect Node.js și am instalat pe parcurs diferite dependințe pentru Express, JWT sau QRCode. Pentru rularea acestuia este necesară comanda `node server.js` în terminal.

Hardhat Acesta a fost inițializat utilizând mai multe comenzi și urmând mai mulți pași specifici versiunii folosite. Hardhat trebuie compilat , testat `npx hardhat compile`, `npx hardhat test`, iar mai apoi implementat în rețea folosind comanda `npx hardhat run scripts/deploy.js --network localhost`.

Capitolul 3

Detalii de implementare

Acest capitol detaliază aspectele tehnice ale implementării sistemului dezvoltat pentru a asigura trasabilitatea produselor utilizând tehnologia blockchain. În primul rând, sunt prezentate arhitectura sistemului și configurarea mediului de dezvoltare, iar apoi sunt prezentate componentele front-end și back-end. În a doua parte, se explică cum se utilizează tehnologiile specifice, cum ar fi Node.js, React și Web3.js. În cele din urmă, capitolul descrie gestionarea portofelelor digitale și modul în care acestea interacționează cu blockchain-ul Ethereum.

3.1 Planificare sistemului

Au fost stabilite cerințele funcționale și nefuncționale ale aplicației pentru a începe planificarea sistemului. Cerințe funcționale esențiale au fost identificate ca capacitatea utilizatorilor de a vedea statusul unui produs printr-un cod `QR`, capacitatea producătorilor de a adăuga produse noi, capacitatea verificatorilor de a valida produsele și de a include informații suplimentare.

Un pas important în planificarea sistemului a fost alegerea tehnologiilor. Pentru frontend, s-a ales `React` datorită flexibilității și performanței sale. Backend-ul folosește Node.js și Express.js pentru a gestiona autentificarea, solicitările HTTP și interacțiunile cu blockchain-ul. Pentru dezvoltare și testare, Hardhat este folosit pentru a implementa contractele inteligente pe o rețea de test a blockchain-ului Ethereum. Pentru a garanta securitatea autentificării producătorilor și verificatorilor, a fost implementat `JWT`. MetaMask a fost selectat pentru a supraveghea tranzacțiile și interacțiunea cu rețeaua blockchain.

Aplicația descentralizată este utilizată de mai multe entități. Producătorii, verificatorii și consumatorii sunt toți membri ai lanțului de aprovizionare. Producătorii realizează produsul și îl înregistrează în sistem, după care acesta primește un cod unic de identificare și este înregistrat pe blockchain cu ajutorul contractelor inteligente.

Verificatorii validează produsele și facilitează comunicarea dintre producători și consumatori, oferind acces la informații despre locația și starea produsului. Aceste entități inițiale sunt clasificate ca administratori la nivel de sistem al aplicației. Consumatorii sunt scopul principal al aplicației, deoarece fac parte din categoria utilizatorilor la nivel de sistem și au capacitatea de a verifica autenticitatea produsului și istoricul acestuia folosind un cod de identificare unic QR.

3.2 Configurarea mediului de dezvoltare

A fost ales Visual Studio Code[1] ca mediu de dezvoltare principal pentru a implementa aplicația DApp de urmărire a produselor pe blockchain. Visual Studio Code este un editor de cod robust și versatil care oferă dezvoltatorilor numeroase extensii utile pentru compilare, interpretare și design de text.

A fost instalat Node.js pentru a începe dezvoltarea aplicației web. Acesta oferă un runtime bun pentru a executa cod `JavaScript` pe server. A fost folosită biblioteca React pentru a crea frontend-ul aplicației.

Limbajul Solidity a fost folosit pentru a dezvolta contracte inteligente. În mod similar cu multe alte limbaje de nivel înalt, Solidity funcționează pe baza unei mașini virtuale. Toate contractele inteligente sunt executate în mașina virtuală Ethereum . Solidity este limbajul de programare folosit pentru a crea o aplicație descentralizată, Ethereum funcționează ca platformă de bază și EVM-urile execută tranzacții.

3.3 Dezvoltarea Interfeței

Dezvoltarea interfeței utilizator a fost esențială pentru a asigura o experiență de utilizare plăcută și eficientă. Procesul a început cu utilizarea de wireframes pentru a stabili modul în care vor fi structurate și aranjate paginile aplicației. În final, am folosit rest-router-dom, o librărie simplă de utilizat. Procesul a început cu configurarea inițială a aplicației în fișierul `App.js`, urmată de adăugarea de pagini și componente specifice pentru fiecare funcționalitate.

Utilizarea React pentru implementarea componentelor front-end a permis crearea de elemente și pagini de interfață reutilizabile. A fost folosit React Router pentru a naviga între pagini, iar CSS și librăriile Material-UI[2] și Bootstrap au fost folosite pentru a stiliza interfața.

Configurarea rutelor principale a inclus paginile de home, autentificare, pagini dedicate producătorilor și verificatorilor, pagina pentru scanarea codurilor QR și pagina

[1]Denumit și VS Code, este un editor de cod sursă dezvoltat de Microsoft pentru Windows, Linux, macOS și browsere web;

[2]Material Design este un limbaj de design dezvoltat de Google în 2014.

pentru vizualizarea detaliilor unui produs specific. Pagina principală home este punctul de intrare în aplicație, oferind utilizatorilor opțiunea de a naviga către pagina de autentificare sau alte funcționalități disponibile. Pagina de autentificare `Login.js` permite utilizatorilor să se autentifice în aplicație utilizând token-uri specifice rolului, fie ca producător, fie ca verificator.

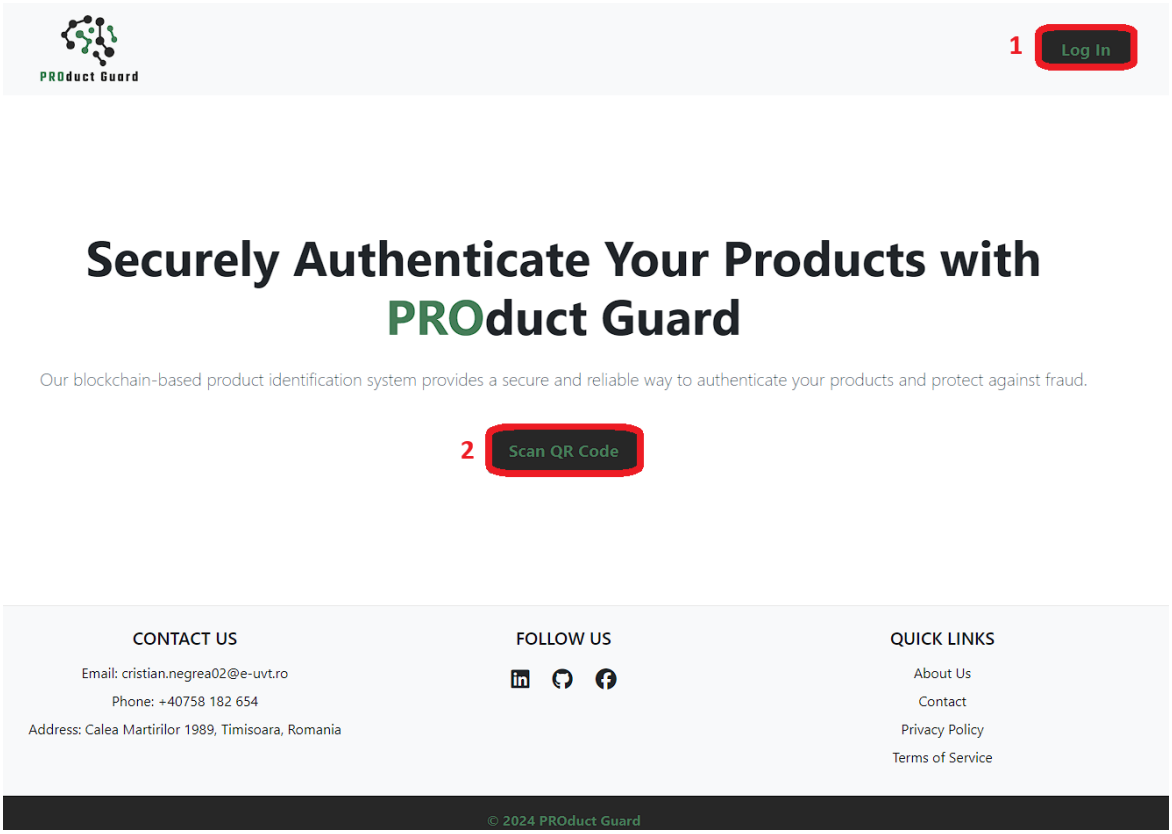

Figura 3.1:
Home Page
(1) Logare
(2) Scanare cod QR

Pentru producători, pagina `Deployer.js` include funcționalități pentru vizualizarea profilului `Profile.js`, adăugarea unui nou produs `AddProduct.js`, și vizualizarea tuturor produselor adăugate `AllProducts.js`. `Profile.js` permite producătorilor să vizualizeze detaliile profilului lor, `AddProduct.js` oferă un formular pentru adăugarea unui nou produs, inclusiv detalii precum nume, brand, numar identificare și încărcarea imaginii și a PDF-ului[3], iar `AllProducts.js` afișează o listă cu toate produsele adăugate de producător.

Pentru verificatori, pagina `Verifier.js` include funcționalități pentru vizualizarea profilului și verificarea produselor. Componenta `Verifier.js` permite verificatorilor să

[3]Este un format de fișier dezvoltat de Adobe în 1992 pentru a prezenta documentele, inclusiv formatarea textului și imaginile, într-o manieră independentă.

scaneze codurile QR ale produselor pentru a verifica detaliile acestora, iar `Product.js` afișează detaliile produsului scanat, inclusiv statusul verificării.

Toate aceste elemente au fost stilizate în fișierele CSS corespunzătoare. Aceste fișiere CSS au fost folosite pentru a da un aspect coerent și stilat interfeței în toate componentele. Acestea includ `Home.css` pentru stilurile paginii principale, `Login.css` pentru stilurile paginii de autentificare, `AddProduct.css` pentru formularul de adăugare a produsului, `AllProducts.css` pentru lista de produse.

3.4 Dezvoltarea Sistemului de Autentificare

La început, s-a ales o autentificare normală cu nume de utilizator și parolă, dar am decis să folosim tokenul web `JSON` pentru securitatea aplicației. Sistemul de autentificare este esențial pentru securitatea aplicației, deoarece permite doar utilizatorilor autorizați să adauge și să valideze produse. Pentru a autentifica utilizatorul, serverul creează un token JWT cu datele utilizatorului prin funcția `jwt.sign`, care sunt semnate digital pentru a evita falsificarea. Tokenul este generat local pe dispozitivul utilizatorului folosind această funcție și trimis înapoi la server de fiecare dată când utilizatorul solicită să-și verifice autentificarea prin endpoint-ul `/login`.

Serverul de autentificare primește cheile publice și folosește cheile private pentru a confirma token-urile folosind funcțtia `jwt.verify`. Atunci când validarea este finalizată, serverul creează un răspuns de autentificare prin endpoint-ul `/protected` și transmite token-ul `JWT` frontend-ului. În partea front-end, un mecanism de gestionare a cererilor adaugă automat token-ul JWT la fiecare cerere API. Acest mecanism garantează că toate cererile trimise către server sunt autentificate. Acest proces facilitează eficient accesul și expirarea token-urilor utilizatorilor, asigurând o autentificare sigură și integritate a proceselor de adăugare și validare a produselor.

3.5 Dezvoltarea Conexiunii cu MetaMask

MetaMask este o extensie de browser care ajută utilizatorii să interacționeze cu rețeaua blockchain, permite semnarea tranzacțiilor și gestionarea portofelelor de criptomonede. Utilizatorii trebuie să instaleze MetaMask în browserul lor înainte de a putea utiliza aplicația. MetaMask a fost configurat pentru a asigura un mediu sigur și controlat în timpul dezvoltării și testării Ethereum.

Utilizatorii pot accesa adresa lor publică și semna tranzacțiile prin conectarea portofelului MetaMask în aplicație. Utilizatorii semnează tranzacția direct din MetaMask la inițierea unei tranzacții, după care aceasta trimite tranzacția către blockchain. Aplicația actualizează starea pe baza tranzacțiilor pe blockchain și a răspunsurilor de

la MetaMask. Biblioteca Web3.js a fost utilizată pentru a facilita comunicarea. Aceste biblioteci permit transmiterea tranzacțiilor către blockchain și gestionarea securizată și eficientă a interacțiunilor cu MetaMask.

Implementarea conexiunii cu MetaMask a început cu definirea funcției de conectare `connectMetaMask`, care solicită accesul la conturile MetaMask ale utilizatorilor. Această funcție folosește API-ul MetaMask pentru a obține adresa publică a utilizatorului. În cazul în care MetaMask nu este instalat, utilizatorului i se va afișa un mesaj de eroare.Atât componenta React `Deployer` cât și `Verifier` oferă opțiuni pentru conectarea la MetaMask.

```
export const connectMetaMask = async () => {
  if (window.ethereum) {
    try {
      const accounts = await window.ethereum.request
      ({method: 'eth_requestAccounts' });

      return accounts[0];
    } catch (error) {
      throw new Error('MetaMask connection failed');
    }
  } else {
    throw new Error('MetaMask is not installed');
  }
};
```

Listing 3.1: Funcția connectMetaMask

3.6 Dezvoltare Server

Backend-ul este compus din mai multe părți esențiale care garantează funcționarea corespunzătoare și eficientă a aplicației. În primul rând, Express.js controlează serverul principal. Aceasta primește toate cererile HTTP de la client și răspunde cu datele solicitate sau, după caz, cu mesaje de eroare. Middleware-urile CORS și body-parser sunt utilizate pentru a permite cereri cross-origin și pentru a parsa corpul cererilor HTTP în format JSON.

Administrarea fișierelor încărcate este o componentă esențială a backend-ului. În acest scop, biblioteca Multer a fost utilizată pentru a gestiona încărcările de fișiere, inclusiv imagini și fișiere PDF legate de produse.

Autentificare JWT Sistemul de autentificare al utilizatorilor este construit pe baza token-urilor JWT. La autentificare, serverul generează un token JWT care conține informații despre utilizator, semnat digital pentru a preveni falsificarea. Token-

ul este trimis la client și stocat local. La fiecare cerere ulterioară către server, token-ul este trimis pentru a verifica autentificarea utilizatorului.

Gestionarea Produselor Endpoint-urile din backend permit adăugarea de noi produse, colectarea de informații despre un anumit produs și colectarea tuturor produselor. Un fișier JSON conține datele produsului, inclusiv fișierele PDF și imaginile. Pentru a menține transparența și imutabilitatea, un număr mic de date, cum ar fi numele, numele și mărcile, sunt înscrise pe blockchain. Dacă un produs este adăugat, serverul colectează informații despre produs, precum și fișierele încărcate, creează un URL pentru produs și generează un cod QR asociat.

Generarea și Verificarea Codurilor QR Pentru fiecare produs adăugat, backend-ul generează un cod QR care conține un URL unic pentru produs. Codul QR este generat utilizând biblioteca `qrcode`și este încorporat în datele produsului.

Verificarea Produselor La primirea unei cereri de verificare, serverul actualizează starea produsului în fișierul JSON și confirmă verificarea printr-un răspuns către client. Acest proces asigură că doar verificatorii autorizați pot schimba starea produselor și că toate schimbările sunt înregistrate și pot fi verificate.

3.7 Inregistrarea tranzacților

MetaMask cere semnătura utilizatorului atunci când dorește să efectueze o acțiune care implică o tranzacție pe blockchain. Acest lucru se întâmplă atunci când utilizatorul dorește să valideze tranzacția. De exemplu, utilizatorul trebuie să semneze o tranzacție pentru a înregistra un produs.

Web3.js este o bibliotecă JavaScript care folosește HTTP pentru a interacționa cu un nod Ethereum local sau de la distanță. Configurarea Web3.js include inițializarea contractului inteligent și conectarea la MetaMask.

Un standard cunoscut sub numele de ABI[4] descrie modul în care datele și funcțiile contractelor inteligente pot fi accesate și utilizate. Semnăturile funcțiilor, structurile parametrilor și tipurile de date utilizate sunt toate reglementate de ABI. Practic, ABI explică cum contractele inteligente comunică cu lumea exterioară, ceea ce permite aplicațiilor front-end să interacționeze cu ele.

[4]Interfață binară de aplicație este o interfață între două module binare de program.

Capitolul 4

Testare

Pentru frontend, testarea a fost făcută utilizând React Testing Library pentru a crea și executa teste unitare și de integrare. Aceste teste au verificat funcționalitatea componentelor React, asigurându-ne că funcționează bine în diferite scenarii de utilizare.

La nivel de backend, testele unitare au fost scrise folosind Mocha și Chai[1], asigurând că funcțiile serverului Express.js funcționează corect și că token-urile JWT sunt generate și validate corespunzător. De asemenea, au fost realizate teste de integrare pentru a verifica interacțiunea dintre server și contractele inteligente pe blockchain, folosind `Hardhat` și `Web3.js`.

Framework-ul Hardhat a fost folosit pentru a realiza testarea contractelor inteligente, scripturile de testare au simulat diferite scenarii de utilizare pentru a verifica corectitudinea logicii contractelor. Aceste teste au garantat că funcțiile de înregistrare și verificare a produselor pe blockchain funcționează corect.

Testarea manuală a procesului de autentificare s-a asigurat că utilizatorii pot introduce token-urile lor și vor primi un token JWT valid după autentificare. Acest token va fi stocat local și utilizat pentru cererile viitoare. În plus, s-a verificat că cererile protejate sunt accesibile doar utilizatorilor autentificați și că sunt gestionate corect token-urile JWT expirate.

Următorul pas a fost verificarea integrării și a conectării cu MetaMask. Au fost oferite instrucțiuni utilizatorilor pentru a instala extensia MetaMask în browserul lor, a înființa un cont și a se conecta la rețelele de testare `Ethereum`. Conectarea portofelului MetaMask la aplicație permite utilizatorilor să acceseze adresa lor publică și să semneze tranzacții direct din MetaMask.

S-a testat că utilizatorii pot completa formularul cu nume, marcă, număr serial, actor și locație pentru funcționalitatea de înregistrare a produselor. S-a verificat că datele către contractul inteligent pe blockchain sunt trimise corect și că produsul este

[1]Mocha și Chai cadru de testare JavaScript pentru programele Node.js

înregistrat fără erori.

De asemenea, au fost efectuate teste manuale pentru a verifica funcționalitatea de căutare și vizualizare a produselor. A fost testată interfața pentru a asigura că utilizatorii pot căuta un produs după scanarea codului QR și că toate informațiile relevante despre produs, inclusiv istoricul său, sunt afișate corect.

A fost necesară colectarea de informații despre mai multe tranzacții de adăugare și verificare a produselor pentru a realiza o analiză amănunțită a costurilor de tranzacție în timpul testării pe `MetaMask`. În cadrul testării s-a observat diferența costurilor în funcție de tipul de tranzacție efectuată. Spre exemplu, costul pentru adăugarea unui produs diferă considerabil față de costul verificării acestuia. Cu toate acestea, vizualizarea datelor nu implică niciun cost, deoarece este o operațiune de citire pe blockchain care nu necesită taxe.

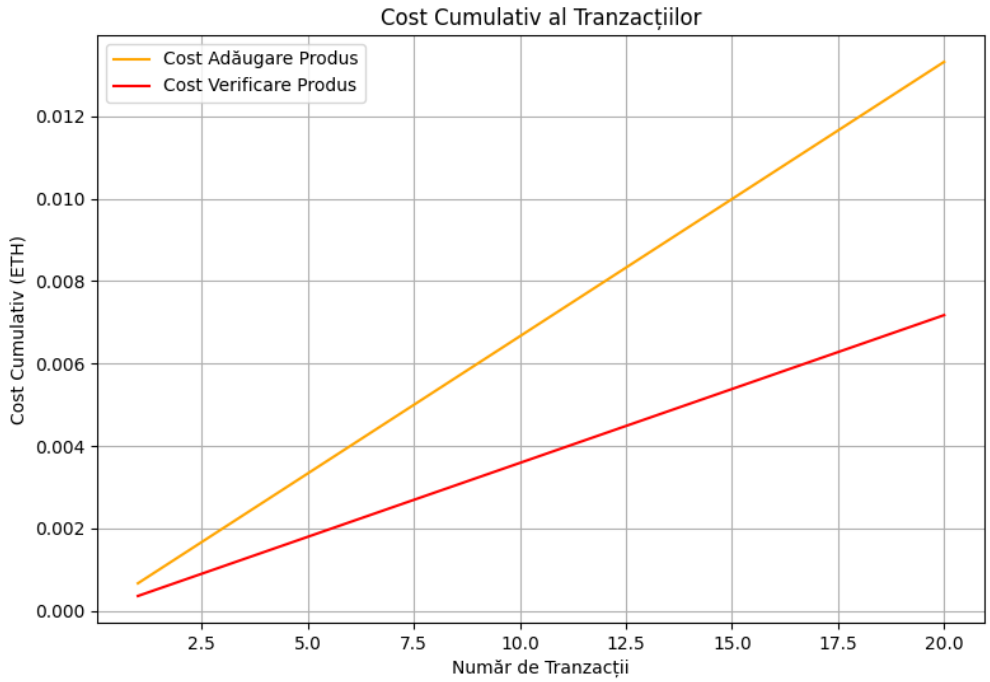

Figura 4.1:
Adăugarea și Verificarea Produselor

Analizarea costurilor tranzacțiilor de adăugare și verificare poate oferi o imagine clară a resurselor necesare pentru a menține funcționalitatea aplicației și pentru a asigura transparența produselor pe blockchain. Costurile tranzacțiilor pot varia, motivul fiind legat de fluctuațiile de taxă de gaz necesară pentru a procesa tranzacțiile pe rețea. Taxa este determinată de cererea și oferta de capacitate de procesare a tranzacțiilor la un moment dat. Taxele de gaz cresc atunci când rețeaua este aglomerată, iar în perioadele de activitate redusă pe rețea acestea scad.

În contextul scalabilității, când există un număr mare de tranzacții, blockchain-ul Ethereum poate întâmpina probleme de scalabilitate. Acest lucru se datorează

faptului că minerii trebuie să adauge fiecare tranzacție la blockchain, ceea ce poate provoca creșterea costurilor și congestionarea rețelei.

Pentru o prezentare a costurilor de tranzacție pentru adăugarea și verificarea a 10 produse și 20 de verificari au fost colectate date specifice. Costul pentru adăugarea unui produs este de `0.00066602 ETH`, iar costul pentru verificarea unui produs este de `0.0003587 ETH`.

Graficul costurilor de adăugare a produselor ilustrează creșterea liniară a costurilor în funcție de numărul de produese adăugate, cu un cost fix per tranzacție de `0.00066602 ETH`.Pentru adăugarea a 10 produse neverificate costurile de tranzacție ajung la `0.0066602 ETH` (vezi figura4.1).

De asemenea în graficul de verificare a produselor, se poate observa creșterea costurilor în funcție de numărul de produse verificate, cu un cost fix per tranzactie de `0.0003587 ETH`.Pentru un singur produs ,verificat de 20 de ori , costul total este de `0.007174 ETH` (vezi figura 4.1).

În final analiza pentru un cost de tranzacție fix oferă o imagine clară asupra costurilor implicate în gestionarea produselor prin intermediul blockchain-ului Ethereum, costurile totale pentru 10 produse adăugate ce au fost verificate de 2 de ori fiecare ajungând la `0.0138342 ETH` (vezi figura 4.1).

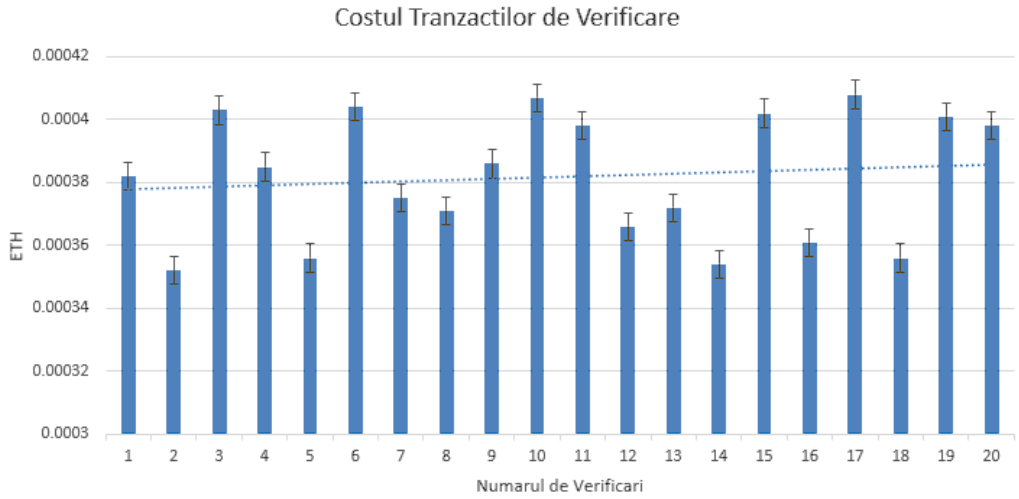

Figura 4.2:
Verificarea Produselor

În diagrama de mai sus se prezintă costul tranzacțiilor de verificare a produselor in ETH, cu o marjă de eroare de 20%. Volatilitatea în costuri este influențată de fluctuațiile taxelor de gaz pe rețeaua Ethereum, care pot varia în funcție de congestia rețelei și de complexitatea tranzacțiilor efectuate. Această variabilitate întărește necesitatea unei gestionări atente a resurselor (vezi figura 4.2).

Utilizând tehnologia blockchain, acest proiect prezintă o soluție sigură și transparentă pentru urmărirea originii și autenticității produselor. Implementarea unui sistem

bazat pe blockchain pentru trasabilitatea produselor aduce numeroase avantaje.

Fiecare mișcare și fiecare schimbare de proprietate a unui produs poate fi înregistrată într-un registru public, accesibil tuturor părților interesate. Acesta nu numai că permite verificarea autenticității și originii produselor, dar asigură și securitate părților implicate. Pe lângă toate acestea, utilizarea contractelor inteligente îmbunătățește eficiența operațională și reduce erorile umane prin automatizarea proceselor.

Cu toate acestea, utilizarea blockchain-ului pentru trasabilitatea produselor prezintă și câteva dezavantaje. Costurile inițiale ridicate de implementare și complexitatea tehnologică pot fi niște obstacole. În plus, problemele de scalabilitate ale blockchain-urilor publice pot împiedica utilizarea pe scară largă a tehnologiei. În ciuda acestor probleme, utilizarea blockchain-ului în trasabilitatea produselor are avantaje semnificative, iar continuarea cercetării și dezvoltării în acest domeniu este esențială.

Concluzii

În cadrul acestei lucrări, am abordat problema gestionării lanțului de aprovizionare și urmărirea produselor folosind tehnologia blockchain. Problema este foarte importantă într-o lume globalizată, unde trasabilitatea și transparența produselor sunt esențiale pentru a garanta calitatea și unicitatea acestora.

Lucrarea ilustrează situația actuală în gestionarea lanțului de aprovizionare și prezintă exemple de soluții inovatoare ce sunt necesare pentru a crește eficiența și transparența produselor. Așadar, utilizarea tehnologiilor blockchain oferă un registru imutabil și transparent pentru toate tranzacțiile. În acest sens, lucrarea prezintă utilizarea acestor tehnologii pentru a crea o aplicație web descentralizată, care să permită urmărirea produselor de la producător la consumator.

Soluția propusă prezintă numeroase avantaje, printre care se numără și transparența și securitatea oferite de blockchain, precum și eficiența crescută datorată automatizării proceselor de urmărire și verificare. Datorită criptării și protecției împotriva modificărilor neautorizate, datele stocate în blockchain sunt mai sigure. Această soluție rezolvă problema autenticității produselor și oferă siguranță tuturor actorilor participanți în lanțul de aprovizionare.

Pe de altă parte, soluția prezintă și câteva dezavantaje. Implementarea și supravegherea unui sistem bazat pe blockchain poate fi dificilă pentru echipele care nu au experiență în această tehnologie. Organizațiile cu resurse limitate ar putea întâmpina dificultăți dacă ar trebui să facă investiții mari în dezvoltarea și implementarea tehnologiei blockchain. În plus, tehnologia blockchain poate întâmpina dificultăți în ceea ce privește adoptarea pe scară largă și integrarea cu sistemele existente.

Cu toate acestea, utilizarea tehnologiei blockchain reprezintă o soluție sigură și transparentă pentru urmărirea originii și autenticității produselor. Implementarea unui sistem bazat pe blockchain pentru trasabilitatea produselor aduce numeroase avantaje. Implementarea unor blockchain-uri private pentru fiecare companie ar putea oferi mai mult control asupra datelor în viitor. Aceste lanțuri private ar putea fi integrate cu lanțurile publice, asigurând confidențialitatea și transparența acestora.

Bibliografie

[AFC+19] Francesca Antonucci, Simone Figorilli, Corrado Costa, Federico Pallottino, Luciano Raso, and Paolo Menesatti. A review on blockchain applications in the agri-food sector. *J. Sci. Food Agric.*, 2019.

[AW18] Andreas M. Antonopoulos and Gavin Wood. *Mastering Ethereum: Building Smart Contracts and DApps*. O'Reilly Media, Inc., 2018.

[But14] Vitalik Buterin. A next-generation smart contract and decentralized application platform. *white paper*, 2014.

[CD05] Colby Ronald Chiles and Marguarette Thi Dau. *An analysis of current supply chain best practices in the retail industry*. PhD thesis, MIT, 2005.

[CDP19] Fran Casino, Thomas K. Dasaklis, and Constantinos Patsakis. A systematic literature review of blockchain-based applications. *Telematics Inform.*, 2019.

[CDS+20] Eloi Chazelas, Mélanie Deschasaux, Bernard Srour, Emmanuelle Kesse-Guyot, Chantal Julia, Benjamin Alles, et al. Food additives in 126,000 food products of the french market. *Sci. Rep.*, 2020.

[Fou24] Ethereum Foundation. Ethereum: The world computer. `https://ethereum.org/en/`, 2024.

[GPJ07] Jerry Zeyu Gao, Lekshmi Prakash, and Rajini Jagatesan. Understanding 2d-barcode technology and applications in m-commerce. In *31st Annu. Int. Comput. Softw. Appl. Conf. (COMPSAC 2007)*. IEEE, 2007.

[GRB19] Tharic Pires Dias Galuchi, Fabricio Pini Rosales, and Mario Otavio Batalha. Management of socioenvironmental factors of reputational risk in the beef supply chain. *Int. Food Agribus. Manag. Rev.*, 2019.

[Lab24] Nomic Labs. Hardhat: Ethereum development environment. `https://hardhat.org/`, 2024.

[LC00] Douglas M Lambert and Martha C Cooper. Issues in supply chain management. *Industrial marketing management*, 2000.

[(MD24] Mozilla Developer Network (MDN). Javascript guide. `https://developer.mozilla.org/en-US/docs/Web/JavaScript`, 2024.

[Met24] MetaMask. Metamask: A crypto wallet & gateway to blockchain apps. `https://docs.metamask.io/`, 2024.

[MMT22] Manuel Adelin Manolache, Sergiu Manolache, and Nicolae Tapus. Decision making using the blockchain proof of authority consensus. *Procedia Comput. Sci.*, 199, 2022.

[Mou16] William Mougayar. *The Business Blockchain: Promise, Practice, and the Application of the Next Internet Technology*. Wiley, 2016.

[Nak08] Satoshi Nakamoto. Bitcoin: A peer-to-peer electronic cash system, 2008.

[Nod24] Node.js. Node.js: Javascript runtime built on chrome's v8 javascript engine. `https://nodejs.org/en/`, 2024.

[Rea24] React. React: A javascript library for building user interfaces. `https://reactjs.org/`, 2024.

[Sta14] Hartmut Stadtler. Supply chain management: An overview. *Supply chain management and advanced planning: Concepts, models, software, and case studies*, 2014.

[Tia16] Feng Tian. An agri-food supply chain traceability system for china based on rfid & blockchain technology. In *2016 13th Int. Conf. on Service Systems and Service Management (ICSSSM)*. IEEE, 2016.

[TT16] Don Tapscott and Alex Tapscott. *Blockchain Revolution: How the Technology Behind Bitcoin Is Changing Money, Business, and the World*. Penguin, 2016.

[VeC24] VeChain. A blockchain platform designed to enhance supply chain management and business processes. `https://www.vechain.org`, 2024.

[Woo14] Gavin Wood. Ethereum: A secure decentralised generalised transaction ledger. In *Ethereum Project Yellow Paper*, 2014.