



VILNIAUS UNIVERSITETAS
MATEMATIKOS IR INFORMATIKOS FAKULTETAS
STUDIJŲ PROGRAMA: INFORMACINĖS TECHNOLOGIJOS

Bakalauro baigiamasis darbas

Kriptovaliutų mokesčių deklaravimo palengvinimo sistema
Cryptocurrency tax declaration facilitation system

Atliko:

Julijus Rancevas

VU el. p.: julijus.rancevas@mif.stud.vu.lt

Vadovas:

dr. Andrius Vytautas Misiukas

Misiūnas

Vilnius
2024

Turinys

Santrauka	4
Summary	5
Ivydas	6
1. Panašių sistemų analizė	9
1.1. Sistemos kriptovaliutų mokesčių deklaravimo palengvinimui	9
1.2. Koinly	9
1.3. Taxbit	9
1.4. Cointracker	10
1.5. Sistemų kriptovaliutų mokesčių deklaravimo palengvinimui palyginimas	10
2. Sistemos reikalavimai	11
2.1. Pajamų gautų vykdant kriptovaliutų sandorius mokestiniai reikalavimai Lietuvoje .	11
2.1.1. Pajamų gautų vykdant kriptovaliutų sandorius mokesčių apžvalga Lietuvoje	11
2.1.2. Pajamų gautų iš pavienių kriptovaliutų sandorių mokesčių deklaravimas Lietuvoje	12
2.1.3. Pajamų gautų vykdant kriptovaliutų sandorius su individualia veikla mokesčių deklaravimas Lietuvoje	12
2.1.4. Apmokestinamos ir neapmokestinamos kriptovaliutų operacijos Lietuvoje .	14
2.2. Palaikomos kriptovaliutų keityklos ir jų operacijos	14
2.3. Kriptovaliutų transakcijų apskaitos reikalavimai	15
2.4. Pelno apskaičiavimo reikalavimai	16
3. Pasirinktos technologijos	18
3.1. Pasirinktų technologijų apžvalga	18
3.2. Galinė dalis: Typescript NodeJS aplinka ir ExpressJS biblioteka	18
3.3. Priekinė dalis: PugJS kalba, Bootstrap 5 ir DataTables bibliotekos	19
3.4. PostgreSQL duomenų bazė	19
3.5. RabbitMQ žinučių brokeris	20
3.6. Programavimo aplinka: kodo tikrinimas ir formatavimas	20
4. Projekto architektūra	21
4.1. Naudojimo atvejai	21
4.2. Nefunkciniai reikalavimai	22
4.3. Sistemos architektūra	23
4.4. MVC serverio architektūra	23
4.5. Vartotojo mokesčių deklaravimas naudojant sistemą	25
4.6. Vartotojo prisijungimas	26
4.7. Transakcijų failo įkėlimas	27
4.8. Vartotojo mokesčių skaičiavimas	28
4.9. Duomenų bazės esybių–ryšių modelis	30
4.10. Registravimas ir klaidų tvarkymas	32
5. Duomenų gavimas ir apdorojimas	33

5.1.	Vartotojo transakcijų įkėlimas	33
5.2.	Sistemoje palaikomų keityklų ir transakcijų etikečių atnaujinimas	33
5.3.	Valiutų kursų nustatymas	34
5.3.1.	Nacionalinių valiutų kursų nustatymas	34
5.3.2.	Kripto valiutų kursų nustatymas	35
6.	Programavimo darbai	36
6.1.	Paskyros valdymas ir duomenų šalinimas	36
6.1.1.	Registracija ir prisijungimas	36
6.1.2.	Paskyros redagavimas ir šalinimas	37
6.1.3.	Dviejų faktorių autentifikacija	37
6.2.	Duomenų apie transakcijas įkėlimas	38
6.3.	Transakcijų peržiūra ir redagavimas	38
6.4.	Pelno ir mokesčių informacijos peržiūra	39
6.5.	Transakcijų suvestinės generavimas	39
6.6.	Naudojimosi sistema vadovas	40
6.7.	Sistemos pritaikymas tarptautinei rinkai	41
6.8.	Išsigijimo išlaidų ir pelno skaičiavimo algoritmai	42
6.9.	Sistemos testavimas su realiais duomenimis	43
6.9.1.	Rankiniai testai	43
6.9.2.	Automatiniai testai	44
	Išvados ir rekomendacijos	46
	Ateities tyrimų planas	47
	Literatūros šaltiniai	48

Santrauka

Didėjant kriptovaliutų naudojimui ir griežtėjant jų reguliavimui, kriptovaliutų vartotojams Lietuvoje metų gale tenka apskaičiuoti, deklaruoti ir sumokėti reikiamus kriptovaliutų mokesčius valstybinei mokesčių inspekcijai, šis procesas gali būti sudėtingas ir reikalaujantis laiko, dėl didelio transakcijų kiekio ir jų pasiskirstymo per skirtingas kriptovaliutų keityklas, pinigines ir kitus servigus, o visą kriptovaliutų mokesčių deklaravimo procesą palengvinančios sistemos Lietuvos rinkoje dar nėra, taigi pagrindinis baigiamojo darbo tikslas – sukurti saugią, lengvai naudojamą ir nesunkiai plečiamą kriptovaliutų mokesčių deklaravimo palengvinimo sistemą skirtą Lietuvos gyventojams.

Šios sistemos vartotojai galės lengvai importuoti duomenis apie transakcijas iš kriptovaliutų keityklų susisteminant šiuos duomenis vienoje vietoje ir leidžiant vartotojui peržiūrėti savo metinį pelną, apskaičiuotus mokėtinus mokesčių dydžius, bei atsisiųsti bendrą metinę pervedimų išsklotinę mokesčių deklaracijos pildymui. Pasirinkta sistemos architektūra suteiks galimybę lengvai plėsti palaikomų kriptovaliutų keityklų sąrašą, nesunkiai pridėti naujų mokesčių tipų ir mokesčių skaičiavimo algoritmų ar pritaikyti sistemą prie naujų mokesčių pokyčių. Taip pat sudaroma galimybė lengvai palaikyti didėjančią sistemos apkrovą.

Darbo metu buvo atliekama panašių sistemų analizė, sistemos reikalavimų formulavimas, tinkamiausių technologijų pasirinkimas, sistemos architektūros sudarymas, duomenų gavimo ir apdorojimo nustatymas, sistemos įgyvendinimas, bei testavimas.

Raktiniai žodžiai: kriptovaliutos, mokesčiai, sistema, svetainė

Summary

Cryptocurrency tax declaration facilitation system

As the regulation of cryptocurrencies improves and the use of cryptocurrencies increases, cryptocurrency users in Lithuania have to calculate, declare and pay the necessary cryptocurrency taxes to the state tax inspectorate at the end of the year. This process can be complicated and time-consuming due to the high volume of transactions and their distribution through different cryptocurrency exchanges, wallets and other services, also there is no system that would facilitate the entire cryptocurrency tax declaration process accessible for Lithuanian residents yet, so the main objective of this work is to create a secure, easy-to-use and easily scalable cryptocurrency tax declaration facilitation system for Lithuanian residents.

Users of this system will be able to easily import their transaction history data from used cryptocurrency exchanges into the system, which formats this data and allows users to view their annual profit, calculated taxes to be paid and download the general annual transfer report for tax filing. To increase the security of user's sensitive data, users can also add two factor authentication to their accounts and delete their data after the tax declaration is complete. System architecture is developed with easy adaptation to future changes and growth in mind, which allows the system to simply adapt to local cryptocurrency tax changes, the ability to rapidly support new cryptocurrency exchanges from which transactions can be imported, also ease of adding additional tax calculation strategies. The system's architecture also allows scaling of the system by adding more virtual machines.

The work included the following tasks: analysis of similar systems, formulation of system requirements, selection of the most suitable technologies, data acquisition and processing determination, system development and testing.

Keywords: cryptocurrencies, taxes, system, website

Iyadas

2008 metų finansinė krizė lėmė daugelio žmonių pasitikėjimo finansų institucijomis ir tradicinėmis finansų sistemomis praradimą [19]. Krizės metu daugelis bankų ir kitų finansinių institucijų susidūrė su bankroto grėsme ir tai turėjo rimtų padarinių ekonomikai ir visuomenei. Žmonės ieškojo alternatyvų tradicinėms finansų sistemoms, kurios pasirodė esančios nepatikimos ir nestabilios krizės metu. *Bitcoin* kūrėjai siekė sukurti naują ir alternatyvią finansų sistemą, kuri būtų nepriklausoma nuo tradicinių finansinių institucijų ir valstybės reguliavimo [16]. *Bitcoin* buvo sukurtas 2009 metais kaip pirmoji kriptovaliuta, sukurta anoniminio autoriaus, pasivadinusio *Satoshi Nakamoto*, kurio tapatybė vis dar nėra žinoma [18]. *Bitcoin* yra skaitmeninė valiuta, kurios kiekis apyvartoje yra ribotas, o šis ribojimas yra įdiegtas jo programinėje įrangoje. Pagal kūrėjų nustatytas taisykles, *Bitcoin* kiekis, kuris gali būti sukurtas, yra ribojamas iki 21 milijono, o dabar apytiksliai yra sukurta jau apie 18,7 milijono vienetų [11]. Tai reiškia, kad *Bitcoin* yra atsparus infliacijai ir skiriasi nuo tradicinių valiutų, kurių tiekimas gali būti reguliuojamas ir padidinamas bet kuriuo metu, tokiu atveju didinant ir infliaciją. *Bitcoin* yra paremtas blokų grandinės technologija, kuri leidžia vykdyti finansinius sandorius tiesiogiai tarp vartotojų ir viešai išsaugant transakcijų sąrašą, o tai pašalina tarpininkų poreikį, tokių kaip bankai ar kitos finansinės institucijos [8]. Tiesioginiai sandoriai tarp vartotojų leidžia sumažinti sandorių mokesčius ir greitai vykdyti finansinius sandorius, bei suteikia didesnę skaidrumą ir pasitikėjimą vartotojams, o tai ypač svarbu krizės metu, kai tradicinės finansų sistemos gali susidurti su sunkumais dėl slaptumo ir netinkamo valdymo. Jokių centrinių valdymo mechanizmų nėra, todėl *Bitcoin* vertė priklauso nuo rinkos paklausos ir pasiūlos [10]. Tai yra svarbus skirtumas nuo tradicinių valiutų, kurių vertę valdo centriniai bankai ir kiti reguliuotojai. Šios savybės paskatino žmones ieškančius alternatyvių finansų sistemų, naudoti *Bitcoin*.

Nuo pirmosios kriptovaliutos paleidimo 2009 metais, kriptovaliutų panaudojimas stipriai išaugo, o šiuo metu *Bitcoin* tinkle kas mėnesį atliekama virš 250 tūkstančių pervedimų [2, 6]. Kriptovaliutų populiarumas ir nauda lėmė šios technologijos integraciją į kasdieninį gyvenimą, šiuo metu, bet kuris asmuo gali laisvai pirkti, parduoti, keisti kriptovaliutas keityklose, atsiskaityti kriptovaliutomis, įvairiose, jas priimančiose parduotuvėse, kurių sąrašas vis didėja, ar net atsiskaityti parduotuvėse kur kriptovaliutos nėra priimanamos, tačiau priimanamos mokėjimo kortelės, naudojant specializuotą mokėjimo kortelę, kuri susieta su vartotojo kriptovaliutų pinigine ir atsiskaitymo metu kriptovaliutas paverčia į reikiamą nacionalinę valiutą. Vienas iš tokių mokėjimo kortelių pavyzdžių yra *Binance Card*, pradėjus naudoti šį servisą, vartotojui yra suteikiama fizinė mokėjimo kortelė, kuri susiejama su kriptovaliutomis laikomomis *Binance* keityklos sąskaitoje, o atliekant mokėjimą kortele, turimos kriptovaliutos keičiamos į mokėjimo valiutą pagal tuo metu esamą kriptovaliutos kainą *Binance* keitykloje [5]. Kriptovaliutų naudojimo populiarėjimo versle tendenciją parodo ir vis dažnesnis *Bitcoin* minėjimas Jungtinių Amerikos Valstijų vertybinių popierių ir biržos komisijos (angl. *U.S. Securities and Exchange Commission*) 13F pranešimuose, kuriuos pateikia institucinių investicijų valdytojai, savo versle naudojantys JAV paštą ir investuojantys daugiau nei 100 mln. JAV dolerių į *Section 13(f)* vertybinius popierius [15]. O didėjanti kriptovaliutų poreikis kasdieniniame gyvenime Lietuvoje atskleidė Vilniaus universiteto mokslininkų atliktas tyrimas, orientuotas į finansų technologijų galimybes turizmo rinkoje, kuris parodė, jog didžiausias baimes vartotojams sukelia kriptovaliutų kainų nestabilumas, tačiau paklausa atsiskaityti kriptovaliutomis Lietuvoje už skrydžio paslaugas, automobilio nuomos paslaugas ir bilietus į renginius yra gan didelė, nes kriptovaliuta nesiejama su kredito kortelėmis, asmens tapatybę patvirtinančiais dokumentais ar parašu [13].

Tačiau norint gauti kriptovaliutų nebūtina jų pirkti, daugumą kriptovaliutų galima išgauti, jas kasant ar *staking* tokiu būdu prisidedant prie kriptovaliutos tinklo veikimo ir už tai gaunant tam tikrą, tos kriptovaliutos kiekį. Mažus kriptovaliutų kiekius taip pat galima gauti dalyvaujant *airdrop* procesuose, kurių metu kriptovaliutų kūrėjai, dažniausiai marketingo tikslais, nusiunčia žmonių grupei nedidelį kriptovaliutos kiekį. Štai *Stellar* kriptovaliutos kūrėjai skyrė net 19 milijardų šios valiutos vienetų asmenims, kurie turėjo daugiau nei 1 *Bitcoin*, norint atsiimti šiuos nemokamus *Stellar* kriptovaliutos vienetus, reikėjo patvirtinti savo tapatybę prisijungiant per *Facebook* socialinį tinklą, susikurti *Stellar* pinigines adresą, bei patvirtinti savo *Bitcoin* adreso nuosavybės teises pasirašant žinutę, tokiu būdu *Stellar* kūrėjai skatino žmones, kurie jau naudojami kriptovaliutomis išbandyti ir jų tinklą [9].

Kriptovaliutų kaina nepaliaujamai svyruoja, štai 2017 metų pradžioje *Bitcoin* vieneto kaina buvo vos 1000 eurų, o 2021 metų balandžio mėnesį kaina pakilo net iki 50000 eurų už vienetą, tačiau po mėnesio kaina nukrito iki 35000 eurų už vienetą [1]. Taigi uždirbti ar prarasti galima, ne tik investuojant į kriptovaliutas, bet ir perkant ar kitaip gaunant jas su tikslu panaudoti šias kriptovaliutas, kaip tikrąsias valiutas – atsiskaitant už prekes, kadangi dėl kainos pokyčio, įsigijimo kaina ir kaina atsiskaitymo metu gali skirtis, o šis pokytis vertinamas kaip netyčinis pelnas ar praradimas.

Visas pelnas iš kriptovaliutų investavimo, naudojimo, kasimo ar kitokios veiklos, gautas Lietuvos gyventojo, privalo būti deklaruojamas ir apmokestinamas, jei per kalendorinius metus gautų pajamų už parduotą kitą turtą (buitinę techniką, meno kūrinius, kriptovaliutą ir pan.), kuris nepriiskiriamas individualios veiklos turtui, ir su jo įsigijimu susijusių išlaidų, pagrįstų juridinę galią turinčiais dokumentais, skirtumas viršija 2,5 tūkst. eurų [20, 23]. Kol kas daugelis žmonių vis tiek nedeklaruoja kriptovaliutų pajamų, nes didžioji dalis kriptovaliutų leidžia išlaikyti anonimiškumą, taip pat valstybės neturi koordinuoto ir automatizuoto būdo gauti informaciją apie gyventojo kriptovaliutų uždarbio prieaugį kriptovaliutų biržose, norint patikrinti ar gyventojas teisingai užpildė savo kriptovaliutų pelno mokesčių deklaraciją [21]. Tačiau stiprėjanti kriptovaliutų reguliacijai, artimiausiu metu tokia veikla taps vis sunkiau įgyvendinama, nes kriptovaliutų keityklos nuo 2025 metų pagal Europos Sąjungos įstatymus turės nustatyti kiekvieno savo vartotojo tapatybę, o nedeklaravus reikiamų mokesčių laiku, per ateinančius 5 metus dar yra galimybė sulaukti iš VMI (valstybinės mokesčių inspekcijos) ne tik reikalavimo susimokėti nustatytus mokesčius, bet ir delspinigius ar baudą už šių mokesčių mokėjimo vėlavimą [14].

Nė viena kriptovaliutų birža nepalaiko visų egzistuojančių kriptovaliutų, tad norint prekiauti kai kuriomis kriptovaliutomis reikia atsidaryti sąskaitą jas palaikančiose biržose. Taigi kai vartotojai atlieka daug operacijų (investuoja, sumoka už prekę, kasą kriptovaliutą ar kitaip naudoja kriptovaliutas), jos dažniausiai atliekamos skirtingose blokų grandinėse, bei pasiskirsto per daugybę piniginių, keityklų ar kitokių servisų, o prireikus deklaruoti pajamas, surinkti visas operacijas į vieną vietą ir apskaičiuoti mokėtinus mokesčius gali būti labai sudėtinga. Taip pat sunkumai gali kilti ir aiškinantis, kaip ir kur deklaruoti tam tikras kriptovaliutų pajamas, nes kol kas Lietuvoje nėra nei vieno lengvai suprantamo, susisteminto informacijos šaltinio, kuriame būtų paaiškinami skirtingų kriptovaliutų veiklų mokesčių deklaravimo procesai ir jų dydis. Šiuo metu pagrindinė informacija apie kriptovaliutų deklaravimą Lietuvos gyventojams yra pateikiama VMI paskelbtame 14-os puslapių *PDF* formato faile *Bendrosios Nuostatos*, kuris yra sunkiai suprantamas, bei neturi visos reikiamos informacijos [23].

Darbo tikslas yra sukurti informacinę sistemą, leidžiančią vartotojams palengvinti kriptovaliutų mokesčių deklaravimo procesą greitai ir patogiai surenkant visas vartotojo atliktas kriptovaliutų operacijas iš skirtingų keityklų ar piniginių ir apskaičiuojant pelną, bei reikiamą deklaruoti pajamų kiekį, taip pat vartotojui suteikiant svarbiausią informaciją apie apmokestinamas kriptovaliutų

veiklas Lietuvoje, bei jų mokesčių dydžius.

Darbo uždaviniai:

1. Atlikti panašių esamų sistemų, veikiančių užsienyje, analizę (žr. 1 skyrių).
2. Surinkti informaciją apie Lietuvoje apmokestinamas kriptovaliutų veiklas, bei jų deklaravimo reikalavimus (žr. 2.1 skyrių).
3. Suformuluoti sistemos funkcijas, bei jos reikalavimus, atsižvelgiant į panašių sistemų savybes ir Lietuvos kriptovaliutų mokesčių deklaravimo reikalavimus ir procesus (žr. 2 skyrių).
4. Atlikti technologijų skirtų internetinių svetainių kūrimui tyrimą ir išsirinkti tinkamiausias technologijas projektui (žr. 3 skyrių).
5. Suprojektuoti sistemos architektūrą, pagal iškeltus reikalavimus (žr. 4 skyrių).
6. Įgyvendinti kriptovaliutų transakcijų įkėlimo iš pasirinktų keityklų ir piniginių funkcionalumą (žr. 5, 6 skyrių).
7. Sukurti algoritmą, kuris pagal atliktas kriptovaliutų operacijas apskaičiuotų vartotojo pelną ir reikiamus deklaruoti mokesčius (žr. 6 skyrių).
8. Testuoti pelno ir mokesčių skaičiavimo algoritmus su realiais duomenimis (žr. 6.9 skyrių).

1. Panašių sistemų analizė

1.1. Sistemos kriptovaliutų mokesčių deklaravimo palengvinimui

Šiuo metu yra gana daug skirtingų sistemų, kurios palengvina kriptovaliutų mokesčių deklaravimo procesą kadangi poreikis tokioms sistemoms išties didelis, tačiau visų šių sistemų tikslinė rinka yra Jungtinės Amerikos Valstijos, Kanada, Jungtinė Karalystė ar kitos didelės šalys, nes būtent didesnių šalių rinkose galima pasiekti daugiau vartotojų, o tuo pačiu ir didesnę pelną. Tačiau kai visos tokios sistemos yra skirtos didesnių šalių gyventojams, vartotojai iš mažesnės šalies tokios kaip Lietuva, negali išnaudoti pilno tokių sistemų funkcionalumo, nes kiekvienos šalies mokesčių sistema yra skirtinga, o dauguma programų nėra pritaikytos Lietuvos mokesčių sistemai, netgi naudojant esamas kriptovaliutų mokesčių deklaravimo palengvinimo sistemas, tenka pačiam taisyti gautus duomenis ir redaguoti juos, jog šie atitiktų Lietuvos mokesčių deklaravimo standartus, bei pačiam tikrinti ir skaičiuoti mokėtinas mokesčių sumas.

1.2. Koinly

Koinly.io yra viena populiariausių ir dažniausiai rekomenduojamų kriptovaliutų mokesčių deklaravimo palengvinimo sistemų. Ši svetainė leidžia importuoti kriptovaliutų transakcijas iš daugelio keityklų ir piniginių įvedant savo *API (application programming interface)* raktą arba įkeliant transakcijų failą, apskaičiuoti metinį pelną, bei atsisųsti visų įkeltų transakcijų išrašą. Taip pat sistema leidžia pridėti pasirinktą etiketę prie transakcijos, kuri padeda atskirti transakcijos šaltinį ir gali keisti transakcijos apmokestinimą, šių etikečių pasirinkimą sudaro net 15 variantų: *Airdrop, Fork, Mining, Loan interest, Income, Gift, Lost, Cost, Interest payment, Margin fee, Realized PL, Swap, Liquidity in, Liquidity out*. *Koinly.io* sistema veikia ne tik kaip kriptovaliutų mokesčių deklaravimo palengvinimo sistema, bet ir asmeninių kriptovaliutų lėšų sekimo sistema, kuri parodo visas turimas kriptovaliutas pasiskirsčiusias per keletą keityklų, bei kriptovaliutų vertės istorinį grafiką, kuris vaizduoja kaip keitėsi laikomų asmeninių kriptovaliutų vertė. Ši sistema turi daug informatyvių gidų, kuriuose paaiškinami skirtingų kriptovaliutų veiklų mokesčiai ir jų dydžiai, tačiau šie gidai skirti tik didesnių valstybių gyventojams, Lietuvos kriptovaliutų mokesčių paaiškinimo nėra. Taip pat yra ir lokalių buhalterių rekomendacijos iš kurių galima gauti profesionalios pagalbos kriptovaliutų mokesčių deklaravimo klausimais, tačiau ir šios rekomendacijos galioja tik didesnių valstybių gyventojams, Lietuvos gyventojams tokių rekomendacijų nėra. Svarbu paminėti, jog ši svetainė nėra nemokama, net pigiausias planas, kuris skirtas vartotojams atlikusiems mažiau nei 100 transakcijų kainuoja 39 eurus. Informacijos kur ir kokio dydžio pelno mokesčių reikia deklaruoti ir sumokėti taip pat nepateikiama. Skaičiuojant kapitalo prieaugį kriptovaliutos, kurios gaunamos, kaip palūkanos nėra atskiriamos, nors Lietuvos mokesčių sistemoje šias pajamas reikia deklaruoti atskirai.

1.3. Taxbit

Taxbit.com yra dar vienas populiarus įrankis naudojamas kriptovaliutų mokesčių deklaravimui palengvinti. Funkcionalumas labai primena anksčiau aprašytą sistemą, tačiau *Taxbit* nepalaiko tiek daug keityklų ir piniginių iš kurių būtų galimybė įkelti kriptovaliutų transakcijas, tačiau palaiko, kai kuriuos servigus, kurių nepalaiko kitos sistemos, pavyzdžiui *PayPal* servisą. Galimų etikečių sąrašas čia nėra didelis, gautoms transakcijoms galima pasirinkti vieną iš 3 etikečių: *Transfer*

in, Buy arba Gift received, išsiųstoms transakcijoms galima pasirinkti taip pat vieną iš 3 etikečių: Transfer out, Expense arba Gift sent. Ši svetainė, kaip ir anksčiau minėta sistema nėra nemokama, tačiau yra nedidelis partnerių sąrašas, iš kurių iškeltas transakcijas galima naudoti ir atsisiųsti nemokamai. Nuo 2023 metų spalio 31 dienos šis Taxbit.com funkcionalumas nebeprieinamas, Taxbit.com nutraukė asmeninių kriptovaliutų mokesčių palengvinimo paslaugą ir šiuo metu siūlo tokią paslaugą tik įmonėms.

1.4. Cointracker

Cointracker.io pagrindinė funkcija yra sekti ir atvaizduoti vartotojo kriptovaliutų portfelį ir jo vertę, tačiau ši svetainė taip pat turi ir kriptovaliutų mokesčių deklaravimo palengvinimo funkcionalumą. Skirtingų transakcijų etikečių, kurios padėtų nustatyti transakcijos šaltinį ar tipą, funkcionalumo šioje svetainėje nėra, yra tik pagrindinis transakcijų įkėlimo, ataskaitų generavimo ir kapitalo prieaugio apskaičiavimo funkcionalumas. Svetainė siūlo nemokamą mokesčių deklaravimo palengvinimą vartotojams atlikusiems iki 25 transakcijų per metus, tačiau atlikus vos keletu pervedimų daugiau už kiekį iki 100 transakcijų reikia susimokėti 53 JAV dolerius. Šioje svetainėje kaip ir ankstesnėse sistemose gautos palūkanų pajamos ataskaitose nėra atskiriamos.

1.5. Sistemų kriptovaliutų mokesčių deklaravimo palengvinimui palyginimas

Atlikus panašių sistemų analizę (žr. 1 lentelę) buvo galima nustatyti, jog pagrindinės sistemos kriptovaliutų mokesčių deklaravimo palengvinimui funkcijos yra kriptovaliutų transakcijų įkėlimas iš keityklų, jų peržiūra ir redagavimas, skirtingų valiutų palaikymas, pelno apskaičiavimas, pranešimas ar reikia deklaruoti pajamas. Tuo tarpu dalies šių funkcijų Lietuvos gyventojai negauna iš dabartinėje rinkoje esamų sistemų.

1 lentelė. Konkurentų analizės lentelė. Viršutinėje eilutėje surašyti konkurentai, o kairiajame stulpelyje nurodomos galimos funkcijos. CoinTax – šiame darbe aprašoma sistema.

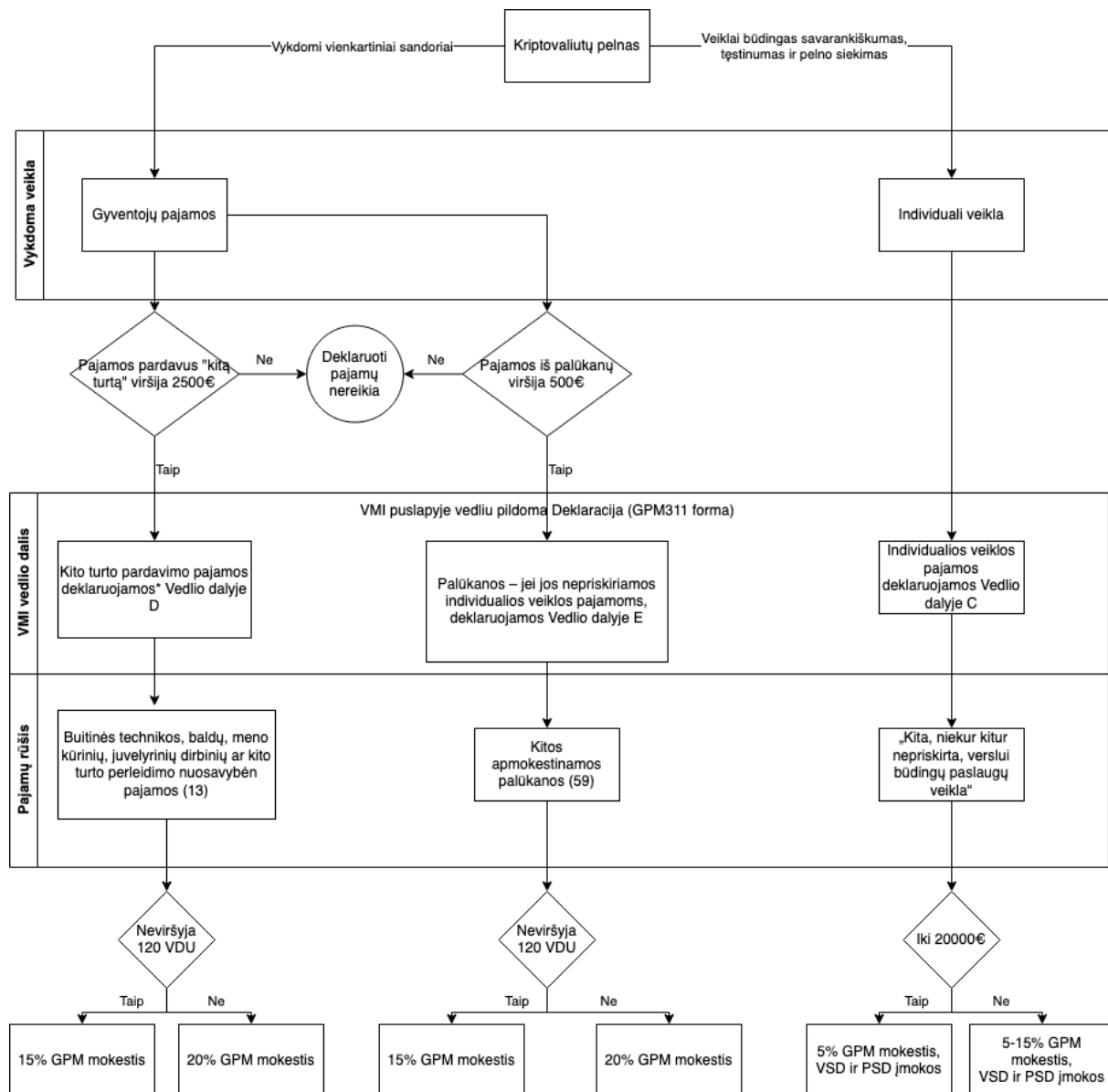
Funkcija	Koinly	Taxbit	Cointracker	CoinTax
Kriptovaliutų transakcijų įkėlimas	Taip	Taip	Taip	Taip
Kriptovaliutų transakcijų peržiūra ir redagavimas	Taip	Taip	Taip	Taip
Skirtingų valiutų palaikymas	Taip	Taip	Taip	Taip
Skirtingų transakcijų etikečių palaikymas	Taip	Taip	Ne	Taip
Pelno apskaičiavimas	Taip	Taip	Taip	Taip
Pranešimas ar reikia deklaruoti pajamas	Ne	Ne	Ne	Taip
Ataskaitoje atskiriamos palūkanų pajamos	Ne	Ne	Ne	Taip
Lietuvos mokesčių sistemos palaikymas	Ne	Ne	Ne	Taip
Asmeninių duomenų pašalinimas iš sistemos	Taip	Ne	Taip	Taip

2. Sistemos reikalavimai

2.1. Pajamų gautų vykdant kriptovaliutų sandorius mokestiniai reikalavimai Lietuvoje

2.1.1. Pajamų gautų vykdant kriptovaliutų sandorius mokesčių apžvalga Lietuvoje

Nuolatiniai Lietuvos gyventojai, pasibaigus kalendoriniams metams, iki kitų kalendorinių metų gegužės 1 dienos privalo pateikti mokesčio administratoriui metinę pajamų mokesčio deklaraciją ir sumokėti joje apskaičiuotą pajamų mokestį. Uždirbti iš kriptovaliutų galima arba per vienkartinius sandorius, arba vykdant individualią veiklą. Mokesčių dydžiai priklauso nuo vykdomos veiklos tipo, pajamų rūšies ir apskaičiuoto pelno (žr. 1 pav.).



1 pav. Kriptovaliutų mokesčių deklaravimo procesų, pagal vykdomą veiklą, diagrama (GPM – gyventojų pajamų mokestis, VDU – vidutinio šalies darbo užmokesčio dydžio suma, VSD – valstybinio socialinio draudimo įmokos, PSD – privalomojo sveikatos draudimo įmokos.)

2.1.2. Pajamų gautų iš pavienių kriptovaliutų sandorių mokesčių deklaravimas Lietuvoje

Jei pajamų gautų pardavus ne individualios veiklos neregistruotą turtą (virtualią valiutą) ir to turto įsigijimo kainos bei kitų GPMĮ (gyventojų pajamų mokesčio įstatymo) 19 straipsnyje nurodytų su šio turto pardavimu susijusių išlaidų ir / arba pasigaminimo išlaidų skirtumas, per kalendorinius metus neviršijantis 2 500 eurų, šių pajamų deklaruoti nereikia.

Pajamos iš kriptovaliutų *stakinimo* ar laikymo keitykloje gaunant palūkanas deklaruojamos atskirai kaip palūkanos. Šias pajamas deklaruoti reikia viršijus 500 eurų pelno ribą [24]. Visų kriptovaliutų pajamų deklaravimas vyksta per VMI vedlio sistemą. Pajamų mokesčio tarifas priklauso nuo pelno dydžio, jei pajamos viršija 120 VDU (vidutinių darbo užmokesčių), kuris 2023 metais yra 1684,90 eurų, pajamų mokesčio tarifas yra 20%, kitu atveju taikomas įprastas 15% pajamų mokesčio tarifas [25, 7]. Mokestinę sumą (M), galima apskaičiuoti iš pelno (P) atėmus neapmokestinamąjį pajamų dydį (NPD) ir padauginus gautą skaičių iš pajamų mokesčio tarifo (T):

$$M = (P - NPD) \cdot T$$

2.1.3. Pajamų gautų vykdant kriptovaliutų sandorius su individualia veikla mokesčių deklaravimas Lietuvoje

Pajamos iš sandorių, susijusių su kriptovaliuta, turi būti apmokestinamos kaip individualios veiklos pajamos, jeigu gyventojas sudaromi sandoriai atitinka šiuos individualios veiklos požymius: veiklos savarankiškumo, tęstinumo ir tokios veiklos tikslo – pajamų ar kitokios ekonominės naudos siekio buvimą. Individuali virtualių valiutų pirkimo–pardavimo bei gamybos (kasimo) veikla priskiriama EVRK (ekonominės veiklos rūšių klasifikatoriaus) 2 red. 82.99 klasei *Kita, niekur kitur nepriskirta, verslui būdingų paslaugų veikla* [12]. Užsiimant individualia veikla mokesčių skaičiavimas yra sudėtingesnis ir gali skirtis priklausomai nuo asmeninės situacijos, kadangi reikia mokėti valstybinio socialinio draudimo ir privalomojo sveikatos draudimo įmokas, taip pat išlaidų atskaitymo skaičiavime, galima naudoti faktinių išlaidų sumą arba 30% sumą nuo pajamų, pasirinkus papildomą pensijos kaupimą, šios įmokos gali būti naudojamos atskaitymuose. Individualios veiklos GPM priklauso nuo apmokestinamų pajamų sumos dydžio ir gali būti tarp 5 ir 15 procentų, tačiau reiktų atkreipti dėmesį, kad individualios virtualių valiutų pirkimo–pardavimo veiklos pajamos gautos iki 2017 metų gruodžio 31 dienos apmokestinamos taikant 5 procentų gyventojų pajamų mokesčio tarifą. Sistemoje individualios veiklos mokesčių suma kol kas nebus apskaičiuojama, ją turėtų apskaičiuoti pats vartotojas pagal savo esamą situaciją, tačiau individualia veikla užsiimantiems gyventojams bus palengvinamas mokesčių skaičiavimas, jiems pateikiant transakcijų išsklotinę su pajamomis ir išlaidomis. Tuo tarpu sistemoje turėtų būti galimybė, esant poreikiui, lengvai pridėti naujų mokesčių skaičiavimo strategijų priklausomai nuo transakcijos atlikimo datos ir pajamų rūšies, kadangi mokesčių dydžiai gali keistis ateityje.

Užsiimant individualia veikla reikia vesti buhalterinę apskaitą (žr. 2 pav.), kurioje registruojamos įvykdytų transakcijų pajamos ir išlaidos. Virtualiąją valiutą įsigijusių asmenų pajamos bei atitinkamai patirtos išlaidos gali būti pripažįstamos, pagal dokumentus, kurie patvirtina, kad buvo sumokėta už virtualiąją valiutą bei iš šio dokumento yra matoma ūkinės operacijos arba ūkinio įvykio turinys ir data, ūkinės operacijos arba ūkinio įvykio rezultatas, prekės pavadinimas ir kaina. Kol kas sistema pagal įkeltas vartotojo transakcijas negalės atlikti individualios veiklos buhalterinės apskaitos, tačiau sistemos kūrimo reikės atsižvelgti į tokio funkcionalumo lengvą pridėjimą

ateityje, kai bus palaikomas ir individualios veiklos mokesčių deklaravimo palengvinimo procesas.

GYVENTOJO INDIVIDUALIOS VEIKLOS PAJAMŲ IR IŠLAIDŲ APSKAITOS ŽURNALAS

(gyventojo vardas, pavardė, asmens kodas)
(gyvenamoji vieta)
(veiklos pavadinimas)

Eil. Nr.	Data	Dokumento data, pavadinimas ir numeris	Operacijos turinys	Pajamų suma (eurais)	Išlaidos ir (arba) leidžiami atskaitymai, susiję su individualios veiklos pajamų gavimu arba uždirbimu		
					Prekių, medžiagų, žaliavų, detalių įsigijimo		
1	2	3	4	5	6	7	8

2 pav. Individualios veiklos buhalterinė apskaita

2.1.4. Apmokestinamos ir neapmokestinamos kriptovaliutų operacijos Lietuvoje

2 lentelėje vaizduojamos apmokestinamos ir neapmokestinamos kriptovaliutų veiklos, matome, jog ne visos kriptovaliutų operacijos yra apmokestinamos, taigi jei mokestinių metų laikotarpiu Lietuvos gyventojas tik pirkė kriptovaliutą už eurus ir šias kriptovaliutas laikė asmeninėje piniginėje, kriptovaliutų keitykloje ar kitokiame servise iš to negaudamas jokių papildomų palūkanų, deklaruoti kriptovaliutų pajamų nereikia.

Tuo tarpu pardavus kriptovaliutą už bet kokią nacionalinę valiutą (eurus, JAV dolerius ar kitą) skirtumas tarp pirkimo ir pardavimo kainos yra apmokestinamas, o pardavus kriptovaliutą ne už eurus, o už kitą nacionalinę valiutą, reikia nustatyti ir gautos nacionalinės valiutos vertę eurai, tai galima padaryti pritaikius tos dienos valiutos rinkos kainą eurai. Keičiant vieną kriptovaliutą į kitą, taip pat reikia nustatyti gaunamos kriptovaliutos kainą eurai, pirkimo momentu, šią kainą galima nustatyti naudojant pasirinktos kriptovaliutų biržos vidutinę dienos kainą. Gavus kriptovaliutą iš kasimo, *stakinimo*, *DeFi* protokolų, *airdrop* procesų, ar kitų apmokestinamų veiklų reikia ne tik mokėti mokesčius šias kriptovaliutas pardavus, bet ir gavus. Šiuo atveju gautų kriptovaliutų kainą eurai irgi reikia įvertinti pagal pasirinktą kriptovaliutų kainų nustatymo metodą.

2 lentelė. Apmokestinamos ir neapmokestinamos kriptovaliutų veiklos.

Neapmokestinamos veiklos
Kriptovaliutų pirkimas naudojant nacionalinę valiutą
Kriptovaliutų laikymas piniginėje ar keitykloje
Kriptovaliutų pervedimas į asmeninę piniginę
Kriptovaliutų dovanojimas
Kriptovaliutų aukojimas
Apmokestinamos veiklos
Kriptovaliutų pardavimas už nacionalinę valiutą
Kriptovaliutų keitimas į kitą kriptovaliutą
Atsiskaitymas kriptovaliutomis už prekes ar paslaugas
Kriptovaliutų kasimas
Kriptovaliutų <i>stakinimas</i>
Kriptovaliutų gavimas naudojant <i>DeFi</i> protokolus
Kriptovaliutų gavimas iš trečiųjų šalių, pavyzdžiui kriptovaliutų gavimas, už jų perdavimą keityklai
Kriptovaliutų <i>airdrop</i> gavimas
Kriptovaliutų gavimas po blokų grandinės atsiskyrimo (angl. block-chain fork)

Apibendrinus, kurioje sistemoje turi būti galimybė vartotojui pasirinkti kriptovaliutų transakcijos paskirtį, kuri atskleistų koku būdu ši kriptovaliuta buvo gauta, pavyzdžiui kasant kriptovaliutą ar dalyvaujant *airdrop* procese. Taip pat sistemoje reikia suformuoti procesus, kuriais būtų galima nustatyti kitos nacionalinės valiutos ar kriptovaliutos kainą eurai, tam tikrą dieną.

2.2. Palaikomos kriptovaliutų keityklos ir jų operacijos

Šiuo metu yra daugybė kriptovaliutų keityklų, kuriomis gali naudotis Lietuvos gyventojai, tačiau šio darbo tikslas sukurti sistemą, kuri leistų įkelti transakcijas iš kelių populiariausių kriptova-

liutų keityklų, bei likusių mokesčių skaičiavimo funkcionalumą, tuo tarpu, atliekant architektūrinius sprendimus leidžiančius ateityje lengvai pridėti naujų kriptovaliutų keityklų palaikymą. Taip pat pradinės keityklos pasirenkamos atitinkamai, jog sistema turėtų palaikymą daugumai kriptovaliutų operacijų minėtų 2 lentelėje (žr. 3 lentelę). Sistemoje bus palaikomos 3 kriptovaliutų keityklos, iš kurių vartotojai galės importuoti transakcijų duomenis

Coinbase yra viena iš populiariausių kriptovaliutų biržų pasaulyje, kurioje galima įsigyti, parduoti ir laikyti, bei keisti įvairias kriptovaliutas. Be to, *Coinbase* taip pat teikia tam tikras finansines paslaugas, tokias kaip kriptovaliutų perleidimas keityklai, už tai gaunant palūkanų, pinigų siuntimas ir priėmimas iš kitų vartotojų. Iš šios sistemos transakcijas galima eksportuoti naudojant svetainės *API* prieigą arba parsisiunčiant *PDF* ar *CSV* failą.

Coinbase Pro yra *Coinbase* kriptovaliutų biržos profesionali prekybos platforma. Ji suteikia galimybę vartotojams prekiauti kriptovaliutomis su mažesniais mokesčiais ir geresnėmis rinkos kainomis, bei gauti daugiau informacijos apie kriptovaliutų kainas ir jų istoriją. Taip pat *Coinbase Pro* suteikia papildomas prekybos galimybes, tokias kaip limito ir stop limito užsakymai. Šioje platformoje vartotojai gali pirkti, parduoti ar keisti kriptovaliutas. Svetainėje transakcijas galima eksportuoti, taip pat, naudojant svetainės *API* prieigą arba parsisiunčiant *PDF* ar *CSV* failą.

Binance yra viena didžiausių kriptovaliutų biržų pasaulyje, kurioje galima prekiauti daugiau nei 100 skirtingų kriptovaliutų. *Binance* taip pat teikia keletą finansinių paslaugų, tokias kaip kriptovaliutų saugojimas, kriptovaliutų siuntimas ir priėmimas, o taip pat ir kriptovaliutų kredito paslaugos. Vartotojui gali ne tik pirkti, parduoti ir keisti kriptovaliutas keitykloje, bet ir tiesiogiai iš kitų vartotojų naudojant *Binance P2P* funkcionalumą. Eksportuoti transakcijas galima, naudojant svetainės *API* prieigą arba parsisiunčiant *PDF* ar *Excel* failą.

3 lentelė. Pasirinktos keityklos ir jų palaikomos kriptovaliutų veiklos.

	Coinbase	Coinbase PRO	Binance
Pirkimas	Taip	Taip	Taip
Pardavimas	Taip	Taip	Taip
Keitimas	Taip	Taip	Taip
Skolinimas (gaunant palūkanas)	Taip	Ne	Taip
Mokėjimas	Taip	Ne	Taip
Mokėjimas kortele	Taip	Ne	Taip

2.3. Kriptovaliutų transakcijų apskaitos reikalavimai

Lietuvoje apskaita tvarkoma ir apskaitos dokumentai surašomi naudojant eurą (BAĮ (būhalterinės apskaitos įstatymo) 5 str. 1 dalis), todėl visi sandoriai, vykdomi virtualiąja valiuta, registruojami naudojant eurą. Virtualiosios valiutos ir euro keitimo santykio teisės aktai nereglementuoja, todėl nustatant virtualios valiutos ir eurų kursą, gali būti naudojama visa prieinama informacija ir palyginamaisiais duomenimis rinkoje. Tačiau pagal virtualiosios valiutos apskaitos taisykles, ūkinėms operacijoms ir ūkiniams įvykiams registruoti apskaitoje taikomo euro ir virtualiosios valiutos santykio paskelbimo šaltinį ūkio subjektas turėtų nusistatyti savo apskaitos politikoje. Atsižvelgiant į tai, kad virtualiosios valiutos kursas gali kisti labai ženkliai per labai trumpą laikotarpį, asmuo savo apskaitos politikoje turėtų nusistatyti ne tik šaltinį, bet ir laiko momentą, kuriuo užfiksuotas kursas bus naudojamas apskaitos vedimui, mokestinių prievolių vykdymui [23].

Pagal PMI (pelno mokesčio įstatymo) 11 str. 4 dalį išlaidos, kurių pagrindu pripažįstamos sąnaudos, gali būti grindžiamos tik juridinę galią turinčiais dokumentais, kurie privalo turėti visus buhalterinę apskaitą reglamentuojančių teisės aktų nustatytus privalomus apskaitos dokumentų rekvizitus. Užsienio vienetų ar fizinių asmenų ūkinės operacijos gali būti įformintos nesilaikant PMI 11 straipsnio 4 dalyje apskaitos dokumentams nustatytų reikalavimų. Pagal PMI 11 str. 6 dalį sąnaudos pagal užsienio vienetų arba fizinių asmenų surašytus dokumentus pripažįstamos, jei iš šių dokumentų galima nustatyti ūkinės operacijos turinį ir virtualiosios valiutos pardavėją. Ūkinės operacijos, kurios negali būti pagrįstos apskaitos dokumentais, pagrindžiamos su jomis susijusių ūkinių operacijų ir ūkinių įvykių apskaitos dokumentais ir (arba) surašant buhalterinę pažymą (BAĮ 12 str. 2 dalis). Bet kuriuo atveju, vadovaujantis teisės aktų nuostatomis, apskaitos dokumentų ir jų privalomųjų rekvizitų visuma turi patvirtinti, kad ūkinė operacija įvyko, ir kad ji įvyko būtent tokiomis aplinkybėmis, kurios atspindėtos dokumentuose [23].

Sistemoje transakcijų pagrindimo apskaitos generavimas, turi atitikti Lietuvos mokesťinius reikalavimus, taigi sugeneruotas failas turi turėti šiuos stulpelius:

- Data – transakcijos įvykdymo data
- Operacijos tipas – operacijos rūšis (pirkimas, pardavimas)
- Pajamų kiekis – gautos valiutos kiekis
- Pajamų valiuta – gautos valiutos trumpinys
- Pajamų valiutos kaina pirkimo metu – gautos valiutos vidutinė dieninė kaina
- Išlaidų kiekis – išleistos valiutos kiekis
- Išlaidų valiuta – išleistos valiutos trumpinys
- Išlaidų valiutos kaina pirkimo metu – išleistos valiutos vidutinė dieninė kaina
- Mokesčio kiekis – mokesčio valiutos kiekis
- Mokesčio valiuta – mokesčio valiutos trumpinys
- Mokesčio valiutos kaina pirkimo metu – mokesčio valiutos vidutinė dieninė kaina
- Įsigijimo kaina – išleistos valiutos kiekio įsigijimo kaina (angl. *cost of basis*)
- Pelnas – apmokestinamas pelnas gautas atlikus transakciją

2.4. Pelno apskaičiavimo reikalavimai

Pelnas skaičiuojamas iš pajamų atimant išlaidas, kurias sudaro pirkimo kaina, bei transakcijos mokestis. Išlaidų pagrindo (angl. *cost basis*) vertinimą galima atlikti naudojant įvairius skirtingus algoritmus *First in first out (FIFO)*, *Last in first out (LIFO)*, *Highest in first out (HIFO)* ar kitą. Kiekvienas šių algoritmų specifinėje situacijoje gali sumažinti vartotojui mokesčių dydį ir yra naudojami siekiant optimizuoti mokesčių dydžius, tačiau dažniausiai naudojamas algoritmas apskaičiuojant kriptovaliutų pelną ir mokesčių dydžius yra *First in first out (FIFO)*, kurio naudojime pirmiausiai įsigytos kriptovaliutos yra parduodamos pirmos, o būtent tai dažniausiai ir atitinka realią situaciją. Taigi sistemoje pelno skaičiavimai bus atliekami naudojant kainą eurai, o pelno

apskaičiavimui naudojamas *FIFO* algoritmas atsižvelgiant į galimybę, jog ateityje sistemoje bus pridedama daugiau skirtingų išlaidų pagrindo skaičiavimo algoritmų.

3. Pasirinktos technologijos

3.1. Pasirinktų technologijų apžvalga

Sprendimo įgyvendinimui, kuriama internetinė svetainė, taigi reikia pasirinkti technologijas, kurios leistų sukurti serverį duomenų apdorojimui ir gavimui, vartotojo sąsaja, žinučių brokerį ir duomenų bazę. Renkantis naudojamas technologijas (žr. 4 lentelę) internetinės informacinės sistemos kūrimui svarbiausios technologijų savybės buvo naudojimo paprastumas ir plėtros greitis, kadangi sistema yra kuriama nuo pradžių ir reikalauja daug funkcionalumo, taip pat svarbu saugumas, nes sistemoje laikomi finansiniai vartotojų duomenys, veikimo greitis ir kuriamo kodo kokybė, dėl ateities palaikymo.

4 lentelė. Pasirinktos technologijos skirtingų sistemos dalių kūrimui.

Galinė dalis	
Aplinka	NodeJS
Kalba	Typescript
Serverio kūrimo biblioteka	ExpressJS
Testavimo biblioteka	Jest
Internacionalizacijos biblioteka	i18next
Duomenų bazė	PostgreSQL
ORM (<i>Object-relational mapper</i>) biblioteka	TypeORM
Žinučių brokeris	RabbitMQ
Procesų valdymas	PM2
Priekinė dalis	
Šablonų generavimas	PugJS
Vartotojo sąsaja	Bootstrap 5
Lentelių kūrimo biblioteka	DataTables
Diagramų atvaizdavimo biblioteka	ChartJS
Piktogramų biblioteka	Bootstrap icons
Programavimo aplinka	
Kodo redaktorius	Visual Studio Code
Statinis kodo klaidų taisymas	ESLint
Kodo formatavimas	Prettier
Duomenų bazės valdymas	Beekeeper Studio

3.2. Galinė dalis: Typescript NodeJS aplinka ir ExpressJS biblioteka

Internetinės svetainės galinės dalies kūrimui buvo naudojama *NodeJS* aplinka rašoma *Typescript* kalba. *NodeJS* aplinka pasižymi savo greičiu, bei paprastumu, o 2022 metų *Stack Overflow* apklausoje, *NodeJS* buvo labiausiai naudojama svetainių kūrimo aplinka tarp interneto svetainių programuotojų, taigi didelė *NodeJS* programuotojų bendruomenė užtikrina, jog atsakymus į daugumą kylančių klausimų lengva rasti internete [17]. *NodeJS* aplinkos naudojimas suteikia prieigą prie populiariausios paketų tvarkyklės *npm*, kurioje yra daugybė atviro kodo bibliotekų, šias bi-

bliotekas galima lengvai įdiegti į savo projektą ir taip sutaupyti laiko nekuriant per naują, jau sukurtą tam tikro funkcionalumo.

Typescript kalba, vietoj paprastos *Javascript* kalbos buvo pasirinkta, dėl kodo kokybės gerinimo, *Typescript* kalba reikalauja aprašyti naudojamų kintamųjų tipus, bei integruojasi į kodo redaktoriaus aplinką, tokiu būdu, kodas tikrinamas jį rašant ir daugelis klaidų gali būti randamos dar prieš kompiliuojant kodą, taip pat kodo redagavimo aplinkoje matoma daugiau informacijos apie esamą kodą, taigi nors pats funkcionalumo kūrimas sulėtėja, dėl tipų aprašymo, tai daryti verta, dėl lengvesnio kodo palaikymo ir tobulinimo ateityje.

Express biblioteka yra vienas populiariausių *NodeJS* įrankių internetinių serverių kūrimui. Ši biblioteka buvo pasirinkta dėl savo paprastumo, greičio ir populiarumo. *Express* karkaso paprastumas pagreitina funkcionalumo kūrimo greitį, o tai labai svarbu kuriant naują svetainę per trumpą nustatytą terminą, taip pat šis karkasas suteikia greitą ir efektyvų užklausų siuntimą, tad atsiradus poreikiui būtų galimybė lengvai didinti apdorojamų užklausų kiekį.

Jest biblioteka pasirinkta automatinių vienetinių testų kūrimui, dėl savo paprastumo ir naujingumo, *Jest* galima lengvai integruoti į esamą *Typescript* programinį kodą, o automatinis testų vykdymas atliekamas greitai, bei su daug papildomų galimybių pavyzdžiui stebėti kodą ar automatiškai kodo redaktoriuje vykdyti tik pakeitimus paveikiančius testus. Taip pat *Jest* turi išsamią ir aiškią dokumentaciją, kuri padeda greitai ir efektyviai įgyvendinti norimus testus.

3.3. Priekinė dalis: PugJS kalba, Bootstrap 5 ir DataTables bibliotekos

Pug šablono variklis buvo pasirinktas, nes naudojant jį kartu su *Express* biblioteka, yra pagreitinamas ir supaprastinamas vartotojo sąsajos kūrimo procesas. Kadangi naudotojo sąsaja šioje sistemoje neturi jokių sudėtingo funkcionalumo reikalavimų, naudoti pažangų priekinės dalies karkasą kaip *React* nebuvo prasmės, tuo tarpu naudojant *Pug* šablonus lengvai ir greitai galima kurti vartotojo sąsajos elementus, o kadangi *Pug* naudojamas kartu su *Express* biblioteka, galima lengvai perduoti kintamuosius iš galinės dalies į priekinę dalį, neskiriant tam per daug pastangų, tai supaprastina ne tik informacijos perdavimą, bet ir vartotojo autentifikacijos ir autorizacijos procesus. Vartotojo sąsajos kūrimui buvo naudojama *Bootstrap 5* biblioteka, šios bibliotekos iš anksto suprojektuoti vizualiniai elementai padeda lengvai kurti pastovią, intuityvią vartotojo sąsają, kuri veikia ant įvairių dydžių ekranų. Tuo tarpu, jog nereikėtų kurti lentelių paieškos, puslapiavimo ir rūšiavimo funkcionalumo pačiam, buvo pasitelkta lengvai konfigūruojama ir daugybę funkcionalumo turinti biblioteka *DataTables*, kuri sistemoje vartotojui pagerina ir palengvina sąveiką su lentelėse atvaizduotais duomenimis. Duomenų atvaizdavimo diagrama funkcionalumui įgyvendinti buvo pasirinkta naudoti *ChartJS* biblioteka, kuri perduotus duomenis atvaizduoja norimo tipo diagrama.

3.4. PostgreSQL duomenų bazė

PostgreSQL duomenų bazė yra galinga atvirojo kodo objektų ir ryšių duomenų bazių sistema, galinti efektyviai valdyti didelius duomenų kiekius. Kuriamoje sistemoje komunikacijos su *PostgreSQL* duomenų baze palengvinimui naudojama *TypeORM* biblioteka, taip duomenų gavimui naudojama *Object-relational Mapping* technika, kuri leidžia manipuluoti duomenų bazės objektais nenaudojant *SQL* kalbos, o šį funkcionalumą įgyvendinant bendrai sistemoje naudojama *Typescript* kalba, taip gaunama daugiau papildomų funkcijų, kaip *Typescript* tipų palaikymas ir patogus klaidų, susijusių su duomenų baze, apdorojimas.

3.5. RabbitMQ žinučių brokeris

RabbitMQ yra populiariausias atvirojo kodo žinučių brokeris pasižymintis paprastumu, patikimumu ir lengva integracija, kurio pritaikymas sistemoje užtikrina lengvesnę sistemos plėtrą, vartotojų skaičiui didėjant. Šis žinučių brokeris yra lankstus ir lengvai įdiegiamas, suteikia galimybę pritaikyti skirtingus perdavimo scenarijus ir priderinti sistemą prie konkrečių reikalavimų, bei lengvai integruojasi su kitais sistemų komponentais ir technologijomis, įskaitant *REST API*, *WebSocket* ir kt. Taip pat *RabbitMQ* privalumas yra patikimumas, siunčiant žinutes tarp klientų, galima lengvai užtikrinti, jog kiekviena žinutė bus sėkmingai perduota ir apdorota.

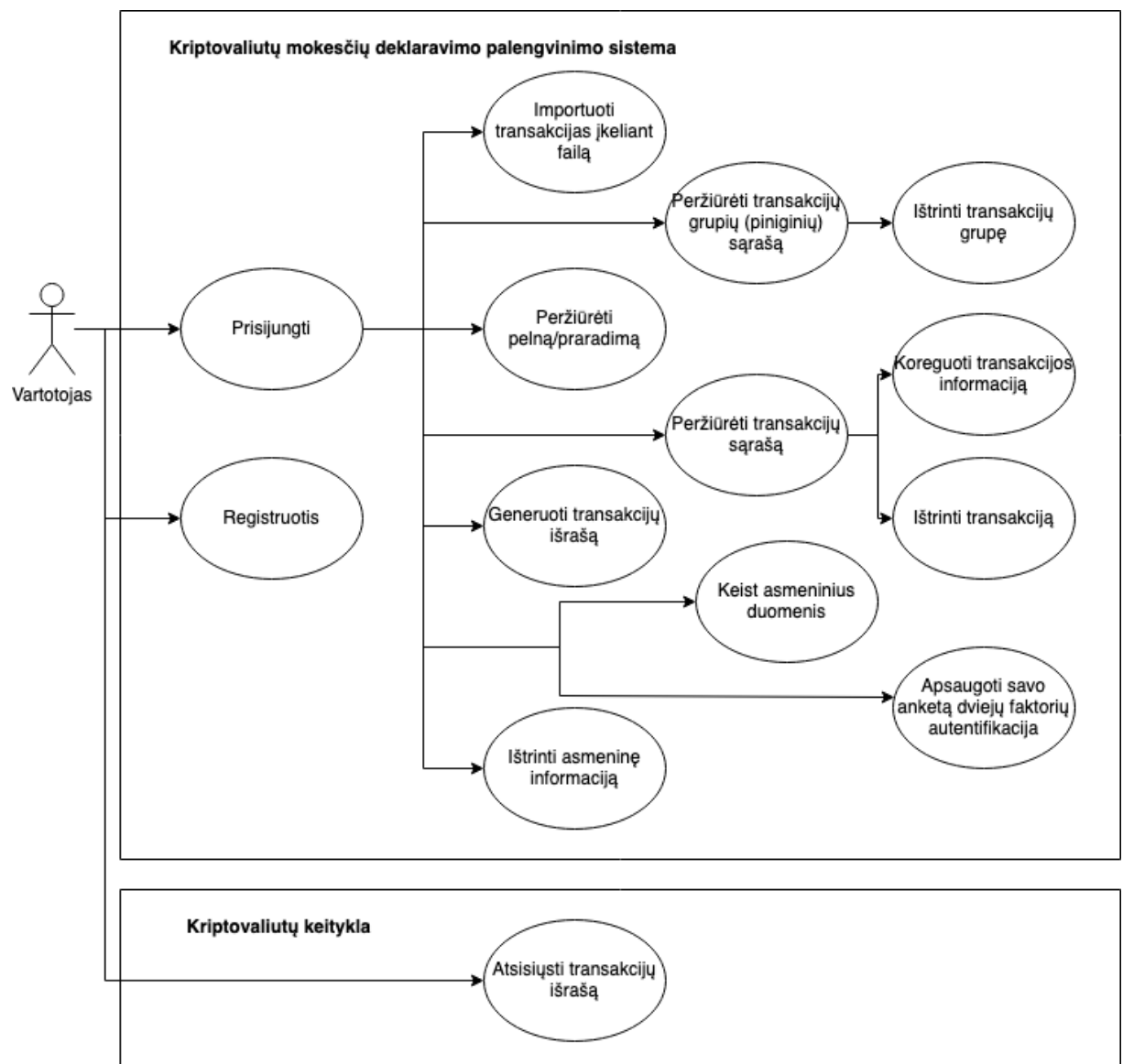
3.6. Programavimo aplinka: kodo tikrinimas ir formatavimas

Sistemos kūrimui naudojamas *Visual Studio Code* kodo redaktorius, kuris yra populiarus, nemokamas, atvirojo kodo, bei turi daugybę plėtinių, kuriuos galima lengvai įsidiegti ir taip pagerinti kodo rašymo greitį ir kokybę. Kodo kokybės palaikymui ir klaidų radimui prieš kompiliavimą yra sukonfigūruojamas ir naudojamas *Visual Studio Code* plėtinys *ESlint*, kuris statiškaiai analizuoja kodą programavimo aplinkoje ir randa klaidas. Kodo formatavimui, bendros tvarkos palaikymui, bei kodo įskaitomumo gerinimui yra sukonfigūruojamas *Visual Studio Code* plėtinys *Prettier*, kuris išsaugojus pakeitimus faile, šį failą suformatuoja pagal konfigūracijoje nurodytus parametrus.

4. Projekto architektūra

Sistemos architektūra yra planuojama atsižvelgiant į 2 skyriuje minėtus reikalavimus, naudojant 3 skyriuje aprašytas technologijas. Pirmiausia aprašoma bendra visos sistemos architektūra ir jos dalys, vėlesniuose poskyriuose plačiau aprašomos skirtingos sistemos dalys, bei vartotojo galimybės sistemoje. Šiame skyriuje yra aprašomas serverio veikimas, vartotojo atliekamų veiksmų galimybės sistemoje, vartotojo mokesčių deklaravimo, bei transakcijų failo įkėlimo procesai, skirtingų mokesčių skaičiavimo strategijų integracija, duomenų bazės struktūra, bei išvesties registravimo ir klaidų tvarkymo procesai sistemoje.

4.1. Naudojimo atvejai



3 pav. Prisijungusio vartotojo naudojimosi sistema atvejų diagrama

Ši naudojimo atvejų diagrama (žr. 3 pav.) parodo skirtingus veiksmus, kuriuos vartotojas gali atlikti sukurtoje sistemoje ir už jos ribų. Prie sistemos funkcionalumo prieiti gali tik prisijungę vartotojai, taigi pradžia vartotojas gali prisijungti arba prisiregistruoti prie sistemos. Prisijungęs

vartotojas turi galimybę apsaugoti savo anketą pridedant papildomą dviejų faktorių autentifikaciją. Norėdamas atlikti kriptovaliutų mokesčių deklaravimą, prisijungęs vartotojas gali importuoti į sistemą savo transakcijas iš įvairių kriptovaliutų keityklų, taigi vartotojas turi apsilankyti naudojamosse kriptovaliutų keityklose, jose sugeneruoti ir atsisiųsti transakcijų išrašus, o šiuos išrašus gali įkelti į sistemą. Tada vartotojas patikrina savo transakcijų pelną ar praradimą ir pažiūri ar už pasirinktus metus vartotojui reikia deklaruoti pajamas. Taip pat jei kyla įtarimų, jog apskaičiuotas pelnas yra klaidingas, vartotojas gali patikrinti įkeltų piniginių, bei transakcijų sąrašą ir ištaisyti klaidas. Atlikus mokesčių deklaravimą vartotojas taip pat turi pasirinkimą ištrinti visus savo duomenis iš sistemos.

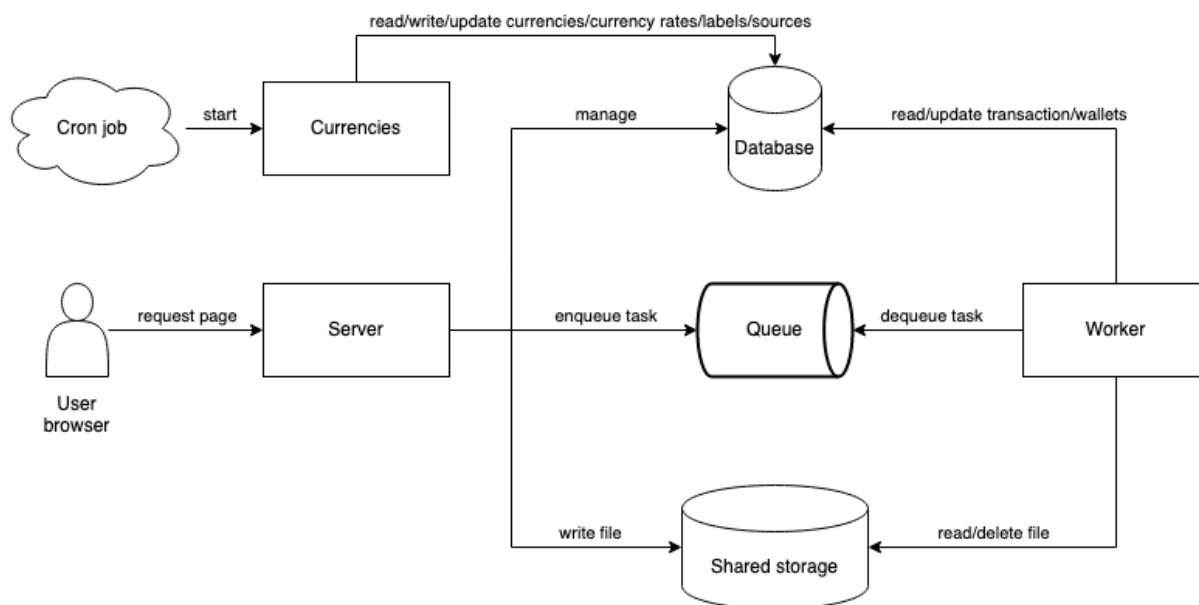
4.2. Nefunkciniai reikalavimai

Pagrindiniai nefunkciniai reikalavimai (žr. 5 lentelę) yra saugumas, sistemos naudojimo lengvumas, lengvas padidėjusio vartotojų kiekio palaikymas, bei sistemos priežiūros paprastumas. Kadangi sistemoje yra saugomi jautrūs vartotojų finansiniai duomenys, reikia užtikrinti vartotojų prisijungimo saugumą, tam naudojamas papildomas dviejų faktorių autentifikacijos prisijungimo funkcionalumas. Vartotojams sistema naudotis turėtų būti paprasta, vartotojo sąsaja neturėtų būti apkrauta nereikalingu funkcionalumu, o įprasto vartotojo naudojimosi sistema veiksmai turėtų būti intuityvūs. Didėjant vartotojų kiekiui ar prisijungiant vartotojams su didesniu kiekiu transakcijų sistemos pajėgumai turėtų būti lengvai plečiami. Sistemos priežiūra turėtų būti nekelianti sunkumų, kadangi programa susideda iš kelių dalių, kurios gali būti pasiskirsčiusios per keletą virtualių mašinų, turėtų būti centralizuota išvesties surinkimo vieta, kurioje būtų galimybė peržiūrėti ir rasti įvykusias klaidas, bei detalesnę jų informaciją.

5 lentelė. Sistemos nefunkciniai reikalavimai.

Saugumas
Naudojimosi paprastumas
Plėtimas
Priežiūros lengvumas

4.3. Sistemos architektūra



4 pav. Sistemos architektūros diagrama, vaizduojamos skirtingos sistemos dalys ir jų sąveika tarpusavyje (dėl aiškumo naudojama programinio kodo terminologija)

Sistemos architektūros diagramoje (žr. 4 pav.) vaizduojamos pagrindinės sistemos dalys. Vartotojo užklausas apdoroja ir atsakymus siunčia *Server* dalis, tuo tarpu daugiau resursų reikalaujančios užduotys, tokios kaip transakcijų failo apdorojimas ir ataskaitų generavimas yra perduodamos *worker* daliai sukuriant užduotį eilėje, bei įkeliant reikiamus failus į bendrą failų saugyklą. Eilėje įkeltas užduotis galima paskirstyti tarp keleto *worker* procesų, taigi didėjant sistemos apkrovai galima sukurti ir integruoti į sistemos veikimą daugiau *worker* procesų ir tokiu būdu lengvai horizontaliai plėsti sistemos pajėgumus. *Currencies* sistemos dalis yra vykdoma nustatytais laiko intervalais naudojant *cron job* įrankį, šis procesas siunčia užklausas ir atnaujiną valiutų, valiutų kursų, etikečių, bei palaikomų kriptovaliutų sąrašą duomenų bazėje. Pasirinkta architektūra leidžia lengvai prisitaikyti prie didėjančios sistemos apkrovos, kuriant daugiau virtualių mašinų, bei užtikrina lengvą ir stabilų svetainės pasiekiamumą, kadangi daugiau resursų reikalaujančios užduotys yra siunčiamos į *worker* procesą, taip nestabdant ir netrukdam serverio užklausų apdorojimo.

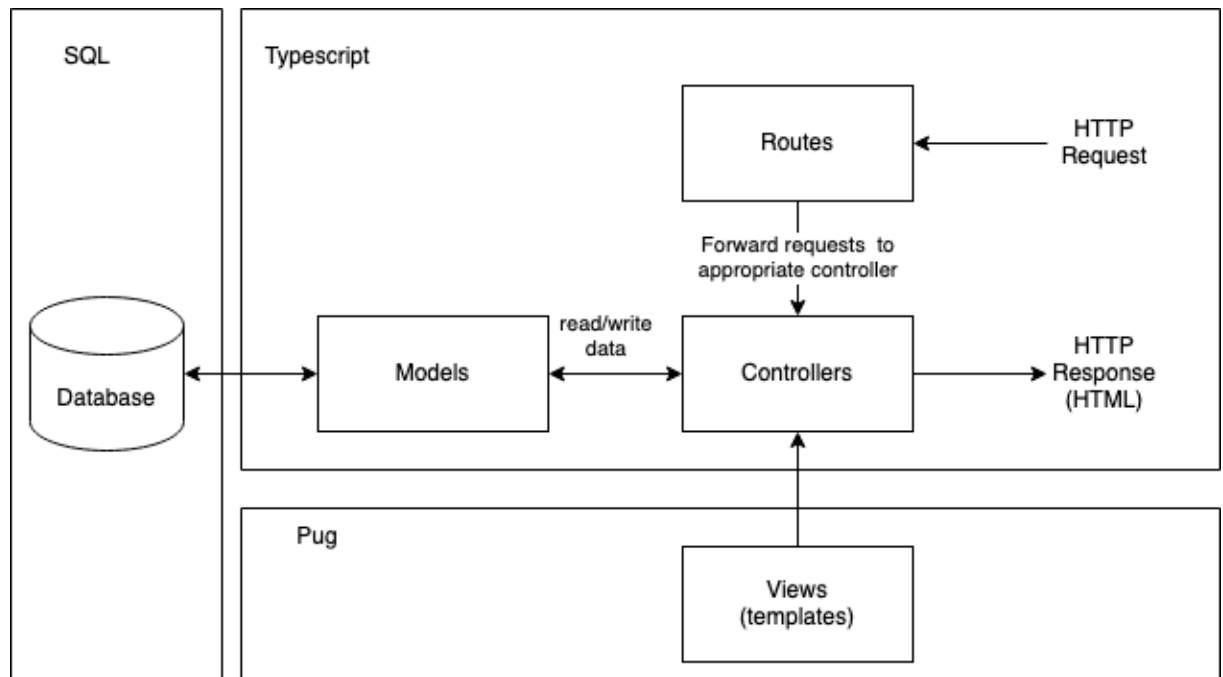
4.4. MVC serverio architektūra

Sistemos serverio dalyje naudojamas *Model–View–Controller* architektūrinis modelis. Šis architektūrinis modelis susideda iš trijų dalių:

- *Model* – sluoksnis atsakingas už duomenų bazės valdymą.
- *View* – sluoksnis atsakingas už duomenų atvaizdavimą vartotojui.
- *Controller* – sluoksnis atsakingas už įvesties priėmimą, bei *Model* ir *View* sujungimą, šioje dalyje nusprendžiama, kurie duomenys turi būti rodomi vartotojui.

Naudojant šį architektūrinį modelį kuriama sistema tampa lengviau suprantama, nes funkcionalumas paskirstomas į atitinkamas programos dalis, taip pat sistema tampa lengviau palaikoma,

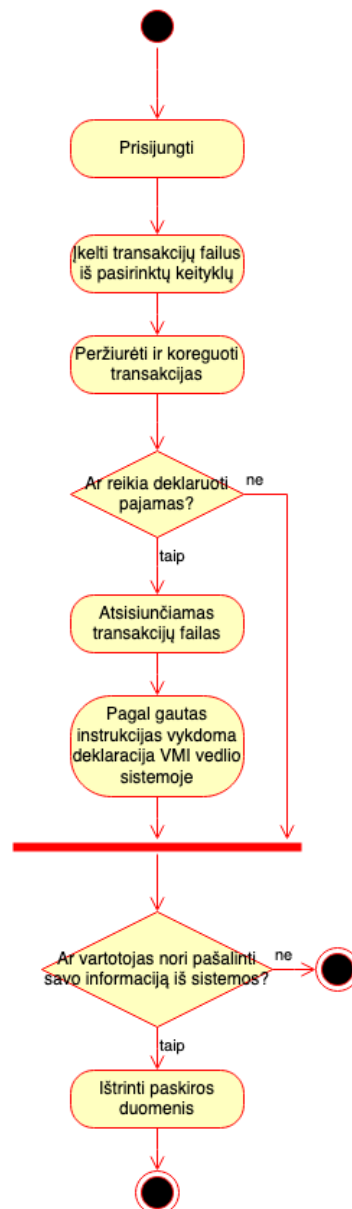
nesunku pakeisti specifinę programos dalį, nekeičiant kitų, bei galima lengvai pridėti naujo funkcionalumo, nes sistema išskaidyta dalimis.



5 pav. Sistemos MVC architektūros diagrama, rodanti naudojamas programavimo kalbas, skirtingose sistemos dalyse (dėl aiškumo naudojama programinio kodo terminologija) [3]

Diagramoje (žr. 5 pav.) vaizduojamas atskirų serverio sluoksnių veikimas tarpusavyje, nuo vartotojo kreipimosi į sistemą (*HTTP Request*) iki sistemos atsako vartotojui pateikimo (*HTTP Response*). Pirmiausia sistemos *Route* sluoksnis pagal vartotojo tinklapio adreso maršrutą, perduoda užklausą atitinkamam *Controller* valdikliui, jei užklausa reikalauja informacijos iš duomenų bazės, *Controller* kreipiasi į *Model* dalį, kuri atlieka reikiamą užklausą į duomenų bazę. *Controller* daliai atlikus reikiamus veiksmus su duomenimis, *View* sluoksnis papildomas reikiamais duomenimis ir vartotojui grąžinama svetainės lango forma. Taip pat diagramoje vaizduojama kiekvienoje dalyje naudojama programavimo kalba, sistemoje *Routes* ir *Controllers* dalies kūrimui naudojama *Express* biblioteka, o *Models* dalies kūrimui *TypeORM* biblioteka, kuri automatiškai generuoja reikiamas *SQL* užklausas, taigi pačioje programoje *SQL* kalbos naudoti nereikia, užtenka naudoti bendrą *Typescript* kalbą. *Views* dalyje naudojama *Pug* kalba, kurią *Controller* sluoksnis prieš grąžindama vartotojui atsakymą paverčia į *HTML*.

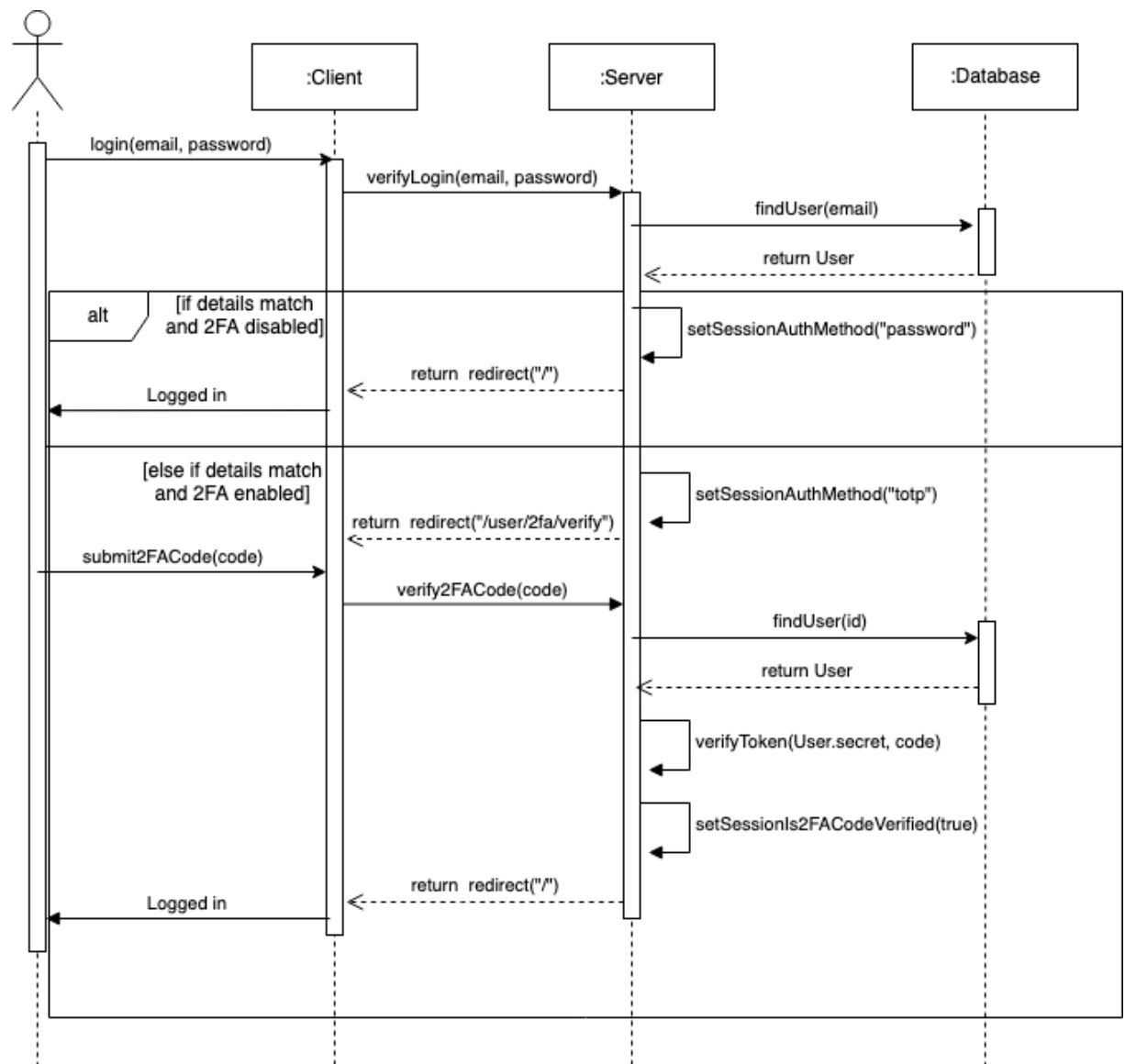
4.5. Vartotojo mokesčių deklaravimas naudojant sistemą



6 pav. Vartotojo mokesčių deklaravimo proceso naudojant sistemą veiklos diagrama

Veiklos diagramoje (žr. 6 pav.) vaizduojamas procesas, kai vartotojas pasitelkdamas sistemą deklaruoja savo kriptovaliutų pelno mokesčius. Vartotojas prisijungęs prie savo paskyros, įkelia transakcijų informaciją iš naudojamų kriptovaliutų keityklų. Peržiūri įkeltų transakcijų sąrašą ir jei yra trūkstamas išlaidų pagrindų ar neatitikimų vartotojas gali pridėti naujų transakcijų ar pakeičti esamas transakcijas. Įsitikinęs, jog informacija teisinga, vartotojas patikrina ar jam deklaruoti pajamas reikia, jei ne, vartotojas gali uždaryti sistemą, o jei deklaruoti būtina, vartotojas gali atsisiųsti transakcijų failą, bei pradėti deklaruoti pajamas pagal pateiktus duomenis ir instrukcijas. Baigęs naudotis sistema vartotojas turi pasirinkimą ištrinti savo asmeninius duomenis iš sistemos.

4.6. Vartotojo prisijungimas



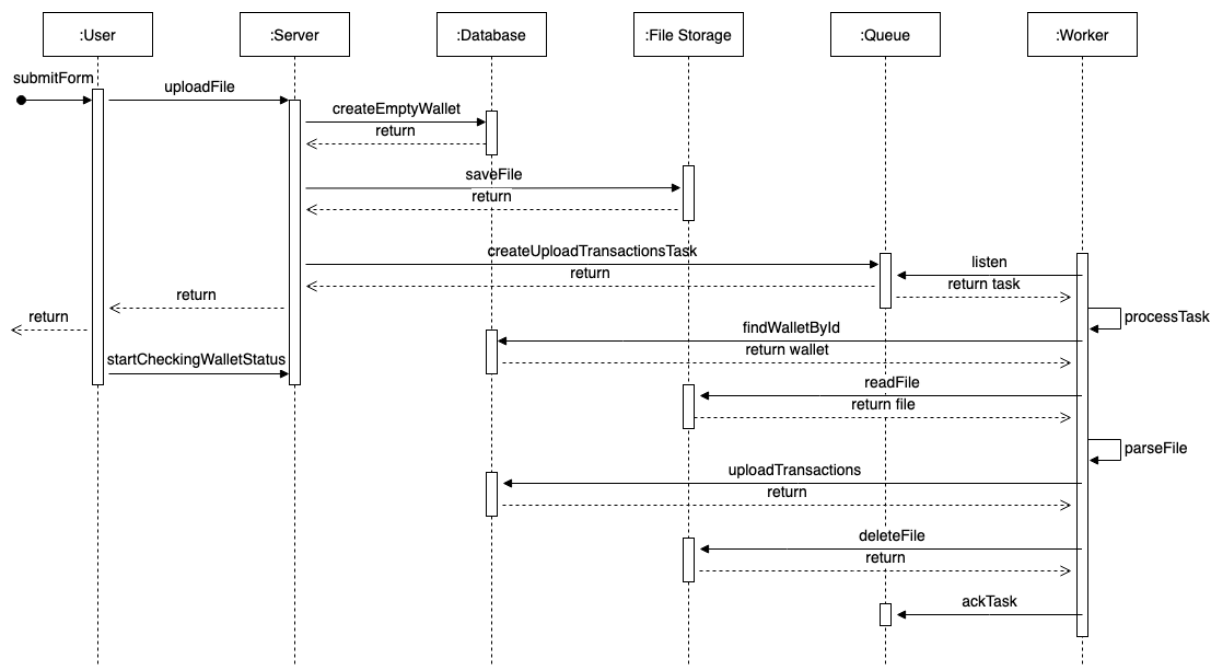
7 pav. Vartotojo prisijungimo prie sistemos sekos diagrama (dėl aiškumo naudojama programinio kodo terminologija)

Vartotojas gali būti anketos nustatymuose pridėjęs papildomą dviejų faktorių autentifikacijos saugumo sluoksnį, taigi vartotojų prisijungimo procesai skiriasi priklausomai nuo vartotojo saugumo lygio poreikio (žr. 7 pav.). Vartotojo saugumas gali būti pagerinamas pridėdant dviejų faktorių autentifikacijos procesą, kuris norint prisijungti prie sistemos iš vartotojo reikalauja ne tik anketos prisijungimo duomenų, bet ir 6 skaičių kombinacijos, kuri yra generuojama vartotojo telefone ir atnaujinama kas 30 sekundžių. Tačiau, bet kuriuo atveju vartotojas turi suvesti savo elektroninio pašto adresą ir slaptažodį, serveriui patikrinus šios įvesties teisingumą, tikrinama ar vartotojas yra pridėjęs prie savo anketos dviejų faktorių autentifikacijos sluoksnį, jei šis funkcionalumas nenaudojamas, serveris prideda prie vartotojo sesijos informaciją, jog vartotojo prisijungimo būdas yra slaptažodis, vartotojas sėkmingai prisijungė ir gali naudotis sistemos funkcionalumu. Kitu atveju, jei vartotojas prie savo anketos yra pridėjęs dviejų faktorių autentifikaciją, serveris sesijoje išsaugo informaciją, jog vartotojo prisijungimo metodas yra dviejų faktorių autentifikacija ir kol kas vartotojas dar nesuvedė teisingo dviejų faktorių autentifikacijos kodo į sistemą, vartotojas nukreipiamas

į dviejų faktorių autentifikacijos kodo patvirtinimo puslapi, kuriame privalo suvesti teisingą kodą ir tik po teisingo kodo įvesties vartotojo sesijoje bus nurodoma, jog vartotojas jau suvedė teisingą dviejų faktorių autentifikacijos kodą ir šis vartotojas taps prisijungusiu, bei galės naudotis sistemos funkcionalumu.

Šis dviejų faktorių autentifikacijos metodas yra neprivalomas, tačiau vartotojui pasirinkus jį naudoti, sistemoje sukuriama ir išsaugoma slapta frazė, kurią vartotojas gali išsaugoti savo telefone nuskaitant sistemoje pateiktą *QR* kodą naudodamas dviejų faktorių autentifikacijai skirtą programėlę, pavyzdžiui *Google Authenticator* ar *Authy*. Išsaugojus slaptą frazę telefone, pasirinktoje programėlėje bus generuojami laiku pagrįsti vienkartiniai slaptažodžiai, kurių galiojimo trukmė yra 30 sekundžių. Prisijungimo metu įvestas kodas lyginimas su serverio pusėje, pagal vartotojo slaptą frazę generuojamu kodu ir jei šie kodai sutampa vartotojo prisijungimas yra patvirtinamas.

4.7. Transakcijų failo įkėlimas

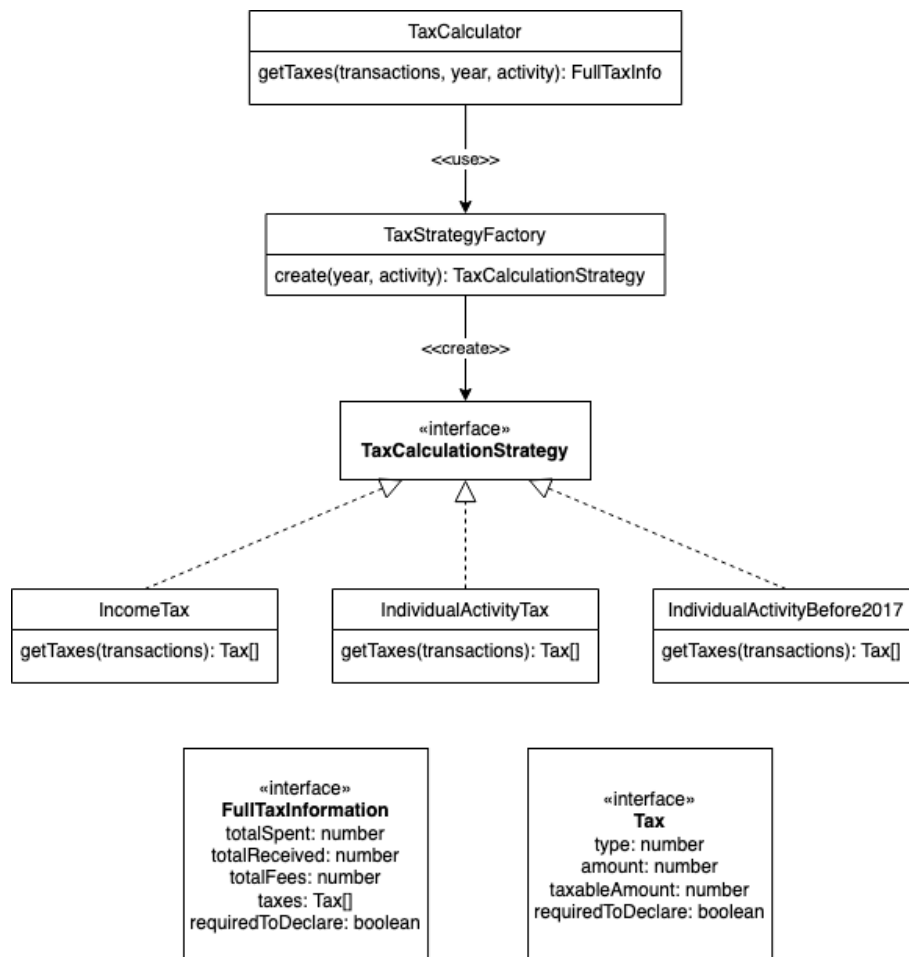


8 pav. Transakcijų failo įkėlimo sekos diagrama (dėl aiškumo naudojama programinio kodo terminologija)

Vartotojo transakcijų failo įkėlimo procesas (žr. 8 pav.) prasideda vartotojui pasirinkus transakcijų failą iš savo kompiuterio ir pateikus formą. Sistema gauna *multipart/form-data* duomenų užklausa ir nuskaito failą, duomenų bazėje sukuria naują tuščios piniginės objektą, su statusu *pending*, jog būtų aišku, kad ši piniginė dar yra apdorojama ir į ją artimiausiu metu bus įkeliamos transakcijos, taip pat gautas failas yra išsaugomas bendroje failų saugojimo aplinkoje, prie kurios prieigą turi ir *server* ir *worker* procesai. Tada įkeliamas naujas užduotis į eilę su piniginės identifikacijos laukeliu ir failo pavadinimu, po šių veiksmų vartotojui iškart grąžinamas atsakas, jog failo apdorojimas ir transakcijų įkėlimas pradėtas. Tuo tarpu *worker* procesas gavęs naują užduotį pradeda ją vykdyti, pirmiausia gauna piniginės informaciją pagal identifikacijos laukelį, ši informacija padeda pasirinkti tinkamą apdorojimo klasę. Sėkmingai gavus informaciją apie piniginę procesas nuskaito transakcijų failo duomenis, juos apdoroja ir gautą transakcijų informaciją įkelia į duomenų bazę, bei pakeičia piniginės statusą į *finished*, ištrina apdorotą failą ir pašalina užduotį

iš eilės. Kadangi faile gali būti dešimtys tūkstančių eilučių apie vartotojo atliktas transakcijas šis procesas gali užtrukti keletą minučių, tačiau pasitelkiant eilę serveris nėra apkraunamas failo apdorojimo darbais, o vartotojas per kelias sekundes gauna atsaką iš sistemos ir toliau gali naudotis sistema, bei neprivalo laukti kol įkeltas failas bus apdorotas. Tuo tarpu vartotojo naršyklė kas keletą sekundžių tikrina įkeltos piniginės statusą ir praneša, kai piniginės apdorojimas yra baigtas.

4.8. Vartotojo mokesčių skaičiavimas

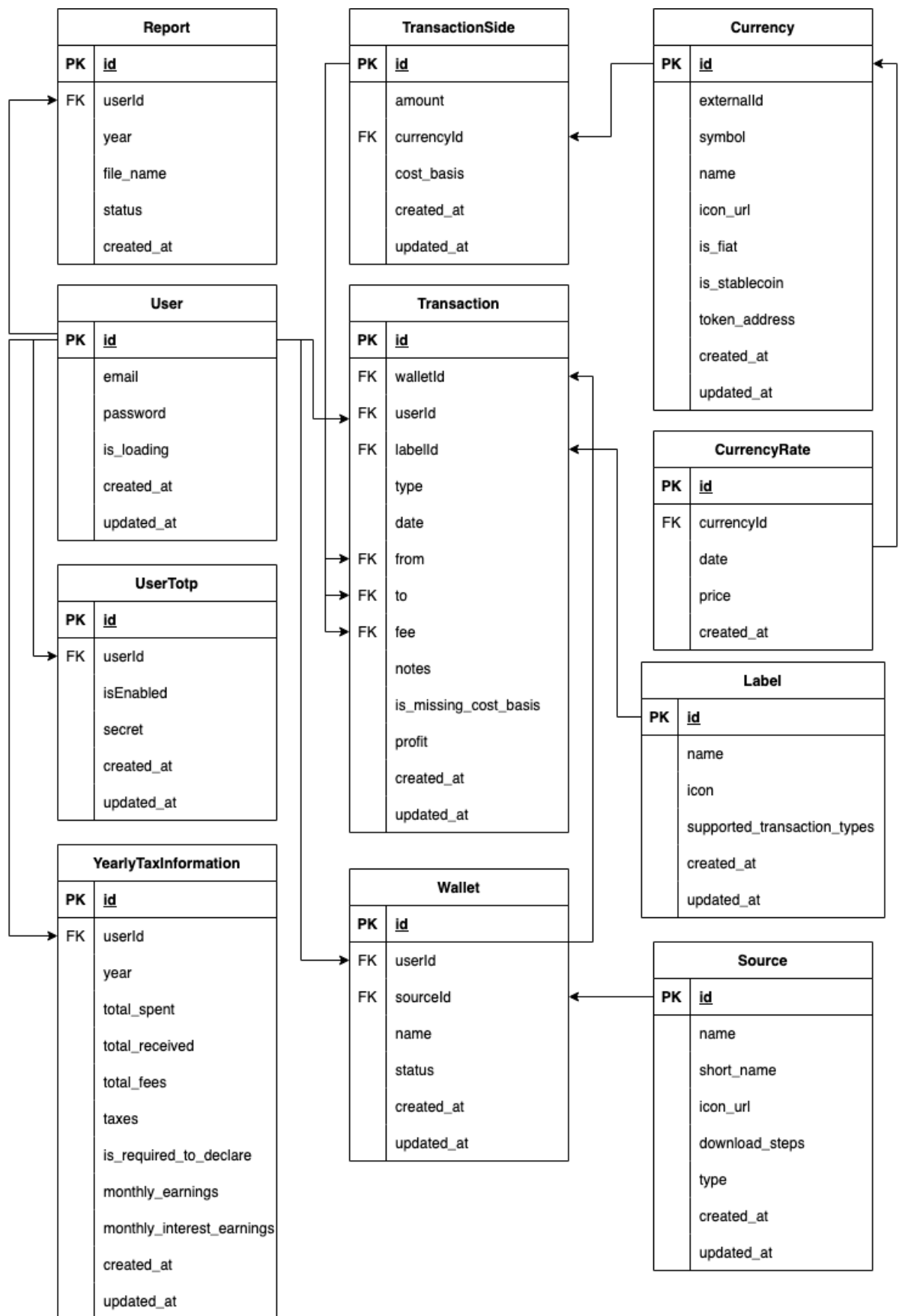


9 pav. Vartotojo mokesčių skaičiavimui naudojamų klasių diagramą (dėl aiškumo naudojama programinio kodo terminologija)

Pirmiausia norint apskaičiuoti vartotojo metinius kriptovaliutų mokesčius, reikia gauti informaciją apie vartotojo vykdomą veiklą ir atliktų transakcijų sąrašą (žr. 9 pav.). Ši informacija perduodama *TaxCalculator* klasei, kuri pagal gautus transakcijų duomenis apskaičiuoja vartotojo metines pajamas ir išlaidas, tada naudojant *TaxStrategyFactory* klasę, pagal pasirinktus metus ir vartotojo atliekamos veiklos rūšį, sukurama atitinkama mokesčių skaičiavimo strategijos klasė. Ši klasė pagal gautų transakcijų informaciją apskaičiuoja vartotojo mokesčius, kuriuos reikia deklaruoti už pasirinktus metus. Apskaičiuoti mokesčiai grąžinami *Tax* masyvo formatu, kuriame nurodomas mokesčio tipas, apmokestintų pajamų dydis, mokėtino mokesčio dydis, bei ar mokesčius privaloma deklaruoti. Mokesčių informaciją grąžinant masyvu, lengva pridėti naujų mokesčių prie mokesčių strategijos, taip pat mokesčių strategijos gali naudoti tą patį *TaxCalculationState-*
gy tipą, pavyzdžiui gyventojų pajamų mokesčių strategija grąžintų mokesčių masyvą su kito turto

pardavimų ir palūkanų mokesčių tipais, o individualios veiklos mokesčių strategija su gyventojų pajamų, valstybinio socialinio draudimo ir privalomojo socialinio draudimo mokesčių tipais, taip pat gavus mokesčius masyvo formatu, juos lengviau atvaizduoti ir sistemos priekinėje dalyje, nebūtina nurodyti specifinių vaizduojamų mokesčių, o užtenka masyvo iteracijos. Šių klasių sudaryme buvo atsižvelgta ir į ateities pokyčių riziką, taigi pritaikyti naujus mokesčių pokyčius būtų nesudėtinga, pasikeitus mokesčių dydžiui, tereikia pridėti papildomą mokesčių skaičiavimo strategiją, kuri atitiktų *TaxCalculationStrategy* tipą ir aprašyti jo naudojimo sąlygas nurodant paveiktą vartotojo veiklą ir metus nuo kurių pokytis įsigaliojo *TaxStrategyFactor* klasėje.

4.9. Duomenų bazės esybių–ryšių modelis



10 pav. Duomenų bazės esybių–ryšių modelio diagrama

Duomenų bazėje saugomos 11 lentelių (žr. 10 pav.):

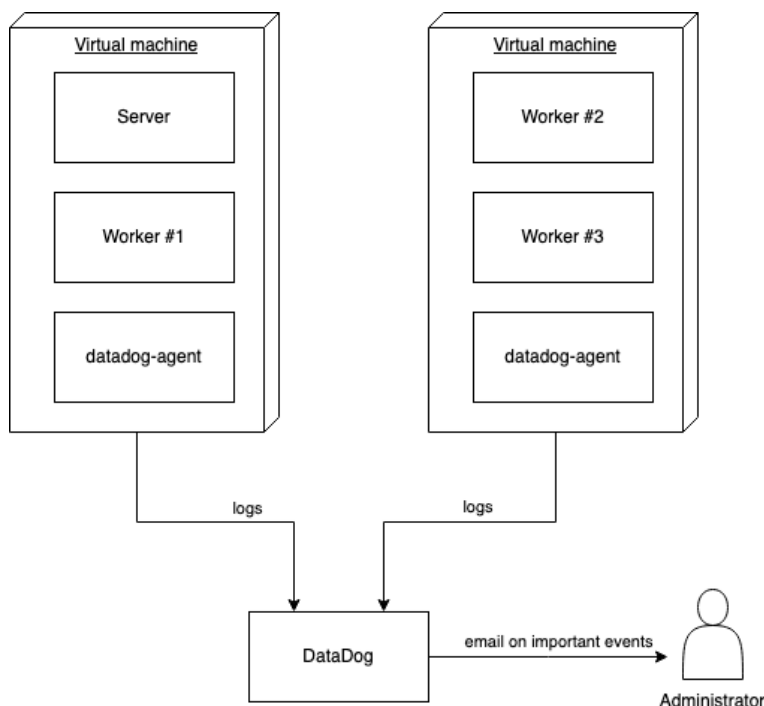
- User – vartotojo prisijungimo informacija
- UserTotp – vartotojo dviejų veiksmų autentifikavimo informacija
- Wallet – vartotojo, įkeltų iš to pačio šaltinio, transakcijų grupė
- Source – kriptovaliutų keitykla ar piniginė, iš kurios vartotojai gali importuoti transakcijas
- Transaction – atlikta kriptovaliutų operacija, tai gali būti pervedimas, gavimas ar keitimas
- TransactionSide – Transakcijos objektas parodantis kiek ir ko transakcijoje buvo gauta ar išleista
- Label – informacija atskleidžianti pajamų priežastį (palūkanų mokėjimas, kasimas, dovana ar kt.)
- Currency – valiuta, tai gali būti tradicinė valiuta ar kriptovaliuta
- CurrencyRate – valiutos vidutinis kursas eurai, tam tikrą dieną
- Report – informacija apie sistemos vartotojui sugeneruotą transakcijų išrašo failą
- YearlyTaxInformation – vartotojo metinė mokesčių informacija, parodo kiek vartotojas išleido kriptovaliutų pirkimams ar pardavimams, kokia suma transakcijų mokesčiams buvo sumokėta, bei kiek ir kokių mokesčių vartotojui reikia sumokėti

Duomenų bazėje vartotojo informacija yra saugoma *User* lentelėje, prisijungimas vyksta naudojant elektroninio pašto adresą ir slaptažodį. Slaptažodis duomenų bazėje laikomas užkoduotas, vartotojui norint padidinti savo paskyros saugumą, galima įgalinti dviejų veiksmų autentifikavimą, tokiu atveju vartotojas savo telefone pagal gautą slaptą frazę, generuoja 6 skaitmenų kodus, kuriuos privalo pateikti per kiekvieną prisijungimą. Slapta frazė ir informacija apie vartotojo dviejų veiksmų autentifikavimo statusą laikoma *UserTotp* lentelėje. Įkeltos transakcijos iš tam tikros kriptovaliutų keityklos failo laikomos sugrupuotos *Wallet* lentelėje, šiai transakcijų grupei suteikiamas pavadinimas, padedantis vartotojui atskirti jų pobūdį. Sistemoje palaikomų kriptovaliutų keityklų sąrašas laikomas *Source* lentelėje, sutrumpintas keityklos pavadinimas leidžia sistemai pasirinkti reikiamą failo apdorojimo funkciją, kai šis yra įkeliamas. Transakcijos yra laikomos *Transaction* lentelėje, išsaugomas transakcijos tipas, įvykdymo data, bei kas ir kiek buvo išleisto, gauta, sumokėta mokesčio forma. Platesnė informacija apie transakcijos turinį saugoma *TransactionSide* lentelėje, kurioje išlieka naudojamos valiutos informacija ir valiutos kiekio vertė transakcijos atlikimo momentu. Gauti valiutos vertę, tam tikru laiko momentu padeda lentelė *CurrencyRate*, kurioje saugomi istoriniai valiutų kursai. Informacija apie sugeneruotus vartotojo transakcijų išrašus saugoma *Report* lentelėje, kuri taip pat saugo failo pavadinimą, pagal kurį šis pateikiamas vartotojui atsisiųsti. Tuo tarpu informacija apie vartotojo mokesčius pasirinktiems metams yra laikoma *YearlyTaxInformation*, kurią atnaujinant *User* lentelės laukelis *is_loading* yra pažymimas *true*, jog vartotojui būtų atvaizduojamas krovimo veiksmas.

4.10. Registravimas ir klaidų tvarkymas

Naudojant pasirinktą architektūrinį modelį, klaidų registravimas ir radimas tampa sudėtingesnis, kadangi vartotojo veiksmo sukelta klaida sistemoje gali atsitikti, ne vienoje sistemos dalyje, tuo tarpu ir ne toje pačioje virtualioje mašinoje. Taigi registruojant klaidas, būtina sukurti identifikacijos kodą, tam tikram vartotojo veiksmui, kuris atskirtų sistemos įrašus, vykdytus šiam veiksmui įgyvendinti. Taip pat įrašai turi būti surinkti į vieną centralizuotą vietą, jog nereikėtų rinkti sistemos išvesties iš kelių skirtingų virtualių mašinų, o šių įrašų peržiūra būtų paprasta ir aiški.

Sistemoje išvesties registravimui naudojama *log4js* biblioteka. Serverio užklausų registravimui naudojama papildyta *log4js.connectLogger* funkcija, kuri automatiškai suteikia kiekvienai vartotojo užklausai unikalų identifikacijos numerį, pagal kurį galima atskirti vartotojo veiksmus ir juose įvykusias klaidas. Išvesties centralizavimui ir apdorojimui naudojamas *DataDog* servisas (žr. 11 pav.), kurio pagalba sistemos išvestis surenkama į vieną vietą, o užregistravus klaidą, siunčiamas elektroninis laiškas sistemos administratoriui, jog iškilusią klaidą būtų galima kuo greičiau ištaisyti. Tačiau šis servisas nuskaityto *JSON* duomenis iš failo, o programavimo metu, *JSON* duomenų išvestį skaityti nepatogu, taigi sukuriama atskiri registravimo srautai, priklausomai nuo pasirinktos *NODE_ENV* aplinkos kintamojo reikšmės. Pasirinkus *development* reikšmę duomenys atvaizduojami terminale žmogui įskaitomu tekstu, taip pat duomenų srautas yra įrašomas darbo katalogo *logs/app.log* faile, jei kūrimo metu reiktų peržiūrėti didesnius kiekius duomenų, kuriuos tikrinti terminale būtų nepatogu. Tuo tarpu pasirinkus *production* reikšmę, išvesties srautas nukreipiamas į */var/log/coin-tax/app.log* failą, iš kurio virtualioje mašinoje veikiantis *datadog-agent* procesas įkelia išvesties duomenis į centralizuotą saugyklą, kurioje duomenys lengvai perskaitomi ir analizuojami.



11 pav. Sistemos išvesties registravimo diagrama (dėl aiškumo naudojama programinio kodo terminologija)

5. Duomenų gavimas ir apdorojimas

Šiame skyriuje aprašomi duomenų, reikalingų sistemos veikimui, gavimo procesai. Pirmiausia sistemos veikimui reikalingi vartotojo transakcijų duomenys gaunami iš skirtingų kriptovaliutų keityklų, taip pat sistemoje turi būti palaikomų kriptovaliutų keityklų ir transakcijų etikečių sąrašai, bei galimybė lengvai juos atnaujinti, galiausiai reikalingi nacionalinių valiutų ir kriptovaliutų sąrašai su istorinėmis kainomis, naudojami nustatant transakcijos vertę eurai.

5.1. Vartotojo transakcijų įkėlimas

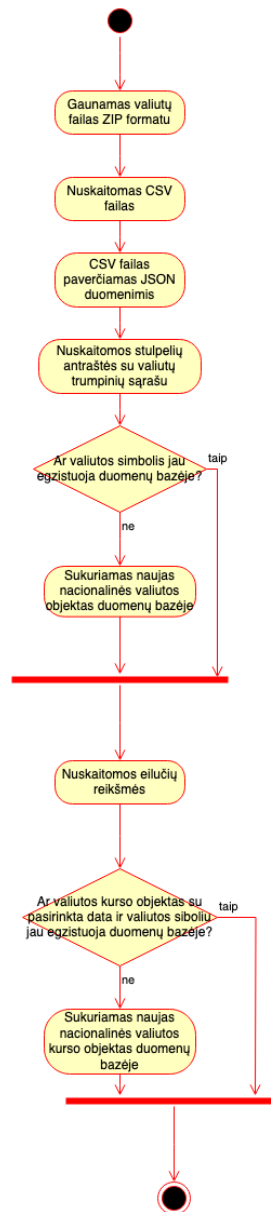
Vartotojų transakcijų duomenų gavimas iš kriptovaliutų keityklų gali būti vykdomas dviem būdais: vartotojui atsisieniūčiant transakcijų išrašo failą iš keityklos ir įkeliant šį failą į sistemą arba vartotojui suteikiant sistemai prieigos teises prie vartotojo keityklos anketos informacijos, perduodant *API* raktus ar prisijungiant prie sistemos su keityklos anketa, jei tokia galimybė yra, ir sistemai siunčiant užklausas duomenų gavimui į keityklos *API*. Kuriant *OAuth* vartotojo prisijungimo su kriptovaliutų keityklos anketa langą, vienas iš keityklų reikalavimų yra nuosavo domeno turėjimas, dėl to pradinis internetinės informacinės sistemos modelis daromas tik su transakcijų įkėlimo iš failo funkcionalumu, nes tai yra paprastas ir aiškus būdas įkelti transakcijų istoriją, taip pat vartotojas nerizikuoja savo keityklos anketa ir jos lėšomis, suteikdamas *API* raktus sistemai.

5.2. Sistemoje palaikomų keityklų ir transakcijų etikečių atnaujinimas

Vienas iš pagrindinių reikalavimų sistemai yra lengvas naujų kriptovaliutų keitykklų ar servisų palaikymo funkcionalumo pridėjimas, taigi ir šių sąrašas, bei susijusi informacija sistemoje turi būti lengvai atnaujinami. Šiam reikalavimui sukuriamas naujas pagalbinis kodo fragmentas, kuriame aprašomi palaikomų kriptovaliutų keityklų ir transakcijų etikečių objektų sąrašai, o kodą įvykdžius atlikti lokalūs pakeitimai įkeliami į duomenų bazę. Tokiu būdu nereikia daryti pakeitimų prisijungiant tiesiogiai prie duomenų bazės.

5.3. Valiutų kursų nustatymas

5.3.1. Nacionalinių valiutų kursų nustatymas



12 pav. Nacionalinių valiutų sąrašo ir kursų atnaujinimo veiklos diagrama

Pinigų įnešimas į kriptovaliutų keityklą ir kriptovaliutų pirkimas ne visada atliekamas eurais, o tam gali būti naudojama ir kita nacionalinė valiuta. Valiutų kursų informacijos kriptovaliutų keityklos nepateikia, taigi sistemoje reikia turėti informaciją apie esamus ir istorinius valiutų kursus. Šiai informacijai gauti naudojami oficialūs Europos centrinio banko viešai prieinami 41 valiutos kasdieniai kursai nuo 1991 m. Sausio 4 dienos. Ši informacija pateikiama CSV failo formatu, taigi sistemoje sukuriamas naujas modulis, kuris gauna ir nuskaito CSV failą su valiutų kursų informacija ir sukuria, bei atnauja duomenų bazėje reikiamus valiutų ir valiutų kursų objektus ir ryšius (žr. 12 pav.) [4].

5.3.2. Kriptovaliutų kursų nustatymas

Transakcijų išrašuose, kuriuos įkelia vartotojas taip pat trūksta kriptovaliutų kainos jų keitimo metu, taigi sistema turi turėti galimybę gauti palaikomų kriptovaliutų kainas tam tikru laiko momentu. Lietuvos kriptovaliutų pajamų mokesčių deklaravimo reikalavimuose nurodyta, jog vartotojas pats gali pasirinkti šių kainų nustatymo šaltinį, taigi šiems duomenims gauti naudojamas *CoinGecko API*, kurio pagalba, galima nustatyti daugiau nei 12 tūkstančių kriptovaliutų dienes kainas. Tačiau užklausos formavimui *CoinGecko* reikalauja pateikti vidinį *CoinGecko* svetainės kriptovaliutos identifikacijos parametą. Taigi esamas valiutų kurso įkėlimo į duomenų bazę modulis papildomas funkcionalumu, kuris gauna palaikomų valiutų sąrašą iš *CoinGecko API* ir išsaugo jį į duomenų bazę su kriptovaliutos pavadinimu, trumpiniu, bei vidiniu *CoinGecko* identifikatoriumi. Tada sistemoje sukuriamą funkcija, kuri pagal kriptovaliutos simbolį ir datą gražina kriptovaliutos kainą, gautą iš *CoinGecko API*. Kol kas naudojama nemokama *CoinGecko API* versija, kuri leidžia atlikti 10-30 užklausų per minutę, šis limitas labai apriboja transakcijų apdorojimo sistemoje greitį, tačiau dabartinei sistemos versijai to užtenka, o atsiradus poreikiui, už papildomą kainą, galima lengvai padidinti šį skaičių.

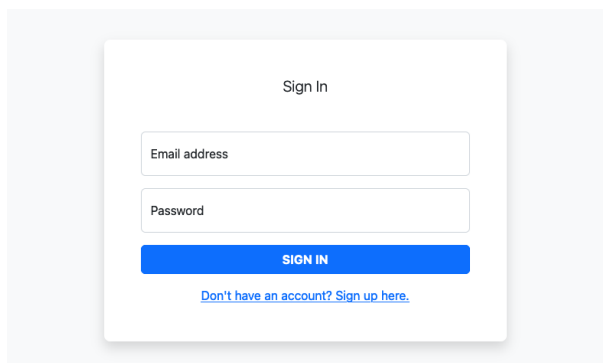
6. Programavimo darbai

Šiame darbe minėtoms problemoms išspręsti, kuriama svetainė *CoinTax*, kurios pagalba Lietuvos gyventojai galės palengvinti savo kriptovaliutų mokesčių deklaravimo procesą. Sistemos kūrimas suskirstytas į tam tikrus skirtingus darbų eiliškumo etapus, kadangi daugelis kuriamo funkcionalumo įgyvendinimo reikalauja jau tam tikro esamo sistemos funkcionalumo. Programos kūrimas pradedamas nuo vartotojo prisijungimo ir registracijos, tada sukuriama transakcijų įkėlimo iš pasirinktų kriptovaliutų keityklų funkcionalumas, bei šių transakcijų saugojimas sistemoje. Vėliau kuriama transakcijų peržiūra ir redagavimas. Esant funkcionalumui, kuris leidžia vartotojui sistemoje turėti teisingą savo transakcijų sąrašą, kuriamas pelno apskaičiavimo ir atvaizdavimo funkcionalumas. Sekančiame etape įgyvendinamas transakcijų suvestinės generavimas ir atsisiuntimas, o galiausiai kuriami deklaravimo gidai, leidžiantys nieko apie šią svetainę ar mokesčius nežinantiems vartotojams lengvai atlikti kriptovaliutų mokesčių deklaravimą naudojant *CoinTax* sistemą. Įgyvendinus visą sistemos funkcionalumą, šis funkcionalumas testuojamas su realiais duomenimis.

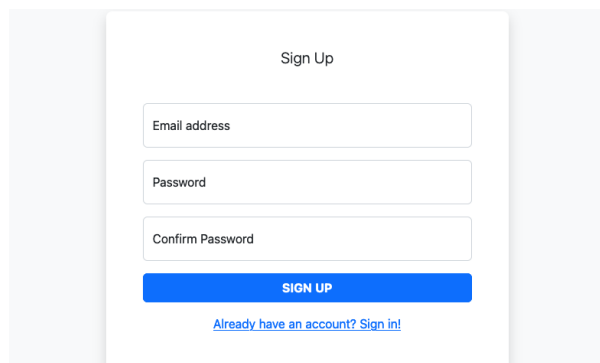
6.1. Paskyros valdymas ir duomenų šalinimas

6.1.1. Registracija ir prisijungimas

Pirmojo etapo metu, kuriamas vartotojo prisijungimas, bei registracija (žr. 13, 14 pav.). Prisijungimui naudojamas vartotojo elektroninio pašto adresas ir slaptažodis. Norint užtikrinti vartotojo duomenų saugumą, kuriama validacija leidžianti vartotojui pasirinkti tik saugų slaptažodį, kurį turi sudaryti ne mažiau nei 8 simboliai, kurių tarpe turi būti bent viena didžioji raidė, bent vienas skaičius ir bent vienas simbolis. Sistemoje slaptažodžiai, vartotojui užsiregistravus, laikomi užšifruoti, naudojant *Blowfish* maišos algoritmu paremtą funkciją, nustatant *salt* vertę 12. Vartotojui prisijungiant gaunamas originalus slaptažodis šifruojamas ta pačia funkcija ir gauta vertė lyginama su esančiu šifru duomenų bazėje, tokiu būdu originali slaptažodžio reikšmė niekada nėra išsaugoma sistemoje. Taip pat pridedamas funkcionalumas leidžiantis vartotojui atsisiuntus į savo telefoną *Google Authenticator* ar *Authy* programėlę, pridėti papildomą dviejų faktorių autentifikacijos saugumo sluoksnį, pasinaudojus šiuo funkcionalumu vartotojui kiekvieną kartą prisijungiant reikės įvesti 6 skaitmenų kodą, kurį vartotojas galės rasti savo pasirinktoje telefono programėlėje. Sukūrus galinės dalies prisijungimo ir registracijos funkcionalumą, kuriami vartotojo prisijungimo ir registracijos langai.

The image shows a 'Sign In' form. It has a title 'Sign In' at the top. Below it are two input fields: 'Email address' and 'Password'. At the bottom, there is a blue button labeled 'SIGN IN' and a link that says 'Don't have an account? Sign up here.'

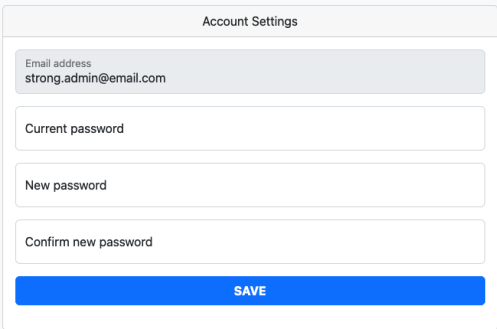
13 pav. Prisijungimas

The image shows a 'Sign Up' form. It has a title 'Sign Up' at the top. Below it are three input fields: 'Email address', 'Password', and 'Confirm Password'. At the bottom, there is a blue button labeled 'SIGN UP' and a link that says 'Already have an account? Sign in!'

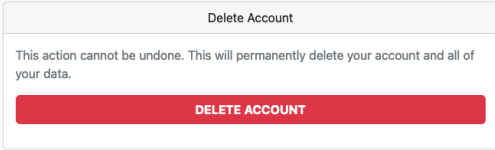
14 pav. Registracija

6.1.2. Paskyros redagavimas ir šalinimas

Vartotojui atlikus mokesčių deklaravimą dažniausiai sistema naudotis nebereikės iki kitų metų, taigi vartotojas atlikęs norimus darbus svetainėje, gali pasirinkti ištrinti savo paskyrą, bei visus savo įkeltus duomenis sistemoje (žr. 16 pav.). Šiam funkcionalumui sukuriama profilio nustatymų puslapis (žr. 15 pav.), kuriame taip pat, pridedama slaptažodžio keitimo ir dviejų faktorių autentifikacijos įjungimo ar išjungimo galimybė. Paspaudus anketos ištrynimo mygtuką iš duomenų bazės ištrinama vartotojo anketa, bei visa su vartotoju susijusi informacija.



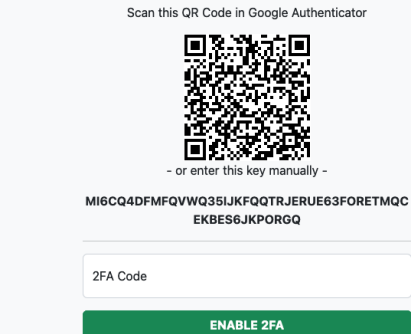
15 pav. Vartotojo anketos duomenų redagavimas



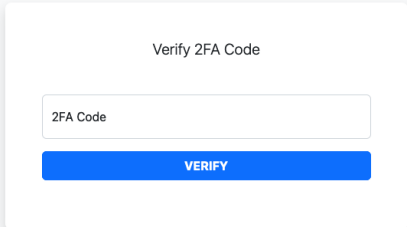
16 pav. Vartotojo anketos ir duomenų pašalinimas

6.1.3. Dviejų faktorių autentifikacija

Vartotojas norėdamas savo paskyrai suteikti daugiau saugumo gali pasirinkti įgalinti dviejų faktorių autentifikaciją, tokiu atveju vartotojui sugeneruojama slapta seka, kuri išsaugoma duomenų bazėje, bei parodoma vartotojui (žr. 17 pav.). Šią frazę įvedęs į pasirinktą dviejų veiksmų autentifikavimo programėlę vartotojui suteikiamas kas 30 sekundžių atsinaujinantis 6 skaitmenų kodas, suvedus teisingą kodą į sistemos langą vartotojo paskyrai yra įjungiamas dviejų faktorių autentifikacijos funkcionalumas ir kiekvieną kartą vartotojas prisijungdamas prie savo paskyros papildomai privalo suvesti teisingą 6 skaitmenų dviejų faktorių autentifikacijos kodą (žr. 18 pav.). Šis funkcionalumas gali būti išjungiamas, jei vartotojui papildomo saugumo sluoksnio nebereikia, tačiau tam taip pat turi būti suvedamas teisingas vartotojo dviejų faktorių autentifikacijos kodas.



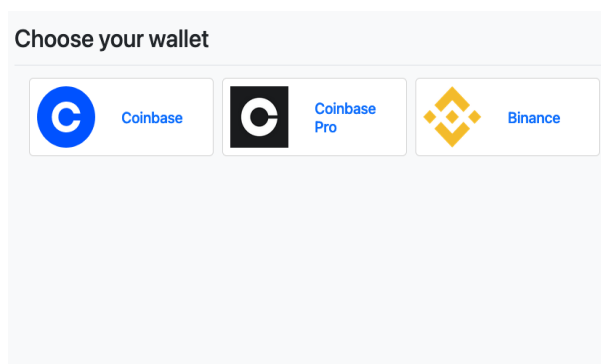
17 pav. Dviejų faktorių autentifikacijos paskyrai įgalinimas



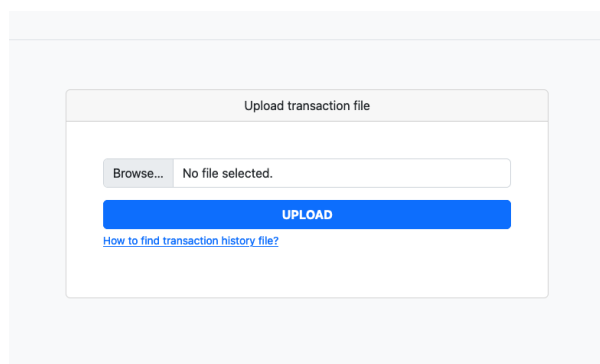
18 pav. Dviejų faktorių autentifikacijos kodo patvirtinimas prisijungiant

6.2. Duomenų apie transakcijas įkėlimas

Kitame etape įgyvendinamas transakcijų įkėlimas iš kriptovaliutų keityklų, pirmiausia duomenų bazėje sukuriama palaikomų kriptovaliutų keityklų objektai, šių objektų atvaizdavimas kliento pusėje leidžia pasirinkti vartotojui norimą šaltinį iš kurio bus įkeliami kriptovaliutų transakcijų informacija (žr. 19 pav.). Kiekvienai palaikomai kriptovaliutų keityklai kuriamas *parser* klasė, kuri apdoroja įkeliamus duomenis ir paverčia juos į lokalų, bendrą, sistemoje naudojamą formatą. Taip pat kiekvienam šaltiniui sukuriamas vadovas, kokius žingsnius palaipsniui reikia atlikti, norint į sistemą įkelti transakcijų informaciją (žr. 20 pav.). Vartotojo įkeliamas failas sistemoje sukuria naują pinigines objektą, kuris vartotojui padeda atskirti transakcijų grupę įkeltą iš tam tikros kriptovaliutų keityklos, tam tikru laiko momentu.



19 pav. Palaikomų kriptovaliutų keityklų sąrašas, duomenų įkėlimui



20 pav. Transakcijų istorijos failo įkėlimas pasirinktai keityklai

6.3. Transakcijų peržiūra ir redagavimas

Transactions									
Show 10 entries		Search:							
Date	Wallet	Type	Spent	Received	Fee	Profit/Loss	Label	Errors	Actions
2022-01-05	Coinbase Pro	Buy	10000 EUR	1 BTC	60 EUR		No label		
2022-01-10	Coinbase Pro	Buy	15000 EUR	1 BTC	90 EUR		No label		
2022-02-15	Coinbase Pro	Exchange	1.5 BTC	3 ETH	0.018 BTC	-10087.77€	No label		
2022-04-01	Coinbase Pro	Sell	0.482 BTC	15015 EUR	15 EUR	7726.62€	No label		
2022-06-01	Coinbase Pro	Sell	3 ETH	30120 EUR	120 EUR	22211.15€	No label		
Showing 1 to 5 of 5 entries								Previous 1 Next	

21 pav. Vartotojo sistemoje įkeltų transakcijų sąrašas

Trečiame etape kuriamas transakcijų peržiūros ir redagavimo funkcionalumas (žr. 21 pav.), leidžiantis vartotojui peržiūrėti, bei keisti įkeltų į sistemą kriptovaliutų transakcijų informaciją, transakcijos, kurios turi klaidų, pavyzdžiui trūksta informacijos apie parduotos kriptovaliutos pirkimo kainą, yra paryškinamos ir iškeliamos į sąrašo viršų. Transakcijų sąrašas yra atvaizduojamas

lentele, naudojant *DataTables* biblioteką, taip vartotojui suteikiama daug reikalingo funkcionalumo. Lentelės duomenis galima rūšiuoti pagal pasirinktą stulpelį, taip pat yra paieškos laukas, kuris palengvina vartotojui norimos transakcijos radimą, bei yra galimybė nustatyti norimą rodomų lentelės puslapyje transakcijų kiekį. Veiksmų stulpelyje vartotojas turi galimybę spustelėjęs pasirinktą piktogramą, redaguoti ar ištrinti transakciją.

6.4. Pelno ir mokesčių informacijos peržiūra

Užtikrinus transakcijų teisingumą, kuriamas funkcionalumas apskaičiuojantis vartotojo metines pajamas, išlaidas, sumokėtus transakcijų mokesčius, įgytą pelną ar nuostolius bei metinius mokesčius. Kadangi vartotojas gali keisti transakcijų duomenis, įkelti naujų transakcijų ar ištrinti esamą transakciją, pelno ir mokesčių apskaičiavimo funkciją reikia įvykdyti kas kart, kai vartotojo transakcijų informacija ar kiekis pasikeičia. Ši informacija atnaujinama *worker* sistemos dalyje ir išsaugoma duomenų bazėje, kol šis procesas vyksta vartotojo laukelis *is_loading* duomenų bazėje pažymimas *true* ir priekinė dalis vietoj mokesčių informacijos atvaizduoja krovimosi procesą, bei tikrina vartotojo krovimosi statusą kas 5 sekundes, vartotojo mokesčių informaciją atnaujinus, puslapis yra perkraunamas.

Sistemoje vartotojas gali pasirinkti norimus metus ir pamatyti šių metų pelno ar praradimo grafiką, mokesčių informaciją, bei pasirinktų metų sugeneruotas ataskaitas. Puslapio viršuje rodoma lentelė (žr. 22 pav.) praneša vartotojui ar reikia deklaruoti pajamas už pasirinktus metus, jei deklaruoti reikia, vartotojas taip pat paspaudęs ant pagalbos mygtuko gali sužinoti tikslūs žingsnius, kuriuos jam reikia atlikti norint sėkmingai deklaruoti kriptovaliutų pajamų mokesčius. Pelno ar praradimo grafikas (žr. 22 pav.) vaizduoja dviejų skirtingų tipų pajamų augimą, šiame grafike x ašis žymi mėnesio pavadinimą, y ašis atvaizduoja pasirinktų metų pelno ar praradimo euraiš augančią sumą, mėlyna linija žymi visų kriptovaliutų sandorių pajamų išskyrus pajamas gautas kaip palūkanas augimą, o raudona linija žymi tik kriptovaliutų palūkanų pajamų augimą. Bendroje metinėje mokesčių informacijoje (žr. 23 pav.) vaizduojamos pasirinktų metų kriptovaliutų sandorių pajamos, išlaidos, sumokėti sandorių mokesčiai, skirtingų tipų apmokestinamų pajamų kiekis ir mokėtini mokesčių dydžiai.



22 pav. Pelno/praradimo grafikas

Tax information		
Total spent	Total received	Total fees
€25,150.00	€45,000.00	€556.62
Taxable capital gains	Taxable interest earnings	
€19,850.00	€0.00	
Capital gains tax	Interest earnings tax	
€2,602.50	€0.00	

23 pav. Metinė mokesčių informacija

6.5. Transakcijų suvestinės generavimas

Sistemoje vartotojas pasirinkties metams gali sugeneruoti ir atsisiųsti metinę transakcijų suvestinę, peržiūrėti jau anksčiau sugeneruotų suvestinių sąrašą ar ištrinti pasirinktą suvestinę (žr. 24

pav.). Suvestinių sąrašas atvaizduojamas lentelė, nurodomi kiekvienos suvestinės mokestiniai metai, sukūrimo data, suvestinės tipas, atsisiuntimo nuoroda bei ištrynimo mygtukas. Prieš generuojant suvestinę, vartotojas gali iš išskleidžiamojo meniu pasirinkti norimos suvestinės tipą, kadangi reikalinga paprastų sandorių suvestinė ir atskirai deklaruojamų palūkanų sandorių suvestinė. Sugeneravęs ir atsisiuntęs reikiamas suvestines vartotojas gali jas naudoti kaip pajamų pripažinimo dokumentą deklaruojant kriptovaliutų pajamas.

Reports				
Report year	Date created	Report type	Link	Actions
2023	2023-11-24 13:44	Capital gains	Download	
2023	2023-11-24 13:44	Interest earnings	Download	
Capital gains				
GENERATE REPORT				

24 pav. Suvestinės generavimas

6.6. Naudojimosi sistema vadovas

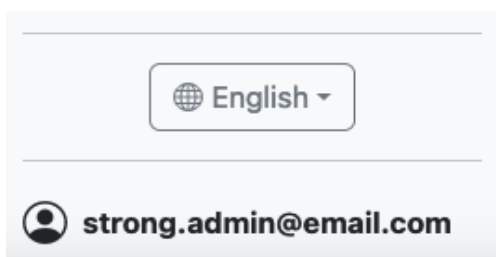
Norint vartotojui palengvinti naudojimosi sistema procesą, sukuriami 3 gidai (žr. 25 pav.): kaip deklaruoti kriptovaliutų mokesčius naudojantis *CoinTax* sistema, kaip deklaruoti kriptovaliutų pajamas Lietuvoje, bei kaip deklaruoti kriptovaliutų palūkanų pajamas Lietuvoje. Kiekvienas iš šių vadovų turi detaliai žingsnis po žingsnio aprašymus, kaip atlikti nurodomą veiksmą, tokiu būdu vartotojas paeiliui atlikdamas nurodytus žingsnius lengvai gali atlikti norimus procesus.

Guide	
How to declare cryptocurrency taxes in Lithuania easier using CoinTax?	
1. Upload transaction history from selected wallets or exchanges in Wallets section 2. Check for errors and make sure all transactions are imported correctly in Transactions section 3. Visit Dashboard section to see your tax information for selected year 4. Download necessary tax report in Dashboard section 5. Follow the guide below to declare your taxes in VMI Elektroninio deklaravimo sistema	
How to declare cryptocurrency gains in Lithuania?	▼
How to declare cryptocurrency interest gains in Lithuania?	▼

25 pav. Sistemos vadovas

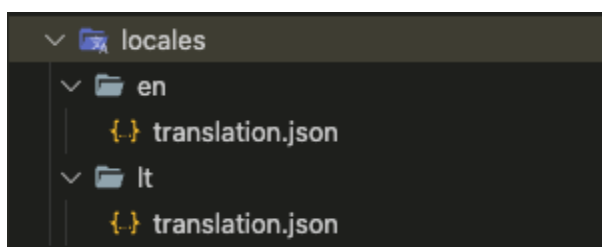
6.7. Sistemos pritaikymas tarptautinei rinkai

Kadangi kriptovaliutų ir su jomis susijusių operacijų sąvokų kalba yra anglų, o Lietuvos mokesčių sistemos kalba yra lietuvių, vartotojui gali kilti neaiškumų dėl tam tikrų sistemos sąvokų ar aprašymų, taip pat sistemos kūrimas yra lengvesnis naudojant anglų kalbą, tačiau tikslinės sistemos vartotojų rinkos kalba yra lietuvių. Dėl šių priežasčių, norint supaprastinti ir palengvinti naudojimosi sistema, bei sistemos kūrimo procesus, pridedamas sistemos lokalizacijos funkcionalumas, leidžiantis vartotojui pasirinkti tarp naudojimosi sistema lietuvių ar anglų kalba (žr. 26 pav.). Sistemos kūrimo pasirinkta kalba lieka anglų, tačiau visi atvaizduojami priekinėje dalyje tekstai taip pat verčiami ir į lietuvių kalbą, o vartotojui atvaizduojami pagal vartotojo pasirinktą kalbą.

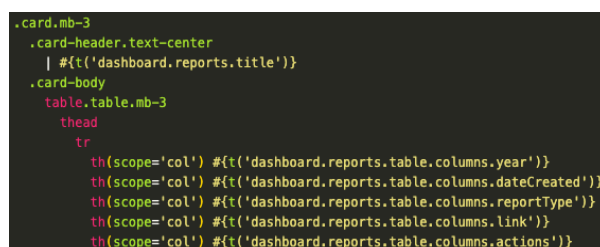


26 pav. Kalbos pasirinkimas

Šiam procesui įgyvendinti naudojama *i18next*, kartu su *i18next-fs-backend* ir *i18next-http-middleware* bibliotekomis. Sistemoje sukuriamas naujas aplankas su anglų ir lietuvių kalbų vertimais *JSON* failuose (žr. 27 pav.). Serveryje sukuriamą naują tarpinę programinę įrangą nustatanti vartotojo pasirinktą kalbą, kuri yra anglų, kol vartotojas jos nepakeičia. Pagal pasirinktą kalbą naudojami vertimai iš šios kalbos vertimų failo, o priekinėje dalyje, vietoj kode įrašomo teksto, naudojamas kreipimasis į tam tikro teksto konstantą, kuri gali būti pasirinktas tekstas anglų ar lietuvių kalba (žr. 28 pav.).



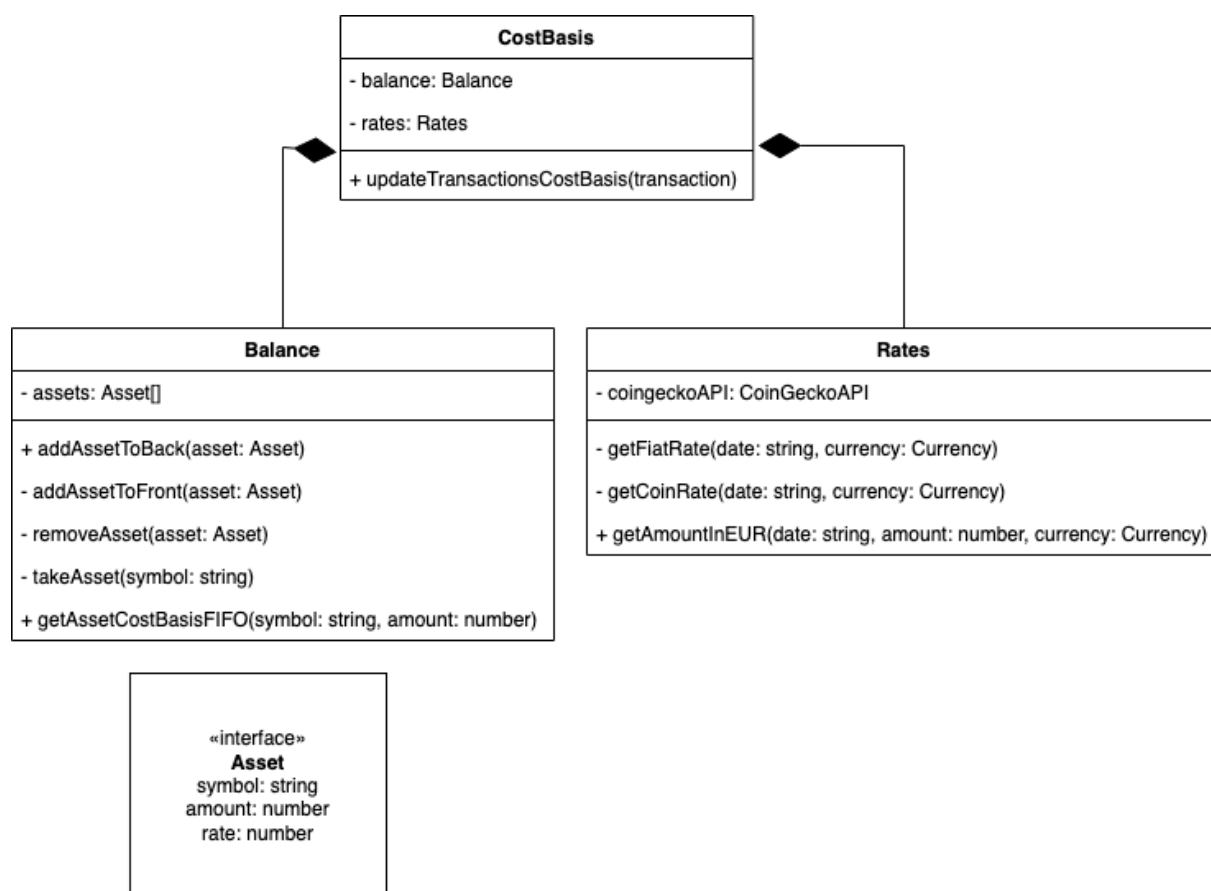
27 pav. Anglų ir lietuvių kalbų vertimų aplanko struktūra



28 pav. Priekinėje dalyje naudojamos vertimų konstantos

6.8. Įsigijimo išlaidų ir pelno skaičiavimo algoritmai

Parduodamos kriptovaliutos galėjo būti prieš tai laikomos ne vienerius metus, taigi pelno skaičiavimas atliekamas visoms vartotojo transakcijoms iš karto, kadangi parduodamos kriptovaliutos įsigijimo išlaidos nustatomos ne tik iš esamų metų atliktų transakcijų, bet ir senesnių įvykusių transakcijų. Kiekviena transakcija jau turi informaciją apie sumokėtų valiutų, gautų valiutų ir valiutos kiekį panaudotą transakcijos įvykdymo mokesčiui apmokėti rūšį, kiekį ir įsigijimo ar dieninės vidutinės vertės kainą. Taigi, gautas pelnas priskiriamas kiekvienai transakcijai esančiai duomenų bazėje, šis pelnas apskaičiuojamas iš gautų valiutų vertės atimant išleistų valiutų įsigijimo kainą, bei sumokėto transakcijos mokesčio vertę. Įsigijimo vertė apskaičiuojama naudojant surūšiuotų nuo seniausios iki naujausios transakcijų ciklą (*FIFO*) ir išsaugant pirktų valiutų simbolių, kiekį, bei perkamos valiutos kursą pagal pirkimo kainą *Balance* klasėje (žr. 29 pav.). Tuo tarpu, jei pardavimo transakcijai įsigijimo išlaidų rasti nepavyksta, tai yra *Balance* klasėje nėra tokio simbolio valiutos ar jos kiekis yra mažesnis, pelnas transakcijai nėra priskiriamas ir transakcijos klaidos laukelyje įrašoma, jog transakcijai trūksta informacijos apie įsigijimą ir vartotojas turėtų įkelti transakcijų failą iš papildomos kriptovaliutų keityklos, kur ši valiuta buvo įsigyta. Tokiu būdu lengva apskaičiuoti vartotojo mokesčių sumą skirtingais metais ar vartotojui pasirinkus skirtingą mokesčių skaičiavimo būdą, šiais atvejais pelno skaičiuoti iš naujo nereikia. Pridėti naują įsigijimo išlaidų skaičiavimo būdą yra visai nesudėtinga, *Balance* klasėje galima sukurti naują metodą *getAssetCostBasisLIFO*, kuris apdorotų pridėtą turtą iš kitos pusės, nuo galo, taip įsigijimo išlaidas jau būtų galima apskaičiuoti naudojant *last-in, first-out (LIFO)* metodą.



29 pav. Transakcijų pelno skaičiavimo klasių diagrama (dėl aiškumo naudojama programinio kodo terminologija)

6.9. Sistemos testavimas su realiais duomenimis

6.9.1. Rankiniai testai

Didžioji dalis sistemos testavimo yra atliekama rankiniu būdu. Šiam testavimui sukuriama sistemos vartotojas, bei 4 testavimui skirti transakcijų duomenų CSV failai (žr. 30 pav.). Prisijungus su testavimui skirtu vartotoju prie sistemos ir naudojant parengtus failus galima patikrinti pagrindinių sistemos funkcijų, bei sistemos priekinės dalies išvaizdos veikimą. Taip tikrinamos su vartotoju susijusios funkcijos: vartotojo prisijungimas ir registracija, atsijungimas, dviejų faktoriaus autentifikacija, vartotojo duomenų šalinimas. Taip pat prisijungus su vartotoju tikrinamas ir transakcijų įkėlimo, pelno ir mokesčių apskaičiavimo, bei ataskaitų generavimo funkcionalumas, naudojant parengtus testavimui CSV formato failus, tikrinama ar gaunami sistemoje duomenys yra teisingi. Galiausiai tikrinamas sistemos priekinės dalies funkcionalumas ir išvaizda, tam naudojamas *Firefox* naršyklės *Responsive Design Mode*, kuris leidžia lengvai keisti svetainės lango dydį, šiuo testavimo būdu užtikrinama, jog svetainė atrodo gerai ir viskas yra matoma skirtingų dydžių ekranų įrenginiuose.

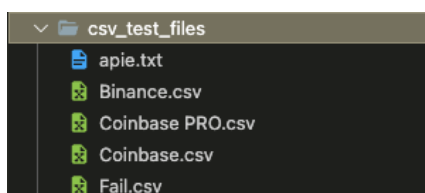
Sistemos testavimui buvo pakeičiami pasirinktų kriptovaliutų keityklų išrašų duomenys, taip jog atitiktų reikiamas testavimo situacijas, tačiau CSV failo formatas specifiniai kriptovaliutų keityklai būtų teisingas. Kiekvienai palaikomai kriptovaliutų keityklai buvo sukurta po 1 testinį failą su specifine situacija:

Coinbase testavimo situacijai buvo sukuriama CSV failas, kuriame vartotojas 2023 metais nusipirko 12 *BTC* už 241200 eurų ir vėliau kas mėnesį pardavinėjo po 1 *BTC*. Tokiu būdu patikrinama, jog sistemoje teisingai apskaičiuojamos pajamos gautos iš kriptovaliutų pirkimų ir pardavimų.

Coinbase PRO testavimo situacijai buvo sukuriama CSV failas, kuriame vartotojas 2022 metais nusipirko 1 *BTC* už 10060 eurų ir vėliau dar 1 *BTC* už 15090 eurų, o vėliau iškeitė 1.518 *BTC* į 3 *ETH*, galiausiai pardavė likusius 0.482 *BTC* už 15000 eurų ir 3 *ETH* už 30000 eurų. Tokiu būdu patikrinama, jog sistemoje teisingai apskaičiuojamos pajamos, kai kriptovaliutos yra ne tik perkamos ir paruošamos, bet ir keičiamos į kitas kriptovaliutas.

Binance testavimo situacijai buvo sukuriama CSV failas, kuriame vartotojas 2021 metais nusipirko 210000 *USDT* už 200000 eurų ir laikė šias valiutas saugykloje 12 mėnesių, kas mėnesį gaudamas 2100 *USDT* palūkanų ir metų pabaigoje pardavęs visus turimus 235200 *USDT* už 223216.5 eurų. Tokiu būdu patikrinama, jog sistemoje teisingai apskaičiuojamos pajamos gautos iš palūkanų, bei atskiriamos nuo įprastų pajamų.

Taip pat buvo sukurtas failas neatitinkantis nė vienos palaikomos keityklos išrašo formato, norint įsitikinti, jog sistema toliau teisingai veiks, bei gražins klaidą, kai bus bandoma įkelti netinkamo formato CSV failą. Ištestuojant sistemą su sukurtais testiniais CSV failais yra užtikrinama, kad sistema teisingai apdoroja kiekvienos palaikomos kriptovaliutų keityklos failus, bei teisingai apskaičiuoja išleistų ir gautų pajamų kiekius, jų išlaidų pagrindus, metinį pelną, bei mokesčius. Šiuos failus, jų aprašymus su situacijomis ir tikėtinais rezultatais galima rasti *csv_test_files* aplanke.



30 pav. Failai naudojami rankiniam testavimui

6.9.2. Automatiniai testai

Norint užtikrinti teisingą sistemos verslo logikos veikimą ir sumažinti klaidų atsiradimo riziką, sistemoje kuriami automatiniai vienetiniai testai (angl. *Unit tests*) vartotojo mokesčių skaičiavimui (žr. 6 lentelę) ir transakcijų išlaidų pagrindo, bei pelno (žr. 7 lentelę), naudojant *Jest* biblioteką. Būtent šis testų tipas buvo pasirinktas vietoj didesnio masto automatinių testų, nes norima patikrinti atskirų skaičiavimo funkcijų teisingą veikimą, tuo tarpu didesnio masto *E2E* testavimas atliekamas rankiniu būdu. Vienetiniai testai kuriami penkioms pagrindinėms skaičiavimų klasėms, iš kurių 3 yra pelno skaičiavimo klasės ir 2 mokesčių skaičiavimo klasės. Šių testų metu yra sukurama išvestis, taigi testui nepavykus, pirmiausia žiūrima į testo išvestį, o vėliau tikrinamos tam testui specifinės kodo vietos, kurios aprašytos mokesčių skaičiavimo (žr. 6 lentelę) ir pelno skaičiavimo (žr. 7 lentelę) klasių testavimo situacijų lentelėse.

6 lentelė. Mokesčių skaičiavimo klasių automatinio testavimo scenarijai

Mokesčių skaičiavimas		
Testuojama klasė	Tikėtinas rezultatas	Kas tikrinama nepavykus?
TaxStrategyFactory	Grąžinama teisinga mokesčių skaičiavimo klasė pagal perduotus metus	Tikrinamas klasės kodas.
	Gaunamas klaidos pranešimas, jei perduoti metai nėra palaikomi	
TaxCalculator	Teisingai apskaičiuojami mokesčiai <i>Coinbase</i> testavimo situacijai	Tikrinamas klasės kodas, atitinkama mokesčių skaičiavimo funkcija.
	Teisingai apskaičiuojami mokesčiai <i>Coinbase Pro</i> testavimo situacijai	
	Teisingai apskaičiuojami mokesčiai <i>Binance</i> testavimo situacijai	

7 lentelė. Pelno skaičiavimo klasių automatinio testavimo scenarijai

Pelno skaičiavimas		
Testuojama klasė	Tikėtinas rezultatas	Kas tikrinama nepavykus?
Rates	Gaunamas teisingas valiutos kursas perduodant valiutos objektą ir datą	Ar norimi valiutų kursai yra duomenų bazėje?
	Gaunamas klaidos pranešimas perduodant neteisingą valiutos objektą ar datą	Tikrinamas klasės kodas.
	Gaunamas teisingas kriptovaliutos kursas perduodant valiutos objektą ir datą	Ar <i>CoinGecko API</i> yra pasiekiamas?
	Gaunamas klaidos pranešimas perduodant neteisingą valiutos objektą ar datą	Tikrinamas klasės kodas.
Balance	Teisingai apskaičiuojamos įsigijimo išlaidos perduodant vieną turto vienetą	Tikrinamas klasės kodas.
	Teisingai apskaičiuojamos įsigijimo išlaidos perduodant kelis turto vienetus	
	Grąžinamas <i>null</i> , jei perduoto turto įsigijimo išlaidų skaičiavimui yra daugiau, nei prieš tai pridėto turto	
CostBasis	Teisingai apskaičiuojamos įsigijimo išlaidos ir pelnas <i>Coinbase</i> testavimo situacijos transakcijoms	Tikrinamas klasės kodas, svarbu atkreipti dėmesį ar prieš tai vykdyti testai pavyko.
	Teisingai apskaičiuojamos įsigijimo išlaidos ir pelnas <i>Coinbase Pro</i> testavimo situacijos transakcijoms	
	Teisingai apskaičiuojamos įsigijimo išlaidos ir pelnas <i>Binance</i> testavimo situacijos transakcijoms	
	Teisingai pažymi transakcijas, kurioms trūksta įsigijimo išlaidų šaltinio	

Darbo rašymo metu visi testai pavyko sėkmingai (žr. 31 pav.). Kadangi testai naudoja *CoinGecko API*, su mažu galimų užklausų kiekiu ir dideliu jų nepavykimu dėl klaidos 429 per daug užklausų, galimi testų nepavykimai dėl laiko baigimosi, šiuo atveju nereikia nieko keisti, tik padidinti testo laiko išsekimo kintamąjį ir bandyti iš naujo.

```
Test Suites: 5 passed, 5 total
Tests:      16 passed, 16 total
Snapshots:  0 total
Time:       162.543 s, estimated 168 s
```

31 pav. Testavimo rezultatai

Išvados ir rekomendacijos

Apibendrinant, baigiamojo bakalauro darbo metu suformuoti uždaviniai ir iškelti tikslai įgyvendinti sėkmingai. Rezultatams pasiekti buvo analizuojami straipsniai ir kita viešai prieinama informacija apie kriptovaliutų mokesčių deklaravimo reikalavimus Lietuvoje. Surinkus ir išsikėlus aiškius reikalavimus buvo sukurta internetinės informacinės sistemos, kuri palengvintų Lietuvos gyventojų kriptovaliutų mokesčių deklaravimo procesą, architektūra, bei įgyvendinti programavimo darbai.

Bendrosios baigiamojo bakalauro darbo išvados ir rezultatai:

- Įgyvendinta internetinė svetainė turi pranašumą panašių sistemų rinkoje, nes yra sukurta atsižvelgiant į Lietuvos kriptovaliutų mokesčių sistemą, tokiu būdu sistema leidžia vartotojams ne tik atsisiųsti metinį transakcijų išrašą kriptovaliutų mokesčių deklaracijos pildymui, bet ir apskaičiuoja mokėtinų mokesčių dydžius ir yra tinkamesnė Lietuvos gyventojams, nei kitos rinkoje esančios sistemos.
- Renkantis technologijas (žr. 3 skyrių) buvo pasirinktos technologijos leidžiančios lengvai ir greitai plėsti sistemos funkcionalumą, naudojamos populiaros, prižiūrimos ir saugios trečiųjų šalių bibliotekos, kurių palaikymas iš jų autorių tikėtinais tėsiai ir ateityje.
- Pasirinkta sistemos architektūra (žr. 4 skyrių) leidžia lengvai plėsti funkcionalumą ateityje, didinti palaikomų kriptovaliutų keityklų sąrašą, pridėti naujų mokesčių skaičiavimo algoritmų bei keisti mokesčių skaičiavimo strategijas, tokiu būdu lengvai prisitaikyti prie naujų mokesčių pakeitimų. Taip pat pasirinkta sistemos architektūra (žr. 4.3 skyrių) užtikrina gerą sistemos svetainės prieinamumą ir lengvą didėjančios apkrovos palaikymą.
- Centralizuotas sistemos išrašų registravimas (žr. 4.10 skyrių) užtikrins greitą klaidų aptikimą, bei lengvą sistemos priežiūrą ateityje.
- Išnagrinėti skirtingi metodai reikiamų duomenų gavimui (žr. 5 skyrių) ir pasirinkti tinkamiausi sistemai metodai, kurių našumas gali būti lengvai didinamas, kai sistemos vartotojų skaičius augs.
- Parengti rankiniai ir automatiniai testai (žr. 6.9 skyrių) užtikrina teisingą sistemos veikimą ne tik šiuo metu, bet ir plečiant sistemos funkcionalumą ateityje.

Ateities tyrimų planas

Priklausomai nuo sistemos vartotojų skaičiaus ir poreikių, sistema gali būti tobulinama, jog palaikytų daugiau vartotojų, automatizuotų daugiau kriptovaliutų mokesčių operacijų ar dar labiau supaprastintų vartotojų atliekamus sistemoje veiksmus. Pirmiausia norint sistemai palaikyti daugiau vartotojų, būtų galima kurti sistemos paleidimo architektūrinius modelius, naudojant daugiau virtualių mašinų, taip didinant sistemos pajėgumus palaikyti didesnę vartotojų kiekį vienu metu. Kitas sistemos tobulinimas pridėtų palaikymą sudėtingesniais mokesčių skaičiavimo procesais, vartotojams, kurie užsiima individualia veikla, jei tokių vartotojų kiekis būtų reikšmingas. Galiausiai būtų galima pridėti kriptovaliutų keityklų transakcijų įkėlimą naudojant šių keityklų programinę programavimo sąsają, taip palengvinant vartotojams transakcijų įkėlimo, bei atnaujinimo procesą.

Taip pat reiktų atsižvelgti į galimą mokesčių pokyčių riziką, jau šiuo metu yra siūlymas keisti mokesčių dydžius, didinant, aukštas pajamas gaunančių, asmenų kitų pajamų ir individualios veiklos pajamų tarifus [22]. Šie pokyčiai galėtų būti nesunkiai integruojami į sistemą pridedant naują mokesčių skaičiavimo strategiją.

Literatūros šaltiniai

- [1] BTC-EUR historical prices, 2022. <https://finance.yahoo.com/quote/BTC-EUR/history>.
- [2] Confirmed transactions per day, 2022. <https://www.blockchain.com/explorer/charts/n-transactions>.
- [3] Express tutorial part 4: Routes and controllers, 2022. https://developer.mozilla.org/en-US/docs/Learn/Server-side/Express_Nodejs/routes.
- [4] European Central Bank. Euro foreign exchange reference rates, 2023. https://www.ecb.europa.eu/stats/policy_and_exchange_rates/euro_reference_exchange_rates/html/index.en.html.
- [5] Binance. Welcome to the binance card, 2022. <https://www.binance.com/en/cards>.
- [6] Wayne Duggan. The history of bitcoin, the first cryptocurrency, 2022. <https://money.usnews.com/investing/articles/the-history-of-bitcoin>.
- [7] Finmeka. Uždirbusiems iš kriptovaliutų gresia mokesčiai: kiek ir kada reikia susimokėti, 2022. <https://www.finmeka.lt/uzdirbusiems-is-kriptovaliutu-gresia-mokesciai-kiek-ir-kada-reikia-susimoketi/>.
- [8] David Floyd. How bitcoin works, 2022. <https://www.investopedia.com/news/how-bitcoin-works/>.
- [9] Stellar Development Foundation. We're distributing 16 billion lumens to bitcoin holders, 2017. <https://stellar.org/blog/bitcoin-claim-lumens-2>.
- [10] Kade Garrett. What determines the price of bitcoin?, 2023. <https://decrypt.co/resources/what-determines-the-price-of-bitcoin-2>.
- [11] Adam Hayes. What happens to bitcoin after all 21 million are mined?, 2023. <https://www.investopedia.com/tech/what-happens-bitcoin-after-21-million-mined/>.
- [12] Mokesčių informacijos departamentas. Gyventojų gautų virtualios valiutos pardavimo pajamų apmokestinimas gyventojų pajamų mokesčiu, 2022. <https://www.vmi.lt/evmi/documents/20142/391071/Virtualios+valiutos+GPM.PDF>.
- [13] Ekonomikos ir verslo administravimo fakultetas. Naujausias vilniaus universiteto mokslininkų tyrimas atskleidė verslo požiūrį į kriptovaliutų panaudojimo tendencijas turizmo rinkoje, 2023. <https://naujienos.vu.lt/naujausias-vilniaus-universiteto-mokslininku-tyrimas-atскеide-verslo-poziuri-i-kriptovaliutu-panaudojimo-tendencijas-turizmo-rinkoje/>.
- [14] Huw Jones. EU calls for fast-track crypto capital rules for banks, 2023. <https://www.reuters.com/markets/europe/eu-calls-fast-track-crypto-capital-rules-banks-2023-02-20/>.
- [15] Jordan Major. SEC filings mentioning 'Bitcoin' skyrocket in 2022 signaling institutional acceptance, 2022. <https://finbold.com/sec-filings-mentioning-bitcoin-skyrocket-in-2022-signaling-institutional-acceptance/>.

- [16] Satoshi Nakamoto. Bitcoin: A peer-to-peer electronic cash system, 2008. <https://bitcoin.org/bitcoin.pdf>.
- [17] Stack Overflow. 2022 developer survey, 2022. <https://survey.stackoverflow.co/2022/web-frameworks-and-technologies>.
- [18] Julie Pinkerton. The history of bitcoin, the first cryptocurrency, 2023. <https://money.usnews.com/investing/articles/the-history-of-bitcoin>.
- [19] Felix Roth. The effects of the financial crisis on systemic trust, 2009. <http://aei.pitt.edu/11334/1/1877-1.pdf>.
- [20] Edgaras Savickas. Uždirbusiems iš kriptovaliutų gresia mokesčiai: kiek ir kada reikia susimokėti, 2021. <https://www.delfi.lt/verslas/verslas/uzdirbusiems-is-kriptovaliutu-gresia-mokesciai-kiek-ir-kada-reikia-susimoketi.d?id=86220889>.
- [21] Andreas Thiemann. Cryptocurrencies: An empirical view from a tax perspective, 2021. <https://joint-research-centre.ec.europa.eu/system/files/2021-08/jrc126109.pdf>.
- [22] Vytautas Budzinauskas Viktorija Chockevičiūtė, Raminta Rakauskė. Pristatyta mokesčių reforma: individualiai dirbančių laukia mokesčių didinimas, atsisakoma populiarios lengvatos, 2023. <https://www.delfi.lt/verslas/verslas/pristatyta-mokesciu-reforma-individualiai-dirbanciu-laukia-mokesciu-didinimas-atsisakoma-populiarios-lengvatos-92855809>.
- [23] VMI. Bendrosios nuostatos, 2022. <https://www.vmi.lt/evmi/documents/20142/391185/Paa%C5%A1kinimas+d%C4%97l+virtuali%C5%B3++RM-21969.pdf/30a8e9a1-132d-5f91-314c-64ff290f8d8b?t=1545390441331>.
- [24] VMI. Dėl 2020 metais gyventojų gautų palūkanų apmokestinimo ir deklaravimo, 2022. <https://www.vmi.lt/evmi/-/d-c4-97l-2020-metais-gyventoj-c5-b3-gaut-c5-b3-pal-c5-abkan-c5-b3-apmokestinimo-ir-deklaravimo>.
- [25] VMI. Gyventojų pajamų mokestis, 2022. <https://www.vmi.lt/evmi/gyventoju-pajamu-mokestis2>.