

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Чорноморський національний університет імені Петра Могили
Факультет комп'ютерних наук
Кафедра інженерії програмного забезпечення

ДОПУЩЕНО ДО ЗАХИСТУ

Завідувач кафедри інженерії
програмного забезпечення

_____ Євген ДАВИДЕНКО

«__» _____ 2025 р.

КВАЛІФІКАЦІЙНА РОБОТА
НА ЗДОБУТТЯ ОСВІТНЬОГО СТУПЕНЯ МАГІСТРА
ІНТЕЛЕКТУАЛЬНА СИСТЕМА ПРОГНОЗУВАННЯ ВРОЖАЙНОСТІ
КУЛЬТУР

Спеціальність 121 Інженерія програмного забезпечення
Освітня програма «Інженерія програмного забезпечення»

Здобувачка

Ірина ЦЕЛИЩЕВА

«__» _____ 2025__ р.

Керівник роботи

д-рка техн. наук,

професорка

Альона ШВЕД

«__» _____ 2025__ р.

Миколаїв – 2025

Завдання на виконання кваліфікаційної роботи

Чорноморський національний університет імені Петра Могили

Факультет	Комп'ютерних наук
Кафедра	Інженерії програмного забезпечення
Рівень вищої освіти	Другий (магістерський)
Освітній ступінь	Магістр
Спеціальність	121 Інженерія програмного забезпечення
Освітня програма	Інженерія програмного забезпечення

ЗАТВЕРДЖУЮ

Завідувач кафедри інженерії
програмного забезпечення

_____ Євген ДАВИДЕНКО

«___» _____ 2025 р.

ЗАВДАННЯ

на кваліфікаційну магістерську роботу здобувача вищої освіти

Целищева Ірина

1. Тема кваліфікаційної роботи «Інтелектуальна система прогнозування врожайності культур» затверджена наказом ректора ЧНУ ім. Петра Могили № 182 від « 02 » _____ липня _____ 2025 р.

2. Строк представлення кваліфікаційної роботи «___» _____ 2025 р.

3. Очікуваний результат роботи та початкові дані якщо такі потрібні.

Очікуваним результатом є інтелектуальна система прогнозування врожайності культур.

4. Перелік питань, що підлягають розробці:

- визначення ключових факторів, що впливають на точність прогнозів;
- побудова моделі прогнозування;

- дослідження алгоритмів аналізу часових рядів та їхню ефективність;
- побудова математичної моделі прогнозування врожайності;
- проєктування, програмна реалізація, тестування та оцінювання ефективності інтелектуальної системи прогнозування врожайності.

5. Перелік графічних матеріалів: презентація.

6. Консультанти:

Консультант	Кафедра (організація)	Частина роботи

Дата видачі завдання «__01__» ____вересня_____ 2025_ р.

КАЛЕНДАРНИЙ ПЛАН
виконання кваліфікаційної роботи

Тема: **Інтелектуальна система прогнозування врожайності культур**

№	Найменування роботи	Початок	Закінчення	Примітки
1.	Розробка та затвердження завдання на виконання КМР	01.09.2025	03.09.2025	виконано
2.	Огляд літератури за темою роботи	03.09.2025	05.09.2025	виконано
3.	Складання календарного плану КМР	09.09.2025	10.09.2025	виконано
4.	Аналіз предметної області	11.09.2025	12.09.2025	виконано
5.	Розробка проєктних рішень	15.09.2025	19.09.2025	виконано
6.	Моделювання та конструювання ПЗ	22.09.2025	30.09.2025	виконано
7.	Кодування, тестування та апробація розробленого ПЗ, аналіз результатів тестування, розробка керівництва користувача	01.10.2025	17.11.2025	виконано
8.	Відгук керівника КМР	17.11.2025	18.11.2025	виконано
9.	Оформлення КМР та презентації	19.11.2025	21.11.2025	виконано
10.	Попередній захист	24.11.2025	24.11.2025	виконано
11.	Рецензування	17.12.2025	17.12.2025	виконано
12.	Завершення оформлення КМР та презентації	20.12.2025	20.12.2025	виконано
13.	Захист кваліфікаційної роботи	22.12.2025	22.12.2025	виконано

Здобувачка

Ірина ЦЕЛИЩЕВА

«__» _____ 2025__ р.

Керівник роботи

д-рка техн. наук,

професорка

Альона ШВЕД

«__» _____ 2025__ р.

АНОТАЦІЯ

до кваліфікаційної магістерської роботи

Інтелектуальна система прогнозування врожайності культур

Здобувачка 608 гр.: Целищева Ірина

Керівник: професорка, докторка технічних наук Швед Альона

Актуальність. Сучасні технології прогнозування врожайності, засновані на аналізі часових рядів і методах машинного навчання, набувають особливого значення в умовах кліматичних змін і зростаючого попиту на продовольство. Використання штучного інтелекту дозволяє автоматизувати процеси, мінімізувати людський фактор і підвищити точність прогнозів, що робить розробку інтелектуальної системи прогнозування врожайності актуальною та перспективною. Для її реалізації застосовується Python та спеціалізовані бібліотеки для аналізу даних і машинного навчання, які забезпечують ефективну обробку великих обсягів інформації та створення точних моделей.

Об'єкт. Процес прогнозування динаміки зміни рівня врожайності сільськогосподарських культур.

Предмет. Методи машинного навчання та аналізу часових рядів у прогнозуванні врожайності сільськогосподарських культур.

Мета. Оптимізація роботи виробництва за рахунок моделювання та передбачення рівня врожайності сільськогосподарських культур, враховуючи вплив факторів на продуктивність культури.

Кваліфікаційна робота складається із вступу, 4 розділів, висновків та переліку джерел посилання.

У вступі обґрунтовано актуальність проблеми прогнозування врожайності сільськогосподарських культур в умовах кліматичних змін та обмежених ресурсів малих фермерських господарств. Визначено мету роботи, що полягає у розробці інтелектуальної системи прогнозування врожайності на основі поєднання статистичних методів та алгоритмів машинного навчання, а також окреслено основні завдання дослідження.

У першому розділі подано аналіз предметної області та існуючих програмних рішень для прогнозування врожайності. Розглянуто сучасні методологічні підходи – від класичних статистичних моделей до методів машинного та глибинного навчання.

Другий розділ присвячено розробці проєктних рішень інтелектуальної системи прогнозування врожайності культур. Запропоновано загальну концепцію та архітектуру системи, яка поєднує модулі збору та обробки даних, прогнозування й візуалізації результатів.

У третьому розділі описано методи прогнозування врожайності, що поєднують статистичні моделі та машинне навчання. Використано SARIMA для короткострокового прогнозу часових рядів та Gradient Boosting для оцінювання впливу агрохімічних і метеорологічних факторів. Також наведено підхід до обробки даних, відновлення пропусків і оцінювання точності моделей за допомогою відповідних метрик.

У четвертому розділі представлено процес розробки та тестування вебзастосунку, призначеного для прогнозування врожайності культур. Описано архітектуру системи, алгоритми обробки даних і моделі прогнозування, наведено результати їх роботи та проведену оцінку точності. Також подано керівництво користувача, що демонструє послідовність взаємодії з інтерфейсом і підтверджує практичну придатність створеного програмного забезпечення.

У висновках проводиться аналіз роботи та отриманих результатів.

Кваліфікаційна робота викладена на 93 сторінках машинописного тексту, складається із вступу, 4 розділів, загальних висновків, переліку джерел посилання з 36 найменувань та 1 додатка. Праця містить 3 таблиць та 47 рисунків.

Ключові слова: прогнозування врожайності, машинне навчання, глибинне навчання, часові ряди, SARIMA, Gradient Boosting, інтелектуальні системи.

ABSTRACT

to the qualifying master's thesis

Intelligent crop yield forecasting system

Student of 608 group: Tselyshcheva Iryna

Supervisor: Doctor of Technical Sciences, professor Shved Alyona

Relevance. Modern yield forecasting technologies based on time series analysis and machine learning methods are becoming particularly important in the context of climate change and growing demand for food. The use of artificial intelligence allows processes to be automated, the human factor to be minimised, and the accuracy of forecasts to be improved, making the development of an intelligent yield forecasting system relevant and promising. Python and specialised libraries for data analysis and machine learning are used to implement it, ensuring the efficient processing of large amounts of information and the creation of accurate models.

Object. The process of forecasting changes in crop yield dynamics.

Subject. Machine learning and time series analysis methods in crop yield forecasting.

Objective. Optimisation of production through modelling and forecasting crop yields, taking into account the impact of factors on crop productivity.

The qualification work consists of an introduction, 4 sections, conclusions and a list of references.

In the introduction justifies the relevance of the problem of forecasting crop yields in the context of climate change and limited resources of small farms. The aim of the thesis is to develop an intelligent crop yield forecasting system based on a combination of statistical methods and machine learning algorithms, and the main research tasks are outlined.

In the first section presents an analysis of the subject area and existing software solutions for yield forecasting. Modern methodological approaches are considered, from classical statistical models to machine learning and deep learning methods.

The second section is devoted to the development of design solutions for an intelligent crop yield forecasting system. A general concept and architecture of the system

is proposed, combining modules for data collection and processing, forecasting and visualisation of results.

The third chapter describes methods for forecasting crop yields that combine statistical models and machine learning. SARIMA is used for short-term time series forecasting, and Gradient Boosting is used to assess the impact of agrochemical and meteorological factors. It also presents an approach to data processing, gap filling, and model accuracy assessment using appropriate metrics.

The fourth chapter presents the process of developing and testing a web application designed to forecast crop yields. The architecture of the system, data processing algorithms, and forecasting models are described, and the results of their work and accuracy assessment are presented. A user guide is also provided, demonstrating the sequence of interaction with the interface and confirming the practical applicability of the created software.

The conclusions analyze the work and the results obtained.

The qualification work is presented on 93 pages of typewritten text, consists of an introduction, 4 sections, general conclusions, a list of references with 36 titles and 1 appendice. The work contains 3 tables and 46 figures.

Keywords: yield forecasting, machine learning, deep learning, time series, SARIMA, Random Forest, Gradient Boosting, intelligent systems.

ЗМІСТ

ПЕРЕЛІК СКОРОЧЕНЬ.....	4
ВСТУП.....	5
1 АНАЛІЗ МЕТОДІВ ТА ПРОГРАМНИХ ЗАСОБІВ ВИРІШЕННЯ ЗАДАЧІ ПРОГНОЗУВАННЯ ВРОЖАЙНОСТІ КУЛЬТУР	7
1.1 Аналіз предметної області	7
1.2 Аналіз наявних програмних рішень.....	9
1.3 Методологічні особливості прогнозування врожайності	15
1.4 Обґрунтування та вибір підходів.....	18
Висновки до розділу 1	19
2 ФУНКЦІОНАЛЬНІ ТА МАТЕМАТИЧНІ МОДЕЛІ ІНТЕЛЕКТУАЛЬНОЇ СИСТЕМИ ПРОГНОЗУВАННЯ	22
2.1 Методи прогнозування врожайності.....	22
2.2 Загальна концепція системи.....	27
2.3 Функціональна модель системи	28
2.4 Алгоритм прогнозування врожайності.....	30
2.5 Специфікація вимог до інформаційної системи прогнозування врожайності.....	34
Висновки до розділу 2	44
3 ПРОЄКТУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ СИСТЕМИ ПРОГНОЗУВАННЯ ВРОЖАЙНОСТІ	46
3.1 Архітектура програмного забезпечення	46
3.2 Розробка компонентів програмного забезпечення	48
3.3 Діаграма варіантів використання інтелектуальної системи	53
3.4 Діаграма діяльності інтелектуальної системи.....	54
3.5 Діаграма послідовності інтелектуальної системи	56
3.6 Інформаційне забезпечення	58
Висновки до розділу 3	62

4 РОЗРОБКА ТА ТЕСТУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ.....	63
4.1 Вибір технологій та мов програмування	63
4.2 Архітектурно-функціональний опис програмного забезпечення	66
4.3 Діаграма класів інтелектуальної системи.....	67
4.3 Опис основного функціоналу інтелектуальної системи	71
4.4 Аналіз результатів прогнозування	73
4.5 Результати рішення.....	77
4.6 Керівництво користувача	80
Висновки до розділу 4	88
ВИСНОВКИ.....	90
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ.....	92
ДОДАТОК А Апробація кваліфікаційної роботи	96
ДОДАТОК Б Програмна реалізація прогнозування	97

ПЕРЕЛІК СКОРОЧЕНЬ

ІТ – інформаційні технології

ПЗ – програмне забезпечення

ОС – операційна система

ІПЗ – інженерія програмного забезпечення

AI – Artificial Intelligence

API – application programming interface

AR – Autoregressive

CSS – cascading style sheets

DL – Deep Learning

HTML – hypertext markup language

JS – JavaScript

MA – Moving Average

ML – Machine Learning

MVC – model-view-controller

ORM – Object-Relational Mapping

UML – Unified Modeling Language

SVM – Support Vector Machines

SQL – Structured Query Language

ВСТУП

Актуальність. Сучасні технології прогнозування врожайності стають дедалі важливішими в умовах глобальних змін клімату та зростаючого попиту на продовольство. Аналіз часових рядів та застосування методів машинного навчання дозволяють суттєво підвищити точність прогнозів, що сприяє ефективному управлінню агропромисловими ресурсами.

Розвиток штучного інтелекту та його впровадження у сільське господарство дозволяє не лише автоматизувати процеси, але й мінімізувати людський фактор у прийнятті рішень. У світлі цих змін розробка інтелектуальної системи прогнозування врожайності є актуальною та перспективною, оскільки забезпечує адаптивність та точність прогнозів.

Для реалізації поставленої задачі використовується мова програмування Python та бібліотеки для аналізу даних і машинного навчання, що забезпечують ефективну обробку великих обсягів інформації та побудову точних моделей прогнозування.

Об'єктом дослідження є процес прогнозування динаміки зміни рівня врожайності сільськогосподарських культур.

Предметом дослідження є методи машинного навчання та аналізу часових рядів у прогнозуванні врожайності сільськогосподарських культур.

Метою роботи є оптимізація роботи виробництва за рахунок моделювання та передбачення рівня врожайності сільськогосподарських культур, враховуючи вплив факторів (сортів рослин, добрива, лімітуючі фактори, технологія вирощування тощо) на продуктивність культури.

Завдання, які необхідно вирішити для реалізації проєкту:

1. провести аналіз сучасних методів прогнозування врожайності сільськогосподарських культур;
2. дослідити алгоритми аналізу часових рядів та їхню ефективність у прогнозуванні врожайності;
3. визначити ключові фактори, що впливають на точність прогнозів;

4. побудувати математичну модель прогнозування врожайності;
5. проектування та програмна реалізація ІС прогнозування врожайності;
6. розробити графічний інтерфейс для взаємодії користувача із системою;
7. провести тестування та оцінку ефективності розробленої системи.

Апробація результатів КМР відбулась під час XXVIII Всеукраїнської науково-практичної конференції «Могилянські читання – 2025», Миколаїв, 10–14 листопада, 2025 р. (Додаток А).

1 АНАЛІЗ МЕТОДІВ ТА ПРОГРАМНИХ ЗАСОБІВ ВИРІШЕННЯ ЗАДАЧІ ПРОГНОЗУВАННЯ ВРОЖАЙНОСТІ КУЛЬТУР

1.1 Аналіз предметної області

Прогнозування врожайності культур є критично важливою задачею в аграрному секторі, оскільки воно забезпечує основу для стратегічного й оперативного планування. Точні прогнози дають можливість фермерам, аграрним компаніям, науковим установам та державним органам своєчасно приймати обґрунтовані рішення щодо управління виробництвом, оптимізації використання ресурсів і забезпечення продовольчої безпеки на національному та регіональному рівнях. Своєчасна інформація про очікуваний урожай допомагає коригувати закупівлі ресурсів, планувати логістичні операції, оцінювати економічні ризики та своєчасно реагувати на можливі загрози, пов'язані з погодними умовами чи фітосанітарними факторами [1]. Зокрема, якісні прогнози сприяють ефективному використанню добрив, води, насіння, засобів захисту рослин, а також зменшують ймовірність фінансових втрат у випадку несприятливих кліматичних умов або поширення хвороб культур.

Ключовими змінними, що визначають урожайність, є кліматичні, ґрунтові, технологічні та біологічні фактори. Серед кліматичних параметрів найвагомішими є температура повітря, кількість і розподіл опадів, інсоляція, вологість, дефіцит тиску пари, частота екстремальних явищ (посухи, заморозки, сильні вітри). ґрунтові характеристики включають гранулометричний склад, рівень органічної речовини, вміст макро- та мікроелементів, кислотність (pH), водопроникність, водоутримуючу здатність, структуру та щільність ґрунту [2]. Значну роль відіграють також агротехнологічні заходи: система сівозміни, способи обробітку ґрунту, норми та строки внесення добрив і пестицидів, методи й інтенсивність зрошення, вибір технології вирощування. Біологічні параметри культури – сорт, генетичний потенціал, стійкість до стресових факторів, фаза росту – визначають реакцію рослин на зовнішні впливи та формують їхній продуктивний потенціал [3].

Історичні дані про врожайність та експлуатаційні показники (урожай за попередні роки, сезонні дані щодо погодних умов, застосованих технологій, характеристик ґрунту) становлять основу для побудови моделей прогнозування. Дані часових рядів, що фіксують зміну характеристик у часі, дозволяють виявити закономірності у вигляді трендів, сезонних коливань, довгострокових циклів або випадкових відхилень. Аналіз таких структур дає змогу глибше зрозуміти взаємозв'язки між погодою, технологіями та врожайністю, а також виявити фактори, що мають найбільший вплив на продуктивність сільськогосподарських культур [4].

Окремо слід виділити аспект просторової неоднорідності. Навіть у межах одного поля ґрунтові та мікрокліматичні умови можуть суттєво відрізнятися, що впливає на ріст і розвиток рослин. На рівні регіонів відмінності можуть бути ще більш відчутними – різна кількість опадів, типи ґрунтів, рівень забезпеченості водними ресурсами, доступ до технологій тощо [5]. Це породжує необхідність адаптації моделей до локальних умов або застосування таких підходів, які здатні враховувати просторові закономірності й включати дані дистанційного зондування Землі. Супутникові знімки та дані з безпілотних літальних апаратів дають змогу оцінювати вегетаційні індекси, ступінь стресу культур, температуру поверхні ґрунту та інші параметри, що істотно підвищують точність прогнозів [6].

Не менш важливим є забезпечення високої якості даних, оскільки будь-які похибки, неточності або пропуски можуть суттєво погіршити результати прогнозування. Етап передобробки даних включає їх очищення, усунення аномалій, заповнення пропусків коректними значеннями (методами інтерполяції, регресії, машинного навчання), приведення показників до єдиних одиниць вимірювання, вирівнювання частот спостережень та масштабування (нормалізація або стандартизація). Часто проводиться також трансформація змінних – логарифмування, сезонні корекції, обчислення похідних показників. Без якісної передобробки навіть складні та високопродуктивні математичні моделі можуть давати нестабільні або хибні прогнози [7].

У підсумку, прогнозування врожайності є складною багатофакторною задачею, яка поєднує аналіз клімату, ґрунту, технологій, біології, історичних спостережень і просторових даних. Ефективність прогнозів значною мірою залежить як від вибору моделі, так і від якості даних, правильності їхньої підготовки та здатності моделі адаптуватися до реальних умов конкретного господарства або регіону.

1.2 Аналіз наявних програмних рішень

За останнє десятиліття у світі було розроблено значну кількість систем та програмних рішень для прогнозування врожайності сільськогосподарських культур. Ці системи відрізняються між собою методами моделювання, використаними джерелами даних, архітектурою та сферою застосування – від суто наукових моделей до комерційних платформ, інтегрованих у виробничі процеси агрокомпаній. Загальними тенденціями розвитку цього напрямку є активна інтеграція супутникових даних, біофізичних показників, кліматичних параметрів та алгоритмів машинного навчання. Такий підхід дозволяє не лише суттєво підвищувати точність прогнозів, а й масштабувати рішення для різних регіонів та агрокліматичних зон.

Одним із найбільш відомих прикладів є система Global Yield Gap Atlas (GYGA) [8] (рис. 1.1), яка забезпечує оцінку потенційного та фактичного рівня врожайності у різних країнах світу. Вона ґрунтується на комплексному поєднанні агрокліматичних даних, інформації про ґрунти, особливостей агротехнологій та моделей врожайності, таких як DSSAT чи APSIM. Дані GYGA використовуються на рівні урядових та міжнародних організацій, оскільки дозволяють формувати науково обґрунтовані стратегії аграрної політики, визначати ефективність виробництва та планувати розвиток аграрного сектору.

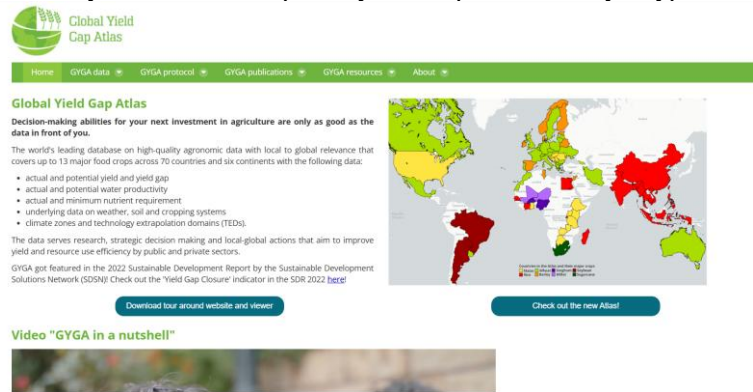


Рисунок 1.1 – Головна сторінка Global Yield Gap Atlas

У промисловому секторі значного поширення набули рішення, що поєднують супутникові спостереження та інтелектуальні алгоритми. Наприклад, платформа Climate FieldView (Bayer) [9] (рис. 1.2) надає аграріям доступ до прогнозів врожайності, аналітики стану полів та рекомендацій щодо управління посівами. Дані надходять із супутникових знімків, датчиків техніки, метеостанцій та історичних спостережень. Подібні сервіси, зокрема Ceres Imaging, спеціалізуються на моніторингу біофізичних параметрів рослин і мають широкий спектр інструментів для виявлення зон ризику, контролю зрошення та оцінки стресу культур.

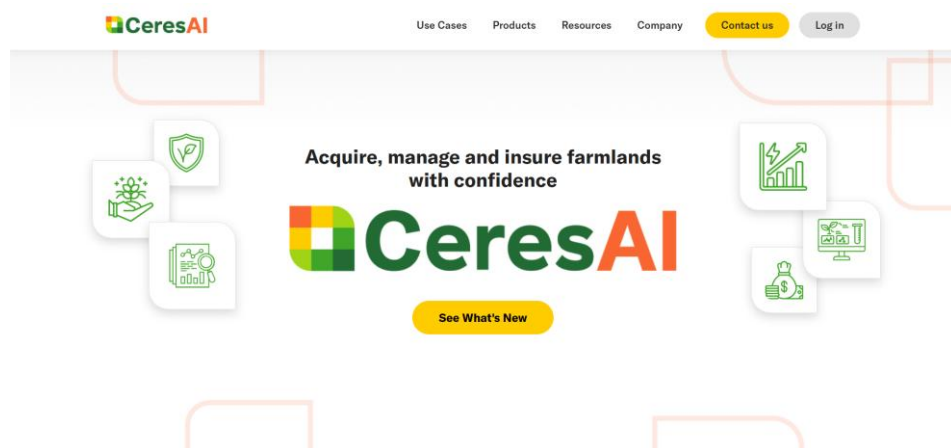


Рисунок 1.2 – Головна сторінка Climate FieldView

У межах концепції «розумного землеробства» (smart farming) активно розвиваються системи управління господарством, що включають модулі прогнозування врожайності. Такі платформи об'єднують дані з різних джерел – польових сенсорів, IoT-пристроїв, дронів, супутникових знімків, ґрунтових карт, 2025 р.

історичних даних та машинного навчання. Завдяки цьому сільськогосподарські виробники отримують не лише прогнози, але й рекомендації щодо оптимізації сівозмін, норм висіву, внесення добрив, управління зрошенням та планування польових робіт. Відомими прикладами є John Deere Operations Center [10] (рис. 1.3) та Granular (Corteva Agriscience) [11] (рис. 1.4), які інтегрують прогнозування у повноцінні системи підтримки прийняття рішень.

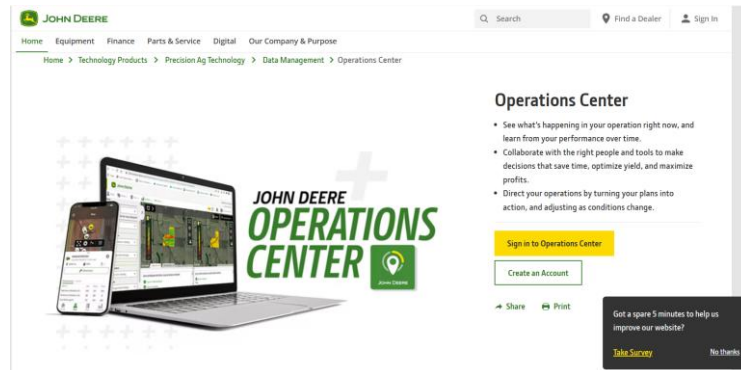


Рисунок 1.3 – Головна сторінка John Deere Operations Center

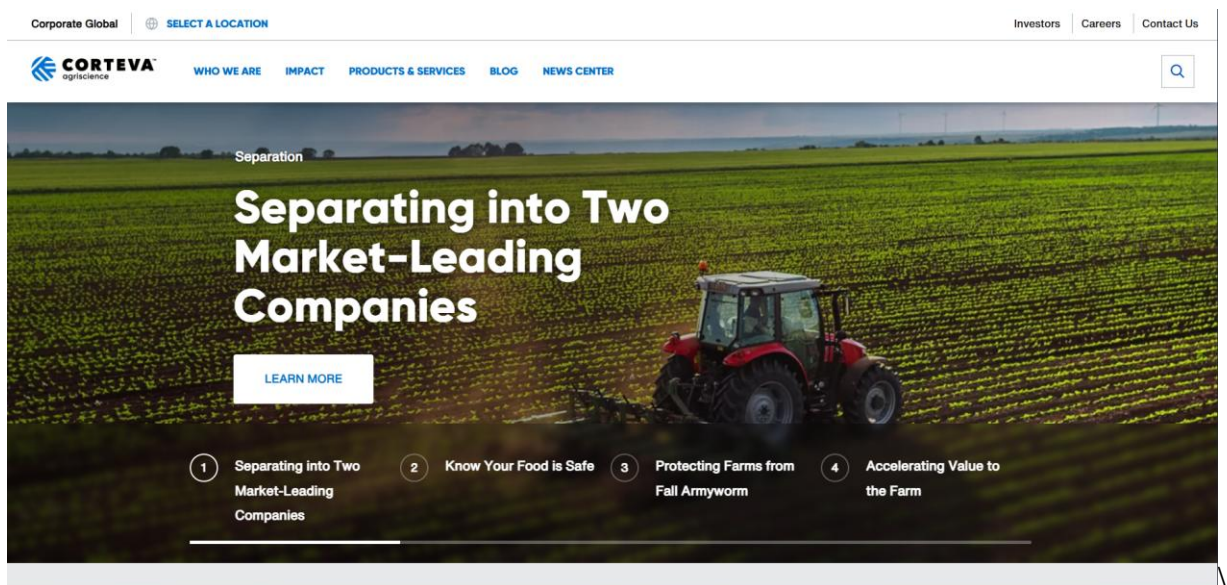


Рисунок 1.4 – Головна сторінка Granular (Corteva Agriscience)

Узагальнюючи, можна зазначити, що сучасні рішення зосереджені на використанні багатоджерельних даних та інтелектуальних алгоритмів, а також на створенні веборієнтованих систем, які роблять прогнозування доступним для широкого кола користувачів – від науковців до аграріїв практиків. Важливим

трендом також є перехід до персоналізованих рішень, які адаптуються до конкретних полів, культур і технологічних умов.

Для узагальнення проведеного аналізу сучасних підходів та програмних рішень у сфері прогнозування врожайності було сформовано порівняльну таблицю, яка відображає ключові характеристики найбільш поширених систем. До порівняння включено наукові моделі, комерційні платформи та комплексні рішення «розумного землеробства». Таблиці 1.1 та 1.2 дозволяють наочно визначити відмінності між системами за джерелами даних, використаними методами моделювання, функціональними можливостями та обмеженнями, що є важливим для вибору оптимального підходу залежно від умов та потреб користувача.

Таблиця 1.1 – Порівняння сучасних систем прогнозування врожайності

Система / Платформа	Тип рішення	Джерела даних	Методи моделювання
Global Yield Gap Atlas (GYGA)	Науково-аналітична платформа	Агрокліматичні дані, ґрунтові карти, моделі DSSAT/ APSIM, історичні дані	Біофізичні моделі врожайності
Climate FieldView (Bayer)	Комерційна аграрна платформа	Супутникові знімки, сенсори техніки, метеостанції, історичні дані	Машинне навчання, аналітичні моделі стану полів
Ceres Imaging	Комерційна система моніторингу	Супутникові та аерофотознімки, біофізичні індекси (NDVI, NDRE)	Аналіз спектральних даних, ML

Кінець таблиці 1.1

Система / Платформа	Тип рішення	Джерела даних	Методи моделювання
John Deere Operations Center	Система управління господарством (smart farming)	IoT-пристрої, техніка John Deere, супутникові дані, історичні спостереження	Аналітичні моделі, машинне навчання
Granular (Corteva)	Комерційна платформа управління агробізнесом	Дані сенсорів, метеоінформація, супутники, історичні дані	ML-моделі прогнозування та планування

Таблиця 1.2 – Переваги та недоліки систем прогнозування врожайності

Система / Платформа	Переваги	Недоліки
Global Yield Gap Atlas (GYGA)	Висока точність на глобальному рівні; використовується державними та міжнародними структурами; наукова обґрунтованість	Немає зручного вебінтерфейсу для аграріїв; складність адаптації до локальних умов; не налаштована на оперативне прогнозування
Climate FieldView (Bayer)	Зручний вебінтерфейс, комплексна аналітика, інтеграція обладнання, доступність рекомендацій	Висока вартість; орієнтація на великі господарства; обмежена адаптованість до умов Східної Європи
Ceres Imaging	Виявлення зон ризику, контроль зрошення, оцінка стресу рослин	Не є повноцінною системою прогнозування; висока вартість; потребує високоякісних зображень

Кінець таблиці 1.2

Система / Платформа	Переваги	Недоліки
John Deere Operations Center	Інтеграція з технікою, рекомендації для технологій, прогнозування в рамках великої системи	Орієнтовано на великі агрохолдинги; обмежена доступність для дрібних фермерів
Granular (Corteva)	Комплексний набір інструментів, підтримка бізнес-рішень	Висока вартість; адаптація під США та Західну Європу

Попри значний розвиток інтелектуальних систем у сфері прогнозування врожайності, більшість сучасних рішень мають певні обмеження, що знижує їхню ефективність у конкретних умовах. По-перше, глобальні комерційні системи (Climate FieldView, John Deere Operations Center, Granular тощо) є орієнтованими на великі агрохолдинги й потребують значних фінансових інвестицій, що робить їх малодоступними для невеликих фермерських господарств. По-друге, багато академічних моделей залишаються на рівні експериментів і не мають зручного вебінтерфейсу, що обмежує їх практичне застосування в реальному агробізнесі.

Попри значний прогрес, більшість сучасних систем мають ряд обмежень, що знижують їхню ефективність у локальному контексті. По-перше, глобальні комерційні платформи (Climate FieldView, John Deere Operations Center, Granular тощо) орієнтовані переважно на великі агрохолдинги й вимагають суттєвих фінансових інвестицій, що робить їх малодоступними для невеликих фермерських господарств. По-друге, багато академічних моделей обмежені лабораторним використанням і позбавлені інтуїтивного вебінтерфейсу, що ускладнює їхнє практичне застосування у реальному виробництві.

Також актуальною проблемою є адаптація моделей до локальних умов. Більшість систем розроблені з орієнтацією на США чи Західну Європу, тоді як

аграрні регіони Східної Європи та України мають інші кліматичні умови, структуру ґрунтів і специфіку агротехнологій. Це призводить до того, що навіть високоточні моделі, розроблені на інших даних, не забезпечують належної якості прогнозів у локальному середовищі.

Розробка програмного забезпечення, орієнтованого на місцеві умови, дозволяє не лише підвищити точність прогнозів врожайності, а й зробити такі технології доступними для малого та середнього агровиробника. Це повністю відповідає сучасним тенденціям цифрової трансформації аграрного сектору, сприяє раціональному використанню ресурсів та підвищенню конкурентоспроможності фермерських господарств у мінливих кліматичних умовах.

1.3 Методологічні особливості прогнозування врожайності

Процес прогнозування врожайності сільськогосподарських культур є складною багатофакторною задачею, що поєднує аналіз історичних даних, моніторинг поточного стану агроценозів та застосування сучасних математичних і статистичних методів. Побудова якісного прогнозу передбачає врахування як біологічних особливостей росту рослин, так і впливу зовнішніх чинників, серед яких ключову роль відіграють погодні умови, агротехнічні заходи та характеристики ґрунту [12]. Усі ці фактори взаємодіють між собою й формують кінцевий продуктивний потенціал культури, тому успішні прогностичні моделі повинні відображати цю взаємозалежність.

Першим етапом прогнозування є аналіз історичних даних, що включає статистичні спостереження за врожайністю в попередні роки, а також оцінку динаміки змін під впливом кліматичних коливань. Одним з найпоширеніших підходів на цьому етапі є застосування регресійного аналізу, який дозволяє встановити залежності між врожайністю та факторами середовища, такими як кількість опадів, температура повітря чи рівень удобрення [13]. За його допомогою можна визначити найбільш впливові фактори та оцінити еластичність врожаю щодо кожного з них.

Другим ключовим напрямом є моніторинг поточного стану рослин у процесі органогенезу. Різні фази розвитку культур характеризуються неоднаковою чутливістю до кліматичних і агротехнічних впливів. Наприклад, під час проростання критично важливими є температура та вологість ґрунту, у фазі цвітіння – забезпеченість вологою та поживними речовинами, а на етапі наливу зерна – температурний стрес і фотосинтетична активність. Регулярний моніторинг біометричних показників (висота рослин, густина рослинного покриву, індекс листової поверхні, індекс NDVI) дозволяє суттєво підвищити точність прогнозів, особливо на середньостроковому горизонті [14].

Оцінка погодних умов та рівня живлення культур є ще одним фундаментальним чинником прогнозування. Нестача опадів у критичні фази розвитку рослин, посуха, температурні стреси чи дефіцит азоту можуть призвести до різкого зниження врожайності. Навпаки, оптимальна кількість вологи, збалансоване живлення та помірні температури створюють умови для формування високого врожаю. На сучасному етапі для отримання даних широко використовуються як метеостанції та агрометеорологічні моделі, так і супутникові спостереження. Дані дистанційного зондування Землі дозволяють оцінювати вологозабезпечення, температуру поверхні ґрунту, інтенсивність фотосинтезу та виявляти стресові зони на полі [15].

Методологія прогнозування врожайності передбачає застосування різноманітних математичних і статистичних інструментів, що дозволяють відтворювати закономірності формування врожаю та враховувати вплив багатьох факторів. Одним з основних напрямів є використання регресійних моделей, які дають можливість кількісно описати залежність урожайності від кліматичних і агротехнічних умов. Такі моделі особливо ефективні у випадках, коли існує достатній обсяг достовірних історичних даних, що відображають багаторічні тенденції [16].

Важливу роль відіграють також статистичні методи, серед яких поширеним є метод екстраполяції. Він ґрунтується на припущенні, що тенденції, зафіксовані у попередні періоди, можуть бути поширені на майбутнє, за умови відносної

2025 р.

стабільності факторів середовища. Попри свою простоту, цей підхід часто використовується як базовий орієнтир у прогнозуванні врожайності, особливо в регіонах із відносно сталими кліматичними умовами [17].

Окрему групу становлять експертні оцінки, що базуються на практичному досвіді агрономів і спеціалістів у галузі. На відміну від суто математичних моделей, експертні системи дають змогу враховувати нетипові явища, локальні особливості ґрунтово-кліматичних умов та інші фактори, які складно формалізувати. Їхня перевага полягає у гнучкості та можливості швидкої адаптації до конкретної виробничої ситуації, однак недоліком є суб'єктивність та залежність від кваліфікації експертів [14].

Методологічні особливості прогнозування врожайності передбачають комплексний підхід, де поєднуються кількісні методи аналізу та якісні оцінки. Основними факторами, що впливають на результат, є: погодні умови (температурний режим, кількість опадів, сонячна інсоляція), ґрунтові характеристики (родючість, вологість, кислотність), агротехнічні заходи (система удобрення, сівозміна, обробіток ґрунту), а також біологічні особливості культур. Урахування цих складових забезпечує більш надійний прогноз і підвищує ефективність управлінських рішень у сільському господарстві [17].

Проведений аналіз підтверджує, що прогнозування врожайності сільськогосподарських культур є складною, багатфакторною задачею, яка потребує комплексного підходу та інтеграції різних джерел даних. Ефективне моделювання враховує як історичні спостереження за врожайністю, так і поточний стан агроценозів, погодні умови, характеристики ґрунту, агротехнічні заходи та біологічні особливості культур. Використання поєднання статистичних методів, математичних моделей та експертних оцінок дозволяє підвищити точність прогнозів і адаптувати їх до локальних умов виробництва. Таким чином, застосування комплексної методології прогнозування є ключовим чинником для прийняття обґрунтованих управлінських рішень, оптимізації ресурсів та забезпечення стабільної продуктивності сільськогосподарських культур.

1.4 Обґрунтування та вибір підходів

Прогнозування врожайності у малих сільськогосподарських господарствах України має низку специфічних особливостей. Ці господарства часто характеризуються обмеженими ресурсами, невеликими площами земельних ділянок, відсутністю складної агротехнічної інфраструктури та високою залежністю від кліматичних умов. Особливо гостро проблема прогнозування проявляється у регіонах із частими аномаліями погодних умов, такими як посуха, зливи або різкі коливання температури. В таких умовах класичні статистичні методи можуть бути недостатньо точними, оскільки вони часто передбачають стабільність і регулярність даних.

З огляду на ці фактори, для розробки інтелектуальної системи прогнозування врожайності для малих господарств доцільно обирати методи, які поєднують високу точність прогнозу з простотою реалізації та незначними вимогами до обчислювальних ресурсів.

Можливість роботи з невеликими обсягами даних – у малих господарствах часто відсутні багаторічні історичні записи врожайності, тому методи повинні давати прогнози навіть при обмеженій інформації.

Гнучкість до аномальних погодних умов – система повинна враховувати нестандартні кліматичні коливання і дозволяти оновлювати прогнози за актуальними даними.

Враховуючи зазначені критерії, найбільш доцільним та ефективним підходом є використання комбінованої методики, у якій для первинного прогнозування сезонних трендів застосовуються моделі ARIMA або SARIMA. Ці методи особливо добре підходять для аналізу часових рядів, оскільки здатні точно відображати базові закономірності та циклічні коливання врожайності, що формуються під впливом попередніх сезонів. Завдяки своїй гнучкості та можливості враховувати як тренд, так і сезонність, ARIMA та SARIMA дозволяють сформувати базовий прогноз, який виступає надійною основою для подальшої корекції та вдосконалення моделі.

Легкі ML-методи, такі як Random Forest або Gradient Boosting здатні враховувати нелінійні впливи кліматичних факторів, наприклад раптові посухи чи надмірні опади. Для малих господарств можна навчати моделі на невеликих наборах даних і обмеженій кількості ознак (температура, опади, тип ґрунту, попередній урожай), що спрощує реалізацію і використання.

Такий підхід дозволяє отримати адекватний прогноз при обмежених даних, мінімізувати ризик перенавчання моделей та забезпечити користувачам зрозумілий інтерфейс у вебдодатку. Користувачі зможуть легко вводити дані про своє господарство і отримувати прогноз врожайності без необхідності складного налаштування моделей, що відповідає принципам практичної агротехнології.

Таким чином, поєднання статистичних моделей для виявлення базових трендів розвитку та використання легких методів машинного навчання для додаткової корекції прогнозів під впливом аномальних чи нестандартних погодних умов виступає найбільш раціональним та оптимальним підходом для малих сільськогосподарських господарств України. Така інтегрована модель дозволяє не лише підвищити точність оцінювання потенційної врожайності, але й зберегти простоту реалізації, відсутність потреби у великих обчислювальних ресурсах та здатність системи швидко адаптуватися до змін зовнішнього середовища. Завдяки поєднанню цих компонентів господарства можуть отримувати більш надійні та гнучкі прогнози, що сприяє ефективнішому плануванню виробництва.

Висновки до розділу 1

Проведений аналіз предметної області дозволяє стверджувати, що прогнозування врожайності сільськогосподарських культур є одним із ключових елементів сучасного аграрного менеджменту, який визначає ефективність виробництва, раціональне використання ресурсів та стійкість фермерських господарств до зовнішніх ризиків. Для малих господарств України, які характеризуються обмеженими фінансовими та матеріальними ресурсами, високою залежністю від аномальних погодних умов та коливань ринку, системи прогнозування врожайності мають особливу практичну цінність. Високоточні

2025 р.

прогнози дозволяють фермерам своєчасно планувати посіви, визначати оптимальні строки сівби, ефективно використовувати добрива та воду, а також мінімізувати фінансові втрати у разі несприятливих кліматичних коливань чи поширення шкідників і хвороб.

Аналіз існуючих програмних рішень показав, що сучасні системи прогнозування врожайності, такі як Global Yield Gap Atlas, Climate FieldView, John Deere Operations Center та Granular, інтегрують великі обсяги даних різного типу, включно з супутниковими знімками, даними дистанційного зондування, метеорологічними показниками, польовими сенсорами та IoT-пристроями. Такі платформи забезпечують користувачів аналітикою стану полів, оцінкою біофізичних параметрів культур, прогнозами врожайності та рекомендаціями щодо управління посівами, внесення добрив та зрошення. Водночас більшість комерційних рішень орієнтовані на великі агрохолдинги і потребують значних фінансових та технічних ресурсів, що робить їх малодоступними для малих фермерських господарств. Адаптація глобальних моделей до локальних умов України обмежена через відмінності кліматичних зон, структури ґрунтів та специфіки агротехнологій, що підкреслює необхідність розробки спеціалізованих рішень для регіонального застосування.

Методичний аналіз існуючих підходів показав, що класичні статистичні методи, такі як ARIMA та SARIMA, добре відображають сезонні тренди та короткострокові залежності у часових рядах даних, проте вони не завжди здатні адекватно моделювати складні нелінійні взаємозв'язки між багатьма факторами, що впливають на врожайність. Сучасні методи машинного навчання, включно з Random Forest та Gradient Boosting, а також підходи глибинного навчання (LSTM, CNN), дозволяють враховувати складні просторово-часові взаємозв'язки та динаміку росту рослин, значно підвищуючи точність прогнозів. Однак широке застосування таких методів у малих господарствах обмежене через потребу у великих обчислювальних ресурсах, спеціалізованих навичках та значних обсягах даних, які часто недоступні на локальному рівні.

На основі проведеного аналізу доцільним є використання комбінованого підходу для малих господарств України. Поєднання класичних статистичних моделей для прогнозування базових сезонних трендів із легкими та ресурсозберігаючими методами машинного навчання для корекції прогнозів під впливом аномальних погодних умов дозволяє забезпечити баланс між точністю, простотою реалізації та мінімальними ресурсними витратами. Такий підхід дозволяє ефективно працювати з обмеженими наборами даних, адаптувати моделі до локальних кліматичних умов та забезпечити зручний і зрозумілий вебінтерфейс для користувача.

Отже, проведений аналіз підтверджує доцільність розробки інтелектуальної системи прогнозування врожайності, яка інтегрує статистичні та машинно-навчальні підходи з урахуванням специфіки малих господарств України, їх обмежених ресурсів та локальних кліматичних особливостей. Впровадження такої системи сприятиме підвищенню ефективності планування виробництва, зменшенню ризиків втрат урожаю та підвищенню економічної стійкості малих фермерських господарств у мінливих кліматичних умовах.

2 ФУНКЦІОНАЛЬНІ ТА МАТЕМАТИЧНІ МОДЕЛІ ІНТЕЛЕКТУАЛЬНОЇ СИСТЕМИ ПРОГНОЗУВАННЯ

2.1 Методи прогнозування врожайності

Прогнозування врожайності культур є складним багатофакторним завданням, яке потребує застосування різних методів аналізу даних. У сучасній науці виділяють кілька основних груп підходів: класичні статистичні методи, методи машинного навчання, глибинне навчання та гібридні методи, що поєднують сильні сторони різних підходів [18].

2.1.1 Статистичні методи

Статистичні методи прогнозування є одними з найдавніших та найпоширеніших у аграрних дослідженнях. Вони базуються на математичному аналізі історичних даних про врожайність та інших факторів (клімат, ґрунт, технології), що дозволяє оцінити тренди, сезонні коливання та випадкові коливання. Статистичні моделі зазвичай використовуються для короткострокових прогнозів та в умовах обмеженої кількості даних, коли застосування складних моделей машинного навчання є недоцільним [19].

Серед найбільш відомих – авторегресійні моделі (AR), моделі ковзного середнього (MA), поєднані моделі ARIMA, а також їхні сезонні модифікації (SARIMA). Ці підходи добре працюють у випадках, коли врожайність значною мірою залежить від попередніх значень і має виражену сезонність [20].

Авторегресійні моделі передбачають, що поточне значення врожайності залежить від попередніх значень серії. Модель AR(p) описується рівнянням 2.1 [21].

$$Y_t = \phi_0 + \sum_{i=1}^p \phi_i Y_{t-i} + \varepsilon_t, \quad (2.1)$$

де Y_t – врожайність у час t ;

p – порядок авторегресії;

ϕ_i – параметри моделі;

ε_t – випадкова похибка.

Моделі МА(q) використовують лінійну комбінацію попередніх випадкових похибок для прогнозування поточного значення, описується рівнянням 2.2 [22].

$$Y_t = \mu + \varepsilon_t + \sum_{i=1}^q \theta_i - \varepsilon_{t-i}, \quad (2.2)$$

де μ – середнє значення ряду;

θ_i – коефіцієнти впливу похибок;

q – порядок моделі.

Ці підходи добре працюють у випадках, коли врожайність значною мірою залежить від попередніх значень і має виражену сезонність. Наприклад, моделі ARIMA успішно застосовувалися для прогнозування врожайності пшениці та рису в умовах стабільних кліматичних змінних [23]. Недоліком статистичних методів є їхня слабка здатність відображати складні нелінійні залежності між багатьма факторами.

2.1.2 Методи машинного навчання

Методи машинного навчання (ML) стали одним із найважливіших напрямів у прогнозуванні врожайності завдяки здатності працювати з великими обсягами даних та враховувати складні нелінійні залежності між змінними. На відміну від статистичних моделей, вони не вимагають жорстких припущень про стаціонарність часових рядів чи лінійність зв'язків, що робить їх більш універсальними для аграрних задач. Основними підходами у цій групі є дерева рішень, ансамблеві методи (Random Forest, Gradient Boosting), метод опорних векторів (SVM) та штучні нейронні мережі (NN).

Дерева рішень є базовим алгоритмом, який виконує рекурсивне розбиття даних на підмножини на основі критеріїв поділу (наприклад, мінімізації ентропії чи індексу Джині), описується рівнянням 2.3.

$$f(X) = \sum_{m=1}^M c_m \cdot I(X \in R_m), \quad 2.3$$

де R_m – регіон простору ознак;

c_m – прогнозоване значення у цьому регіоні;

I – індикаторна функція.

Дерева рішень легко інтерпретувати, однак вони часто схильні до перенавчання [24].

Щоб подолати обмеження дерев рішень, використовуються ансамблеві методи. Random Forest (RF) створює велику кількість дерев рішень, кожне з яких будується на випадковій підмножині ознак і даних. Остаточний прогноз отримується шляхом усереднення результатів окремих дерев, описується рівнянням 2.4.

$$\hat{y} = \frac{1}{B} \sum_{b=1}^B f_b(X), \quad 2.4$$

де B – кількість дерев;

$f_b(X)$ – прогноз b -го дерева.

Ще одним потужним методом є Gradient Boosting Machines, що поєднує дерева рішень у послідовний ансамбль. На кожному кроці нове дерево будується для апроксимації похибки попередньої моделі, завдяки чому досягається висока точність навіть при невеликих наборах даних [25].

Метод опорних векторів особливо ефективний для задач регресії в умовах багатовимірних ознак. Його суть полягає у знаходженні оптимальної гіперплощини, що мінімізує відхилення між прогнозованими та фактичними значеннями [26].

Таким чином, методи машинного навчання демонструють високу точність і гнучкість у прогнозуванні врожайності культур. Проте їхнє використання потребує значних обчислювальних ресурсів, ретельної підготовки даних та налаштування гіперпараметрів, що обмежує їх у деяких практичних випадках.

2.1.3 Методи глибинного навчання

Методи глибинного навчання (Deep Learning, DL) за останні роки стали одним із ключових напрямів у прогнозуванні врожайності сільськогосподарських культур. Їхня особливість полягає у здатності автоматично виявляти приховані залежності у великих і різномірних масивах даних, зокрема часових рядах кліматичних показників, характеристиках ґрунтів та супутникових знімках. На відміну від класичних статистичних методів, які потребують попереднього визначення ознак, DL-моделі самостійно формують представлення даних під час навчання, що дозволяє досягати високої точності прогнозів [27].

Рекурентні нейронні мережі створені для роботи з послідовними даними, зокрема часовими рядами, які є типовими для аграрних задач (опади, температура, динаміка врожайності). Завдяки використанню зворотних зв'язків вони здатні зберігати інформацію про попередні стани та враховувати залежності у часі. Проте класичні RNN мають суттєве обмеження – труднощі із засвоєнням довгострокових залежностей через проблему «зникання градієнтів» [28].

Щоб вирішити зазначені проблеми, було розроблено архітектуру LSTM (довга короткострокова пам'ять). Вона включає спеціальні комірки пам'яті, що дозволяють зберігати важливу інформацію протягом тривалих періодів та відкидати незначні дані. Це робить LSTM надзвичайно ефективними у прогнозуванні врожайності культур, які залежать від багаторічних кліматичних закономірностей [29].

Згорткові нейронні мережі традиційно застосовуються для обробки зображень, проте вони знайшли широке застосування у сільському господарстві завдяки можливості аналізу супутникових і аерофотознімків полів. CNN дозволяють виявляти ознаки, пов'язані зі станом рослин, рівнем хлорофілу, індексом вегетації (NDVI), а також прогнозувати майбутню врожайність на основі візуальних характеристик рослинного покриву [30].

Таким чином, методи глибинного навчання відкривають широкі перспективи у сфері прогнозування врожайності сільськогосподарських культур. Вони

дозволяють моделювати складні залежності, враховувати динаміку кліматичних факторів і використовувати дані дистанційного зондування Землі. Однак їх впровадження потребує значних обчислювальних ресурсів і великих якісних наборів даних, що залишається одним із головних викликів.

2.1.4 Гібридні підходи

Гібридні методи прогнозування врожайності поєднують різні підходи – статистичні, машинного навчання та глибинного навчання – з метою використання їхніх переваг та компенсації обмежень. Такий підхід дозволяє враховувати як лінійні залежності та сезонність (характерні для статистичних моделей), так і складні нелінійні зв'язки між великою кількістю змінних (що краще моделюються алгоритмами ML та DL). Використання гібридних моделей особливо актуальне у сільському господарстві, де врожайність визначається комбінацією багатьох факторів: кліматичних умов, характеристик ґрунту, агротехнічних заходів і стану рослинного покриву.

Одним із найбільш поширених напрямів є інтеграція моделей ARIMA з методами машинного навчання. Наприклад, ARIMA ефективно прогнозує лінійні тренди та сезонні коливання, тоді як Random Forest або SVM уточнюють прогноз, моделюючи нелінійні залежності [31].

Ще одним напрямом є поєднання ARIMA чи SARIMA з LSTM. Такий підхід дозволяє спершу врахувати сезонність та тренди часових рядів (завдяки ARIMA), а потім відобразити нелінійні залежності та довготривалі взаємозв'язки за допомогою LSTM. Результати досліджень показують, що ARIMA-LSTM моделі знижують похибку прогнозування у порівнянні з окремим використанням кожного з методів, особливо у випадках, коли врожайність залежить від багаторічних кліматичних факторів [31].

На практиці значну ефективність показали гібридні архітектури, що комбінують згорткові та рекурентні нейронні мережі. У такій схемі CNN відповідає за аналіз просторових даних (наприклад, супутникових знімків полів), тоді як LSTM або GRU обробляють часові ряди кліматичних показників (температура, 2025 р.

опадів, вологість ґрунту). Це дозволяє враховувати одночасно і просторові, і часові характеристики процесів, що визначають врожайність.

Таким чином, гібридні підходи є перспективним напрямом у прогнозуванні врожайності, оскільки поєднують у собі сильні сторони різних методів. Вони дозволяють досягати високої точності за рахунок урахування багатофакторності аграрних процесів. Водночас їх основними обмеженнями залишаються висока обчислювальна складність, потреба у великих обсягах якісних даних та складність у налаштуванні комбінованих моделей.

2.2 Загальна концепція системи

Розробка інтелектуальної системи прогнозування врожайності культур базується на необхідності забезпечення точних, своєчасних та адаптивних прогнозів для користувачів з різними рівнями доступу до аграрних даних. У контексті малих господарств України така система повинна поєднувати простоту використання, мінімальні вимоги до обчислювальних ресурсів та можливість роботи з обмеженими історичними даними. Основна мета системи полягає у наданні користувачеві інформації про очікувану врожайність на основі комбінації історичних даних, актуальних кліматичних показників та характеристик ґрунтів.

Система включає кілька ключових функціональних модулів. Перший модуль відповідає за збір і попередню обробку даних, до яких належать історичні показники врожайності, погодні дані, характеристики ґрунтів та просторові дані (наприклад, супутникові знімки або дані дистанційного зондування). Обробка даних включає очищення від пропусків, нормалізацію, трансформації та видалення аномалій, що забезпечує якість і коректність вхідної інформації для наступних модулів системи.

Другий модуль реалізує процеси прогнозування врожайності. Для забезпечення точності прогнозів та адаптивності до аномальних погодних умов застосовується комбінований підхід, який поєднує статистичні моделі (ARIMA, SARIMA) для визначення сезонних трендів та методи машинного навчання (Random Forest, Gradient Boosting) для моделювання нелінійних взаємозв'язків між

2025 р.

кліматичними, ґрунтовими та агротехнічними факторами. Такий підхід дозволяє системі ефективно працювати з обмеженими наборами даних, характерними для малих господарств, і адаптувати прогнози під конкретні локальні умови.

Третій модуль відповідає за візуалізацію результатів та взаємодію з користувачем через вебінтерфейс. Інтерфейс забезпечує зручний доступ до прогнозів, дозволяє вводити локальні дані про господарство та оперативно отримувати рекомендації щодо оптимізації посівів, поливу та внесення добрив. Важливим аспектом є забезпечення інтуїтивної зрозумілості та доступності системи навіть для користувачів без спеціальної підготовки в галузі інформаційних технологій або агрономії.

Загальна концепція системи полягає у створенні інтегрованого рішення, яке поєднує збір, обробку та аналіз даних із прогнозуванням та візуалізацією результатів. Це дозволяє підвищити ефективність управління малими господарствами, мінімізувати ризики, пов'язані з аномальними погодними умовами, та забезпечити науково обґрунтовану підтримку прийняття рішень.

2.3 Функціональна модель системи

Функціональна модель є ключовим елементом проєктування інтелектуальної системи прогнозування врожайності, оскільки дозволяє формально описати всі основні функції, які виконує система, та їх взаємозв'язки. На першому рівні моделі система розглядається як цілісний блок (рис. 2.1), що реалізує загальну функцію «Прогнозування врожайності поля». Вхідними потоками є дані про погодні умови, історичні дані врожайності, а також додаткові агротехнічні параметри. Керуючими впливами виступають налаштування користувача, обмеження моделі та нормативні довідники, які визначають правила інтерпретації даних. Механізмами виступає програмне забезпечення та апаратні ресурси. Вихідним потоком інформації є прогнозні дані щодо врожайності у вигляді числових значень і візуалізацій та рекомендації щодо агротехнічних дій.

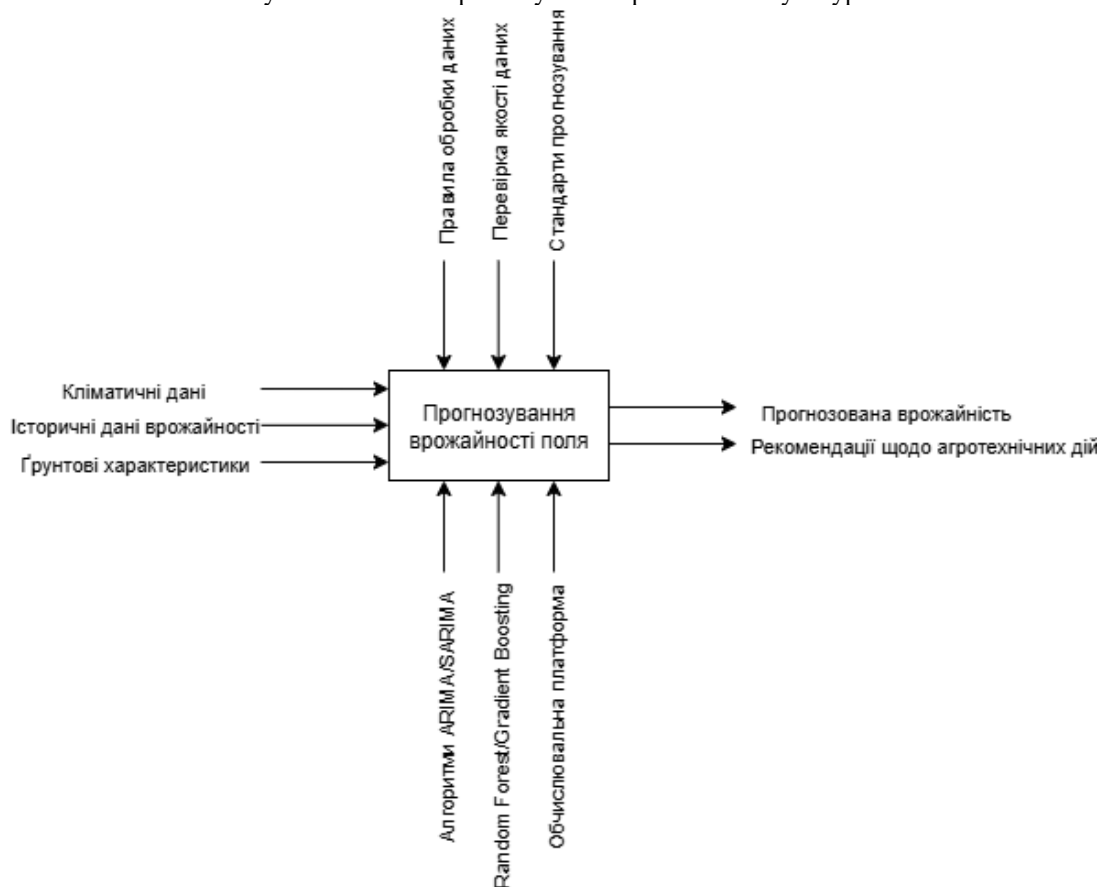


Рисунок 2.1 – Діаграма IDEF0

Центральною функцією системи є прогнозування врожайності. Для її реалізації система виконує кілька взаємопов'язаних підфункцій: збір та обробка даних, аналіз та прогнозування, та візуалізація прогнозу (рис. 2.2). На вході система отримує різноманітні дані: погодні показники, агрономічні характеристики, а також історичні відомості про фактичні врожаї. Виходом першої функції є нормалізовані дані.

Другим блоком виступає аналіз і прогнозування, у рамках якого реалізуються математичні та інтелектуальні методи роботи з часовими рядами. На цьому етапі система застосовує обрані алгоритми прогнозування для обчислення прогнозних значень врожайності з урахуванням як історичних даних, так і поточних погодних умов.

Завершальним блоком є візуалізація результатів, яка забезпечує зручне подання прогнозної інформації користувачеві. У вебінтерфейсі реалізуються інтерактивні графіки динаміки врожайності та аналітичні таблиці, що дозволяють

фермеру швидко оцінити потенційні ризики і прийняти обґрунтовані управлінські рішення.

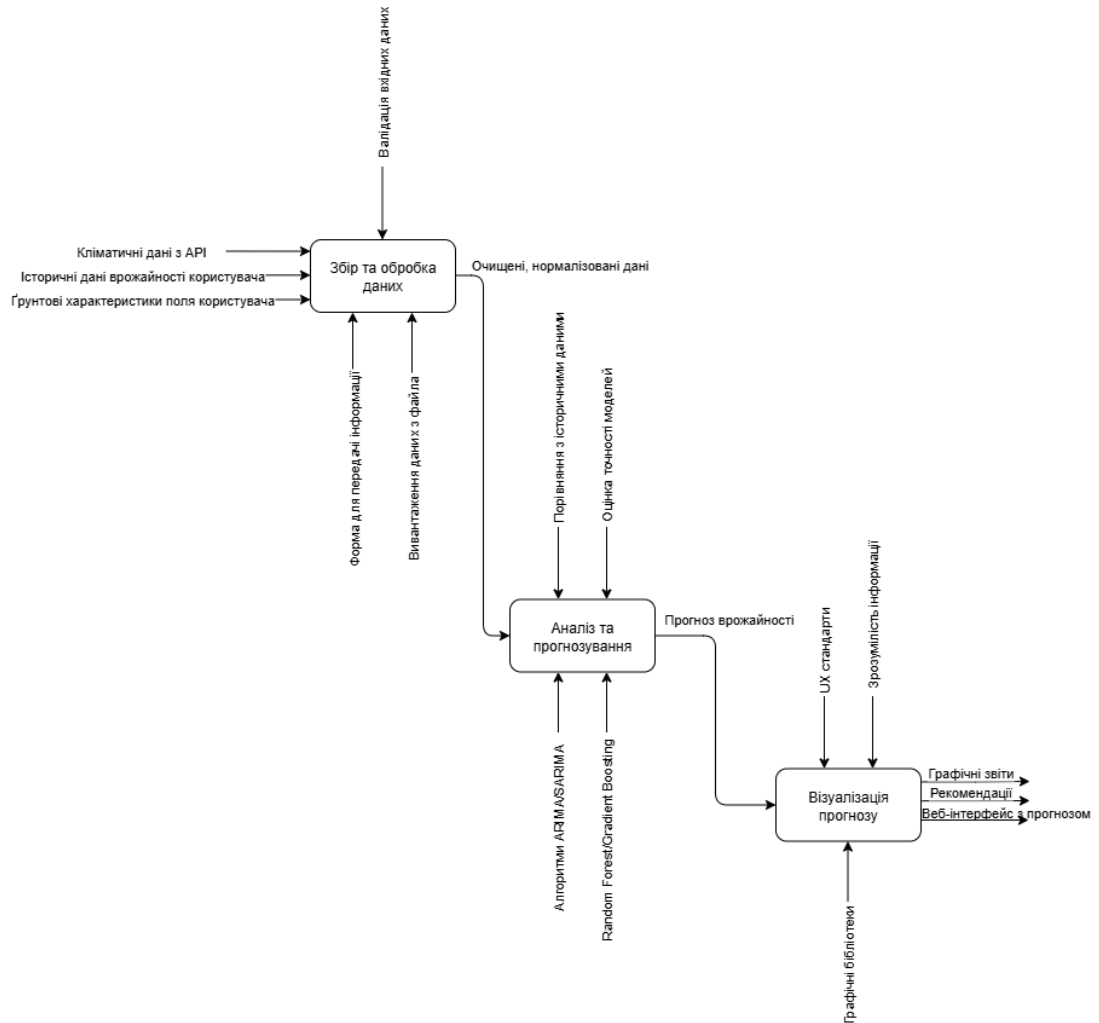


Рисунок 2.2 – Діаграма IDEF0 першого рівня

Таким чином, функціональна модель системи включає три взаємопов’язані підсистеми – збір та обробку даних, аналіз і прогнозування, а також візуалізацію результатів, які спільно забезпечують досягнення головної мети – формування достовірних прогнозів врожайності для малих аграрних господарств України.

2.4 Алгоритм прогнозування врожайності

Алгоритм прогнозування врожайності в розроблюваній системі побудований на поєднанні статистичної моделі SARIMA та машинного навчання з використанням Gradient Boosting. Основна ідея полягає у використанні історичних даних про врожайність, характеристик ґрунту та погодних показників для

формування максимально точного прогнозу. Процес роботи алгоритму можна поділити на кілька етапів, які забезпечують систематичну обробку даних та отримання результатів, придатних для практичного використання фермерами.

На першому етапі здійснюється збір даних із бази HistoricalYield та погодних API. Дані включають історичні врожаї, хімічний склад ґрунту (азот, фосфор, калій, органічну речовину), а також погодні параметри (температура, вологість, опади) за критичні місяці для відповідних культур. Використання критичних місяців дозволяє моделі концентруватися на тих періодах, які найбільше впливають на врожайність.

Другий етап полягає у попередній обробці та очищенні даних. Пропущені значення погодних показників заповнюються за допомогою SARIMA-моделі, що дозволяє прогнозувати відсутні дані на основі наявних часових рядів. Такий підхід гарантує цілісність даних та підвищує точність подальшого прогнозування врожайності.

На третьому етапі формуються навчальні та тестові вибірки для Gradient Boosting. Модель навчається на відомих значеннях врожайності та відповідних ознаках, а потім робить прогноз для відсутніх значень `yield_amount`. Одночасно здійснюється обчислення метрик якості моделі (RMSE, MAE, R^2), що дозволяє оцінити точність передбачення та визначити, наскільки добре модель відтворює фактичні дані.

Паралельно реалізується прогнозування за допомогою SARIMA для часових рядів врожайності, що дозволяє отримати трирічний прогноз на основі сезонних та трендових компонентів. Поєднання двох підходів забезпечує гнучкість: SARIMA добре відтворює сезонні закономірності, тоді як Gradient Boosting дозволяє враховувати вплив додаткових факторів, таких як склад ґрунту та погодні аномалії.

Результати прогнозування відображаються у вигляді графіків та таблиць, а також супроводжуються аналітичними рекомендаціями для фермера, наприклад, оптимальний рівень внесення азоту або ймовірність посухи у критичні місяці. Порівняльний аналіз показує, що Gradient Boosting зазвичай демонструє меншу середню абсолютну похибку (MAE) на історичних даних, тоді як SARIMA

2025 р.

забезпечує стабільний сезонний прогноз на кілька років вперед. Такий підхід дозволяє комбінувати сильні сторони обох методів та забезпечувати більш точне і практично корисне прогнозування врожайності. На рисунку 2.3 представлено блок-схему алгоритму, яка відображає послідовність дій від завантаження даних до виведення результатів.

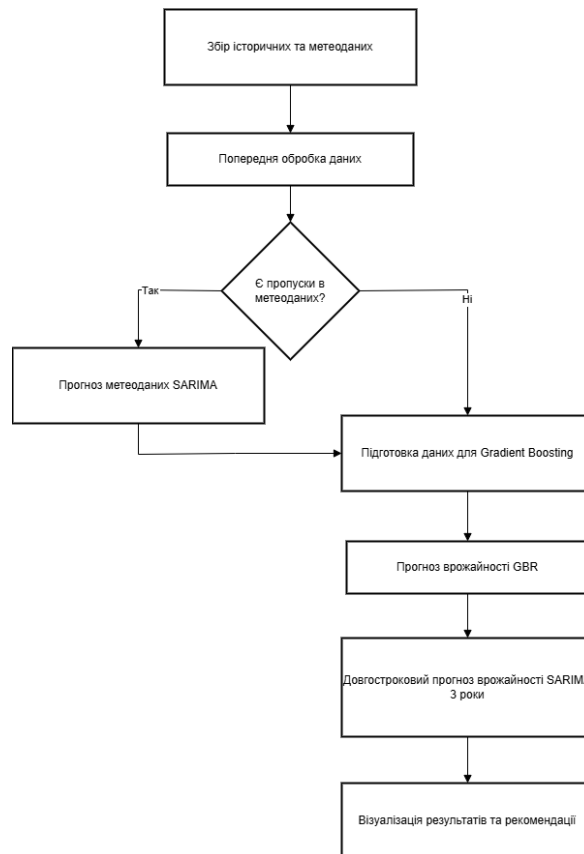


Рисунок 2.3 – Блок схема алгоритму побудови прогнозу врожайності

Погодні дані містять пропуски, які необхідно відновити перед навчанням моделей. Оскільки погодні параметри мають чітко виражену сезонність, для їх відновлення використовується модель, описана рівнянням 2.5.

$$X_t = \phi_1 X_{t-1} + \theta_1 \varepsilon_{t-1} + \dots + \Phi_1 X_{t-s} + \Theta_1 \varepsilon_{t-s} + \varepsilon_t, \quad (2.5)$$

де X_t – значення погодного параметра в момент часу t ;

s – сезонний період (довжина критичних для культури місяців);

ϕ_1, θ_1 – коефіцієнти AR та MA;

Φ_1, Θ_1 – сезонні коефіцієнти AR та MA;

ε_t – випадкова похибка.

Ця модель адаптована у коді системи через модуль `statsmodels` і використовується окремо до кожного стовпця погоди, що містить пропущені значення. Завдяки SARIMA відновлені дані зберігають часову структуру та не порушують послідовності залежностей, характерних для метеорологічних рядів.

Після відновлення метеорологічних характеристик здійснюється прогнозування пропущених значень врожайності. Для цього використовується модель Gradient Boosting – ансамблевий метод, який будує прогноз як суму послідовних дерев рішень. У загальному вигляді модель описується формулою 2.6.

$$\hat{y}(x) = \sum_{m=1}^M \gamma_m h_m(x) \quad (2.6)$$

де $h_m(x)$ – дерево рішень, яке апроксимує залишкову похибку на ітерації m ;

γ_m – ваговий коефіцієнт.

Такий підхід дозволяє моделі враховувати складні взаємодії між ґрунтовими характеристиками, метеорологічними змінними та агрономічними показниками. Gradient Boosting у системі коректно працює на наборах даних, у яких показники врожайності частково відсутні, що робить його ефективним інструментом для поповнення історичних рядів перед довгостроковим прогнозуванням.

Завершальним етапом є прогнозування врожайності на три роки вперед на основі сезонної моделі SARIMA, що застосовується до очищеної та доповненої вибірки фактичних значень врожайності. На відміну від попереднього використання SARIMA для погодних параметрів, тут модель будується на часовому ряді врожайності поля, що дозволяє врахувати сезонність, тренд і внутрішню інерцію агровиробничих циклів.

Поєднання SARIMA та Gradient Boosting забезпечує комплексне моделювання різних аспектів даних: статистична модель дозволяє враховувати сезонність та роботу з часовими рядами, тоді як ансамблевий метод ефективно обробляє багатовимірні набори ознак і виявляє складні нелінійні залежності. Такий підхід робить прогнозування більш надійним у випадку обмежених або частково

2025 р.

відсутніх даних. Відповідно, результати, отримані обома моделями, дозволяють сформулювати повноцінний прогноз, а також виконати порівняльний аналіз точності за допомогою RMSE, MAE та коефіцієнта детермінації R^2 , що надає змогу оцінити якість отриманих моделей та їх придатність до практичного використання у агрономічній діяльності.

Таким чином, алгоритм забезпечує комплексне, багатфакторне прогнозування врожайності, що робить його ефективним інструментом підтримки рішень для малих фермерських господарств.

2.5 Специфікація вимог до інформаційної системи прогнозування врожайності

2.5.1 Призначення та межі проєкту

Призначення системи (застосунку), для якої розробляється програмне забезпечення

Система призначена для прогнозування врожайності сільськогосподарських культур у малих господарствах України з урахуванням аномальних погодних умов. Вона надає можливість фермерам отримувати точкові прогнози врожайності на основі історичних даних та актуальної кліматичної інформації. Система допомагає оптимізувати агротехнічні рішення та планування посівних площ, зменшуючи ризики втрат через несприятливі погодні умови.

Погодження, що ухвалені в програмній документації

Система погоджена з вимогами до малого бізнесу у сільському господарстві, включає легкий інтерфейс користувача та мінімальні вимоги до обладнання.

Межі проєкту ПЗ

Система обмежена прогнозуванням врожайності на основі доступних даних (кліматичних показників, історії врожаю та характеристик ґрунтів). Вона не включає автоматичне управління поливом чи агротехнічними пристроями.

Прогнози носять рекомендаційний характер і залежать від точності введених даних.

2.5.2 Загальний опис

Сфера застосування

Система спеціально орієнтована на малі та середні господарства України, які часто стикаються з нестабільними та непередбачуваними погодними умовами, а також з обмеженими ресурсами для ведення виробництва. Її метою є значне підвищення точності планування врожаю шляхом надання достовірних прогнозів врожайності, що дозволяє фермерам своєчасно приймати управлінські рішення. Водночас система спрямована на зменшення фінансових та матеріальних втрат, що виникають унаслідок несприятливих погодних коливань, а також на оптимізацію використання наявних ресурсів, підвищення ефективності господарської діяльності та підтримку сталого розвитку малих агровиробників у регіонах з мінливим кліматом.

Характеристики користувачів

- фермери з обмеженим досвідом роботи з комп'ютерами;
- володіння базовими навичками введення даних через вебінтерфейс;
- не потребується знання програмування чи складних статистичних моделей.

Загальна структура і склад системи

Система складається з трьох основних модулів:

1. Модуль введення даних – внесення кліматичних даних, історії врожаю та характеристик ґрунту.
2. Модуль прогнозування – обробка даних моделями ARIMA та Random Forest/Gradient Boosting для отримання прогнозу врожайності.
3. Модуль відображення результатів – вебінтерфейс для перегляду прогнозу, графіків та рекомендацій.

Загальні обмеження

- система потребує інтернет-з'єднання для оновлення кліматичних даних;
- обмежена точність прогнозів при дуже малих обсягах історичних даних;
- не підтримує інтеграцію з зовнішніми агротехнічними пристроями.

2.5.3 Функції системи

Функція введення даних

Користувач має можливість вносити дані про врожайність за попередні роки, кліматичні показники та характеристики ґрунтів.

Вхідна інформація: історичні дані врожайності, кліматичні параметри, дані про ґрунт.

Вихідна інформація: структуровані дані, що зберігаються у базі даних для подальшого аналізу.

Функціональні вимоги:

- система повинна перевіряти коректність введених значень;
- підтримка завантаження даних у вигляді таблиць (CSV, Excel);
- автоматичне збереження даних у базі.

Функція прогнозування врожайності

Система аналізує введені дані та будує прогноз врожайності на основі моделей ARIMA та Gradient Boosting.

Вхідна інформація: дані з бази (історичні врожаї, клімат, ґрунт), актуальні кліматичні дані.

Вихідна інформація: прогнозовані показники врожайності для вибраної культури.

Функціональні вимоги:

- використання статистичних і машинних моделей для обробки даних;
- можливість налаштовувати параметри моделювання;
- забезпечення оновлення прогнозу при надходженні нових даних.

Функція відображення результатів

Користувач може переглядати прогноз врожайності у вигляді числових значень, графіків та таблиць, а також отримувати рекомендації для оптимізації агротехнічних рішень.

Вхідна інформація: результати прогнозування, дані з бази.

Вихідна інформація: відображення прогнозу у веб-інтерфейсі (графіки, таблиці, рекомендації).

Функціональні вимоги:

- відображення даних у зручному для користувача вигляді;
- підтримка інтерактивних графіків;
- формування текстових рекомендацій.

2.5.4 Вимоги до інформаційного забезпечення

Джерела і зміст вхідних даних

Основними джерелами вхідної інформації для системи є історичні дані врожайності сільськогосподарських культур, кліматичні показники та характеристики ґрунтів. Історичні дані включають відомості про обсяги врожаю за попередні роки, що дозволяють формувати часові ряди для моделювання. Кліматичні показники охоплюють температуру повітря, кількість опадів, вологість та інші метеорологічні параметри, які можуть бути отримані з відкритих джерел, зокрема з даних метеостанцій та супутникового моніторингу. Дані про характеристики ґрунтів (тип ґрунту, кислотність, родючість) вводяться користувачем безпосередньо або імпортуються з локальних аграрних довідників. Сукупність цих даних є необхідною для забезпечення точності та надійності прогнозів врожайності.

Нормативно-довідкова інформація

Для функціонування системи передбачається використання нормативно-довідкової інформації, що регламентує структуру та формати даних. До таких

2025 р.

джерел належать класифікатори сільськогосподарських культур, які забезпечують єдині правила кодування та уніфікації назв, а також довідники агрокліматичних зон України. Використання довідкових матеріалів уніфікує процес введення даних, підвищує їхню достовірність та забезпечує можливість інтеграції інформації з іншими джерелами.

Вимоги до способів організації, збереження та ведення інформації

Організація інформаційного забезпечення повинна ґрунтуватися на принципах структурованості, цілісності та доступності даних. Вся інформація зберігається у реляційній базі даних, що забезпечує ефективне виконання запитів та обробку великих масивів. Для підвищення надійності передбачається резервне копіювання та можливість відновлення даних у разі збою. Ведення інформації здійснюється шляхом контролю актуальності та періодичного оновлення кліматичних показників, а також перевірки коректності введених користувачем даних. Доступ до інформаційної бази регламентується правами користувачів, що дозволяє забезпечити захист від несанкціонованого втручання.

2.5.5 Вимоги до технічного забезпечення

Технічне забезпечення системи прогнозування врожайності повинно забезпечувати стабільне функціонування програмних компонентів та можливість обробки великих обсягів інформації у прийнятні терміни. Оскільки система орієнтована на малі фермерські господарства України, вимоги до апаратних ресурсів залишаються помірними, що робить її доступною для широкого кола користувачів.

З боку серверного середовища система потребує використання обчислювальних потужностей, здатних виконувати ресурсоємні операції моделювання. Мінімальні характеристики сервера включають багатоядерний процесор (від 4 ядер), обсяг оперативної пам'яті від 8 ГБ та дисковий простір не менше 100 ГБ. Для забезпечення швидкої обробки та зниження затримок рекомендується використання SSD-накопичувачів. Наявність доступу до GPU є

2025 р.

бажаною, але не обов'язковою умовою. Сервер повинен підтримувати сучасні операційні системи та мати можливість масштабування ресурсів у разі зростання обсягів даних.

Клієнтська частина вебзастосунку не потребує високопродуктивного обладнання, оскільки більшість обчислень виконується на сервері. Для роботи користувачу достатньо персонального комп'ютера або ноутбука з процесором рівня Intel Core i3 (або аналогом), оперативною пам'яттю від 4 ГБ та стабільним доступом до мережі Інтернет зі швидкістю не менше 5 Мб/с. Взаємодія із системою здійснюється через сучасні браузері (Google Chrome, Opera, Mozilla Firefox, Microsoft Edge), які підтримують роботу з інтерактивними вебінтерфейсами.

2.5.6 Вимоги до програмного забезпечення

Архітектура програмної системи

Архітектура системи побудована за принципами клієнт–серверної моделі, що забезпечує розподіл функцій між серверною частиною, яка виконує основні обчислення та обробку даних, та клієнтською частиною, яка відповідає за взаємодію з користувачем через вебінтерфейс. Такий підхід підвищує масштабованість системи, спрощує її підтримку та забезпечує можливість подальшого розвитку функціоналу. Вебзастосунок реалізується на основі багаторівневої архітектури: рівень введення даних, рівень логіки (прогнозування врожайності), рівень доступу до бази даних та рівень відображення результатів.

Системне програмне забезпечення

Функціонування системи потребує використання сучасної операційної системи, здатної забезпечити стабільну роботу серверного середовища. Рекомендованим є використання Linux-дистрибутивів, які мають високу надійність, гнучкість налаштування та підтримують розгортання вебсерверів. Для клієнтської частини вимоги мінімальні та обмежуються наявністю будь-якої

сучасної операційної системи (Windows, Linux, macOS), яка підтримує роботу сучасних веббраузерів.

Мережне програмне забезпечення

Для забезпечення коректної роботи вебзастосунку необхідне використання вебсервера, що підтримує обробку HTTP(S)-запитів. Крім того, система має бути здатна інтегруватися з протоколами безпечної передачі даних, що гарантує захист інформації від несанкціонованого доступу. Мережне програмне забезпечення повинно забезпечувати безперервний доступ до сервера та можливість масштабування при збільшенні кількості користувачів.

Програмне забезпечення ведення інформаційної бази

Інформаційна база реалізується у вигляді реляційної бази даних. Для збереження історичних даних, кліматичних показників та результатів прогнозування передбачається використання системи управління базами даних MySQL або її аналогів (PostgreSQL). Обрана СУБД повинна підтримувати механізми транзакцій, забезпечувати цілісність даних, резервне копіювання та відновлення у разі збоїв. Оптимізація запитів до бази даних є необхідною для забезпечення швидкодії вебзастосунку.

Мова і технологія розробки ПЗ

Програмна реалізація системи здійснюється з використанням мови програмування Python, яка забезпечує ефективну інтеграцію алгоритмів машинного навчання та обробку часових рядів. Серверна частина вебзастосунку реалізується за допомогою фреймворку Django, що підтримує модульну структуру, засоби роботи з базою даних та забезпечує гнучкість розширення функціоналу. Для реалізації моделей прогнозування застосовуються бібліотеки машинного навчання (scikit-learn, statsmodels, XGBoost). Клієнтська частина базується на технологіях HTML5, CSS3 та JavaScript з можливим використанням бібліотек для побудови інтерактивних графіків (Chart.js, D3.js).

2.5.7 Вимоги до зовнішніх інтерфейсів

Інтерфейс користувача

Інтерфейс користувача системи має бути веборієнтованим та реалізованим у вигляді зручного і зрозумілого графічного середовища. Основними вимогами є інтуїтивна навігація, простота введення та редагування даних, а також наочне відображення результатів прогнозування у вигляді таблиць і графіків. Важливою характеристикою є адаптивність інтерфейсу, що забезпечує коректне відображення на різних пристроях (настільних комп'ютерах, ноутбуках, планшетах, смартфонах). Дизайн інтерфейсу повинен бути мінімалістичним, без перевантаження зайвими елементами, але водночас інформативним і доступним для користувачів із базовими навичками роботи з комп'ютером.

Апаратний інтерфейс

Система не передбачає прямої взаємодії з агротехнічними пристроями чи датчиками, тому апаратний інтерфейс обмежується стандартними засобами взаємодії користувача з персональним комп'ютером. Використовуються звичайні пристрої введення та виведення інформації – клавіатура, миша, монітор. Також інтеграція з мобільними пристроями для перегляду результатів прогнозування, однак спеціалізоване апаратне обладнання не є обов'язковим.

Програмний інтерфейс

Програмний інтерфейс системи забезпечує взаємодію між вебзастосунком та зовнішніми джерелами даних. Передбачається використання API для отримання метеорологічних показників із відкритих. Програмний інтерфейс також включає механізми взаємодії серверної частини з базою даних, що спрощує роботу з інформацією та підвищує узгодженість даних. Важливою вимогою є забезпечення стандартизованих форматів обміну інформацією (JSON, CSV), що підвищує сумісність із зовнішніми системами.

Комунікаційний протокол

Для забезпечення коректної та безпечної передачі даних система повинна використовувати сучасні мережні протоколи. Основним протоколом є HTTPS, який забезпечує шифрування переданої інформації та захист від несанкціонованого доступу. Обмін даними між клієнтською та серверною частинами здійснюється за допомогою стандартних вебтехнологій REST API, що гарантує сумісність і масштабованість системи. Для інтеграції з зовнішніми джерелами кліматичної інформації застосовуються стандартні протоколи HTTP(S) із використанням ключів доступу до API.

2.5.8 Властивості програмного забезпечення

Доступність

Програмне забезпечення повинно бути доступним для користувачів у будь-який час через веббраузер за умови наявності стабільного інтернет-з'єднання. Система орієнтована на роботу у середовищі з обмеженими ресурсами, тому її використання не потребує спеціалізованого обладнання. Доступність також передбачає можливість одночасної роботи кількох користувачів без суттєвого зниження продуктивності.

Супроводжуваність

Система має бути розроблена з урахуванням принципів модульності та документованого коду, що полегшує внесення змін та вдосконалення функціоналу. Супроводжуваність забезпечується використанням фреймворку Django, який підтримує структурування компонентів і спрощує процес оновлення. Наявність технічної документації та стандартизованих інтерфейсів сприяє зручному обслуговуванню та розширенню системи.

Переносимість

Програмне забезпечення повинно бути переносимим між різними середовищами експлуатації. Використання Python та вебтехнологій забезпечує можливість розгортання системи як на локальних серверах, так і у хмарних інфраструктурах. Переносимість також передбачає незалежність клієнтської частини від операційної системи, оскільки доступ здійснюється через браузер, що робить систему універсальною для Windows, Linux та macOS.

Продуктивність

Система має забезпечувати обробку великих масивів даних у прийнятні терміни. Алгоритми машинного навчання повинні бути оптимізовані для швидкого формування прогнозів. Продуктивність вебзастосунку підтримується завдяки використанню реляційної бази даних та ефективної архітектури клієнт–сервер. У разі збільшення навантаження можливе масштабування серверних ресурсів, що дозволяє підтримувати стабільний рівень швидкодії.

Надійність

Надійність системи визначається її здатністю безперервно функціонувати протягом тривалого часу без збоїв. Для цього передбачено механізми резервного копіювання даних, а також вбудовані засоби відновлення працездатності після можливих помилок. Надійність також досягається за рахунок обмеження доступу користувачів лише до дозволених функцій та забезпечення контролю введених даних.

Безпека

Безпека програмного забезпечення передбачає захист даних від несанкціонованого доступу, викривлення або втрати. Для цього використовується протокол HTTPS, що забезпечує шифрування інформаційних потоків. Система повинна підтримувати автентифікацію користувачів та контроль доступу

відповідно до їхніх прав. Регулярне оновлення програмних компонентів та використання захищених бібліотек дозволяє мінімізувати ризики кібератак.

Висновки до розділу 2

У другому розділі було детально розроблено проєктні рішення інтелектуальної системи прогнозування врожайності сільськогосподарських культур, які охоплюють концептуальні, методологічні та функціональні аспекти побудови сучасного програмного забезпечення. У межах цього розділу здійснено системний аналіз вимог до програмного комплексу, визначено ключові компоненти його архітектури та описано логіку їхньої взаємодії. Розглянуто загальну архітектуру системи, що передбачає інтеграцію модулів збору, обробки та аналізу даних, модулів прогнозування врожайності, а також підсистем візуалізації, подання результатів та формування рекомендацій для кінцевого користувача. Така структура забезпечує комплексний підхід до підтримки ухвалення агротехнічних рішень у малих фермерських господарствах та дозволяє створити гнучкий, масштабований і зручний у використанні інструмент для щоденної роботи аграріїв. Особлива увага приділяється узгодженості роботи компонентів системи, що гарантує цілісність і безперервність потоків даних, а також стабільність функціонування навіть у разі суттєвого збільшення обсягу інформації чи підключення нових джерел даних.

Важливе місце у проєктуванні системи займає детальний аналіз методів прогнозування, серед яких розглянуто як традиційні статистичні моделі, так і сучасні алгоритми машинного та глибинного навчання, а також гібридні підходи, здатні поєднувати переваги різних методів. Значну увагу приділено можливості врахування сезонності, довгострокових трендів, циклічності, а також складних нелінійних взаємозв'язків між погодними, ґрунтовими, біологічними та історичними даними, що є критично важливим у разі моделювання врожайності в умовах природної мінливості. Це дозволяє створити гнучку й адаптивну модель прогнозування, що динамічно реагує на зміни довкілля та підвищує точність оцінювання майбутніх врожаїв. Окремо визначено інформаційне забезпечення

2025 р.

системи, яке включає широкий спектр джерел даних: історичні аграрні статистичні показники, метеорологічні спостереження, супутникові знімки, сенсорні дані з полів, а також нормативно-довідкові матеріали, необхідні для коректної обробки та інтерпретації вхідної інформації.

Алгоритм прогнозування врожайності, реалізований у системі, ґрунтується на комбінації методів часових рядів та машинного навчання, що дозволяє враховувати як динаміку зміни врожайності у часі, так і вплив багатofакторних агрометеорологічних характеристик.

Розроблені у другому розділі проєктні рішення формують концептуальну та функціональну основу для створення повноцінної інтелектуальної системи прогнозування врожайності, здатної забезпечити високу точність прогнозів і практичну користь для малих аграрних підприємств України. Запропонована система сприятиме підвищенню ефективності управління ресурсами, оптимізації виробничих процесів, зменшенню невизначеності та мінімізації ризиків, пов'язаних із нестабільними чи аномальними погодними умовами. В умовах сучасних кліматичних викликів таке рішення є надзвичайно актуальним та здатне суттєво підвищити конкурентоспроможність і стійкість аграрного сектору.

3 ПРОЄКТУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ СИСТЕМИ ПРОГНОЗУВАННЯ ВРОЖАЙНОСТІ

3.1 Архітектура програмного забезпечення

Архітектура розроблюваної інтелектуальної системи прогнозування врожайності побудована за клієнт-серверним принципом із чітким розподілом обчислювальних, аналітичних та інтерфейсних функцій між серверною частиною та користувацьким рівнем. Такий підхід забезпечує масштабованість, гнучкість у розгортанні й можливість подальшого розширення функціональності системи без внесення змін у базову логіку. Завдяки модульній структурі забезпечується незалежність компонентів, їх логічна ізоляція та можливість повторного використання окремих модулів у майбутніх версіях системи.

Вибір саме клієнт-серверної архітектури обумовлений потребами малих фермерських господарств, які мають обмежені технічні ресурси та не володіють потужною комп'ютерною інфраструктурою для локальних обчислень. Значна частина фермерів використовує мобільні пристрої або застарілі комп'ютери, тому централізація обчислювальних процесів на сервері є оптимальним рішенням. Обробка та аналіз великих масивів даних – таких як погодні ряди, багаторічні дані врожайності – виконуються сервером, що істотно знижує вимоги до клієнтської сторони. Користувач взаємодіє з системою через вебінтерфейс, який не потребує встановлення додаткового програмного забезпечення й працює у звичайному браузері.

Серверна частина виконує ключові функції інтелектуального прогнозування: збір і попередню обробку даних, нормалізацію та доповнення пропусків, запуск алгоритмів машинного навчання й моделей часових рядів, формування візуалізацій і рекомендацій, а також управління базою даних. Використання REST API як основного методу взаємодії між модулями забезпечує стандартизовану комунікацію і дозволяє легко інтегрувати систему з зовнішніми інформаційними сервісами.

Зберігання даних у системі організовано згідно з принципами модульної архітектури. Кожен компонент функціонує незалежно та має уніфікований інтерфейс взаємодії з іншими частинами системи. Такий підхід підвищує надійність роботи, адже збої одного модуля не впливають на працездатність інших. Крім того, модульність полегшує оновлення та тестування програмного забезпечення, забезпечує можливість швидкого розширення функціоналу, наприклад шляхом додавання нових алгоритмів прогнозування або інтеграції додаткових джерел даних.

Окремим аспектом архітектури є підтримка використання відкритих джерел даних, що є критично важливим для фермерів, де нерідко відсутні локальні метеостанції або дані державних реєстрів. Система здатна автоматично отримувати погодні показники з відкритих API. Такий підхід мінімізує фінансові витрати фермерів і робить інструмент доступним навіть для найменших господарств.

У рамках архітектури передбачено механізм кешування та буферизації даних, що зменшує кількість запитів до зовнішніх API та прискорює роботу системи. Також реалізовано можливість ручного завантаження історичних даних врожайності користувачем, що дозволяє працювати навіть у випадку неповних або фрагментарних даних.

На рисунку 3.1 представлено архітектуру інтелектуальної системи прогнозування врожайності, реалізовану за клієнт-серверним принципом. Користувач взаємодіє з веб-інтерфейсом через браузер, який обмінюється даними із сервером через REST API. На серверній стороні реалізовано модулі обробки даних, прогнозування, а також генерації аналітичних звітів. Всі дані зберігаються у базі даних, що забезпечує структуроване збереження історичних та поточних показників. Система також інтегрована з зовнішніми джерелами даних, такими як погодні API.

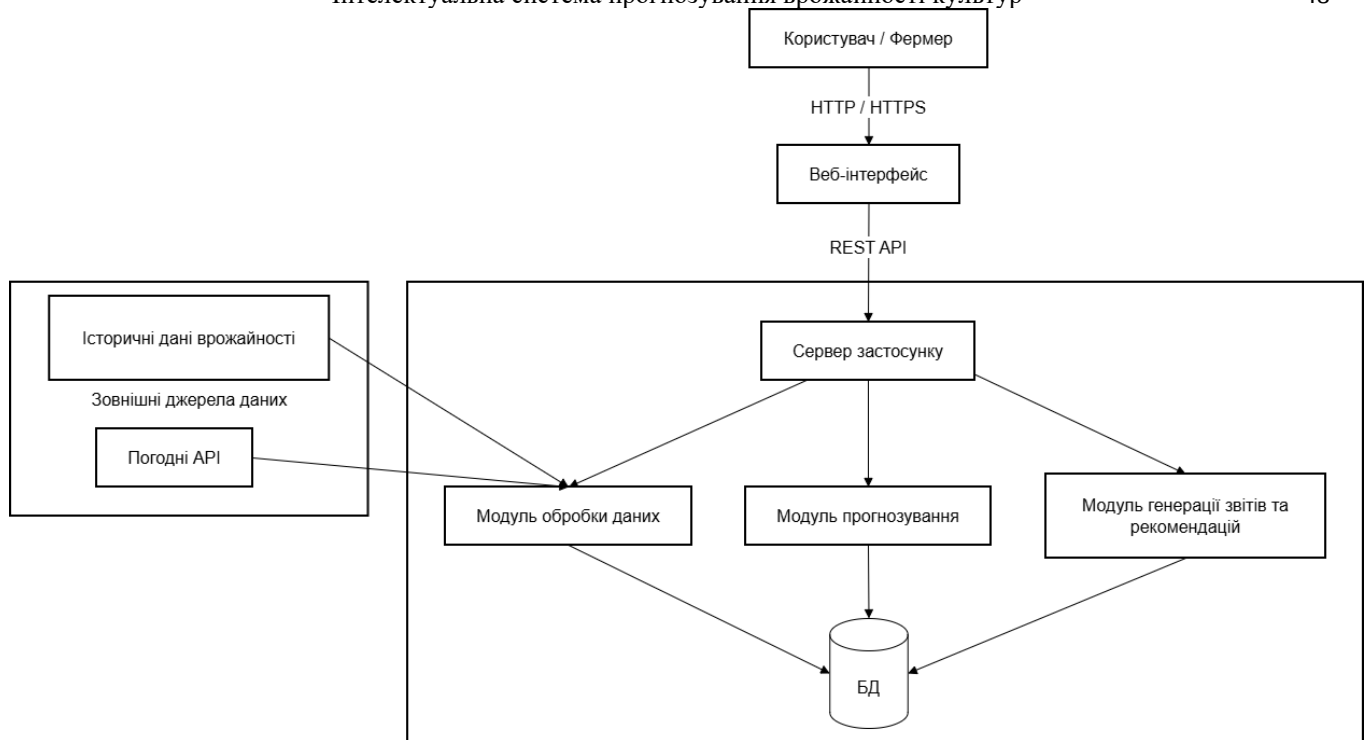


Рисунок 3.1 – Архітектура ІС

Архітектура системи, побудована за клієнт-серверним принципом, забезпечує масштабованість, гнучкість та стабільність роботи, розподіляючи ресурсоємні обчислення на серверну частину і мінімізуючи навантаження на користувача. Модульна структура дозволяє легко оновлювати та розширювати функціональність, інтегрувати нові джерела даних і підтримувати роботу навіть на малопотужних пристроях. Такий підхід робить систему доступною для малих фермерських господарств і гарантує надійну взаємодію між усіма її компонентами.

3.2 Розробка компонентів програмного забезпечення

Розробка інтелектуальної системи прогнозування врожайності передбачає чіткий поділ функціоналу на окремі програмні модулі, кожен з яких відповідає за певний етап обробки інформації або взаємодії з користувачем. Такий підхід дозволяє не лише ізолювати логіку роботи різних компонентів, а й підвищити загальну надійність системи, забезпечуючи стабільність її роботи в умовах зміни обсягів даних чи появи нових вимог. Крім того, модульність значно спрощує процеси модернізації, тестування та масштабування у майбутньому, що є критично важливим для систем, орієнтованих на довготривале використання.

Система складається з кількох ключових підсистем, кожна з яких виконує окремі, але взаємопов'язані завдання: збір і попередню обробку даних, їх аналіз та побудову прогнозів врожайності, а також візуалізацію результатів та формування аналітичних рекомендацій для користувача. Завдяки цьому система залишається адаптивною, розширюваною та здатною ефективно реагувати на зміни у середовищі чи вимогах.

3.2.1 Модуль збору та обробки даних

Модуль збору та обробки даних відповідає за інтеграцію з зовнішніми джерелами інформації, такими як Open-Meteo, а також за прийом даних, введених користувачем. Окрім базових метеорологічних параметрів, модуль опрацьовує й водні показники – рівень вологості ґрунту, обсяг опадів, показники водного балансу та інші характеристики, що впливають на стан рослин. Після отримання даних здійснюється попередня обробка: очищення від пропусків та аномалій, нормалізація значень, агрегація за необхідними часовими інтервалами та обчислення похідних показників, зокрема водних індексів. Структуровані та перевірені дані зберігаються у базі у вигляді `DataFrame`, придатному для подальшого аналізу та використання в алгоритмах прогнозування. Такий підхід забезпечує цілісність, узгодженість і надійність вхідної інформації.

На рисунку 3.2 діаграма показує послідовність дій користувача та системи при зборі та підготовці даних для прогнозування врожайності. Спочатку користувач відкриває веб-інтерфейс та проходить авторизацію. Після вибору конкретного поля користувач додає історичні дані врожайності. Якщо наявні дані про дати посіву та збору, система автоматично отримує погодні дані через зовнішні API, а потім агрегує їх по місяцях. Далі виконується очищення та нормалізація даних, формування структурованого `DataFrame`, який надалі використовується для прогнозування.

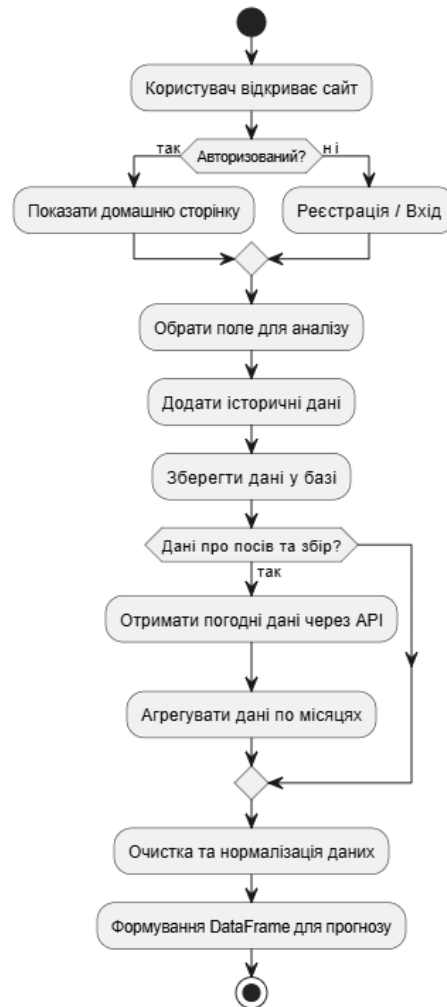


Рисунок 3.2 – Діаграма діяльності модуля збору та обробки даних

Модуль збору та обробки даних забезпечує повний цикл підготовки інформації для прогнозування: інтеграцію з зовнішніми сервісами, обробку введених користувачем даних, їх очищення, нормалізацію та агрегацію. Завдяки цьому формується структурований та цілісний набір даних, придатний для подальшого аналізу.

3.2.2 Модуль аналізу та прогнозування

Цей модуль є ядром інтелектуальної системи. Він відповідає за побудову прогнозів врожайності за допомогою методів SARIMA та Gradient Boosting. Після отримання структурованого DataFrame модуль:

- заповнює пропущені значення погодних даних та врожайності;
- виконує навчання моделей на історичних даних;

- обчислює метрики точності прогнозу (RMSE, MAE, R^2);
- виконує прогнозування врожайності для поточного та наступних років.

Комбінація статистичних (SARIMA) та машинного навчання (Gradient Boosting) методів дозволяє отримувати більш точні та надійні результати, враховуючи сезонні коливання та нелінійні залежності у даних.

На рисунку 3.3 діаграма ілюструє процес обробки підготовлених даних та виконання прогнозування врожайності. Модуль отримує DataFrame, заповнює пропущені значення, після чого навчає дві моделі: SARIMA та Gradient Boosting. Для кожної моделі обчислюються метрики точності (RMSE, MAE, R^2) для оцінки якості прогнозів.

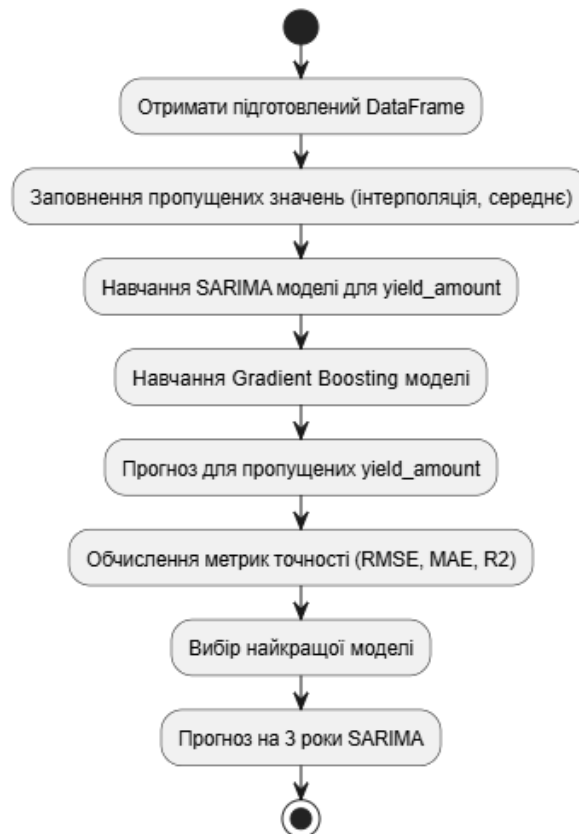


Рисунок 3.3 – Діаграма діяльності модуля аналізу та прогнозування

На основі результатів вибирається найкраща модель для прогнозування пропущених значень, а також виконується прогноз на кілька наступних років за SARIMA.

3.2.3 Модуль візуалізації прогнозу та рекомендацій

Модуль візуалізації відповідає за перетворення результатів прогнозів у зрозумілу для користувача форму. Він створює графіки фактичної та прогнозованої врожайності, таблиці з прогнозними значеннями та аналітичні звіти. Крім того, модуль генерує рекомендації щодо оптимізації агротехнологій: внесення добрив, очікуваний рівень врожайності.

Використання веб-інтерфейсу забезпечує доступність інформації на будь-яких пристроях, включаючи смартфони та планшети, що особливо важливо для фермерів у польових умовах.

На рисунку 3.4 діаграма демонструє етап відображення результатів прогнозування та формування рекомендацій для користувача. Система отримує результати прогнозів і створює графіки фактичної та прогнозованої врожайності, таблиці з прогнозними значеннями, а також рекомендації щодо оптимізації агротехнологій (наприклад, рівень внесення азоту або ризик посухи). Всі ці дані надаються у зручному вигляді для перегляду користувачем через веб-інтерфейс.



Рисунок 3.4 – Діаграма діяльності модуля візуалізації прогнозу та рекомендацій

Модуль прогнозування виконує ключову аналітичну функцію системи, формуючи точні прогнози врожайності на основі підготовлених даних. Поєднання моделей SARIMA та Gradient Boosting забезпечує врахування як сезонних закономірностей, так і складних нелінійних залежностей, що підвищує надійність результатів. Процес включає заповнення пропусків, навчання моделей та оцінку їх точності, після чого обирається найефективніший підхід для побудови прогнозів на поточний і майбутні роки.

3.3 Діаграма варіантів використання інтелектуальної системи

Для повного розуміння логіки роботи системи прогнозування врожайності та взаємодії її компонентів застосовано UML-моделювання. UML-діаграми дозволяють наочно відобразити структуру, функціональні сценарії та послідовність дій у системі, що особливо важливо при розробці складного програмного забезпечення з кількома модулями та інтеграцією зовнішніх джерел даних [31].

Для формалізації вимог до майбутнього програмного забезпечення була створена діаграма варіантів використання, яка відображає можливих акторів та зв'язки між ними. Така діаграма дає змогу проаналізувати структуру програмного забезпечення, основні функціональні можливості та способи взаємодії з зовнішніми сутностями.

На рисунку 3.5 діаграма варіантів використання відображає взаємодію користувача з системою прогнозування даних. Користувач може створити нове поле, при цьому система обов'язково включає внесення необхідної інформації про поле. Основною функцією є виконання прогнозування, що здійснюється на основі створеного поля. Після формування прогнозу користувач має можливість переглянути результати, а ця дія може бути розширена додатковою опцією експорту результатів у потрібному форматі.

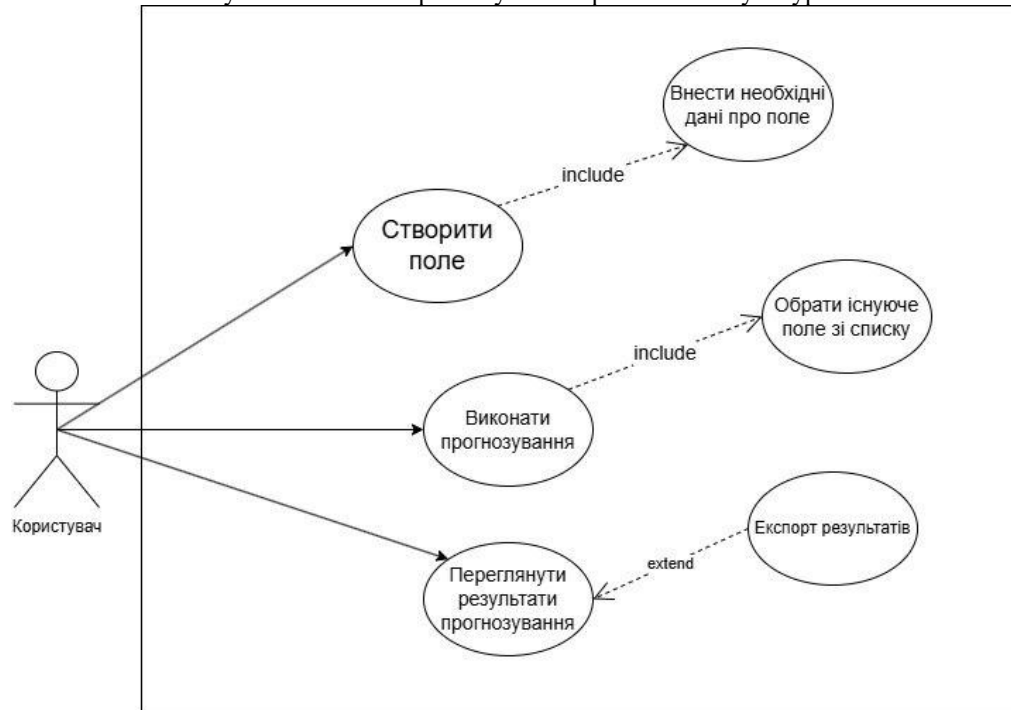


Рисунок 3.5 – Діаграма варіантів використання

Діаграма варіантів використання демонструє основні сценарії роботи користувача із системою: створення нового поля, введення необхідних даних, запуск прогнозування та перегляд отриманих результатів. Така послідовність дій забезпечує інтуїтивну взаємодію з системою та дозволяє користувачу оперативно отримувати прогнози врожайності з можливістю подальшого експорту даних.

3.4 Діаграма діяльності інтелектуальної системи

Діаграма діяльності (Activity Diagram) відображає послідовність дій, що виконуються в межах конкретного процесу або сценарію роботи системи, та дозволяє формалізувати алгоритмічні операції [32]. На відміну від Use Case Diagram, яка лише описує функції на рівні взаємодії акторів, Activity Diagram показує логічний і функціональний workflow: як дані рухаються між етапами, в якій послідовності запускаються операції, де можуть використовуватися розгалуження, цикли або паралельні процеси. Такі діаграми є ключовими для систем, які виконують послідовність складних обчислень, адже дозволяють заздалегідь виявити можливі логічні помилки, визначити місця синхронізації, залежності між модулями та потенційні точки відмови. У задачах прогнозування врожайності

Activity Diagram дозволяє чітко описати весь ланцюжок дій – від введення вхідних даних до формування рекомендацій, – що є критично важливим для реалізації архітектури системи.



Рисунок 3.6 – Діаграма діяльності

Activity Diagram є важливим інструментом для формалізації послідовності операцій у системі, оскільки дозволяє наочно відобразити рух даних, алгоритмічні переходи та логічні розгалуження. На відміну від Use Case Diagram, вона показує

не лише функції, а й внутрішню логіку їх виконання, що дає змогу виявити залежності та можливі помилки ще на етапі проєктування. Для системи прогнозування врожайності така діаграма допомагає чітко структурувати весь процес – від введення користувачем даних до отримання кінцевих рекомендацій – забезпечуючи прозорість та узгодженість роботи модулів.

3.5 Діаграма послідовності інтелектуальної системи

Діаграма послідовностей (Sequence Diagram) належить до групи поведінкових UML-діаграм та показує порядок взаємодії між об'єктами або компонентами системи в часі. Вона моделює обмін повідомленнями, що відбувається між учасниками певного сценарію, фіксуючи, який компонент викликає методи іншого, коли і в якому напрямку передаються дані. Sequence Diagram дає змогу чітко описати логіку роботи одного бізнес-процесу або функціональної можливості, включаючи часові залежності. У системах прогнозування та аналітики такі діаграми відіграють важливу роль, оскільки дозволяють показати точний порядок дій: як користувач ініціює обчислення, як сервер обробляє запит, як працюють алгоритми машинного навчання, і як результати повертаються назад користувачу.

На рисунку 3.7 зображена діаграма послідовностей моделює повний життєвий цикл процесу формування прогнозу врожайності в інтелектуальній системі, починаючи від взаємодії користувача. У контексті системи прогнозування врожайності така діаграма є критично важливою, оскільки демонструє не лише послідовність викликів між компонентами, а й відображає логіку руху даних між різними модулями: формами Django, моделями бази даних, API-провайдерами погоди, модулем обробки прогнозів. Вона інтегрує у собі як роботу з локальними даними, так і взаємодію із зовнішніми джерелами, показуючи, як система синхронізує отримані значення та запускає алгоритми машинного навчання. Оскільки прогнозування врожайності включає багатоетапні обчислення та залежності між модулями, саме така діаграма дозволяє цілісно побачити складний

сценарій роботи, забезпечуючи прозорість архітектурних рішень і точність подальшої розробки програмної логіки.

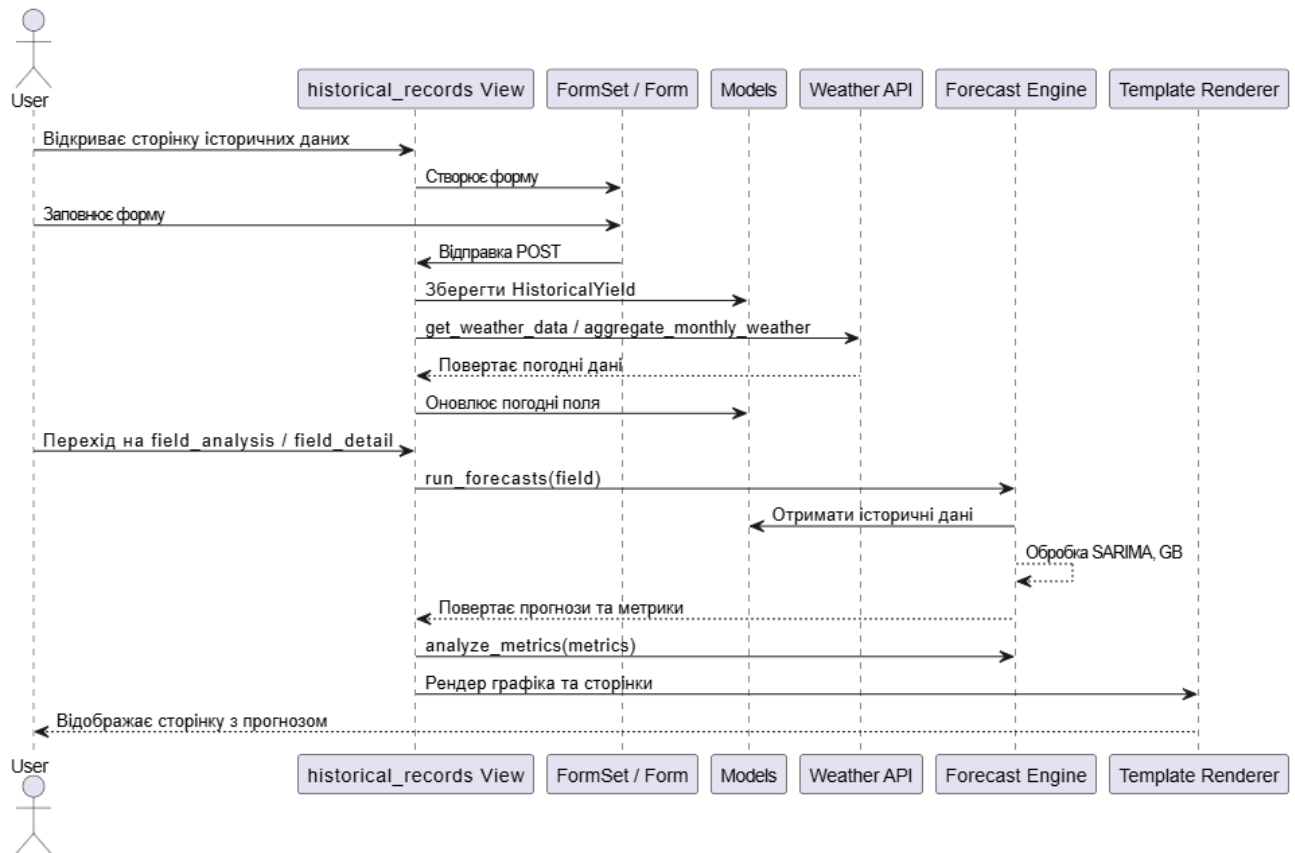


Рисунок 3.7 – Діаграма послідовності

На діаграмі детально відображено взаємодію між користувачем та ключовими компонентами системи. Процес починається з відкриття сторінки історичних даних, після чого `historical_records View` створює форму для внесення даних про врожайність. Користувач заповнює форму, надсилає POST-запит, і система зберігає об'єкти *HistoricalYield* у базі даних через Django-моделі. Після цього викликається модуль погоди – `Weather API`, який отримує актуальні метеорологічні дані та передає їх у систему для агрегування (`aggregate_monthly_weather`). Оновлені погодні значення зберігаються у моделі, після чого користувач переходить до перегляду аналітики поля (`field_analysis / field_detail`). На цьому етапі `View` викликає основний обчислювальний модуль – `Forecast Engine`, який отримує історичні дані, виконує SARIMA-моделювання та Gradient Boosting, генерує прогноз врожайності та повертає метрики точності

(RMSE, MAE, R^2). Далі результати проходять додаткову обробку у функції `analyze_metrics`, після чого система передає їх у `Template Renderer` для формування графіків, таблиць та сторінки прогнозу. Завершальним етапом є відображення користувачу згенерованої сторінки з прогнозами та рекомендаціями. Таким чином, діаграма точно демонструє, як система координує роботу різних модулів, забезпечуючи безперервний цикл – від введення даних до отримання інтелектуального прогнозу.

Use Case Diagram показує основні ролі користувачів та сценарії їх взаємодії із системою, Activity Diagram відтворює послідовність дій від введення даних до отримання прогнозу, Class Diagram демонструє структуру основних класів у Django, а Sequence Diagram ілюструє процес обміну даними між користувачем, сервером, модулями прогнозування та базою даних. Завдяки таким моделям створюється чітке уявлення про архітектуру та функціонування системи, що спрощує подальшу розробку, тестування та масштабування програмного забезпечення.

3.6 Інформаційне забезпечення

Інформаційне забезпечення є фундаментальною складовою інтелектуальної системи прогнозування врожайності, оскільки саме від якості та повноти даних залежить точність та надійність прогнозів. В межах розроблюваної системи дані поділяються на три основні категорії: вхідні дані, нормативно-довідкова інформація та агреговані результати проміжного аналізу.

Вхідні дані включають історичні показники врожайності, агрометеорологічні характеристики та агротехнічні параметри. Для малих господарств України особливо важливим є врахування локальних кліматичних аномалій, таких як різкі коливання температури, посухи чи надмірні опади. Джерелами таких даних можуть бути як державні агрометеорологічні служби, так і локальні сенсори та метеостанції, що дозволяє отримувати дані високої точності для конкретного поля.

Супутникові дані забезпечують просторову інформацію про стан рослинності, індекси вегетації (NDVI), рівень зволоження ґрунту та вплив

2025 р.

аномальних погодних явищ. Для історичних та поточних метеоданих можна використовувати сервіси Meteostat та OpenWeatherMap, які надають безкоштовний доступ через API до температури, опадів та інших кліматичних параметрів.

Інтеграція різнорідних джерел даних потребує попередньої обробки: нормалізації, заповнення пропусків, стандартизації одиниць вимірювання та перевірки на наявність аномалій. У контексті вебсистеми така інтеграція може реалізовуватись через автоматичне отримання даних через API, їх збереження у базі даних MySQL та обробку перед передачею на модулі прогнозування. Такий підхід дозволяє забезпечити системі високий рівень адаптивності до локальних умов та аномальних погодних явищ, що є критично важливим для малих господарств.

Важливим аспектом є також історичні записи врожайності культур, які слугують основою для побудови часових рядів і застосування методів прогнозування.

Нормативно-довідкова інформація включає класифікатори культур, типів ґрунтів, агротехнічних заходів, а також довідники щодо оптимальних строків сівби, норм внесення добрив та водного режиму. Ці дані виконують роль керуючих параметрів для моделей, забезпечуючи стандартизацію та коректність інтерпретації вхідних даних, а також дозволяють системі формувати рекомендації для користувача відповідно до норм агротехніки.

Організація та збереження інформації реалізується у вигляді бази даних, розробленої з використанням MySQL, що дозволяє ефективно зберігати великі обсяги структурованих і напівструктурованих даних. Дані зберігаються у таблицях з нормалізованими полями, що забезпечує швидкий доступ і коректну інтеграцію з моделями прогнозування. Для забезпечення цілісності інформації впроваджуються механізми перевірки на коректність введених даних, обробка пропусків та виявлення аномалій.

Особлива увага приділяється підготовці даних для моделей машинного навчання та глибинного навчання. Це включає нормалізацію числових показників, кодування категоріальних змінних, а також агрегування даних за певними

2025 р.

часовими інтервалами для побудови часових рядів. Такий підхід дозволяє забезпечити сумісність інформації з алгоритмами прогнозування, зменшити вплив шумів та підвищити точність прогнозів.

На рисунку 3.8 зображено логічну модель бази даних системи, що відображає основні сутності та зв'язки між ними. Центральним елементом є таблиця `predictions_field`, яка представляє сільськогосподарські поля користувача та пов'язана з обліковими записами у таблиці `auth_user`. Для кожного поля зберігаються історичні дані врожайності (`predictions_historicalyield`), результати прогнозування (`predictions_forecastresult`) та погодні показники (`predictions_weatherfeature`). Таблиця погодніх характеристик також може посилатися на конкретний історичний запис врожайності, що забезпечує можливість аналізу погодніх умов у контексті конкретної культури та року. Таким чином, модель забезпечує цілісність даних та підтримує ключові функції системи з аналізу й прогнозування врожайності.

Таблиця `predictions_historicalyield` містить детальні історичні характеристики полів, включаючи рік вирощування культури, тип ґрунту, рН, вміст азоту, фосфору та калію, органічну речовину, рівень вологості ґрунту, опади, температуру, тривалість сонячного сяйва, вологість повітря та швидкість вітру. Крім того, фіксуються дати сівби та збирання врожаю та фактичні показники продуктивності. Цей набір даних забезпечує основу для аналізу динаміки врожайності та формування якісних вибірок для навчання моделей прогнозування.

Таблиця `predictions_forecastresult` зберігає результати роботи прогнозних алгоритмів: тип моделі, передбачену врожайність, рівень довіри (`confidence`) та пояснювальний коментар. Це дозволяє відстежувати різні методи прогнозування та порівнювати їхню точність. Таблиця `predictions_weatherfeature` містить погодні характеристики для визначених дат: середню температуру, опади, вологість, а також координати точки вимірювання. Кожен запис прив'язується до конкретного поля та, за потреби, до певного історичного сезону вирощування культури. Така деталізація дозволяє системі моделювати вплив погодніх умов на врожайність у різні періоди.

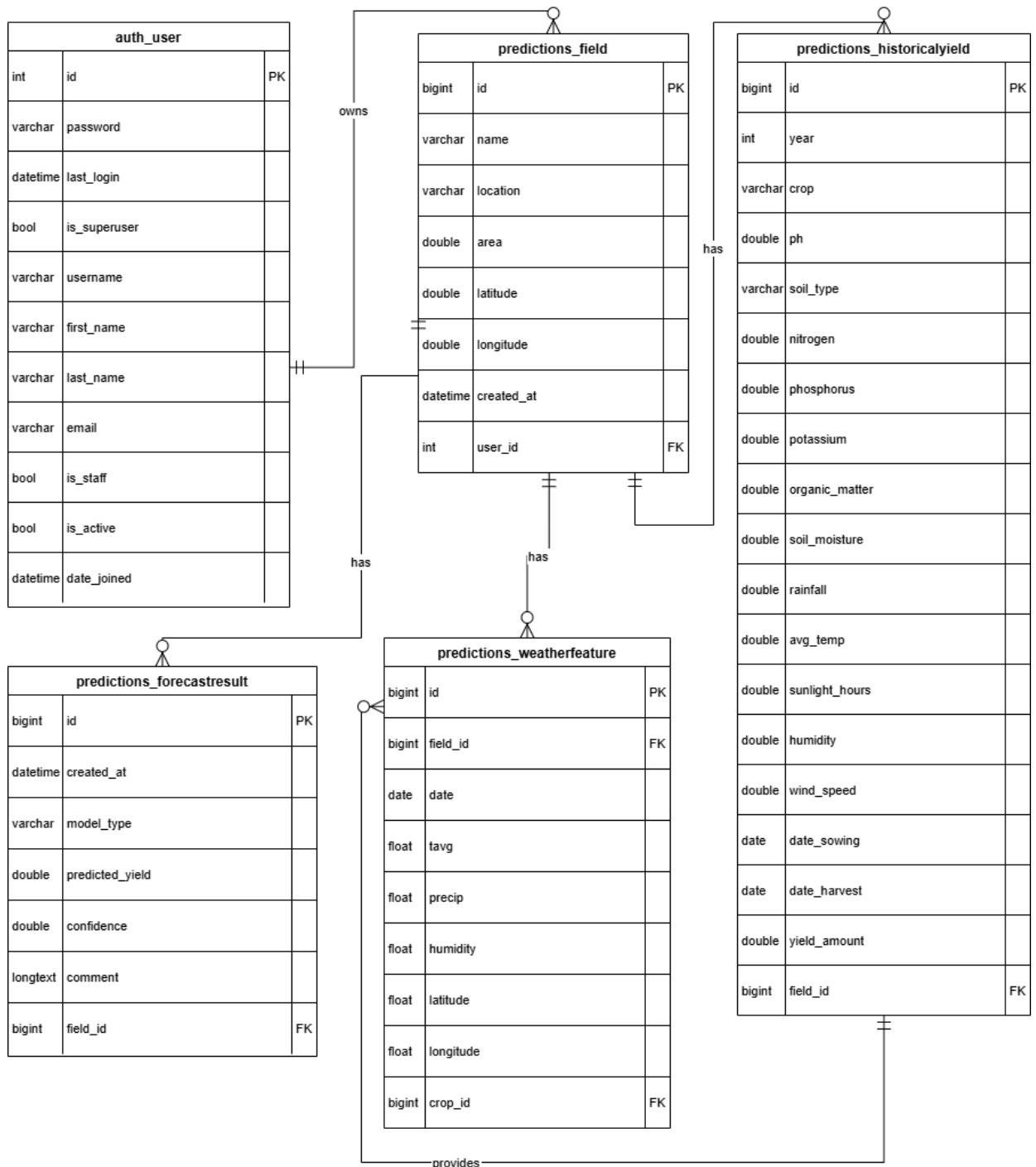


Рисунок 3.8 – Логічна структура БД

Інформаційне забезпечення розроблюваної системи включає комплекс заходів щодо збору, обробки, збереження та стандартизації даних, що забезпечує основу для точного прогнозування врожайності та формування практично корисних рекомендацій для малих фермерських господарств України.

Висновки до розділу 3

У третьому розділі здійснено комплексне проектування програмного забезпечення інтелектуальної системи прогнозування врожайності, що охоплює архітектурні рішення, вибір технологічного стеку, розробку модульної структури та побудову UML-діаграм. Сформована архітектура ґрунтується на клієнт-серверній моделі, яка забезпечує надійну взаємодію між інтерфейсом користувача та серверною частиною, де зосереджені всі обчислювальні та аналітичні процеси. Такий підхід є оптимальним для забезпечення доступності системи для малих фермерських господарств, що часто не мають потужних апаратних ресурсів.

На основі архітектурних рішень у розділі розроблено ключові програмні модулі системи: модуль збору та обробки даних, модуль аналізу та прогнозування, а також модуль візуалізації результатів і формування рекомендацій. Кожен з них виконує ізольований набір функцій, що дає змогу забезпечити чіткий поділ відповідальностей та мінімізувати ймовірність збоїв.

UML-діаграми, виконані у межах розділу, відображають структурну й поведінкову частину системи. Діаграма варіантів використання формалізує взаємодію користувачів із системою, діаграма діяльності демонструє послідовність операцій у процесі обробки та прогнозування даних, діаграма класів описує ключові сутності системи й логіку зберігання даних, а діаграма послідовності деталізує обмін повідомленнями між компонентами під час формування прогнозу.

Третій розділ сформував повноцінне архітектурне, технологічне та логічне підґрунтя для реалізації інтелектуальної системи прогнозування врожайності. Визначені рішення забезпечують її надійність, точність, масштабованість та доступність для кінцевого користувача, створюючи основу для практичної реалізації та подальшої інтеграції алгоритмів прогнозування у наступних розділах роботи.

4 РОЗРОБКА ТА ТЕСТУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

4.1 Вибір технологій та мов програмування

В основу розроблюваної інтелектуальної системи прогнозування врожайності покладено зв'язку мова програмування Python та вебфреймворк Django [34, 35]. Такий вибір є обґрунтованим як з погляду задач аналізу даних і машинного навчання, так і з погляду побудови веборієнтованого застосунку для кінцевого користувача – малого фермера. Python належить до високорівневих мов загального призначення, що відзначається лаконічним синтаксисом, низьким порогом входження та розвиненою екосистемою бібліотек для роботи з даними, статистики та штучного інтелекту. Це дозволяє зосередитися на реалізації предметної логіки та моделей прогнозування, а не на технічних деталях мови.



Рисунок 4.1 – Логотип Python

Важливою перевагою Python є активна спільнота розробників та науковців, що забезпечує постійний розвиток бібліотек і наявність численних прикладів застосування подібних підходів у сільському господарстві. Це сприяє скороченню часу розробки, оскільки частина типових завдань (імпорт даних, попередня обробка, розрахунок метрик MAE/RMSE/R²) вже має готові, перевірені реалізації. Для візуалізації результатів прогнозування можуть використовуватися бібліотеки matplotlib або Plotly, що дає змогу будувати графіки динаміки врожайності, порівнювати фактичні дані з прогнозами та відображати вплив окремих факторів.

Ключовими інструментами для реалізації інтелектуальних моделей у проєкті є бібліотеки pandas, NumPy, scikit-learn та statsmodels, кожна з яких виконує окрему, чітко визначену функцію в технологічному стеку

Бібліотека `pandas` забезпечує високорівневу роботу з табличними даними, включаючи агрегацію, очищення, перетворення часових рядів і підготовку даних до машинного навчання. Для задач прогнозування врожайності вона дозволяє структурувати історичні записи по роках, інтегрувати погодні ознаки, виконувати злиття наборів даних із різних джерел та формувати коректні вибірки для тренування моделей. Саме `pandas` визначає основу всього конвеєра підготовки даних, від початкового завантаження до формування матриці ознак.

Бібліотека `NumPy` використовується як базовий інструмент для високопродуктивних числових обчислень. Всі операції лінійної алгебри, оптимізації та статистичного аналізу внутрішньо виконуються з використанням масивів `NumPy`, що значно прискорює обробку великих масивів аграрних даних. Для задач прогнозування врожайності це є критично важливим, оскільки градієнтний бустинг та `SARIMA`-моделі оперують великими обсягами числових операцій під час тренування та підбору параметрів.

`scikit-learn` забезпечує реалізацію алгоритмів машинного навчання і використовується у проєкті як базова платформа для побудови моделей `Gradient Boosting`. Перевага цієї бібліотеки полягає в стабільності реалізації, широкому наборі інструментів для налаштування гіперпараметрів, можливості застосування крос-валідації та інтуїтивно зрозумілому API. Крім того, `scikit-learn` інтегрується з `pandas`, що спрощує побудову пайплайнів обробки даних.

Для моделювання часових рядів використовується `statsmodels`, яка містить реалізацію `ARIMA` та `SARIMA`, а також широкий набір статистичних тестів. Ця бібліотека дозволяє проводити діагностику моделей, оцінювати автокореляції та визначати оптимальні параметри. Вона є критично важливою для точного прогнозування врожайності в динаміці, коли необхідно враховувати сезонні коливання агрокліматичних показників.

Фреймворк `Django` обрано як основу серверної частини системи завдяки його орієнтації на швидку розробку, вбудовану підтримку роботи з базами даних та наявність структурованої архітектури «додатків» (apps). `Django` надає потужний ORM (`Object-Relational Mapping`), що дозволяє описувати моделі (`Field`,
2025 р.

HistoricalYield, WeatherFeature, ForecastResult) у вигляді Python-класів і прозоро зберігати їх у реляційній базі даних без написання сирих SQL-запитів. Це особливо важливо для системи, яка оперує історичними записами врожайності, метеопоказниками та результатами прогнозування. Крім того, Django має вбудовану систему автентифікації користувачів, що спрощує реалізацію облікових записів фермерів та розмежування доступу до даних.



Рисунок 4.2 – Логотип Django

Ще однією причиною вибору Django є наявність потужного механізму шаблонів і підтримка організації MVC/MVT-подібної архітектури, що чітко розділяє логіку, дані та представлення. Це дозволяє легко поєднувати аналітичну частину (моделі прогнозування, реалізовані на Python із використанням scikit-learn та statsmodels) з вебінтерфейсом, який відображає результати у зручному для користувача вигляді. У межах однієї технологічної стеку забезпечується як доступ до алгоритмів машинного навчання, так і можливість створити інтуїтивно зрозумілий вебінтерфейс для малих фермерських господарств.

У розроблюваній системі використовується MySQL, яка поєднує високу продуктивність, надійність та простоту адміністрування. MySQL є однією з найпопулярніших реляційних баз даних, що забезпечує швидку роботу з великими обсягами структурованих даних, такими як історичні врожаї, метеорологічні записи та результати прогнозування. Однією з причин вибору MySQL є її оптимізація для читання великих обсягів даних, що особливо актуально у сценаріях, де користувач часто запитує показники врожайності за різними періодами або переглядає історію прогнозів.

MySQL забезпечує ефективну підтримку індексів, транзакцій та цілісності даних, що важливо для аграрної інформаційної системи, де дані повинні

зберігатися без спотворень, а оновлення мають виконуватися коректно. Крім того, MySQL добре інтегрується з Django через вбудований ORM, що дозволяє зручно описувати структуру бази даних у вигляді моделей Python і автоматично генерувати схеми таблиць. Такий підхід значно спрощує розгортання проєкту на різних серверах, не вимагаючи глибоких знань SQL від розробника.

MySQL також легко масштабується – у разі збільшення кількості користувачів або обсягів даних можна розширити базу шляхом реплікації або міграції на більш продуктивні сервери. Це дозволяє забезпечити майбутній розвиток системи без необхідності змінювати її архітектуру.

4.2 Архітектурно-функціональний опис програмного забезпечення

Специфікація програмного забезпечення містить формальний опис структури, архітектури та функціональних компонентів розробленої системи прогнозування врожайності. Представлена специфікація забезпечує цілісне розуміння програмної системи, її функціональних можливостей та внутрішньої логіки роботи.

Архітектура розробленого програмного забезпечення ґрунтується на використанні фреймворку Django, що реалізує шаблон MVC (Model–View–Controller) у його варіації MTV (Model–Template–View). Такий підхід забезпечує логічне розділення даних, бізнес-логіки та відображення, що істотно спрощує підтримку, оновлення та масштабування системи. Логічна структура застосунку організована у вигляді окремих модулів, кожен з яких відповідає за власний сегмент функціональності – роботу з базою даних, обробку введених користувачем даних, формування прогнозів або генерацію результатів для інтерфейсу.

Основою серверної частини є модуль моделей, у якому описуються сутності предметної області – поле, історичний запис врожайності, погодні характеристики та результати прогнозування. Ці моделі формують структуру бази даних MySQL та забезпечують доступ до даних на високому рівні абстракції. Модуль представлень (views) реалізує основну логіку системи: обробку HTTP-запитів, інтеграцію з алгоритмами машинного навчання, виконання SARIMA-моделювання та побудову

2025 р.

прогнозів за допомогою Gradient Boosting. Також у цьому модулі здійснюється координація процесів очищення даних, заповнення пропусків і формування наборів ознак для моделей.

Окреме місце в архітектурі займає модуль прогнозування, який включає реалізацію алгоритмів обробки часових рядів та побудови моделей регресії. Він ізольований від веб-частини системи, що дозволяє використовувати прогнозні функції автономно, у тому числі під час подальшого розширення або інтеграції з мобільними застосунками чи API-сервісами. Додатково система містить модуль форм, який відповідає за валідацію введених користувачем даних та структурування інформації перед її передаванням до логіки обробки.

Фронтендна частина системи реалізована за допомогою шаблонів Django і забезпечує відображення ключових екранів: інтерактивної форми введення даних про поля, інтерфейсу для додавання історичних погодних спостережень, а також сторінки з візуалізацією результатів моделювання. Це дозволяє користувачеві отримувати прогноз врожайності у зручному вигляді, включно з графіками, метриками точності та сформованими рекомендаціями.

4.3 Діаграма класів інтелектуальної системи

Діаграма класів (Class Diagram) є центральним елементом структурного моделювання системи, оскільки саме вона визначає внутрішню статичну архітектуру програмного забезпечення. Вона описує основні сутності (класи), їх властивості, методи, а також типи зв'язків між ними: асоціації, композиції, агрегації, залежності та наслідування. Така діаграма дозволяє формально відобразити структуру системи та взаємодію її компонентів на рівні даних. У вебсистемах на базі Django діаграма класів зазвичай відображає структуру моделей бази даних, що полегшує проєктування структури таблиць, визначення відношень «один до багатьох» або «багато до багатьох», а також визначає цілісність даних. У випадку проєктування інтелектуальної системи прогнозування врожайності діаграма класів є необхідною для розуміння того, як зберігатимуться історичні врожаї, погодні дані, результати прогнозування та інші ключові складові.

На рисунку 4.3 представлена діаграма класів. Центральним елементом є модель Field, що містить ключові характеристики поля – геолокацію, площу, назву та зв'язок із користувачем. До кожного поля прив'язані результати прогнозування (ForecastResult) та історичні дані врожайності (HistoricalYield), які включають агрохімічні показники ґрунту, дати сівби та збору, інформацію про урожай та окремі погодні характеристики. Модель WeatherFeature зберігає погодні параметри за місяцями, пов'язані з історичною врожайністю. Також діаграма містить стандартні Django-моделі для керування користувачами, групами та дозволами (User, Group, Permission, ContentType), які забезпечують автентифікацію, авторизацію та розмежування доступу. Така структура дозволяє гнучко зберігати як вхідні, так і результуючі дані, забезпечуючи цілісність та взаємозв'язність інформації для подальшого аналізу та прогнозування.

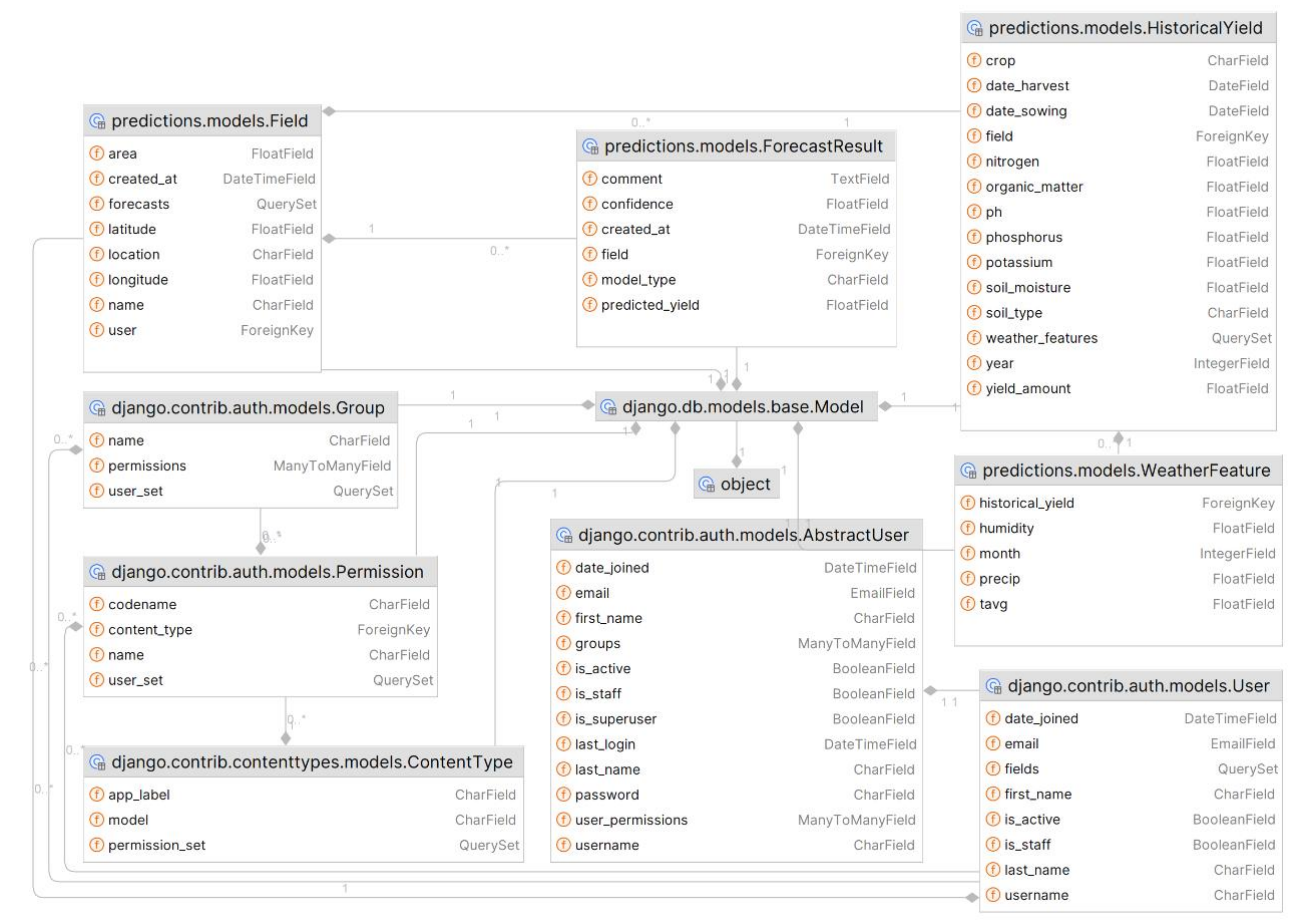


Рисунок 4.3 – Діаграма класів

У системі виділено кілька ключових класів, що реалізують основну функціональність програмного забезпечення та відображають сутності предметної області. Клас `Field` представляє окреме поле користувача та містить атрибути його назви, площі, розташування та координат (рис. 4.4). Даний клас пов'язаний із користувачем через зовнішній ключ, що дозволяє реалізувати персоналізацію даних та забезпечує збереження зв'язку між користувачем і його полями.

```
class Field(models.Model):
    user = models.ForeignKey(User, on_delete=models.CASCADE, related_name='fields')
    name = models.CharField(max_length=100, help_text="Назва для поля")
    location = models.CharField(max_length=255, help_text="Область")
    area = models.FloatField(help_text="Площа (га)")
    latitude = models.FloatField(null=True, blank=True, help_text="Широта (з центра поля). "
                                                                    "Широта повинна бути між -90 та 90")
    longitude = models.FloatField(null=True, blank=True, help_text="Довгота (з центра поля). "
                                                                    "Довгота повинна бути між -180 та 180")
    created_at = models.DateTimeField(auto_now_add=True)

    def __str__(self):
        return f"{self.name} ({self.user.username})"
```

Рисунок 4.4 – Блок коду класу `Field`

Клас `HistoricalYield` відповідає за зберігання історичних даних врожайності, включаючи характеристики ґрунту, добрива, погодні умови та періоди посіву і збору (рис. 4.5). Він пов'язаний із класом `Field`, що дозволяє вести облік показників для кожного поля окремо.

Для зручності обробки погодних даних введено клас `WeatherFeature`, який містить середні значення температури, опадів та вологості по місяцях та пов'язаний із конкретним історичним записом.

Прогнозні результати зберігаються у класі `ForecastResult`, який містить тип моделі прогнозу (SARIMA або Gradient Boosting), розраховану врожайність, довіру до прогнозу та текстовий коментар із рекомендаціями. Це дозволяє вести історію прогнозів та оцінювати їх точність.

Логіка прогнозування реалізована у функціях модуля `forecast.py`. Функція `run_forecasts(field)` формує навчальні та тестові набори даних для обох моделей, заповнює пропуски в погодних ознаках за допомогою SARIMA, здійснює навчання Gradient Boosting та прогнозування врожайності. Функція `analyze_metrics(metrics)`

обробляє розраховані метрики точності (RMSE, MAE, R^2) та формує пояснення для користувача щодо якості моделі.

```
class HistoricalYield(models.Model): 11 usages
    field = models.ForeignKey(Field, on_delete=models.CASCADE)
    year = models.IntegerField()
    crop = models.CharField(
        max_length=100,
        choices=[
            ('пшениця', 'Пшениця'),
            ('кукурудза', 'Кукурудза'),
            ('ячмінь', 'Ячмінь'),
        ],
        verbose_name="Культура"
    )
    soil_type = models.CharField(
        max_length=100,
        choices=[
            ('чорнозем', 'Чорнозем'),
            ('суглинок', 'Суглинок'),
            ('глина', 'Глина'),
            ('піщаний', 'Піщаний'),
        ],
        verbose_name="Тип ґрунту"
    )
    ph = models.FloatField(blank=True, null=True, verbose_name="pH ґрунту")
    nitrogen = models.FloatField(blank=True, null=True, verbose_name="Вміст азоту (N)")
    phosphorus = models.FloatField(blank=True, null=True, verbose_name="Вміст фосфору (P)")
    potassium = models.FloatField(blank=True, null=True, verbose_name="Вміст калію (K)")
    organic_matter = models.FloatField(blank=True, null=True, verbose_name="Органічна речовина (%)")
    soil_moisture = models.FloatField(blank=True, null=True, verbose_name="Вологість ґрунту (%)")
    date_sowing = models.DateField(null=True, blank=True, verbose_name="Дата посіву")
    date_harvest = models.DateField(null=True, blank=True, verbose_name="Дата збору")
    yield_amount = models.FloatField(blank=True, null=True, help_text="Врожайність (т/га)")
```

Рисунок 4.5 – Блок коду класу HistoricalYield

Першим кроком функція отримує історичні записи врожайності з бази даних через модель HistoricalYield та формує список ознак для кожного року, включно з характеристиками ґрунту, рівнем добрив та погодними показниками. Для врахування критичних періодів росту культур використовується словник CRITICAL_MONTHS, що визначає місяці, важливі для конкретного виду культури.

```
CRITICAL_MONTHS = {
    'пшениця': [3, 4, 5, 6],
    'кукурудза': [6, 7, 8],
    'ячмінь': [3, 4, 5]
}
```

Рисунок 4.6 – Блок коду з словником CRITICAL_MONTHS

Для заповнення пропусків у погодних даних застосовується модель SARIMA.

Якщо у колонці є відсутні значення, алгоритм будує часовий ряд по наявних даних і прогнозує відсутні значення. У разі виникнення помилок заповнення здійснюється середнім значенням колонки.

Для прогнозування врожайності у пропущені роки застосовується GradientBoostingRegressor. Модель навчається на повних даних із минулих років, а після навчання здійснює прогнозування врожайності для заданих років. Одночасно розраховуються метрики точності моделі: RMSE, MAE та R^2 (Додаток Б).

Для оцінки якості прогнозу створена функція `analyze_metrics(metrics)`, яка формує словник пояснень для кожної метрики та допомагає користувачу інтерпретувати результати прогнозу.

```
def analyze_metrics(metrics): 3 usages
    analysis = {}
    for key, value in metrics.items():
        if 'RMSE' in key:
            analysis[key] = f"{key}: {value:.2f} - чим менше, тим точніше модель відтворює фактичні дані."
        elif 'MAE' in key:
            analysis[key] = f"{key}: {value:.2f} - середня абсолютна похибка, показує середню величину помилки."
        elif 'R2' in key:
            explanation = "дуже добре" if value > 0.8 else "добре" if value > 0.5 else "слабко"
            analysis[key] = f"{key}: {value:.2f} - показує, яка частка варіації пояснюється моделлю ({explanation})."
    return analysis
```

Рисунок 4.7 – Блок коду з функцією `analyze_metrics(metrics)`

4.3 Опис основного функціоналу інтелектуальної системи

У фронтенді основні функції представлені через Django views. `historical_records` відповідає за відображення і редагування історичних даних полів, `field_detail` – за візуалізацію врожайності та прогнозів, а `field_analysis` – за формування графіків і рекомендацій для користувача. Ці методи взаємодіють із формами `FieldForm` та `HistoricalYieldForm`, що забезпечують валідацію введених даних та правильне форматування перед збереженням у базі.

Інтерфейс користувача програмного забезпечення YieldVision побудований на основі веб-фреймворку Django та шаблонів HTML, стилізованих за допомогою фреймворку Bootstrap. Загальна структура інтерфейсу орієнтована на простоту взаємодії та логічну послідовність роботи користувача – від реєстрації до 2025 р.

отримання прогнозів. Основним шаблоном є сторінка `base.html`, що визначає верхню панель навігації, кольорову стилістику та стандартне розташування контенту. У навігаційній панелі розміщено доступ до авторизації, перегляду аналітики та додавання нових полів, причому набір елементів змінюється відповідно до того, чи користувач увійшов у систему.

Сторінки реєстрації, входу та виходу забезпечують стандартну взаємодію з обліковими записами та використовують уніфіковані форми з чітким і лаконічним оформленням. Після входу користувач отримує доступ до головної сторінки сервісу, яка містить короткий опис можливостей системи, іконки сервісів та кнопку переходу до прогнозування. Головна сторінка виконує роль ознайомчого блоку та стартової точки навігації.

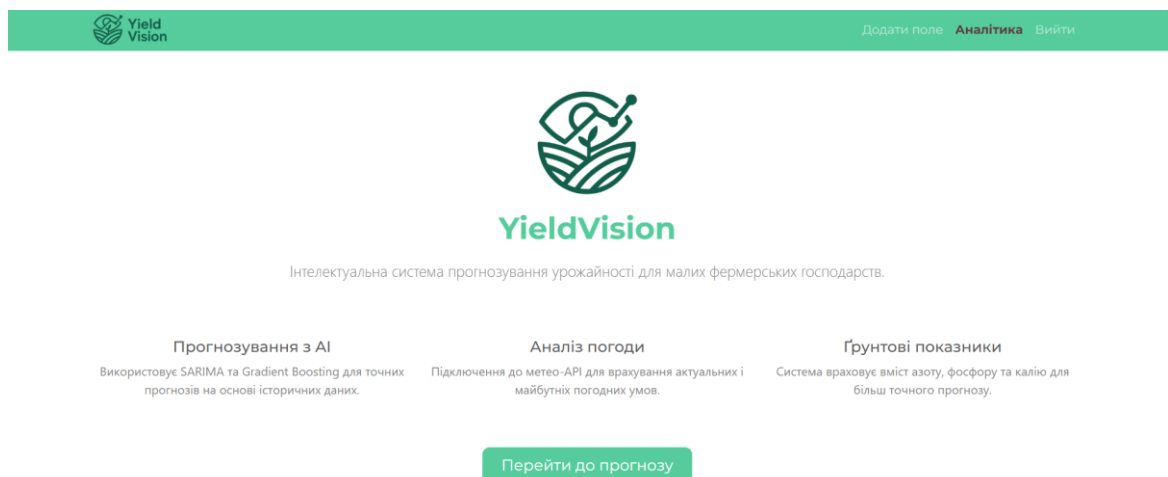


Рисунок 4.8 – Інтерфейс головної сторінки вебзастосунка

Розділ для роботи з полями реалізовано через шаблон `field_list.html`, де відображається таблиця зі всіма полями користувача, включно з назвою, площею та координатами. Для кожного поля передбачено можливість переходу до редагування історичних даних або перегляду аналітики. Додавання нового поля здійснюється на окремій формі зі зрозумілими полями введення, що забезпечують мінімізацію помилок при заповненні.

Ключовим компонентом інтерфейсу є сторінка `historical_records.html`, на якій користувач може ввести або відредагувати багаторічні дані врожайності та

параметри ґрунту. Таблиця редагування реалізована у вигляді адаптивної форми з можливістю додавання записів і видалення існуючих. Додатково реалізований механізм вибору кількості років, що дозволяє користувачу гнучко формувати набір історичних даних. На сторінці розміщується інформаційне повідомлення про те, що залишення показника врожайності порожнім дозволяє отримати прогноз для конкретного року.

Рисунок 4.9 – Інтерфейс сторінки вводу історичних даних

Таким чином, інтерфейс YieldVision поєднує в собі функціональність, інтуїтивність та візуальну лаконічність, забезпечуючи зручну взаємодію фермера з системою прогнозування та можливість швидко отримати аналітичні дані, необхідні для прийняття рішень.

4.4 Аналіз результатів прогнозування

Тестування програмного забезпечення є ключовим етапом життєвого циклу розробки, який забезпечує виявлення дефектів, оцінювання працездатності системи та підтвердження відповідності реалізованих компонентів початковим вимогам. Цей процес дозволяє мінімізувати ризики помилок на етапі експлуатації та підвищує надійність програмного продукту. Особливо важливим тестування є для систем, що виконують аналітичні розрахунки або оперують критично важливими

даними, оскільки навіть незначні відхилення у роботі можуть призвести до некоректних результатів.

Метою тестування у межах цього програмного продукту є перевірка коректності реалізованих алгоритмів, стабільності роботи користувацького інтерфейсу та узгодженості взаємодії між окремими модулями системи. Сучасні підходи до тестування передбачають комплексне оцінювання програмного забезпечення як з позиції функціональності, так і з позиції ефективності, стійкості та зручності використання. Тому у процесі оцінювання застосовувалися кілька типів тестів, спрямованих на всебічний аналіз поведінки системи в різних умовах.

Загальна стратегія тестування включала комбінування модульного, інтеграційного та системного тестування. Модульне тестування було спрямоване на ізольовану перевірку роботи окремих функцій і класів, зокрема тих, що відповідають за виконання статистичних розрахунків та формування прогнозів. Інтеграційні тести дозволили переконатися в коректному обміні даними між компонентами, а системні – оцінити поведінку продукту в цілому, включно з інтерфейсом користувача та обробкою помилок.

Окрему увагу було приділено перевірці точності математичних моделей і алгоритмів прогнозування, оскільки обчислювальні помилки можуть суттєво вплинути на коректність підсумкового результату. У межах тестування було розроблено низку тестових сценаріїв, що охоплюють як типові, так і граничні випадки, а також некоректні вхідні дані. Це дозволило оцінити стійкість системи до непередбачуваної поведінки та її здатність адекватно реагувати на помилки користувача.

Тестування окремих функцій застосунку наведено в табл. 4.1.

Таблиця 4.1 – Опис функцій та тестових випадків для операцій у системі

Функція	Опис	Тестовий випадок	Очікуваний результат
Реєстрація користувача (рис. 4.10)	Додавання нового користувача у систему	Користувач вводить коректні дані у форму реєстрації	Обліковий запис створено, користувач перенаправляється на сторінку входу

Кінець таблиці 4.1

Функція	Опис	Тестовий випадок	Очікуваний результат
Вхід у систему (рис. 4.11)	Авторизація користувача за логіном та паролем	Користувач вводить правильні дані для входу	Користувач успішно входить, відображається головна сторінка
Додавання нового поля (рис. 4.12)	Додавання інформації про нове поле	Користувач заповнює форму та натискає "Зберегти"	Поле зберігається в базі, відображається у списку полів
Введення історичних даних	Введення врожайності, NPK та інших параметрів для поля	Користувач додає дані для декількох років	Дані зберігаються коректно, всі поля заповнені без помилок
Прогнозування врожайності (рис. 4.13)	Формування прогнозів за допомогою SARIMA та Gradient Boosting	Відсутня врожайність для деяких років, натискання "Переглянути аналітику"	Відображаються графіки прогнозів, таблиці результатів та метрики точності
Вивід метрик точності моделей	Показ RMSE, MAE, R ² для моделей прогнозування	Перевірка наявності та правильності метрик після прогнозування	Метрики відображаються коректно, значення відповідають розрахункам

```
def test_registration(self):
    response = self.client.post( path: '/accounts/register/', data: {
        'username': 'newuser',
        'password1': 'StrongPass123!',
        'password2': 'StrongPass123!'
    })
    self.assertTrue(User.objects.filter(username='newuser').exists())
    self.assertEqual(response.status_code, second: 302)
```

Рисунок 4.10 – Код для тестування реєстрації користувача

```
def test_login(self):
    response = self.client.post( path: '/accounts/login/', data: {
        'username': 'testuser',
        'password': 'password123'
    })
    self.assertEqual(response.status_code, second: 302)
    self.assertTrue('_auth_user_id' in self.client.session)
```

Рисунок 4.11 – Код для тестування входу у систему

```
def test_add_field(self):
    self.client.login(username='testuser', password='password123')
    response = self.client.post(path: '/fields/add/', data: {
        'name': 'Test Field',
        'location': 'Test Location',
        'area': 15.0, # обов'язкове поле
    })
    self.assertTrue(Field.objects.filter(name='Test Field', user=self.user).exists())
    self.assertEqual(response.status_code, second: 302)
```

Рисунок 4.12 – Код для тестування додавання нового поля

```
class ForecastingTests(TestCase):
    def setUp(self):
        area=12.0
    )

    for y in range(2018, 2023):
        yield_val = None if y == 2022 else 4.5 + (y - 2018) * 0.1
        record = HistoricalYield.objects.create(
            field=self.field,
            year=y,
            ph=6.5,
            nitrogen=80,
            phosphorus=40,
            potassium=30,
            organic_matter=2.1,
            soil_moisture=23,
            yield_amount=yield_val,
            crop="пшениця"
        )
        for m in [3, 4, 5, 6]:...

    def test_run_forecasts(self):
        gb, sarima, metrics, years = run_forecasts(self.field)
        self.assertIsInstance(gb, list)
        self.assertGreaterEqual(len(gb), b: 1)
        self.assertIsInstance(sarima, list)
        self.assertGreaterEqual(len(sarima), b: 1)
        self.assertIn( member: "GB_RMSE", metrics)
        self.assertIn( member: "GB_MAE", metrics)
        self.assertIn( member: "GB_R2", metrics)
        self.assertEqual(years, second: [2018, 2019, 2020, 2021, 2022])
```

Рисунок 4.13 – Фрагмент коду для тестування прогнозування врожайності

Під час тестування було виявлено, що система адекватно обробляє різні варіанти вводу, включно з неповними чи пропущеними даними, що важливо для роботи з реальними історичними даними фермерських господарств. Метрики точності моделей SARIMA та Gradient Boosting підтвердили їхню ефективність у прогнозуванні врожайності, а аналітичний інтерфейс відображає результати у зручній та наочній формі.

В цілому, проведене тестування підтвердило, що розроблена система відповідає вимогам, закладеним у технічному завданні, та може бути рекомендована до використання у малих фермерських господарствах. Система забезпечує надійність роботи, точність прогнозів та зручність користувацького інтерфейсу, що є критично важливим для ефективного планування агротехнічних заходів.

4.5 Результати рішення

Для проведення тестування використовувалися історичні дані щодо врожайності різних культур, включно з показниками ґрунту (рН, вміст азоту, фосфору, калію, органічної речовини, вологості ґрунту) та погодними параметрами (середня температура, опади, вологість) у критичні місяці вегетації. Ці дані дозволяють оцінити, як система враховує зовнішні фактори та внутрішні характеристики ґрунту для прогнозування майбутнього врожаю.

Для побудови прогнозів використовуються дві моделі: SARIMA, яка дозволяє моделювати часові ряди врожайності та враховувати сезонні коливання, та Gradient Boosting, який забезпечує прогнозування на основі комплексного аналізу всіх доступних характеристик. Для оцінки точності моделей застосовуються стандартні метрики: середньоквадратична похибка (RMSE), середня абсолютна похибка (MAE) та коефіцієнт детермінації (R^2).

Для проведення тестування розробленої системи прогнозування врожайності були використані дані двох полів із різними наборами історичних даних. Перше поле має записи за 9 років, на яких вирощувалася пшениця, і на даний момент також росте пшениця (рис. 4.15). Друге поле містить дані за 5 років з вирощуванням кукурудзи (рис. 4.16). Такі дані дозволяють оцінити роботу системи для різних культур та тривалості спостережень.

Історичні дані:

Рік	Культура	pH	N	P	K	Орг. речов.	Вологість ґрунту	Врожайність
2017	пшениця	6.4	12.0	9.0	11.0	6.0	19.0	4.0
2018	пшениця	6.4	12.0	9.0	11.0	6.0	19.0	4.0
2019	пшениця	6.2	12.0	9.0	11.0	6.0	19.0	3.5
2020	пшениця	6.3	10.0	10.0	10.0	5.0	20.0	4.0
2021	пшениця	6.2	11.0	9.0	12.0	4.0	20.0	4.0
2022	пшениця	6.4	10.0	10.0	10.0	5.0	20.0	3.0
2023	пшениця	6.2	20.0	10.0	10.0	5.0	20.0	4.5
2024	пшениця	6.2	10.0	10.0	10.0	5.0	20.0	5.0
2025	пшениця	6.4	10.0	10.0	10.0	5.0	20.0	None

Рисунок 4.15 – Історичні дані поля з пшеницею

Історичні дані:

Рік	Культура	pH	N	P	K	Орг. речов.	Вологість ґрунту	Врожайність
2021	кукурудза	6.3	10.0	10.0	10.0	5.0	20.0	10.0
2022	кукурудза	6.5	10.0	10.0	10.0	5.0	20.0	11.0
2023	кукурудза	6.3	8.0	7.0	5.0	4.0	20.0	8.0
2024	кукурудза	6.4	10.0	6.0	7.0	5.0	20.0	9.0
2025	кукурудза	6.5	10.0	10.0	10.0	5.0	20.0	None

Рисунок 4.16 – Історичні дані поля з кукурудзою

Основними параметрами, що використовуються для прогнозування, є показники ґрунту та погодні умови. До ґрунтових характеристик відносяться: вміст азоту (N), фосфору (P), калію (K), значення pH, вміст органічної речовини та вологість ґрунту. Погодні дані включають середню температуру, кількість опадів та відносну вологість для критичних місяців вегетації культури. Комбінація цих параметрів дозволяє моделі оцінювати як внутрішні, так і зовнішні фактори, що впливають на врожайність.

Додатково для аналізу даних використовуються графічні методи візуалізації: гістограми розподілу врожайності та ключових параметрів ґрунту. Це дозволяє виявити закономірності, тренди та аномалії в даних, що може впливати на точність прогнозів і рекомендацій для користувачів. На рисунках 4.17 та 4.18 продемонстровані графіки для поля з пшеницею.

Графік прогнозів:

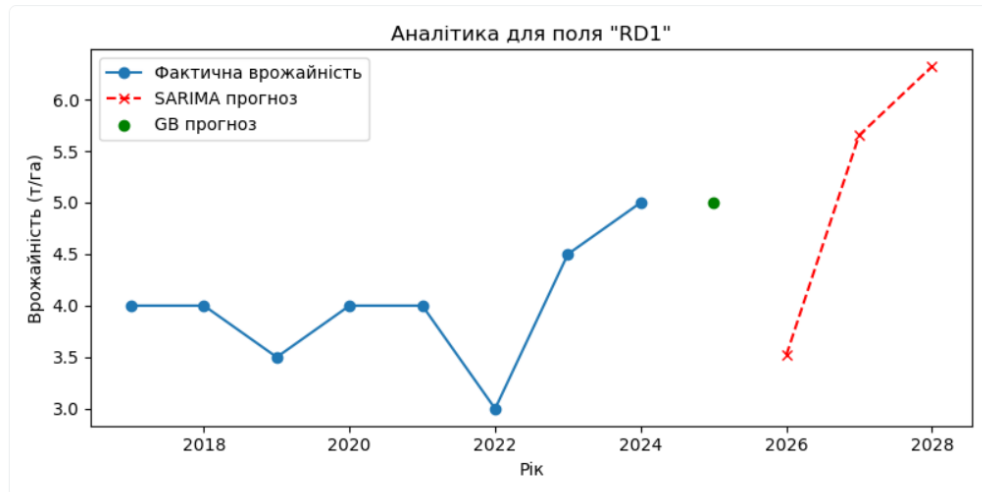


Рисунок 4.17 – Графік фактичної та прогнозованої врожайності

Графіки показників ґрунту:

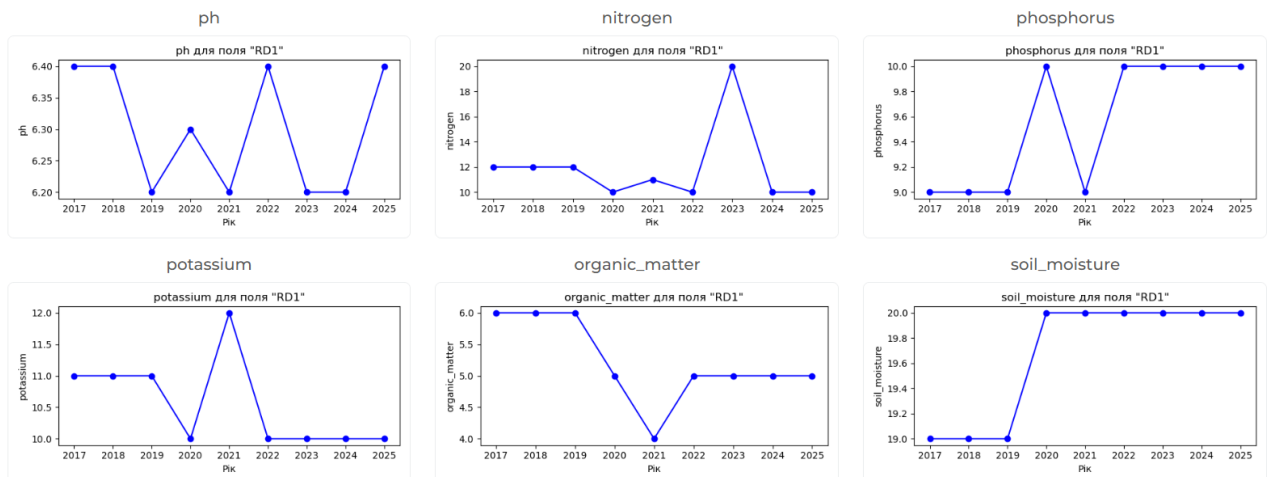


Рисунок 4.18 – Графік показників ґрунту

Для оцінки точності моделей були розраховані метрики RMSE, MAE та R^2 . Результати наведено у вигляді таблиць (рис. 4.19), що дає змогу оцінити ефективність SARIMA та Gradient Boosting. Метрики показують не лише загальну точність прогнозів, а й ступінь відповідності моделі реальній варіації врожайності.

Метрики точності моделей:

Метрика	Пояснення
GB_RMSE	GB_RMSE: 0.00 – чим менше, тим точніше модель відтворює фактичні дані.
GB_MAE	GB_MAE: 0.00 – середня абсолютна похибка, показує середню величину помилки.
GB_R2	GB_R2: 1.00 – показує, яка частка варіації пояснюється моделлю (дуже добре).

Рисунок 4.19 – Таблиця з метриками

Практична значимість системи полягає у можливості швидкого прогнозування врожайності без потреби у складних статистичних розрахунках вручну. Інтерфейс дозволяє користувачу отримати наочну візуалізацію ключових показників, таких як врожайність та вміст поживних речовин у ґрунті, а також порівняти прогнози різних моделей. Крім того, система формує рекомендації для агрономів щодо оптимізації агротехнічних параметрів, що сприяє підвищенню ефективності господарства.

Незважаючи на позитивні результати, існують певні обмеження. Система наразі використовує лише базові погодні параметри та історичні дані, що може впливати на точність прогнозів у довгостроковій перспективі. Можливі напрями покращення включають інтеграцію додаткових метеоданих, розширення історичних наборів даних, а також оптимізацію моделей машинного навчання для роботи з нестандартними або малими наборами даних.

Розроблена система є стабільним та практично корисним інструментом для підтримки рішень у сільському господарстві, дозволяючи агрономам та фермерам швидко оцінювати потенційну врожайність та приймати обґрунтовані рішення щодо ведення господарства.

4.6 Керівництво користувача

Керівництво користувача – це офіційний документ або інтерактивний ресурс, який детально описує принципи роботи з програмним забезпеченням та послідовність дій, необхідних для ефективного використання всіх функціональних можливостей системи. Його основною метою є забезпечення інтуїтивно

зрозумілого та логічного ознайомлення користувача з інтерфейсом веб-додатка, сприяння швидкому освоєнню функцій та мінімізація можливих помилок під час роботи [36].

Керівництво охоплює повний цикл роботи користувача, включаючи етапи реєстрації в системі, створення та налаштування власних полів, внесення історичних даних, запуск процесу прогнозування, перегляд результатів, аналіз графіків та таблиць, а також інтерпретацію сформованих аналітичних звітів. Крім того, воно містить докладні пояснення кожної функції, візуальні підказки, приклади виконання операцій та рекомендації щодо оптимального використання інструментів системи.

Керівництво користувача також спрямоване на підвищення доступності системи для користувачів з різним рівнем підготовки, забезпечує підтримку прийняття рішень у практичній діяльності та сприяє більш ефективному використанню ресурсів фермерських господарств. Воно може включати інтерактивні елементи, такі як контекстні підказки, відеоінструкції, інфографіку та FAQ, що значно полегшує процес навчання та підвищує комфорт роботи з програмою. Таким чином, керівництво стає не лише інструментом навчання, а й постійним помічником користувача у роботі з системою прогнозування врожайності.

На рисунку 4.20 представлено вітальну сторінку вебзастосунку для прогнозування врожайності культур YieldVision.

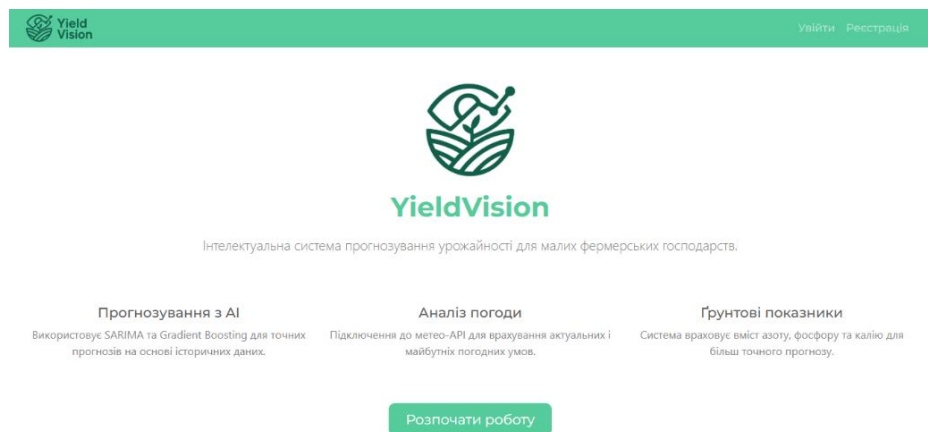
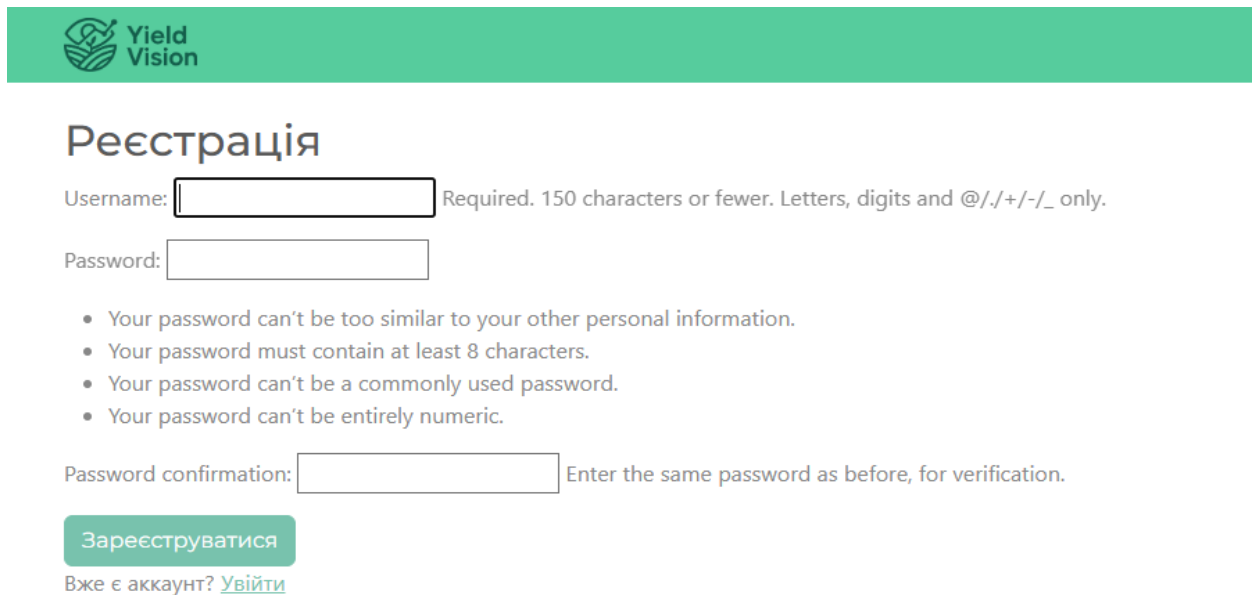


Рисунок 4.20 — Вітальня сторінка вебзастосунку YieldVision

На цій сторінці користувач може ознайомитися із коротким описом функціональних можливостей вебзастосунку, після чого перейти на сторінку авторизації або реєстрації (рис. 4.21).



Yield Vision

Реєстрація

Username: Required. 150 characters or fewer. Letters, digits and @/./+/-/_ only.

Password:

- Your password can't be too similar to your other personal information.
- Your password must contain at least 8 characters.
- Your password can't be a commonly used password.
- Your password can't be entirely numeric.

Password confirmation: Enter the same password as before, for verification.

[Зареєструватися](#)

Вже є аккаунт? [Увійти](#)

Рисунок 4.21 — Сторінка реєстрації у вебзастосунку

На цій сторінці для успішної реєстрації користувач має виконати наступні кроки:

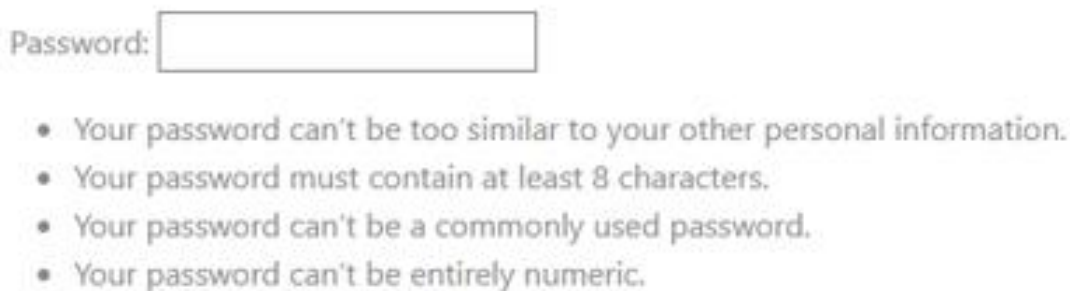
Крок 1. Введення username. Для успішної реєстрації username користувача має відповідати певним правилам. Біля поля username ці правила коротко описані(рис. 4.22).



Username: Required. 150 characters or fewer. Letters, digits and @/./+/-/_ only.

Рисунок 4.22 — Поле для username

Крок 2. Створення надійного паролю. YieldVision являється вебзастосунком, тому потрібно приділити особливу увагу паролю, щоб інші особи не змогли отримати доступ до профілю на якому може зберігатися конфіденційна інформація про поля, урожай та стан ґрунту. Саме для цього надаються рекомендації під час створення пароля (рис. 4.23).



Form for creating a password. It includes a text input field labeled "Password:" and a list of requirements:

- Your password can't be too similar to your other personal information.
- Your password must contain at least 8 characters.
- Your password can't be a commonly used password.
- Your password can't be entirely numeric.

Рисунок 4.23 — Поле для створення паролю

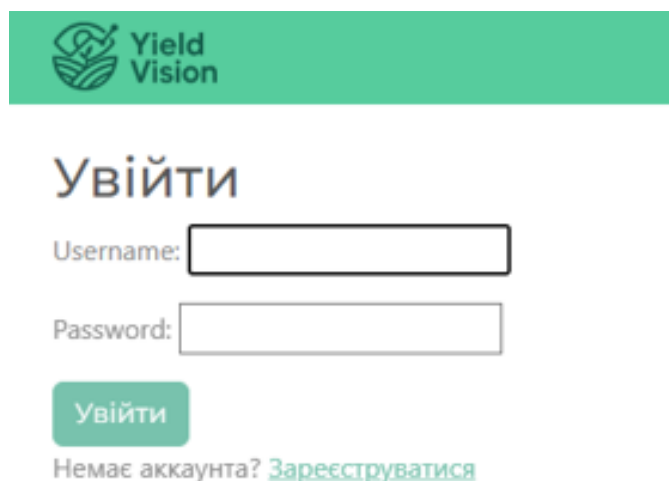
Крок 3. Підтвердження паролю. Для підтвердження паролю його треба ввести повторно, ця процедура потрібна щоб користувач мав змогу перевірити чи точно він ввів саме той пароль якій хотів(рис. 4.24).



Form for password confirmation. It includes a text input field labeled "Password confirmation:" and a placeholder text: "Enter the same password as before, for verification."

Рисунок 4.24 — Поле для підтвердження паролю

Після успішної реєстрації система попросить користувача пройти авторизацію, ввівши свій логін та пароль (рис. 4.25). Якщо користувач ввів логін або пароль неправильно то його буде проінформовано повідомленням.



Login page of the Yield Vision application. It features the Yield Vision logo at the top, followed by the heading "Увійти". Below the heading are two text input fields labeled "Username:" and "Password:". A green button labeled "Увійти" is positioned below the password field. At the bottom, there is a link that says "Немає аккаунта? [Зареєструватися](#)".

Рисунок 4.25 — Сторінка для авторизації сторінка у вебзастосунку

Після успішної авторизації користувач потрапляє на головну сторінку застосунку яка схожа із вітальною, але функціонал додавання поля та аналітика розблоковані (рис. 4.26).

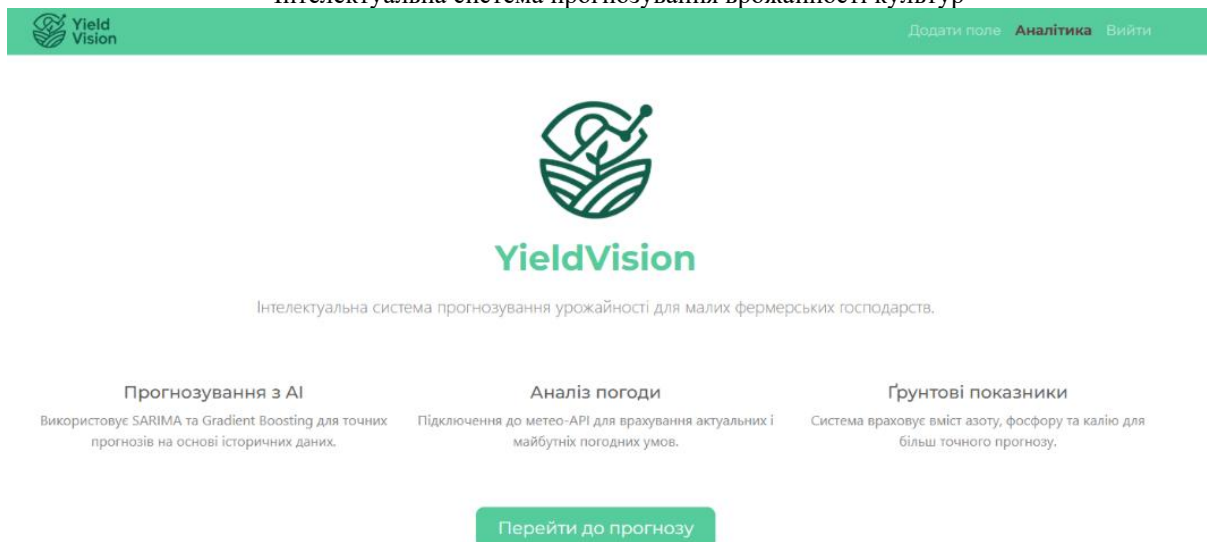


Рисунок 4.26 — Головна сторінка вебзастосунку

Щоб виконати прогнозування потрібно надати всі необхідні вхідні дані такі як інформація про поле та погодні умови. Якщо у списку полів не має потрібного або жодного поля його слід створити, на рисунку 4.27 представлено форму для заповнення інформації про поле.

Рисунок 4.27 — Сторінка для створення поля

Для кожного поля де користувач хоче провести прогнозування, користувач має ввести необхідну інформацію. Наявна підказка для чого воно призначене, і

одиницю виміру. Всі поля обов'язкові до заповнення. Після успішного створення поля воно має з'явитися у списку полів (рис. 4.28).

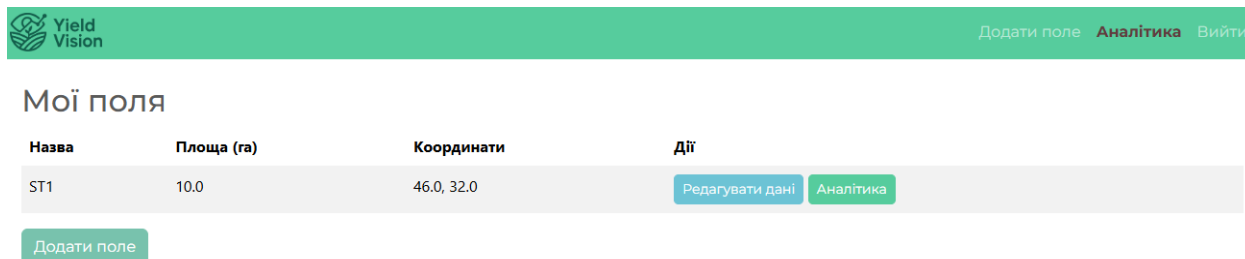


Рисунок 4.28 — сторінка для відображення списку полів

З цієї сторінки можна повернутися на сторінку для створення поля, редагування даних існуючого поля або додавання аналітики по полю.

На сторінці «Редагувати дані», користувач має додати необхідні історичні дані на основі яких буде проходити процес прогнозування врожайності та надання рекомендацій для підвищення врожаю. Чим більше аналітичних даних (мається на увазі кількість років з даними) тим точніше та якісніше буде прогноз. На рисунку 4.29 представлено сторінку де користувач вводить історичні дані поля. Для року на який потрібен прогноз треба залишити поле врожайності пустим.

Історичні дані для RD1

Для прогнозування залиште врожайність пустою

Рік	Культура	Врожайність	pH ґрунту	Тип ґрунту	NPK та органіка	Дата посіву	Дата збору	Видалити
2017	Пшени	4,0	6,4	Чорн	N: 12,0 P: 9,0 K: 11,0 Органіка: 6,0 Вологість: 19,0	14.09.2017	10.06.2018	☐
2018	Пшени	4,0	6,4	Чорн	N: 12,0 P:	14.09.2018	10.06.2019	☐

Рисунок 4.29 — Сторінка для додавання історичних даних поля

Після збереження введених даних користувач автоматично переходить на сторінку аналітики, де наочно представлена вся необхідна інформація для оцінки стану полів та прогнозування врожайності. На цій сторінці доступні кілька ключових елементів:

- історичні дані, що були введені користувачем (рис. 4.30), які дозволяють відстежити минулі результати врожайності, порівняти динаміку змін показників та виявити тренди у розвитку культур;

- графік прогнозів та фактичних даних врожайності (рис. 4.31), що дає змогу наочно оцінити відповідність прогнозних значень реальним результатам і визначити точність моделей на конкретних ділянках;

- прогнози врожайності за різними моделями (рис. 4.32), які відображають результати роботи статистичних та машинно-навчальних алгоритмів, дозволяючи користувачу порівнювати ефективність різних підходів та обирати найбільш надійний варіант прогнозу для своїх умов;

- графіки показників ґрунту та рекомендації (рис. 4.33), що демонструють ключові характеристики ґрунту, а також надають поради щодо оптимального внесення добрив;

- метрики точності моделі (рис. 4.34), що включають стандартні показники якості прогнозу, такі як середня абсолютна похибка (MAE), середньоквадратична помилка (RMSE) та коефіцієнт детермінації (R^2), які дозволяють оцінити надійність прогнозів і приймати більш обґрунтовані рішення щодо управління господарством.

Завдяки такій структурі сторінки аналітики користувач отримує повноцінну інформаційну панель, що поєднує історичні дані, прогнозні значення, аналітичні графіки та рекомендації, що значно спрощує процес планування, аналізу та прийняття рішень у межах системи. Крім того, інтерактивні елементи, такі як масштабування графіків та вибір моделей для порівняння, дозволяють користувачу оперативно оцінювати результати та адаптувати стратегію ведення господарства під поточні умови.

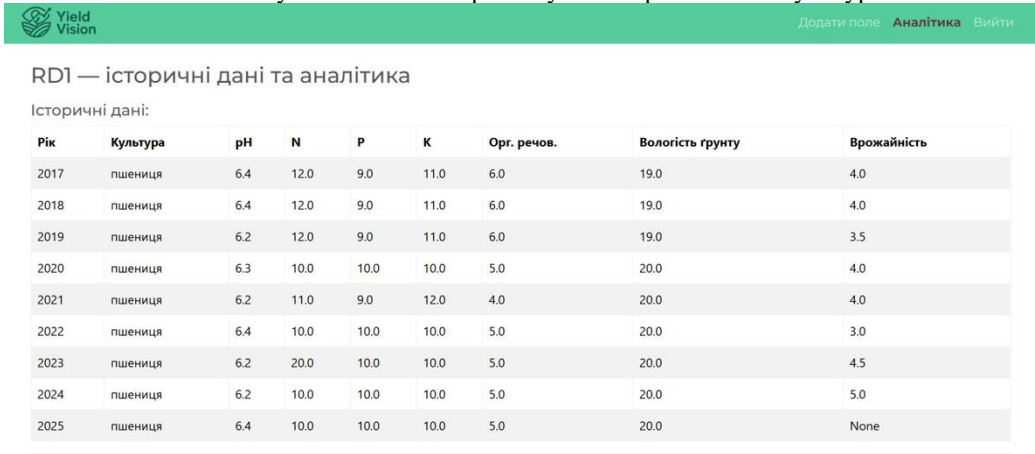


Рисунок 4.30 — Сторінка прогнозування врожайності поля

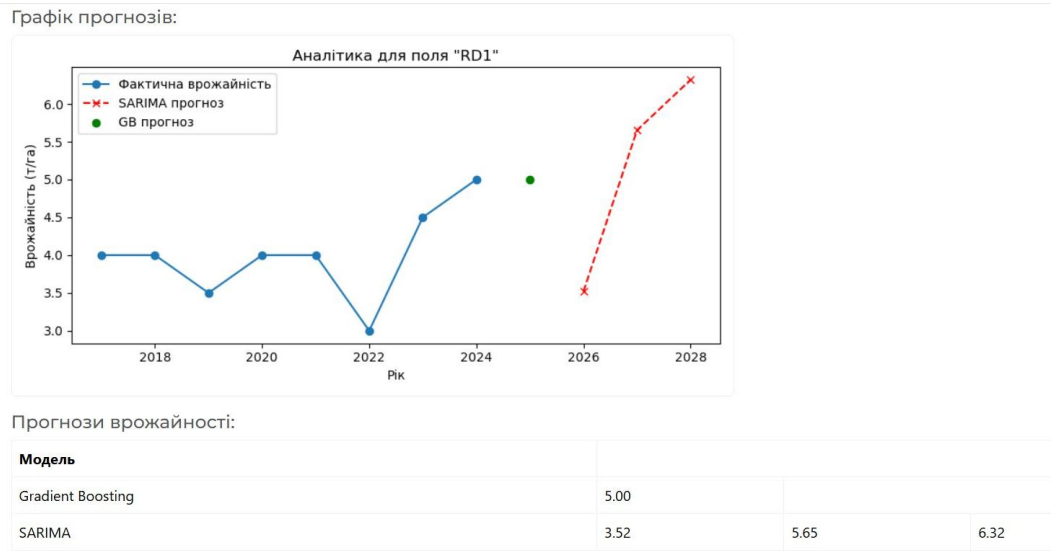


Рисунок 4.31 — Сторінка прогнозування врожайності поля

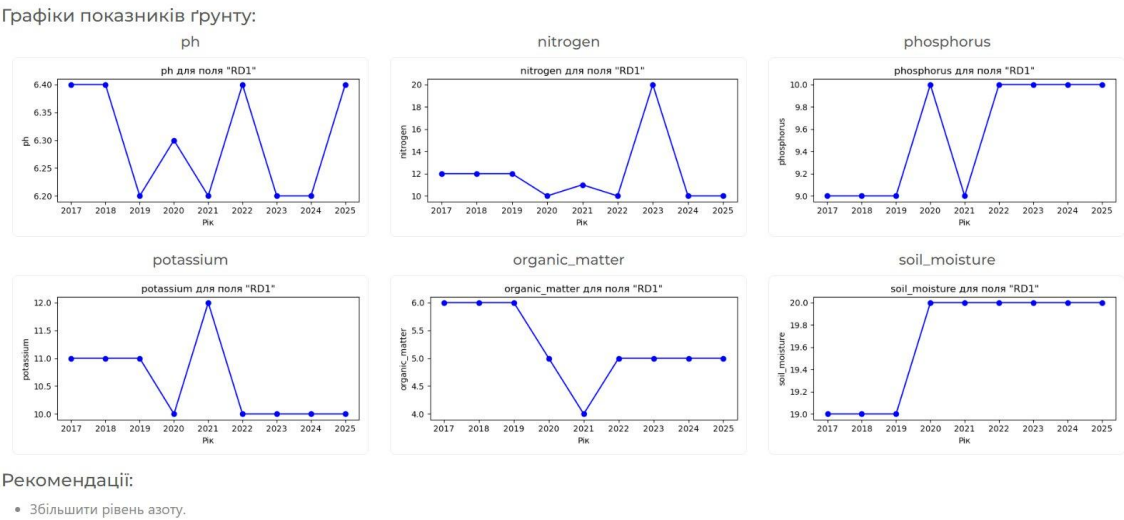


Рисунок 4.32 — Сторінка прогнозування врожайності поля

Метрики точності моделей:

Метрика	Пояснення
GB_RMSE	GB_RMSE: 0.00 – чим менше, тим точніше модель відтворює фактичні дані.
GB_MAE	GB_MAE: 0.00 – середня абсолютна похибка, показує середню величину помилки.
GB_R2	GB_R2: 1.00 – показує, яка частка варіації пояснюється моделлю (дуже добре).

Експортувати результати

Рисунок 4.33 — Сторінка прогнозування врожайності поля

Таким чином, вебзастосунок YieldVision забезпечує повний та зручний цикл роботи користувача – від створення облікового запису до отримання глибокого аналітичного звіту щодо прогнозування врожайності. Інтерфейс системи побудований таким чином, щоб кожен етап був зрозумілим, а робота з даними – максимально простою та структурованою. Завдяки послідовному виконанню описаних кроків користувач може швидко адаптуватися до функціональних можливостей застосунку й ефективно використовувати його для прийняття обґрунтованих агрономічних рішень.

Висновки до розділу 4

В четвертому розділі було представлено процес розробки, тестування та оцінювання ефективності програмного забезпечення YieldVision, призначеного для прогнозування врожайності сільськогосподарських культур.

У ході аналізу технологічних рішень аргументовано визначено вибір Python як основної мови програмування та фреймворку Django для реалізації серверної логіки. Використання бібліотек pandas, NumPy, scikit-learn та statsmodels забезпечує повний цикл роботи з даними – від очищення та структуризації до тренування моделей та валідації прогнозів. Реляційна СУБД MySQL обрана як надійне сховище структурованих даних, яке забезпечує високу продуктивність, підтримку транзакцій та інтеграцію з ORM Django.

Окрема увага була приділена реалізації алгоритмів SARIMA та Gradient Boosting, механізмам формування ознак, обробці пропусків та генерації інтерфейсних компонентів.

Також було розглянуто процес тестування системи, що охоплює модульний, інтеграційний і системний рівні. Система коректно обробляє як повні, так і неповні набори даних, забезпечуючи стабільність роботи та надійність отримуваних прогнозів.

У підрозділі 4.3 наведено результати застосування модулів прогнозування на реальних наборах історичних даних. Аналіз отриманих метрик точності показав, що моделі SARIMA та Gradient Boosting демонструють високу ефективність у передбаченні врожайності для різних культур і різної глибини історичних даних. Візуалізація результатів та сформовані рекомендації підтверджують практичну цінність системи для прийняття агротехнічних рішень.

У підрозділі 4.4 подано детальне керівництво користувача, яке забезпечує зрозумілу навігацію вебзастосунком і підтримує користувача на всіх етапах роботи – від реєстрації до аналізу прогнозів. Керівництво містить покрокові інструкції, візуальні підказки, пояснення основних функцій і приклади використання.

Загалом результати розробки та тестування підтверджують, що YieldVision є функціонально повноцінною та практично корисною системою для аграрних господарств. Вона забезпечує точне прогнозування врожайності, дисципліновану обробку даних та інтуїтивно зрозумілий інтерфейс, що робить її ефективним інструментом для підтримки прийняття рішень у сільському господарстві.

ВИСНОВКИ

У кваліфікаційній магістерській роботі було спроектовано та розроблено інтелектуальну систему прогнозування врожайності сільськогосподарських культур, орієнтовану на потреби малих фермерських господарств України. Виконане дослідження дозволило досягти поставленої мети – створити програмний інструмент, що забезпечує підвищення точності прогнозів та підтримку прийняття рішень в умовах нестабільних кліматичних факторів.

Проведено детальний аналіз предметної області та досліджено сучасні методи прогнозування врожайності. Визначено переваги та недоліки статистичних методів, алгоритмів машинного навчання, глибинного навчання та гібридних підходів. На основі аналізу встановлено, що найбільш доцільним для малих господарств є комбінування моделей часових рядів та легких ML-алгоритмів, здатних працювати з обмеженими обсягами даних.

Проаналізовано існуючі програмні рішення (Climate FieldView, John Deere Operations Center, GYGA тощо) та виявлено, що більшість із них орієнтовані на великі агрохолдинги, є високовартісними та недостатньо адаптованими до локальних умов України, що підтвердило актуальність створення спеціалізованої, доступної та гнучкої системи.

Розроблено інформаційну, функціональну та математичну моделі системи, включно з IDEF0-діаграмами, описом потоків даних, вимогами до інформаційного, технічного та програмного забезпечення. Визначено структуру вхідних даних, механізми їх обробки, зберігання та інтеграції.

Створено архітектуру програмного забезпечення, що включає модулі збору й обробки даних, модуль прогнозування та інтерактивний модуль візуалізації результатів. Для реалізації системи використано мову програмування Python та відповідні бібліотеки для аналізу даних і моделювання.

Реалізовано комбінований підхід до прогнозування, що поєднує сезонні статистичні моделі (ARIMA/SARIMA) для базового прогнозу та алгоритми Gradient Boosting для врахування нелінійних залежностей, зокрема впливу

агротехнічних та кліматичних факторів. Такий підхід продемонстрував підвищену точність щодо моделей, побудованих на одному методі.

Розроблено та протестовано вебзастосунок, який забезпечує введення даних, побудову прогнозу та візуалізацію результатів у вигляді графіків і таблиць. Створений інтерфейс є інтуїтивно зрозумілим і придатним для практичного використання фермерами без спеціальної підготовки.

Проведено оцінку ефективності системи, результати якої підтвердили працездатність обраних методів та адекватність прогнозів у межах доступних даних. Система забезпечує можливість динамічного оновлення прогнозів при надходженні нової кліматичної інформації.

Отже, у рамках кваліфікаційної роботи було успішно створено інтелектуальну систему прогнозування врожайності культур, яка поєднує сучасні алгоритми аналізу даних, адаптивність до локальних умов та доступність для малих фермерських господарств України. Розроблене рішення може бути використане як практичний інструмент для планування агротехнічних заходів, оптимізації виробничих процесів та зменшення ризиків, пов'язаних із кліматичними коливаннями. Перспективами подальших досліджень є розширення набору моделей глибокого навчання, інтеграція даних дистанційного зондування, додавання модулів рекомендацій та створення мобільної версії системи.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Mahore P. S., Bardekar D. A. A. Crop Yield Prediction. *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*. 2021. P. 561–569. DOI: 10.32628/cseit2173168
2. Simulating and Predicting Crop Yield and Soil Fertility under Climate Change with Fertilizer Management in Northeast China Based on the Decision Support System for Agrotechnology Transfer Model / W. Yan et al. *Sustainability*. 2020. Vol. 12, no. 6. P. 2194. DOI: 10.3390/su12062194
3. Jabed M. A., Azmi Murad M. A. Crop Yield Prediction in Agriculture: A Comprehensive Review of Machine Learning and Deep Learning Approaches, with Insights for Future Research and Sustainability. *Heliyon*. 2024. DOI: 10.1016/j.heliyon.2024.e40836
4. Winter wheat yield prediction using convolutional neural networks from environmental and phenological data / A. K. Srivastava et al. *Scientific Reports*. 2022. Vol. 12. DOI: 10.1038/s41598-022-06249-w
5. Maestrini B., Basso B. Drivers of within-field spatial and temporal variability of crop yield across the US Midwest. *Scientific Reports*. 2018. Vol. 8, no. 1. U DOI: 10.1038/s41598-018-32779-3
6. Precision agriculture for improving crop yield predictions: a literature review / S. Saha et al. *Frontiers in Agronomy*. 2025. Vol. 7. DOI: 10.3389/fagro.2025.1566201
7. Călin A. D., Coroiu A. M., Mureșan H. B. Analysis of Preprocessing Techniques for Missing Data in the Prediction of Sunflower Yield in Response to the Effects of Climate Change. *Applied Sciences*. 2023. Vol. 13, no. 13. P. 7415. DOI: 10.3390/app13137415
8. Global Yield Gap Atlas. URL: <https://www.yieldgap.org> (дата звернення: 12.09.2025).
9. Climate FieldView. URL: <https://climate.com/en-us.html> (дата звернення: 12.09.2025).

10. John Deere Operations Center. URL: <https://operationscenter.deere.com> (дата звернення: 12.09.2025).
11. Granular Insights. URL: <https://www.corteva.com/us/products-and-solutions/digital-solutions/granular-insights.html> (дата звернення: 12.09.2025).
12. Buklagin D. S. Agricultural crop yield forecasting methods. *Machinery and Equipment for Rural Area*. 2020. No. 12. P. 25–28. DOI: 10.33267/2072-9642-2020-12-25-28
13. Heumann C., Schomaker M., Shalabh. Linear Regression. *Introduction to Statistics and Data Analysis*. Cham, 2016. P. 249–295. DOI: 10.1007/978-3-319-46162-5_11
14. Boote K. J., Jones J. W., Pickering N. B. Potential Uses and Limitations of Crop Models. *Agronomy Journal*. 1996. Vol. 88, no. 5. P. 704–716. DOI: 10.2134/agronj1996.00021962008800050005x
15. A method review of the climate change impact on crop yield / X. Feng et al. *Frontiers in Forests and Global Change*. 2023. Vol. 6. DOI: 10.3389/ffgc.2023.1198186
16. Kumar P., Bharti P. K., Kumar B. Crop Yield Prediction using Machine Learning: A Review of Recent Approaches. *International Journal of Computer Applications*. 2023. Vol. 185, no. 24. P. 27–32. DOI: 10.5120/ijca2023922994
17. Climate change & food security / ed. by D. M. 1954- et al. New Delhi : New India Pub. Agency, 2008. 342 p.
18. Rani I., Verma N., Verma C. K. A Rigorous Statistical Comparison of Deep Learning Models for US Treasury Yield Prediction. *Operations Research Forum*. 2025. Vol. 6, no. 3. DOI: 10.1007/s43069-025-00497-y
19. Wulff S. S. Time Series Analysis: Forecasting and Control, 5th edition. *Journal of Quality Technology*. 2017. Vol. 49, no. 4. P. 418–419. DOI: 10.1080/00224065.2017.11918006
20. De Gooijer J. G. Classic Nonlinear Models. *Elements of Nonlinear Time Series Analysis and Forecasting*. Cham, 2017. P. 29–85. DOI: 10.1007/978-3-319-43252-6_2

21. Time Series Analysis: Forecasting and Control / G. M. Jenkins et al. Wiley & Sons, Incorporated, John, 2015. 712 p.
22. Sujatha, R., and P. Isakki. A study on crop yield forecasting using classification techniques. 2016 International Conference on Computing Technologies and Intelligent Data Engineering (ICCTIDE'16). IEEE, 2016. pp 1-4. DOI: 10.1109/ICCTIDE.2016.7725357.
23. Sapatinas T. The Elements of Statistical Learning. *Journal of the Royal Statistical Society: Series A (Statistics in Society)*. 2004. Vol. 167, no. 1. P. 192. DOI: 10.1111/j.1467-985x.2004.298_11.x
24. Gradient Boosting / Z. Wang et al. *Numerical Machine Learning*. 2023. P. 116–159. DOI: 10.2174/9789815136982123010007
25. Cherkassky V. The Nature Of Statistical Learning Theory~. *IEEE Transactions on Neural Networks*. 1997. Vol. 8, no. 6. P. 1564. DOI: 10.1109/tnn.1997.641482
26. LeCun Y., Bengio Y., Hinton G. Deep learning. *Nature*. 2015. Vol. 521, no. 7553. P. 436–444. DOI: 10.1038/nature14539
27. JIANG, Zehui, et al. Predicting county level corn yields using deep long short term memory models. 2018.
28. Bengio Y., Simard P., Frasconi P. Learning long-term dependencies with gradient descent is difficult. *IEEE Transactions on Neural Networks*. 1994. Vol. 5, no. 2. P. 157–166. DOI: 10.1109/72.279181
29. Garber D., Kaplan A., Sabach S. Improved complexities of conditional gradient-type methods with applications to robust matrix recovery problems. *Mathematical Programming*. 2019. DOI: 10.1007/s10107-019-01452-6
30. Albeladi K., Zafar B., Mueen A. Time Series Forecasting using LSTM and ARIMA. *International Journal of Advanced Computer Science and Applications*. 2023. Vol. 14, no. 1. DOI: 10.14569/ijacsa.2023.0140133
31. Kabbilawsh P., Kumar D. S., Chithra N. R. Forecasting long-term monthly precipitation using SARIMA models. *Journal of Earth System Science*. 2022. Vol. 131, no. 3. DOI: 10.1007/s12040-022-01927-9

32. The unified modeling language user guide / ed. by R. James, J. Ivar. 2nd ed. Upper Saddle River, NJ : Addison-Wesley, 2005. 475 p.
33. The object primer: Agile modeling-driven development with UML 2.0. 3rd ed. Cambridge, UK : Cambridge University Press, 2004. 545 p.
34. Welcome to Python.org. URL: <https://www.python.org> (дата звернення: 12.10.2025).
35. Django: The web framework for perfectionists with deadlines. URL: <https://www.djangoproject.com> (дата звернення: 12.10.2025).
36. Sriram D. Designing the user interface: Strategies for effective human-computer interaction. *Artificial Intelligence in Engineering*. 1987. Vol. 2, no. 2. P. 125–126. DOI: 10.1016/0954-1810(87)90189-0

ДОДАТОК А

Апробація кваліфікаційної роботи

Результати досліджень були представлені на «Могилянські читання – 2025»: Досвід та тенденції розвитку суспільства в Україні: глобальний, національний та регіональний аспекти: XXVIII Всеукраїнській науково-практичній конференції: тези доповідей: Комп'ютерні науки. Технічні науки, Миколаїв, 10–14 листопада 2025 р. / ЧНУ ім. Петра Могили. – Миколаїв: Вид-во ім. Петра Могили, 2025р.

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Чорноморський національний університет імені Петра Могили
ДНУ «Інститут модернізації змісту освіти»
Південний науковий центр НАН та МОН
Інститут української археографії та джерелознавства
імені М. С. Грушевського НАН України
Первинна профспілкова організація ЧНУ імені Петра Могили



XXVIII ВСЕУКРАЇНСЬКА ЩОРІЧНА НАУКОВО–ПРАКТИЧНА КОНФЕРЕНЦІЯ

**«МОГИЛЯНСЬКІ ЧИТАННЯ–2025: досвід та
тенденції розвитку суспільства в Україні:
глобальний, національний та регіональний
аспекти»**

ПРОГРАМА

Миколаїв, 10–14 листопада 2025 року

Миколаїв – 2025

Рисунок А.1 – Обкладинка збірника тез конференції Могилянські читання –
2025

ДОДАТОК Б

Програмна реалізація прогнозування

```
import pandas as pd
from statsmodels.tsa.statespace.sarimax import SARIMAX
from sklearn.ensemble import GradientBoostingRegressor
from sklearn.metrics import mean_squared_error, mean_absolute_error,
r2_score
from .models import HistoricalYield, ForecastResult
from math import sqrt

CRITICAL_MONTHS = {
    'пшениця': [3, 4, 5, 6],
    'кукурудза': [6, 7, 8],
    'ячмінь': [3, 4, 5]
}

def run_forecasts(field):
    records =
HistoricalYield.objects.filter(field=field).order_by('year')
    if not records.exists():
        return None, [], {}, [], [], []

    features_list = []
    years = []

    for r in records:
        features = {
            'ph': r.ph or None,
            'nitrogen': r.nitrogen or None,
            'phosphorus': r.phosphorus or None,
            'potassium': r.potassium or None,
            'organic_matter': r.organic_matter or None,
            'soil_moisture': r.soil_moisture or None,
        }

        month_features = r.weather_features.all()
        for wf in month_features:
            month = wf.month
            if month in CRITICAL_MONTHS.get(r.crop, []):
                features[f'tavg_m{month}'] = wf.tavg
                features[f'precip_m{month}'] = wf.precip
                features[f'hum_m{month}'] = wf.humidity

        features['yield_amount'] = r.yield_amount
        features_list.append(features)
        years.append(r.year)

    df = pd.DataFrame(features_list)

    weather_cols = [col for col in df.columns if
```

```
col.startswith(('tavg_', 'precip_', 'hum_'))]
for col in weather_cols:
    if df[col].isnull().any():
        known = df[col].dropna()
        if len(known) < 2:
            df[col] = df[col].fillna(known.mean() if not
known.empty else 0)
            continue

    try:
        model = SARIMAX(known, order=(1,1,1),
seasonal_order=(1,1,1,3)).fit(dispatch=False)
        n_missing = df[col].isnull().sum()
        forecast =
model.get_forecast(steps=n_missing).predicted_mean.tolist()
        df.loc[df[col].isnull(), col] = forecast
    except Exception as e:
        df[col] = df[col].fillna(known.mean())

df_train = df[df['yield_amount'].notnull()]
df_predict = df[df['yield_amount'].isnull()]

X_train = df_train.drop(columns='yield_amount').dropna()
y_train = df_train.loc[X_train.index, 'yield_amount']
X_pred = df_predict.drop(columns='yield_amount').dropna()

gb_forecast = []
metrics = {}

if not X_train.empty and not X_pred.empty:
    gb = GradientBoostingRegressor(n_estimators=100,
random_state=0)
    gb.fit(X_train, y_train)
    y_pred = gb.predict(X_pred)
    gb_forecast = list(y_pred)

    y_train_pred = gb.predict(X_train)
    rmse = sqrt(mean_squared_error(y_train, y_train_pred))
    mae = mean_absolute_error(y_train, y_train_pred)
    r2 = r2_score(y_train, y_train_pred)
    metrics['GB_RMSE'] = rmse
    metrics['GB_MAE'] = mae
    metrics['GB_R2'] = r2

sarima_forecast = []
df_yield = df_train['yield_amount']
if not df_yield.empty:
    try:
        model = SARIMAX(df_yield, order=(1,1,1),
seasonal_order=(1,1,1,3)).fit(dispatch=False)
        sarima_forecast =
model.get_forecast(steps=3).predicted_mean.tolist()
    except Exception as e:
```

```
return gb_forecast, sarima_forecast, metrics, years

def analyze_metrics(metrics):
    analysis = {}
    for key, value in metrics.items():
        if 'RMSE' in key:
            analysis[key] = f"{key}: {value:.2f} - чим менше, тим
точніше модель відтворює фактичні дані."
        elif 'MAE' in key:
            analysis[key] = f"{key}: {value:.2f} - середня абсолютна
похибка, показує середню величину помилки."
        elif 'R2' in key:
            explanation = "дуже добре" if value > 0.8 else "добре"
            if value > 0.5 else "слабко"
            analysis[key] = f"{key}: {value:.2f} - показує, яка
частка варіації пояснюється моделлю ({explanation})."
    return analysis
```