

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ ЕКОНОМІЧНИЙ УНІВЕРСИТЕТ
ІМЕНІ ВАДИМА ГЕТЬМАНА
навчально-науковий інститут
«Інститут інформаційних технологій в економіці»
кафедра інформаційних систем в економіці**

галузь знань	12 Інформаційні технології
спеціальність	122 Комп'ютерні науки
освітня програма	Системи штучного інтелекту

форма навчання: денна

КВАЛІФІКАЦІЙНА МАГІСТЕРСЬКА РОБОТА

на тему:

**ІНТЕЛЕКТУАЛЬНА СИСТЕМА ПРИЙНЯТТЯ РІШЕНЬ В СФЕРІ
ЗЕРНОТРЕЙДИНГУ**

здобувача Калашнікова Кирила Станіславовича _____

Науковий керівник: к.е.н., доцент

_____ Лозовик Ю.М.

**Робота допущена до захисту перед
екзаменаційною комісією з атестації
здобувачів вищої освіти (ЕК)**

в.о. завідувача кафедри: к.е.н., доцент

_____ Тішков Б.О.

Київ 2023

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДЕРЖАВНИЙ ВИЩИЙ НАВЧАЛЬНИЙ ЗАКЛАД
“КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ ЕКОНОМІЧНИЙ УНІВЕРСИТЕТ
імені Вадима Гетьмана”
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ
«ІНСТИТУТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ В ЕКОНОМІЦІ»

Кафедра інформаційних систем в економіці

Затверджую:
Завідувач кафедри _____
“___” _____ 20__ р.

ІНДИВІДУАЛЬНЕ ЗАВДАННЯ
на кваліфікаційну магістерську роботу

на тему:

Інтелектуальна система прийняття рішень в сфері зернотрейдингу

Студенту VI курсу спеціалізації «Системи штучного інтелекту»,

Калашнікову Кирилу Станіславовичу

(П.І.Б. студента)

Тему кваліфікаційної магістерської роботи затверджено наказом ректора від “20” 02 2023 р.

№ 333-ст

Кваліфікаційна магістерська робота виконується на матеріалах

Орієнтовний план кваліфікаційної магістерської роботи і терміни подання розділів науковому керівникові⁴:

Розділ I Теоретичний розділ. Дослідження та аналіз підходів до створення

(назва — термін подання (денна, заочна форми – до 04.04)

предметної області експертної системи

Розділ II Аналітичний розділ. Характеристика експертної системи та

(назва — термін подання (денна, заочна форми – до 25.04)

постановка задачі

Розділ III Конструктивний розділ. Розроблення проектних рішень

(назва — термін подання (денна, заочна форми – до 10.05)

Термін подання завершеної роботи науковому керівникові для оформлення відгуку

(денна, заочна форми – до 30.05)

Предмет дослідження

⁴ Порушення студентом термінів подання заяви на затвердження теми кваліфікаційної магістерської роботи, погодження з керівником індивідуального завдання, несвоєчасне завершення розділів та роботи в цілому є підставою для його відрахування з університету як тако- го, що не виконує навчальний план

Предметом дослідження є основні характеристики діяльності відбуваються в зернотрейдингового підприємства та властивості інформаційних процесів, що середині даного суб'єкта.

Мета кваліфікаційної магістерської роботи

Метою кваліфікаційної магістерської роботи було розроблення експертної системи для зернотрейдингової компанії з метою автоматизації процесів прийняття рішень при виборі культури для вирощення.

Конкретні завдання, які студент повинен виконати для досягнення поставленої мети:

У розділі I Провести аналіз задачі та навести перелік існуючих програмних засобів для підтримки прийняття рішень та автоматизації процесів

У розділі II Розробити модель, необхідну для прийняття управлінських рішень, а також обґрунтувати вибір відповідної структури експертної системи

У розділі III Описати конструктивну частину, програмне та технічне забезпечення для системи, продемонструвати прототип

Завдання підготував

науковий керівник

_____ (підпис)

_____ (прізвище, ім'я, по-батькові)

“ _____ ” _____ 20__ р.

Завдання одержав

студент

_____ (підпис)

_____ (прізвище, ім'я, по-батькові)

“ _____ ” _____ 20__ р.

Примітка: Перша сторінка індивідуального завдання на кваліфікаційну магістерську роботу заповнюється студентом під керівництвом наукового керівника, друга — науковим керівником.

Відгук

про кваліфікаційну магістерську роботу
здобувача навчально-наукового інституту
«Інститут інформаційних технологій в економіці»
освітньо-наукової програми
«Системи штучного інтелекту»
Калашнікова Кирила Станіславовича
підготовлену на тему

«Інтелектуальна система прийняття рішень в сфері зернотрейдингу»

1. Актуальність дипломної роботи полягає у потребі моніторингової системи постійного відслідковування змін у функції попиту на сільськогосподарські культури для зернотрейдера.

2. Позитивні риси кваліфікаційної магістерської роботи:

Автором проведено ґрунтовне теоретико-практичне дослідження та аналіз підходів для створення інформаційної системи з підтримкою прийняття аналітичних рішень для моніторингу попиту на зернові культури, вивчено методи та моделі, проаналізовані аналоги програмних рішень та здійснено проектування та розробку власної системи для підтримки ухвалення рішень. Для цього був застосований принцип нормалізації – приведення всіх даних до єдиного вигляду. У роботі використовується сучасний стек технологій для проектування та розробки ПЗ.

3. Наявність самостійних розробок автора

Робота виконана самостійно. У роботі чітко розмежовано текст автора та запозичення, наявні посилання на використані джерела. Автором, Калашніковим Кирилом Станіславовичом, запропоновано і реалізовано експертну систему для визначення попиту на сільськогосподарську продукцію.

4. Цінність теоретичних висновків та практичних рекомендацій:

Автором опрацьований великий масив інформації, систематизовано і узагальнено інформацію щодо розробки ІС для підтримки прийняття рішень для зернотрейдерів.

Розроблена система з успіхом використовується на підприємстві, а алгоритм програми – універсальний, що забезпечує можливість використання даного продукту іншими аграрними підприємствами. Формат даних – масиви JSON, в яких відображається інформація щодо великої кількості факторів, таких як: ціни, кліматичні умови, застосування техніки, забезпеченість трудовими ресурсами, рівень фінансування тощо.

5. Наявність недоліків:

- диспропорції у викладенні матеріалу;
- слабкий блок щодо проектування системи та використання UML-діаграм;
- слабка обґрунтованість факторів для підтримки ухвалення рішень;
- необхідність розробки моніторингової системи;
- необхідність підтримки вхідних даних та роботи експерта.

6. Загальна оцінка кваліфікаційної магістерської роботи та її допущення до захисту перед ЕК:

Зазначене вище дозволяє констатувати, що дипломна робота Калашнікова К.С. являє собою самостійне завершене дослідження. Автор роботи вміє працювати з сучасними технологіями проектування та програмування програмних продуктів, такими як JS, NodeJs, PostgreSQL, MongoDB, Docker, EC2 AWS для підтримки роботи зернотрейдерів.

Дипломна робота Калашнікова К.С. відповідає вимогам щодо написання та оформлення магістерських робіт і може бути рекомендована до захисту в ДЕК.

Робота може бути оцінена на добре (90б).

Науковий керівник,
доцент кафедри інформаційних систем
в економіці, к. е. н.

Лозовик Ю.М.

(підпис)

(прізвище, ініціали)

“15” травня 2023 р.

Рецензія
на кваліфікаційну магістерську роботу
здобувача вищої освіти
Калашнікова Кирила Станіславовича
(прізвище, ім'я, по батькові)

Тема Інтелектуальна система прийняття рішень в сфері зернотрейдингу

Актуальність теми кваліфікаційної магістерської роботи і доцільність її дослідження Актуальність роботи полягає в необхідності впровадження ефективних технологій експертних систем у зернотрейдинговій галузі для підвищення ефективності прийняття рішень та оптимізації процесів.

Якість проведеного дослідження Дослідження, проведені в рамках роботи, відрізняються достатньою якістю, охоплюють широкий спектр аспектів зернотрейдингової галузі та підтверджуються об'єктивними результатами та аналізом даних.

Позитивні риси кваліфікаційної магістерської роботи Позитивними рисами цієї кваліфікаційної магістерської роботи є глибокий аналіз проблеми, використання сучасних методів та технологій, сучасний підхід до розробки системи та реалізація практичних рішень для зернотрейдингової компанії.

Зауваження Зауваження до магістерської роботи включають необхідність більш детального обґрунтування вибору деяких методів та алгоритмів, а також розширення обсягу дослідження для забезпечення більш повної оцінки ефективності експертної системи.

Практична значимість висновків і рекомендацій Практична значимість висновків і рекомендацій полягає в їх потенціалі впливу на покращення процесів прийняття рішень в зернотрейдинговій компанії, забезпеченні оптимального вибору урожаю та підвищенні її ефективності.

Місце роботи та посада рецензента
Науковий ступінь, учене звання (за наявності)

Підпис засвідчую:




(підпис, ПІБ)

(посада, підпис)

АНОТАЦІЯ

кваліфікаційної магістерської роботи

студента 6 курсу

Калашнікова Кирила Станіславовича

виконану на тему:

«Інтелектуальна система прийняття рішень в сфері зернотрейдингу»

Кваліфікаційна магістерська робота на тему "Інтелектуальна система прийняття рішень в сфері зернотрейдингу" є актуальною у зв'язку зі зростаючою конкуренцією та складністю управління в цій галузі. Робота складається з трьох розділів, які пов'язані між собою.

У першому розділі проведено аналіз задачі вибору культури для посіву та наведено перелік існуючих програмних засобів для підтримки прийняття рішень та автоматизації процесів в даній сфері.

У другому розділі розглянуто проектну частину, розроблено модель, необхідну для прийняття управлінських рішень, а також обґрунтовано вибір відповідної структури експертної системи.

У третьому розділі наведено конструктивну частину, програмне та технічне забезпечення для системи підтримки прийняття рішень на зернотрейдинговому підприємстві.

Висновки містять рекомендації щодо доцільності розробки та впровадження системи прийняття рішень в галузі зернотрейдингу. Робота є важливим внеском у розвиток інтелектуальних систем прийняття рішень в цій галузі та може бути використана для подальших досліджень та розробок в цьому напрямку.

РЕФЕРАТ

Кваліфікаційна магістерська робота містить 88 сторінок, 3 таблиці, 46 рисунків, список літератури з 42 найменувань, 2 додатки.

«Інтелектуальна система прийняття рішень в сфері зернотрейдингу»

Предметом дослідження є основні характеристики діяльності зернотрейдингового підприємства та властивості інформаційних процесів, що відбуваються в середині даного суб'єкта.

Об'єктом дослідження виступають інформаційні процеси, на яких ґрунтується робота зернотрейдингового підприємства.

Мета магістерського дипломного проекту полягає в дослідженні агротрейдингової сфери діяльності, проектуванні та розробці експертної системи для розв'язання задачі вибору оптимальної культури для вирощення.

Завданнями магістерського дипломного проекту є розробка експертної системи зернотрейдингової компанії з вибору оптимальної культури для вирощення.

Апаратура, використана при дослідженні – ноутбук Apple, монітор HP.

Результатами, досягнутими під час магістерського проектування є повне дослідження процесів підприємства, що займається зернотрейдингом, ринку зерна, проектування та розробка прототипу експертної системи.

Одержані результати можуть бути використані на діючому зернотрейдинговому підприємстві

ЗМІСТ

ВСТУП	2
1 ТЕОРЕТИЧНИЙ РОЗДІЛ. ДОСЛІДЖЕННЯ ТА АНАЛІЗ ПІДХОДІВ ДО СТВОРЕННЯ ПРЕДМЕТНОЇ ОБЛАСТІ ЕКСПЕРТНОЇ СИСТЕМИ.....	3
1.1 Дослідження предметної області. Збір інформації та вивчення матеріалів з теми кваліфікаційної магістерської роботи	3
1.2 Аналіз існуючих СШ й інтелектуальних систем предметної області	11
1.3 Постановка проблеми та формування задач.....	22
1.4 Обґрунтування вибору підходів і технологій для проектування та створення експертної системи та її компонентів	23
2 АНАЛІТИЧНИЙ РОЗДІЛ. ХАРАКТЕРИСТИКА ЕКСПЕРТНОЇ СИСТЕМИ ТА ПОСТАНОВКА ЗАДАЧІ	34
2.1 Характеристика об'єкта дослідження	34
2.2 Структура і характеристика експертної системи	42
2.3 Методи, моделі й моделювання процесів і елементів складних систем....	51
3 КОНСТРУКТИВНИЙ РОЗДІЛ. РОЗРОБЛЕННЯ ПРОЕКТНИХ РІШЕНЬ	62
3.1 Моделювання та проектування бази знань для системи прийняття інтелектуальних рішень.....	62
3.2 Розроблення користувацького інтерфейсу. Елементи та структура	70
3.3 Проектування забезпечувальних підсистем експертної системи.	77
3.3.1. Інформаційне забезпечення	77
3.3.2. Програмне забезпечення	79
3.3.3. Технічне забезпечення	82
3.3.4. Організаційно-економічне забезпечення	87
ВИСНОВКИ.....	89
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	91
ДОДАТКИ.....	ОШИБКА! ЗАКЛАДКА НЕ ОПРЕДЕЛЕНА.
Додаток А Фрагменти JSON-об'єктів бази знань	Ошибка! Закладка не определена.
Додаток Б Фрагменти програмного коду інференційного рушія.....	Ошибка! Закладка не определена.

ВСТУП

У сучасному світі зернові культури відіграють ключову роль у глобальній торгівлі та є важливим продуктом для бізнесу. Ефективне управління торгівлею зерном може забезпечити успіх компанії, але прийняття правильних рішень в цій галузі потребує великої кількості інформації та навичок швидкої аналітики та обробки даних.

Це може бути досягнуто за допомогою експертної системи прийняття рішень. Така система буде здатна аналізувати великі обсяги даних, робити прогнози та давати рекомендації щодо прийняття оптимальних рішень. Система може допомогти працівникам галузі швидше та ефективніше знаходити оптимальні рішення в умовах невизначеності та складності.

Метою даного дипломного проекту є розробка експертної системи прийняття рішень в сфері зернотрейдингу, що допоможе обрати оптимальну культуру для вирощення на наявних сільськогосподарських угіддях. Для досягнення цієї мети будуть використані сучасні методи та інструменти розробки експертних систем.

У даному дипломному проекті будуть розглянуті такі питання, як аналіз задачі що вирішується, дослідження предметної області, вибір методів та засобів розробки системи, проектування системи.

Отже, цей дипломний проект має великий потенціал для покращення управління зернотрейдингом та може становити вагомий внесок в сферу зернової галузі.

1 ТЕОРЕТИЧНИЙ РОЗДІЛ. ДОСЛІДЖЕННЯ ТА АНАЛІЗ ПІДХОДІВ ДО СТВОРЕННЯ ПРЕДМЕТНОЇ ОБЛАСТІ ЕКСПЕРТНОЇ СИСТЕМИ

В першому теоретичному розділі дипломного проекту буде проведений аналіз підходів до визначення предметної області інтелектуальної системи прийняття рішень в сфері зернотрейдингу.

Для розуміння сутності експертної системи необхідно детально проаналізувати основні поняття, пов'язані з її розробкою та функціонуванням. Також в цьому розділі буде розглянуто сучасні методи, які використовуються при розробці інтелектуальних систем прийняття рішень.

Проведення аналізу дозволить виявити можливі напрямки для подальшого вдосконалення та розробки нових інтелектуальних систем прийняття рішень.

Окрім того, у цьому розділі будуть проаналізовані найбільш популярні сфери застосування штучного інтелекту в зернотрейдингу та сільському господарстві в цілому.

У підсумку, теоретичний розділ дипломного проекту дозволить розкрити сутність сучасних інтелектуальних систем прийняття рішень в сфері зернотрейдингу, описати основні підходи та шляхи їх вдосконалення. Також будуть проаналізовані існуючі підходи до моделювання процесів управління зернотрейдингом.

1.1 Дослідження предметної області. Збір інформації та вивчення матеріалів з теми кваліфікаційної магістерської роботи

Для проектування сучасних експертних систем чи систем підтримки прийняття рішень необхідно детально вивчити предметну область зернотрейдингу та тісно пов'язану сферу сільського господарства, що дозволить розробити власну методологію та алгоритми.

Збір інформації передбачає вивчення матеріалів, що стосуються торгівлі зерном, аналіз відомих проблем в галузі та вивчення підходів до їх вирішення.

Крім того, необхідно дослідити методи прийняття рішень та моделі, що застосовуються в сфері зернотрейдингу, та проаналізувати їхні переваги та недоліки.

Сфера зернотрейдингу пов'язана з вирощуванням, переробкою та продажем зерна, бобових культур, заморожених фруктів та овочів. Це велика промисловість, яка забезпечує харчову безпеку в багатьох країнах світу. Зернотрейдинг містить в собі всі етапи виробництва та продажу зерна, включаючи його вирощування, збирання, транспортування, зберігання, переробку та розподіл.

Зернотрейдинг та сільське господарство - це важлива галузь, яка забезпечує людство продовольством та розвиває економіку країн.

Ця галузь має значний вплив на економіку, оскільки їжа - це необхідний елемент життя для людей, і більшість країн залежать від сільського господарства для забезпечення своєї національної безпеки. Зернотрейдинг є важливим чинником у глобальному харчовому ланцюжку, оскільки великі країни експортери, такі як США та Канада, є ключовими гравцями у цій галузі. У той же час, імпортери залежать від зернових культур, щоб забезпечити своє населення продуктами харчування.

Зернотрейдинг та сільське господарство також мають значний вплив на навколишнє середовище. Неправильне використання хімічних речовин та неефективне використання ресурсів може призвести до забруднення довкілля та зниження родючості ґрунту. Тому важливо мати на увазі не лише прибуток, а й екологічні наслідки при вирощуванні та торгівлі зерновими культурами.

Зернотрейдинг та сільське господарство є складними галузями, які вимагають глибоких знань та досвіду.

Основні групи зернових культур, які вирощуються для комерційного використання, включають пшеницю, кукурудзу, сою, ячмінь, рис, овес, сорго. Ці

культури використовуються як основне джерело продуктів харчування та комерційних продуктів, таких як корм для тварин, біопаливо.

В зернотрейдингу головними гравцями є фермери, посередники, оптові продавці, біржі зерна, транспортні компанії та інші учасники ланцюжка поставок зерна такі як брокери, сюрвеєри, фахівці з фумігації. Вони взаємодіють між собою з метою максимізації свого прибутку та забезпечення потреб споживачів.

У цій сфері використовуються різні інструменти та технології, такі як аналіз ринку, прогнозування погодних умов, моніторинг цін та попиту на зерно, управління логістикою та складськими запасами, аналіз фінансових показників та ризиків. Для успішної діяльності важливо мати доступ до якісної та достовірної інформації та вміти приймати правильні рішення в умовах невизначеності та змінності ринку.

Для ефективного управління в зернотрейдингу необхідні інструменти прийняття рішень, які базуються на аналізі різноманітної інформації. Однак, зі зростанням обсягу даних, які зібрані в цій галузі, виникає необхідність у використанні експертних систем прийняття рішень для забезпечення точних та оперативних рішень на основі аналізу великих обсягів інформації.

Поточна ситуація на світовому зерновому ринку є досить складною та нестабільною. З одного боку, світові запаси зерна в цілому є досить великими, що повинно б давати певну стабільність цінам. З іншого боку, існують фактори, які впливають на складнопередбачувану динаміку цін на зерно, такі як погода, політична ситуація в країнах-експортерах, торговельні війни, епідемії.

Одним з найбільш важливих гравців на світовому зерновому ринку є США, які є найбільшим експортером зерна у світі. Також великі обсяги зерна виробляються в країнах, таких як Китай, Бразилія, Аргентина, Україна.

Глобально, на сьогоднішній день спостерігається зростання попиту на зерно внаслідок збільшення населення світу та збільшення вживання зернових

продуктів як основного джерела харчування. У той же час, зростає конкуренція на ринку зерна, що може вплинути на ціни та призвести до нестабільності ринку.

Україна є одним з провідних експортерів зерна в світі та має значний вплив на світовий зерновий ринок. До повномасштабної агресії росії в Україні спостерігалось стабільне зростання виробництва зернових культур, зокрема пшениці, ячменю, кукурудзи.

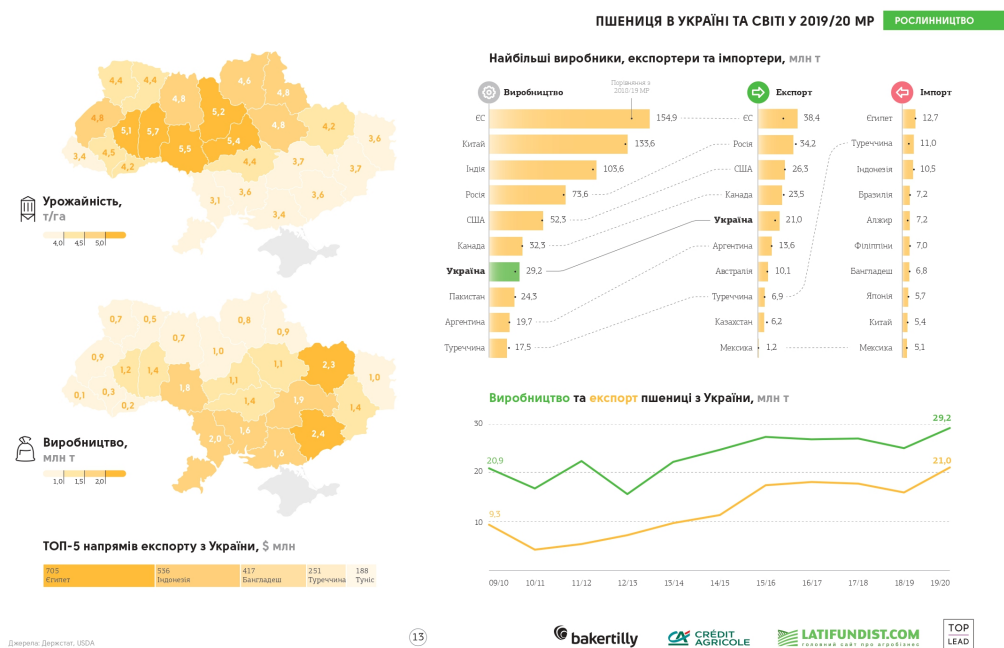


Рисунок 1.1 – Пшениця в Україні та Світі [1]

За даними Державної служби статистики України, загальний обсяг збору зернових та зернобобових культур в 2021 році склав приблизно 86 млн тонн.[2]

За даними Міністерства розвитку економіки, торгівлі та сільського господарства України, загальний обсяг експорту зернових та олійних культур у 2021 році склав 48,8 млн тонн. Найбільшими покупцями українського зерна були країни Азії та Африки, зокрема Індія, Єгипет, Бангладеш та Іран. Тільки пшениці було експортовано за рік 18,7 млн. тонн (Рисунок 1.2).

Динаміка виробництва та експорту української пшениці, млн тонн

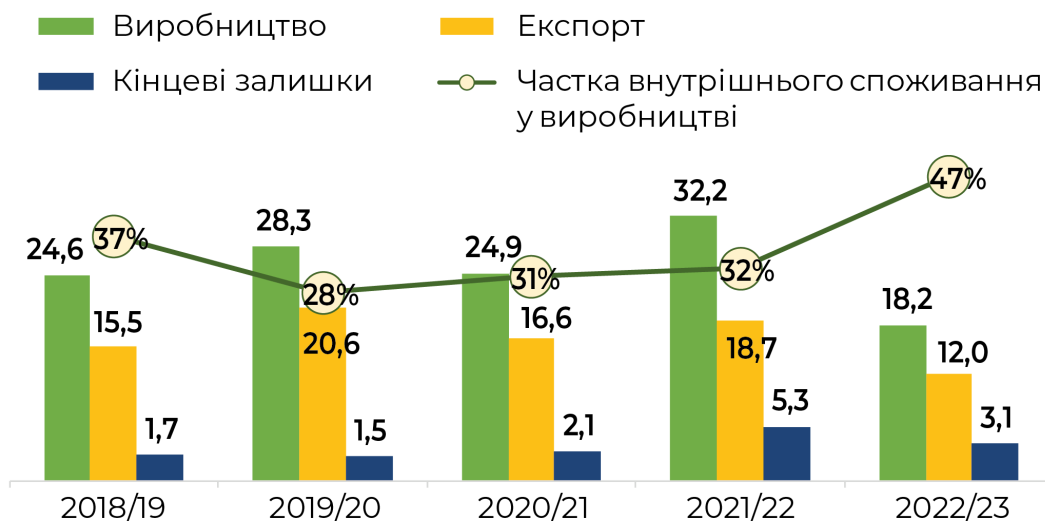


Рисунок 1.2 – Експорт та виробництво пшениці

Стан справ на зерновому ринку в Україні в 2022 році був надзвичайно складним. Після повномасштабної агресії росії, що призвела до блокування морських портів був створений зерновий коридор. Коридор повинен був забезпечити безперервний експорт зернових культур через українські порти на Чорному морі. З початку 2023 року експорт зернових за допомогою коридору зменшився на 35%, порівняно з аналогічним періодом минулого року [3].

Однак, експерти зазначають, що зменшення експорту зерновим коридором було зумовлене рядом проблем, зокрема збільшенням вартості транспортування та недостатньою підготовкою портів до виконання збільшеного обсягу робіт.

Згідно з останніми даними, з початку роботи зернового коридору Україна експортувала понад 26 мільйонів тонн зернових та інших агропродуктів.

В цілому, використання зернового коридору було надважливим для підтримки експортного потенціалу зернового ринку в умовах геополітичних та економічних труднощів. Однак, існували серйозні технічні та організаційні

проблеми, які необхідно було вирішувати для покращення роботи коридору та підтримки стабільного функціонування зернового ринку в Україні.

Основний напрям роботи зернотрейдингових компаній в Україні полягає в закупівлі зернових культур у фермерів та інших виробників, їх зберіганні на складах, переробці та продажу на внутрішньому та зовнішньому ринках.

Закупівля зернових культур здійснюється шляхом придбання врожаю від виробників, організацію контрактних відносин з фермерськими господарствами та іншими постачальниками сільгосппродукції. Компанії також ведуть роботу з визначення цін на зернові культури, які формуються на основі світових тенденцій та попиту на ринку.

Трейдингові компанії зберігають зерна на складах, де ведеться жорсткий контроль умов зберігання та забезпечується його якість. Після зберігання зерно може бути піддане переробці, такій як очищення, сушіння, сортування або упаковка, після чого воно готове для продажу.

Основні зернові культури, що вирощуються та використовуються в зернотрейдингу, включають пшеницю, кукурудзу, ячмінь та сою. Кожна з цих культур має свої унікальні властивості, використовується в різних галузях та має свої класифікації.

Пшениця - одна з основних зернових культур, яку вирощують у всьому світі. Залежно від сорту та використовуваного напрямку, пшениця класифікується на такі види:

- м'яка пшениця (біла та червона) - використовується для виробництва хліба, випічки та інших пекарських виробів;
- тверда пшениця (також біла та червона) - використовується для виробництва макаронних виробів та інших продуктів з високим вмістом білка.

Кукурудза - ще одна основна зернова культура, яка вирощується в багатьох країнах. Вона використовується як корм для тварин, для виробництва харчових

продуктів, а також для виробництва біопалива. Класифікація кукурудзи включає такі види:

- жовта кукурудза - найбільш поширена, використовується як корм та для виробництва кукурудзяного шроту;
- біла кукурудза - використовується для виробництва кукурудзяної муки, харчових продуктів та кукурудзяного шроту;
- цукрова кукурудза - використовується для виробництва цукру та біопалива.

Ячмінь - зернова культура, яка використовується як корм для тварин, для виробництва пива, а також відіграє важливу роль у вирощуванні насіння. Класифікація ячменю включає такі види:

- пивоварний ячмінь - використовується для виробництва пива та інших алкогольних напоїв;
- кормовий ячмінь - використовується як корм для тварин;
- технічний ячмінь - використовується для виробництва насіння та інших технічних цілей.

Овес - це зернова культура, яка використовується як корм для тварин, а також як продукт харчування для людей, наприклад, вівсяна каша. Класифікація овесу включає такі види:

- кормовий овес - використовується як корм для тварин;
- харчовий овес - використовується для виробництва харчових продуктів та як корм для тварин;
- насіннєвий овес - використовується для вирощування нових партій овесу.

Зернотрейдингові компанії здійснюють вирощення або покупку та експорт зернових культур, що є однією з ключових галузей експорту

сільськогосподарської продукції з України. Для цього вони забезпечують відповідну логістику для транспортування зернових культур до морських портів, де вони експортуються до інших країн світу.

Окрім того, зернотрейдингові компанії займаються розвитком та впровадженням новітніх технологій в галузі, що дозволяє їм ефективніше здійснювати свою діяльність та підвищувати її конкурентоспроможність.

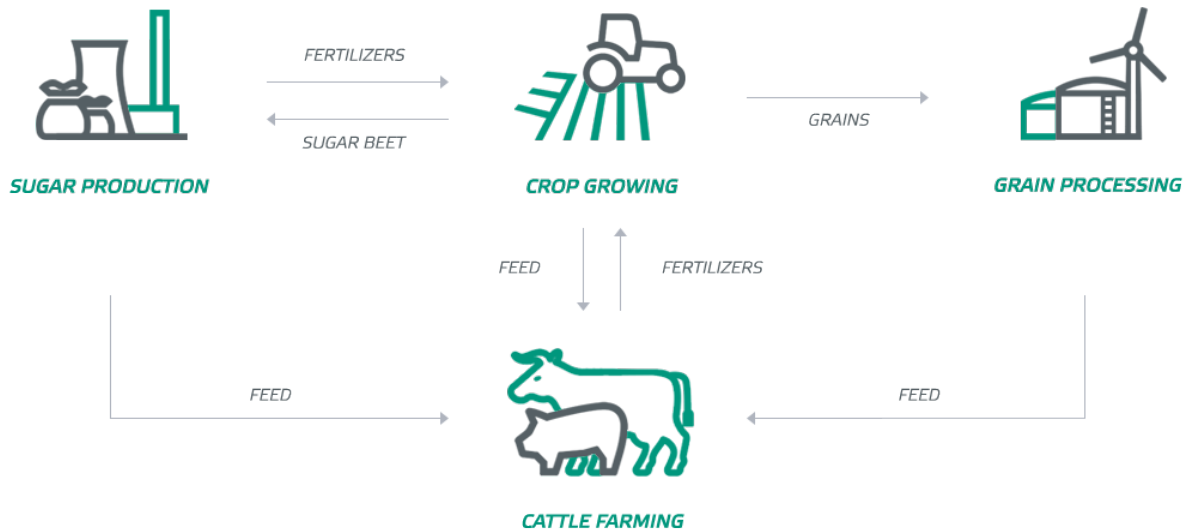


Рисунок 1.3 – Умовна структура обігу сировини на зернотрейдинговому господарстві[4]

Описуючи предметну область теми магістерського дипломного проекту, варто також роз'яснити одну з найважливіших речей в сфері торгівлі агрокультурами – моніторинг цін.

Моніторинг цін на ринку зерна є важливою частиною діяльності трейдерів, які купують та продають зерно на міжнародному або внутрішньому ринках. Дана процедура дозволяє встановлювати оптимальні ціни на продукцію та забезпечувати максимальну прибутковість підприємства.

Моніторинг дає змогу оцінити поточну ситуацію на ринку та визначити оптимальні ціни для купівлі та продажу зерна. Він допомагає вирішувати такі питання, як вибір ринку збуту, максимально можливу ціну продажу зерна та підтримку оптимальних рівнів запасів на складах. Крім того, моніторинг цін на

зерно дозволяє знайти нових клієнтів та зберігати існуючих, що забезпечує стабільний розвиток бізнесу.

Спостереження за цінами проводиться на різних рівнях: глобальному, регіональному та місцевому. Зернотрейдери отримують інформацію про ціни на товар від різних джерел, таких як інтернет-ресурси, фінансові організації, аграрні журнали або безпосередньо від виробників та фермерів. Для моніторингу використовуються також спеціальні програми, які дозволяють вести аналіз цін на зерно на різних ринках світу. Про принцип роботи подібних програм детальніше розказано в наступному підрозділі.

Вибір правильних культур для посадки на сільськогосподарських землях є надзвичайно важливим фактором для досягнення успішних показників при зборі урожаю і збалансованого розвитку господарства. Неправильний вибір може призвести до зниження врожайності, збільшення витрат на вирощування, використання землі та ресурсів, а також до забруднення навколишнього середовища і виснаження ґрунта.

Вибір урожаю залежить від різних факторів, таких як клімат, ґрунт, наявність води та добрив, рівень попиту на ринку, наявність робочої сили тощо. Важливо також враховувати ротацію культур, яка допоможе зберегти поживні речовини в ґрунті та знизити ризик зараження шкідниками та хворобами.

1.2 Аналіз існуючих СШ й інтелектуальних систем предметної області

В данному розділі буде розглянуто аналіз існуючих систем штучного інтелекту в галузі сільського господарства та зернотрейдингу. З використанням інноваційних технологій, в сучасному світі з'явилися нові можливості в обробці великих об'ємів даних для прийняття більш точних рішень.

Існує кілька видів систем штучного інтелекту, які використовуються в галузі. Ось декілька прикладів:

- Системи машинного навчання та аналізу даних: вони використовуються для прогнозування цін на зерно, аналізу попиту та пропозиції на ринку, а також для прогнозування погодних умов та впливу цих умов на врожайність;
- Системи автоматичного контролю якості: вони допомагають підтримувати відповідність якості зерна до вимог покупців шляхом автоматичного контролю вологості, густини, чистоти та інших параметрів;
- Системи автоматичного моніторингу: вони дозволяють автоматично відстежувати ціни на зерно та інші показники ринку зерна, забезпечуючи зручний і швидкий доступ до актуальних даних;
- Системи автоматичного управління: вони використовуються для автоматизації процесів торгівлі зерном, таких як контроль запасів, управління логістикою та забезпечення доставки зерна на місце призначення.

Системи машинного навчання та аналізу даних стають все більш популярними в галузі сільського господарства. Розвиток цих систем дозволяє підвищити ефективність торгівлі зерном і зменшити ризики, пов'язані з коливанням цін на ринку. Одним з прикладів використання систем машинного навчання є прогнозування цін на зерно на основі аналізу історичних даних та врахування показників попиту та пропозиції на ринку. Це дозволяє зернотрейдерам зробити більш обґрунтовані рішення при купівлі та продажу.

Інший приклад - використання систем аналізу даних для виявлення тенденцій на ринку та ідентифікації можливих ризиків. За допомогою цих систем можна виявити зміни на ринку зерна, зокрема зміни попиту та пропозиції, тенденції щодо цін та інші фактори, що можуть вплинути на результативність торгівлі.

Також існують системи, які допомагають зернотрейдерам вибрати оптимальний маршрут доставки зерна. Вони аналізують інформацію про вартість палива, вартість доставки та інші фактори, що впливають на вибір маршруту.

Узагальнюючи, використання систем машинного навчання та аналізу даних у зернотрейдингу дозволяє знизити ризики та підвищити ефективність торгівлі агрокультурами, що робить їх важливим інструментом для професіоналів у цій галузі.

Необхідно розглянути існуючі експертні системи з зернотрейдингу, що є на ринку, розпочнемо з системи Crop Monitoring від компанії EOS [5]. Система є інноваційним рішенням у сфері моніторингу врожаїв та земельних ділянок у сільському господарстві. Вона забезпечує зручний та швидкий доступ до актуальної інформації про стан врожаїв, включаючи стан їхнього здоров'я, рівень зростання та плодоношення.



Рисунок 1.4 – Інтерфейс EOS Crop Monitoring

Crop Monitoring використовує високоточні космічні технології для збору даних про врожаї, а також інші технології, такі як штучний інтелект, машинне навчання та глибинний аналіз даних, для обробки цих даних та надання точних та зрозумілих звітів користувачам.

За допомогою Crop Monitoring фермери можуть відслідковувати рівень вологості ґрунту та метеорологічні умови, що значно сприяє зменшенню ризику втрат врожаю в результаті негативних погодних умов. Також система дає

можливість користувачам стежити за розвитком хвороб та шкідників, що дозволяє своєчасно вживати заходів для їхнього усунення та збереження врожаю.

Crop Monitoring є надійним та ефективним інструментом для оптимізації сільського господарства, що дозволяє підвищити якість вирощування врожаїв та збільшити врожайність земельних ділянок.

Переваги системи EOS Crop Monitoring включають:

1. Висока точність: система використовує супутникові дані та машинне навчання для точного визначення стану рослинності на полях.

2. Швидкість та ефективність: система працює в режимі реального часу, що дозволяє отримувати швидку інформацію про стан посівів та вчасно реагувати на можливі проблеми.

3. Зручний інтерфейс користувача: EOS Crop Monitoring має зручний та інтуїтивно зрозумілий інтерфейс, що дозволяє користувачам легко здійснювати моніторинг посівів.

4. Можливість відстежувати історію: система зберігає дані про стан посівів протягом тривалого часу, що дозволяє відстежувати їхню історію та проводити аналізи.

5. Доступність даних: система надає доступ до даних про стан посівів та рекомендації щодо їхнього поліпшення в будь-який час та з будь-якого місця з використанням Інтернету.

Недоліки системи EOS Crop Monitoring:

1. Висока вартість: система EOS Crop Monitoring може бути витратною для невеликих фермерських господарств.

2. Залежність від погодних умов: система EOS Crop Monitoring може бути обмежено ефективною в разі погіршення погодних умов.

3. Обмежена географічна доступність: EOS Crop Monitoring може бути недоступною в деяких регіонах, де не має достатнього покриття супутникових даних.

Наступна система яку розглянемо – BushelFarm [6]. Це інформаційно-аналітичний онлайн-сервіс для сільськогосподарських підприємств, який дозволяє власникам і менеджерам дистанційно відстежувати та управляти виробництвом. Зокрема, система надає можливість моніторити врожайність, аналізувати погоду, контролювати витрати на господарство та здійснювати облік продукції. Крім того, сервіс дає змогу взаємодіяти з постачальниками та покупцями зерна, забезпечує зв'язок з бухгалтерією та банківськими рахунками. Доступ до сервісу можливий як з комп'ютера, так і з мобільних пристроїв через веб-браузер або мобільний додаток.

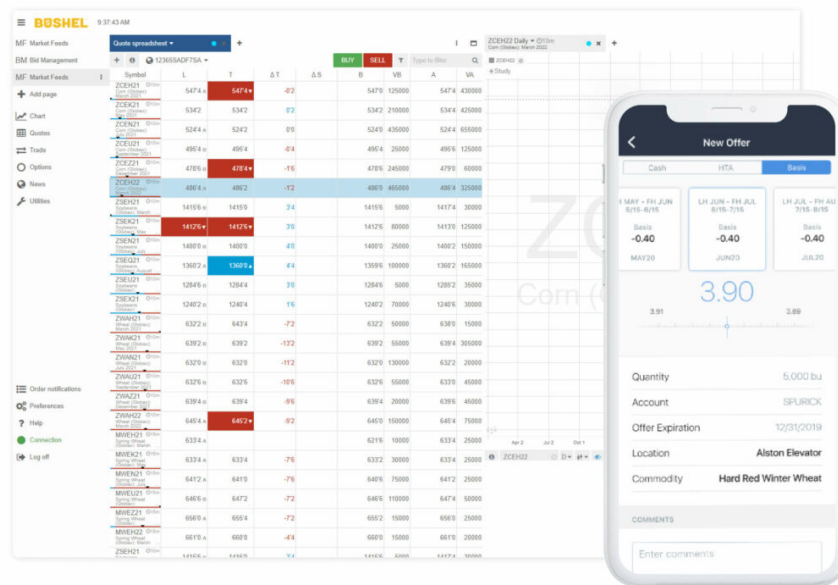


Рисунок 1.5 – Інтерфейс BushelFarm

Переваги системи Bushel Farm:

1. Зручний та простий інтерфейс, який дозволяє швидко та легко отримати необхідну інформацію про стан полів, врожайність та інші показники.

2. Висока точність та швидкість отримання даних завдяки використанню сучасних технологій, таких як супутникова зйомка та машинне навчання.

3. Можливість моніторингу стану полів та врожайності в режимі реального часу, що дозволяє оперативно реагувати на зміни у погодних умовах та інші фактори, що впливають на вирощування рослин.

4. Можливість інтеграції з іншими системами та пристроями, що дозволяє використовувати отримані дані в різних сферах сільського господарства та зернотрейдингу.

Недоліки системи Bushel Farm:

1. Платна підписка на користування системою, що може бути недосяжною для деяких користувачів.

2. Обмежені можливості вибору параметрів моніторингу та аналізу даних, що може не задовольнити потреби деяких користувачів.

3. Потреба у високій якості зображень для точності аналізу, що може бути проблемою в деяких регіонах з обмеженим доступом до високошвидкісного Інтернету або супутникової зв'язку.

Остання система яку буде проаналізовано – Agworld [7]. Це хмарне програмне забезпечення для управління господарством, яке допомагає зернотрейдерам та агрономам управляти роботами на господарстві, збирати та аналізувати дані про врожайність, витрати на засоби захисту рослин, обладнання та робочу силу. Система забезпечує зручний доступ до даних, що дозволяє користувачам управляти господарством ефективніше, використовуючи більш точну та актуальну інформацію.

Agworld також містить функції для планування робіт на господарстві, включаючи планування посіву, захист від хвороб та шкідників, а також підживлення рослин. Крім того, система дозволяє співпрацювати з іншими

учасниками господарювання, такими як консультанти, постачальники та покупці, та спільно працювати над різними проектами.

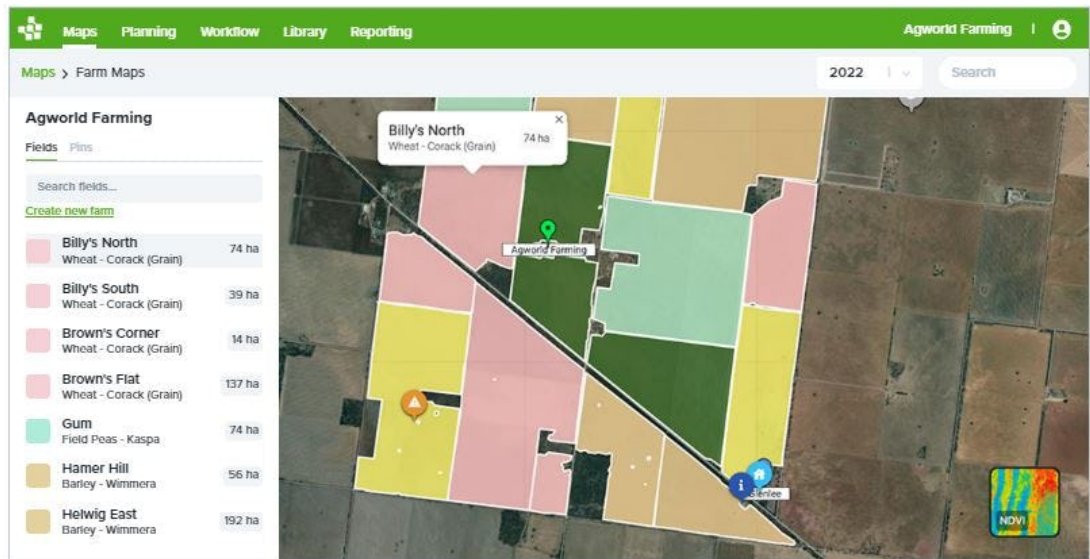


Рисунок 1.6 – Інтерфейс Agworld

Переваги системи Agworld:

1. Зручний та інтуїтивно зрозумілий інтерфейс: система Agworld має простий та зрозумілий інтерфейс, що дозволяє користувачам легко орієнтуватись в програмі та швидко знаходити необхідну інформацію.

2. Комплексні можливості: система Agworld дозволяє вести облік виробництва на всіх етапах, починаючи від планування посівів та закінчуючи збором врожаю та обліком витрат.

3. Можливість співпраці: система Agworld дозволяє співпрацювати з іншими фермерами, агрономами та постачальниками, що дозволяє покращити ефективність та рентабельність виробництва.

4. Мобільність: система Agworld має мобільну версію, що дозволяє користувачам працювати з програмою в будь-якому місці та в будь-який час.

Недоліки системи Agworld:

1. Висока вартість: система Agworld вимагає певних інвестицій від користувачів, що може стати перешкодою для невеликих фермерських господарств.

2. Залежність від Інтернету: система Agworld потребує підключення до Інтернету, що може стати проблемою в умовах низької швидкості Інтернету або відсутності зв'язку в деяких районах.

3. Необхідність тренінгу та підтримки: через великий функціонал та можливості системи Agworld, користувачам може знадобитись додаткова підтримка та навчання, щоб максимально використовувати систему.

Візуалізація даних та використання штучного інтелекту в сільському господарстві стає все популярнішим. Система EOS Crop Monitoring надає точну інформацію про стан посівів та забезпечує моніторинг та прогнозування врожаю. BushelFarm і Agworld допомагають фермерам у веденні документації та аналізі даних щодо їх сільськогосподарських ділянок. Такі системи є важливим інструментом для фермерів, оскільки вони дозволяють збільшувати врожайність та оптимізувати процеси вирощування рослин. Однак, вони можуть бути високо вартісними та потребувати певного рівня технічних знань для використання.

Штучний інтелект дає можливість пов'язати умови вирощування сільськогосподарських культур із поведінкою після збору врожаю. На наступному етапі він може надати виробникам або трейдерам інформацію про те, як адаптувати вибір сортів і умови вирощування, щоб отримати продукт, який задовольняє очікування ринку з точки зору моменту збору врожаю, якості, терміну зберігання та зменшення втрат і відходів.

Представники промисловості вирощування культур та зернотрейдингу формулюють деякі погляди у спробах відповісти на такі запитання:

- Наскільки надійним має бути контролер експертної системи?

- Які можливі сценарії, за яких штучний інтелект може бути розгорнутий для прийняття рішень на фермі, управлінням автопарком, функціональної оптимізації або ухвалення керованих машинами рішень?
- Який рівень якості даних необхідний для експертної системи?
- Хто несе відповідальність, коли рішення приймає система ШІ?
- Правова основа та база безпеки для розгортання рішень щодо збору даних.

У європейських інститутах заклик до «орієнтованого на людину штучного інтелекту» все ще є сильним, але означає, що людей потрібно захистити від штучного інтелекту, і це призводить до пропозиції, що регулювання штучного інтелекту розглядається більше як інструмент захисту споживачів і громадян, ніж як справжнє регулювання штучного інтелекту. У кількох випадках штучний інтелект сприймається як додаткова неконтрольована залежність сільськогосподарського сектора. Однак людина, машина та прогрес пов'язані між собою: люди будують і використовують машини, щоб покращити чи спростити життя. У цьому сенсі штучний інтелект є лише ще одним кроком у покращенні машин і досягненні показників виробництва, економії та якості, які в іншому випадку вимагали б більш тривалого часу або були б недоступні взагалі.

Сфера дії регулювання настільки величезна, що побічний збиток здається неминучим. Сільське господарство та торгівля зерном представляє різні фази та аспекти виробництва та реалізації, які можуть потребувати більшої автоматизації та застосування штучного інтелекту без істотного збільшення ризику порівняно з реальними операціями, керованими людьми. Невичерпні приклади використання штучного інтелекту в сільському господарстві:

- автономне керування транспортними засобами/машинами;
- автоматизована координація/управління логістикою;

- визначення якостей культури на елеваторах;
- протоколи моделювання виробництва сільгосп продукції;
- управління культурами від посіву до збору врожаю (виявлення хвороб і цвілі, виявлення шкідників);
- координація роботи людини;
- підтримка прийняття рішень, що значно відрізняється порівняно з чистою функцією консультування;
- координація всіх попередніх функцій в одному виробничому ланцюжку;
- ведення загальної бази знань та підтримка прийняття рішень в економічних, фінансових та організаційних аспектах;
- прогнозування ринкових цін на продукцію.

Використання штучного інтелекту пропонує незаперечні економічні та виробничі переваги, коли йдеться про ефективність роботи. Цей інструмент надає ключ до розв'язання багатьох проблем, які виникають через зростання обсягів торгівлі, підвищує стійкість комерційних процесів, та зменшує зростаючу складність систем. Вдосконалення штучного інтелекту в сфері зернотрейдингу та сільськогосподарських роботах може потребувати більш орієнтованої на сектор оцінки ризику. Коли виробник обладнання використовує штучний інтелект у власному внутрішньому процесі проектування та власних контрольованих наборах даних, він контролює ризики, у тому числі ті, які є унікальними для штучного інтелекту, і несе повну відповідальність. Хоча алгоритми штучного інтелекту досить добре відомі та доступні, головною зміною є наявність великої кількості даних, які потрібно оцінити та зв'язати один з одним. Кількість вхідних даних і побудова нових статистичних співвідношень викликають відчуття втрати контролю, оскільки люди не можуть контролювати кожен крок з однаковою швидкістю. Однак контроль якості у джерелах даних

може вимагати від систем обробки та оцінки цінності даних високу точність, щоб отримати надійний результат. Безпека також є майбутнім ключовим елементом для створення довіри до таких схем.

Окрім основних даних, наданих машині на початку розробником, машина повинна продовжувати свій розвиватися протягом усього терміну служби. Системи самонавчання все ще вдосконалюються після розміщення на ринку, використовуючи локально доступні дані та статистичні дані, отримані з інших машин. Слід зауважити, що в багатьох випадках функція, керована штучним інтелектом, все ще повинна працювати під наглядом людини, але відповідні ризики для всієї машини та її середовища можуть бути не повністю зрозумілі та охоплені заздалегідь.

З розповсюдженням штучного інтелекту та зростанням складності систем керування все більше й більше інтелектуальних систем інтегруються на підприємствах, а оператори простих та циклічних процесів замінюються такими системами. Зі зростанням кількості «розширених» функцій та сервісів, які вимагають підключення та співпраці фахівців з різних підрозділів, зростає потреба в контрольованому обміні даними та заходах кібербезпеки. Для промисловості це означатиме доступ через платформу, а не прямий доступ до машин, щоб захистити інтелектуальну власність, комерційні таємниці та відповідність продукту.

На мою думку, сільське господарство перебуває на початку чергової революції, в основі якої лежать датчики, та безперервне підключення до апаратного забезпечення в поєднанні з використанням моделей даних та інтелектуальних систем. Штучний інтелект у агропродовольчій сфері розвивається як важливий інструмент разом із багатьма технологіями, обладнанням і програмним забезпеченням, які взаємопов'язані між собою та можуть призвести до збільшення прибутку від економічної діяльності та послуг із доданою вартістю. Вони збирають дані з багатьох ресурсів і датчиків, застосовують методології машинного навчання та готують управлінські рішення

або навіть виконують їх самостійно. Штучний інтелект, аналітика, підключені датчики та інші новітні технології можуть ще більше збільшити врожайність, підвищити ефективність використання ресурсів, зменшити негативні економічні ризики при реалізації товару, а також створити сталість і стійкість у вирощуванні та продажі сільськогосподарських культур. Існує потреба в подальшому розвитку апаратного забезпечення для визначення та виконання рішень, але багато роботи та уваги також потрібно приділяти даним, аналітиці та моделям.

1.3 Постановка проблеми та формування задач

Постановка задачі для розроблюваної експертної системи полягає в тому, щоб на основі введених даних про родючість ґрунту, кліматичні умови, доступні водні ресурси та багатьох інших факторів, рекомендувати оптимальний вибір культур для вирощування на даних сільськогосподарських землях. Задача полягає в тому, щоб забезпечити максимальний врожай і найбільший дохід для зернотрейдерів, враховуючи наявні обмеження, такі як погодні умови, зміну клімату, трудові ресурси тощо. Система повинна забезпечувати користувачів різними факторами вибору культур, щоб забезпечити оптимальний вибір для наявних умов.

Експертна система призначена для допомоги зернотрейдерам та фермерам у виборі найкращого врожаю для посіву на їх полях. Основна мета системи полягає в забезпеченні фермерів інформацією про найбільш підходящі культури для їх земельних ділянок на основі даних про кліматичні умови, тип ґрунту, обробіток і рівень доступних ресурсів.

Основна задача експертної системи полягає в тому, щоб допомогти максимально оптимізувати зусилля та ресурси підприємства, що призведе до підвищення продуктивності та ефективності їх сільськогосподарської діяльності. Для досягнення цієї мети система повинна має аналізувати велику кількість даних про різні культури, зерновий ринок, кліматичні умови та інші чинники, а також забезпечувати зручний та доступний інтерфейс для користувачів.

Задачі зернотрейдингової експертної системи включають:

Збір та аналіз даних про землі господарства, включаючи тип ґрунту, попередні культури, вологість ґрунту, доступні ресурси;

Аналіз кліматичних умов та прогнозування погоди в регіоні та визначення культур, які будуть найплодовитішими в даних умовах;

Визначення найбільш підходящих культур для посіву на конкретних ділянках ферми на основі аналізу даних;

Порівняння вибраних культур за кількістю врожаю, вимогами до ресурсів та іншими факторами;

Рекомендації щодо оптимальних методів обробки та догляду за посівами, що дозволить фермерам отримати найвищу продуктивність врожаю.

Крім того, система повинна бути здатна працювати в режимі онлайн. Також, система повинна бути здатна враховувати не тільки рекомендації щодо культур, але і інформацію про їх вирощування, таку як тип ґрунту, потребу в поливі. Система повинна мати можливість збирати та зберігати статистичні дані про вирощування культур на різних фермах, щоб забезпечити кращі рекомендації для фермерів у майбутньому.

Узагальнюючи, метою експертної системи є надання оптимальних рекомендацій щодо вибору культур для вирощування на землях, з урахуванням різних факторів, таких як погодні умови, тип ґрунту, стан зернового ринку та інші. При цьому система повинна працювати в режимі онлайн та збирати та зберігати статистичні дані для подальшого аналізу та удосконалення рекомендацій.

1.4 Обґрунтування вибору підходів і технологій для проектування та створення експертної системи та її компонентів

Обраний підхід мікросервісної архітектури [8] в експертній системі дозволяє забезпечити її гнучкість та масштабованість. Замість створення одного

великого монолітного додатку, система розбита на декілька незалежних сервісів, які можуть бути розгорнуті окремо і змінені без впливу на решту системи.

Використання технологій `node.js` [9] та `Webpack` [10] дозволяє розробникам забезпечити високу продуктивність та швидкість розробки. База даних `PostgreSQL` [11] обрана через її стабільність та розширюваність. Система контролю версій `Git` [12] дозволяє зберігати історію змін коду, що допомагає розробникам працювати в команді та контролювати якість розробки.

Для контейнеризації використовується `Docker` [13], що дозволяє створювати та розгортати віртуальні контейнери з програмним забезпеченням системи на будь-якому сервері, що підтримує `Docker`. Розгортання на сервері `Amazon` дозволяє забезпечити швидкий доступ до системи та забезпечити її безпеку та стабільність.

Мікросервісна архітектура - це підхід до проектування програмного забезпечення, де додаток розбивається на невеликі незалежні модулі - мікросервіси, які можуть працювати окремо один від одного і взаємодіяти між собою за допомогою `API`. Кожен мікросервіс має свою власну базу даних та незалежно масштабується, що дозволяє легко додавати нові функціональність без впливу на роботу інших мікросервісів.

Один з основних переваг мікросервісної архітектури полягає в тому, що вона дозволяє зменшити час розробки та впровадження нового функціоналу, оскільки окремі мікросервіси можуть розроблятися та впроваджуватися незалежно один від одного. Крім того, така архітектура забезпечує більшу гнучкість, масштабованість та надійність, оскільки помилки у одному мікросервісі не впливають на роботу інших, а також масштабування окремих компонентів системи може відбуватися незалежно від інших.

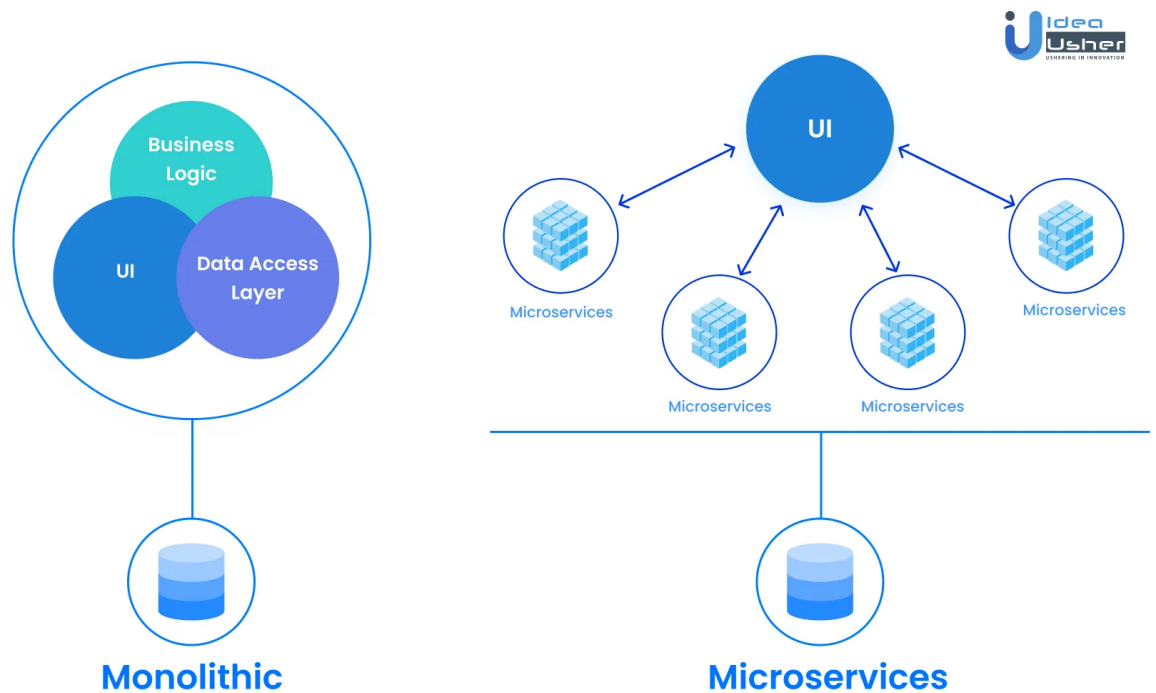


Рисунок 1.7 – Порівняння двох основних типів архітектур[13]

Монолітна архітектура є традиційним підходом до розробки програмного забезпечення, при якому весь код розміщується в одному блоку. Із збільшенням розміру проекту цей підхід може виявитись недостатньо ефективним, оскільки при зміні будь-якої частини коду необхідно перекомпілювати та розгортати весь моноліт. Крім того, монолітна архітектура може призводити до технічного боргу, оскільки нові функції можуть бути додані без уваги до залежностей та зруйнувати структуру системи.

Мікросервісна архітектура вирішує деякі з цих проблем шляхом розбиття системи на окремі компоненти - мікросервіси. Кожен мікросервіс відповідає за конкретну функціональність, і може бути розгорнутий незалежно від інших сервісів. Це дозволяє розробникам ефективніше працювати, оскільки кожен сервіс може бути розгорнутий, перевірений та оновлений окремо від інших. Мікросервіси також дозволяють розробникам використовувати різні технології та мови програмування для кожного сервісу в залежності від його функціональних вимог.

Однак, мікросервісна архітектура також має свої недоліки. Розробка та управління мікросервісами може виявитись складнішою та витратнішою, оскільки необхідно забезпечувати комунікацію між сервісами та контролювати залежності між ними.

Node.js - це середовище виконання JavaScript, яке дозволяє виконувати JavaScript-код поза браузером на сервері. Node.js побудований на основі рушія V8 від Google, який використовується в браузері Chrome. Однією з головних переваг Node.js є те, що він дозволяє розробникам використовувати JavaScript для написання серверного коду, що спрощує розробку веб-додатків.



Рисунок 1.8 – node.js

Node.js є досить легким та швидким середовищем виконання, що дозволяє збільшити продуктивність розробки веб-додатків. Node також має вбудовану підтримку асинхронності, що дозволяє виконувати багато операцій без очікування результату. Це забезпечує більшу ефективність та швидкість додатків.

Node.js має широку спільноту розробників та багато npm пакетів, які можна використовувати для розробки веб-додатків. Це забезпечує більшу гнучкість та швидкість розробки проектів.

Однак, Node.js має деякі недоліки, такі як висока витрата пам'яті та нестабільність під час високого навантаження. Також, якщо розробник необізнаний з асинхронним програмуванням, може виникнути проблема зі зростанням складності коду та важкістю його розуміння.

React [14] є однією з найпопулярніших бібліотек для розробки інтерфейсів користувача веб-додатків. Він був розроблений компанією Facebook і вперше був випущений в 2013 році. React працює на стороні клієнта і дозволяє швидко створювати ефективні та масштабовані веб-додатки.

Однією з головних переваг React є використання віртуального DOM (Document Object Model), що дозволяє покращити продуктивність додатків. React оновлює лише ті частини сторінки, які дійсно змінилися, а не всю сторінку, що дозволяє запобігти затримкам та нерівномірній продуктивності. Крім того, React має широкий вибір сторонніх бібліотек та компонентів, що дозволяє значно спростити розробку та збільшити її ефективність.

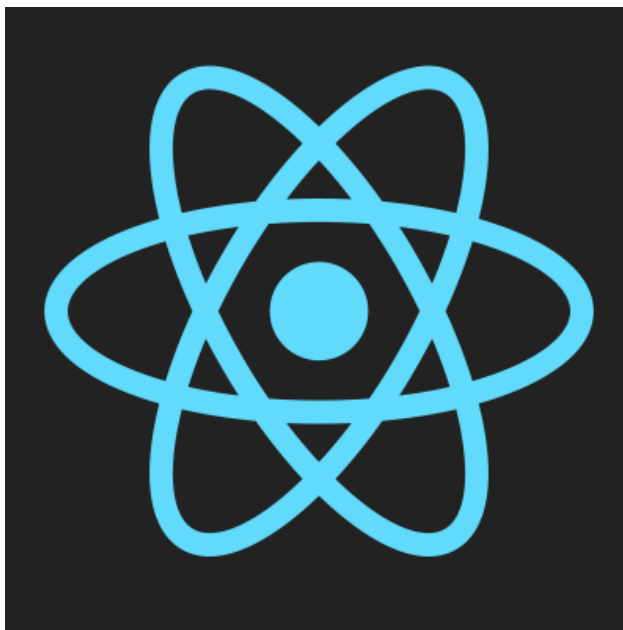


Рисунок 1.9 – React

Основними принципами React є декларативний підхід та компонентна архітектура. Розробники описують, як повинен виглядати результат, а React відповідає за те, щоб відображення було згідно з цим описом. Компонентна

архітектура дозволяє розбити додаток на невеликі, самодостатні блоки, що зменшує складність коду та спрощує його розуміння та підтримку.

Завдяки своїм перевагам та докладній документації, React є популярним інструментом для розробки веб-додатків, особливо для складних та вимогливих.

Іншою технологією що була застосоввана є Webpack. Це модульний збірник для JavaScript, що використовується для збірки веб-додатків. Він дозволяє зібрати багато окремих файлів JavaScript, CSS і зображень в один готовий для використання файл більшого розміру, який можна підключити до сторінки HTML.

Однією з основних переваг використання Webpack є його здатність до збирання коду, що складається з багатьох модулів, у один файл. Це дозволяє зменшити час завантаження сторінки, що особливо важливо для великих проектів з великою кількістю залежностей.

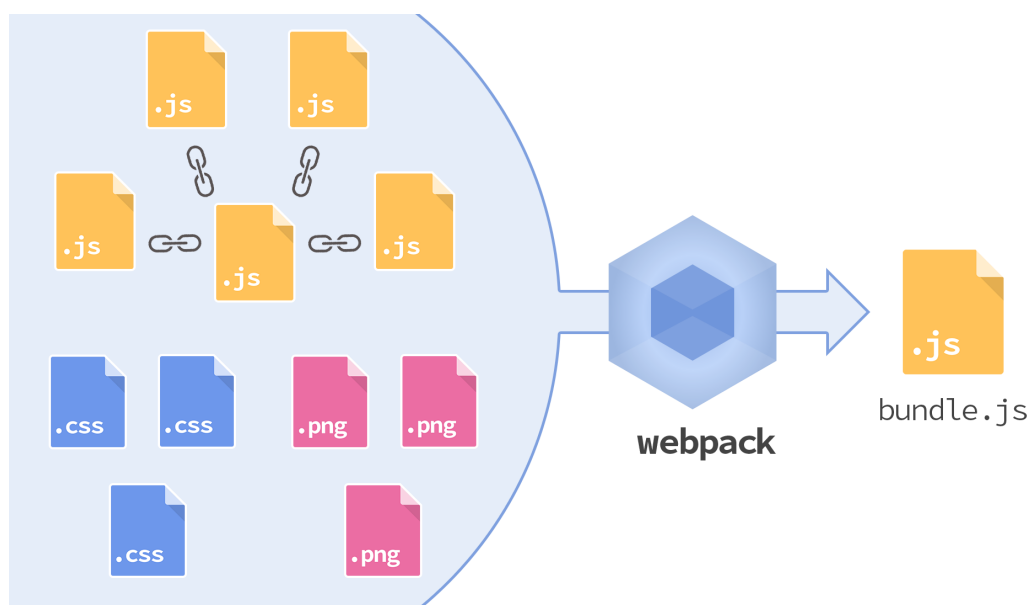


Рисунок 1.10 – Збірка Webpack

Webpack також дозволяє використовувати різні типи файлів, такі як CSS, SCSS, зображення, шрифти та інші, і перетворювати їх в готові для використання

файли, що допомагає знизити кількість запитів до сервера і покращити продуктивність додатку.

Іншою важливою перевагою Webpack є його здатність до роботи з модульною архітектурою, що забезпечує зручну і легку розробку веб-додатків. Також, завдяки використанню Webpack можливо автоматизувати процес збірки проекту, що забезпечує більш ефективну і швидку розробку.

Однак, використання Webpack може вимагати додаткового часу на налагодження процесу збірки, тому що потрібно налаштувати велику кількість різних опцій і параметрів. Також, при неправильному використанні може виникати проблема з перевантаженням сервера.

Node.js та Webpack - це потужні інструменти для створення веб-додатків. Node.js дозволяє використовувати JavaScript на стороні сервера, а Webpack дозволяє збирати та оптимізувати ресурси, такі як CSS, JavaScript та зображення, для покращення продуктивності та швидкості завантаження веб-додатків.

Webpack може бути використаний разом з Node.js, якщо встановити пакети webpack та webpack-cli через NPM (Node Package Manager). Для конфігурування Webpack у проекті на Node.js необхідно створити файл webpack.config.js, в якому можна налаштувати параметри збірки, такі як вихідні та вхідні файли, обробка різних типів файлів та оптимізація.

Завдяки Webpack можна розбити великі проекти на окремі модулі та бібліотеки, що дозволяє зменшити час завантаження та віддаленого виконання коду на сервері. Крім того, Webpack дозволяє зменшити розмір вихідних файлів та ресурсів за допомогою різних методів оптимізації, таких як мініфікація та злиття файлів.

PostgreSQL - це відкрита реляційна база даних з багатими можливостями, яка використовує мову SQL для зберігання та управління даними. Вона підтримує транзакції, що забезпечують безпеку та цілісність даних, і має

широкий набір функцій, що дозволяє ефективно виконувати запити до бази даних.

Однією з основних переваг PostgreSQL є її висока стабільність і надійність, що робить її популярним вибором для важких завдань. Вона також має гнучкі настройки конфігурації, що дозволяє її використовувати як для малих проектів, так і для великих додатків зі складними вимогами до обробки даних.

PostgreSQL має досить велику спільноту розробників та користувачів, що дозволяє швидко знаходити рішення для багатьох проблем і отримувати підтримку від інших користувачів. Також вона є безкоштовною та доступною для використання в різних проектах.



Рисунок 1.11 – PostgreSQL

Однією з можливостей PostgreSQL є підтримка масштабованої архітектури, що дозволяє розширювати базу даних залежно від потреб проекту. Вона також підтримує реплікацію, що дозволяє створювати резервні копії баз даних та забезпечувати безпеку даних.

Однак, PostgreSQL має деякі недоліки. Наприклад, вона може бути складною для налаштування та використання для початківців, і вона може вимагати більш високої ресурсоемності порівняно з іншими базами даних.

Docker – це відкрите програмне забезпечення, яке дозволяє контейнеризувати програмні додатки та віртуалізувати їх середовище

виконання. Docker дає можливість ізолювати додатки в окремих контейнерах, які містять усі необхідні залежності та бібліотеки для їх роботи.

Однією з головних переваг Docker є його портативність. Контейнеризовані додатки можуть бути легко переміщені між різними середовищами виконання, включаючи різні операційні системи та хмарні платформи. Крім того, Docker дозволяє ефективно використовувати ресурси сервера, зменшуючи витрати на інфраструктуру та забезпечуючи швидкий запуск додатків.

У поєднанні з Node.js, Docker дозволяє розробникам забезпечити консистентність та стабільність середовища для своїх додатків на будь-якій платформі. Контейнери Docker можуть бути легко налаштовані та розгорнуті з використанням інструментів управління контейнерами, таких як Docker Compose. Це дозволяє розробникам швидко створювати, тестувати та розгортати свої додатки відповідно до потреб проекту.

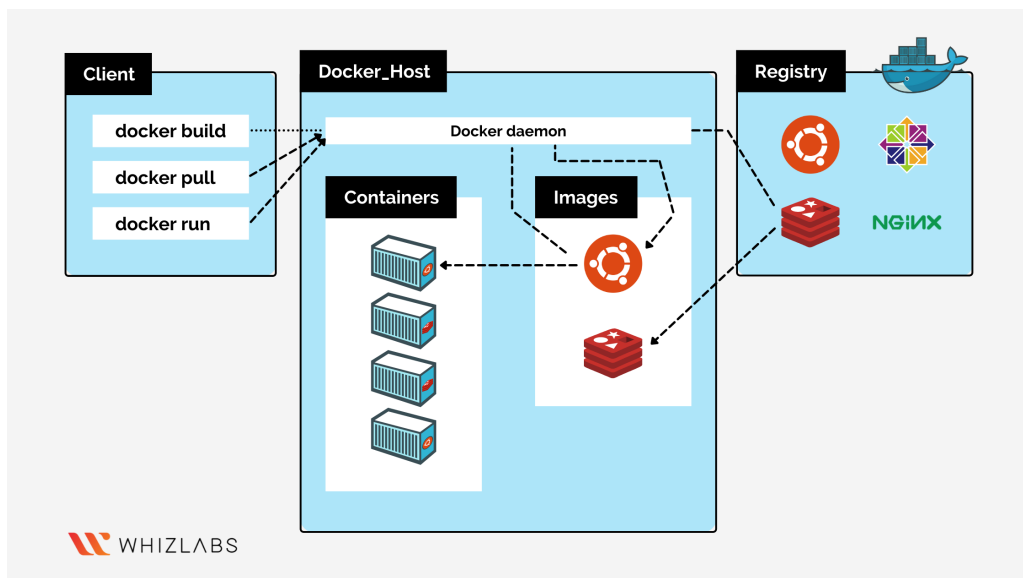


Рисунок 1.12 – Архітектура Docker[15]

Однак, Docker не є універсальним рішенням для всіх випадків використання. Деякі додатки можуть бути складними для контейнеризації, тому їх можна було б найкраще запускати на віртуальних машинах або фізичних серверах. Крім того, Docker може бути вимогливим до ресурсів системи та потребувати певного рівня досвіду для його ефективного використання.

AWS (Amazon Web Services) [16] - це хмарна платформа для розгортання та керування додатками та сервісами в інтернеті. Для розгортання експертної системи на AWS можна скористатися різними сервісами, такими як Amazon Elastic Compute Cloud (EC2), Amazon Elastic Container Service (ECS), Amazon Elastic Kubernetes Service (EKS) та інші.

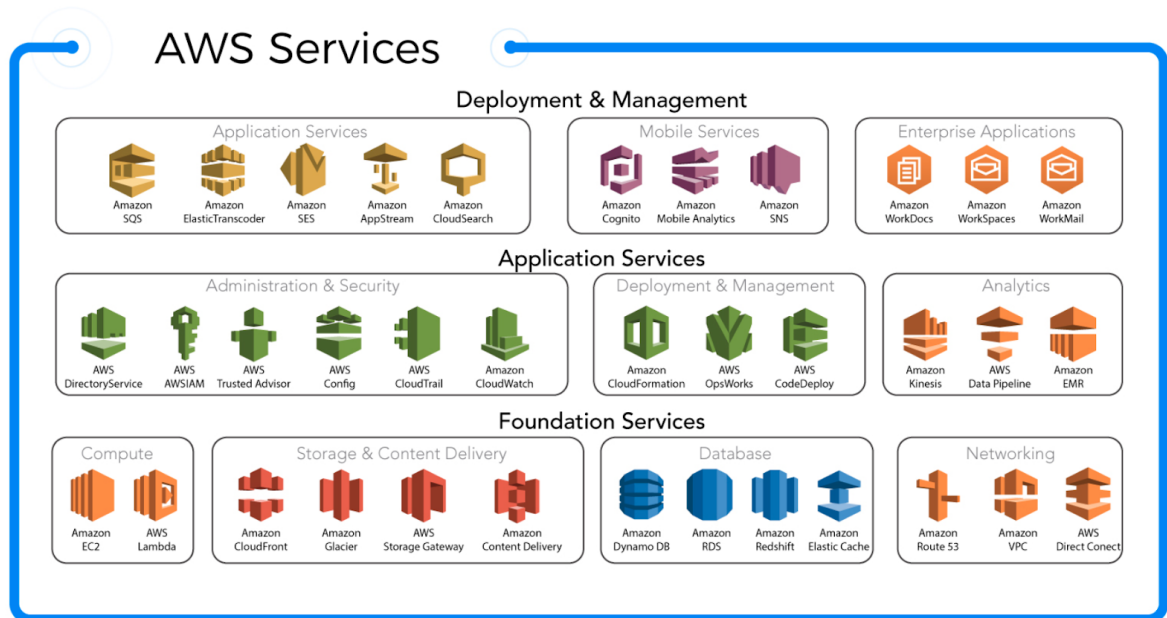


Рисунок 1.13 – Сервіси AWS

Наприклад, для розгортання експертної системи можна використовувати сервіс Amazon EC2 [17]. Для цього спочатку необхідно створити інстанс (віртуальний сервер) в AWS, встановити необхідні залежності та сконфігурувати додаток. Далі можна залити код додатку на сервер та запустити його.

Крім того, для розгортання можна використовувати контейнерні технології, такі як Docker та Amazon ECS або EKS. За допомогою Docker можна створити образ додатку та розгорнути його в контейнері на Amazon ECS або EKS. Це дозволяє швидко та зручно розгорнути та керувати додатками в хмарі.

AWS надає широкі можливості для керування та моніторингу додатків, такі як Amazon CloudWatch для моніторингу метрик, Amazon S3 для зберігання даних та інші сервіси, що дозволяють забезпечувати високу доступність та масштабованість додатків.

Розгортання експертної системи на AWS дозволяє забезпечити швидкий та безпечний доступ до додатку, максимальну доступність та масштабованість.

Отже, для розробки експертної системи та її компонентів вирішено використовувати мікросервісну архітектуру, що дозволяє ефективно розгортати та масштабувати систему. Для розробки обрано `node.js`, який дозволяє писати код на JavaScript як серверної так і клієнтської частини. Також використовуємо `webpack` для збирання JavaScript-файлів, що дозволяє знизити розмір файлів фронтенду. Для зберігання обрано PostgreSQL, який є потужною та надійною системою управління базами даних. Для контейнеризації використовується `Docker`, що дозволяє забезпечити незалежність середовища та легко розгорнути систему на іншому сервері. Нарешті, система розгортається на Amazon Web Services (AWS), що дозволяє нам забезпечити високу доступність, масштабованість та безпеку. Всі ці компоненти дозволяють ефективно розробляти та розгортати експертну систему.

2 АНАЛІТИЧНИЙ РОЗДІЛ. ХАРАКТЕРИСТИКА ЕКСПЕРТНОЇ СИСТЕМИ ТА ПОСТАНОВКА ЗАДАЧІ

В сучасному світі, з розвитком інформаційних технологій, значно зросла необхідність у використанні експертних систем для різноманітних задач. Однією з сфер, де можна успішно їх використовувати є зернотрейдинг та сільське господарство.

В процесі зернотрейдингу можуть виникати складні задачі прийняття рішень, пов'язані зі зберіганням та переробкою зерна, вибором ділянок для вирощування культур, визначенням якості товару, вибором ринку для продажу та оптимальним маршрутом доставки.

Одним з рішень для ефективного вирішення цих задач є використання інтелектуальної системи прийняття рішень, яка забезпечує оптимальний вибір між декількома варіантами.

Метою даного аналітичного розділу є дослідження можливостей використання експертної системи для прийняття рішень в зернотрейдингу та розробка такої системи на основі аналізу існуючих методів та технологій.

2.1 Характеристика об'єкта дослідження

Об'єктом дослідження є інформаційні процеси що протікають в зернотрейдингу, що складаються з інформації та даних, що застосовуються для вибору оптимальної культури при посіві. Цей процес є складним та динамічним, та становить значну складову галузі зернотрейдинга та сільського господарства. Зернові культури є ключовими продуктами в світовій торгівлі та є важливими джерелами харчових продуктів, кормів та біопалива. Процес зернотрейдингу складається з вирощування зернових культур, зберігання, транспортування, продажу та покупки.

Характеристика об'єкта дослідження полягає у тому, що вибір оптимальної культури для вирощення є складним процесом, який пов'язаний з багатьма

факторами, такими як кліматичні та географічні умови, ринкові умови, валютні курси та багато іншого. Крім того, зерновий ринок є глобальним, що охоплює багато країн, що робить його ще більш складним.

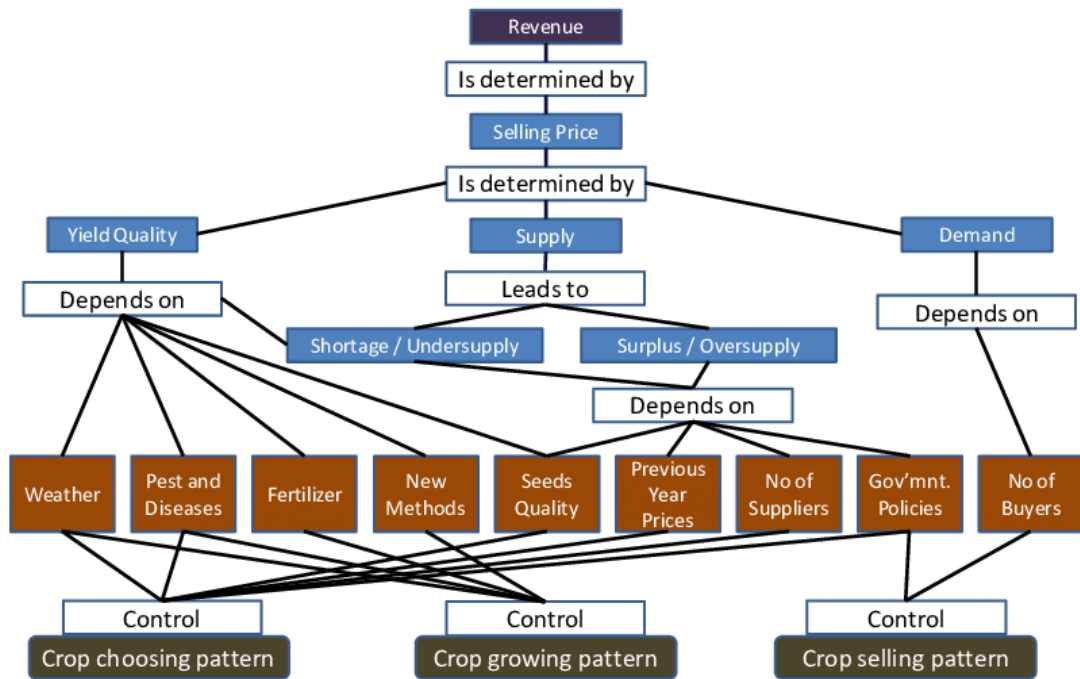


Рисунок 2.1 – Фактори, що впливають на вибір культур, моделі вирощування та продажу [18]

Метою дослідження є проектування та прототипування експертної системи прийняття рішень в сфері зернотрейдингу з використанням правил. Ця система допоможе фахівцям приймати обґрунтовані рішення з використанням експертних знань та правил при виборі земельних ділянок для інвестицій або виборі певної культури для посіву на наявних угіддях. Така система може допомогти знизити ризики та покращити ефективність прийнятих рішень, що зробить її дуже важливою для фермерів, зернотрейдерів та інших фахівців у цій сфері.

Інформаційні процеси в зернотрейдингу є важливим елементом в економіці та забезпечують безперебійний потік зерна від виробника до споживача. Зернотрейдингові компанії залежать від швидкої та ефективної обробки величезних обсягів інформації, такої як дані про ціни на зерно, інформація про транспортування, складські запаси та інші елементи ланцюга

постачання зерна. До інформаційних процесів зернотрейдингу належить також моніторинг та аналіз урожаю та ринку, що допомагає визначати ціни на зерно та прогнозувати можливі коливання на ринку.

Оскільки зерно - це продукт, який може бути неякісний або зіпсований, інформаційні процеси в зернотрейдингу також повинні забезпечувати контроль якості та безпеки зерна на всіх етапах постачання, від збирання до зберігання та транспортування.

Для успішної роботи в зернотрейдингу необхідна велика кількість точної та надійної інформації. Інформаційні технології можуть значно полегшити обробку та аналіз даних в зернотрейдингу, забезпечуючи ефективність та точність в прийнятті рішень.

Для більш точної характеристики об'єкта дослідження вирішено проаналізувати існуючі наукові праці в даній предметній області.

Стаття "Understanding the potential applications of Artificial Intelligence in Agriculture Sector" [19] опублікована в журналі "Big Data and Cognitive Computing" в 2021 році. У статті автори досліджують застосування штучного інтелекту в сільському господарстві, зокрема, для поліпшення продуктивності, зниження витрат та підвищення якості продукції.

У статті автори проводять огляд літератури та аналізують різні використання штучного інтелекту в сільському господарстві, такі як системи прогнозування урожаю, виявлення хвороб рослин, автоматичне поливання та догляд за рослинами, аналіз даних про ґрунт, розвиток системи підтримки прийняття рішень та багато інших.

У результаті аналізу автори висновують, що штучний інтелект може бути використаний в різних аспектах сільського господарства та він може допомогти у покращенні продуктивності, зменшенні витрат та підвищенні якості продукції. Однак, автори також вказують на деякі проблеми, які пов'язані з використанням

штучного інтелекту в сільському господарстві, такі як висока вартість технології та складнощі з інтеграцією з наявною інфраструктурою.

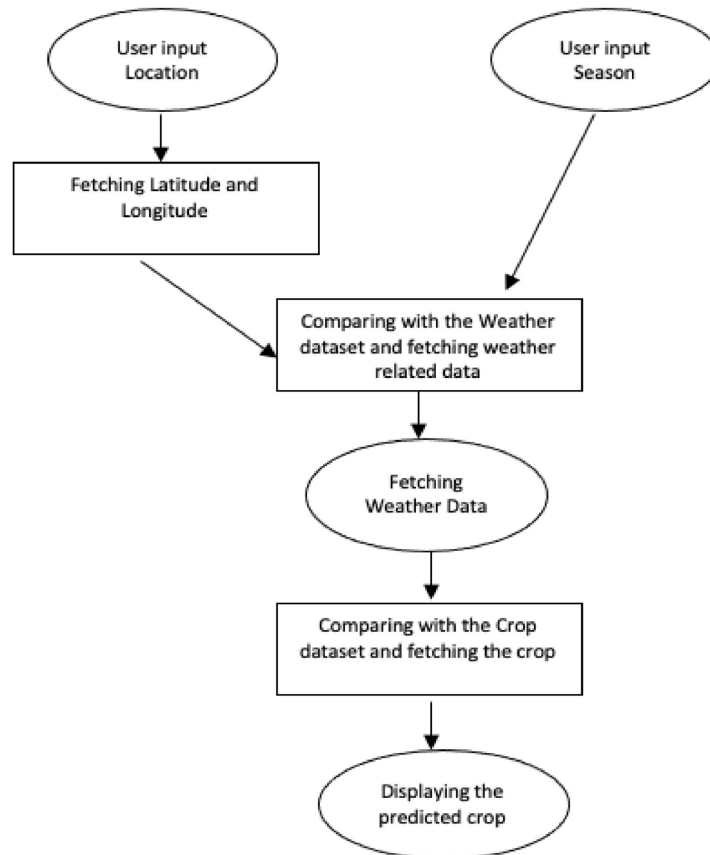


Рисунок 2.2 – Структура системи, що запропонована у статті

У цілому, стаття дає корисний огляд потенційних застосувань штучного інтелекту в сільському господарстві та показує, як ці технології можуть допомогти в покращенні сільського господарства.

Наступна стаття пропонує [20] методологію використання великих даних для аналізу вибору культур для вирощування в різних зонах землі. Автори зазначають, що відповідно до кліматичних умов та особливостей ґрунту, окремі культури можуть бути більш придатними для вирощування в певних регіонах.

Стаття розглядає використання методів обробки даних та машинного навчання для аналізу великої кількості інформації про різні культури та земельні ділянки. Автори розглядають такі критерії як кліматичні умови, родючість ґрунту та його склад, вологість ґрунту, а також історичні дані вирощування різних культур на даній ділянці.

Далі автори використовують алгоритми машинного навчання для класифікації земельних ділянок на підставі зазначених критеріїв та аналізу відповідної інформації про культури. В результаті, система може запропонувати культури, які найбільш підходять для вирощування на конкретній земельній ділянці.

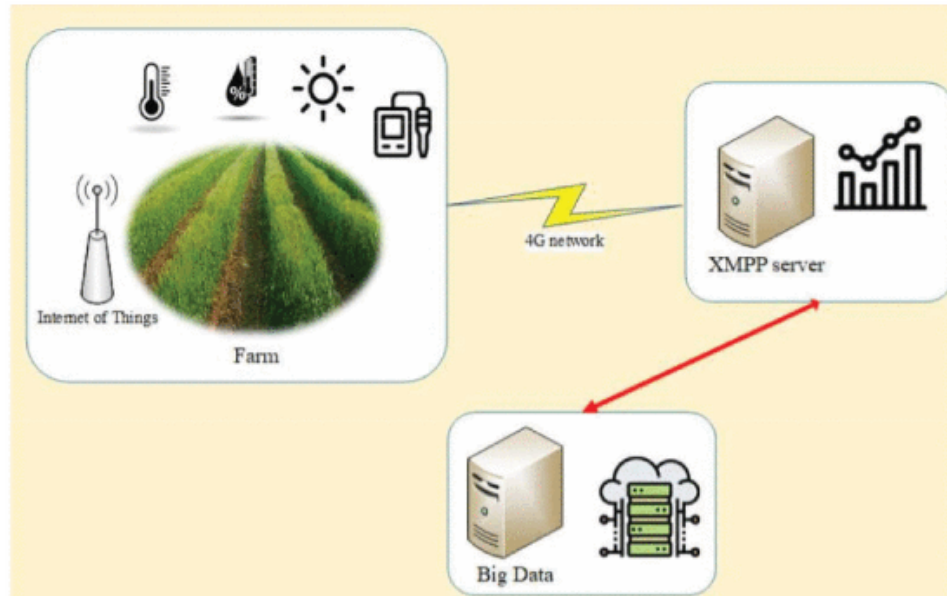


Рисунок 2.2 – Архітектура системи, що описують автори у статті

Стаття має важливість для розвитку сільського господарства та ефективного використання земельних ресурсів, що може допомогти збільшити виробництво продуктів харчування та забезпечити стійкий розвиток галузі.

Отже, проаналізувавши статті, можна сказати що для оптимізації процесів зернотрейдингу, дослідження розробляють інтелектуальні системи прийняття рішень, такі як системи на основі правил та експертні систем, а також системи на основі машинного навчання та нейронних мереж.

У підсумку, складові об'єкту дослідження можна перерахувати наступним списком:

- Дані про ринок зерна, включаючи ціни, попит та пропозицію, тенденції розвитку ринку тощо;

- Дані про стан транспортування та інфраструктуру, включаючи дороги, залізниці, морські порти та склади;
- Дані про законодавство та регулювання в зернотрейдингу;
- Методи аналізу даних та статистичні моделі для прогнозування ринку та визначення оптимальних рішень;
- Експертні знання та досвід зернотрейдерів для оцінки ризиків та прийняття рішень;
- Метеорологічні дані та дані про географічне положення виробничих потужностей господарства;
- Дані про ґрунт, на якому господарство планує вирощувати урожай
- Інформаційні технології для збору, обробки та аналізу даних, включаючи бази даних, програмні засоби та алгоритми машинного навчання;
- Моделі маркетингу та стратегії розвитку бізнесу для досягнення конкурентної переваги в зернотрейдингу.

Ці складові взаємодіють між собою в рамках інтелектуальної системи прийняття рішень, що дозволяє аналізувати дані про ринок та інфраструктуру, прогнозувати тенденції, оцінювати ризики та розвивати стратегії розвитку бізнесу в зернотрейдингу.

Дані про ринок зерна включають в себе інформацію про ціни на зерно на світових ринках, динаміку їх зміни, аналіз попиту та пропозиції на зерно, а також прогнози розвитку ринку на майбутнє.

Ціни на зерно є одним з найважливіших показників на ринку, оскільки вони впливають на вартість зерна для виробників, переробників та споживачів. Дані про ціни на зерно включають інформацію про ціни на основні сорти зерна, такі як пшениця, кукурудза, соя, ячмінь тощо, а також про ціни на фактори, що впливають на ринок зерна, наприклад, ціни на нафту, курс валют тощо.

Попит на зерно відображає, скільки товару потрібно для задоволення потреб споживачів, таких як виробники кормів, млини, виробники алкогольних напоїв. Дані про попит дозволяють зрозуміти потреби споживачів та спрогнозувати зміни на ринку.

Пропозиція на зерно відображає обсяги, які можуть бути запропоновані на ринку від виробників, переробників або експортерів. Дані дозволяють зрозуміти обсяги, які можуть бути доступні для продажу, та спрогнозувати майбутні зміни пропозиції на ринку, що безпосередньо вплинуть на ціну.

Законодавство та регулювання є важливим аспектом зернотрейдингу, який регулюється як на міжнародному, так і на національному рівнях. Вони включають нормативні документи, правила, стандарти, процедури та механізми, які регулюють збір, перевезення, зберігання, обробку та торгівлю зерном.

На національному рівні законодавство регулюється міністерствами аграрної політики, економіки, транспорту та іншими відомствами, які здійснюють контроль та нагляд за виробництвом, перевезенням та торгівлею зерном. Також існують національні стандарти, які регулюють виробництво, якість та безпеку.

У кожній країні законодавство та регулювання можуть відрізнитися, тому для здійснення зернотрейдингу необхідно вивчити національне законодавство в країнах, з якими планується співпрацювати.

Методи аналізу даних та статистичні моделі грають важливу роль в зернотрейдингу для прогнозування ринку та визначення оптимальних рішень. Один з популярних методів - це аналіз трендів, що дає можливість встановити тенденції на ринку зерна та прогнозувати подальший його розвиток. Крім того, статистичні моделі дозволяють визначити різні фактори, що впливають на ринок зерна, та їх взаємозв'язки.

До інших методів аналізу даних можна віднести кластерний аналіз, який дозволяє групувати дані на основі схожості характеристик та визначати

найбільш підходящі ринки для збуту зерна. Також широко використовуються методи машинного навчання, які дозволяють створювати моделі, що прогнозують попит та ціни на зерно в залежності від різних факторів, таких як зміни клімату, економічні та політичні чинники тощо.

Експертні знання та досвід грають важливу роль у прийнятті рішень. Зернотрейдери повинні мати глибокі знання про ринок зерна, ціни, попит та пропозицію, тенденції розвитку ринку, а також про стан транспортування та інфраструктуру, законодавство та регулювання в зернотрейдингу.

Крім того, зернотрейдери повинні мати досвід у роботі з аналізом даних та статистичними моделями для прогнозування ринку та визначення оптимальних рішень. Це допомагає їм визначати потреби ринку та прогнозувати зміни в цінах, що дозволяє їм приймати рішення щодо закупівлі та продажу зерна.

Інформаційні технології в зернотрейдингу включають в себе широкий спектр інструментів для збору, обробки та аналізу даних. Одним з найважливіших засобів є бази даних, які зберігають інформацію про ціни на зерно, обсяги поставок, попит та пропозицію на ринку, а також інші релевантні дані.

Для обробки та аналізу цих даних використовуються різноманітні програмні засоби, такі як Excel, SPSS, SAS, R або експертні системи. Деякі з них мають вбудовані статистичні алгоритми для прогнозування ринку та визначення оптимальних рішень. Для забезпечення точності та ефективності аналізу, використовуються також алгоритми машинного навчання, які можуть самостійно визначати залежності між різними параметрами та зробити прогноз розвитку ринку.

Метеорологічні дані є важливою складовою для аналізу вирощування різних культур, оскільки погода може вплинути на їхній ріст та врожайність. Методані включають температуру повітря, вітряні умови, опади, вологість та інші показники. Збирання метеоданих може проводитися за допомогою

метеостанцій, які розміщуються на полях або в інших місцях, де вирощуються культури.

Дані про географічне положення виробничих потужностей господарства також є важливим елементом для аналізу вирощування різних культур. Географічне положення може вплинути на кліматичні умови, ґрунтові властивості та доступність до водних ресурсів, що може визначити успішність вирощування певних культур. Збирання даних про географічне положення може включати в себе використання GPS-координат, аерофотознімків та інших геоданих. Ці дані можуть бути оброблені та аналізовані з метою визначення оптимальних умов для вирощування різних культур.

Можна зробити висновок про те, що об'єкт дослідження складається зі складних інформаційних процесів, що взаємодіють між собою, а саме дослідження полягає в розробці та використанні інтелектуальної системи для вибору оптимального врожаю для вирощування з використанням даних з метеорологічних джерел та географічного положення виробничих потужностей господарства.

2.2 Структура і характеристика експертної системи

Під час виконання дипломного проекту було обрано підхід з використанням експертної системи. Експертні системи є програмними засобами, які використовують знання та інформацію для розв'язання складних задач. В контексті зернотрейдингу, експертна система може допомогти аналізувати ринок, прогнозувати ціни, визначати оптимальні рішення та оцінювати ризики.

Структура експертних систем відображає розуміння інженерами того, як організувати знання та забезпечити підтримку прийняття рішень за допомогою комп'ютерної системи. Комплексні рішення вимагають поєднання фактичних та евристичних знань. Щоб комп'ютер міг ефективно використовувати евристичний підхід, знання мають бути в легко доступному форматі, що розрізняє дані та

структури керування. Тому експертну систему варто організувати на трьох рівнях, що дозволяє досягти оптимальної ефективності та якості роботи системи:

- База знань: складається з правил вирішення проблем, процедур та внутрішніх даних, які відповідають області проблеми. База знань також містить факти які експерт з відповідної області проблем може використовувати для вирішення проблем. Знання зазвичай зберігаються у форматі if-then правил. Для даної системи було вирішено використовувати базу даних PostgreSQL як основу для зберігання знань, що потрібні для функціонування експертної системи;

- робоча пам'ять: відноситься до специфічних для задачі даних, що розглядаються. Це динамічний модуль системи - база даних. Загалом, робоча область містить набір, який називається базою правил, тобто вона містить набір правил, які має використовувати система в даний момент. Базу правил в експертній системі теж вирішено розмістити в базі даних PostgreSQL;

- механізм формування висновків: загальний керуючий механізм, який застосовує аксіоматичні знання в базі знань до специфічних даних задачі для отримання відповіді або висновку.

База знань експертної системи може бути конкретною діагностичною базою даних, складеною консалтинговою фірмою, а дані про проблему можуть бути надані кінцевими користувачами. База знань є ядром структури експертної системи, що створюється інженерами, які перетворюють знання експертів в предметній галузі на правила та стратегії. Ці правила та стратегії можуть змінюватися залежно від поточної проблемної ситуації. База знань забезпечує експертну систему можливістю рекомендувати напрямки дослідження для користувача. Загальна структура експертної системи наведена на наступному рисунку.

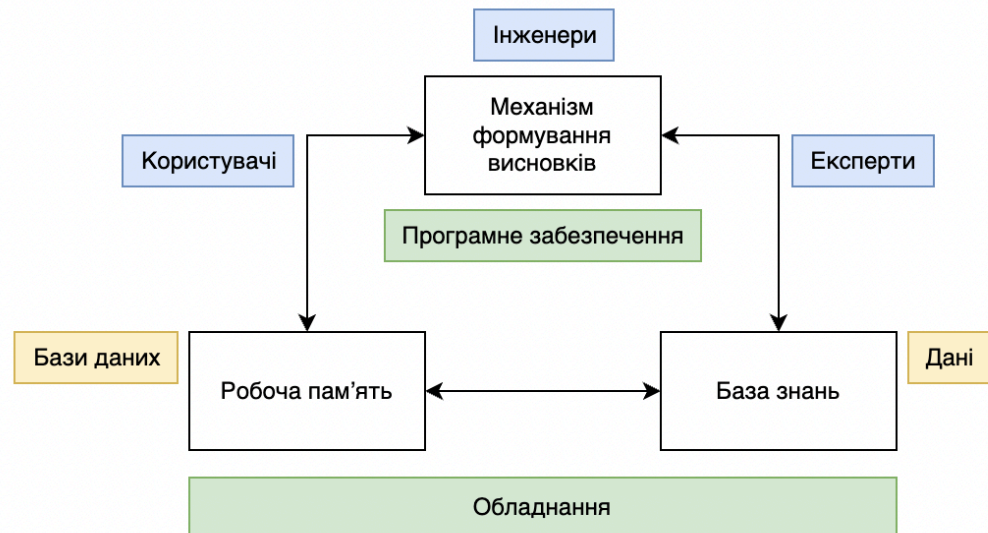


Рисунок 2.3 – Загальна структура експертної системи

Було прийняте рішення зберігати дані експертної системи у форматі JSON (JavaScript Object Notation). Це формат обміну даними, який використовується для передачі структурованих даних між різними додатками та системами, що базується на синтаксисі JavaScript і складається з набору пар ключ-значення, що групуються в об'єкти та масиви. JSON є легким, простим та зручним для використання форматом, який може бути використаний для обміну даними між серверами та клієнтськими додатками, для збереження даних та конфігураційних параметрів. JSON підтримується більшістю мов програмування, що робить його універсальним інструментом для обробки та передачі даних між різними системами.

```

{
  "conditions": {
    "all": [
      {
        "fact": "Soil Type",
        "operator": "equal",
        "value": "Sandy Soils"
      }
    ]
  },
  "event": {
    "type": "Bad soil for any crop"
  }
},

```

Рисунок 2.4 – Приклад JSON об'єкта

Однією з важливих відмінних рис експертних систем від звичайних комп'ютерних програм є модульність. Це досягається за допомогою трьох компонентів, які можуть походити з різних джерел. Для того, щоб експертна система була ефективною, вона повинна навчатися по відгукам користувачів та змінюватися відповідно до контексту проблеми.

Найбільш поширеною формою архітектури, яка використовується в експертних та інших типах систем, що базуються на знаннях, є продукційна система або система, що базується на правилах. Цей тип систем використовує знання, закодовані у формі продукційних правил, тобто правил типу "якщо-то". Правило має умовну частину на лівій стороні та висновок або дію по праву сторону.

Кожне правило представляє собою маленький шматок знань для експертної галузі. Архітектура системи експертних систем на основі правил складається з експерта в предметній області, інженера знань, інференційного рушія, робочої пам'яті, бази знань, зовнішніх інтерфейсів, інтерфейсу користувача, модуля пояснень, бази даних, електронних таблиць і виконуваних програм. Приклад структури експертної системи на основі правил показаний на рисунку.

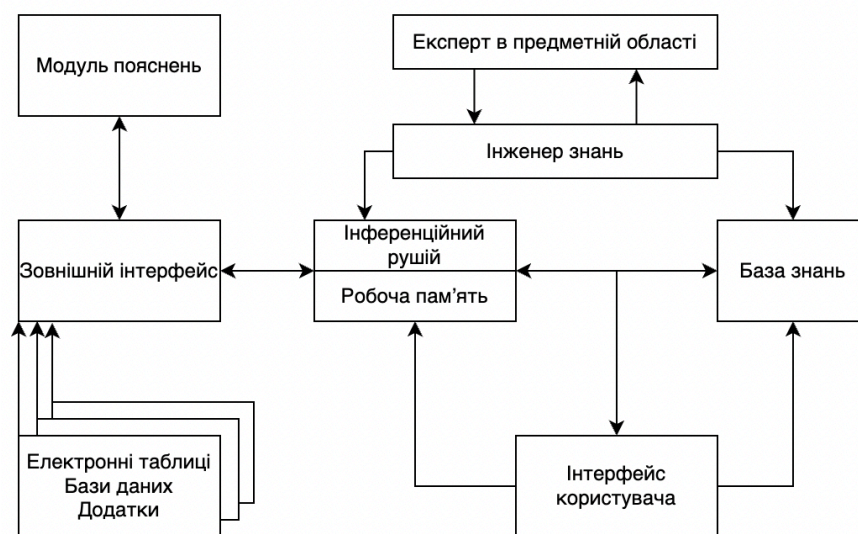


Рисунок 2.5 – Структура експертної системи на основі правил

Компоненти архітектури експертної системи на основі правил включають інтерфейс користувача, базу знань, базу правил та інференційний рушій.

Інтерфейс користувача: це механізм, за допомогою якого користувач і експертна система взаємодіють між собою. Цей модуль приймає запити користувача та надсилає їх експертній системі. Зазвичай користувачі звертаються до експертної системи з таких причин:

- для отримання відповіді на свої запити;
- для отримання шляху вирішення поставленої проблеми;
- для отримання пояснення щодо свого рішення для психологічного задоволення.

Модуль інтерфейсу користувача має бути розроблений таким чином, що на рівні користувача він приймає запит мовою, зрозумілою для експертної системи. Щоб зробити експертну систему зручною для користувачів, інтерфейс користувача повинен взаємодіяти з користувачем природною мовою. Інтерфейс користувача надає якомога більше можливостей, таких як меню вибору, графічні інтерфейси, щоб зробити діалог з користувачем зручним та привабливим.

Для максимальної зручності в експертній системі зернотрейдингу вирішено додати можливість перегляду правил у вигляді дерева. Дерево правил (Rule tree) - це структура даних, яка використовується для відображення взаємозв'язку правил, які використовуються в експертній системі. Дерево складається з вузлів, які представляють правила, та гілок, які показують, які правила застосовуються в яких умовах. Кожен вузол містить логічну умову та список підправил, які будуть застосовані, якщо умова вірна. Дерево правил може бути дуже корисним інструментом для відладки та оптимізації експертної системи, а також для візуалізації процесу вирішення задачі екпертом. В експертній системі зернотрейдингу додавання можливості перегляду правил у

вигляді дерева дозволить експертам швидко зорієнтуватися в системі та знайти найбільш оптимальний шлях для вирішення задачі.

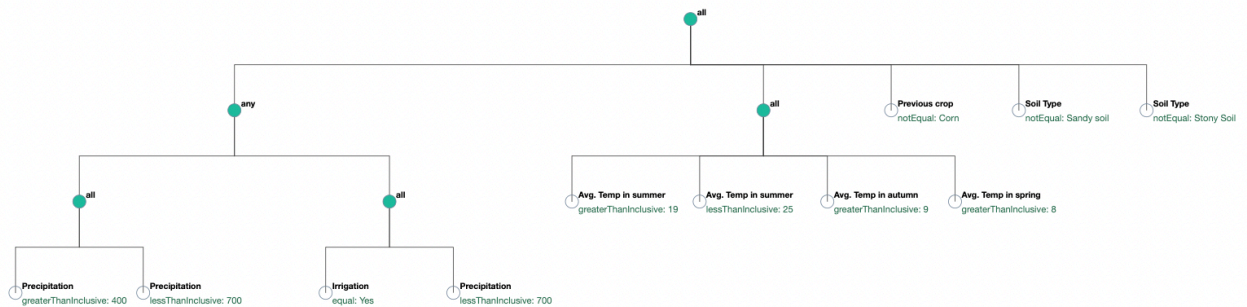


Рисунок 2.6 – Прототип перегляду правил у вигляді дерева

В архітектурі експертної системи, що базується на правилах, база знань - це набір правил. Пари умовних дій представлені у вигляді правил, де передумови правил відповідають умові, а висновок відповідає дії. Дані про конкретний випадок зберігаються в робочій пам'яті. Основною частиною експертної системи є база знань, тому експертну систему також називають системою, заснованою на знаннях. Знання експертної системи зазвичай структуровані у формі дерева, яке складається з кореневого фрейму та кількох підфреймів. Проста база знань може мати лише один фрейм, тобто кореневий фрейм, тоді як велика і складна база знань може бути структурована на основі кількох фреймів.

Інференційний рушій приймає запити користувача та відповіді на питання через інтерфейс вводу/виводу. Він використовує динамічну інформацію разом зі статичними знаннями, збереженими в базі знань. Знання в базі використовуються для отримання висновків про поточний випадок, з яким користувач звертається до системи. Інференційний рушій - це модуль, який знаходить відповідь у базі знань. Він застосовує ці знання, щоб знайти розв'язок проблеми. Зазвичай інференційний процес здійснюється рекурсивно в 3 етапи: збіг, вибір та виконання. Під час етапу збігу зміст робочої пам'яті порівнюється з фактами та правилами, що містяться в базі знань. Коли знайдено відповідні збіги, відповідні правила поміщаються у набір.

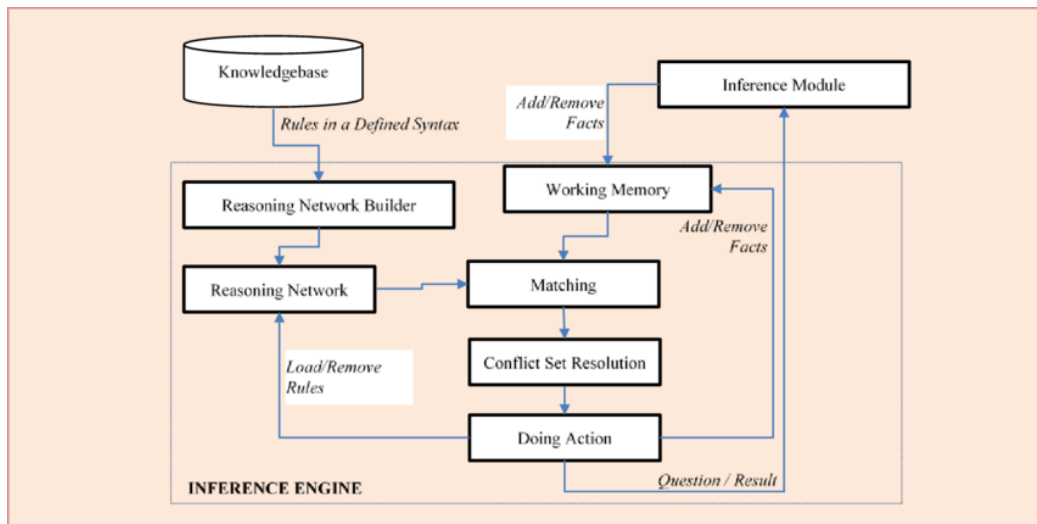


Рисунок 2.7 – Типова архітектура інференційного рушія [21]

В даному проекті буде застосовано бібліотеку JSON Rule Engine. Це бібліотека для виконання бізнес-правил на базі JSON-даних, яка дозволяє зосередитися на логіці в проекті. За допомогою нього можна визначити правила з використанням зрозумілих для предметної області термінів та побудувати відносно прості та ефективні правила.

Бібліотека дозволяє використовувати правила з різноманітних джерел, таких як файлові системи, бази даних або API, забезпечуючи гнучкість та простоту інтеграції. Крім того, вона надає зручний інтерфейс для визначення бізнес-правил, що дозволяє ефективно працювати з ними.

JSON Rule Engine базується на використанні JSON-структур для опису бізнес-правил. Кожне правило містить умову та відповідну дію, яку необхідно виконати, якщо умова виконується. Бібліотека підтримує різноманітні типи даних, такі як числа, рядки, масиви та об'єкти, що дозволяє створювати складні правила.

Загалом, JSON Rule Engine є потужним інструментом для визначення бізнес-правил, який дозволяє покращити ефективність та гнучкість бізнес-процесів в проектах.

Системи на основі правил є потужним інструментом для формалізації знань в проектуванні та реалізації систем на основі знань в області штучного інтелекту та інженерії знань. Вони також надають універсальну програмну парадигму в галузях, таких як моніторинг систем, інтелектуальне управління, прийняття рішень, класифікація ситуацій, діагностика систем та кодування операційних знань.

Хоча парадигма програмування на основі правил здається відносно концептуально простою, проектування та реалізація системи, що працює на основі правил, є складним та втомлюючим завданням. Проблеми виникають, коли істотно збільшується кількість правил. Складно зберігати правила консистентними, охоплювати всі варіанти операцій та забезпечувати роботу системи згідно з бажаним алгоритмом. Добре визначена система повинна бути безпечною, надійною та ефективною. Ці характеристики далі перетворюються на набір точно визначених рис (наприклад, повнота, детермінованість, правильність), які можуть бути перевірені у формальному вигляді.

Потужність експертної системи полягає в знаннях, якими вона володіє, більшу увагу при розробці експертної системи слід приділяти частині отримання знань.

У книзі про створення експертних систем *Fundamentals of expert systems* [22], процес побудови описується як процес отримання знань. Основне правило полягає в тому, що здобуття знань та створення експертної системи - це взаємодіючі процеси, які не можна розділяти. Інші основні правила для процесу створення експертних систем також є важливими. Потрібно ознайомитись з предметною областю та зібрати необхідну інформацію від експерта, а також якомога швидше розпочати побудову прототипу системи. Важливо звернути увагу на те, що процес збирання знань та створення експертної системи повинен відбуватися при взаємодії між експертом та розробником.

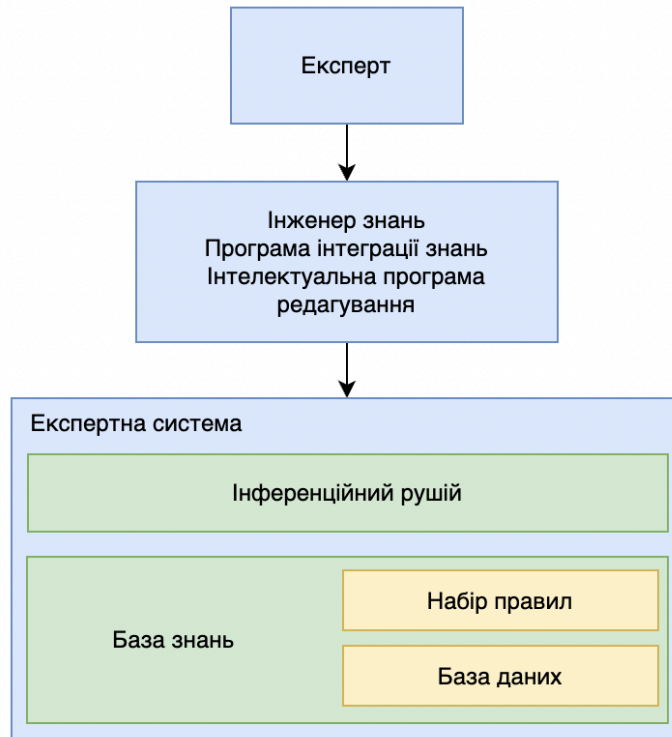


Рисунок 2.8 – Ієрархія взаємодії експертів, інженерів та системи

Якщо дані можна визначити як колекцію дискретних, неорганізованих фактів або спостережень, то інформація буде відноситися до оброблених даних, зібраних в змістовному контексті, який забезпечує додаткову цінність. Знання будуть відноситися до інформації з контекстом, готової до використання для прийняття рішень. Знання можуть бути колекцією правил, керівних принципів, подій та мір, що є контекстуальними для проблеми. За допомогою знань, фрагменти даних та інформації можуть бути зібрані, організовані та представлені як рішення для вирішення проблеми в руках. Відносини між даними, інформацією та знаннями можуть бути зображені у візуальній формі (Рисунок 2.4). Ця взаємодія вперше була запропонована в літературі як ієрархічна модель Дані – Інформація – Знання – Мудрість.



Рисунок 2.9 – Ієрархічна модель Дані – Інформація – Знання – Мудрість

Як показано на Рисунку 2.9, дані утворюють фундаментальну основу для отримання інформації, знань та мудрості. Комп'ютери корисні для обробки даних та інформації та знань до певної міри, тоді як досягнення або реалізація мудрості все ще є компетенцією людей. Системи управління базами даних у свою чергу спеціалізуються на управлінні та зберіганні даних. Інформаційні системи корисні для класифікації та організації даних в зв'язному порядку. Знання можна отримати зібравши різні шматочки інформації зі збіркою правил і процедур, які мають схожий контекст. Експертні системи корисні для збору, зберігання, обробки та демонстрації цього знання. У найнижчому ряді піраміди залишаються дані, які можна програмувати алгоритмічно. У вершині піраміди знаходяться знання та мудрість, які мають неалгоритмічний та евристичний характер.

2.3 Методи, моделі й моделювання процесів і елементів складних систем

У цьому розділі коротко описана методологія, що використовується для розробки експертної системи. Процес оцінки землі починається з широкої оцінки ресурсів землі на регіональному рівні і переходить до вибору оптимальних

культур, які можуть бути вирощені на сільськогосподарському угідді. Оскільки оцінка землі - це багатодисциплінарна проблема, вона потребує різних видів знань на різних масштабах. Регіональна оцінка землі в основному ґрунтується на даних про ґрунти та фізіографію, клімат і доступність води та дає приблизну індикацію можливих культур, які можна вирощувати в регіоні. Господарський рівень прийняття рішень вимагає участі власника землі і є інтерактивним процесом, в якому фермери та зернотрейдери можуть обирати або виключати певні фактори, які будуть використовуватися при оцінці культур для їх господарських угідь. Такі фактори включають економічні дані (інформацію про ринок), клімат (середню кількість опадів на господарському угідді), фізіографію ґрунту, доступність води (для іригованих культур), соціальні аспекти та регулювання уряду.

Слід наголосити на тому, що модель оцінки землі на основі знань може більш ефективно враховувати якісні, кількісні та евристичні фактори оцінки порівняно з іншими підходами. Щоб довести, що цей підхід на основі знань працює, було розроблено прототипову модель концепції доведення. Типи факторів, які використовуються в процесі оцінки землі, обираються таким чином, що деякі з них є якісними та класифікуються на відносній шкалі (низька, середня та висока), тоді як інші є кількісними та числово кількуються (кількість опадів, ціна на ринку). Інші є евристичними, тобто базуються на людських судженнях, наприклад, на рівні підтримки від уряду для культури.

Метою розробки експертної системи, що базується на знаннях, є надання допомоги зернотрейдинговим компаніям у виборі оптимальної культури, яку можна вирощувати на їх землі.

Спеціалісти з геодезії та землеустрою використовують дані про ґрунти та фізіографію, щоб створювати карти придатності різних культур для регіону на масштабі картографування. Ці карти відіграють важливу роль у стратегічному плануванні земельного використання на регіональному рівні та у прийнятті рішень про землекористування на рівні регіону. Такі регіональні плани

допомагають аграрним підприємствам розробляти свої плани землекористування на рівні всього господарства.

Кожна фермерська ділянка оцінюється на основі ряду фізичних факторів, таких як топографія, водозбереження, солоність, умови дренажу, глибина ґрунту, хімічні бар'єри для росту коренів, родючість ґрунту та потенціал ерозії, для класифікації землі за придатністю для вирощування певної культури. Землю класифікують за однією з восьми категорій придатності, від дуже високої до не придатної, як показано в Таблиці 2.1. Наприклад, в ArcGIS створюється карта "потенційно придатних земель для вирощування пшениці" (див. Рисунок 2.10), на якій земельні ділянки показані вісьмома категоріями (від високої до непридатної), окремо зазначено перешкоди, типу водойм або забудови.

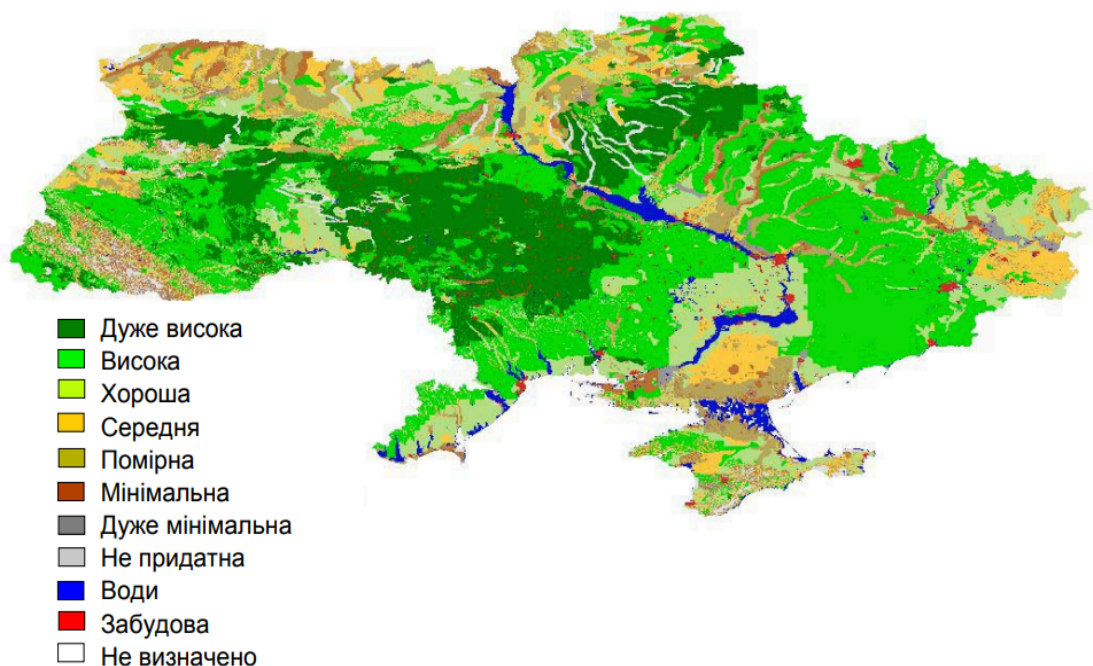


Рисунок 2.10 – Потенційно придатні землі для вирощування пшениці [23]

На основі цих даних можна побудувати правила, варто класифікувати кожен тип землі за рівнем придатності, щоб мати змогу перетворити логічні судження в формат, зрозумілий експертній системі.

Таблиця 2.1 – Класифікація земель за рівнем придатності

Клас	Врожайність	Визначення
Клас 1	Дуже висока	Земля з високим потенціалом продуктивності і не потребує більш ніж стандартної практики управління для підтримки продуктивності.
Клас 2	Висока	Земля з високим потенціалом продуктивності і не потребує більш ніж стандартної практики управління для підтримки продуктивності.
Клас 3	Хороша	Земля з помірно високим потенціалом продуктивності та / або потребує специфічних, але широко використовуваних і прийнятих методів управління для підтримки продуктивності.
Клас 4	Середня	Земля з помірно високим потенціалом продуктивності та / або потребує специфічних, але широко використовуваних і прийнятих методів управління для підтримки продуктивності.
Клас 5	Помірна	Земля з помірним потенціалом продуктивності та / або потребує спеціалізованих методів управління для збереження продуктивності.
Клас 6	Мінімальна	Земля з низьким потенціалом продуктивності і / або потребує дуже високоспеціалізованих управлінських навичок для підтримки продуктивності.
Клас 7	Дуже мінімальна	Земля з низьким потенціалом продуктивності та / або постійними обмеженнями, які фактично перешкоджають її використанню.
Клас 8	Не придатна	Земля не придатна для вирощування сільськогосподарських культур

Для отримання географічних даних про земельні ділянки можна скористатися ArcGIS [24]. Це комплексне програмне забезпечення для геоінформаційного аналізу, картографії та візуалізації геоданих. Це платформа, яка включає в себе різноманітні інструменти для збору, організації, аналізу та відображення геоданих.

Основною метою ArcGIS є надання користувачам можливості здійснювати комплексний аналіз геоданих інтерактивними способами, що дозволяє ефективно вирішувати різноманітні завдання в області геопросторового аналізу. До складу платформи входять різні модулі та інструменти, які можуть бути

використані для створення та виконання геопросторових аналітичних завдань, а також для відображення результатів аналізу у вигляді карт.

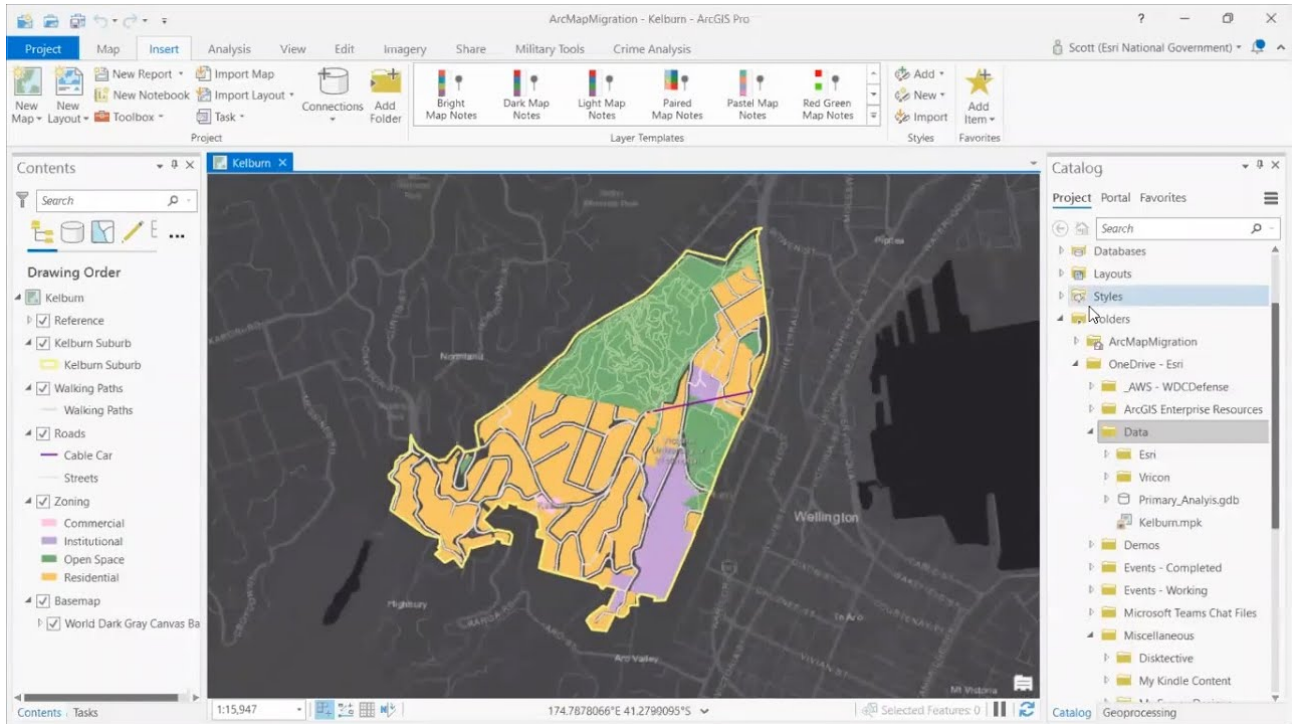


Рисунок 2.11 – Інтерфейс ArcGIS

ArcGIS також підтримує створення та використання власних додатків на основі API, що дає можливість користувачам розширювати можливості платформи та розробляти власні геоінформаційні рішення.

В цілому, ArcGIS є потужним інструментом для роботи з геоданими та геопросторовим аналізом, який дозволяє зробити висновки на основі великого обсягу даних та відобразити їх у вигляді карт та візуалізацій.

Головною метою ArcGIS карт оцінки придатності землі, розроблених різними інститутами та науковими колективами, є їх використання у регіональному плануванні землевикористання та стратегічному прийнятті рішень. Картографування мають різний масштаб, що означає, що вони в деяких випадках мало корисні для процесів прийняття рішень на рівні господарства. Однак, оскільки немає іншої просторової інформації, доступної на масштабі картографування, ближчому до рівня господарства, дані масштабу 1:100 000 також використовуються в поточних дослідженнях. Експертна система на основі

знань розширює можливості існуючих картографічних досліджень, щоб змогти запропонувати підходящі культури для вирощування на земельній ділянці.

Основна мета програми – вилучити недійсні культури та відобразити найбільш придатні для кожної земельної ділянки. Першим кроком процесу є об'єднання окремих оцінок земель в картах GIS, щоб видобути базу даних придатності культур. Наступний крок - ранжування вибору культур за ступенем придатності. Результати бази даних будуть містити для кожної земельної ділянки відповідні придатні культури, розміщені в порядку спадання врожайності: найбільш придатна культура у верхній частині, а найменш придатна - у нижній частині. Пізніше, з доповненням інформації від власників землі або експертів, система має звужуватися до найбільш підходящої (оптимальної) культури, яку можна вирощувати на цій ділянці. До експертної інформації відносяться постачання води (рівень опадів або зрошення), робоча сила (низька, середня або висока), наявність сільськогосподарської техніки для ділянки, що розглядається. Власник землі або зацікавлена особа також має можливість включити в оціночний процес фактори, такі як державні регулювання та ринкові чинники (ціна та попит).

Проблеми управління землею, такі як оцінка земель, планування використання землі та вибір місць потребують знань з багатьох дисциплін. Часто ці проблеми є складними, оскільки фактори, пов'язані з ними та їх взаємовідносини, важко виміряти та моделювати. Експерти, які вирішують такі проблеми, можуть потребувати врахування неповних, неточних або розмитих наборів даних. Тому ці проблеми потребують гнучкого, заснованого на знаннях, евристичного підходу, на відміну від алгоритмічних, жорстких методів вирішення проблем. Подвійною метою розробки системи на основі знань є здобуття людських знань про предметну область проблеми та використання їх для імітації людського підходу до вирішення проблеми.

Головним чином, база знань для експертної системи містить умовні вимоги кожного вибору культури та те, як фактор вирішує придатність культури для

фермерської ділянки. Умовні вимоги стосуються вимог води для культури - середньорічний опад або зрошувальна вода, витрати на робочу силу - низькі, середні або високі, потреба машин для обробки культури - потрібна або не потрібна, ринкові фактори - ціна на ринку, попит на культуру - сприятлива або несприятлива. Значення для кожного фактору є лише приблизним, оскільки у деяких випадках фактор може мати більше, ніж дві-три категорії. Наприклад, вимоги до механізації можуть бути класифіковані за низькими, середніми та високими вимогами замість класів “необхідні” чи “не обов’язкові”. Оскільки дане дослідження передбачало створення прототипної моделі, використовувалися прості класи. Ці правила знань визначені в Таблиці 2.2. Перевагою табличного формату є те, що дуже легко перевірити, змінити та оновити базу знань.

Таблиця 2.2 – Умовні вимоги для вибору культури [25]

Культура	Ціна usd/t	Опади	Трудові ресурси	Застосування техніки	Рівень підтримки уряду	Ринковий індекс	Нормалізована ринкова ціна	Попит
Сочевиця	513	300 - 500	Середньо	Ні	Ні	46.63	93.27	Високо
Ячмінь	208	350 - 500	Низько	Так	Так	12.47	37.81	Середньо
Соя	550	350 - 550	Середньо	Ні	Так	33	100	Середньо
Кукурудза	300	400- 700	Низько	Так	Так	27.27	54.54	Високо
Овес	210	350 - 600	Низько	Так	Ні	12.59	38.18	Середньо
Просо	305	350 - 500	Низько	Так	Ні	27.72	55.45	Високо
Пшениця	320	350 - 600	Низько	Так	Так	19.19	58.18	Середньо

Дуже важливо зрозуміти, що база знань складається лише з діапазону основних факторів. Обсяг та різноманітність факторів обмежена часовими та фінансовими факторами, але, тим не менш, вони є відповідними. Методи, що використовувалися для отримання цих факторів, наведені нижче.

Для визначення потреби у воді культури класифікуються на дощові або зрошувані. У випадку дощових культур кожна культура має мінімальне та

максимальне значення середньорічного опаду (у міліметрах), необхідного для оптимального врожаю. Залежно від середньорічного опаду, який отримує господарська ділянка, культури, які не є доцільними для даної господарської ділянки, виключаються з варіантів вибору культур. Поточна ринкова ціна культури та її попит на ринку формують ринковий індекс, який є кількісною мірою ринкової цінності культури. Ринкові ціни на культури коливаються від 208 до 550 доларів за тонну, тому використовується відносна ринкова ціна.

Нормалізована ринкова ціна = [Актуальна ціна врожаю ÷ Найвища ціна врожаю] × 100. Тобто нормалізована ціна на сочевицю дорівнює $[513 \div 550] \times 100 = 93,27$

Для розрахунку ринкового індексу нормалізована ціна множиться на ринковий попит, тобто Ринковий індекс = нормалізована ринкова ціна × ринковий попит.

Ринковий попит поділяється на одну з трьох категорій: "Низький" (0,17), "Середній" (0,33) або "Високий" (0,5), і кожній категорії призначається відповідне числове значення. Щоб забезпечити більшу перевагу культурам, які користуються великим попитом (тому більш прибуткові), вводиться фактор попиту. Значення отримується шляхом пропорційного представлення трьох категорій попиту, що відповідає загальній вартості 100. Значення 100 пропорційно діляться на три класи, 50%, 33% і 17% представляють відповідно категорії Високий, Середній і Низький попит на ринку. Коли вони перетворюються на еквівалент 1, вони представляють 0,5, 0,33 і 0,17.

Найвище значення ринкового індексу може становити 50 (якщо розглядається нормалізована ринкова ціна 100 і 0,5 для «високого» ринкового попиту), і, отже, ринковий індекс може змінюватися від 1 до 50.

Тобто якщо дані з Таблиці 3.2 перевести в формат даних JSON, отримаємо наступний об'єкт:

```

"conditions": {
  "all": [
    {
      "any": [
        {
          "all": [
            {
              "fact": "Precipitation",
              "operator": "greaterThanInclusive",
              "value": 350
            },
            {
              "fact": "Precipitation",
              "operator": "lessThanInclusive",
              "value": 600
            }
          ]
        },
        {
          "all": [
            {
              "fact": "Irrigation",
              "operator": "equal",
              "value": "Yes"
            },
            {
              "fact": "Precipitation",
              "operator": "lessThanInclusive",
              "value": 600
            }
          ]
        }
      ]
    },
    {
      "all": [
        {
          "fact": "Avg. Temp in summer",
          "operator": "greaterThanInclusive",
          "value": 18
        },
        {
          "fact": "Avg. Temp in summer",
          "operator": "lessThanInclusive",
          "value": 26
        },
        {
          "fact": "Avg. Temp in autumn",
          "operator": "greaterThanInclusive",
          "value": 8
        },
        {
          "fact": "Avg. Temp in spring",
          "operator": "greaterThanInclusive",
          "value": 7
        }
      ]
    },
    {
      "fact": "Previous crop",
      "operator": "notEqual",
      "value": "Wheat"
    },
    {
      "fact": "Soil Type",
      "operator": "notEqual",
      "value": "Sandy soil"
    }
  ]
}

```

Рисунок 2.12 – Фрагмент JSON-об'єкту

Перетворення табличних даних про урожай в дані в JSON об'єктах можна виконати за допомогою програмного забезпечення для роботи з даними, таких як Excel або Python Pandas. Спочатку, дані про урожай збираються в табличному форматі, наприклад у вигляді CSV файлу. Після цього, за допомогою програмного забезпечення можна прочитати ці дані та перетворити їх у JSON об'єкти.

JSON (або JavaScript Object Notation) є легким та зручним форматом для обміну даними. Він складається з пар "ключ: значення" і може містити об'єкти, масиви, числа, рядки, логічні значення та значення null. Для перетворення табличних даних про урожай в JSON об'єкти, потрібно відповідним чином створити пари "ключ: значення" для кожного рядка даних та додати їх у JSON об'єкт.

Як видно на рисунку 2.12, умови системи для вибору урожаю побудовані у вигляді об'єктів, які складають собою деревовидну структуру. Оператор "all" означає що для позитивного результату має бути збіг по всім заданим фактам. Оператор "any" допускає хоча б один збіг для позитивного результату. Факти записуються кожен в окремі об'єкти та розміщуються на потрібних ієрархічних рівнях. Об'єкт факту складається з назви, оператора (наприклад, "lessThanInclusive" дорівнює математичному оператору \leq) та значення. З подібних об'єктів створюється структура та в цілому являє собою правило експертної системи. Такі умови були побудовані по кожній культурі відповідно до Таблиці 3.2.

У даному розділі було детально описано основні технічні характеристики експертної системи для зернотрейдингу, а саме - структуру, функції та основні завдання. Було визначено, що основною метою системи є надання експертної підтримки в процесі прийняття рішень щодо посіву зернових культур на земельних ділянках підприємства. Також були розглянуті основні компоненти системи та їх взаємодію.

Постановка задачі включає в себе розробку ефективного механізму автоматизованої підтримки прийняття рішень, що базується на використанні експертних знань про культуру, погодні умови в регіоні, особливості механізації на підприємстві, а також розробку системи моніторингу цін на зернові культури на ринку. Для вирішення цих задач було визначено найбільш ефективні підходи та технології, зокрема, використання експертних систем та баз даних у поєднанні з інструментами візуалізації.

Отже, розробка такої експертної системи може допомогти вирішити проблему вибору оптимальної стратегії інвестицій в земельні ділянки, або допоможе зробити правильний вибір при посіві зернових культур, зменшити витрати та підвищити ефективність бізнесу в цілому.

3 КОНСТРУКТИВНИЙ РОЗДІЛ. РОЗРОБЛЕННЯ ПРОЕКТНИХ РІШЕНЬ

Основною метою поточної експертної системи є запропонувати оптимальну сільськогосподарську культуру, яка найкраще підходить для земельної ділянки на основі її потенціалу. Землю можна оцінити по її придатності для вирощування певної культури на основі кількох параметрів. Ці параметри та ступінь їх впливу на придатність для вирощування культур наведені в другому розділі. У поточному розділі наводиться опис проектування та специфікації експертної системи що допоможе зернотрейдинговій компанії уникнути ризиків при земельних інвестиціях, охоплюючи специфікації системи, використані дані та цільових користувачів. Розділ досліджує, як будується база знань та які програмні інструменти та середовище застосування використовуються у процесі розробки експертної системи. Також детально описується архітектура, на якій ґрунтується експертна система, та які програмні інструменти застосовані в цій архітектурі.

3.1 Моделювання та проектування бази знань для системи прийняття інтелектуальних рішень

База знань є ключовою складовою експертної системи і містить набір правил різних сільськогосподарських культур та їх вимог до ресурсів, що визначає критерії вибору відповідної культури. В рамках цього дослідження, для аналізу придатності вирощування культур, на основі вивчення електронних джерел, були визначені додаткові фактори, такі як соціальні, економічні та політичні фактори, які були внесені до таблиці критеріїв. Для створення бази даних використовуються просторові дані, такі як цифрові кадастрові карти, карти ґрунтів та ландшафтів, дані про опади та фізичні карти придатності вирощування культур. Дані надходять з різних ґрунтових обстежень та методів

дистанційного зондування (спостереження з супутника та повітряної фотозйомки).

Наприклад, дані про опади збираються шляхом щоденних вимірювань на метеостанціях, а база даних ґрунтів підготовлена шляхом інтерпретації зображень, отриманих з дистанційного зондування з обмеженою кількістю спостережень на рівні поля. Просторові бази даних були перетворені на формат бази даних, який легко використовувати для експертної системи. Підготовлено базу даних, яка містить перелік властивостей та правил щодо вибору кожної культури в обраній досліджуваній області з середніми значеннями опадів, які потрібні для вирощення урожаю, ця база даних є вхідним даними для процесу визначення придатності культур.

Соціальні, економічні та політичні фактори, як правило, не є просторовими. Економічні фактори, такі як ціна на ринку та попит на врожай, наявність робочої сили та машин відображені також в базах знань експертної системи. Інформація про ринкові ціни була зібрана з порталу Державної служби статистики.

Ринковий попит є важливим фактором у визначенні економіки попиту та пропозиції певного врожаю. Якщо точно виміряти ринковий попит або передбачити сезонний попит на продукцію врожаю, можна визначити його оптимальну ціну продажу. Якщо попит на ринку високий, ціна на цю культуру буде високою, і навпаки. На даний момент попит на ринку класифікується на три категорії: низький, середній і високий. Політичні фактори, такі як правові обмеження чи субсидії на певні культури є важливими для вибору культури і тому є частиною бази даних. Фактор політики може розглядатися як евристичний, який залежить від розуміння експерта, який вводить цю інформацію в базу знань. Цей тип інформації є дуже динамічним і потребує періодичного оновлення експертом.

Дані та їх джерела наведені в Таблиці 3.1.

Таблиця 3.1 – Дані експертної системи та їх джерела

Дані	Специфікації	Рік публікації	Джерело
Обсяг виробництва, урожайність та зібрана площа сільськогосподарських культур за їх видами	Інформація про збір урожаю у розрізі культур	2021	Державна служба статистики України [26]
Посівні площі озимих культур за їх видами	Інформація про посівні площі	2018	Державна служба статистики України [27]
Площі, валові збори та урожайність сільськогосподарських культур за їх видами та по регіонах	Інформація про урожайність культур у розрізі регіонів	2021	Державна служба статистики України [28]
Аналіз погодних умов в Україні в 2021 році	Інформація про погодні умови в різних регіонах	2021	Метео Фарм [29]
Виробництво зернових та зернобобових культур	Картографічна інформація про виробництво культур	2012	Державна служба статистики України [30]
Індекс обсягу сільськогосподарського виробництва	Картографічна інформація про індекс обсягу с-г. виробництва	2012	Державна служба статистики України [31]

Створення прототипу бази даних та заповнення її JSON об'єктами є важливим етапом в розробці експертної системи. Тому на основі аналізу джерел інформації були підготовлені JSON об'єкти, що містять інформацію про урожайність, погоду, географічні координати та інші характеристики різних культур. В подальшому ці дані можуть бути використані в експертній системі для прийняття рішень щодо вибору оптимального варіанту вирощування певного

виду культури в конкретних географічних умовах. JSON формат є зручним для зберігання даних, оскільки він простий у використанні та може бути легко зчитаний та оброблений комп'ютерними програмами.

```

1  {
2    "name": "",
3    "attributes": [
4      {
5        "name": "",
6        "type": ""
7      }
8    ],
9    "decisions": [
10   {
11     "conditions": {
12       "all": [
13         {
14           "fact": "",
15           "operator": "",
16           "value": ""
17         }
18       ]
19     },
20     "event": {
21       "type": ""
22     }
23   }
24 ]
25 }
26

```

Рисунок 3.1 – Структура JSON об'єкту

Об'єкт, що продемонстровано на рисунку 3.1 є структурою правил для експертної системи. Він складається з трьох основних блоків:

- "name": вказує назву набору правил. В даному випадку це правило для вибору виду зернових культур, то назва може бути "Crop Advisor ruleset";
- "attributes": перелічує всі характеристики об'єкта, які будуть використовуватися для встановлення правил. Кожна характеристика містить ім'я та тип;
- "decisions": визначає умови та події, які будуть відбуватися при заданій комбінації характеристик. Умови містять оператори порівняння та значення, що порівнюються з характеристиками. Події визначають тип події, яка відбудеться, коли всі умови будуть виконані. Наприклад, якщо для правила "вибір сорту зерна" характеристика "врожайність" вище 10 тонн і "стійкість до шкідників" вище середнього, то подія може бути "рекомендація вибрати сорт А".

Процес створення прототипу та заповнення бази даних показаний на рисунках нижче.

```

postgres=# create database knowledgebase
postgres=# ;
CREATE DATABASE
postgres=# create user knowledgeuser
postgres=# ;
CREATE ROLE
postgres=# alter user knowledgeuser with encrypted password '123';
ALTER ROLE
postgres=# grant all privileges on database knowledgebase to knowledgeuser;
GRANT
postgres=# \l

```

List of databases					
Name	Owner	Encoding	Collate	Ctype	Access privileges
knowledgebase	postgres	UTF8	C	C	=Tc/postgres + postgres=CtC/postgres + knowledgeuser=CtC/postgres
logistics_info	k.kalashnikov	UTF8	C	C	=Tc/"k.kalashnikov" + "k.kalashnikov"=CtC/"k.kalashnikov"+ logisticsuseer=CtC/"k.kalashnikov"
postgres	k.kalashnikov	UTF8	C	C	=Tc/"k.kalashnikov" + "k.kalashnikov"=CtC/"k.kalashnikov"+ postgres=CtC/"k.kalashnikov"
template0	k.kalashnikov	UTF8	C	C	=c/"k.kalashnikov" + "k.kalashnikov"=CtC/"k.kalashnikov"
template1	k.kalashnikov	UTF8	C	C	=c/"k.kalashnikov" + "k.kalashnikov"=CtC/"k.kalashnikov"

(5 rows)

```

postgres=# \du

```

List of roles		
Role name	Attributes	Member of
k.kalashnikov	Superuser, Create role, Create DB, Replication, Bypass RLS	{}
kirill		{}
knowledgeuser		{}
logisticsuseer		{}
postgres	Superuser	{}
user1		{}

```

postgres=#

```

Рисунок 3.2 – Створення бази даних

На Рисунку 3.1 продемонстроване створення бази даних під назвою **knowledgebase**. Після створення БД командою *create user* створюється користувач **knowledgeuser**, задається пароль. Користувачу надаються всі права на створену базу даних.

Успішний результат вищенаведених команд можемо спостерігати в таблиці, що містить список всіх локальних баз даних. Також ще нижче виведений список всіх користувачів, де можна побачити в тому числі щойноствореного **knowledgeuser**.

Для зручності роботи з базою даних можна використати графічний інтерфейс DBeaver [32]. DBeaver - це універсальний інструмент для роботи з

базами даних, який дозволяє зручно підключатися та працювати з різними типами баз даних, включаючи MySQL, PostgreSQL, Oracle, SQLite, SQL Server.

Одна з головних переваг DBeaver - це його багатofункціональність, яка дозволяє працювати з базами даних будь-якої складності. Інструмент має розширені можливості для створення запитів, моделювання схеми баз даних та забезпечення безпеки даних.

Dbeaver має зручний інтерфейс користувача з підтримкою графічних елементів для відображення баз даних та їх залежностей. Також він дозволяє працювати з керуванням версіями, експортувати та імпортувати дані з баз даних у різних форматах.

Крім того, DBeaver може бути використаний для роботи з базами даних в різних операційних системах, включаючи Windows, macOS та Linux. Інструмент є відкритим програмним забезпеченням з відкритим вихідним кодом, що дозволяє розширювати його можливості за допомогою додаткових плагінів.

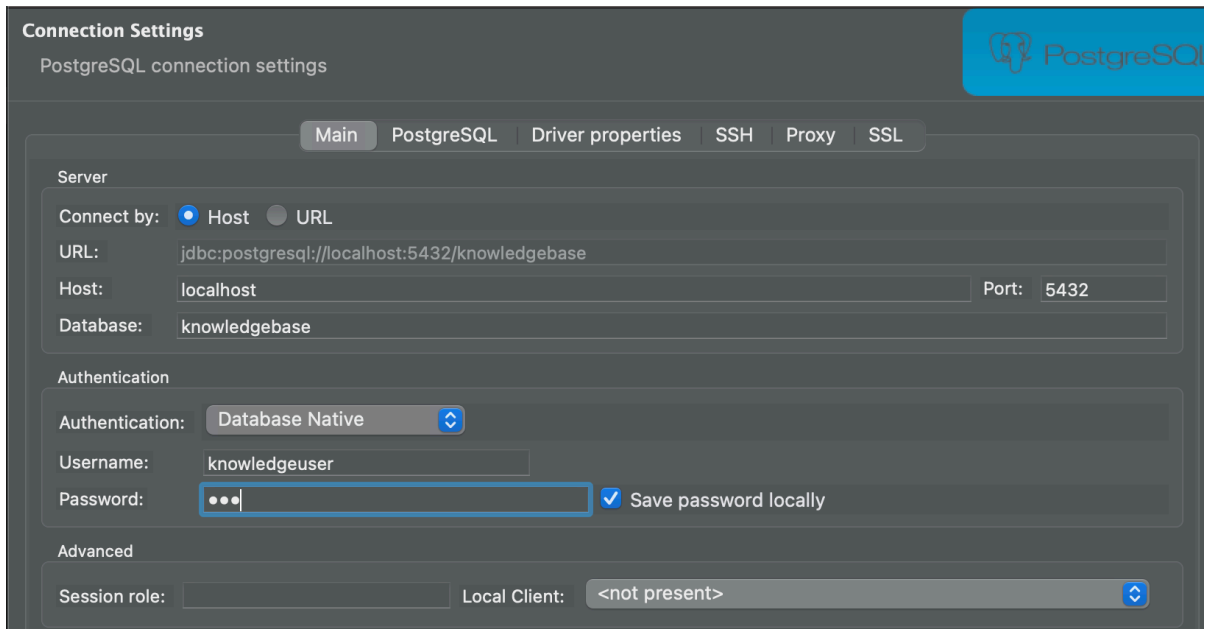
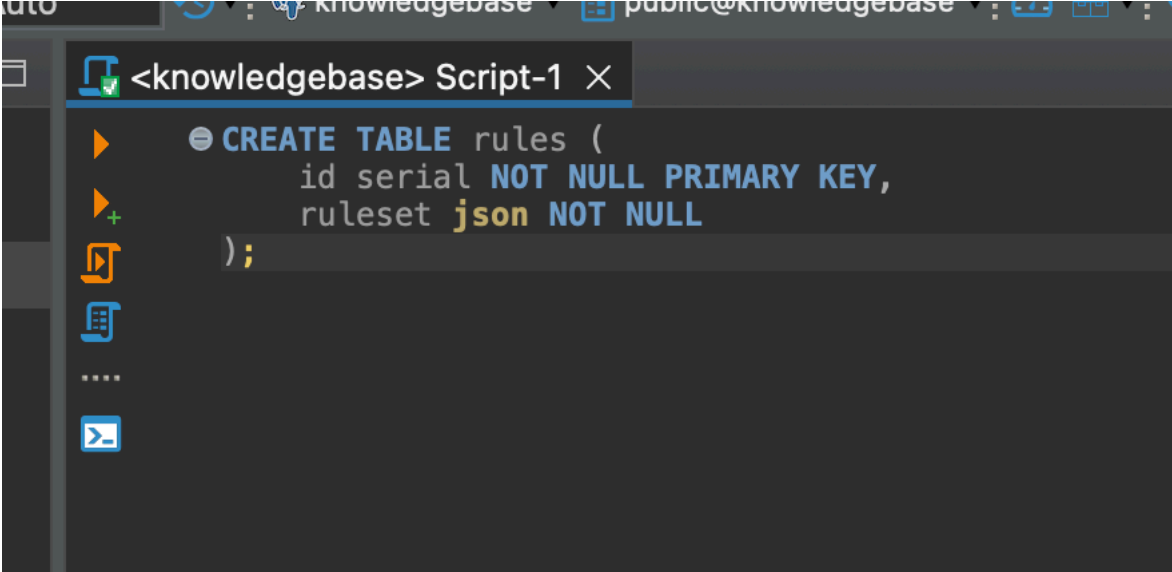


Рисунок 3.3 – Підключення до бази даних через DBeaver

Наступним кроком є створення таблиці з атрибутами та правилами. Для цього потрібно відкрити редактор скриптів та виконати SQL-запит для створення таблиці:



```
<knowledgebase> Script-1 X
CREATE TABLE rules (
  id serial NOT NULL PRIMARY KEY,
  ruleset json NOT NULL
);
```

Рисунок 3.4 – Створення таблиці rules

Таблиця з правилами буде містити всього два поля:

- Поле з ідентифікатором;
- Поле що буде містити JSON об'єкт з правилами.

Як тільки таблиця створена, можна спробувати внести перший набір правил. Звичайно, для зручності користування експертною системою буде розроблений графічний інтерфейс, через який користувач може швидко завантажити об'єкт з правилами до бази даних з JSON-файлу. Але поки графічний інтерфейс системи не підключений до бази даних, внесемо перший набір правил через CLI(Рисунок 3.5).

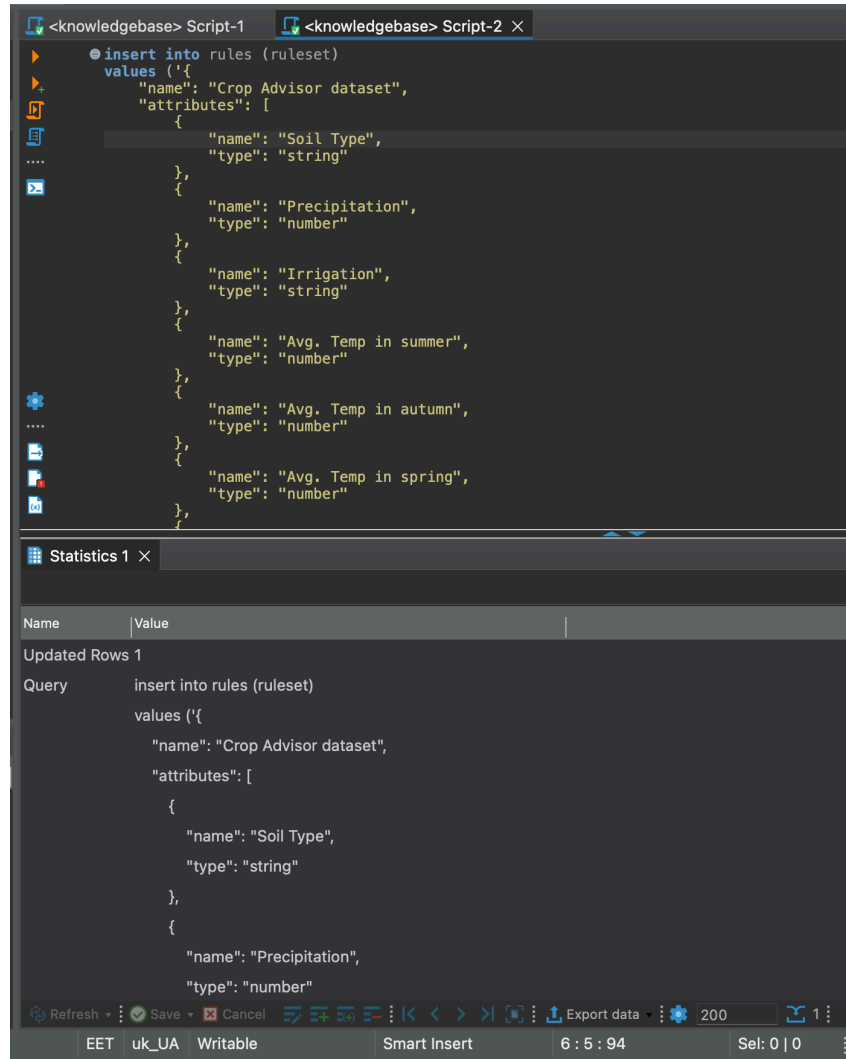


Рисунок 3.5 – Внесення до бази даних набору правил

В результаті, на Рисунку 3.6 можна побачити результат вищенаведеної команди.

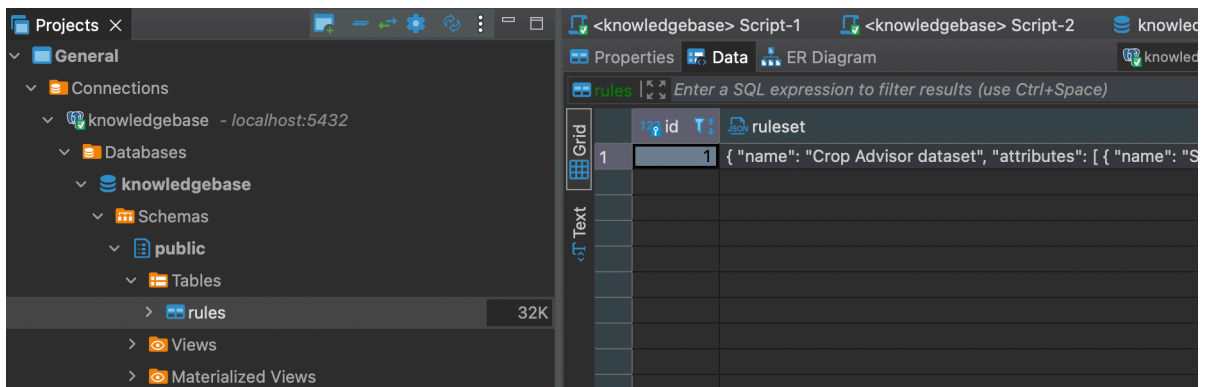


Рисунок 3.6 – Результат внесення перших даних до БД

Прийняте рішення, що експертна система матиме змогу вміщувати в собі багато наборів правил, а користувач за допомогою графічного інтерфейса зможе обирати потрібний набір в залежності від умов задачі, яку потрібно вирішити

3.2 Розроблення користувацького інтерфейсу. Елементи та структура

Перед розробкою користувацького інтерфейсу першим кроком було визначення функціональних та нефункціональних вимог, після чого розроблено структуру та елементи інтерфейсу.

До функціональних вимог користувацького інтерфейсу можна віднести наступні вимоги:

- Можливість перегляду списку різних культур та їх характеристик. Перегляд характеристик для зручності має бути побудований у вигляді дерева;
- можливість додавати нові факти до бази знань;
- можливість додавання нових даних про культури в базу даних системи;
- можливість пошуку культур;
- можливість візуалізації даних у вигляді ієрархічного дерева;
- можливість налаштування параметрів системи відображення даних та їх фільтрації;
- можливість отримання рекомендацій щодо оптимального вибору культур для конкретного господарства відштовхуючись від наявних ресурсів та географічного розположення;
- можливість імпорту та експорту даних для обміну з іншими системами та програмами.

Основні нефункціональні вимоги до користувацького інтерфейсу експертної системи зернотрейдингу включають:

- Ефективність: інтерфейс повинен бути простим та інтуїтивно зрозумілим, щоб користувачі могли швидко зрозуміти, як користуватися системою та ефективно виконувати завдання;
- надійність: система повинна бути стійкою до помилок та збоїв, а також мати засоби для відновлення роботи після відмови;
- безпека: інтерфейс повинен мати механізми для захисту конфіденційної інформації, такої як логін та пароль, від несанкціонованого доступу;
- сумісність: система повинна бути сумісна з різними операційними системами та браузерами, щоб користувачі могли використовувати її на різних пристроях;
- масштабованість: система повинна бути готовою до масштабування та розширення в разі необхідності.

Для більш детального опису інтерфейсу користувача була створена блок-схема, що зображує структуру користувацького інтерфейсу експертної системи (Рисунок 3.7). На блок-схемі виділені різні елементи інтерфейсу, включаючи головне меню та вкладки. Кожен елемент забезпечує певну функціональність системи. Наприклад, через головне меню можна перейти до будь-якої вкладки, де знаходиться необхідна інформація. На вкладках можна переглядати різні типи даних, набори правил, набори фактів та бази правил. Всі ці елементи об'єднуються в інтуїтивно зрозумілому інтерфейсі, що дозволяє користувачам швидко та легко знаходити необхідну інформацію та виконувати необхідні дії.

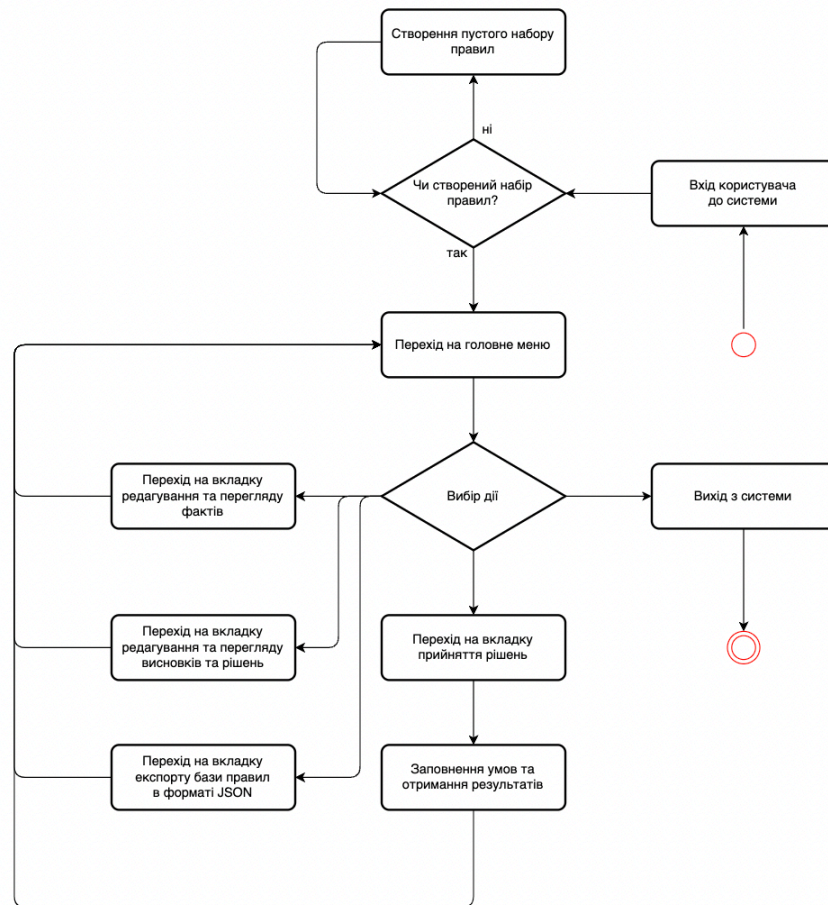


Рисунок 3.7 – Блок-схема користувацького інтерфейсу

Спираючись на блок схему та описані вимоги було розроблено користувацький інтерфейс, який дозволить зручно взаємодіяти з базою знань.

Елемент інтерфейсу, що показаний на Рисунку 3.8, відповідає за вхід в експертну систему. На рисунку можна побачити дві кнопки, які дозволяють завантажити набір правил з локальної директорії в систему, або, відповідно, створити новий. Ця форма є доволі простою, тому важливо було забезпечити його максимальну її лаконічність та зручність використання.

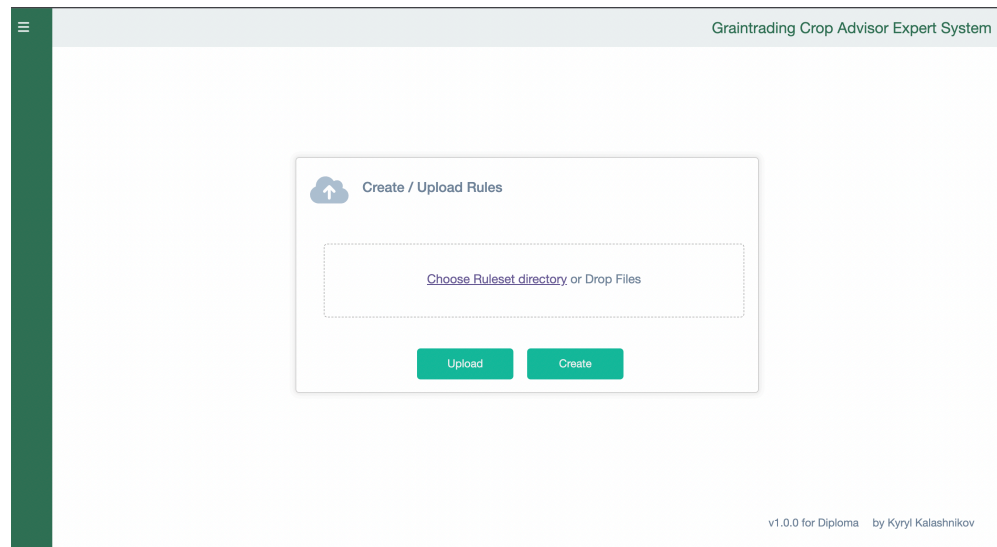


Рисунок 3.8 – Вікно завантаження або створення набору правил

Потім користувач бачить головне меню системи, яке відкривається після завантаження або створення набору правил, на вкладці фактів. На цій вкладці користувач може переглядати (Рисунок 3.9) та редагувати (Рисунок 3.10) факти, які використовуються для прийняття рішень в експертній системі. Для зручності користувача, факти відображені з типом даних, необхідним для вводу. Також на даній вкладці є можливість додавання нових фактів за допомогою кнопки "Додати факт".

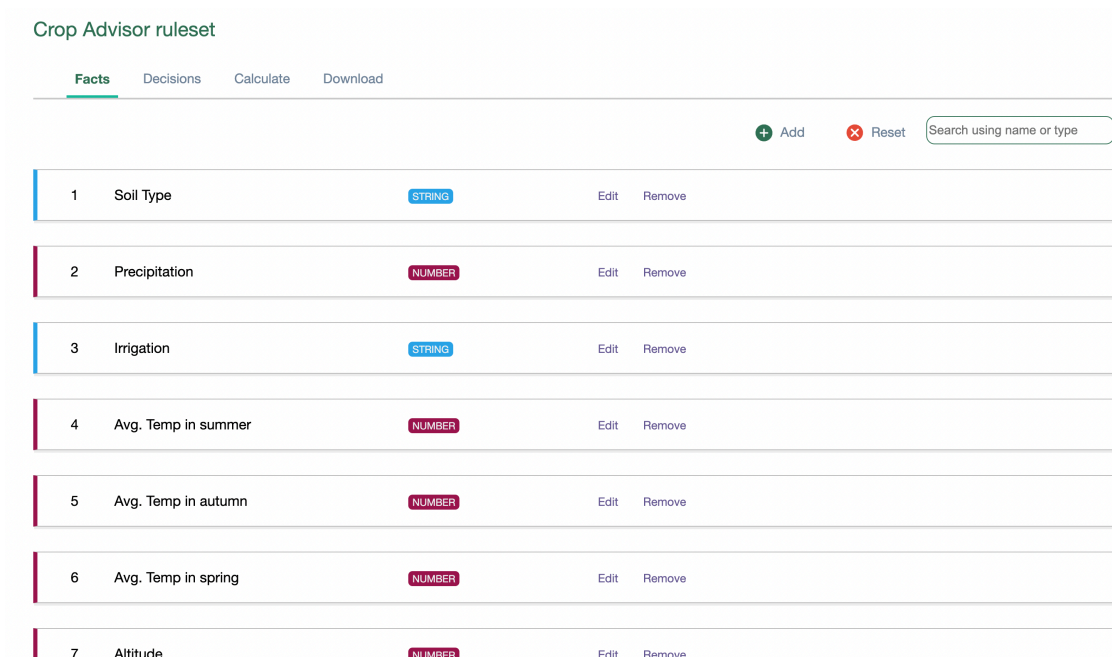


Рисунок 3.9 – Вікно перегляду наявних фактів експертної системи

1	Soil Type	STRING	Edit	Remove
---	-----------	--------	------	--------

Name:

Type:

Рисунок 3.10 – Редагування факту експертної системи

Наступна вкладка містить список усіх рішень, які були створені в системі. На цій вкладці можна переглянути детальну інформацію про кожне рішення, включаючи його умови та наслідки, а також можливість редагувати або видаляти рішення. Також на цій вкладці можна додавати нові рішення, використовуючи наявні умови та наслідки, або створюючи нові.

Crop Advisor ruleset				
Facts	Decisions	Calculate	Download	
				<input type="button" value="+ Add"/> <input type="button" value="✗ Reset"/> <input type="text" value="Search using name or type"/>
1	Bad soil for any crop	conditions 2	View Conditions	Remove
2	Too high to farming	conditions 1	View Conditions	Remove
3	Not enough precipitation	conditions 1	View Conditions	Remove
4	Lentil	conditions 1	View Conditions	Remove
5	Corn	conditions 1	View Conditions	Remove
6	Wheat	conditions 1	View Conditions	Remove

Рисунок 3.11 – Редагування висновків експертної системи

При натиску кнопки детального перегляду умов відкривається блок з деревовидною структурою рішення. Перегляд висновків в експертній системі можливий через деревовидну структуру, яка відображена на Рисунку 3.11.

Кожен вузол дерева представляє собою факт, на основі якого було зроблено висновок. Верхній вузол є початковим фактом, а кожен наступний вузол є висновком, зробленим на основі попередніх фактів. Наприклад, на рисунку можна побачити, що з якщо збігаються умови де на земельній ділянці піщаний тип ґрунта, опади менше 300мм та відсутній штучний полив то буде зроблено висновок "Недостатня вологість" для будь якої культури.

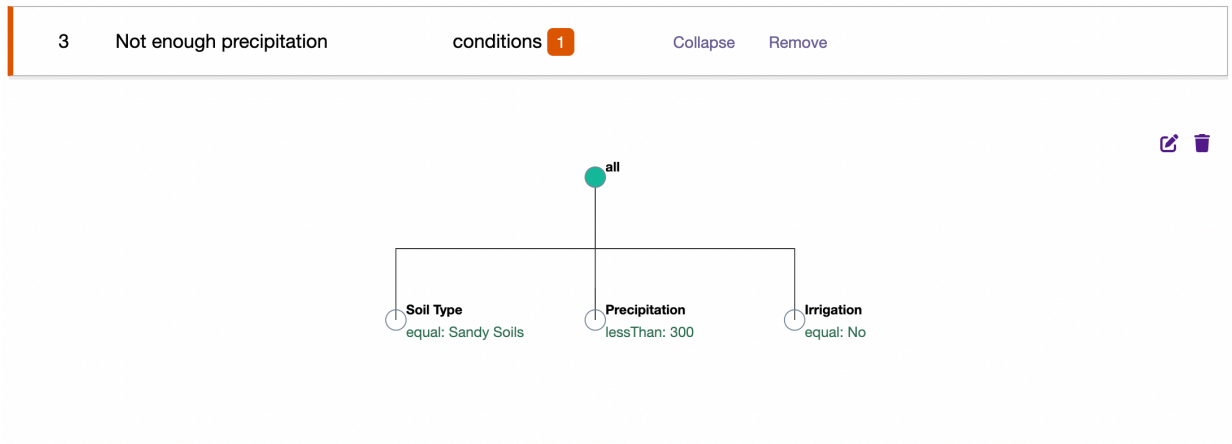


Рисунок 3.12 – Редагування факту експертної системи

Створення такої структури було можливим за допомогою React-d3-tree [33]. Це бібліотека компонентів для React, що дає можливість створювати деревовидні структури з використанням D3.js. Вона дозволяє легко і швидко створювати різноманітні діаграми та дерева з використанням стандартних React-компонентів та стилів.

React-d3-tree є потужним інструментом для відображення складних структур даних в деревовидній формі з використанням React та D3.js. Ця бібліотека дозволяє зручно і швидко створювати візуалізації, які забезпечують легкий доступ до даних та зручний інтерфейс для їх взаємодії з користувачем.

Наступна вкладка валідації та формування висновку (Рисунок 3.12) є головною в системі, саме тут користувач вводить дані, по яким буде формуватися висновок. В правій колонці можна побачити список доступних фактів.

Crop Advisor ruleset

Facts Decisions **Calculate** Download

Name	Value
Soil Type	<input type="text"/>
Precipitation	<input type="text"/>
Irrigation	<input type="text"/>
Avg. Temp in summer	<input type="text"/>
Avg. Temp in autumn	<input type="text"/>
Avg. Temp in spring	<input type="text"/>
Altitude	<input type="text"/>
Labour resources	<input type="text"/>
Mechanization	<input type="text"/>
Include market factors	<input type="text"/>
State benefits	<input type="text"/>

Рисунок 3.13 – Блок формування висновків

Для отримання висновку експертної системи потрібно заповнити відповідні поля у формі та натиснути кнопку Get Decision.

Avg. temp in summer	<input type="text" value="24"/>
Avg. Temp in autumn	<input type="text" value="9"/>
Avg. Temp in spring	<input type="text" value="9"/>
Altitude	<input type="text" value="300"/>
Labour resources	<input type="text" value="Medium"/>
Mechanization	<input type="text" value="Yes"/>
Include market factors	<input type="text" value="No"/>
State benefits	<input type="text" value="No"/>
Previous crop	<input type="text" value="Barley"/>

Outcomes

Type	Lentil
Type	Corn
Type	Wheat

Рисунок 3.14 – Графічний інтерфейс отримання висновків

У вихідній інформації можемо бачити список культур, що підходять для вирощення, відсортований по спаду продуктивності.

Остання вкладка, яка є в системі – вивантаження бази правил. База правил вивантажується в форматі JSON. Даний функціонал може бути корисним для завантаження даних в інші системи зернотрейдингового господарства.

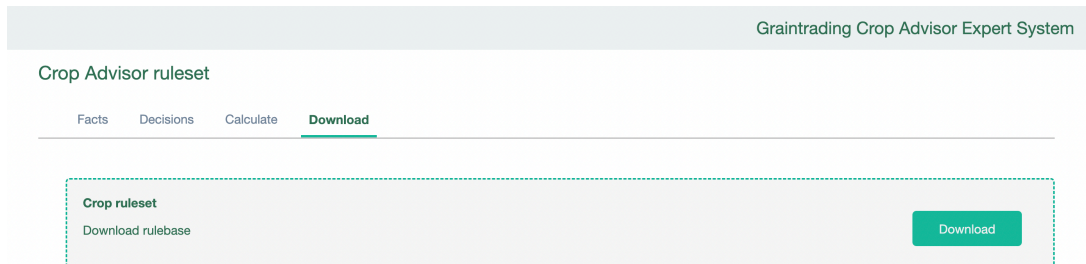


Рисунок 3.15 – Інтерфейс вивантаження бази правил

3.3 Проектування забезпечувальних підсистем експертної системи

Під забезпечувальними підсистемами експертної системи мається на увазі програмне забезпечення, яке забезпечує коректну роботу системи та підтримку її функціоналу. Розробка забезпечувальних підсистем є важливим етапом в проектуванні експертної системи, оскільки вони забезпечують надійність та стабільність роботи системи.

3.3.1 Інформаційне забезпечення

Інформаційне забезпечення передбачає вирішення завдань поширення інформації, проведення адміністративно-організаційних, науково-дослідних і

виробничих заходів по її ефективному використанню. Цілі і завдання інформаційного забезпечення можливо визначити наступним чином:

- задоволення інформаційних потреб підрозділів підприємства, надання доступу підтримки прийняття рішень в електронному вигляді;
- наповнення та реалізація даних експертної системи;
- розширення можливостей та розвиток експертної системи.

Важливим аспектом інформаційного забезпечення є формування даних для експертної системи. Інформація є необхідним елементом всієї системи, без якої неможливе її технічне функціонування. Машинна обробка інформації що вноситься до системи - основа сучасних експертних систем.

Гарантування безпеки експертної системи являється важливим завданням, що реалізується впровадженням системи безпеки. Проблема захисту інформації є багатоплановою та охоплює ряд важливих завдань. Проблеми інформаційної безпеки постійно поглиблюються процесами проникнення в усі сфери суспільства технічних засобів обробки та передачі даних і, перш за все, обчислювальних систем.

Експертна система з вибору культур зернотрейдингового підприємства розроблена з дотриманням основних базових принципів інформаційної безпеки

- цілісність даних – захист від збоїв, що ведуть до втрати інформації, а також захист від неавторизованого створення або видалення даних;
- конфіденційність інформації;
- доступність інформації для всіх користувачів, які під'єднані до корпоративної мережі.

Бізнес-логіка сформована таким чином, що неавторизований користувач не матиме доступу до інформації, не зможе змінювати або видаляти її зміст.

Контроль інформації у системі здійснюється за допомогою методу контролю формату повідомлення. При форматному контролі введеної інформації проводяться наступні дії:

- вираховується число окремих символів та порівнюється з потрібною системою кількістю;
- перевіряється довжина запису, що не повинна перевищувати встановлені рамки;
- проводиться перевірка символів на відповідність (там, де система потребує вводу числового значення має бути лише числове значення).

У випадку виявлення помилок, користувачу який вводив інформацію до системи, виводиться повідомлення, що містить опис та причину помилки.

Користувачі експертної системи мають можливість вносити інформацію у вигляді об'єкта JSON вручну через спеціальну форму перед початком роботи з системою, а також заповнювати інформацію у формі на вкладці "Calculate" при розрахунку оптимальної культури для вирощування. Після відправки форми, дані автоматично зберігаються до бази даних. Частота надходження нової інформації залежить від активності та потреб користувачів експертної системи.

Для проекту було обрано реляційну базу даних PostgreSQL. Основними причинами вибору PostgreSQL були його надійність, розширюваність і широкі можливості. PostgreSQL володіє стабільною репутацією як надійна та безпечна база даних з високою стійкістю до помилок і збоїв. Вона також підтримує широкий спектр функцій і можливостей, включаючи гнучкість в роботі зі схемою даних, повнотекстовий пошук, транзакційність і підтримку зовнішніх ключів. Крім того, PostgreSQL має активну спільноту розробників і багато документації, що дозволяє легко знайти підтримку та вирішити потенційні проблеми. Враховуючи всі ці фактори, PostgreSQL був ідеальним вибором для проекту з метою забезпечення стабільної та ефективної взаємодії експертної системи з базою даних.

3.3.2 Програмне забезпечення

Добре розроблена експертна система спирається на цілісну основу, яка підтримує її роботу. Інфраструктура інформаційної системи, складається в основному з телекомунікаційних мереж, баз даних та сховищ даних,

програмного забезпечення, обладнання та процедур, якими керують різні спеціалісти.

Комп'ютерне програмне забезпечення поділене на дві частини: системне та прикладне програмне забезпечення.

Операційна система є головним системним програмним забезпеченням. Вона керує апаратним забезпеченням, файлами, програмами та іншими системними ресурсами. Операційна система надає можливість користувачу для управління комп'ютером за допомогою графічного інтерфейсу (graphical user interface). Прикладне програмне забезпечення - це програми, які потрібні для розв'язання обраних задач користувачів.

Програмне забезпечення експертної системи зернотрейдингової компанії можна розділити на чотири групи:

- програмне забезпечення бізнесу;
- програмне забезпечення взаємодії з контрагентами;
- програмне забезпечення управління продуктами;
- технологічне програмне забезпечення.

Програмне забезпечення бізнесу в основному використовуються при автоматизації управління. Складається з пакетів офісних програм, програм бухгалтерського обліку та фінансових програм. На даному підприємстві вирішено використовувати CTRM Graintrack. В якості програми бухгалтерського обліку обрано технологічну платформу BAS Бухгалтерія.

CTRM Graintrack [34] є інтегрованою системою управління торгівлею зерном (Commodity Trading and Risk Management). Вона спеціально розроблена для зернотрейдингових компаній з метою автоматизації і оптимізації процесів, пов'язаних з купівлею, продажем і логістикою зернових товарів. CTRM Graintrack надає широкий функціонал, включаючи керування контрактами, управління складами, ведення фінансових операцій, аналітику ринку та ризик-

менеджмент. Система дозволяє покращити ефективність роботи, знизити ризики і забезпечити точність обліку та звітності.



Рисунок 3.16 – Интерфейс Graintrack

Технологічне програмне забезпечення використовується для управління та організації роботи зернотрейдерів, купівлю або продажем зерна, стеження за процесом вирощення урожаю.

Програмне забезпечення управління продуктами дозволяє оптимізувати весь процес роботи зернотрейдингової компанії, за допомогою цього типу ПЗ можливо контролювати використання матеріальних і не матеріальних ресурсів на підприємстві.

BAS Бухгалтерія [35] є програмним забезпеченням для автоматизації бухгалтерського обліку і фінансового управління. Вона надає широкий набір функцій, включаючи облік доходів і витрат, розрахунок податків, формування фінансової звітності, облік заборгованостей і оплат, управління банківськими операціями та інше. BAS Бухгалтерія дозволяє підприємствам ефективно вести облік фінансових операцій, забезпечувати точність та надійність фінансової звітності, спрощувати процеси звітності перед податковими органами і

підвищувати ефективність бухгалтерської роботи. BAS Бухгалтерія є корисним інструментом для підприємств будь-якого розміру, які прагнуть покращити свої бухгалтерські процеси та забезпечити дотримання вимог фінансового обліку і податкового законодавства.

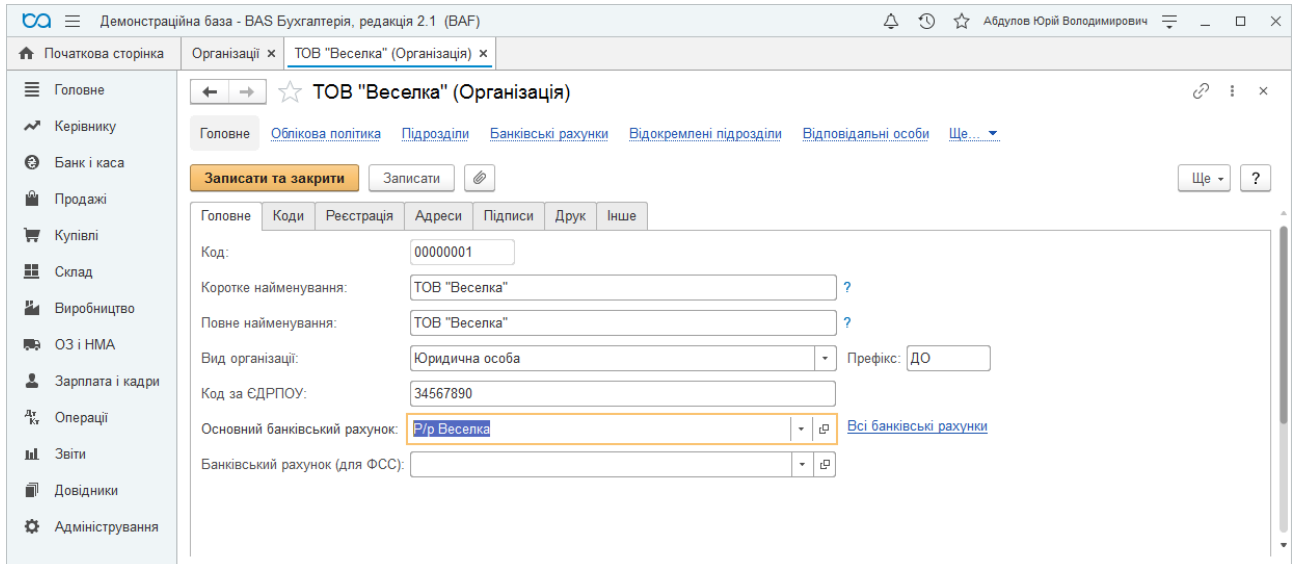


Рисунок 3.17 – Інтерфейс BAS Бухгалтерія

3.3.3 Технічне забезпечення

Експертна система зернотрейдингової компанії потребує наявності хостингу, що є формою оренди простору на сервері для зберігання системних файлів та бази даних. Доступ працівників до системи забезпечується через браузер з доступом до мережі Інтернет.

В проєкті був обраний підхід використання хмарної інфраструктури для розгортання та забезпечення експертної системи. Хмарна інфраструктура, зокрема платформа Amazon Web Services (AWS), була обрана з декількох причин. По-перше, вона надає широкий спектр послуг, таких як обчислювальні ресурси, зберігання даних, мережеві сервіси та інші, що дозволяє гнучко масштабувати інфраструктуру під потреби проєкту. По-друге, вона забезпечує високий рівень безпеки, включаючи захист даних, резервне копіювання та контроль доступу. По-третє, використання хмарної інфраструктури дозволяє знизити витрати на придбання та підтримку фізичного обладнання, оскільки

ресурси надаються на оренду за потребою. Використання хмарної інфраструктури також дозволяє забезпечити високу доступність та надійність системи шляхом розподілення ресурсів на різних фізичних серверах та центрах обробки даних. Загалом, використання хмарної інфраструктури дозволяє ефективно розгортати та керувати інфраструктурою проекту, забезпечуючи гнучкість, безпеку та економічну вигоду.

В якості хостингу було обране використання EC2 (Elastic Compute Cloud). Це сервіс віртуальних серверів в хмарній інфраструктурі Amazon Web Services (AWS). Використання EC2 дає можливість забезпечити високий рівень доступності та масштабованості, який може бути необхідним для системи, що отримує велику кількість запитів від користувачів.

EC2 забезпечує можливість створення та керування віртуальними машинами (instances) з різними характеристиками, такими як об'єм пам'яті, кількість процесорів та об'єм дискового простору. Це дає можливість забезпечити необхідну продуктивність та масштабованість системи в залежності від потреб користувачів.

Доступ до системи здійснюватиметься через браузер з доступом до мережі Інтернет, що дозволяє працівникам компанії користуватися системою з будь-якого пристрою, який має доступ до Інтернету.

Щоб створити EC2 інстанс (сервер), потрібно виконати наступні кроки:

1. Створити акаунт в AWS та зареєструвати кредитну картку для оплати сервісу.
2. Зайти в консоль AWS та вибрати сервіс EC2.
3. Натиснути кнопку "Launch Instance" та вибрати базовий образ (АМІ - Amazon Machine Image), який містить операційну систему, на якій буде працювати інстанс.

4. Встановити тип інстансу відповідно до потреб, що може бути визначено кількістю процесорів, об'ємом оперативної пам'яті та іншими характеристиками.

5. Визначити обсяг жорсткого диску та тип зберігання, який використовуватиметься для даного інстансу.

6. Налаштувати мережеві настройки, визначивши вхідні та вихідні правила доступу до інстансу, а також визначивши тип мережі (наприклад, публічна чи приватна).

7. Налаштувати безпеку, додавши правила безпеки для інстансу та оберігаючи ключі для доступу до інстансу.

8. Переглянути налаштування та запустити інстанс.

Після цього, інстанс буде створений та можна зайти на нього, використовуючи SSH-ключ або пароль. Відомості про інстанс, такі як його IP-адреса та інші характеристики, можна знайти в консолі EC2 у вкладці "Instances".

The screenshot displays the AWS Management Console interface for an EC2 instance. At the top, there's a header for 'Instances (1/1)' with a search bar and filters. A table lists the instance 'Diploma Server' with ID 'i-061ecad79b089b714', state 'Running', type 't2.micro', and availability zone 'eu-central-1a'. Below this, the 'Instance: i-061ecad79b089b714 (Bot Server)' details are shown, including a 'Details' tab. The 'Instance summary' section provides key information:

Property	Value
Instance ID	i-061ecad79b089b714 (Bot Server)
Public IPv4 address	3.75.187.252 open address
Private IPv4 addresses	172.31.27.175
IPv6 address	-
Instance state	Running
Public IPv4 DNS	ec2-3-75-187-252.eu-central-1.compute.amazonaws.com open address
Hostname type	IP name: ip-172-31-27-175.eu-central-1.compute.internal
Private IP DNS name (IPv4 only)	ip-172-31-27-175.eu-central-1.compute.internal
Answer private resource DNS name IPv4 (A)	-
Instance type	t2.micro
Auto-assigned IP address	3.75.187.252 [Public IP]
VPC ID	vpc-0dec9a80dc57956d6
Elastic IP addresses	-
AWS Compute Optimizer finding	Opt-in to AWS Compute Optimizer for recommendation s.

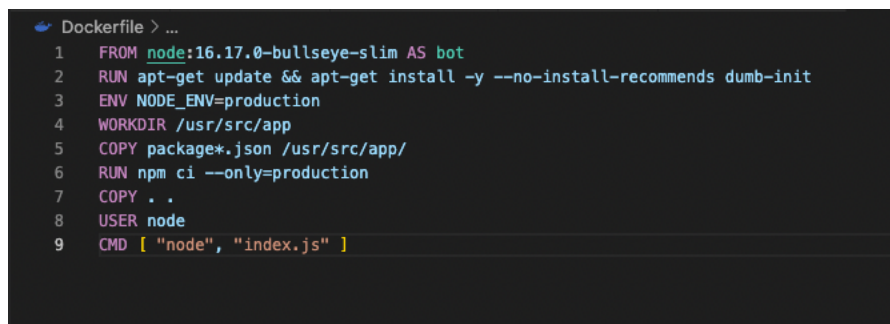
Рисунок 3.18 – Сервер в консолі AWS

Операційною системою EC2 інстансу обрано Ubuntu Server 22.04 [36]. Це операційна системою на основі Linux, яка призначена для використання на

серверах. Вона має вбудований пакет програм, які дозволяють встановлювати та налаштовувати сервер, включаючи веб-сервер, бази даних, фаїрвол та інші додатки. Ubuntu Server 22.04 є останньою версією на момент написання цього тексту та має багато нових функцій та покращень порівняно з попередніми версіями. Однією з особливостей Ubuntu Server є те, що вона є безкоштовною та з відкритим вихідним кодом, що дозволяє знизити витрати на створення серверів. Крім того, Ubuntu Server має велику спільноту користувачів та розробників, які допомагають вирішувати проблеми та підтримують операційну систему з оновленнями та безпековими патчами.

Для більшої надійності та масштабованості система була контейнеризована на сервері за допомогою Docker. Для створення образу системи використовуються докерфайли. Dockerfile - це текстовий файл, який містить інструкції для автоматичного створення Docker-образу. Інструкції Dockerfile описують, як побудувати Docker-образ, включаючи початковий образ, необхідні залежності, середовище виконання і кінцеву конфігурацію.

Файл Dockerfile складається з рядків інструкцій, кожна з яких описує окрему дію, таку як копіювання файлів, встановлення пакетів або налаштування середовища виконання. Кожен рядок інструкції починається з ключового слова, такого як FROM, RUN, COPY, EXPOSE, CMD і т.д., та виконує певну дію, що допомагає побудувати потрібний Docker-образ.



```
Dockerfile > ...
1 FROM node:16.17.0-bullseye-slim AS bot
2 RUN apt-get update && apt-get install -y --no-install-recommends dumb-init
3 ENV NODE_ENV=production
4 WORKDIR /usr/src/app
5 COPY package*.json /usr/src/app/
6 RUN npm ci --only=production
7 COPY . .
8 USER node
9 CMD [ "node", "index.js" ]
```

Рисунок 3.19 – Структура Dockerfile

Після того, як Dockerfile створено, його можна використовувати для збирання Docker-образу, використовуючи команду "docker build". Під час

збирання Docker-образу, Dockerfile інтерпретується і виконується, створюючи кінцевий образ, готовий до запуску в контейнері.

Для оркестрації контейнерів з експертною системою та базою даних використовувався docker-compose.

Docker Compose дозволяє описати всі контейнери, що необхідні для додатку в одному файлі - docker-compose.yml. У цьому файлі можна задати налаштування для кожного контейнера, включаючи базовий образ, з якого він буде запускатися, порти, які потрібно відкрити, змінні середовища, мережі і т.д.

Після опису всіх контейнерів у файлі docker-compose.yml, можна запустити їх одночасно з допомогою команди *docker-compose up*. Docker Compose автоматично створить всі контейнери з описаними у файлі налаштуваннями та підключить їх до спільної мережі, щоб вони могли спілкуватися між собою.

```
docker-compose.yml - The Compose specification establishes a standard for the definition of multi-container applications.
1  version: "3.9"
2  services:
3    expert_sys:
4      build:
5        context: .
6        dockerfile: Dockerfile
7      depends_on:
8        - postgres
9      environment:
10     PSQL_HOST: postgres
11     restart: always
12     mem_limit: 400m
13     ports:
14       - 8080:8080
15   postgres:
16     image: postgres:13.3
17     env_file: .env
18     environment:
19       POSTGRES_DB: ${DB_NAME}
20       POSTGRES_USER: ${DB_USER}
21       POSTGRES_PASSWORD: ${DB_PASSWORD}
22     volumes:
23       - db:/var/lib/postgresql/data
24     ports:
25       - "5432:5432"
26     restart: always
27     mem_limit: 300m
28
29   volumes:
30     db:
```

Рисунок 3.20 – docker-compose.yml

Використання Docker Compose [37] дозволяє спростити та автоматизувати процес розгортання додатків, забезпечуючи їхню однорідність та портативність між середовищами розробки, тестування та виробництва.

Проект був успішно розгорнутий на сервері з використанням Docker, після запуску контейнерів отримано працюючу систему. Кожен контейнер відповідає окремій складовій системи, і всі вони можуть взаємодіяти між собою, обмінюватися даними та інформацією.

Precipitation	700
Irrigation	No
Avg. Temp in summer	23
Avg. Temp in autumn	9
Avg. Temp in spring	9
Altitude	300
Labour resources	Low
Mechanization	No
Include market factors	Yes
State benefits	No
Previous crop	Wheat

Validate Ruleset

Outcomes

Type	Corn

Рисунок 3.21 – система на сервері

3.3.4 Організаційно-економічне забезпечення

Організаційно-економічне забезпечення грає важливу роль у забезпеченні успішної реалізації проектних завдань. Організація та ефективне використання ресурсів є ключовими аспектами проекту розробки експертної системи, які необхідно врахувати при плануванні та реалізації дослідження. Деякі з найважливіших аспектів ресурсного забезпечення в магістерському проекті включають:

1. Навчальні ресурси: Для магістерського проекту знадобились спеціальні навчальні матеріали, підручники, наукові статті, доступ до електронних

бібліотек та інших ресурсів, які допомогли в розробці дослідження і формулюванні наукових основ проекту. Перелік використаних наукових праць та публікацій наведений у Списку використаних джерел;

2. Технічне обладнання: Залежно від характеру магістерського проекту, знадобилось спеціалізоване обладнання. Для написання роботи та побудови експертної системи був використаний ноутбук Apple MacBook Air M1. Додатково до ноутбука був використаний монітор HP, що дозволяє розширити робочу площу та покращити комфорт під час роботи. Цей монітор володіє високою якістю відображення, чіткістю та насиченістю кольорів, що є важливими факторами при розробці та візуалізації експертної системи;

3. Експертні ресурси: На етапах дослідження властивостей сільськогосподарських культур магістерський проект вимагав співпраці з експертами для отримання спеціалізованої інформації, консультацій або навчання. Це були фахівці з галузі зернотрейдингу, які допомогли досягти розуміння предметної області та сприяли вирішенню проблеми, що досліджувалась;

4. Фінансові ресурси: Під час написання дипломного проекту було використано фінансові ресурси на придбання ліцензованого програмного забезпечення Microsoft Office. Також, з метою економії коштів, в процесі дипломного проекту була використана безкоштовна trial версія ArcGIS. Використання безкоштовної trial-версії дозволило отримати доступ до необхідних функцій та ресурсів ArcGIS без додаткових витрат на ліцензію.

Можна підкреслити, що правильне організаційно-економічне забезпечення стало важливим фактором успіху проекту. Організаційно-економічне забезпечення допомогло забезпечити раціональне використання ресурсів, уникнення затримок та перешкод у процесі виконання проекту.

ВИСНОВКИ

Дипломна робота присвячена розробці експертної системи для зернотрейдингової компанії. У роботі було розглянуто основні питання, що виникають при проектуванні та розробці експертних систем, такі як архітектура системи, вибір інструментів, процес розробки та розгортання.

Було проведено аналіз діяльності зернотрейдингової компанії та виявлено проблеми, з якими стикається компанія при прийнятті рішень щодо вирощування нового зерна. Для вирішення цих проблем було запропоновано розробити експертну систему, яка буде допомагати збирати та обробляти дані щодо вибору культур, які збирається вирощувати зернотрейдингова компанія

Для розробки експертної системи було використано програмну платформу Node.js, мову програмування JavaScript, фреймворк React для розробки веб-додатку, базу даних PostgreSQL та сервіси AWS для розгортання та хостингу додатку.

У результаті роботи була розроблена прототип експертної системи, яка здатна збирати та обробляти дані щодо якості зерна, прогнозувати його ціну та допомагати при прийнятті рішень щодо його закупівлі. Розроблена система є гнучкою та масштабованою, що дає можливість додавати нові функції та розширювати її функціональність в майбутньому.

В рамках розробки експертної системи для зернотрейдингової компанії було зібрано дані з різних джерел, що дозволило збільшити обсяг інформації та забезпечити більш точні прогнози. Зокрема, використано дані про ціни на зерно, властивості ґрунтів, погодні умови та статистику виробництва зернових культур.

Також було проведено аналіз ринку та конкурентів, що дозволило зрозуміти потреби та вимоги до експертної системи. За результатами аналізу було визначено ключові функції та можливості, які повинна мати система.

Під час проектування та розробки системи ми використовували сучасні технології, такі як Docker, React та D3.js, що забезпечило швидку та ефективну

розробку системи з високим рівнем функціональності та зручним інтерфейсом користувача.

У результаті роботи над проектом було створено прототип експертної системи, який дозволяє отримувати точні прогнози щодо вибору культури для вирощення. Прототип може бути успішно інтегрований в роботу зернотрейдингової компанії та допомагати зернотрейдерам приймати правильні рішення.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Показники виробництва пшениці та ячменю у 2019/2020 МР [Електронний ресурс] / 1 – Режим доступу до ресурсу: <https://superagronom.com/multimedia/infographics/69-pokazniki-virobnitstva-pshenitsi-ta-yachmenyu-u-2019-2020-mr>.
2. Площі, валові збори та урожайність сільськогосподарських культур за їх видами та по регіонах у 2021 році (остаточні дані) [Електронний ресурс] – Режим доступу до ресурсу: https://ukrstat.gov.ua/operativ/operativ2022/sg/pvzu/pvzu_2021.zip.
3. ЗЕРНОВИЙ КОРИДОР [Електронний ресурс] – Режим доступу до ресурсу: <https://www.ukrinform.ua/tag-zernovij-koridor>.
4. COMPANY STRUCTURE [Електронний ресурс] – Режим доступу до ресурсу: <https://www.upi-agro.com.ua/en/Home/Structure>.
5. EOS Data Analytics [Електронний ресурс] – Режим доступу до ресурсу: <https://eos.com/uk/products/crop-monitoring/>.
6. FARM MANAGEMENT SOFTWARE 2.0 POWERED BY BUSHEL [Електронний ресурс] – Режим доступу до ресурсу: <https://bushelfarm.com/>.
7. <https://www.agworld.com/us/> [Електронний ресурс] – Режим доступу до ресурсу: Agriculture's preferred independent farm information management ecosystem.
8. Мікросервісна архітектура [Електронний ресурс] – Режим доступу до ресурсу: <https://www.globallogic.com/ua/insights/blogs/microservices-architecture-for-beginners-part-one/>.
9. Node.js [Електронний ресурс] – Режим доступу до ресурсу: <https://nodejs.org/uk>.
10. Webpack [Електронний ресурс] – Режим доступу до ресурсу: <https://webpack.js.org/>.
11. PostgreSQL: The World's Most Advanced Open Source Relational Database [Електронний ресурс] – Режим доступу до ресурсу: <https://www.postgresql.org/>.
12. Git open source distributed version control system [Електронний ресурс] – Режим доступу до ресурсу: <https://git-scm.com/>.
13. Docker [Електронний ресурс] – Режим доступу до ресурсу: <https://www.docker.com/>.
14. React JavaScript-бібліотека для створення користувацьких інтерфейсів [Електронний ресурс] – Режим доступу до ресурсу: <https://uk.legacy.reactjs.org/>.
15. Docker Architecture in Detail [Електронний ресурс] – Режим доступу до ресурсу: <https://www.whizlabs.com/blog/docker-architecture-in-detail/>.

16. Top 25 AWS Services List 2023 [Електронний ресурс] – Режим доступу до ресурсу: <https://allcode.com/top-aws-services/>.
17. Amazon EC2 Secure and resizable compute capacity for virtually any workload [Електронний ресурс] – Режим доступу до ресурсу: <https://aws.amazon.com/ru/ec2/>.
18. Factors affecting crop choosing, growing and selling patterns. [Електронний ресурс] – Режим доступу до ресурсу: https://www.researchgate.net/figure/Factors-affecting-crop-choosing-growing-and-selling-patterns_fig4_267511079.
19. Understanding the potential applications of Artificial Intelligence in Agriculture Sector [Електронний ресурс] – Режим доступу до ресурсу: <https://www.sciencedirect.com/science/article/pii/S277323712200020X>.
20. Applying Big Data for Intelligent Agriculture-Based Crop Selection Analysis [Електронний ресурс] – Режим доступу до ресурсу: <https://ieeexplore.ieee.org/abstract/document/8801889>.
21. Typical Structure of the Inference Engine [Електронний ресурс] – Режим доступу до ресурсу: https://www.researchgate.net/figure/Typical-Structure-of-the-Inference-Engine_fig2_290481752.
22. FUNDAMENTALS OF EXPERT SYSTEMS [Електронний ресурс] – Режим доступу до ресурсу: <https://stacks.stanford.edu/file/druid:gh880nv8408/gh880nv8408.pdf>.
23. ОЦІНКА ПОТЕНЦІАЛУ СІЛЬСЬКОГОСПОДАРСЬКИХ ЗЕМЕЛЬ УКРАЇНИ ЗА МЕТОДОЛОГІЄЮ АГРОЕКОЛОГІЧНОГО ЗОНУВАННЯ ФАО [Електронний ресурс] – Режим доступу до ресурсу: http://eip.org.ua/docs/EP_06_4_55_uk.pdf.
24. Build interactive web maps with ArcGIS Online, Esri's web-based mapping software [Електронний ресурс] – Режим доступу до ресурсу: <https://www.arcgis.com/index.html>.
25. KNOWLEDGE-BASED EXPERT SYSTEM FOR AGRICULTURAL LAND USE PLANNING [Електронний ресурс] – Режим доступу до ресурсу: <https://digital.library.adelaide.edu.au/dspace/bitstream/2440/96414/3/02whole.pdf>.
26. Обсяг виробництва, урожайність та зібрана площа сільськогосподарських культур за їх видами [Електронний ресурс] // Державна служба статистики України – Режим доступу до ресурсу: https://ukrstat.gov.ua/operativ/operativ2021/sg/ovuzpsg/ovuzpsg_1221.xls.
27. Посівні площі озимих культур за їх видами [Електронний ресурс] // Державна служба статистики України – Режим доступу до ресурсу: https://ukrstat.gov.ua/operativ/operativ2018/sg/ppsgk/ppozk_2018.xls.

28. Площі, валові збори та урожайність сільськогосподарських культур за їх видами та по регіонах [Електронний ресурс] // Державна служба статистики України. – 2021. – Режим доступу до ресурсу: 9. https://ukrstat.gov.ua/operativ/operativ2022/sg/pvzu/pvzu_2021.zip.
29. Аналіз погодних умов в Україні в 2021 році [Електронний ресурс] // Метео Фарм. – 2021. – Режим доступу до ресурсу: 10. <https://superagronom.com/blog/871-analiz-pogodnih-umov-v-ukrayini-v-2021-rotsi>.
30. Виробництво зернових та зернобобових культур [Електронний ресурс] // Державна служба статистики України. – 2012. – Режим доступу до ресурсу: 11. https://ukrstat.gov.ua/SIMap/map.html?m=0000000000&t=13_7.
31. Індекс обсягу сільськогосподарського виробництва [Електронний ресурс] // Державна служба статистики України. – 2012. – Режим доступу до ресурсу: 1. https://ukrstat.gov.ua/SIMap/map.html?m=0000000000&t=13_6.
32. Universal Database Tool [Електронний ресурс] – Режим доступу до ресурсу: <https://dbeaver.io/>.
33. React D3 Tree [Електронний ресурс] – Режим доступу до ресурсу: <https://www.npmjs.com/package/react-d3-tree>.
34. Graintrack [Електронний ресурс] – Режим доступу до ресурсу: <https://graintrack.com/>.
35. BAS Бухгалтерія [Електронний ресурс] – Режим доступу до ресурсу: <https://www.bas-soft.eu/soft/bas-mass/bas-accounting/>.
36. Ubuntu Server [Електронний ресурс] – Режим доступу до ресурсу: <https://ubuntu.com/download/server>.
37. Docker Compose overview [Електронний ресурс] – Режим доступу до ресурсу: <https://docs.docker.com/compose/>.
38. The State of Food and Agriculture 2019 [Електронний ресурс] / 1 – Режим доступу до ресурсу: <https://www.fao.org/3/ca6030en/ca6030en.pdf>.
39. Grain: World Markets and Trade [Електронний ресурс] – Режим доступу до ресурсу: <https://apps.fas.usda.gov/psdonline/circulars/grain.pdf>.
40. Principles of Grain Marketing: Some Lessons from Australian Experience [Електронний ресурс] – Режим доступу до ресурсу: https://www.aciar.gov.au/sites/default/files/legacy/node/2147/tr38_pdf_20931.pdf.
41. Optimization Models for Training Belief-Rule-Based Systems [Електронний ресурс] – Режим доступу до ресурсу: https://personalpages.manchester.ac.uk/staff/jian-bo.yang/JB%20Yang%20Journal_Papers/SMCA_YLXWW_08_RIMER_Learning.pdf.

42. Combination of Evidence in DempsterShafer Theory [Электронный ресурс] –
Режим доступа до ресурсу:
https://www.stat.berkeley.edu/~aldous/Real_World/dempster_shafer.pdf

ДОДАТКИ

Додаток А Фрагменти JSON-об'єктів бази знань

```
{  
  "name": "Crop Advisor ruleset",  
  "attributes": [  
    {  
      "name": "Soil Type",  
      "type": "string"  
    },  
    {  
      "name": "Precipitation",  
      "type": "number"  
    },  
    {  
      "name": "Irrigation",  
      "type": "string"  
    },  
    {  
      "name": "Avg. Temp in summer",  
      "type": "number"  
    },  
    {  
      "name": "Avg. Temp in autumn",  
      "type": "number"  
    },  
    {  
      "name": "Avg. Temp in spring",  
      "type": "number"  
    },  
    {
```

```
        "name": "Altitude",
        "type": "number"
    },
    {
        "name": "Labour resources",
        "type": "string"
    },
    {
        "name": "Mechanization",
        "type": "string"
    },
    {
        "name": "Include market factors",
        "type": "string"
    },
    {
        "name": "State benefits",
        "type": "string"
    },
    {
        "name": "Previous crop",
        "type": "string"
    }
],
"decisions": [
    {
        "conditions": {
            "all": [
                {
                    "fact": "Soil Type",
                    "operator": "equal",
```

```
        "value": "Sandy Soils"
      }
    ]
  },
  "event": {
    "type": "Bad soil for any crop"
  }
},
{
  "conditions": {
    "all": [
      {
        "fact": "Altitude",
        "operator": "greaterThanInclusive",
        "value": 1500
      }
    ]
  },
  "event": {
    "type": "Too high to farming"
  }
},
{
  "conditions": {
    "all": [
      {
        "fact": "Soil Type",
        "operator": "equal",
        "value": "Stony Soils"
      }
    ]
  }
}
```

```
    },
    "event": {
      "type": "Bad soil for any crop"
    }
  },
  {
    "conditions": {
      "all": [
        {
          "fact": "Soil Type",
          "operator": "equal",
          "value": "Sandy Soils"
        },
        {
          "fact": "Precipitation",
          "operator": "lessThan",
          "value": 300
        },
        {
          "fact": "Irrigation",
          "operator": "equal",
          "value": "No"
        }
      ]
    },
    "event": {
      "type": "Not enough precipitation"
    }
  },
  {
    "conditions": {
```

```
"all": [  
  {  
    "fact": "Soil Type",  
    "operator": "notEqual",  
    "value": "Sandy Soils"  
  },  
  {  
    "fact": "Soil Type",  
    "operator": "notEqual",  
    "value": "Stony Soils"  
  },  
  {  
    "fact": "Precipitation",  
    "operator": "greaterThan",  
    "value": 300  
  },  
  {  
    "fact": "Precipitation",  
    "operator": "lessThan",  
    "value": 500  
  },  
  {  
    "fact": "Avg. Temp in summer",  
    "operator": "greaterThanInclusive",  
    "value": 20  
  },  
  {  
    "fact": "Avg. Temp in summer",  
    "operator": "lessThanInclusive",  
    "value": 24  
  },  
]
```

```

        {
            "fact": "Previous crop",
            "operator": "notEqual",
            "value": "Lentil"
        }
    ]
},
"event": {
    "type": "Lentil",
    "params": {}
}
},
{
    "conditions": {
        "all": [
            {
                "any": [
                    {
                        "all": [
                            {
                                "fact": "Precipitation",
                                "operator":
"greaterThanInclusive",
                                "value": 400
                            },
                            {
                                "fact": "Precipitation",
                                "operator":
"lessThanInclusive",
                                "value": 700
                            }
                        ]
                    }
                ]
            }
        ]
    }
}
]

```

```

    },
    {
      "all": [
        {
          "fact": "Irrigation",
          "operator": "equal",
          "value": "Yes"
        },
        {
          "fact": "Precipitation",
          "operator":
"lessThanInclusive",
          "value": 700
        }
      ]
    }
  ],
},
{
  "all": [
    {
      "fact": "Avg. Temp in summer",
      "operator": "greaterThanInclusive",
      "value": 19
    },
    {
      "fact": "Avg. Temp in summer",
      "operator": "lessThanInclusive",
      "value": 25
    },
    {
      "fact": "Avg. Temp in autumn",

```

```
        "operator": "greaterThanInclusive",
        "value": 9
    },
    {
        "fact": "Avg. Temp in spring",
        "operator": "greaterThanInclusive",
        "value": 8
    }
]
},
{
    "fact": "Previous crop",
    "operator": "notEqual",
    "value": "Corn"
},
{
    "fact": "Soil Type",
    "operator": "notEqual",
    "value": "Sandy soil"
},
{
    "fact": "Soil Type",
    "operator": "notEqual",
    "value": "Stony Soil"
}
]
},
"event": {
    "type": "Corn",
    "params": {}
}
```

```

    },
    {
      "conditions": {
        "all": [
          {
            "any": [
              {
                "all": [
                  {
                    "fact": "Precipitation",
                    "operator":
"greaterThanInclusive",
                    "value": 350
                  },
                  {
                    "fact": "Precipitation",
                    "operator":
"lessThanInclusive",
                    "value": 600
                  }
                ]
              },
              {
                "all": [
                  {
                    "fact": "Irrigation",
                    "operator": "equal",
                    "value": "Yes"
                  },
                  {
                    "fact": "Precipitation",

```

```

"lessThanInclusive",
    "operator":
    "value": 600
    }
]
}
]
},
{
    "all": [
        {
            "fact": "Avg. Temp in summer",
            "operator": "greaterThanInclusive",
            "value": 18
        },
        {
            "fact": "Avg. Temp in summer",
            "operator": "lessThanInclusive",
            "value": 26
        },
        {
            "fact": "Avg. Temp in autumn",
            "operator": "greaterThanInclusive",
            "value": 8
        },
        {
            "fact": "Avg. Temp in spring",
            "operator": "greaterThanInclusive",
            "value": 7
        }
    ]
},

```

```
    {
      "fact": "Previous crop",
      "operator": "notEqual",
      "value": "Wheat"
    },
    {
      "fact": "Soil Type",
      "operator": "notEqual",
      "value": "Sandy soil"
    },
    {
      "fact": "Soil Type",
      "operator": "notEqual",
      "value": "Stony Soil"
    }
  ]
},
"event": {
  "type": "Wheat",
  "params": {}
}
]
}
```

Додаток Б Фрагменти програмного коду інференційного рушія

```
var Engine = function (_EventEmitter) {  
  _inherits(Engine, _EventEmitter);  
  
  function Engine() {  
    var rules = arguments.length > 0 && arguments[0] !== undefined ?  
arguments[0] : [];  
    var options = arguments.length > 1 && arguments[1] !== undefined ?  
arguments[1] : {};  
  
    _classCallCheck(this, Engine);  
  
    var _this = _possibleConstructorReturn(this, (Engine.__proto__ ||  
Object.getPrototypeOf(Engine)).call(this));  
  
    _this.rules = [];  
    _this.allowUndefinedFacts = options.allowUndefinedFacts || false;  
    _this.pathResolver = options.pathResolver;  
    _this.operators = new Map();  
    _this.facts = new Map();  
    _this.status = READY;  
    rules.map(function (r) {  
      return _this.addRule(r);  
    });  
  }  
};
```

```
_engineDefaultOperators2.default.map(function (o) {  
  return _this.addOperator(o);  
});  
return _this;  
}
```

```
_createClass(Engine, [{  
  key: 'addRule',  
  value: function addRule(properties) {  
    if (!properties) throw new Error('Engine: addRule() requires options');  
  
    var rule = void 0;  
  
    if (properties instanceof _rule2.default) {  
      rule = properties;  
    } else {  
      if (!Object.prototype.hasOwnProperty.call(properties, 'event')) throw new  
Error('Engine: addRule() argument requires "event" property');  
  
      if (!Object.prototype.hasOwnProperty.call(properties, 'conditions')) throw  
new Error('Engine: addRule() argument requires "conditions" property');  
  
      rule = new _rule2.default(properties);  
    }  
  
    rule.setEngine(this);  
  }  
}]
```

```

    this.rules.push(rule);

    this.prioritizedRules = null;

    return this;
}

}, {

    key: 'updateRule',

    value: function updateRule(rule) {

        var ruleIndex = this.rules.findIndex(function (ruleInEngine) {

            return ruleInEngine.name === rule.name;

        });

        if (ruleIndex > -1) {

            this.rules.splice(ruleIndex, 1);

            this.addRule(rule);

            this.prioritizedRules = null;

        } else {

            throw new Error('Engine: updateRule() rule not found');

        }

    }

}, {

    key: 'removeRule',

    value: function removeRule(rule) {

```

```

var ruleRemoved = false;

if (!(rule instanceof _rule2.default)) {
  var filteredRules = this.rules.filter(function (ruleInEngine) {
    return ruleInEngine.name !== rule;
  });
  ruleRemoved = filteredRules.length !== this.rules.length;
  this.rules = filteredRules;
} else {
  var index = this.rules.indexOf(rule);
  if (index > -1) {
    ruleRemoved = Boolean(this.rules.splice(index, 1).length);
  }
}

if (ruleRemoved) {
  this.prioritizedRules = null;
}

return ruleRemoved;
}

}, {
  key: 'addOperator',
  value: function addOperator(operatorOrName, cb) {
    var operator = void 0;

```

```

if (operatorOrName instanceof _operator2.default) {
    operator = operatorOrName;
} else {
    operator = new _operator2.default(operatorOrName, cb);
}

(0, _debug2.default)('engine::addOperator name:' + operator.name);
this.operators.set(operator.name, operator);
}

}, {
    key: 'removeOperator',
    value: function removeOperator(operatorOrName) {
        var operatorName = void 0;

        if (operatorOrName instanceof _operator2.default) {
            operatorName = operatorOrName.name;
        } else {
            operatorName = operatorOrName;
        }

        return this.operators.delete(operatorName);
    }
}

```

```
}, {  
  key: 'addFact',  
  value: function addFact(id, valueOrMethod, options) {  
    var factId = id;  
    var fact = void 0;  
    if (id instanceof _fact2.default) {  
      factId = id.id;  
      fact = id;  
    } else {  
      fact = new _fact2.default(id, valueOrMethod, options);  
    }  
    (0, _debug2.default)('engine::addFact id:' + factId);  
    this.facts.set(factId, fact);  
    return this;  
  }  
}
```

```
}, {  
  key: 'removeFact',  
  value: function removeFact(factOrId) {  
    var factId = void 0;  
    if (!(factOrId instanceof _fact2.default)) {  
      factId = factOrId;  
    } else {
```

```
factId = factOrId.id;
```

```
}
```

```
return this.facts.delete(factId);
```

```
}
```

```
}, {
```

```
key: 'prioritizeRules',
```

```
value: function prioritizeRules() {
```

```
if (!this.prioritizedRules) {
```

```
var ruleSets = this.rules.reduce(function (sets, rule) {
```

```
var priority = rule.priority;
```

```
if (!sets[priority]) sets[priority] = [];
```

```
sets[priority].push(rule);
```

```
return sets;
```

```
}, {});
```

```
this.prioritizedRules = Object.keys(ruleSets).sort(function (a, b) {
```

```
return Number(a) > Number(b) ? -1 : 1; // order highest priority -> lowest
```

```
}).map(function (priority) {
```

```
return ruleSets[priority];
```

```
});
```

```
}
```

```
return this.prioritizedRules;
```

```
}
```

Ім'я користувача: Інформаційних систем в економіці Шкуратовська Те... ID перевірки: 1015132062
Дата перевірки: 17.05.2023 15:26:00 +03 Тип перевірки: Doc vs Internet + Library
Дата звіту: 17.05.2023 15:35:55 +03 ID користувача: 100005745

Назва документа: СШІ_Калашніков

Кількість сторінок: 95 Кількість слів: 15762 Кількість символів: 123073 Розмір файлу: 13.77 MB ID файлу: 1014813427

4.9% Схожість

Найбільша схожість: 2.47% з джерелом з Бібліотеки (ID файлу: 1008302424)

1.26% Джерела з Інтернету 274 Сторінка 97

4.83% Джерела з Бібліотеки 378 Сторінка 98

0% Цитат

Вилучення цитат вимкнене

Вилучення списку бібліографічних посилань вимкнене

0% Вилучень

Немає вилучених джерел

Модифікації

Виявлено модифікації тексту. Детальна інформація доступна в онлайн-звіті.

Замінені символи 1

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ ЕКОНОМІЧНИЙ УНІВЕРСИТЕТ
ІМЕНІ ВАДИМА ГЕТЬМАНА
навчально-науковий інститут
«Інститут інформаційних технологій в економіці»
кафедра інформаційних систем в економіці

галузь знань 12 Інформаційні технології
спеціальність 122 Комп'ютерні науки
освітня програма Системи штучного інтелекту

форма навчання: денна

КВАЛІФІКАЦІЙНА МАГІСТЕРСЬКА РОБОТА

на тему:

ІНТЕЛЕКТУАЛЬНА СИСТЕМА ПРИЙНЯТТЯ РІШЕНЬ В СФЕРІ
ЗЕРНОТРЕЙДИНГУ

здобувача Калашнікова Кирила Станіславовича _____

Науковий керівник: к.е.н., доцент

_____ Лозовик Ю.М.

Робота допущена до захисту перед
екзаменаційною комісією з атестації
здобувачів вищої освіти (ЕК)
в.о. завідувача кафедри: к.е.н., доцент

_____ Тішков Б.О.