

ИМПЛЕМЕНТАЦИЈА ETHEREUM БАЗИРАНЕ МУЗИЧКЕ ПЛАТФОРМЕ

IMPLEMENTATION OF ETHEREUM-BASED MUSIC PLATFORM

Силвија Тепшић, Факултет техничких наука, Нови Сад

Област – ЕЛЕКТРОТЕХНИКА И РАЧУНАРСТВО

Кратак садржај – Задатак рада представља развој децентрализоване апликације за куповину, продају и репродукцију музике употребом *Ethereum* платформе. Циљ апликације је да демонстрира употребу *blockchain* базираних платформи која може да замени корпоративне посреднике

Кључне речи: *blockchain*, *Ethereum*, *dApp*, *Solidity*, музика

Abstract – The thesis deals with the development of a decentralized *Ethereum*-based application for purchasing, selling and playing music. The goal of the application is to demonstrate the use of a *blockchain*-based platform that can replace corporate intermediaries.

Keywords: *blockchain*, *Ethereum*, *dApp*, *Solidity*, music

1. УВОД

У савременом свету корисници који желе да слушају музику и при том подрже извођаче углавном користе музичке платформе попут *iTunes*-а или *Amazon Music*-а. Да би се песма објавила на таквим платформама, она мора да прође кроз низ провера и мора да буде одобрена од стране таквих платформи. Издавачи такође морају и да верују платформама да су њихове песме складиштене на безбедном месту. Још једна мана таквих платформи је што узимају не тако мали проценат од продаје песама. Са циљем да се избегне потреба за коришћење посредничке платформе са комплексним процедурама и високим накнадама, развијена је музичка платформа *SongCloud*.

SongCloud представља децентрализовану апликацију која се ослања на *Ethereum* платформу и омогућава корисницима да издају, купују и репродукују музику. За потребе извршавања трансакција на платформама уводи и нову валуту - капљице.

2. BLOCKCHAIN

Да би се разумело функционисање *blockchain*-а, неопходно је да се разуме појам дистрибуираног система.

Дистрибуирани систем представља рачунарску парадигму где два или више чворова координисано сарађују како би постигли заједнички циљ. Моделован је тако да га крајњи корисници виде као

јединствену логичку платформу [1].

Чвор представља индивидуалног члана (рачунар) у дистрибуираном систему. Чворови садрже меморију и процесор и могу међусобно да комуницирају путем порука. Чвор може да буде искрен, неисправан или злонамеран, као и да се понаша произвољно. У том случају се он зове византијски чвор. Назив потиче од приче о проблему византијских генерала [1].

Дизајн дистрибуираних система је поприлично изазован задатак јер је неопходно да се реше проблеми координације између чворова и отпорности на грешку, односно да чак и ако неки од чворова постану неисправни или се прекину линије комуникације, дистрибуирани систем би требао да настави успешно да функционише. Међутим, доказано је да дистрибуирани систем не може истовремено да поседује све три жељене особине: конзистентност, доступност и отпорност на партиције [1].

Blockchain представља дистрибуирану базу која постоји истовремено на више рачунара. Константно расте додавањем нових група забелешки односно блокова на њу. Сваки блок садржи временску ознаку и линк ка претходном блоку, тако да блокови заправо формирају ланац [2].

Кључна одлика *blockchain*-а је да је децентрализован, што значи да га не контролише један ентитет попут банке или владе. Уместо тога, одржава га мрежа рачунара који раде заједно да би валидирани и забележили трансакције. Овај приступ отежава модификацију података на *blockchain*-у од стране појединца, јер би то значило да тај појединац мора да убеди већину рачунара на мрежи да прихвате његове измене [3].

Постоји више различитих врста *blockchain*-а, укључујући јавни, приватни, конзорцијум и хибридни *blockchain*. Јавни *blockchain* је доступан свима и децентрализован, што значи да није контролисан од стране једног ентитета. Приватни *blockchain* је доступан само одређеној групи појединаца. Обично га користе организације које желе да им подаци буду приватни и безбедни. Конзорцијум *blockchain* је делимично децентрализован, пошто га контролише група организација, а не један ентитет. Обично се користи у индустријама где више странака треба да сарађују, попут финансијске индустрије. Хибридни *blockchain* комбинује аспекте јавног и приватног *blockchain*-а. Може да буде доступан јавно, али и да има неке елементе којима само одређени појединци или групе имају приступ [3].

НАПОМЕНА:

Овај рад проистекао је из мастер рада чији ментор је био др Жељко Вуковић, доцент.

Промене протокола *blockchain* мреже и структура података се називају рачвања (енгл. *forks*). Могу да се поделе у две категорије: *soft forks* и *hard forks* [5]. *Soft fork* представља промену у *blockchain* имплементацији која је компатибилна са претходним верзијама. *Hard fork* представља промену у *blockchain* имплементацији која није компатибилна са претходним верзијама.

3. ETHEREUM

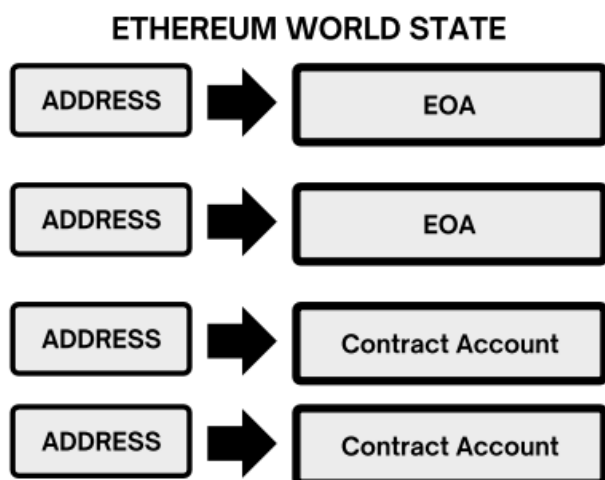
Ethereum представља децентрализовану *Peer-to-Peer* мрежу *Ethereum* клијената, који су покренути на чворовима. *Ethereum* клијент представља софтвер који може да верификује нове трансакције, извршава паметне уговоре и обрађује нове блокове ланца. Криптовалута која се користи на мрежи за плаћање трансакција и рачунарске обраде се назива *ether*, а званичан симбол јој је *ETH* [4].

3.1. Налози

Када корисник покрене *Ethereum* софтвер додељује му се идентитет у виду *Ethereum* налога са одређеним псеудонимом. Постоји и још једна врста налога, а то су налози уговора [6]. Налози у спољном власништву (енгл. *Externally Owned Accounts - EOA*) су кориснички налози који могу да креирају трансакције и којима се управља помоћу пара кључева, које *Ethereum* софтвер изгенерише. Налози уговора (енгл. *Contract accounts - CA*) представљају паметне уговоре који су испоручени на *Ethereum* и контролисани од стране њиховог интерног кода.

Стање *Ethereum*-а (енгл. *world state*) представља базу података која мапира *Ethereum* адресе на налоге у спољном власништву и налоге уговора. Уколико се било шта ново деси, стање *Ethereum*-а мора да се ажурира. Било да *EOA* пребацује средства другом *EOA* или интерагује са налогом уговора, долази до промене стања. Стање *Ethereum*-а се мења на сваких 12 секунди на основу свих трансакција које су корисници иницирали [6].

На слици 1 је дат приказ стања *Ethereum*-а.



Слика 1. *Ethereum* стање [6]

3.2. Паметни уговори и *Solidity*

Паметни уговори представљају непроменљиве рачунарске програме који се извршавају детерминистички у контексту *Ethereum* виртуелне

машине као део *Ethereum* мрежног протокола, односно на децентрализованом *Ethereum* светском рачунару [7].

Децентрализована апликација (енгл. *Decentralised Application - dApp*) представља комбинацију паметног уговора са технологијама које нису део *blockchain*-а, како би се унапредило искуство корисника. Обично подразумева да постоји клијентски део у форми традиционалног веб сајта [6].

Solidity [8] је најпопуларнији програмски језик за писање паметних уговора. Он спада у статички типизирани језике, што значи да је потребно експлицитно дефинисати све типове података пре коришћења.

3.3. Трансакције

Постоје два главна типа трансакција у оквиру *Ethereum*-а. Први је позив поруке (енгл. *Message Call*). Позиви поруке креирани од стране *EOAs* подразумевају промену *Ethereum* стања. Други тип трансакције јесте креирање уговора (енгл. *Contract Creation*), који представља захтев да се дода нови налог уговора у *Ethereum* стање [6].

Да би се трансакција реализовала у оквиру *Ethereum*-а, неопходно је да се плати одговарајућа накнада. За разумевање накнада за трансакције, неопходно је да се разуме разлика између цене гаса и накнаде за гас. у оквиру *Ethereum*-а циклуси обраде се изражавају помоћу мерне јединице која се назива гас. Гасни трошкови нису повезани са неком валутом, већ искључиво са бројем циклуса рачунарске обраде [6].

Свака јединица гаса која се потроши ће да резултује додатном накнадом за гас. Накнада за гас се изражава помоћу гвеја (енгл. *gas*), а један гвеј вреди 0.0000000001 *ETH*. Накнада за гас се састоји из два дела: основне накнаде (енгл. *base fee*), која се алгоритамски рачуна на основу вредности гвеја, као и накнаде за приоритет (енгл. *priority fee*), која представља додатни износ који корисник може да плати како би његова трансакција била брже прихваћена [6].

3.4. Консензусни слој

Консензусни механизам који се користи у оквиру *Ethereum*-а се зове доказ улога (енгл. *Proof of Stake - PoS*). Заснива се на постојању валидатора чија су задужења валидирање трансакција и креирање блокова. Да би корисник постао валидатор, неопходно је да уложи 32 *ETH* на депозитни уговор као улог и да користи потпуни чвор у комбинацији са клијентом за валидацију. Постојање улога је битно како би се спречили Сибил напади [6].

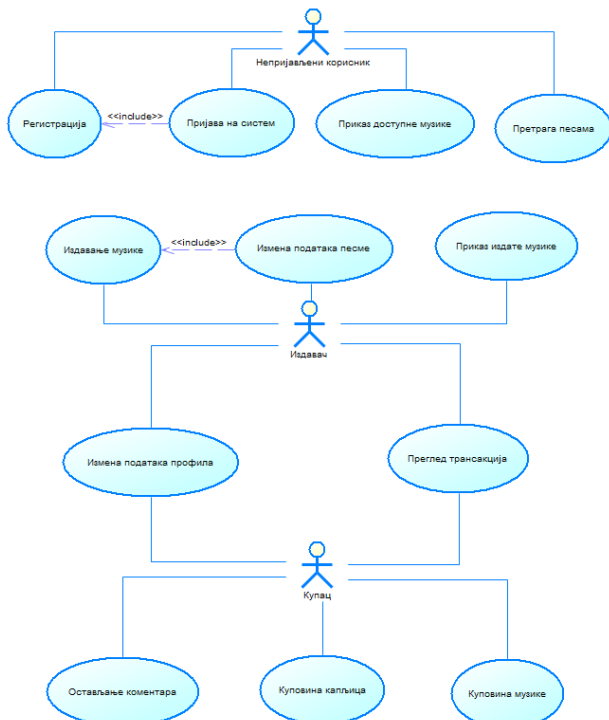
3.5. *IPFS*

IPFS (*Inter-Planetary File System*) представља децентрализовани складиштни систем са адресираним садржајем који дистрибуира сачуване објекте између учесника у *P2P* мрежи. Адресирани садржај подразумева да је сваки део садржаја (датотека) хеширан и да се та хеш вредност користи за идентификовање те датотеке. Могуће је добити било коју датотеку од било ког *IPFS* чвора захтевањем њеног хеша [7].

4. СПЕЦИФИКАЦИЈА

4.1. Спецификација захтева

На слици 2 је приказан дијаграм случајева коришћења. Он представља графичку репрезентацију корисника система (актора) и њихове могуће интеракције са системом. Као што се може видети на дијаграму, апликацију могу користити три типа корисника: непријављени корисник, издавач и купац.



Слика 2. Дијаграм случајева коришћења

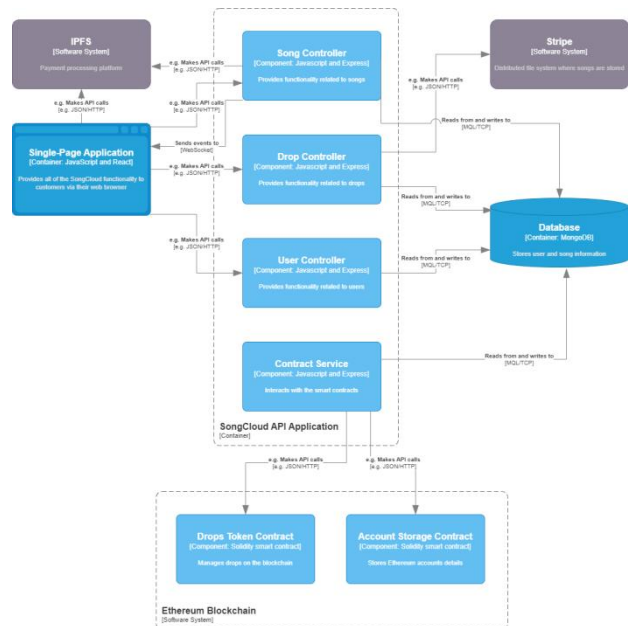
4.2. Спецификација система

Архитектура система *SongCloud* платформе се састоји од једностраничне апликације (енгл. *Single-page application*) са клијентске стране, *SongCloud API* апликације и паметних уговора са серверске стране, као и базе за трајно складиштење података.

Клијентска апликација омогућава корисницима интеракцију са системом и директно приступа датотекама песама на *IPFS*, што омогућава корисницима да их одслушају у претраживачу.

Главне делове *SongCloud API* апликације чине контролери, као и сервис који омогућава интеракцију са паметним уговорима – *Contract Service*. Основна улога контролера је да прихватају захтеве од клијентске апликације и да врате одговор који садржи информације о траженом ресурсу. *Drop Controller* врши и интеракцију са процесором плаћања *Stripe API*-ом, који омогућава реализацију трансакција у еврима. Посебну улогу има *Song Controller* који може са серверске стране да иницира комуникацију са клијентом употребом *Web Socket* технологије. Поред тога *Song Controller* је задужен и да иницира слање датотека песама на *IPFS*. Ради чувања података о корисницима и песмама, *SongCloud API* апликација интерагује са базом података.

На слици 3 је приказана архитектура система *SongCloud* платформе.



Слика 3. Дијаграм компоненти

5. ИМПЛЕМЕНТАЦИЈА

5.1. Паметни уговори

У оквиру развоја система креирана су и коришћена два паметна уговора: *AccountStorage.sol* и *DropsToken.sol*.

AccountStorage.sol се користи превасходно за управљање корисничким идентитетом и власништвом. Он врши мапирање *Ethereum* адреса налога корисника на њихове *email* адресе, као и мапирање адреса налога корисника на песме које они поседује односно на хеш песме и њену цену.

DropsToken.sol се користи превасходно за управљање капљицама на платформи. *DropsToken.sol* уговор имплементира *ERC20* стандард, који представља протокол за имплементацију паметног уговора који ће се користити као токен на *Ethereum* мрежи. Он прописује одређене функционалности које паметни уговор мора да садржи. Овај уговор садржи и мапирање *Ethereum* адреса налога корисника на количину токена који тај налог поседује. На овај начин је онемогућена неовлашћена измена финансијског стања на рачуну корисника.

5.2. *SongCloud API* апликација

SongCloud API апликација представља главни серверски део платформе. Она комуницира са паметним уговорима испорученим на *Ethereum* мрежу, комуницира са базом података, као и са клијентским делом платформе.

Позиви ка паметним уговорима се реализују у *Contract Service*-у. У оквиру овог сервиса имплементирани су функционалности добављања и испоруке паметних уговора, креирања *Ethereum* налога корисника, функционалности за управљање капљицама, као и функционалности за управљање подацима о песмама на *blockchain*-у.

У оквиру *SongCloud API* апликације постоје три контролера: *User Controller*, *Drop Controller* и *Song*

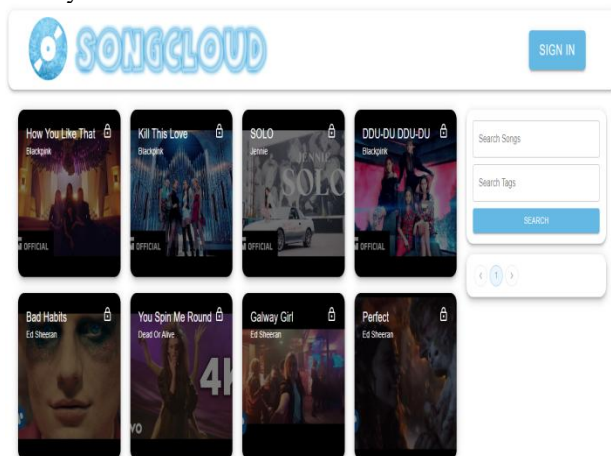
Controller. Преко њих се врши комуникација са клијентом, а уколико је за реализацију њихових функционалности неопходна комуникација са *blockchain* –ом, у њима се позива и *Contract Service*.

User Controller је задужен да реализује креирање налога корисника на платформи, пријаву корисника на платформу, добављање и измену корисничких података. У оквиру *Drop Controller*-а обавља се куповина капљица и добављање корисникових трансакција на платформи. У оквиру *Song Controller*-а извршавају се функционалности објаве, добављања, измене, куповине, коментарисања и претраге песама.

6. ДЕМОНСТРАЦИЈА

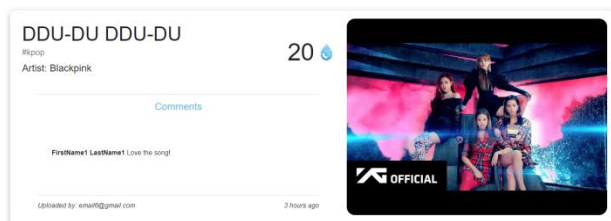
У овом поглављу описан је сценарио коришћења *SongCloud* апликације од стране непријављеног корисника.

Када корисник приступи *SongCloud* апликацији, прикаже му се почетна страница (слика 4) са свим доступним песмама на платформи. На њој се налази и навигациони бар, који омогућава кориснику да се креће кроз апликацију, као и форма за претрагу, помоћу које корисник може да претражује песме на основу назива и ознака песме.



Слика 4. Почетна страница

Притиском на жељену песму из листе песама, отвара се страница која садржи додатне информације о песми. Осим назива и извођача песме, видљиви су и подаци о томе кад је песма издата и од стране кога, цена песме у изражена у капљицама, ознаке песме, као и коментари о песми, уколико постоје. На слици 5 се може видети како изгледа детаљнији приказ песме.



Слика 5. Приказ песме

7. ЗАКЉУЧАК

Пројекат представља музичку платформу која се извршава на *Ethereum* мрежи. Тренутно имплементирани функционалности апликације су

издавање, куповина и репродукција музике, остављање коментара о песмама у реалном времену, измена информација о издатим песмама, куповина капљица помоћу картице, преглед трансакција, креирање, приказ и измена профила корисника.

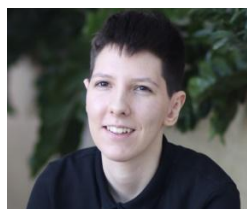
Пројекат је замишљен и као прототип, који треба да покаже могућности *blockchain* технологије, односно да је могуће променити тржиште музичке индустрије избацивањем корпоративних посредника који узимају велики проценат од продаје песама, а имају и потпуну контролу над датотекама и подацима корисника којима такође могу да тргују.

За имплементацију система коришћене су модерне технологије, што представља добру основу за даљи развој. Могућ је развој у правцу повећања децентрализованост апликације, отклањања ограничења на постојећим функционалностима или развојем нових.

8. ЛИТЕРАТУРА

- [1] I. Bashir, "Mastering Blockchain", 2023
- [2] A. Banafa, "Introduction to Blockchain Technology", 2023
- [3] V. Krishnan, "The Essential Guide to Web3", 2023
- [4] X. Wu, Z. Zou, D. Song, "Learn Ethereum", 2023
- [5] D. Yaga, P. Mell, N. Roby, K. Scarfone, "Blockchain Technology Overview", 2018
- [6] P. Dylan-Ennis, "Absolute Essentials of Ethereum", 2024
- [7] A. M. Antonopoulos, Dr. G. Wood, "Mastering Ethereum", 2019

Кратка биографија:



Силвија Тепшић рођена је у Руми 1999. године. Дипломира на Факултету техничких наука 2022. године, након чега исте године уписује мастер академске студије на студијском програму Рачунарство и аутоматика, смер Електронско пословање.