Универзитет у Београду
Факултет организационих наука

Завршни рад

Петар Луковац

Београд, 2022. година

Универзитет у Београду

Факултет организационих наука

Електронско пословање и управљање системима

Технологије електронског пословања

Завршни рад

Тема: Паметни систем за праћење порекла органског меда заснован на *blockchain* технологијама

Петар Луковац 2021/3502

Београд, 2022. Година

Сагласност чланова комисије за одбрану

Попуњавају чланови Комисије за одбрану:

Комисија која је прегледала рад
кандидата ЛУКОВАЦ (ПРЕДРАГ) ПЕТАР
под насловом ПАМЕТНИ СИСТЕМ ЗА ПРАЋЕЊЕ ПОРЕКЛА ОРГАНСКОГ МЕДА
ЗАСНОВАН НА BLOCKCHAIN ТЕХНОЛОГИЈАМА и одобрила одбрану:

Ментор: др Александра Лабус, ванредни професор

_____

Члан: др Маријана Деспотовић-Зракић, редовни професор

_____

Члан: др Дејан Симић, редовни професор

_____

Радна биографија

Петар Луковац, рођен је 21.03.1999. у Београду. Завршио је основну школу „Иван Гундулић“ као и основну музичку школу „Станислав Бинички“ на Новом Београду. Учествовао је на бројним такмичењима из математике, географије, хемије итд. Основну школу завршио је са Вуковом дипломом. Након тока, уписује Девету гимназију „Михаило Петровић Алас” на Новом Београду, друштвени смер. Године 2017. завршава средњу школу са одличним успехом. Због интересовања у области информатике опредељује се за Факултет организационих наука у Београду и уписује смер информациони системи и технологије. На почетку друге године студија постаје члан међународне организације младих AIESEC, где је водио тим за развој пословања. Учествовао је у реализацији пројеката, како у студентској организацији, тако и ван ње. Током четвртое године студија постаје сарадник у настави у лабораторији за Електронско пословање. Основне студије, завршио је са крајним просеком 9,23. Након тога уписује мастер академске студије на смеру Електронско пословање и управљање системима и одлучује се за модул Технологије електронског пословања. Убрзо затим постаје сарадник у настави на катедри за Електронско пословање, где учествује у реализацији практичне наставе и бројним пројектима катедре. Мастер академске студије завршава са просеком 10,00.

Изјава о академској честитости

Попуњава кандидат:

Луковац, Предраг, Петар	2021/3502
Презиме, име једног родитеља, име	Број индекса

Студијски програм: Електронско пословање и управљање системима

Модул: Технологије електронског пословања

Аутор завршног рада под насловом: Паметни систем за праћење порекла органског меда заснован на blockchain технологијама

Чија је израда одобрена на Седници Већа студијских програма мастер академских студија одржаној: 09.05.2022.

Потписивањем изјављујем:

- Да је рад искључиво резултат мог сопственог истраживачког рада;
- Да сам рад и мишљења других аутора које сам користио у овом раду назначио или цитирао у складу са Упутством;
- Да су сви радови и мишљења других аутора наведени у списку литературе/референци који су саставни део овог рада и писани су у складу са Упутством;
- Да сам добио све дозволе за коришћење ауторског дела који се у потпуносзи/целости уносе у предати рад и да сам то јасно навео;
- Да сам свестан да је плагијат коришћење туђих радова у било ком облику (као цитата, парафраза, слика, табела, дијаграма, дизајна, планова, фотографија, филма, музике, формула, веб сајтова, компјутерских програма и сл.) без навођења аутора или представљање туђих ауторских дела као мојих, кажњиво по закону (Закон о ауторским и сродним правима, Службени гласник Републике Србије, бр. 104/2009, 99/2011, 119/2012), као и других закона и одговарајућих аката Универзитета у Београду и Факултета организационих наука;
- Да сам свестан да плагијат укључује и представљање, употребу и дустрибуирање рада предавача или других студената као сопствених;
- Да сам свестан последица које код доказаног плагијата могу проузроковати на предати завшни мастер рад и мој статус;
- Да је електронска верзија завршног рада идентична штампаном примерку и пристајем на његово објављивање под условима прописаним актима Универзитета и Факултета.

Београд, _____

Потпис студента _____

Апстракт

Предмет истраживања овог мастер рада представља паметни систем за праћење порекла органског меда заснован на *blockchain* технологијама.

У теоријском делу рада биће анализирани релевантни концепти интернета интелигентних уређаја и *blockchain* технологија. Биће детаљно сагледане могућности примене IoT и blockchain технологија за праћење порекла хране. Анализом литературе биће истражени актуелни пословни модели и софтверска решења која се примењују у праћењу порекла хране, са фокусом на примену у праћењу порекла меда.

У практичном делу рада ће предложен пословни модел и пројектован софтверски систем за праћење порекла меда. Систем се састоји од паметних кошница опремљених одговарајућим IoT уређајима за прикупљање података из окружења и условима у којима мед настаје. Прикупљени подаци ће се складиштити у blockchain мрежи како би се обезбедила њихова веродостојност и доступност. Крајњи купци ће имати могућност приступа подацима у blockchain мрежи, како би проверили порекло меда. Имплементација прототипа предложеног решења биће реализована у лабораторијском окружењу, коришћењем одабраног скупа сензора и IoT уређаја. Софтверске компоненете биће имплементиране коришћењем JavaScript технологија. Back-end део система ће бити имплементиран у Node.js-у, а front-end коришћењем React библиотеке. Blockchain део система ће бити имплементиран коришћењем Hyperledger платформе.

Циљ рада је развој паметног система за праћење порекла органског меда, у коме су подаци доступни крајњим купцима. У развијеном систему се користи blockchain технологија како би се обезбедила истинитост и непромењивост информација.

Кључне речи: Електронско пословање, *Blockchain*, Производња меда, Паметна агрикултура

Abstract

The subject of research of this master thesis is a smart system for tracking the origin of organic honey based on blockchain technologies.

In the theoretical part of the paper, the relevant concepts of the Internet of Things devices and blockchain technologies will be analyzed. The possibilities of applying IoT and blockchain technologies for tracking the origin of food will be considered in detail. The analysis of the literature will explore the current business models and software solutions used in tracking the origin of food, with a focus on the application in tracking the origin of honey.

In the practical part of the paper, a business model and a designed software system for tracking the origin of honey will be proposed. The system consists of smart hives equipped with appropriate IoT devices for collecting data from the environment and the conditions in which honey is produced. The collected data will be stored in a blockchain network to ensure their credibility and availability. End customers will have the opportunity to access data in the blockchain network, in order to check the origin of honey. The implementation of the prototype of the proposed solution will be realized in a laboratory environment, using a selected set of sensors and IoT devices. Software components will be implemented using JavaScript technologies. The back-end part of the system will be implemented in Node.js, and the front-end using the React library. The blockchain part of the system will be implemented using the Hyperledger platform.

The aim of this paper is to develop a smart system for tracking the origin of organic honey, in which data are available to end customers. The developed system uses blockchain technology to ensure the truthfulness and immutability of the information.

Key words: E-Business, Blockchain, Honey Production, Smart Agriculture

Садржај слика

Слика **1** - Ток рада blockchain-а, прилагођен од (Bogdanović et al., 2021) 3

Слика **2** - Стејкхолдери предложеног система за праћење порекла органског меда (Bogdanović et al., 2021) 15

Слика **3** - Паметни уговори у систему за праћење органског меда (Bogdanović et al., 2021) 16

Слика **4** - Модел процеса за процес праћења порекла хране 17

Слика **5** - Модел података система за праћење порекла хране 18

Слика **6** - Модел паметног уговора 19

Слика **7** - Софтверска архитектура (Lukovac et al., 2022; Pranto et al., 2021) 21

Слика **8**: Предложена инфраструктура паметне кошнице 22

Слика **9**: Прототип сензора на паметној кошници 23

Слика **10**: Прављење нове кошнице 23

Слика **11**: Поглед свих кошница 24

Слика **12**: Преглед температуре 24

Слика **13**: Мерење влажности ваздуха за одређену кошницу 25

Слика **14**: Избор кошнице за проверу квалитета ваздуха 25

Слика **15**: Приказ квалитета ваздуха за одређену кошницу 26

Слика **16**: Упозорење при промени места кошнице 26

Слика **17** - Приказ организације фајлова на backend делу аплицаије 27

Слика **18** - Повезивање са базом и креирање апликације 27

Слика 19 - Шема објекта корисника 28

Слика 20 - Контролер и функција регистрације (Traversy, 2022) 29

Слика 21 - Руте за операције пријаве и регистрације 30

Слика **22** - Приказ организације фајлова при изради паметног уговора - први део 31

Слика **23** - Приказ организације фајлова при изради паметног уговора - други део..... 31

Слика 24 - Варијабле и операције паметног уговора 32

Слика **25** - Функција за проверу фазе животног циклуса 32

Слика **26** - Позив одговарајуће операције у зависности од фазе животног циклуса 33

Слика **27** - Трансакција креирања 34

Слика **28** - Трансакција којом лабораторија прихвата уговор 35

Слика **29** - Трансакција којом се шаљу резултати и плаћа накнада 36

Слика **30** - Унутрашња трансакција за извршавање плаћања 37

Слика **31** - Провера фазе животног циклуса користећи TEAL 37

Слика **32** - Приказ уписа вредности у локалну променљиву користећи TEAL..... 38

Слика **33** - Скрипта која симулира прихватање уговора 38

Слика **34** - Скрипта која симулира креирање уговора 39

Слика **35** - Модел за испитивање спремности за коришћење система за праћење порекла меда 44

Слика **36** - Структурни модел мерења латентних варијабли 49

Слика **37** - Резултати примене алгоритма PLS 50

Слика **38** - Структурни модел мерења латентних варијабли са ревидираним индикаторима..... 52

Слика **39** - Резултати примене PLS алгоритма на ревидираном моделу..... 53

Слика **40** - Резултати bootstrapping-а 58

Садржај табела

Табела 1 - Пословни модел 15

Табела 2 - Резултати познавања IoT и blockchain технологија (Bogdanović et al., 2022) 41

Табела 3 - Резултати о перципираној користи (Bogdanović et al., 2022) 41

Табела 4 - Резултати о спремности за инвестиције и уочене жртве (Bogdanović et al., 2022) 42

Табела 5 - Индикатори којим се испитује намера потрошача за коришћење предложеног система 45

Табела 6 - Демографски подаци испитаника 47

Табела 7 - Вредности outer loadings индикатора у односу на варијабле 51

Табела 8 - Оцена валидности мерења латентних варијабли 54

Табела 10 - Оцена вредности дискриминанте - Fornell-Lackner критеријум 56

Табела 11 - VIF вредности 56

Табела 12 - Приказ коефицијената путање 57

Табела 13 - Тестирање хипотеза 59

Садржај

1. Увод 1

2. Преглед литературе 3

2.1. *Blockchain* технологије 3

2.1.1. Консензус алгоритми 4

2.1.2. Паметни уговори 5

2.1.3. Ethereum Blockchain 6

2.1.4. Hyperledger Fabric 8

2.1.5. NEM 10

2.1.6. Algorand 11

2.2. Интернет интелигентних уређаја – IoT 12

2.2.1. IoT у праћењу порекла производа 12

3. Пројектовање паметног система за праћење порекла органског меда 14

3.1. Пројектни задатак 14

3.2. Пословни модел 14

3.3. Моделовање процеса 16

3.4. Моделовање података 17

3.5. Моделовање паметних уговора 18

3.5.1. Паметни уговори у *Algorand*-у 19

3.6. Софтверска архитектура система 20

4. Развој прототипа 22

4.1. Предложена инфраструктура 22

4.2. Развој прототипа уређаја 22

4.3. Предложени изглед апликације 23

4.4. Апликациона логика 26

4.5. Имплементација паметног уговора 30

5. Анализа спремности стејкхолдера за усвајање предложеног модела 40

5.1. Анализа спремности произвођача меда за прихватање предложеног модела ... 40

5.1.1. Истраживачка питања и дизајн 40

5.1.2. Познавање *IoT* и *blockchain* технологија 41

5.1.3. Перципиране користи 41

5.1.4. Спремност да се прихвате потребна улагања и уочене жртве 42

5.2. Анализа спремности потрошача меда за прихватање предложеног модела 43

5.2.1. Инструменти истраживања 45

5.2.2. Узорак 47

5.2.3. Резултати анализе спремности за коришћење предложеног система за праћење порекла органског меда 48

Закључак 60

Референце 61

1. Увод

У овом мастер раду ће бити представљен паметни систем за праћење порекла органског меда заснован на *blockchain* технологијама. Разлог за истраживање ове теме се везује за потрошаче који све више виде важност здраве исхране и воде рачуна о пореклу хране коју конзумирају. У складу са тим, повећава се потреба за коришћењем органске хране у свакодневној исхрани и увиђа се њен утицај на здравље. Проблеми на које потенцијални купци наилазе су разликовање органске хране од неорганске јер визуелно нема начина да буду сигурни у порекло и квалитет. Услед тога потрошачи су осуђени да верују продавцима на реч. Како би се наведени проблеми превазишли, решење се може тражити у праћењу порекла хране. На овај начин, купци могу добити све потребне информације о производима које купују и доносити одлуке засноване на чињеницама.

Главни циљ овог рада је обезбеђивање поменутих информација пре свега купцима, а затим и осталим стејкхолдерима. Наведени циљ ће бити остварен развојем паметног система за праћење порекла органског меда. Систем ће импленетирати *blockchain* технологију како би се обезбедила сигурност и непромењивост свих података у мрежи. Појединачни циљеви рада су: омогућено праћење меда кроз систем, развој паметног система за прикупљање информација о меду у лабораторијским условима и примена паметних уговора у систему за праћење меда

Очекивани резултати истраживања су приступ информацијама о пореклу производа и његовом кретању, као и аутоматске трансакције применом паметних уговора.

У оквиру истраживања потребно је одговорити на питање „Да ли се применом *blockchain* технологија у праћењу порекла меда може омогућити потребна транспарентност, сигурност и непромењивост информација?“ Поред тога потребно је истражити да ли се применом паметних уређаја могу обезбедити све потребне информације које гарантују сигурност и исправност производа

Први део рада даје теоријску основу технологија које се користе. Обухвата *blockchain* технологије са нагласком на начин функционисања различитих паметних уговора. Такође, дат је опис IoT уређаја са фокусом на примену истих на праћење порекла производа. Наредно поглавље обухвата пројектовање паметног система за праћење порекла органског меда заснованог на *blockchain* технологијама. Приказан је пословни

модел, моделовање података и моделовање процеса, као и моделовање паметног уговора и софтверска архитектура система.

Поглавље предмет литературе објашњава основне концепте *blockchain* технологије са детаљнијим описом консензус алгоритама и паметних уговора. Након тога наставља се са описом појединих *blockchain* платформи у којима се могу реализовати *blockchain* апликације и паметни уговори. На самом крају објашњава се концепт интернета интелигентних уређаја и детаљније описује примена IoT у праћењу порекла производа.

Пројектовање паметног система за праћење порекла органског меда заснованог на *blockchain* технологијама је објашњено у наредном поглављу. Дат је опис пројектног задатка, пословног модела, моделовања података, моделовања процеса, а затим и опис моделовања паметних уговора и софтверска архитектура система.

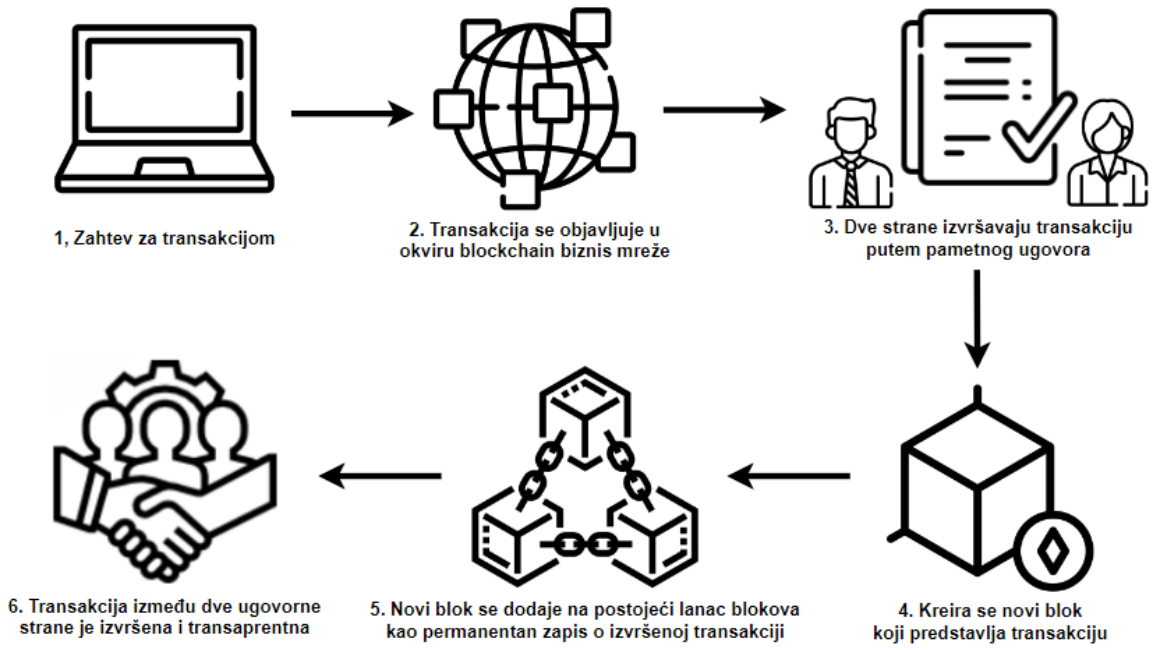
Развој прототипа система за мерење различитих параметара у окружењу кошнице приказан је у четвртом поглављу, уз навођење потребних сензора који се неизоставно налазе на уређају. Други део поглавља приказује предлог изгледа клијентске стране апликације. У наставку се налази и апликациона логика где је на примеру приказано на који начин је апликација повезана са базом, како се креира шема објекта и одговарајуће функције. На самом крају поглавља приказана је имплементација паметног уговора на *Algorand blockchain* платформи, уз коришћење *PyTeal* библиотеке.

Поглавље анализа резултата приказује анализу спремности произвођача меда за прихватање предложеног модела и анализу спремности потрошача меда за прихватање предложеног модела. Анализа спремности произвођача меда заснована је на VAM моделу и обухвата анализу познавања IoT и *blockchain* технологија, затим анализу перципиране користи и на крају анализу спремности за инвестиције и уочене жртве. Анализа спремности потрошача меда се односи на анализу мотива потрошача за коришћење предложеног система. Као зависна варијабла препознат је Став о коришћењу, а модел је креиран по узору на UTAUT2 и TAM модел. Детаљније испитивање узрочно-последичних веза између променљивих врши се применом PLS SEM методе.

2. Преглед литературе

2.1. *Blockchain* технологије

Blockchain је технологија која обезбеђује чување шифрованих података у дистрибуираној бази података и несметану верификацију извршених трансакција (Bogdanović et al., 2019; Diaz, M., Martin, C., Rubio, 2016). Када неко пошаље захтев за трансакцијом, администратор *blockchain* мреже прикупља захтеве и објављује их у блокчејн пословној мрежи. Радници у оквиру blockchain-а усклађују интересе купца и продавца путем паметног уговора који имплементира правила трансакција која су све стране прихватиле (Bogdanović et al., 2019). Ако паметни уговор испуњава услове купца и продавца, онда се трансакција валидира и пакује у нови блок који се везује за постојећи ланац блокова и постаје транспарентан (Bogdanović et al., 2019). Слика 1 приказује ток рада *blockchain*-а.



Слика 1 - Ток рада blockchain-а, прилагођен од (Bogdanović et al., 2021)

Многи истраживачи су већ увидели могућности коришћења комбинације IoT уређаја и *Blockchain* технологија у агрикултури и ланцима сабдевања хране. Многи недавно објављени чланци разматрају ове технологије у контексту дигиталне трансформације у агрикултурној индустрији и ланцима снабдевања хране. Такође, многи аутори се баве управо проблемима у дизајну и предлажу интеграцију са сервисима вештачке интелигенције.

На пример, неки истраживачи предлажу коришћење паметних уговора са агентима који су ту да провере да ли све стране испуњавају постављене услове и одредбе уговора. Агенти такође могу да издају казне и пенале уколико примете да уговорна страна не испуњава све услове, што чини систем поузданијим и ефикаснијим (Ehsan et al., 2022).

У великом број чланака се предлаже коришћење Blockchain технологије за развој децентралисоване мреже и елиминисање потребе за посредником што отвара могућност праћења било које трансакције и обезбеђује непроменљивост података (Balamurugan et al., 2021; Borah et al., 2020; Ehsan et al., 2022).

Како би се умањиле мане *blockchain* технологије, предлаже се комбиновање дистрибуиране базе података са карактеристикама *blockchain*-а у *BigchainDB* бази података (Tian, 2017).

Иако постоји читав низ предности који долази са примено *blockchain* технологија, можемо приметити да су неке од главних препрека за кориснике мањак знања и одговарајућих тренинга, као и непостојање регулатива везаних за имплементацију ових технологија (Srivastava & Dashora, 2022).

2.1.1. Консензус алгоритми

За имплементацију трансакција у *blockchain* -у потребни су консензус алгоритми и протоколи (Bogdanović et al., 2019; van Hilten et al., 2020; Zheng et al., 2020). Алгоритам консензуса је синхронизовани процес који прати трансакције на нивоу мреже. Алгоритми консензуса потврђују поузданост блокова (Zheng et al., 2020). Након верификације, подаци у бази се ажурирају (Bogdanović et al., 2019). Бројни консензус алгоритми су развијени и коришћени у *blockchain*-у, као што су (Bogdanović et al., 2019):

- *Proof of Work* (PoW) – користи рударење за решавање проблема и верификацију трансакција,
- *Proof of Stake* (PoS) – промовише оне учеснике који имају веће учешће,
- *Proof of Importance* (PoI) – промовише оне кориснике који чешће врше трансакције,
- *Proof of Activity* (PoA) – комбинује PoW и PoS,
- *Delegated Proof of Stake* (DPoS) – бира делегате базирано на њиховој репутацији

- *Pure Proof of Stake* (PPoS) – утицај сваког корисника је пропорционалан броју токена који они имају, насумично и у тајности се бирају они корисници који ће учествовати у гласању

Дистрибуирани консензус алгоритми могу осигурати да се трансакције обављају без интервенција трећих лица попут банака (Zheng et al., 2020). Протоколи омогућавају верификацију трансакција како би се нови блокови додали у постојећи ланац.

2.1.2. Паметни уговори

Свака *blockchain* трансакција се реализује на основу специфичног паметног уговора који садржи уговорне клаузуле написане у компјутерским програмима и оне се аутоматски извршавају када се испуне унапред дефинисани услови (Zheng et al., 2020). Паметни уговори су програми који се сами покрећу и извршавају и који активирају одредбе и услове одређеног договора или уговора користећи софтверске кодове и рачунарску инфраструктуру (Hewa et al., 2021). Извршење сваког члана уговора се бележи као непроменљива трансакција ускладиштена у *blockchain*-у (Zheng et al., 2020).

Животни циклус паметног уговора обухвата следеће фазе (Zheng et al., 2020):

- **Креирање.** Различите интересне стране морају да се договоре око садржаја и циљева уговора. Овај процес укључује пролазак кроз итеративне рунде преговора. Циљ је да се дизајнира, имплементира и валидира паметни уговор који су договориле све стране.
- **Распоређивање**. Након што паметни уговор заврши валидацију, чува се у *blockchain* мрежи. Уговор постаје отворен и доступан како би сви заинтересовани могли да му приступе. Програм чека да се испуне сви услови.
- **Извршење**. Када се фаза распоређивања заврши, трансакције се аутоматски покрећу. Током извршења паметног уговора, нови подаци се валидирају коришћењем одговарајућег консензус алгоритма.
- **Завршетак**. Након извршења паметног уговора, трансакције и резултирајући подаци се чувају у дистрибуираној мрежи. Дигитална имовина договорена правилима уговора се преноси.

За дизајнирање и креирање паметних уговора могу се користити различите платформе за паметние уговоре. Апликације које користе *blockchain* технологије могу се развити помоћу многих одговарајућих *blockchain* платформи (Hewa et al., 2021):

- **Ethereum** платформа омогућава извршавање паметних уговора коришћењем гаса као интерне јединице која представља вредност накнаде за извршење трансакције (Wood, 2014).
- **Hyperledger Fabric** је конзорцијумски blockchain са ограниченим бројем учесника који имају дозволу да се придруже мрежи. То је архитектура заснована на микро услугама погодна за предузећа и користи *Docker* контејнере за покретање кода паметног уговора. Паметни уговор у *Hyperledger*-у се зове *ChainCode*.
- **R3 Corda** је приватна *blockchain* платформа која омогућава реализацију правно употребљивих паметних уговора, на начин сличан уговорима у стварном свету. Фокусира се на примену дигиталних валута.
- **NEM** је платформа за криптовалуте заснована на blockchain-у. Нуди специфичне карактеристике, као што су доказ идентитета, временско означавање докумената и креирање прилагођених дигиталних средстава.
- **Stellar** је специјализована *blockchain* платформа за дигиталне валуте која се користи за апликације дигиталног новца и финансијске трансакције.
- **Waves** платформа за паметне уговоре омогућава корисницима да креирају, издају и преносе ресурсе по веома ниској цени и да размењују токене великом брзином.

2.1.3. Ethereum Blockchain

Ethereum је платформа отвореног кода заснована је на *blockchain* технологији. Етеријум свакоме даје могућност да прави паметне уговоре и децентрализоване апликације. Кориник има могућност да одреди сопствена правила, трансакције и функционалности Апстрактни слој му омогућава да то и постигне користећи blockchain са уграђеним програмским језиком који је Туринг комплетан (Buterin, 2013).

Постоје две врсте налога на етеријум мрежи, а то су екстерни и уговорни налози. Екстерни су контролисани помоћу приватног кључа од стране људи, док су уговорни дефинисани унутар кода самог уговора (Ranđić et al., 2018).

Сваки налог садржи четири поља (Buterin, 2013):

- *nonce* – бројач који осигурава да се трансакција може извршити само једнпут,
- *ether balance* – тренутно стање на налогу,
- *contract code* – хеширан код уговора,
- *storage* – складиште података.

Ethereum дозвољава програмерима да развијају разне децентрализоване апликације, а за писање паметних уговора користи се језик *Solidity*.

Главне одлике *Ethereum* су следеће:

- Више метода размене, као што су криптовалуте, паметни уговори, као и етеријум виртуелна машина.
- Коришћење *Proof-of- Work* система у коме се трансакције верификују рударењем
- Дозвољава и јавне и приватне трансакције.
- Време обраде једног блока *Ethereum*-а је око 12 секунди.
- Мајнерима даје део трошка трансакције.

Предност *Ethereum*-а је та што се не користи само за размену новца већ и за многе друге примене. Компаније као што су *Intel* и *Microsoft* увелико инвестирају средства у *Ethereum* платформу и имају велике планове у будућности. Главна идеја при креирању *Ethereum*-а је била да сама платформа служи за више ствари од самог трансфера новца који је *Bitcoin* омогућавао. О напредку *Ethereum* мреже говори и жеља *PayPal* и *Visa* платформи да пређу на *Ethereum* мрежу, као и дозвола да се врши плаћање у *Ethereum*-у помоћу *Visa* картица и *PayPal* платформе.

У *Ethereum* -у може било ко учествовати у мрежи у било ком тренутку. Самим тим *Ethereum* спада у јавне *blockchain* мреже. На *Ethereum* платформи, сви учесници мреже (или чворови) морају постићи консензус око свих трансакција. Независно да ли чвор учествује у одређеној трансакцији или не. Тренутно *Ethereum* успоставља свој механизам консензуса рударством на темељу алгоритма *Proof-of-Work* (PoW). Трансакције се верификују сложеним математичким обрачунима које обављају машине (процесори, графичке картице) и за чији рад људи који поседују машине бивају награђени. Награда је део трошка трансакције. Због тога *Ethereum* тежи да то промени и пређе на *Proof-of-Stake* (PoS), јер су трансакције због тих обрада веће и људи се

одлучују за друге опције где су цене трансакције ниже. Сви чворови морају се сложити око заједничке књиге и сви имају приступ свим уносима икад забележеним.

2.1.4. Hyperledger Fabric

Hyperledger Fabic је платформа отвореног кода која је креирана да унапреди *blockchain* технологије у различитим индустријама. То је глобална сарадња која укључује лидере у финансијама, банкарству, IoT-у, ланцу снабдевања, производњи и технологији. *Linux Foundation* је оснивач *Hyperledger* -а.

Hyperledger Fabic има следеће циљеве:

- Креирање корпоративних, отворених, дистрибуиране *ledger*-а и база кодова за подршку пословним трансакцијама
- Обезбедити неутралну, отворену инфраструктуру коју покреће заједница, уз подршку техничког и пословног управљања
- Изградите техничку заједницу за развој *blockchain* и заједничких књига PoC-ева, случајева коришћења, теренских стаза и имплементација
- Образовати јавност о тржишним приликама за блоцкцхаин технологију
- Промовисати заједницу користећи приступ алатима на више платформи и оквира.

Hyperledger Fabic је замишљен као основа за развој апликација или решења са модуларном архитектуром. *Hyperledger Fabic* омогућава да компоненте као што су консензус и услуге чланства буду *plug-and-play*. Одликује га модуларни и прилагодљиви дизајн који задовољава широк спектар индустријских примена. Нуди јединствен приступ консензусу који омогућава перформансе на великом нивоу уз очување приватности. *Hyperledger* нема уграђену криптовалуту, тако да не постоји потреба за рударењем. То омогућује скалабилни алгоритам консензуса који је способан за обраду високих стопа трансакција које захтева већина пословних апликација

Blockchain сервиси: Основни део *Hyperledger Fabic*-а је *blockchain* сервис. Консензус менаџер, дистрибуирани *ledger*, *peer-to-peer* протокол и *ledger* база спадају у ову категорију. Менаџер консензуса је одговоран за обезбеђивање интерфејса за консензус алгоритам и примање трансакција из друге *Hyperledger* мреже, тако да се оне извршавају према изабраном типу консензус алгоритма. Консензус је погодан за имплементацију и тренутно постоје три типа консензус алгоритама у *Fabic* -у: *PBFT*

протокол, *SIEVE* алгоритам и *NOOPS*. Дистрибуирани *ledger* је база података коју користе паметни уговори за чување релевантних информација о стању током извршавања трансакција. Ове трансакције садрже *chaincode*, који извршава трансакције.

Сваки чвор чува целокупно стање система на диску. *Fabric* блок структура има велики број поља као што су: верзија, време, хеш трансакције, хеш стања, претходни хеш, консензус мета подаци и нехеш податке. Друга компонента *peer-to-peer* протокола имплементирана је помоћу *googleRPC* -а. Структура поруке у *Fabric* -у је дефинисана бафером протокола. Коришћењем различитих порука, мрежа открива *peer*-ове и извршавање је једнако поуздано као и јавне трансакције. *Ledger* складиште чува стање помоћу *RockDB* -а.

Кориснички сервиси имају функције као што су провера идентитета корисника, регистрација корисника и додељивање одговарајућих нивоа приступа корисницима у зависности од њихових улога. Заједно, ове функције формирају контролу приступа за *Fabric* кориснике. За подршку операцијама ауторизације и управљања идентитетом, они користе инфраструктуру засновану на јавним кључевима. (*A Blockchain Platform for the Enterprise — Hyperledger-Fabricdocs Main Documentation*, 2022)

Chaincode сервиси: *Chaincode*-ови се креирају у оквиру овог сервиса и покрећу се унутар безбедносног контејнера. Састоји се од две компоненте: сигурносног контејнера и сигурносног регистра. Дистрибуирани *ledger* протокол у *Fabric* мрежи се извршава на *peer*-овима. Постоје 2 типа *peer*-ова: *peer* који валидира и *peer* који не валидира. Валидирајући чворови или *peer*-ови су одговорни за постизање консензуса у *Fabric* мрежи. Они потврђују трансакције и одржавају *ledger*. Поред извршења и верификације, чворови који не потврђују валидност издају трансакције чворовима за валидацију. Валидирајући *peer*-ови користи *BFT* (*Byzantine fault tolerance*) као консензусни протокол који аутоматски извршава стање. Трансакције имплементације, трансакције позивања и трансакције упита су типови трансакција које прихвата *chaincode*.

Трансакција имплементације користи инсталирани *chaincode* написан на *Go* језику за *peer*-ове и који је спреман за позивање.

Трансакције позива се користи за одређене *chaincode* -ове који су претходно инсталирани. *Chaincode* извршава трансакције и ажурира стање, показујући на тај начин да ли је трансакција успешна или не.

Трансакција упита враћа улазно стање читајући перзистентна стања *peer*-ова и може бити нелинеарна. *BFT* консензус обезбеђује валидацију. На пример, ако постоји n валидационих чворова међу којима има f "неисправних" чворова, под претпоставком да ће се већина понашати произвољно са ограничењем f<(n/3), сви остали чворови исправно извршавају *chaincode*. Да би се извршиле на *PBFT*, *chaincode* трансакције морају бити детерминистичке. Сигурносна инфраструктура за *Fabric* укључује упис и ауторизацију трансакција кроз сертификацију путем јавног кључа. Поузданост *chaincode*-а, остварена кроз *in-band* шифровање и стања за кључеве специфичне за *blockchain*, омогућена је за све *peer*-ове са сертификатима за регистрацију у *blockchain* (Vemuri, 2019).

2.1.5. NEM

NEM (*New Economy Movement*) представља blockchain технолошку платформу која ефикасно управља подацима по ниској цени. Има своју криптовалуту XEM која није намењена за плаћање. Има подршку за *Nano wallet* (Seth, 2022).

NEM користи концепт *Proof-of-Importance* (PoI) механизма. Друге мреже криптовалута користе *Proof-of-Work* (PoW) или *Proof-of-Stake* (PoS) системе који захтевају уређаје за рударење (мајнинг) како би функционисале. Награде се тако додељују рударима у складу са њиховим доприносом што даје велику предност уређајима са високом снагом за обраду података, али чини процес енергетски неефикасним.

PoI механизам даје предност онима који су највише укључени у систем на основу количине XEM-ова и периода задржавања новца. Да би била награђена новчићима, особа мора да има више од 10 000 уложених XEM-ова на свом рачуну (сваког дана 10% укупних средстава на рачуну прелази у “уложена” средства). Поред тога, награде се добијају и на основу учешћа у трансакцијама са другима у мрежи (Seth, 2022).

Жетва представља алтернативу стандардном процесу рударења који троши велике количине енергије и успорава трансакције. Код жетве, особа повезује свој налог на постојећи супер-чвор и на основу њеног PoI ранга се повећавају шансе за додавање

нових блокова у систем, без потребе за повећањем снаге за обраду података (Seth, 2022).

NEM користи *Eigentrust++* алгоритам који помаже да се избалансира притисак на мрежу како би она била ефикасна и прилагодљива.

NEM *blockhain* се може применити у различитим ситуацијама као што су:

- Повезивање јавних и приватних *blockchain*-ова
- Праћење и одржавање програма лојалности
- Систем за праћење логистике
- Системи за сигурно гласање и чување података
- Лако креирање, расподела и трговина новим криптовалутама
- Коришћење NEM -а као API-ја и интеграција у апликације на различитим уређајима

2.1.6. Algorand

Приликом развоја *blockchain* технологија, увек се јавља проблем такозване *blockchain* трилеме која представља проблем балансирања између сигурности, скалабилности и децентрализације. *Algorand* се позиционира управо као *blockchain* који је успео да реши овај проблем.

Велики део решења се везује за начин на који се нови блок додаје на постојећи ланац. У случају *Algorand*-а, нови блок се креира у две фазе. По насумичном принципу, бира се један токен. Његов власник је управо корисник који предлаже додавање новог блока. Друга фаза подразумева избор 1000 токена из система. Њихови власници постају део комитета који прихвата предложени токен. Узимајући у обзир да систем садржи 10 милијарди токена, шансе за превару у систему и утицање на избор су јако ниске. Такође, сваки корисник који је предложен на свом рачунару покреће криптографску лутрију за сваки токен који поседује. Уколико ни један токен није изабран, глас тог корисника се занемарује, у супротном на мрежу се шаље број освојених гласова и мишљење о новом блоку (Micali, 2019).

Проблем који се у *blockchain* мрежи често јављао јесте *forking*. Наиме, у оквиру *blockchain* мреже као што је на пример *Bitcoin*, постоји могућност нестанка блокова у ситуацији да се *blockchain* раздвоји у две одвојене гране. *Algorand* ту могућност своди

на миниму, прецизније вероватноћа да дође до *forking*-а је једном у 10^{18} блокова (Micali, 2019).

Сваки *blockchain* има сопствену изворну валуту која игра кључну улогу у подстицању доброг понашања мреже која се зове *Algo*. Корисници који имају *Algo* токене, могу да се региструју за учешће у консензусу, односно процесу предлагања и гласања о новим блоковима. *Algo* се такође користи као помоћни токен. Приликом развоја апликације, потребни су *Algo*-и за плаћање накнада за трансакције и служе као минимални депозити за чување података на *blockchain*-у (*Why Algorand? - Algorand Developer Portal*, 2022).

2.2.Интернет интелигентних уређаја – IoT

Интернет интелигентних уређаја није једна технологија, већ иновативни скуп неколико комплементарних технологија уједињених са циљем премошћавања јаза између дигиталног и физичког света (Balaji & Roy, 2016; de Vass et al., 2018).

Типична IoT мрежа укључује четири главна кључна слоја (Xu et al., 2014):

- Сензорни слој који интегрише различите врсте „ствари“ као што су RFID ознаке, сензори, актуатори
- Мрежни слој који подржава пренос информација преко жичане или бежичне мреже
- Сервисни слој који интегрише услуге и апликације кроз међуверску технологију
- Интерфејс за приказ информација кориснику и омогућава интеракцију са системом

2.2.1. IoT у праћењу порекла производа

У контексту праћења порекла производа, интернет интелигентних уређаја можемо дефинисати као мрежу физичких објеката који су дигитално повезани да чују, надгледају и комуницирају унутар компаније и између компаније и њеног ланца снабдевања омогућавајући агилност, видљивост, праћење и дељење информација ради лакшег благовременог планирања, контроле и координације процеса ланца снабдевања (Ben-Daya et al., 2017).

Утицаји IoT -а на ланца снабдевања (Abdel-Basset et al., 2018):

- Унапређено управљање залихама: коришћење интелигентних уређаја омогућава доступност тачних информација о залихама у сваком тренутку и елиминисање потребе за ручним уносом информација које је чест узрок грешака.
- Управљање ланцем снабдевања у реалном времену: у традиционалном ланцу снабдевања информације поседује само један партнер у ланцу уместо да се оне деле. Али нове технологије RFID ознака омогућавају праћење свих врста информација (датум производње и истека рока трајања, период гаранције...) чиме се може постићи ефикасно управљање ланцем снабдевања.
- Максимално повећање транспарентности логистике: све информације о транспорту (стање транспорта, одредиште, итд.) биће доступне за цео ланац снабдевања кроз коришћење паметних уређаја. Ово ће повећати шансе за праћење и уштеду робе. Такође може минимизирати трошкове поврата и имати велики утицај на задовољство купаца.

3. Пројектовање паметног система за праћење порекла органског меда

3.1. Пројектни задатак

Систем којим се врши праћење порекла органског меда реализован је кроз веб апликацију. Она обухвата део система који намењен пчеларима и купцима, који представљају кључне стејкхолдере.

За сваку кошницу се коришћењем различитих сензора прикупљају подаци о локацији кошнице и квалитету ваздуха у њеној околини. На основу њих се потврђује да је мед настао у одговарајућим условима што доприноси одржавању квалитета меда.

Пчелари и купци имају приступ веб апликацији. Пчелари је користе како би прегледали локацију својих кошница и податке са сензора. Купци имају могућност да очитају QR код и прегледају додатне информације о купљеном производу укључујући информације са сензора и локације на којима се мед налазио.

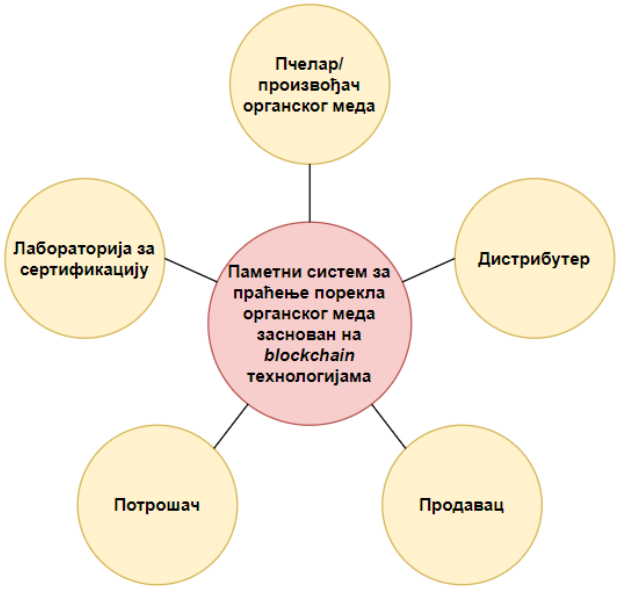
3.2. Пословни модел

Пословни модел је подељен на девет кључних сегмената, приказаних у Табели 1, који обухватају унутрашње и спољашње утицаје (Osterwalder & Pigneur, 2010). Произвођачи меда и пчеларске заједнице се овде виде као партнери ћије знање може значајно допринети да развијена платформа и паметни уговори буду дизајнирани у складу са стварним потребама. Платформи се може приступити директно или преко информација са амбалаже производа. Предложена платформа омогућава приступ детаљним информацијама о пореклу и условима производње и на тај начин максимизује поверење у органске производе од меда. Сви учесници у ланцу се виде као потенцијални купци јер предложени систем свима доноси бенефите као што су повећавање ефикасности процеса и смањење административних трошкова. Приходи се могу убирати кроз продају производа од органског меда и покривају трошкове као што су развој и одржавање платформе и IoT инфраструктуре.

Табела 1 - Пословни модел

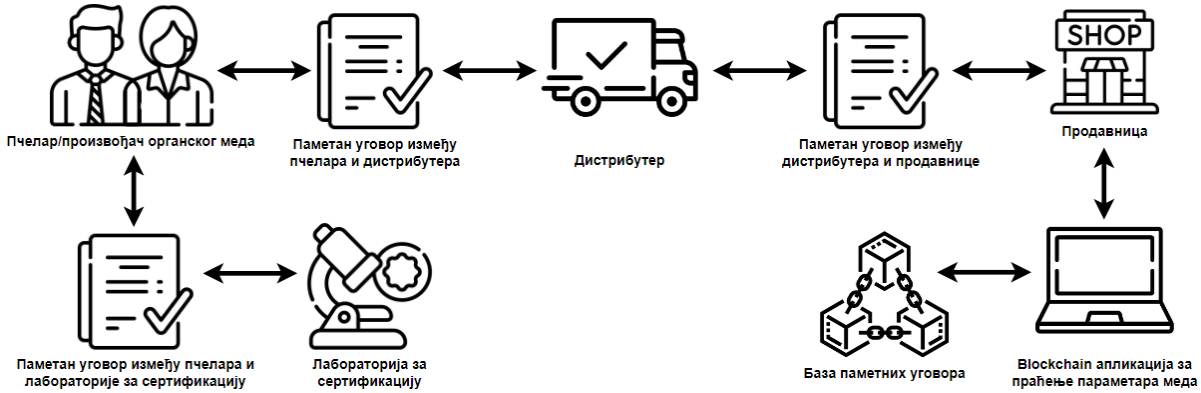
<table>
<tr><td rowspan="2">Кључни партнери
Пчеларске заједнице
Појединачни пчелари</td><td>Кључне активности
Развој платформе
Генерисање паметних уговора</td><td rowspan="2">Предлог вредности
Приступ детаљним информацијама о пореклу и условима производње
Максимизирање поверења у органске производе од меда</td><td>Однос са купцима
Пружање свих потребних информација којима се може приступити било кад и било где</td><td rowspan="2">Сегменти купаца
Купци
Пчелари
Произвођачи
Дистрибутери</td></tr>
<tr><td>Кључни ресурси
Веб платформа
Blockchain оријентисан софтвер
Подаци са IoT уређаја</td><td>Канали
Веб апликација
Амбалажа производа</td></tr>
<tr><td colspan="3">Структура трошкова
Развој и одржавање
IoT инфраструктура</td><td colspan="2">Токови прихода
Продаја производа од меда (производи са сертификатом порекла и квалитета)</td></tr>
</table>

У управљању предложеним *blockchain* системом могу учествовати различити актери уз контролу квалитета коју регулише надлежно министарство.



Слика 2 - Стејкхолдери предложеног система за праћење порекла органског меда (Bogdanović et al., 2021)

Свака трансакција у ланцу снабдевања органске производње меда се евидентира у *blockchain*-у након закључивања паметних уговора између различитих заинтересованих страна (слика 3).

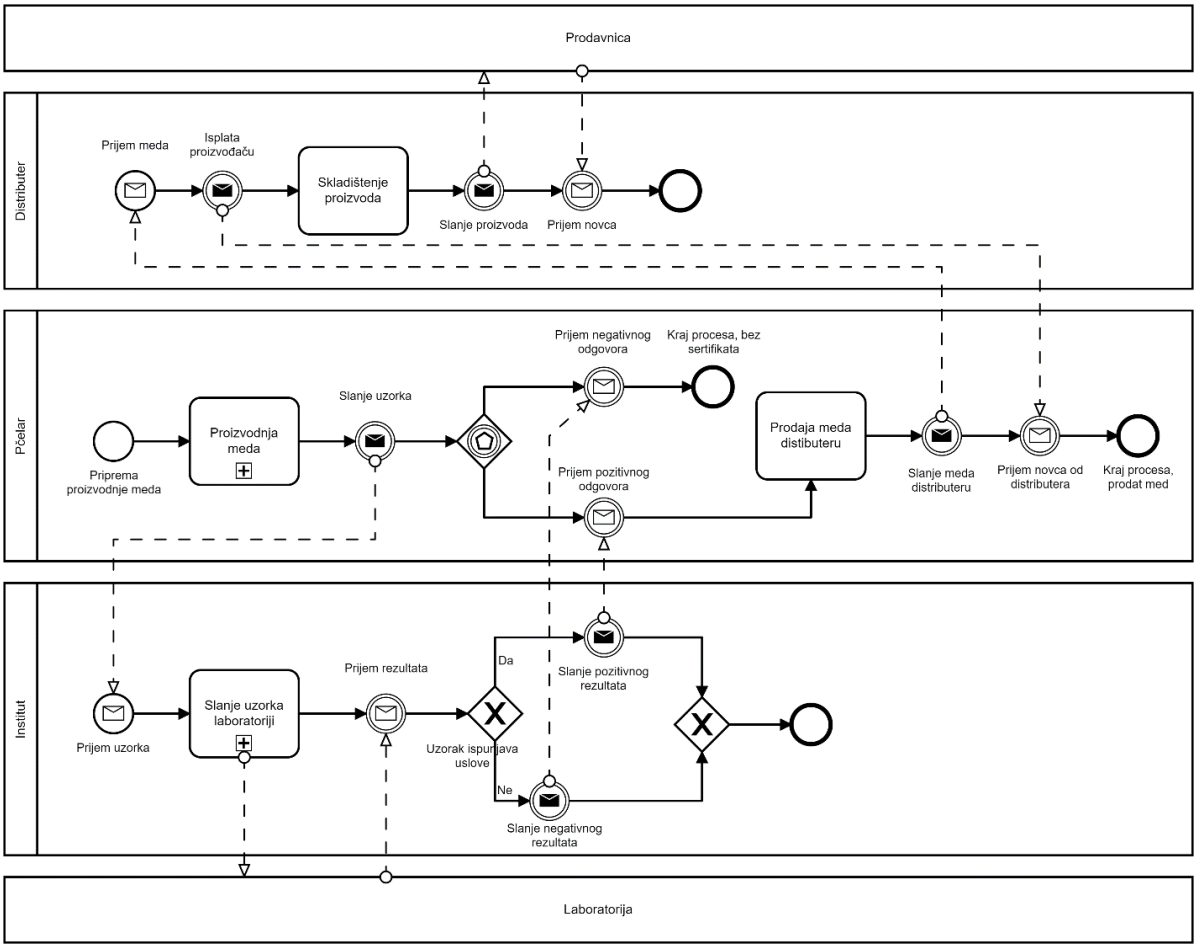


Слика 3 - Паметни уговори у систему за праћење органског меда (Bogdanović et al., 2021)

Произвођач органског меда, у оквиру процеса контроле, потписује паметни уговор са лабораторијом за сертификацију, потврђеном од стране Министарства, која је задужена за испитивање порекла и квалитета хране. Након што утврди да је органски мед у производњи задовољава све дефинисане стандарде, потписује се паметни уговор између произвођача органског меда и дистрибутера/трговца на велико. Након тога се потписује паметни уговор између дистрибутера/велетрговца и продавца на мало и на тај начин се комплетира ланац. Паметни уговори који су договорени и потписани се чувају у *ledger*-у како би били свим учесницима били доступни у путем *blockchain* апликације. Сви трансфери новца током процеса закључивања паметних уговора могу бити регулисани од стране банке или путем унутрашњих трансфера криптовалута.

3.3.Моделовање процеса

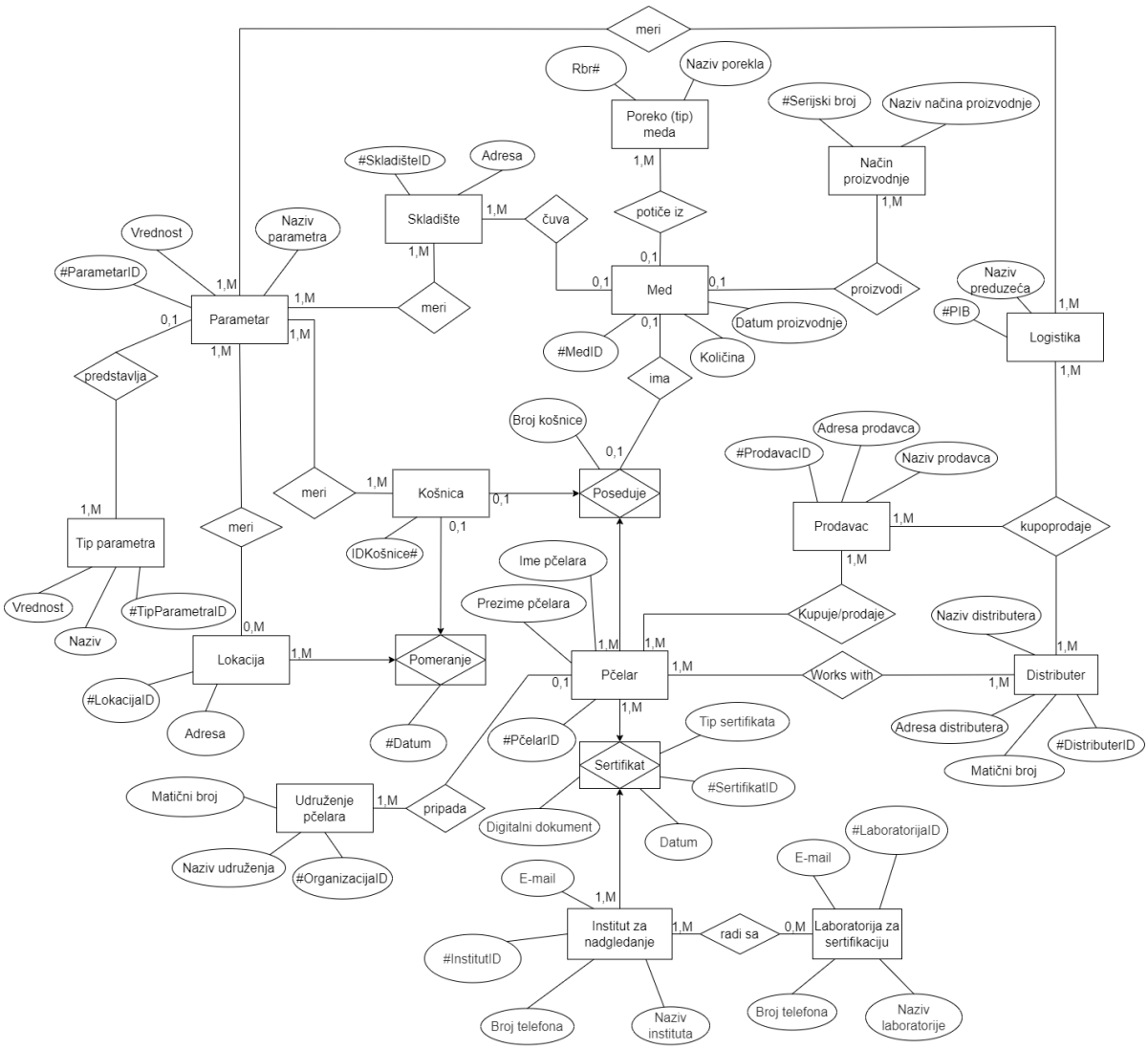
Мед пролази пут од пчелара до продавнице као што је приказано на слици 4. Процес започиње пчелар који врши производњу меда. Након тога, узорак се шаље на институт како би се извршиле потребне анализе у лабораторији. Након тога мед добија сертификат о исправности. Уколико је резултат неповољан, процес се завршава. Уколико је резултат повољан, процес се наставља. Мед се продаје дистрибутеру који преузима робу и складишти је. Након тога, мед се испоручује продавници како би био доступан крајњем купцу.



Слика 4 - Модел процеса за процес праћења порекла хране

3.4.Моделовање података

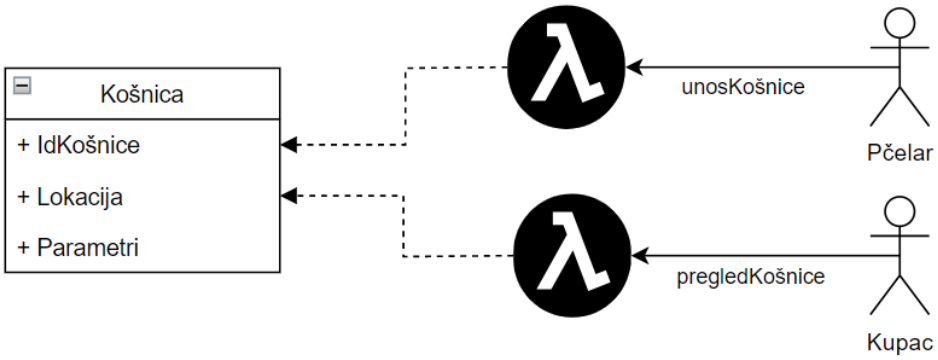
У наставку је приказан модел података целог система који се може видети на слици 5. Можемо приметити да се за сваког пчелара бележи идентификациони број, име и презиме, као и којој пчеларско организацији припада. Мед који пчелар производи се прати по кошници. За сваку кошницу пратимо локацију и добијамо инфомацију у случају премештања кошнице. Поред тога пратимо различите параметре који гарантују квалитет меда. Мед се може продавати преко дистрибутера или директно продавници. Сваки пчелар је дужан да своје производе пошаље на анализу који му издаје сертификат о о производњи у складу са стандардима.



Слика 5 - Модел података система за праћење порекла хране

3.5.Моделовање паметних уговора

Основни елементи који се користе за моделовање паметних уговора су уговорне стране, трансакције, везе и објекта моделовања. Терминологија се наравно може разликовати у складу са типом *blockchain*-а за који се паметан уговор моделује. На слици 6 је приказан пример једног модела паметног уговора представљен у складу са iContractML 2.0 моделом представљеним у (Hamdaqa et al., 2022).



Слика 6 - Модел паметног уговора

На датом примеру, објекат моделовања је кошница која садржи и одговарајуће параметре. Затим имамо две уговорне стране, пчелара и купца, као и две трансакције које се покрећу када желимо да унесемо нове податке или извршимо измену. Трансакције могу имати индикатор који означава покретање догађаја када се одговарајућа трансакција изврши (Hamdaqa et al., 2022).

3.5.1. Паметни уговори у *Algorand*-у

Algorand паметни уговори (ASC1) су мали програми који служе различитим функцијама на blockchain-у и раде на слоју-1. Паметни уговори су подељени у две главне категорије, паметни уговори и паметни потписи. Тип уговора који је написан ће одредити када и како ће се проценити логика програма. Обе врсте уговора су написане на *Transaction Execution Approval Language* (TEAL), који је језик сличан асемблеру који се тумачи помоћу *Algorand* виртуелне машине (AVM) која ради унутар *Algorand* чвора. TEAL програми се могу писати ручно или коришћењем језика *Python* са *PyTeal* компајлером. (*Smart Contract Details - Algorand Developer Portal*, n.d.).

Паметни уговори су уговори који се након примене могу даљински позвати са било ког чвора у Алгоранд blockchain-у. Када се примени, инстанца уговора на ланцу се назива апликација и додељује јој се идентификациони број апликације. Ове апликације се покрећу специфичном врстом трансакције која се зове трансакција позива апликације. Ове апликације на ланцу управљају примарном децентрализованом логиком dApp-а (децентрализоване апликације) прикзаном кроз следеће функционалности (*Smart Contract Details - Algorand Developer Portal*, n.d.).

- Промена стања везаног за апликацију (глобално стање) или стања везаног за налог повезан на апликацију (локално стање).

- Приступ вредностима на ланцу, као што су стања на рачуну, параметри конфигурације средстава или време потребно за валидацију и верификацију блока.
- Извршавање трансакције у оквиру извршења логике. Једна врста трансакције коју могу да изврше је трансакција позива апликације која омогућава једној апликацији да позове другу. Ова могућност позивања других апликација омогућава састављање (*composing*) апликација.
- Имају придружени рачун апликације који може држати Algos или ASA билансе и може се користити као рачуни за депоновање у ланцу.

3.6.Софтверска архитектура система

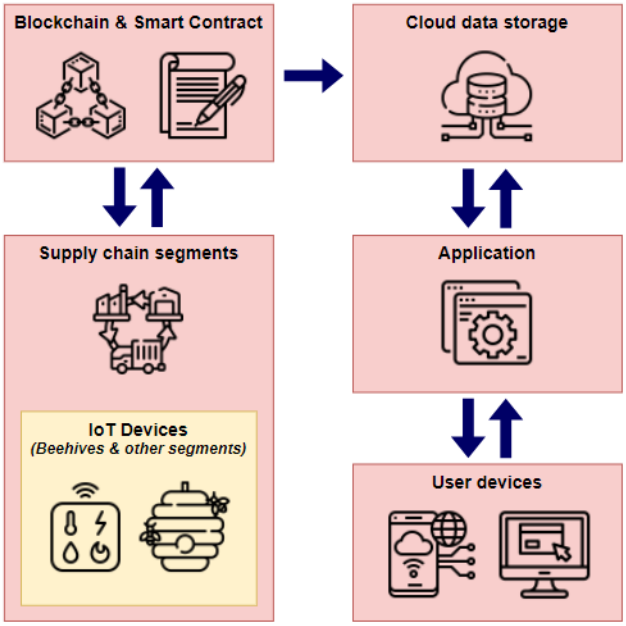
Веб апликација која имплементира предложени систем за праћење порекла органског меда представља *full stack* апликацију са клијентском и серверском страном.

Кориснички део апликације је реализован коришћењем *React* библиотеке и приказује све потребне делове корисничког интерфејса. У случају пчелара приказују је подаци са различитих сензора за сваку кошницу. На страницу намењену купцу води нас QR код где се приказују различити подаци о купљеном производу, укључујући праћење путање меда од кошнице до саме продавнице и осталих података који потврђују његов квалитет.

Серверски део апликације је реализован у *Node.js* коришћењем *Express.js* оквира. Подаци који се прикупљају са сензора се уписују у *MongoDB* базу података. За олакшано повезивање и комуникацију између *Express.js* оквира и *MongoDB* базе података користи се *Mongoose* библиотека. Неопходни подаци се затим прослеђују на *blockchain* мрежу где се чека на испуњење свих услова за окидање паметног уговора.

Подаци са кошнице прикупљени коришћењем сензора се коришћењем MQTT протокола шаљу на мрежу „у облаку“ како би били лако доступни корисницима са различитим улогама.

На слици 7 приказана је софтверска архитектура на примеру праћења производње меда. Свака *IoT* компонента укључује сензоре и интелигентне уређаје постављене широм ланца снабдевања. У оквиру слоја повезивања могу се користити различити протоколи и мреже, као што су *Zigbee, Bluetooth, LoRa* и мобилне мреже.



Слика 7 - Софтверска архитектура (Lukovac et al., 2022; Pranto et al., 2021)

Произвођачима су обезбеђене апликације које могу да се користе за приступ свим подацима из *IoT* компоненте и управљање њиховом фармом кошница. Компоненте интернета ствари су постављене у целом ланцу снабдевања и прилагођене специфичностима сваког учесника у ланцу (производња, транспорт, сертификација, трговина итд.). Прикупљени подаци се претходно обрађују у оквиру сваке компоненте учесника, а изабрани подаци се чувају у блокчејн мрежи, како би били доступни другим заинтересованим странама у ланцу. *Cloud* инфраструктура се користи за складиштење података (Bogdanović et al., 2022).

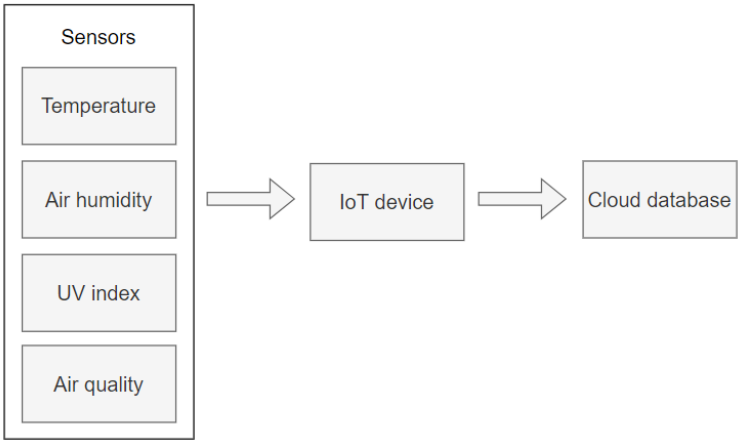
Крајњи купци могу да очитају QR код производа и приступе веб апликацији коришћењем различитих уређаја као што је паметни телефон или таблет. У апликацији могу пронаћи све податке везане за производњу меда, његово порекло и путању којом се кретао. Сви подаци се чувају „у облаку“ и могу бити прегледани од стране крајњег корисника.

У свим сегментима ланца набавке, информације се прикупљају коришћењем IoT уређаја и шаљу на blockchain мрежу. Подаци се редовно проверавају у случају одступања.

4. Развој прототипа

4.1. Предложена инфраструктура

Приказ предложене инфраструктуре која би се могла користити за прикупљање потребних информација налази се на слици 8. На левој страни можемо видети више сензора који прикупљају податке о окружењу и условима око кошнице. Они чине први ниво нашег система и шаљу све податке на *IoT* уређај. Сви подаци које прикупља *IoT* уређај шаљу се у *cloud* базу података користећи интернет везу.

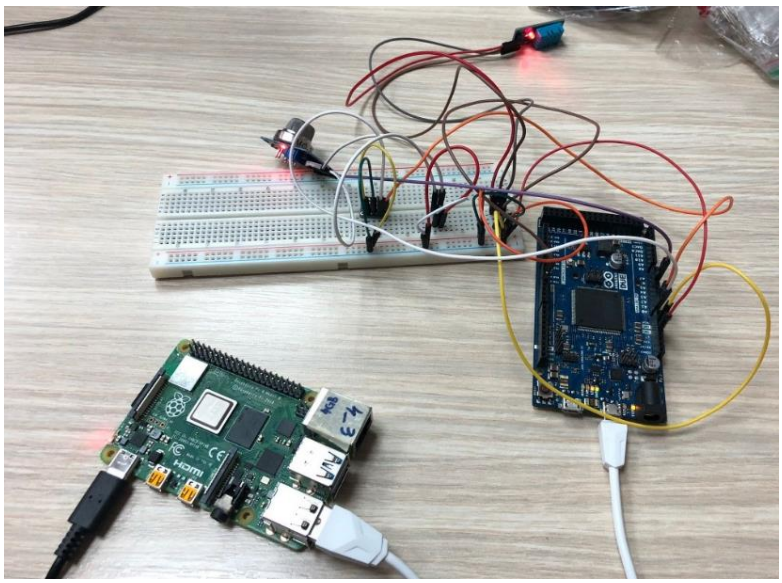


Слика 8: Предложена инфраструктура паметне кошнице

Складиштење свих података у *cloud* бази података чини их лако доступним за будуће сврхе развоја веб апликације. Кључни подаци који су важни за доказивање порекла меда морају се чувати унутар *blockchain* мреже како би се онемогућило манипулисање. Имплементација овог дела биће представљена у наредном истраживању

4.2. Развој прототипа уређаја

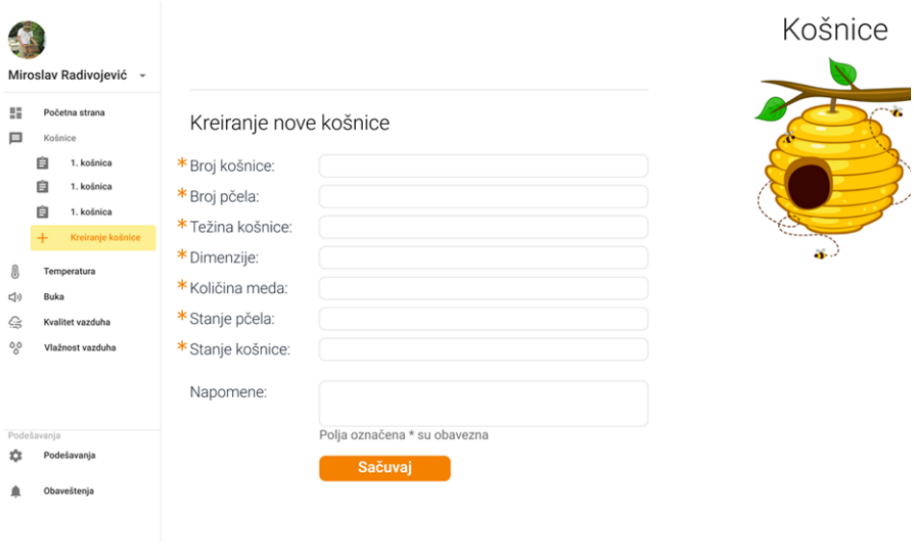
Предложена инфраструктура је имплементирана у лабораторијским условима. На слици 9 приказани су одговарајући сензори за мерење влажности ваздуха, температуре ваздуха, UV зрачења и количине пропана у ваздуху. Наведени сензори се затим повезују са микропроцесором и шаљу на *Raspberry Pi* рачунар који је способан за стримовање na *cloud* базу података.



Слика 9: Прототип сензора на паметној кошници

4.3.Предложени изглед апликације

Вишеструки прикази предложеног изгледа апликације приказани су на сликама 10-16. На слици 10 приказана је страница за креирање нове кошнице. Пчелар је дужан да попуни идентификациони број кошнице и прикљученог *IoT* уређаја, као и димензије и тежину кошнице.

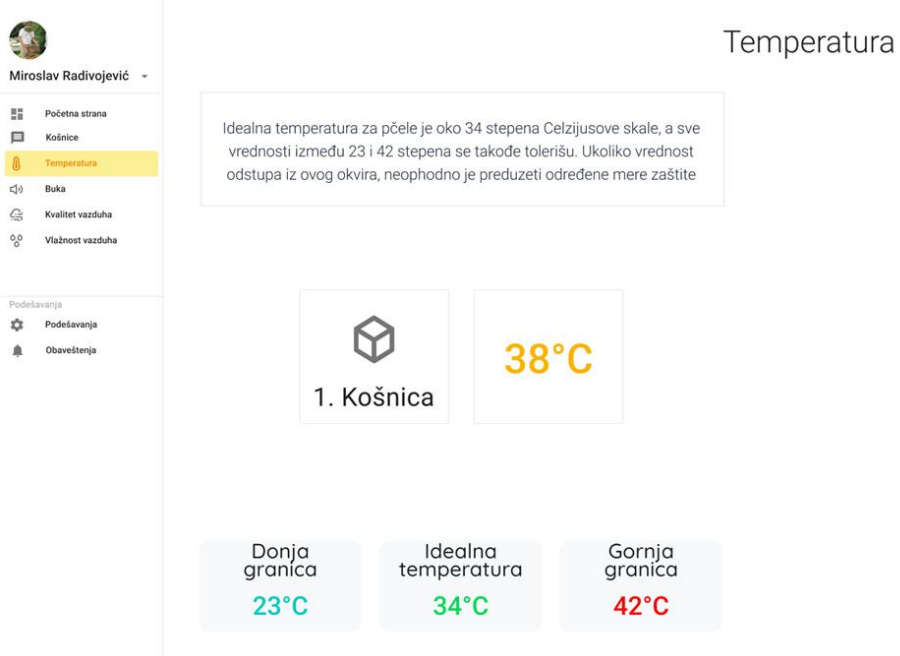


Слика 10: Прављење нове кошнице

На слици 11 приказана је страница на којој пчелар може изабрати одређену кошницу и видети доступне информације о њој. Пчелар такође може додати нову кошницу из овог приказа.

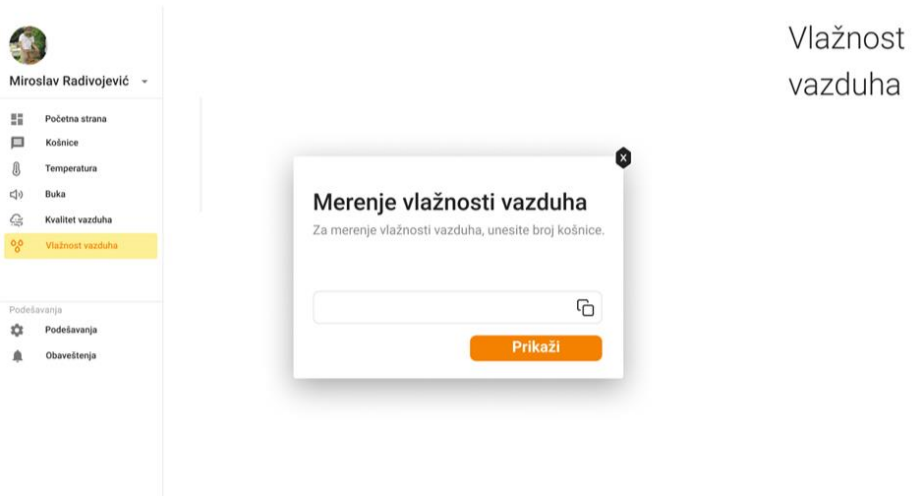
Слика 11: Поглед свих кошница

Преглед температуре може се видети на слици 12. Овде корисник може да види тренутну температуру одређене кошнице, заједно са информацијама о горњој и доњој дозвољеној температури за кошницу. Ако је измерена температура ван опсега, потребне су додатне мере заштите.



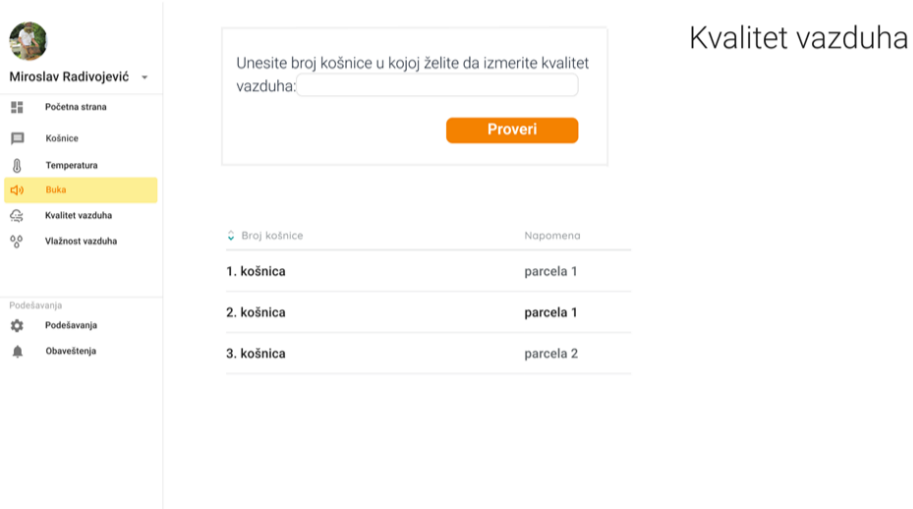
Слика 12: Преглед температуре

На слици 13 приказан је искачући прозор где корисник може да унесе идентификациони број кошнице и погледа информације о влажности ваздуха око кошнице.



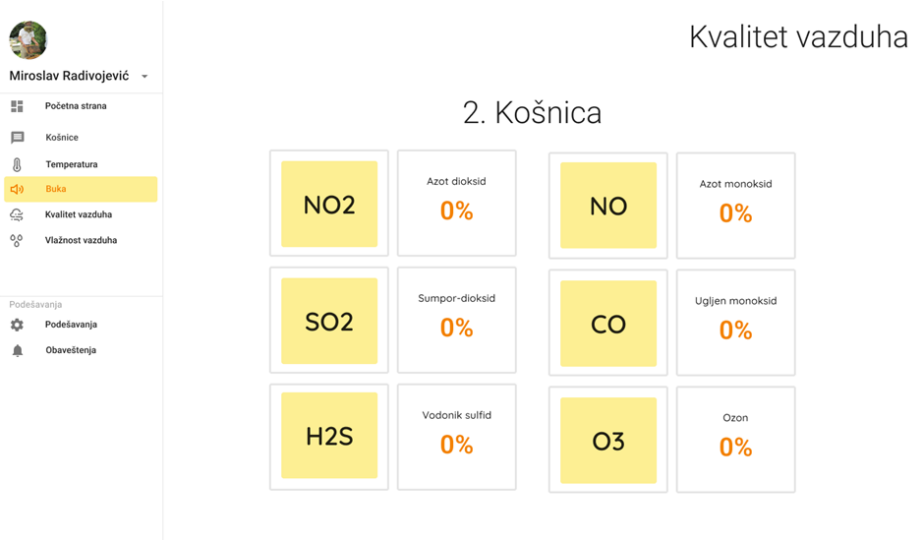
Слика 13: Мерење влажности ваздуха за одређену кошницу

Приликом провере квалитета ваздуха око пчелињака, пчелару се приказује листа кошница за избор, као што је приказано на слици 14.



Слика 14: Избор кошнице за проверу квалитета ваздуха

Након избора одређене кошнице апликација ће показати више различитих параметара за квалитет ваздуха са одговарајућим вредностима, као што је приказано на слици 15.



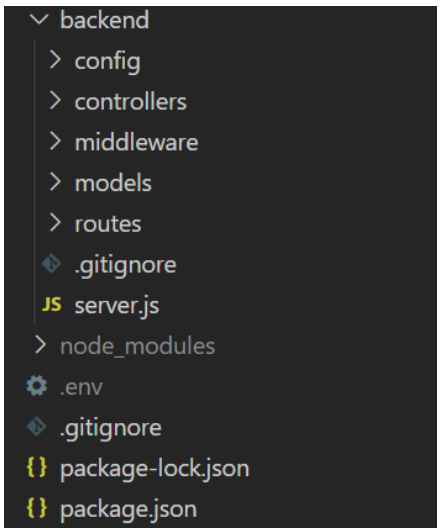
Слика 15: Приказ квалитета ваздуха за одређену кошницу

Све пчелиње кошнице су опремљене и *GPS* уређајем који приказује њихову тренутну локацију. Сваки пут када се промени локација кошнице, пчелар је обавештен као што је приказано на слици 16.

Слика 16: Упозорење при промени места кошнице

4.4. Апликациона логика

Функционалности веб апликације садржане су у оквиру апликационе логике. Њихова улога јесте да приме захтеве са корисничког интерфејса, обраде их, изврше одговарајући упит над базом и врате потребне информације. Костур *backend* дела апликације може се видети на слици 17.



Слика 17 - Приказ организације фајлова на backend делу аплицаије

У фајлу package.json нам је битно да приметимо који додаци су инсталирани што се може видети у делу dependencies. Инсталирана је MongoDB база података, Express оквир и библиотека dotenv.

За једноставније повезивање користимо Mongoose, који представља алат за лакше моделирање објеката за MongoDB базу података. Креирана фунцкија *connectDB()* се извози као модул, што јој омогућава да се једноставно позива из других фајлова, што можемо видети на примеру фајла server.js који је приказан у коду испод.

```
const express = require("express");
const dotenv = require("dotenv").config();
const { errorHandler } = require("./middleware/errorMiddleware");
const connectDB = require("./config/db");
const port = process.env.PORT || 5000;

connectDB();

const app = express();

app.use(express.json());
app.use(express.urlencoded({ extended: false }));

app.use("/api/something", require("./routes/someRoutes"));
app.use("/api/bee", require("./routes/beeRoutes"));
app.use("/api/keeper", require("./routes/keeperRoutes"));
app.use("/api/users", require("./routes/userRoutes"));

app.use(errorHandler);

app.listen(port, () => console.log(`Server started on port ${port}`));
```

Слика 18 - Повезивање са базом и креирање апликације

Наведени код приказује повезивање на одговарајућу базу података, креирање саме апликације. Затим омогућава читање json података у телу захтева и специфицира суфикс за руте који се користи за слање HTTP захтева у зависности од групе рута.

```
const mongoose = require("mongoose");

const userSchema = mongoose.Schema(
  {
    email: {
      type: String,
      required: [true, "Email value needs to be filled in"],
      unique: true,
    },
    name: {
      type: String,
      required: [true, "Name value needs to be filled in"],
    },
    password: {
      type: String,
      required: [true, "Password value needs to be filled in"],
    },
  },
  {
    timestamps: true,
  }
);

module.exports = mongoose.model("User", userSchema);
```

Слика 19 - Шема објекта корисника

Као што је раније наведено, за моделирање објекта се користи Mongoose. Шема садржи поља за име, мејл адресу и лозинку и сва поља су обавезна. Поље за мејл адресу има додатно ограничење да мора бити јединствено у бази. Такође, имамо и поље које бележи тренутак у коме је креиран корисник.

Све операције које је могуће вршити над објектом корисник приказане су у фајлу userController.js који се може видети у коду испод.

```
const bcrypt = require("bcryptjs");
const User = require("../models/userModel");
const asyncHandler = require("express-async-handler");

const registerUser = asyncHandler(async (req, res) => {
  const { name, email, password } = req.body;
  if (!name || !email || !password) {
    res.status(400);
    throw new Error("Please fill in all info");
  }
  # check database for same user
  const userExists = await User.findOne({ email });
  # return appropriate message or continue
  if (userExists) {
    res.status(400);
    throw new Error({ text: "User with same email already exists" });
  }
  # encrypting password value
  const salt = await bcrypt.genSalt(10);
  const hashedPassword = await bcrypt.hash(password, salt);

  const user = await User.create({
    name,
    email,
    password: hashedPassword,
  });

  if (user) {
    res.status(201).json({
      _id: user.id,
      name: user.name,
      email: user.email,
    });
  } else {
    res.status(400);
    throw new Error("User not registered");
  }
});
module.exports = {
  registerUser,
  loginUser,
};
```

Слика 20 - Контролер и функција регистрације (Traversy, 2022)

Приликом регистрације корисника, прво се проверава да ли су сви потребни подаци успошно попуњени и уколико нису враћа се грашка. Након тога се проверава да ли корисник са истом мејл адресом већ постоји у бази и уколико постоји враћа се грешка. Уколико су сви подаци исправно попуњени, лозинка се хешира и креира се одговарајући корисник. У случају успешно извршене операције, враћа се статусни код 201 и одговарајући подаци корисника.

Функција за пријаву корисника је нешто једноставнија, те је није неопходно додатно објашњавати.

Одговарајуће руте за операције регистрације и пријаве корисника дате су у фајлу userRoutes.js приказаном у наредном коду.

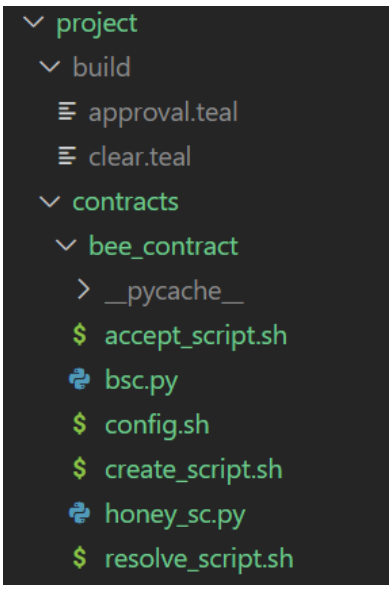
```
const express = require("express");
const router = express.Router();
const {
    registerUser,
    loginUser
} = require("../controllers/userController");
# parts of route added to /api/users
router.post("/", registerUser);
router.post("/login", loginUser);

module.exports = router;
```

Слика 21 - Руте за операције пријаве и регистрације

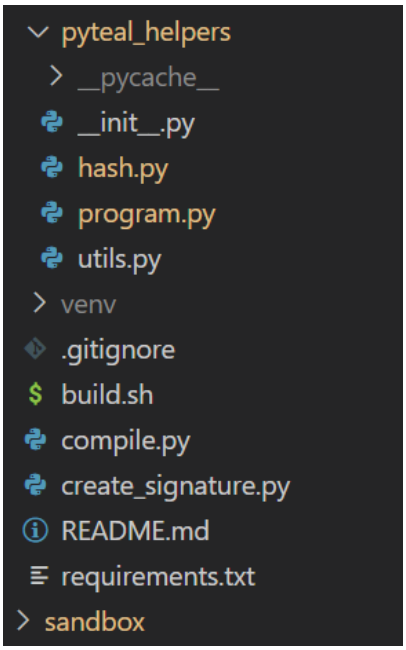
4.5.Имплементација паметног уговора

Имплементација паметног уговора биће приказана на примеру паметног уговора између пчелара и лабораторије за сертификацију. За развој паметног уговора коришћена је *Algorand blockchain* платформа. *Algorand* омогућава писање уговора коритећи *Python* или прецизније *PyTeal* библиотеку која је развијена искључиво за потребе *Algorand*-а. Друга варијанта јесте коришћење *Reach* језика који је сличан *JavaScript*-у и омогућава једноставније писање паметних уговора како у *Algorand*-у, тако и у *Ethereum blockchain*-у. За потребе овог паметног уговора биће коришћена *PyTeal* библиотека. Предуслов за покретање паметног уговора је постојање *Sandbox*-а који је покренут користећи *Docker* контејнере.

Костур пројекта који имплементира паметан уговор може се видети на сликама 18 и 19.



Слика 22 - Приказ организације фајлова при изради паметног уговора - први део



Слика 23 - Приказ организације фајлова при изради паметног уговора - други део

Сваки паметни уговор садржи *approval()* и *clear()* функцију. Унутар *approval()* функције потребно је дефинисати променљиве које ће бити коришћене унутар паметног уговора. Оне могу бити глобалне и локалне. С обзиром да ће паметан уговор бити коришћен од стране више пчелара и више лабораторија, изабране су локалне варијабле које дају више флексибилности и омогућавају вишеструко покретање уговора у истом тренутку.

Након варијабли дефинишемо операције које паметан уговор извршава током свог животног циклуса. Свака операција уствари представља једну трансакцију или групу трансакција које се извршавају на *Algorand blockchain* мрежи.

```
def approval():
    # locals - enable multiple users to use the contract simultaneously
    local_lab = Bytes("lab")  # byteslice - address of the account
    local_honey_wager = Bytes("wager")  # uint64 amount of algos transfered to
lab
    local_honey_guid = Bytes("honey_guid")  # byteslice - unique id
    local_honey_type = Bytes("honey_type")  # uint64 honey type number
    local_honey_resolve = Bytes("honey_resolve")  # byteslice - honey quality

    # operations contract needs to support
    op_create = Bytes("create")
    op_accept = Bytes("accept")
    op_resolve = Bytes("resolve")
```

Слика 24 - Варијабле и операције паметног уговора

Функција *event()* очитава у којој фази животног циклуса се налази паметан уговор. Фаза *opt_in* обухвата укључивање различитих уговорних страна у уговор. Унутар фазе *no_op* можемо дефинисати понашање уговора у складу са његовом применом.

```
def event(
    init: Expr = Reject(),
    delete: Expr = Reject(),
    update: Expr = Reject(),
    opt_in: Expr = Reject(),
    close_out: Expr = Reject(),
    no_op: Expr = Reject(),
) -> Expr:
    return Cond(
        [Txn.application_id() == Int(0), init],
        [Txn.on_completion() == OnComplete.DeleteApplication, delete],
        [Txn.on_completion() == OnComplete.UpdateApplication, update],
        [Txn.on_completion() == OnComplete.OptIn, opt_in],
        [Txn.on_completion() == OnComplete.CloseOut, close_out],
        [Txn.on_completion() == OnComplete.NoOp, no_op],
    )
```

Слика 25 - Функција за проверу фазе животног циклуса

Поменута *approval()* функцијадефинише шта се дешава приликом одређене фазе животног циклуса. Приликом *opt_in* фазе позива се функција *reset()* поставља све варијабле на њихову иницијалну вредност. Параметар *Int(0)* означава да из низа налога који се чувају приликом позива паметног уговора узимамо онај који има индекс 0, што је увек онај налог који је и покренуо трансакцију. Тако ће се у случају пчелара ова функција позвати приликом операције *op_create*, а у случају лабораторије приликом

операције *op_accept*. Приликом фазе *no_op*, на основу аргумента прослеђеног приликом позива паметног уговора одређује се која операција ће бити извршена *op_create*, *op_accept*, *op_resolve*.

```
return program.event(
    # since everything is in local storage
    # we can Approve transaction immediately
    init=Approve(),

    opt_in=Seq(
        # setting all variables to initial value
        reset(Int(0)),  # Txn.account[0]
        Approve(),
    ),

    no_op=Seq(
        Cond(
            [Txn.application_args[0] == op_create, create_quality_check()],
            [Txn.application_args[0] == op_accept, accept_quality_check()],
            [Txn.application_args[0] == op_resolve, resolve_quality_check()],
        ),
        Reject(),
    ),
)
```

Слика 26 - Позив одговарајуће операције у зависности од фазе животног циклуса

Уколико је преко аргумента прослеђена операција create, извршава се функција *create_quality_check()*. У оквиру ове операције је потребно извршити групну трансакцију с обзиром да имамо две трансакције, где је прва унос потребних информација о меду, а друга извршавање плаћања на налог саме аликације како би средства била резервисана. Ограничења су постављена у складу са препорукама (*Guidelines - Algorand Developer Portal*, 2022) . Проверавамо да ли је друга трансакција типа *Payment*, да ли је налог на који се плаћање врши стварно адреса наше апликације, као и који налози имају приступ трансакцији плаћања.

Проверавамо да ли су два налога прикључена паметном уговору, да ли су све потребне променљиве постављене на иницијалну вредност и да ли је одговарајући број аргумената прослеђен.

На самом крају, одговарајуће вредности се уписују у локалне варијабле на налозима пчелара и лабораторије.

```
@Subroutine(TealType.none)
def create_quality_check():
    return Seq(
        # setting group transaction
        program.check_self(
            group_size=Int(2),
            group_index=Int(0)
        ),
        program.check_rekey_zero(2),
        Assert(
            And(
                # setting payment transaction parameters
                Gtxn[1].type_enum() == TxnType.Payment,
                Gtxn[1].receiver() == Global.current_application_address(),
                Gtxn[1].close_remainder_to() == Global.zero_address(),

                App.optedIn(Int(1), Int(0)),

                # check that no account has data from another contract
                # they are free to join another game
                is_empty(Txn.sender()),
                is_empty(Txn.accounts[1]),

                # check for the commitment argument
                Txn.application_args.length() == Int(3),
            ),
        ),
        # putting data in local variables
        App.localPut(Txn.sender(), local_lab, Txn.accounts[1]),
        App.localPut(Txn.sender(), local_honey_wager, Gtxn[1].amount()),
        App.localPut(Txn.sender(), local_honey_guid,
                     Txn.application_args[1]),
        App.localPut(Txn.accounts[1], local_honey_guid,
                     Txn.application_args[1]),
        App.localPut(Txn.sender(), local_honey_type,
                     Btoi(Txn.application_args[2])),
        App.localPut(Txn.accounts[1], local_honey_type,
                     Btoi(Txn.application_args[2])),
        Approve(),
    )
```

Слика 27 - Трансакција креирања

Функција *accept_quality_check()* се покреће приликом трансакције инициране од стране лабораторије. Потребно је проверити да ли је пчелар прикључен и да ли је његов налог одговарајући. Проверава се да ли је тражени износ за услугу једнак износу који је заложен од стране пчелара и да ли су сви аргументи попуњени. Одговарајуће вредности се уписују у локалне варијабле на налогу лабораторије.

```
    @Subroutine(TealType.none)
    def accept_quality_check():
        return Seq(
            # setting single transaction
            program.check_self(
                group_size=Int(1),
                group_index=Int(0)
            ),
            program.check_rekey_zero(1),
            Assert(
                And(
                    # check that beekeeper account has opted in
                    App.optedIn(Txn.accounts[1],
                                Global.current_application_id()),
                    # check that beekeeper account has contract with exactly this
account
                    App.localGet(Txn.accounts[1], local_lab) == Txn.sender(),
                    # check if amount == to wager on beekeeper account
                    Btoi(Txn.application_args[2]) == App.localGet(
                        Txn.accounts[1], local_honey_wager),

                    Txn.application_args.length() == Int(3),
                ),
            ),
            # putting data in local variables
            App.localPut(Txn.sender(), local_lab, Txn.accounts[1]),
            App.localPut(Txn.sender(), local_honey_type,
                         App.localGet(Txn.accounts[1], local_honey_type)),
            App.localPut(Txn.sender(), local_honey_wager,
                         Btoi(Txn.application_args[2])),
            App.localPut(Txn.sender(), local_honey_resolve,
                         Txn.application_args[1]),
            Approve(),
        )
```

Слика 28 - Трансакција којом лабораторија прихвата уговор

Последња операција *resolve* покреће функцију *resolve_quality_check()*. Потребно је проверити да ли је налог лабораторије који је пчелар запамтио, исти онај који се у систему приказује као други налогу у низу. По истом принципу се проверава и супровтан смер.

Уколико су сву наведени услови испуњени, проверавамо да ли је унесени износ накнаде довољан да покрије трошкове трансакција. Налогу лабораторије се исплаћује одговарајући износ, а пчелар добија резултате анализе.

```
    @Subroutine(TealType.none)
    def resolve_quality_check():
        return Seq(
            # setting single transaction
            program.check_self(
                group_size=Int(1),
                group_index=Int(0)
            ),
            program.check_rekey_zero(1),
            Assert(
                And(
                    # checking if account that created has only one opponent and
it is the same one that is set and vice versa
                    App.localGet(
                        Txn.sender(), local_lab) == Txn.accounts[1],
                    App.localGet(Txn.accounts[1],
                                 local_lab) == Txn.sender(),
                    # check same wager
                    App.localGet(Txn.sender(), local_honey_wager) ==
App.localGet(Txn.accounts[1], local_honey_wager),
                    # check resolve from lab is not empty
                    App.localGet(Txn.accounts[1], local_honey_resolve) !=
Bytes(""),
                    # check reveal from lab
                    Txn.application_args.length() == Int(2),
                ),
            ),
            # send wager to lab, write resolve to beekeeper local storage
            Seq(
                # checking set fee is greater than 2 times minimum fee
                Assert(Txn.fee() >= Global.min_txn_fee() * Int(2)),
                send_payment(Int(1), App.localGet(
                    Txn.sender(), local_honey_wager)),
                App.localPut(
                    Txn.sender(), local_honey_resolve, Txn.application_args[1])
            ),
            reset(Txn.sender()),
            reset(Txn.accounts[1]),
            Approve(),
        )
```

Слика 29 - Трансакција којом се шаљу резултати и плаћа накнада

Само плаћање извршава функција *send_payment()* којој шаљемо индекс налога на који се средства исплаћују и износ. Плаћање се обавља као унутрашња трансакција (eng. *Inner transaction*). Постављамо тип трансакције на *Payment*, затим налог на који се исплаћују средства и одговарајући износ. Такође, наводимо који налог ће сносити саме трошкове извршавања трансакције.

```
@Subroutine(TealType.none)
def send_payment(account_index: Expr, amount: Expr):
    return Seq(
        # building inner transaction
        InnerTxnBuilder.Begin(),
        InnerTxnBuilder.SetFields({
            TxnField.type_enum: TxnType.Payment,
            TxnField.receiver: Txn.accounts[account_index],
            TxnField.amount: amount,
            # optional moving payment of transaction
            TxnField.fee: Int(0),
        }),
        InnerTxnBuilder.Submit(),
    )
```

Слика 30 - Унутрашња трансакција за извршавање плаћања

Битно је нагласити да се *PyTeal* уствари користи за генерисање *TEAL* кода, који се затим смешта на *blockchain*. Раније поменуте основне функције паметног уговора *approval()* и *clear()* се преводе у одвојене фајлове *TEAL* кода. Проверава се верзија *TEAL*-а која се користи, а затим се пролази кроз сваку фазу животног циклуса.

```
#pragma version 5
txn ApplicationID
int 0
==
bnz main_l19
txn OnCompletion
int DeleteApplication
==
bnz main_l18
txn OnCompletion
int UpdateApplication
==
bnz main_l17
txn OnCompletion
int OptIn
==
bnz main_l16
txn OnCompletion
int CloseOut
==
bnz main_l15
txn OnCompletion
int NoOp
==
bnz main_l7
```

Слика 31 - Провера фазе животног циклуса користећи TEAL

Дат је пример уписа вредности у променљиву. Адреса налога који је покренуо трансакцију се уписује на стек. Уписује се локална варијабла на којој се врши измена.

Из низа прослеђених аргумената, узима се други и уписује на стек. Сви аргументи који се прослеђују су типа byte, тако да је потребно узети вредност аргумента, претворити је у позитиван цели број и уписати је на стек. На самом крају врши се функција уписивања претворене вредности у одговарајућу променљиву.

```
txn Sender
byte "honey_type"
txna ApplicationArgs 2
btoi
app_local_put
```

Слика 32 - Приказ уписа вредности у локалну променљиву користећи TEAL

Све трансакције се позивају користећи GOAL интерфејс командне линије. За сваку од трансакција направљена је скрипта која извршава потребне команде. За све три операције потребне вредности вучемо из фајла *config.sh.*

Приликом операције *accept()* прослеђујемо идентификациони број апликације, налог који позива операцију, налог који ће учествовати у трансакцији и одговарајуће аргументе које је потребно проследити.

```
#!/usr/bin/env bash

# load variables from config file
source "$(dirname ${BASH_SOURCE[0]})/config.sh"

# create accept transaction
goal app call \
    --app-id "$APP_ID" \
    -f "$LAB_ACCOUNT" \
    --app-account "$BEEKEEPER_ACCOUNT" \
    --app-arg "str:accept" \
    --app-arg "str:$HONEY_REVEAL" \
    --app-arg "int:$WAGER"
```

Слика 33 - Скрипта која симулира прихватање уговора

Приликом операције *create()* креирамо прву трансакцију којом шаљемо одговарајуће податке као и у претходном примеру, али је не извршавамо одмах већ је уписујемо у фајл *honey-call.tx.* То радимо зато што имамо и трансакцију плаћања и неопходно је да их групишемо пре извршења. Другу трансакцију такође уписујемо у фајл, комбинујемо их, потписујемо и тек тада можемо да их извршимо.

```bash
#!/usr/bin/env bash

# load variables from config file
source "$(dirname ${BASH_SOURCE[0]})/config.sh"

# create challenge transaction
goal app call \
    --app-id "$APP_ID" \
    -f "$BEEKEEPER_ACCOUNT" \
    --app-account "$LAB_ACCOUNT" \
    --app-arg "str:create" \
    --app-arg "str:$HONEY_GUID" \
    --app-arg "int:$HONEY_TYPE" \
    -o honey-call.tx

# --dryrun-dump -o tx.dr
# create wager transaction
goal clerk send \
    -a "$WAGER" \
    -t "$APP_ACCOUNT" \
    -f "$BEEKEEPER_ACCOUNT" \
    -o honey-wager.tx

# group transactions
cat honey-call.tx honey-wager.tx > honey-combined.tx
goal clerk group -i honey-combined.tx -o honey-grouped.tx
goal clerk split -i honey-grouped.tx -o honey-split.tx

# sign individual transactions
goal clerk sign -i honey-split-0.tx -o honey-signed-0.tx
goal clerk sign -i honey-split-1.tx -o honey-signed-1.tx

# re-combine individually signed transactions
cat honey-signed-0.tx honey-signed-1.tx > honey-signed-final.tx

# send transaction
goal clerk rawsend -f honey-signed-final.tx
```

Слика 34 - Скрипта која симулира креирање уговора

Скрипта за операцију resolve() је слична претходним уз додатан параметар fee који представља износ накнаде за извршавање трансакција.

5. Анализа спремности стејкхолдера за усвајање предложеног модела

Према истраживању Савеза пчеларских организација Србије (СПОС), утврђено је да 40 одсто меда на тржишту хране Србије није оригинално (*Savez Pčelarskih Organizacija Srbije*, n.d.). У 2020. години из Србије је извезено 2700 тона меда у вредности од око 13 милиона евра, док је у Србију увезено 611 тона меда у вредности од 1,5 милиона евра. Просечна извозна цена била је 4,85 евра, а увозна 2,46 евра, што говори о квалитету меда који Србија извози у односу на онај који увози. Количина лажног меда у извозу је мања јер купци из других земаља могу сами да обаве тестове и открију вештачки мед. То доводи до ситуације да се највећи део вештачког меда продаје на локалном тржишту Србије (Bogdanović et al., 2022).

5.1. Анализа спремности произвођача меда за прихватање предложеног модела

5.1.1. Истраживачка питања и дизајн

Циљ истраживања је да се прикупе информације о спремности српских произвођача меда за коришћење овог система.

Спремност произвођача меда проучавана је помоћу упитника, који је заснован на VAM моделу (*Value-Added Model*) (Koedel et al., 2015). У почетној анкети, 63 индивидуална произвођача меда су упозната са предложеним системом и замољени су да попуне упитник. Очекује се да ће резултати дати одговоре на следећа питања:

- Колико су појединачни произвођачи меда у Србији упознати са могућностима ИоТ и блоцкцхаин технологија?
- Које су уочене предности коришћења предложеног система?
- Да ли су произвођачи меда спремни да прихвате потребна улагања и уочене жртве?

5.1.2. Познавање *IoT* и *blockchain* технологија

У табели 2 приказани су резултати који се односе на упознатост произвођача меда са *IoT* и *blockchain* технологијама у контексту производње меда. Више од половине испитаника (52,4%) раније није чуло за *blockchain*. Резултати указују на опште интересовање за *IoT* и *blockchain*, али су идентификовани проблеми у вези са законодавством и очекиваном једноставношћу коришћења.

Табела 2 - Резултати познавања IoT и blockchain технологија (Bogdanović et al., 2022)

Питање	**Средња вредност**	**Стандардна девијација**
Спремност за паметне уговоре у ланцу набавке меда у Србији	3.56	1.26
Значај коришћења *IoT* и *blockchain*-а у партнерским организацијама	3.14	1.28
Перципирана једноставност преласка на *IoT* и *blockchain*	3.08	1.21
Перципирана једноставност коришћења	2.90	1.20
Квалитет релевантног националног законодавства и прописа	2.38	1.29

5.1.3. Перципиране користи

Табела 3 приказује резултате о уоченим предностима коришћења *IoT* и *blockchain*-а у праћењу производње меда.

Табела 3 - Резултати о перципираној користи (Bogdanović et al., 2022)

Питање	**Ср. Вр.**	**Ст. Дев.**
Перципирани утицај доступности *IoT* података у *blockchain*-у на намеру у куповини	3.76	1.28
Перципирани утицај доступности *IoT* података у *blockchain*-у на селекцију партнера	4.13	1.17
Спремност за дељење *IoT* података у *blockchain*-у	4.43	0.91
Перципиране користи паметних уговора	4.11	1.05
Очекивана конкурентска предност	3.16	1.24
Перципирана корисност од праћења података кроз ланац снабдевања	4.08	1.13

Резултати указују на високу спремност произвођача меда да поделе своје *IoT* податке у *blockchain*-у. Поред тога, треба истаћи да произвођачи меда сматрају да се веће користи од технологија огледају у праћењу читавог ланца снабдевања и приликом одабира и сарадње са партнерима, него код повећања куповне намере купаца или конкурентске предности.

5.1.4. Спремност да се прихвате потребна улагања и уочене жртве

Табела 4 приказује резултате о спремности за инвестиције и уочене жртве. Произвођачи меда су највише заинтересовани за улагање у образовање. Интересовање за улагања у *IoT* и *blockchain* систем је углавном близу неутралног, али су више заинтересовани за улагање уколикона тај начин цена њихових производа порасте. Ово може бити важан блокирајући фактор за увођење ове врсте иновација. Поред тога, најниже оцене су добијене за питања која се односе на спремност потрошача да купе скупљи производ или сложеност увођења нових технологија у ланац снабдевања. Ове уочене жртве је потребно пажљиво проверити и размотрити приликом дизајнирања пословног модела за предложени систем.

Табела 4 - Резултати о спремности за инвестиције и уочене жртве (Bogdanović et al., 2022)

Питање	**Средња вредност**	**Стандардна девијација**
Спремност за инвестиције, уз исту цену производа	3.05	1.35
Спремност за инвестиције, уз повећану цену производа	3.67	1.28
Перципирана спремност купаца на повећану цену производа	2.79	1.14
Спремност за улагање у *IoT* опрему и сервисе	3.13	1.34
Спремност за улагање времена и новца у *IoT* и *blockchain* едукације	4.06	1.05
Спремност за куповину скупљег производа са сертификацијом базираном на *blockchain*-у	3.22	1.26
Перципирана комплексност за увођење *IoT* и *blockchain* у ланац снабдевања	2.84	1.26

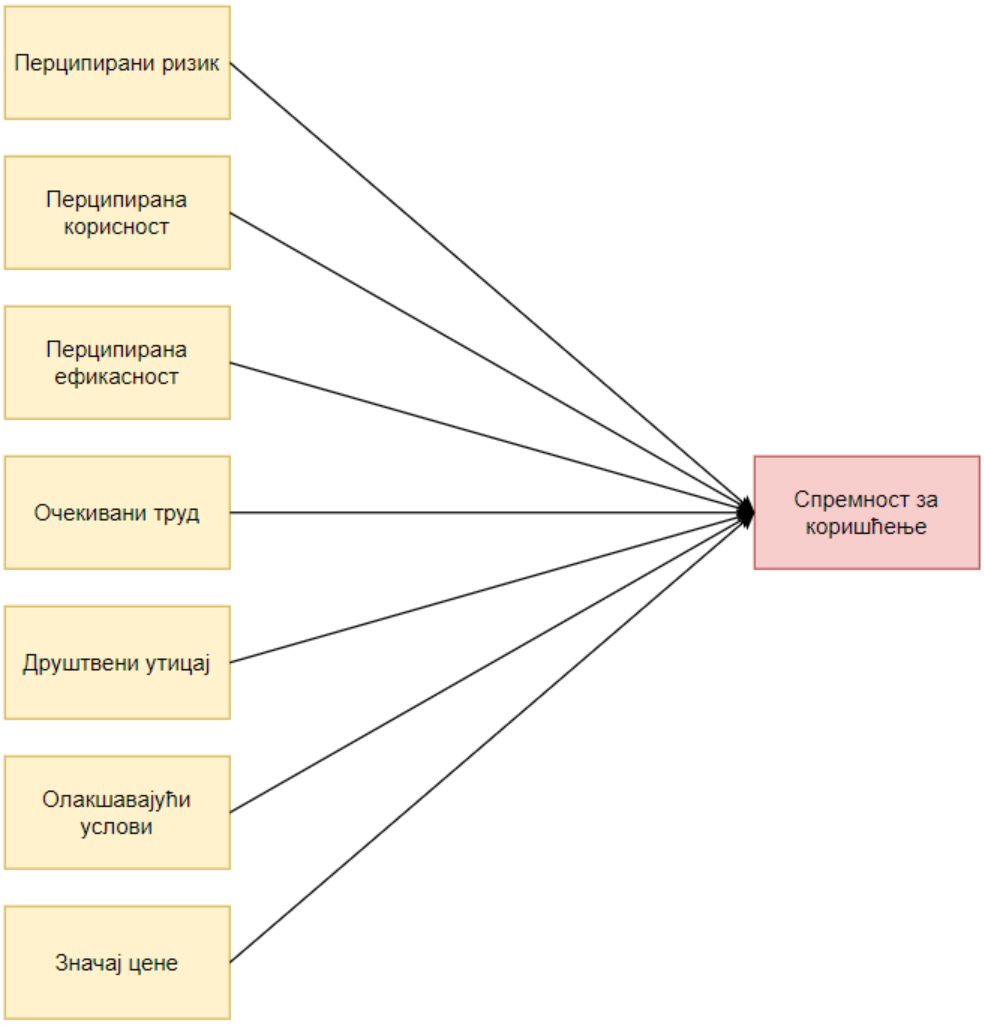
5.2. Анализа спремности потрошача меда за прихватање предложеног модела

Испитивање мотива потрошача меда за коришћење паметног система којим би се пратило порекло органског меда заснованог на *blockchain* технологијама реализовано је по угледу на моделе UTAUT2 и TAM (Marangunić & Granić, 2014; Venkatesh et al., 2012). Као зависна варијабла изабрана је „Спремност за коришћење“, налик *Behavioral intention* варијабли која је део UTAUT2 модела. Она нам је корисна у покушају да предвидимо понашање корисника за систем који још увек није тестиран у реалним условима.

Изабране независне варијабле су:

- Перципирани ризик, по аналогији са *Perceived Risk* варијаблом из UTAUT2 модела.
- Перципирана корисност, по аналогији са *Perceived Usefulness* варијаблом из TAM модела.
- Перципирана ефикасност, по аналогији са *Perceived Efficiency* варијаблом из UTAUT2 модела.
- Очекивани труд, по аналогији са *Expected Effort* варијаблом из UTAUT2 модела.
- Друштвени утицај, по аналогији са *Social Influence* варијаблом из UTAUT2 модела.
- Олакшавајући услови, по аналогији са *Facilitating Conditions* варијаблом из UTAUT2 модела.
- Значај цене, по аналогији са *Price Value* варијаблом из UTAUT2 модела.

Зависна варијалба, као и наведене независне варијабле приказане су на слици 35.



Слика 35 - Модел за испитивање спремности за коришћење система за праћење порекла меда

Постављене су следеће хипотезе:

X1: Перципирани ризик утиче на спремност потрошача меда да користи систем за праћење меда заснован на *blockchain*-у.

X2: Перципирана корисност утиче на спремност потрошача меда да користи систем за праћење меда заснован на *blockchain*-у.

X3: Перципирана ефикасност утиче на спремност потрошача меда да користи систем за праћење меда заснован на *blockchain*-у.

X4: Очекивани труд утиче на спремност потрошача меда да користи систем за праћење меда заснован на *blockchain*-у.

X5: Друштвени утицај утиче на спремност потрошача меда да користи систем за праћење меда заснован на *blockchain*-у.

X6: Олакшавајући услови утичу на спремност потрошача меда да користи систем за праћење меда заснован на *blockchain*-у.

X7: Значај цене утиче на спремност потрошача меда да користи систем за праћење меда заснован на *blockchain*-у.

5.2.1. Инструменти истраживања

За прикупљање података о спремности потрошача меда да користи паметни систем за праћење меда коришћена је метода анкетирања. Подаци потребни за истраживање су прикупљени коришћењем упитника који се састојао од 4 демографска питања и 31 питања везаног за мерење спремности за коришћење предложеног система за праћење порекла меда. На почетку сваке секције упитника, испитаници могу прочитати текст са описом сваке секције на која се питања односе.

Списак индикатора који се користе како би се анализирало колико су потрошачи спремни за коришћење предложеног система за праћење порекла меда приказан је у табели 5.

Табела 5 - Индикатори којим се испитује намера потрошача за коришћење предложеног система

Индикатор	Ср. Вр.	Стд. Дев.	Excess Kurtosis	Skewness
Перципирани ризик:				
PR1	4.71	0.536	3.317	-1.845
PR2	4.773	0.529	7.245	-2.6
PR3	3.703	1.028	0.026	-0.617
PR4	3.595	1.054	-0.342	-0.434
PR5	4.74	0.597	9.3	-2.788
PR6	3.19	0.915	-0.084	0.229
PR7	2.814	1.018	-0.323	-0.237
Перципирана корисност:				
PU1	4.818	0.518	17.877	-3.773
PU2	3.717	1.37	-0.878	-0.676
PU3	3.561	1.125	-0.744	-0.099

Индикатор	Ср. Вр.	Стд. Дев.	Excess Kurtosis	Skewness
Перципирана ефикасност:				
PE1	3.387	1.363	-1.083	-0.344
PE2	3.922	1.203	-0.339	-0.841
PE3	4.164	1.007	0.755	-1.146
PE4	3.747	1.075	0.028	-0.728
Очекивани труд:				
EE1	4.086	0.97	1.016	-1.082
EE2	4.316	1.006	1.539	-1.46
EE3	3.554	1.158	-0.237	-0.587
EE4	3.677	1.326	-0.733	-0.659
Друштвени утицај:				
SI1	2.197	0.723	-1.048	-0.318
Олакшавајући услови:				
FC1	4.647	0.614	4.791	-1.932
FC2	4.342	0.768	0.101	-0.921
FC3	4.606	0.674	2.656	-1.746
FC4	4.684	0.634	6.465	-2.347
FC5	4.279	1.147	1.528	-1.587
Значај цене:				
PV1	4.204	0.813	1.732	-1.102
PV2	4.059	0.947	0.467	-0.913
PV3	3.312	1.27	-0.967	-0.286
PV4	3.855	1.191	-0.693	-0.674
Склоност за коришћење:				
BI1	4.123	0.95	0.891	-1.032
BI2	4.212	0.914	0.789	-1.049
BI3	4.39	0.998	2.356	-1.702

Вредности параметара *kurtosis* и *skewness* за већи део индикатора улазе у границе између -1 и +1 које нам показују да ли вредности припадају нормалној расподели. Вредности за које наведени параметри излазе ван опсега показују да постоји одређено

одступање од нормалне расподеле. Одређена одступања се сматрају значајним и у складу са тим неки индикатори ће бити изузети из даљег истраживања.

5.2.2. Узорак

Креирани упитник је попунило укупно 269 испитаника. Демографски подаци испитаника који чине узорак су приказани у табели 6.

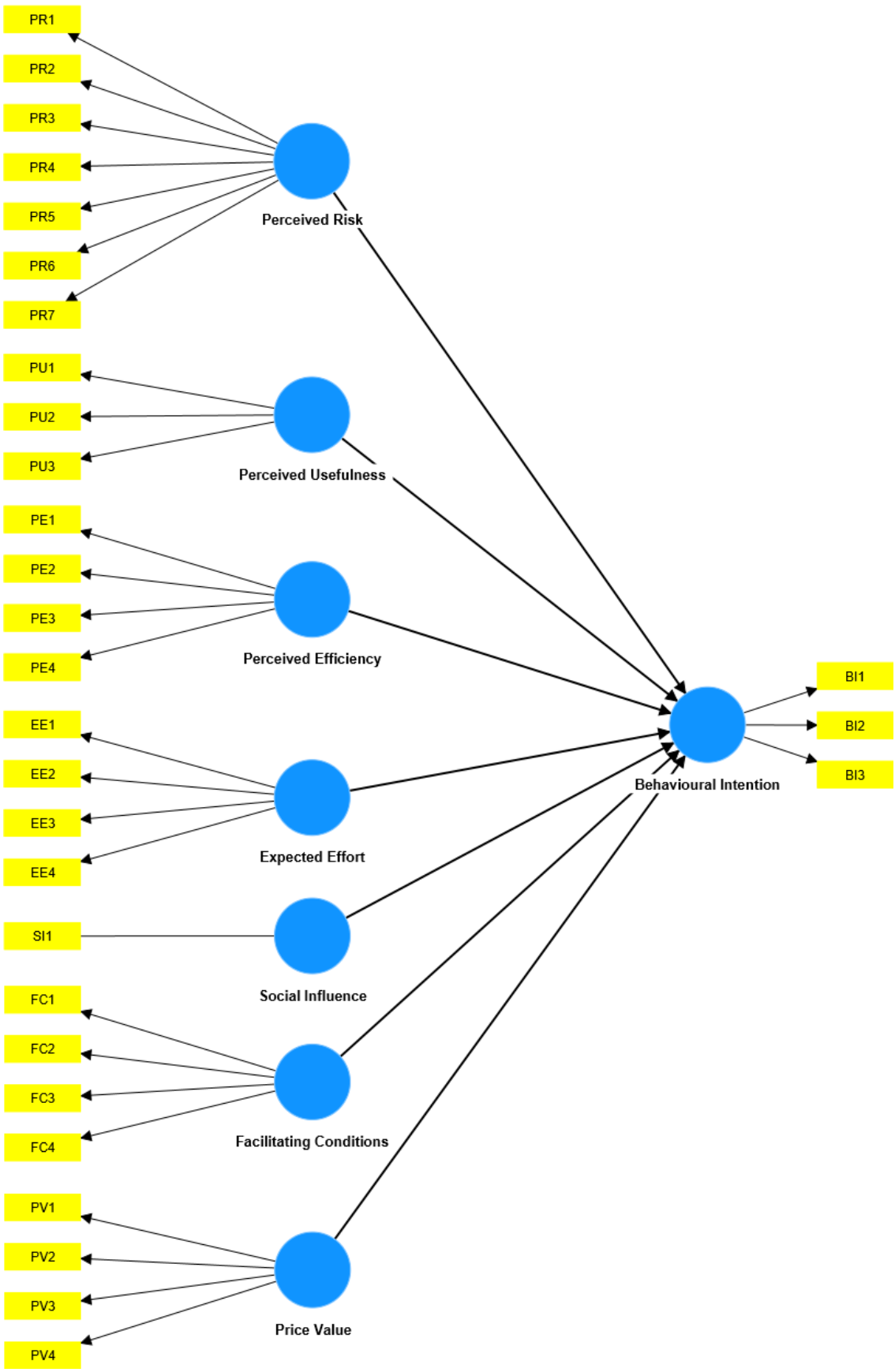
Табела 6 - Демографски подаци испитаника

Варијабла	Вредности	Фреквенција	%
Године	Мање од 20	6	2,23%
	Од 20 до 25	81	30,11%
	Од 25 до 35	77	28,62%
	Више од 35	105	39,03%
Пол	Мушки	120	44,61%
	Женски	149	55,39%
Статус	Ученик	0	0%
	Студент	78	29,00%
	Запослен	169	62,83%
	Незапослен	12	4,46%
	Пензионер	10	3,72%
Образовање	Основно образовање	0	0%
	Средњошколско образовање	75	27,88%
	Виша школа / Факултет	130	48,33%
	Постдипломске студије	64	23,79%

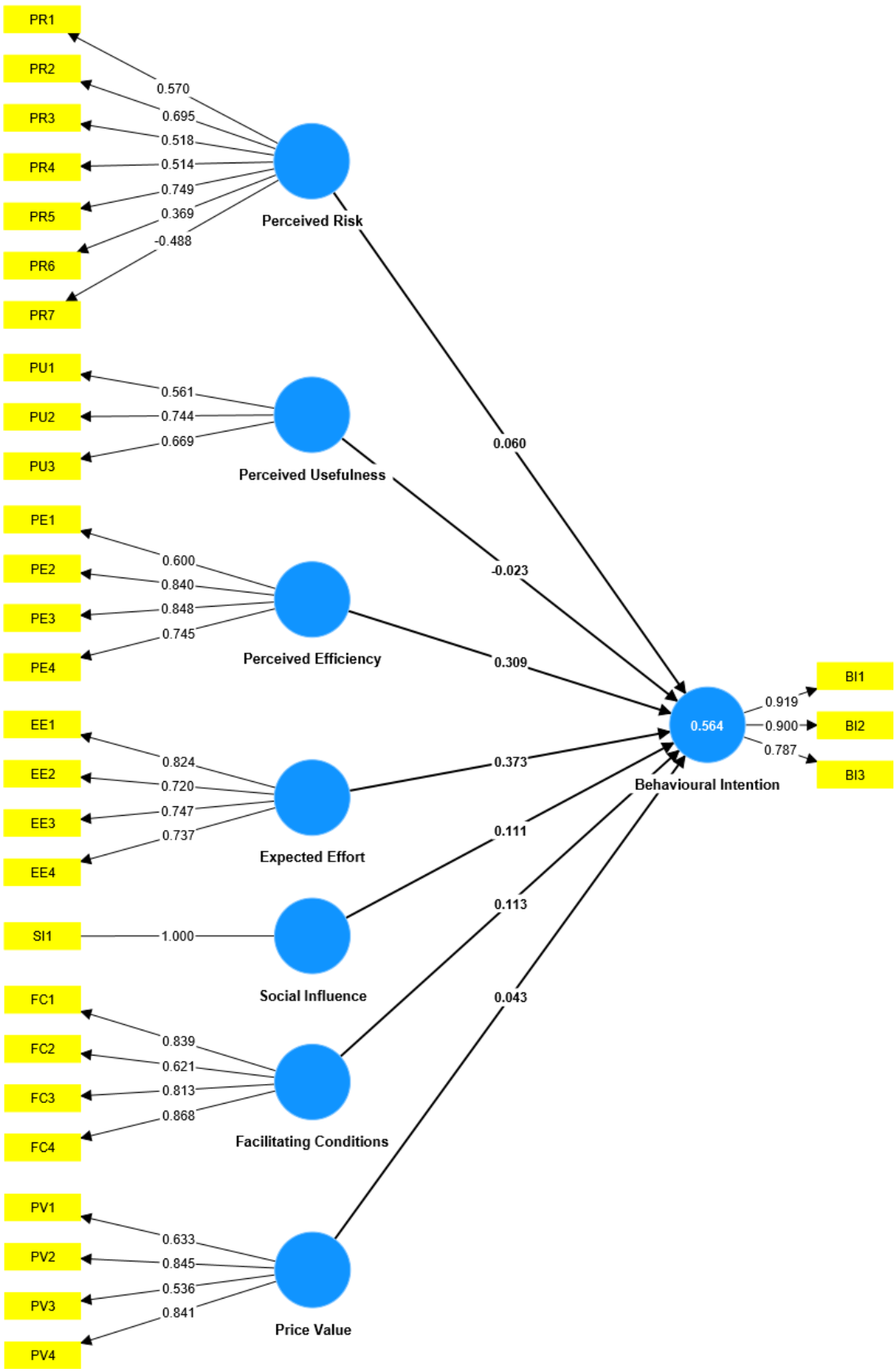
5.2.3. Резултати анализе спремности за коришћење предложеног система за праћење порекла органског меда

Коришћењем методе PLS-SEM вршимо детаљније испитивање узрочно-последичних веза између променљивих које користимо за мерење мотива за коришћење предложеног система за праћење порекла органског меда. Коришћењем PLS-SEM методе развија се комплексни модел чији је задатак да у задовољавајућој мери објасни варијансе разматраних варијабли и идентификује везе између њих (Hair et al., 2011, 2014). Битно је нагласити да PLS-SEM метода не захтева одређену дистрибуцију података (Radenković, 2018).

Анализа је извршена коришћењем SmartPLS 4.0 софтверског алата. Графички приказ структуре модела мерења латентних варијабли дат је на слици 36. На слици 37 приказани су резултати примене PLS алгоритма.



Слика 36 - Структурни модел мерења латентних варијабли



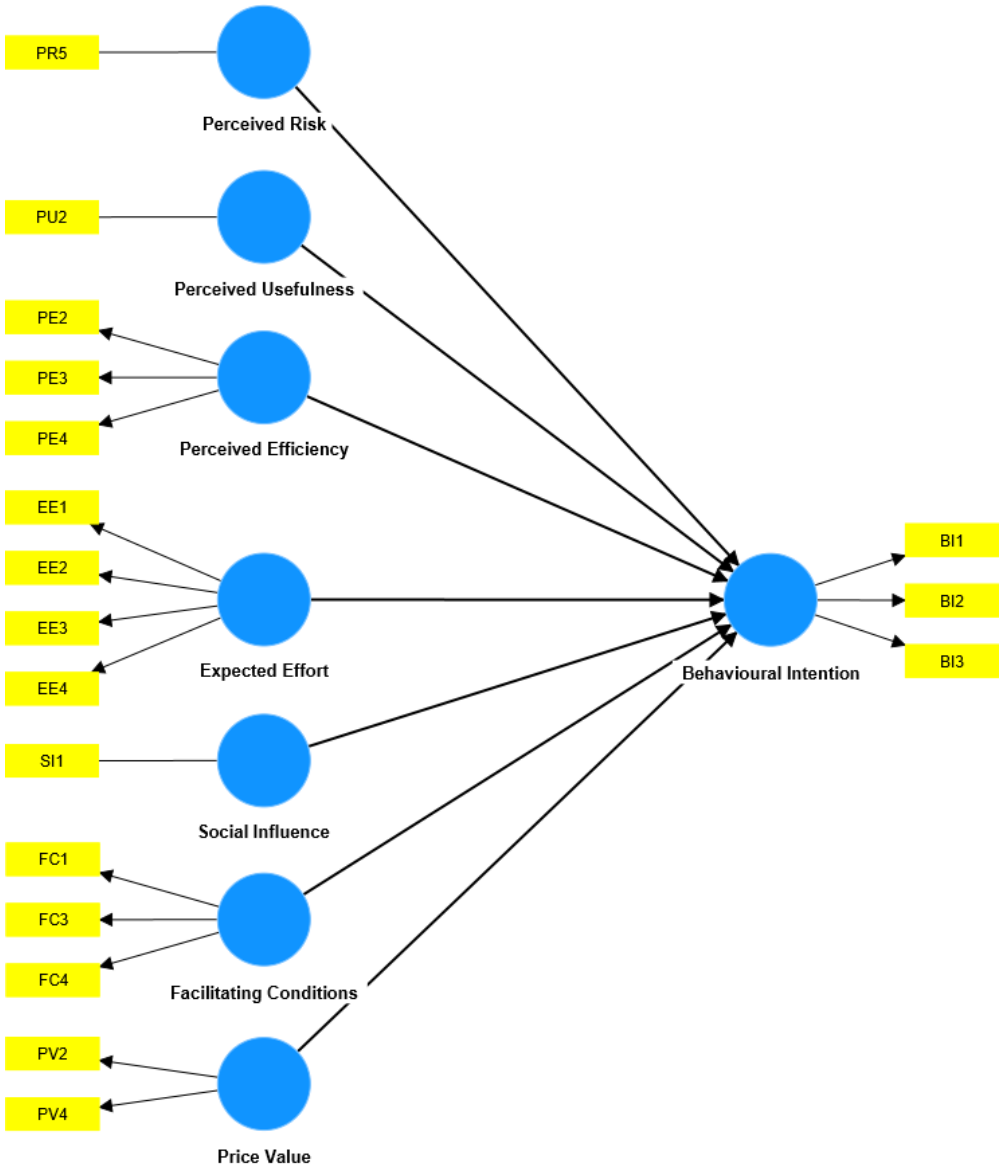
Слика 37 - Резултати примене алгоритма PLS

Табела 7 - Вредности outer loadings индикатора у односу на варијабле

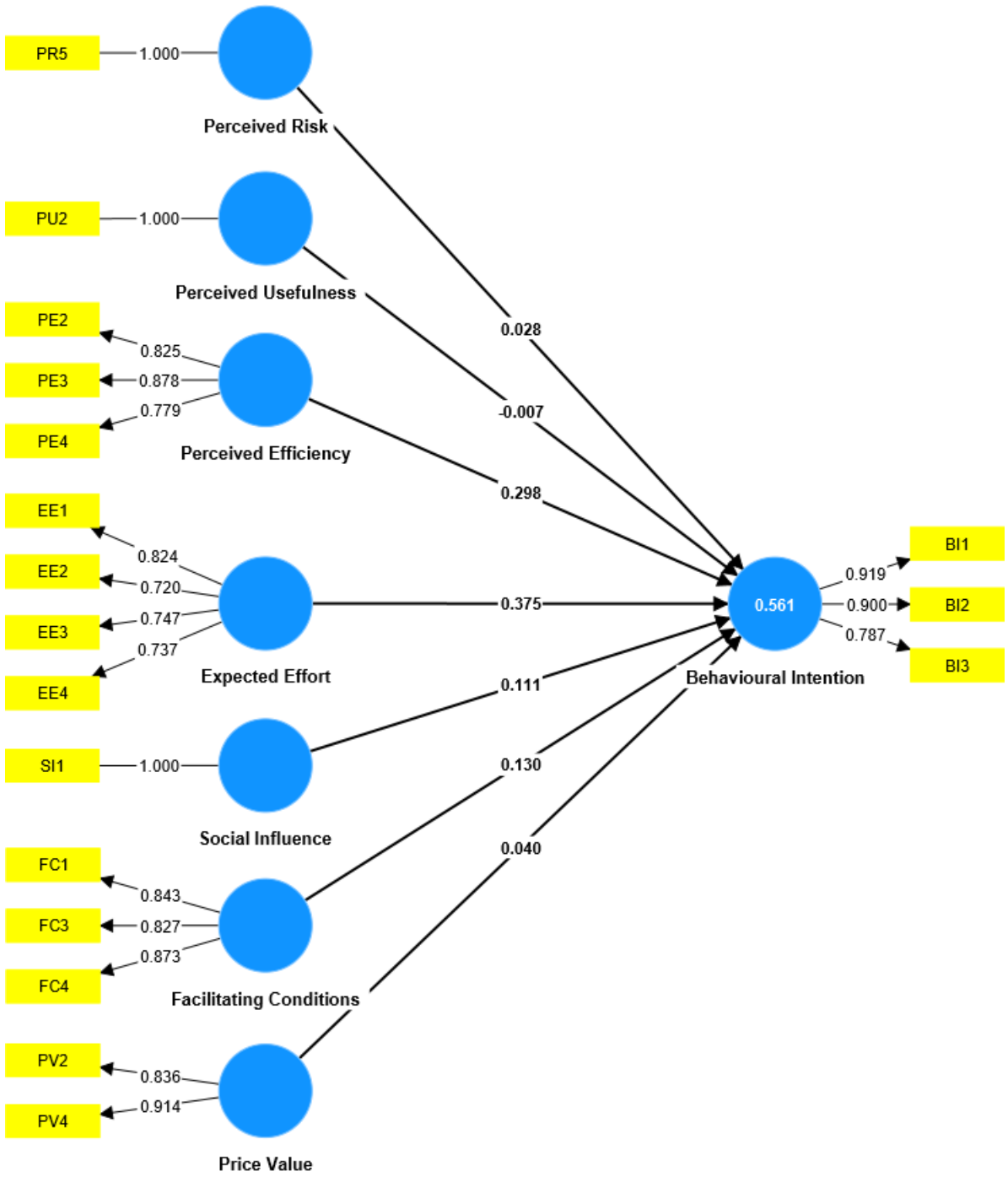
	Beh. Intention	**Exp. Effort**	**Fac. Cond.**	**Per Eff.**	**Per. Risk**	**Per. Use.**	**Price Value**	**Social Infl.**
BI1	**0.919**							
BI2	**0.9**							
BI3	**0.787**							
EE1		**0.824**						
EE2		**0.72**						
EE3		**0.747**						
EE4		**0.737**						
FC1			**0.839**					
FC2			0.621					
FC3			**0.813**					
FC4			**0.868**					
PE1				0.6				
PE2				**0.84**				
PE3				**0.848**				
PE4				**0.745**				
PR1					0.57			
PR2					0.695			
PR3					0.518			
PR4					0.514			
PR5					**0.749**			
PR6					0.369			
PR7					-0.488			
PU1						0.561		
PU2						**0.744**		
PU3						0.669		
PV1							0.633	
PV2							**0.845**	
PV3							0.536	
PV4							**0.841**	
SI1								**1**

Прихватљивим се сматра варијабла која објашњава најмање 50% варијансе одговарајућег индикатора. У случају параметра *loading*, допуштене вредности су изнад квадратног корена вредности варијансе и износи приближно 0,708 (Radenković, 2018). Може се уочити да су изнад 0,708 оне вредности *loading* параметра које одговарају индикаторима BI1, BI2, BI3, EE1, EE2, EE3, EE4, FC1, FC3, FC4, PE2, PE3, PE4, PR5, PU2, PV2, PV4, SI1. Управо те индикаторе ћемо и разматрати у наредном делу истраживања.

На сликама 38 и 39 приказан је структурни модел мерења латентних варијабли са ревидираним индикаторима, а затим и резултати примене PLS SEM методе на новом моделу.



Слика 38 - Структурни модел мерења латентних варијабли са ревидираним индикаторима



Слика 39 - Резултати примене PLS алгоритма на ревидираном моделу

Табела 8 - Оцена валидности мерења латентних варијабли

	Cronbach's alpha	**Composite reliability (rho_a)**	**Composite reliability (rho_c)**	**AVE**
Behavioural Intention	0.839	0.86	0.903	0.758
Expected Effort	0.754	0.762	0.844	0.575
Facilitating Conditions	0.805	0.811	0.885	0.719
Perceived Efficiency	0.771	0.783	0.867	0.686
Price Value	0.703	0.747	0.868	0.767

Параметар AVE (Average Variance Extracted) објашњава у којој мери постоји позитивна корелација између индикатора којима се описује једна варијабла. Препоручена минимална вредност износи 0,5. На основу резултата у табели 9 можемо закључити да су све вредности у одговарајућој колони изнад препоручене (Hair et al., 2014).

За оцену интерне конзистентности важан параметар је композитна поузданост (енг. *composite reliability*), а задовољавајућим се сматрају вредности од 0,70 до 0,95 (Hair et al., 2011, 2014; Sarstedt et al., 2022). Из табеле 8 се може видети да све вредности припадају препорученом интервалу.

Коришћењем *cross loadings* параметра, оцењује се у којој се мери варијабле у моделу међусобно разликују (Hair et al., 2014; Radenković, 2018). Из табеле 9 може се закључити да сваки индикатор најбоље описује варијаблу којој припада, с обзиром да су вредности индикатора за сваку од варијабли веће од вредности која објашњава повезаност са другим варијаблама.

Табела 9 - Оцена валидности дискриминанте - Cross loadings вредности

	Beh. Intention	Exp. Effort	Fac. Cond.	Per. Eff.	Per. Risk	Per. Use.	Price Value	Social Infl.
BI1	**0.919**	0.628	0.431	0.557	0.292	0.244	0.317	0.381
BI2	**0.9**	0.624	0.427	0.593	0.251	0.232	0.338	0.421
BI3	**0.787**	0.508	0.332	0.434	0.139	0.198	0.291	0.239
EE1	0.592	**0.824**	0.341	0.549	0.238	0.195	0.316	0.352
EE2	0.531	**0.72**	0.443	0.501	0.304	0.354	0.257	0.262
EE3	0.458	**0.747**	0.243	0.327	0.241	0.181	0.32	0.452
EE4	0.453	**0.737**	0.262	0.369	0.194	0.314	0.287	0.307
FC1	0.423	0.359	**0.843**	0.352	0.256	0.173	0.329	0.199
FC3	0.345	0.395	**0.827**	0.312	0.318	0.23	0.317	0.22
FC4	0.393	0.348	**0.873**	0.326	0.294	0.223	0.296	0.241
PE2	0.488	0.454	0.317	**0.825**	0.106	0.271	0.289	0.167
PE3	0.567	0.52	0.377	**0.878**	0.12	0.206	0.226	0.262
PE4	0.459	0.485	0.267	**0.779**	0.118	0.121	0.266	0.217
PR5	0.268	0.324	0.339	0.138	**1**	0.092	0.226	0.231
PU2	0.259	0.341	0.244	0.242	0.092	**1**	0.185	0.218
PV2	0.265	0.277	0.287	0.168	0.179	0.085	**0.836**	0.265
PV4	0.359	0.39	0.355	0.353	0.214	0.221	**0.914**	0.241
SI1	0.407	0.449	0.259	0.262	0.231	0.218	0.285	**1**

Поређење AVE вредности са корелацијом варијабли врши се применом *Fornell-Larcker* критеријума валидности модела (Fornell & Larcker, 2018). У табели 10 приказана је матрица корелација на којој се може приметити да се највеће вредности за сваку варијаблу налазе на дијагонали матрице, што сугерише да се валидним сматрају све варијабле.

Табела 9 - Оцена вредности дискриминанте - Fornell-Lackner критеријум

	Beh. Int.	**Exp. Eff.**	**Fac. Cond.**	**Per. Eff.**	**Per. Risk**	**Per. Use.**	**Price Value**	**Social Infl.**
Beh. Intention	**0.871**							
Expected Effort	0.677	**0.758**						
Fac. Conditions	0.459	0.431	**0.848**					
Per. Efficiency	0.612	0.587	0.39	**0.828**				
Perceived Risk	0.268	0.324	0.339	0.138	**1**			
Per. Usefulness	0.259	0.341	0.244	0.242	0.092	**1**		
Price Value	0.363	0.388	0.37	0.311	0.226	0.185	**0.876**	
Social Influence	0.407	0.449	0.259	0.262	0.231	0.218	0.285	**1**

Приликом анализирања структурног модела, коришћењем VIF фактора (*variance inflation factor*) оцењена је колинеарност. Можемо закључити да нема колинеарности варијабли зато што су добијене вредности у табели 11 су испод 5 (Hair et al., 2014; Radenković, 2018).

Табела 101 - VIF вредности

	Behavioural Intention
Expected Effort	2.06
Facilitating Conditions	1.436
Perceived Efficiency	1.615
Perceived Risk	1.214
Perceived Usefulness	1.158
Price Value	1.282
Social Influence	1.293

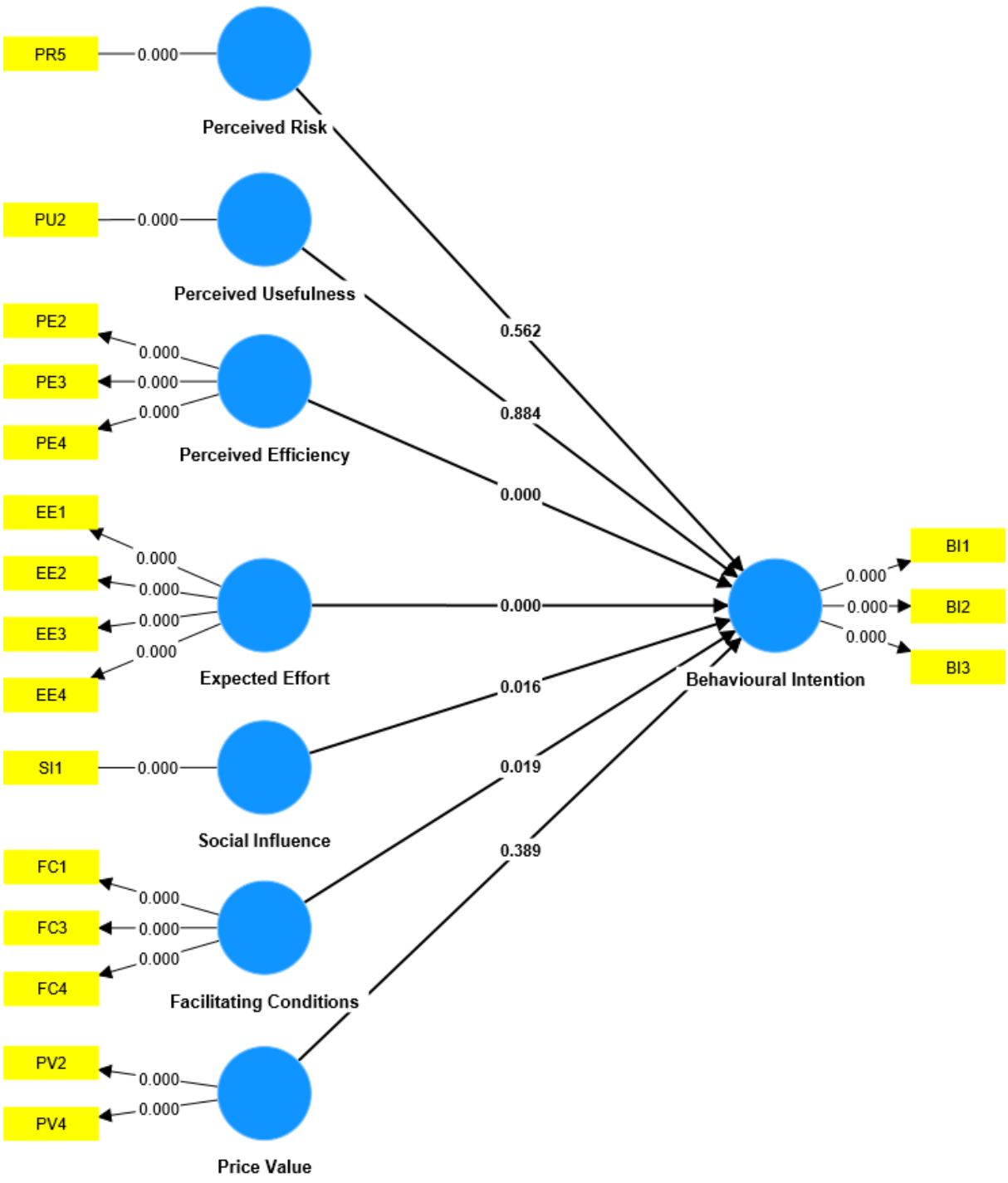
Обрачуната вредност коефицијента одређености (R^2) износи износи 0,561. На тај начин закључујемо да модел има високу предиктивну тачност јер се обрачуната вредност сматра високом у истраживањима која предвиђају понашање корисника (Hair et al., 2014).

Вредности коефицијената путање структурног модела (енг. *path coefficient*) користе се за анализу веза између разматраних варијабли. Јаком позитивном везом сматрају се вредности које су близу +1 и скоро увек су статистички значајне. Вредности близу -1 означавају јаке негативне везе. Вредност близу 0 говори нам да утицај не постоји (Radenković, 2018). Са табеле 12 може се закључити да очекивани труд и перципирана ефикасност имају највећи утицај на спремност за коришћење предложеног система за праћење порекла органског меда. Мањи утицај се види и код олакшавајућих услова и друштвеног утицаја, док остале варијабле готовао да немају утицаја.

Табела 112 - Приказ коефицијената путање

	Path coefficients
Expected Effort -> Behavioural Intention	0.375
Facilitating Conditions -> Behavioural Intention	0.13
Perceived Efficiency -> Behavioural Intention	0.298
Perceived Risk -> Behavioural Intention	0.028
Perceived Usefulness -> Behavioural Intention	-0.007
Price Value -> Behavioural Intention	0.04
Social Influence -> Behavioural Intention	0.111

Како би се извршила детаљнија анализа значајности, искоришћена је метода *bootstrapping*-а са 5.000 узорака и значајности од 5%. Графички приказ се налази на слици 40.



Слика 40 - Резултати bootstrapping-а

Резултати *bootstrapping*-а приказани су у табели 13. Најјачи утицај на Спремност за коришћење има Очекивани труд (β=0,375, T>1,96), а затим Перципирана ефикасност (β=0,298, T>1,96). Мањи утицај се види и код Олакшавајућих услова и Друштвеног утицаја. Утицај Перципираног ризика, Перципиране корисности и Вредности цене на зависну варијаблу није статистички значајан.

Табела 123 - Тестирање хипотеза

	Original sample (O)	**Sample mean (M)**	**Standard deviation (STDEV)**	**T statistics (|O/STDEV|)**	**P values**
Expected Effort -> Behavioural Intention	0.375	0.382	0.064	5.871	0
Facilitating Conditions -> Behavioural Intention	0.13	0.133	0.056	2.35	0.019
Perceived Efficiency -> Behavioural Intention	0.298	0.292	0.065	4.59	0
Perceived Risk -> Behavioural Intention	0.028	0.028	0.047	0.58	0.562
Perceived Usefulness -> Behavioural Intention	-0.007	-0.006	0.048	0.146	0.884
Price Value -> Behavioural Intention	0.04	0.041	0.046	0.862	0.389
Social Influence -> Behavioural Intention	0.111	0.109	0.046	2.414	0.016

Закључак

У овом раду је представљен предлог паметног система за праћење порекла органског меда коришћењем blockchain технологија чији је главни циљ да купцима пружи неопходне инофрмације о производу који купује и обезбеди поверење. Главна јединица праћења система је кошница опремљена различитим сензорима који прикупљају податке о локацији, квалитету ваздуха и осталим параметрима и шаљу их директно на мрежу. На основу њих се прати квалитет произведеног меда, а кључни подаци се прослеђују на blockchain-у како би подаци били транспарентни и непроменљиви. Овакав систем гарантује порекло и карактеристике произведеног меда и производа од меда и крајњем купцу обезбеђује све информације које су потребне за доношење одлуке засноване на чињеницама.

Будући рад на овој теми могао би укључити проширење предложеног система на све заинтересоване стране и сарадњу са нашим кључним партнерима у циљу праћења производа од меда од кошнице до полице. Технике цровдсоурцинга могу се користити приликом тестирања система у спољашњем лабораторијском окружењу како би се проверила евентуална одступања и потенцијална побољшања (Staletić et al., 2020). У циљу бољег управљања комплетним системом, интеграција са симулационим системима такође може бити део будућих истраживања (Despotović-Zrakić et al., 2014; Stefanovic et al., 2009).

Референце

A Blockchain Platform for the Enterprise — hyperledger-fabricdocs main documentation. (2022). https://hyperledger-fabric.readthedocs.io/en/release-2.2/

Abdel-Basset, M., Manogaran, G., & Mohamed, M. (2018). Internet of Things (IoT) and its impact on supply chain: A framework for building smart, secure and efficient systems. *Future Generation Computer Systems*. https://doi.org/10.1016/j.future.2018.04.051

Balaji, M. S., & Roy, S. K. (2016). Value co-creation with Internet of things technology in the retail industry. *Https://Doi.Org/10.1080/0267257X.2016.1217914*, *33*(1–2), 7–31. https://doi.org/10.1080/0267257X.2016.1217914

Balamurugan, S., Ayyasamy, A., & Joseph, K. S. (2021). IoT-Blockchain driven traceability techniques for improved safety measures in food supply chain. *International Journal of Information Technology 2021 14:2*, *14*(2), 1087–1098. https://doi.org/10.1007/S41870-020-00581-Y

Ben-Daya, M., Hassini, E., & Bahroun, Z. (2017). Internet of things and supply chain management: a literature review. *Https://Doi.Org/10.1080/00207543.2017.1402140*, *57*(15–16), 4719–4742. https://doi.org/10.1080/00207543.2017.1402140

Bogdanović, Z., Despotović-Zrakić, M., Labus, A., Lukovac, P., & Radenković, B. (2022). IOT AND BLOCKCHAIN IN MONITORING HONEY PRODUCTION. *II Международная Научно-Практическая Конференция «Развитие Сервисной Деятельности в Условиях Цифровизации Экономики: Актуальные Проблемы и Их Решение» 29 Апреля 2022г.*

Bogdanović, Z., Labus, A., Barać, D., Despotović-Zrakić, M., & Radenković, B. (2021). Blockchain Technologies In Tracking Organic Honey Production. *International Scientific Conference Sustainable Development of Rural Areas*.

Bogdanović, Z., Radenković, B., Despotović-Zrakić, M., Barać, D., Labus, A., & Naumović, T. (2019). BLOCKCHAIN TECHNOLOGIES: CURRENT STATE AND PERSPECTIVES. In D. Bećirović, A. Čavalić, A. Zahirović Suhonjić, & N. Halilagić (Eds.), *Second International Scientific Conference on Digital Economy DIEC 2019* (pp. 1–12). Internacionalna poslovno - informaciona akademija, ISSN 2566 - 4514. http://ipi-akademija.ba/file/zbornik-diec-2019/44

Borah, M. D., Naik, V. B., Patgiri, R., Bhargav, A., Phukan, B., & Basani, S. G. M. (2020). *Supply Chain Management in Agriculture Using Blockchain and IoT*. 227–242. https://doi.org/10.1007/978-981-13-8775-3_11

Buterin, V. (2013). *ETHEREUM WHITE PAPER: A NEXT GENERATION SMART CONTRACT & DECENTRALIZED APPLICATION PLATFORM*.

de Vass, T., Shee, H., & Miah, S. (2018). The effect of "Internet of Things" on supply chain integration and performance: An organisational capability perspective. *Australasian Journal of Information Systems*, *22*. https://doi.org/10.3127/AJIS.V22I0.1734

Despotović-Zrakić, M., Barać, D., Bogdanović, Z., Jovanić, B., & Radenković, B. (2014). Software Environment for Learning Continuous System Simulation. *Acta Polytechnica Hungarica*, *11*(2), 187–202.

Diaz, M., Martin, C., Rubio, B. (2016). *chalenge in integration IOt and CC 2016.pdf* (pp. 99–117). http://www.adilmedya.com/yakin-gelecekte-tum-nesneler-birbirleriyle-iletisim-kuracak-nesnelerin-interneti-internet-of-things/

Ehsan, I., Irfan Khalid, M., Ricci, L., Iqbal, J., Alabrah, A., Sajid Ullah, S., & Alfakih, T. M. (2022). A Conceptual Model for Blockchain-Based Agriculture Food Supply Chain System. *Scientific Programming*, *2022*. https://doi.org/10.1155/2022/7358354

Fornell, C., & Larcker, D. F. (2018). Evaluating Structural Equation Models with Unobservable Variables and Measurement Error: *Https://Doi.Org/10.1177/002224378101800104*, *18*(1), 39–50. https://doi.org/10.1177/002224378101800104

Guidelines - Algorand Developer Portal. (2022). https://developer.algorand.org/docs/get-details/dapps/avm/teal/guidelines/

Hair, J. F., Hult, G. T. M., Ringle, C. M., & Sarstedt, M. (2014). A Primer on Partial Least Squares Structural Equation Modeling (PLS-SEM). Sage Publications. *European Journal of Tourism Research*, *6*(2), 211–213.

Hair, J. F., Sarstedt, M., Ringle, C. M., & Mena, J. A. (2011). An assessment of the use of partial least squares structural equation modeling in marketing research. *Journal of the Academy of Marketing Science 2011 40:3*, *40*(3), 414–433.

https://doi.org/10.1007/S11747-011-0261-6

Hamdaqa, M., Met, L. A. P., & Qasse, I. (2022). iContractML 2.0: A domain-specific language for modeling and deploying smart contracts onto multiple blockchain platforms. *Information and Software Technology*, *144*, 106762. https://doi.org/10.1016/J.INFSOF.2021.106762

Hewa, T., Ylianttila, M., & Liyanage, M. (2021). Survey on blockchain based smart contracts: Applications, opportunities and challenges. *Journal of Network and Computer Applications*, *177*, 102857. https://doi.org/10.1016/J.JNCA.2020.102857

Koedel, C., Mihaly, K., & Rockoff, J. E. (2015). Value-added modeling: A review. *Economics of Education Review*, *47*, 180–195. https://doi.org/10.1016/J.ECONEDUREV.2015.01.006

Lukovac, P., Miletić, A., & Radenković, B. (2022). DESIGNING A BLOCKCHAIN-BASED E-BUSINESS SYSTEM FOR TRACKING HONEY PRODUCTION. *INTERNATIONAL Symposium Sustainable Business Management and Digital Transformation: Challenges and Opportunities in the Post-COVID Era*.

Marangunić, N., & Granić, A. (2014). Technology acceptance model: a literature review from 1986 to 2013. *Universal Access in the Information Society 2014 14:1*, *14*(1), 81–95. https://doi.org/10.1007/S10209-014-0348-1

Micali, S. (2019, April 4). *ALGORAND'S CORE TECHNOLOGY (in a nutshell)*. Medium. https://medium.com/algorand/algorands-core-technology-in-a-nutshell-e2b824e03c77

Osterwalder, A., & Pigneur, Y. (2010). *Business Model Generation: A handbook for visionaries, game changers and challengers*.

Pranto, T. H., Noman, A. A., Mahmud, A., & Haque, A. B. (2021). Blockchain and smart contract for IoT enabled smart agriculture. *PeerJ Computer Science*, *7*, 1–29. https://doi.org/10.7717/PEERJ-CS.407/SUPP-1

Radenković, M. (2018). *An e-business model for the participation of households in the Serbian electricity market based on Smart Grid technologies*. University of Belgrade.

Ranđić, S., Vujičić, D., & Jagodić, D. (2018). Blockchain technology, bitcoin, and Ethereum: A brief overview. *17th International Symposium Infoteh-Jahorina (Infoteh)*, 1–6.

https://doi.org/10.1109/INFOTEH.2018.8345547

Sarstedt, M., Ringle, C. M., & Hair, J. F. (2022). Partial Least Squares Structural Equation Modeling. *Handbook of Market Research*, 587–632. https://doi.org/10.1007/978-3-319-57413-4_15

Savez pčelarskih organizacija Srbije. (n.d.). Retrieved September 8, 2022, from https://spos.info/

Seth, S. (2022, May 20). *All About NEM (XEM), the Harvested Cryptocurrency*. https://www.investopedia.com/tech/meet-nem-xem-harvested-cryptocurrency/

Smart contract details - Algorand Developer Portal. (n.d.). Retrieved September 1, 2022, from https://developer.algorand.org/docs/get-details/dapps/smart-contracts/apps/

Srivastava, A., & Dashora, K. (2022). Application of blockchain technology for agrifood supply chain management: a systematic literature review on benefits and challenges. *Benchmarking*, *ahead-of-print*(ahead-of-print). https://doi.org/10.1108/BIJ-08-2021-0495/FULL/XML

Staletić, N., Labus, A., Bogdanović, Z., Despotović-Zrakić, M., & Radenković, B. (2020). Citizens' readiness to crowdsource smart city services: A developing country perspective. *Cities*, *107*, 102883. https://doi.org/10.1016/J.CITIES.2020.102883

Stefanovic, D., Stefanovic, N., & Radenkovic, B. (2009). Supply network modelling and simulation methodology. *Simulation Modelling Practice and Theory*, *17*(4), 743–766. https://doi.org/10.1016/J.SIMPAT.2009.01.001

Tian, F. (2017). A supply chain traceability system for food safety based on HACCP, blockchain & Internet of things. *14th International Conference on Services Systems and Services Management, ICSSSM 2017 - Proceedings*. https://doi.org/10.1109/ICSSSM.2017.7996119

Traversy, B. (2022, February). *bradtraversy/mern-tutorial*. https://github.com/bradtraversy/mern-tutorial/blob/main/backend/controllers/userController.js

van Hilten, M., Ongena, G., & Ravesteijn, P. (2020). Blockchain for Organic Food Traceability: Case Studies on Drivers and Challenges. *Frontiers in Blockchain*,

3(September), 1–13. https://doi.org/10.3389/fbloc.2020.567175

Vemuri, V. K. (2019). Blockchain: a practical guide to developing business, law, and technology solutions. *Https://Doi.Org/10.1080/15228053.2019.1588546*, *20*(3–4), 161–163. https://doi.org/10.1080/15228053.2019.1588546

Venkatesh, V., Thong, J. Y. L., & Xu, X. (2012). Consumer acceptance and use of information technology: Extending the unified theory of acceptance and use of technology. *MIS Quarterly: Management Information Systems*, *36*(1), 157–178. https://doi.org/10.2307/41410412

Why Algorand? - Algorand Developer Portal. (2022). https://developer.algorand.org/docs/get-started/basics/why_algorand/

Wood, G. (2014). Ethereum: a secure decentralised generalised transaction ledger. *Ethereum Project Yellow Paper*, 1–32.

Xu, L. Da, He, W., & Li, S. (2014). Internet of things in industries: A survey. *IEEE Transactions on Industrial Informatics*, *10*(4), 2233–2243. https://doi.org/10.1109/TII.2014.2300753

Zheng, Z., Xie, S., Dai, H. N., Chen, W., Chen, X., Weng, J., & Imran, M. (2020). An overview on smart contracts: Challenges, advances and platforms. *Future Generation Computer Systems*, *105*, 475–491. https://doi.org/10.1016/j.future.2019.12.019

"Icons used in illustrations made by Freepik from www.flaticon.com"