

**ПОЛТАВСЬКИЙ ДЕРЖАВНИЙ АГРАРНИЙ УНІВЕРСИТЕТ**  
**Навчально-науковий інститут економіки, управління, права та**  
**інформаційних технологій**  
**Кафедра інформаційних систем та технологій**

## **КВАЛІФІКАЦІЙНА РОБОТА**

на здобуття ступеня вищої освіти магістр

на тему: **«Розроблення мобільного додатка персонального асистента  
фермера на основі локальних великих мовних моделей»**

Виконав: здобувач вищої освіти  
за освітньою програмою  
Інформаційні управляючі системи та  
технології  
спеціальності 126 Інформаційні системи  
та технології  
ступеня вищої освіти магістр  
групи 126ІСТ\_мд\_2024  
Шубка Максим Ігорович  
Керівник: Поночовний Юрій Леонідович  
Рецензент: Муравльов Володимир  
Вячеславович

**Полтава – 2025 року**

**ПОЛТАВСЬКИЙ ДЕРЖАВНИЙ АГРАРНИЙ УНІВЕРСИТЕТ**  
**Навчально-науковий інститут економіки, управління, права та**  
**інформаційних технологій**  
**Кафедра інформаційних систем та технологій**

Освітня програма Інформаційні управляючі системи та технології  
Спеціальність 126 Інформаційні системи та технології  
Рівень вищої освіти другий (магістерський)

**ЗАТВЕРДЖУЮ**  
Завідувач кафедри  
\_\_\_\_\_ Юрій УТКІН  
«08» листопада 2024 року

**З А В Д А Н Н Я**  
**НА КВАЛІФІКАЦІЙНУ РОБОТУ ЗДОБУВАЧА ВИЩОЇ ОСВІТИ**  
**Шубки Максима Ігоровича**

1. Тема кваліфікаційної роботи:  
«Розроблення мобільного додатка персонального асистента фермера на основі локальних великих мовних моделей»,  
Керівник роботи: д. т. н., професор, професор кафедри інформаційних систем та технологій Поночовний Юрій Леонідович.  
Затверджено наказом закладу вищої освіти від «31» жовтня 2025 року № 1332-ст
2. Строк подання здобувачем вищої освіти роботи «09» грудня 2025 р.
3. Вихідні дані до роботи: методи та підходи локальної обробки природної мови (NLP), локальні великі мовні моделі (LLM), архітектура трансформерних нейронних мереж, бібліотека llama.cpp, формат квантизованих моделей GGUF, кросплатформна розробка мобільних додатків (Flutter), мобільні операційні системи Android та iOS, принципи автономних інтелектуальних систем.
4. Зміст пояснювальної записки (перелік питань, які потрібно розробити):  
Розділ 1. Теоретичні основи застосування локальних великих мовних моделей в агросекторі.  
Розділ 2. Аналіз предметної області та постановка задачі розробки мобільного асистента фермера.  
Розділ 3. Проектування та реалізація мобільного додатка персонального асистента фермера.
5. Перелік графічного матеріалу: схеми, рисунки, діаграми за темою та об'єктом дослідження.

## 6. Консультанти розділів кваліфікаційної роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання отримав
Оцінювання економічної ефективності результатів дослідження	Калініченко О. В., к. е. н., доцент, доцент кафедри економіки та публічного управління	24.11.2025	04.12.2025

## 7. Дата видачі завдання «08» листопада 2024 р.

### КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів роботи	Строк виконання етапів кваліфікаційної роботи	Примітка
1.	Вибір і затвердження теми роботи	29.10.2024 р.	
2.	Складання і затвердження розгорнутого плану та завдання на кваліфікаційну роботу	30.10.2024 р. – 08.11.2024 р.	
3.	Опрацювання джерел інформації	11.11.2024 р. – 27.12.2024 р.	
4.	Збір, вивчення і обробка інформації, необхідної для виконання роботи	30.12.2024 р.– 19.01.2025 р.	
5.	Виконання теоретико-методологічного розділу роботи	17.02.2025 р.– 16.05.2025 р.	
6.	Виконання дослідницько-аналітичного розділу роботи	02.06.2025 р.– 13.07.2025 р.	
7.	Виконання проектно-рекомендаційного розділу роботи	08.09.2025 р.– 14.11.2025 р.	
8.	Оцінювання економічної ефективності результатів дослідження	24.11.2025 р.– 04.12.2025 р.	
9.	Оформлення тексту роботи	05.12.2025 р.– 08.12.2025 р.	
10.	Попередній захист роботи на кафедрі	09.12.2025 р.	
11.	Доопрацювання роботи з урахуванням зауважень і пропозицій	10.12.2025 р.- 14.12.2025 р.	
12.	Нормоконтроль	15.12.2025 р. – 16.12.2025 р.	
13.	Захист кваліфікаційної роботи	18.12.2025 р.	

**Здобувач вищої освіти**

**Максим ШУБКА**

**Керівник роботи**

**Юрій ПОНОЧОВНИЙ**

**ПОЛТАВСЬКИЙ ДЕРЖАВНИЙ АГРАРНИЙ УНІВЕРСИТЕТ  
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ЕКОНОМІКИ, УПРАВЛІННЯ,  
ПРАВА ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ  
КАФЕДРА ІНФОРМАЦІЙНИХ СИСТЕМ ТА ТЕХНОЛОГІЙ**

**ШУБКА МАКСИМ ІГОРОВИЧ**

**«РОЗРОБЛЕННЯ МОБІЛЬНОГО ДОДАТКА ПЕРСОНАЛЬНОГО  
АСИСТЕНТА ФЕРМЕРА НА ОСНОВІ ЛОКАЛЬНИХ ВЕЛИКИХ  
МОВНИХ МОДЕЛЕЙ»**

Освітньо-професійна програма  
Інформаційні управляючі системи та технології  
Спеціальність 126 Інформаційні системи та технології  
Ступінь вищої освіти Магістр

**РЕФЕРАТ**  
кваліфікаційної роботи на здобуття кваліфікації –  
магістр з інформаційних систем та технологій

Полтава – 2025 року

Кваліфікаційна робота складається зі вступу, 3 розділів, висновків, списку використаних джерел (43 найменування), 2 додатків. Кваліфікаційна робота містить 6 таблиць, 10 рисунків, викладена на 74 сторінках.

### **Основний зміст роботи**

У першому розділі розглянуто сучасний стан та основні тенденції розвитку мобільних інформаційних систем у аграрному секторі. Проаналізовано роль цифровізації у підвищенні ефективності сільськогосподарського виробництва, зокрема використання мобільних додатків як інструментів підтримки прийняття рішень безпосередньо в польових умовах. Окрему увагу приділено аналізу існуючих підходів до застосування штучного інтелекту та мовних моделей у аграрних інформаційних системах, включаючи хмарні AI-сервіси та локальні інтелектуальні рішення. Визначено ключові недоліки хмарних підходів, серед яких залежність від стабільного інтернет-з'єднання, затримки обробки запитів, додаткові фінансові витрати та ризики порушення конфіденційності даних. На основі проведеного аналізу обґрунтовано доцільність використання локальних великих мовних моделей як основи для створення автономних мобільних асистентів фермера.

У другому розділі виконано обґрунтування функціональних та нефункціональних вимог до програмної архітектури мобільного додатка персонального асистента фермера. Запропоновано модульну багаторівневу архітектуру, що забезпечує автономність роботи, слабку зв'язаність компонентів та ефективне використання обмежених обчислювальних ресурсів мобільного пристрою. Детально розглянуто призначення та взаємодію основних функціональних модулів, зокрема модуля управління взаємодією з користувачем, модуля керування контекстом діалогу, локального інтелектуального ядра на основі мовної моделі та допоміжних сервісних компонентів. Обґрунтовано вибір кросплатформного підходу до розробки мобільного додатка з використанням фреймворку Flutter та доцільність інтеграції нативних обчислювальних модулів для реалізації ресурсоемних операцій інференсу.

У третьому розділі розглянуто практичну реалізацію мобільного додатка персонального асистента фермера з використанням локальної великої мовної моделі. Описано процес вибору та інтеграції мовної моделі з використанням бібліотеки llama.cpp, а також застосування квантизованих моделей у форматі GGUF для зменшення вимог до оперативної пам'яті та підвищення продуктивності. Розглянуто механізми взаємодії між кросплатформним інтерфейсом користувача та нативним модулем інференсу. Проведено тестування прототипу на мобільному пристрої середнього класу з оцінкою швидкодії, стабільності роботи та енергоспоживання. Також наведено економічне обґрунтування розробки, яке підтверджує доцільність використання локального підходу за рахунок відсутності витрат на хмарні сервіси та серверну інфраструктуру.

## **Висновки**

1. Проведено аналіз сучасного стану та тенденцій розвитку аграрних інформаційних систем, що підтвердив зростаючу роль мобільних додатків і методів штучного інтелекту у підтримці прийняття рішень у сільському господарстві. Встановлено, що локальні великі мовні моделі є перспективною альтернативою хмарним AI-сервісам завдяки автономності роботи та зменшенню затримок обробки інформації. Отримані результати свідчать про доцільність використання локальних інтелектуальних підходів у мобільних аграрних системах.

2. Обґрунтовано функціональні та нефункціональні вимоги до програмної архітектури мобільного асистента фермера з урахуванням особливостей його використання в польових умовах. Основними вимогами визначено автономність, продуктивність, стабільність роботи та ефективне використання ресурсів мобільного пристрою. Сформульовані вимоги стали основою для вибору архітектурних і технологічних рішень у межах роботи.

3. Запропоновано модульну архітектуру мобільного додатка з розмежуванням логіки взаємодії з користувачем, інтелектуального ядра та сервісних компонентів. Такий підхід забезпечує слабку зв'язаність компонентів, спрощує процес розробки й тестування та створює передумови для подальшого розширення функціональних можливостей системи.

4. Реалізовано прототип мобільного додатка персонального асистента фермера з інтеграцією локальної мовної моделі, який забезпечує діалогову взаємодію та збереження контексту розмови в автономному режимі. Проведене тестування підтвердило стабільну роботу системи та прийнятний час відповіді при використанні ресурсомістких обчислень.

5. Проведено економічне обґрунтування розробки, яке показало доцільність використання локального підходу завдяки відсутності витрат на хмарні сервіси та серверну інфраструктуру. Запропоноване рішення характеризується низькими експлуатаційними витратами та має потенціал подальшого практичного застосування і масштабування.

Таким чином, поставлені у магістерській роботі завдання виконано у повному обсязі, а отримані результати підтверджують наукову та практичну значущість розробленого мобільного додатка персонального асистента фермера.

## **Список публікацій здобувача**

1. Шубка М. «Стартап-ініціативи у сфері аграрних інформаційних технологій: інноваційні рішення для відновлення України»: Матеріали III Міжнародної науково-практичної конференції «Стратегічний менеджмент агропродовольчої сфери в умовах глобалізації економіки: безпека, інновації, лідерство» від 23 вересня 2025 р., ПДАУ, Полтава. С. 143-145.

## АНОТАЦІЯ

Шубка М. І. «Розроблення мобільного додатка персонального асистента фермера на основі локальних великих мовних моделей». Кваліфікаційна робота на правах рукопису.

Кваліфікаційна робота на здобуття ступеня вищої освіти магістр за освітньо-професійною програмою Інформаційні управляючі системи та технології спеціальності 126 Інформаційні системи та технології. Полтавський державний аграрний університет, Полтава, 2025.

Виконано аналіз сучасного стану використання штучного інтелекту та мовних моделей у мобільних аграрних інформаційних системах, зокрема вимог до автономної роботи та обмежень, пов'язаних із використанням хмарних сервісів. Обґрунтовано вимоги до програмної архітектури мобільного додатка персонального асистента фермера та запропоновано модульну архітектуру з локальним інтелектуальним ядром. Розроблено прототип мобільного додатка з інтеграцією локальної великої мовної моделі на основі бібліотеки Llama.cpp, реалізовано механізми обробки текстових запитів і збереження контексту діалогу. Проведено тестування та економічне обґрунтування розробленого програмного продукту.

Ключові слова: мобільний додаток, персональний асистент фермера, аграрні інформаційні системи, локальні великі мовні моделі, штучний інтелект, автономна робота.

## ANNOTATION

Shubka M.I. «Development of a Mobile Farmer Assistant Application Based on Local Large Language Models». Qualification work as a manuscript.

Qualification work for obtaining the Master's degree in the educational and professional program *Information Management Systems and Technologies*, specialty 126 *Information Systems and Technologies*. Poltava State Agrarian University, Poltava, 2025.

The thesis analyzes the current state of using artificial intelligence and language models in mobile agricultural information systems, with particular attention to the requirements for autonomous operation and the limitations associated with cloud-based services. The requirements for the software architecture of a mobile farmer assistant application are substantiated, and a modular architecture with a local intelligent core is proposed. A prototype mobile application with an integrated local large language model based on the llama.cpp library is developed, including mechanisms for processing user text queries and maintaining dialogue context. Testing and economic justification of the developed software product are carried out.

Keywords: mobile application, farmer assistant, agricultural information systems, local large language models, artificial intelligence, autonomous operation.

## ЗМІСТ

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ .....	7
ВСТУП .....	8
РОЗДІЛ 1 ТЕОРЕТИЧНІ ОСНОВИ ЗАСТОСУВАННЯ ЛОКАЛЬНИХ ВЕЛИКИХ МОВНИХ МОДЕЛЕЙ В АГРОСЕКТОРІ .....	11
1.1 Сучасний стан розвитку штучного інтелекту та великих мовних моделей.....	11
1.2 Особливості та переваги локальних великих мовних моделей.....	15
1.3 Застосування штучного інтелекту та мовних моделей у сільському господарстві .....	18
1.4 Архітектура трансформерів та технології оптимізації мовних моделей для автономної роботи.....	22
Висновки до розділу 1 .....	25
РОЗДІЛ 2 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАДАЧІ РОЗРОБКИ МОБІЛЬНОГО АСИСТЕНТА ФЕРМЕРА .....	27
2.1 Аналіз предметної області та існуючих рішень у сфері цифрового фермерства .....	27
2.2 Порівняльний аналіз існуючих програмних рішень та AI-асистентів для агросектору .....	31
2.3 Формування вимог до мобільного додатка персонального асистента фермера.....	36
2.4 Постановка задачі розроблення та вибір концепції програмного рішення .....	42
РОЗДІЛ 3 ПРОЄКТУВАННЯ ТА РЕАЛІЗАЦІЯ МОБІЛЬНОГО ДОДАТКА ПЕРСОНАЛЬНОГО АСИСТЕНТА ФЕРМЕРА.....	48
3.1 Загальна архітектура мобільного додатка персонального асистента фермера.....	48
3.2 Проєктування програмної архітектури та взаємодії компонентів системи .....	52

3.3 Реалізація основних функціональних модулів та інтерфейсу користувача .....	60
3.4 Тестування, оцінка продуктивності та аналіз результатів роботи додатка.....	65
3.5 Економічне обґрунтування розробки мобільного додатка .....	69
Висновки до розділу 3 .....	71
ВИСНОВКИ.....	73
СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ .....	75
ДОДАТКИ.....	80

## СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ

AI – Artificial Intelligence (штучний інтелект)

API – Application Programming Interface (інтерфейс прикладного програмування)

CPU – Central Processing Unit (центральний процесор)

GGUF – General Graph Unified Format (формат зберігання квантизованих мовних моделей)

GPU – Graphics Processing Unit (графічний процесор)

LLM – Large Language Model (велика мовна модель)

NPU – Neural Processing Unit (нейронний процесор для прискорення AI-обчислень)

RAM – Random Access Memory (оперативна пам'ять)

UI – User Interface (інтерфейс користувача)

UX – User Experience (користувацький досвід)

## ВСТУП

*Актуальність* теми роботи обумовлена стрімким розвитком технологій штучного інтелекту (ШІ) та їх впровадженням у всі сфери людської діяльності, включаючи аграрний сектор. Сучасні фермерські господарства функціонують у складних умовах, які визначаються кліматичними змінами, непередбачуваністю погодних умов, нестачею трудових ресурсів, високими вимогами до ефективності виробництва та необхідністю оперативного прийняття рішень. Це формує потребу в інтелектуальних інструментах, здатних зменшувати когнітивне навантаження на фермера, автоматизувати рутинні задачі та підвищувати продуктивність аграрних процесів.

Найбільш перспективним напрямом підтримки фермерської діяльності сьогодні є використання великих мовних моделей (LLM), які демонструють високий рівень розуміння природної мови, здатність до узагальнення інформації, надання рекомендацій та аналізу контекстних даних. До недавнього часу використання LLM вимагало доступу до хмарних обчислювальних сервісів, що створювало значні обмеження, зокрема залежність від інтернет-з'єднання, підвищені витрати на обробку даних та ризики витоку конфіденційної інформації. Проте у 2023–2025 рр. відбувся прорив у розвитку локальних, оптимізованих мовних моделей, здатних працювати без підключення до мережі, зокрема завдяки бібліотекам llama.cpp, MLC LLM, TensorRT-LLM та появі компактних моделей сімейств LLaMA, Mistral, Gemma, Phi тощо.

Це створює можливість побудови автономного мобільного інтелектуального асистента фермера, який забезпечує обробку запитів, аналіз даних, генерацію рекомендацій і роботу з локальною базою знань без інтернету, що є критично важливим для сільськогосподарської діяльності, де покриття мережі є нестабільним або відсутнім.

*Мета* кваліфікаційної роботи полягає у розробці концепції, архітектури та програмної реалізації мобільного додатка персонального асистента фермера

на основі локальних великих мовних моделей, який забезпечує автономність, швидкодію та практичну користь у вирішенні прикладних аграрних задач.

*Завданнями* кваліфікаційної роботи є:

- проаналізувати сучасний стан розвитку ШІ та локальних LLM, їх можливості та обмеження;
- дослідити інструменти, методи та платформи для реалізації локальних LLM у мобільних додатках;
- сформулювати вимоги до функціоналу інтелектуального асистента фермера;
- розробити архітектуру та прототип мобільного додатка на основі локальної LLM;
- провести порівняльний аналіз продуктивності різних моделей та інструментів;
- здійснити економічне обґрунтування доцільності впровадження даного рішення.

*Об'єктом дослідження* є процес взаємодії користувача (фермера) з інтелектуальними мовними моделями при виконанні агротехнічних задач.

*Предметом дослідження* є методи адаптації та використання локальних великих мовних моделей у складі мобільного застосунку персонального асистента фермера.

*Методи* дослідження включають: системний аналіз, моделювання, порівняльний аналіз інструментів розробки, формалізацію вимог, експериментальне тестування продуктивності мовних моделей, проєктування програмної архітектури та економічні розрахунки.

*Інформаційну базу* роботи складають наукові публікації, технічна документація мовних моделей, матеріали офіційних репозиторіїв, аналітичні огляди систем штучного інтелекту, а також власні напрацювання автора з кваліфікаційної роботи бакалавра.

Наукова новизна дослідження полягає в обґрунтуванні та розробці архітектури мобільного асистента фермера на основі локальних, а не хмарних

LLM, що дозволяє забезпечити автономність роботи, знизити витрати на інференс та підвищити рівень конфіденційності даних.

*Практична значущість* роботи полягає у можливості впровадження розробленого рішення на фермерських господарствах, використанні для навчання персоналу, підтримки прийняття рішень та автоматизації аграрних процесів.

*Апробація результатів* дослідження відбувалася шляхом оприлюднення доповідей на наукових конференціях, семінарах.

*Публікації.* За результатами проведеного дослідження опубліковано тези: «Стартап-ініціативи у сфері аграрних інформаційних технологій: інноваційні рішення для відновлення України»: Матеріали III Міжнародної науково-практичної конференції «Стратегічний менеджмент агропродовольчої сфери в умовах глобалізації економіки: безпека, інновації, лідерство» від 23 вересня 2025 р., ПДАУ, Полтава.

*Структура кваліфікаційної роботи* логічно пов'язана з завданнями досліджень і містить вступ, три розділи основної частини, висновки, список використаних джерел, додатки. Загальний обсяг пояснювальної записки кваліфікаційної роботи складає 74 сторінки формату А4. Вона містить 10 рисунків 6 таблиць.

# РОЗДІЛ 1

## ТЕОРЕТИЧНІ ОСНОВИ ЗАСТОСУВАННЯ ЛОКАЛЬНИХ ВЕЛИКИХ МОВНИХ МОДЕЛЕЙ В АГРОСЕКТОРІ

### 1.1 Сучасний стан розвитку штучного інтелекту та великих мовних моделей

Штучний інтелект (ШІ) пройшов довгий шлях становлення – від ранніх математичних моделей 1950-х років до надпотужних генеративних систем, здатних аналізувати та створювати складні інформаційні структури. Трансформація цієї галузі відбувалася хвилеподібно: періоди активного розвитку змінювалися так званими «зимами ШІ». Проте сучасний етап, що триває з середини 2010-х років, характеризується безпрецедентним проривом, пов'язаним з появою глибокого навчання та великих мовних моделей (LLM). Саме LLM сформували основу нового покоління цифрових систем, здатних навчатися на гігантських масивах даних та генерувати контекстно релевантні відповіді, що раніше вважалося виключно людською компетенцією.

Перші концептуальні засади ШІ були започатковані А. Тюрингом, який у 1950 році поставив запитання: чи може машина мислити? З середини ХХ століття дослідники намагалися створювати експертні системи, що працювали на основі ручного формування правил. Проте такі підходи виявилися складними в масштабуванні. Справжній прорив відбувся у 2012 році після успіху нейронної мережі AlexNet, яка продемонструвала значну перевагу глибоких багат шарових моделей над традиційними методами машинного навчання. Саме тоді стало зрозуміло, що масштабування моделей за кількістю параметрів та даних веде до суттєвого підвищення якості [1].

У 2017 році дослідники Google запропонували архітектуру Transformer у роботі “Attention Is All You Need”. Це стало поворотним моментом: механізм self-attention дозволив моделі аналізувати залежності між елементами тексту не послідовно, а паралельно, що забезпечило небувалу масштабованість. На

відміну від рекурентних мереж, трансформери могли навчатися на величезних корпусах тексту та відтворювати високорівневі логічні залежності.

У подальші роки з'явилися серії моделей, які заклали фундамент сучасного генеративного ШІ:

- BERT (2018) – модель контекстної двобічної уваги, що змінила підходи до NLP [2].

- GPT-2 і GPT-3 (2019–2020) – моделі, що продемонстрували здатність до генерації зв'язного тексту на основі коротких підказок.

- T5, XLNet, PaLM, Chinchilla – моделі, що розширили масштабування та показали важливість оптимального співвідношення параметрів і обсягу даних.

Особливою подією став вихід GPT-3, який містив 175 млрд параметрів і продемонстрував можливість *few-shot learning*, де модель виконує нове завдання після кількох прикладів без додаткового тренування. Це стало переломним моментом для галузі, оскільки такі системи почали перевершувати традиційні алгоритми та експертні системи в реальних умовах.

У 2023–2025 роках розвиток LLM став ще динамічнішим. З'явилися моделі, що не лише масштабувалися, але й оптимізувалися. Навіть моделі з відносно невеликою кількістю параметрів (1–7 млрд) почали демонструвати якість, порівнянну або вищу, ніж великі моделі попередніх поколінь.

Особливе значення мають відкриті та локальні моделі:

- LLaMA / LLaMA 2 / LLaMA 3 (Meta) – моделі, оптимізовані для дослідницьких і локальних застосувань.

- Mistral 7B – приклад компактної моделі, що перевершує більші моделі GPT-3 класу.

- Gemma (Google) – серія моделей, орієнтована на енергоефективність та локальну інференцію.

- Phi (Microsoft) – «маленькі, але потужні» моделі, що доводять: якість залежить не лише від розміру, а й від якості навчальних даних [3].

Однією з ключових тенденцій є перехід від “великих” до “розумних” моделей. У 2024–2025 роках дослідження показали, що правильно підібраний корпус даних та оптимізація дозволяють компактним моделям випереджати гігантів з сотнями мільярдів параметрів. Це створило умови для переходу до локальних рішень.

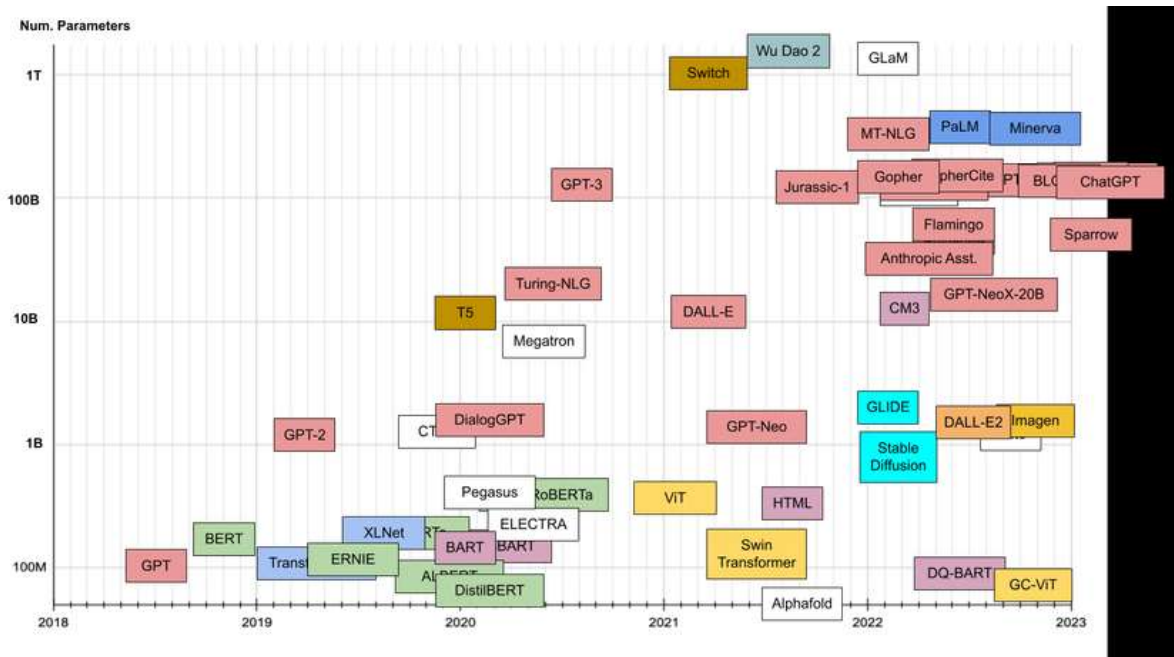


Рисунок 1.1 – Динаміка зростання параметрів великих мовних моделей у 2018–2024 рр

Хмарні моделі були першими масово доступними LLM, оскільки їх навчання та виконання вимагає значних обчислювальних ресурсів. Вони мають переваги у швидкодії, масштабі та доступі до оновлень. Проте існують вагомні обмеження:

- залежність від інтернет-підключення;
- високі витрати на API-запити;
- ризики приватності даних;
- затримки у передачі запитів (latency) [4].

Локальні моделі вирішують більшість цих проблем. Проривним стало впровадження квантування (4-bit, 8-bit), що зменшило розмір моделей у 4–8

разів без значної втрати якості. Завдяки цьому моделі на 3–7 млрд параметрів можна запускати на смартфонах або ноутбуках.

Таблиця 1.1 – Порівняння хмарних та локальних LLM

Ознака	Хмарні LLM	Локальні LLM
Робота без інтернету	Ні	Так
Конфіденційність	Обмежена	Висока
Вартість використання	Висока	Майже нульова
Затримка відповіді	Вища	Низька
Потреба в ресурсах	На сервері	На пристрої

Таблиця демонструє принципові відмінності між двома підходами та підкреслює переваги локальних моделей для мобільних застосунків.

Серед ключових факторів, які сприяли сучасному прогресу, виділяють:

1. Зростання апаратної потужності мобільних пристроїв. Процесори Apple A17, Snapdragon 8 Gen 3 та подібні мають спеціалізовані NPU для виконання моделей ШІ.

2. Оптимізація та нові формати. GGUF, GPTQ, AWQ – формати, що забезпечують швидку інференцію на звичайних пристроях.

3. Уніфіковані фреймворки для локального запуску. llama.cpp, MLC LLM, TensorRT-LLM дозволяють розгортати моделі без складної інфраструктури.

Взаємодія цих факторів створила умови, у яких LLM можуть працювати у повністю автономному режимі – що є фундаментально важливим для агросектору та мобільних застосунків.

Розвиток штучного інтелекту пройшов шлях від простих експертних правил до автономних інтелектуальних систем, здатних працювати без підключення до інтернету. Великі мовні моделі стали основою нової парадигми взаємодії людини та цифрових систем, а їх локальні реалізації відкривають можливості для високоефективних мобільних рішень. Це робить

їх надзвичайно перспективними для створення персонального асистента фермера, який може працювати в польових умовах, забезпечуючи оперативний доступ до знань та рекомендацій.

## **1.2 Особливості та переваги локальних великих мовних моделей**

Поява локальних великих мовних моделей (LLM), здатних працювати без підключення до Інтернету, стала однією з найважливіших тенденцій у розвитку сучасного штучного інтелекту. Якщо перше покоління LLM вимагало використання потужних хмарних дата-центрів, то з 2023–2025 років спостерігається стрімкий перехід до компактних, оптимізованих моделей, що можуть виконувати обчислення безпосередньо на пристрої користувача. Цей перехід став можливим завдяки технічним проривам у сфері квантування, стиснення моделей, оптимізації алгоритмів інференсу та вдосконаленню мобільних апаратних платформ.

У контексті аграрних застосувань локальні LLM відкривають принципово нові можливості, оскільки забезпечують доступ до інтелектуальних функцій незалежно від місця розташування фермера. Більшість польових робіт виконуються в умовах відсутності або нестабільності мобільного покриття, тому автономна робота на пристрої є не лише бажаною, але й необхідною. Таким чином, локальні LLM стають технологією, яка дозволяє переносити складні інтелектуальні алгоритми у реальне середовище без залежності від зовнішньої інфраструктури.

Сучасні мобільні пристрої оснащені потужними обчислювальними компонентами, включаючи:

- високопродуктивні CPU (ARM Cortex X4, Apple A-серії);
- мобільні GPU, здатні виконувати паралельні обчислення;
- NPU (Neural Processing Unit), розроблені спеціально для ШІ-алгоритмів;

– оперативну пам'ять обсягом 8–16 ГБ, чого достатньо для виконання моделей у квантуванні 4-bit.

У поєднанні з оптимізованими форматами моделей (GGUF, GPTQ, AWQ) це дозволяє запускати мовні моделі обсягом 1–7 млрд параметрів прямо на смартфоні. Ранні експерименти продемонстрували, що моделі на кшталт LLaMA 3.2 3B, Mistral 7B або Gemma 2B можуть працювати з прийнятною швидкістю навіть на середньобюджетних пристроях, забезпечуючи час відповіді 0,3–1,2 секунди для коротких запитів.

Найважливішою характеристикою локальних мовних моделей є їхня здатність працювати повністю офлайн. Це особливо важливо для фермерів, які:

- проводять більшість часу в полі, на віддалених ділянках та відкритій місцевості;
- не можуть розраховувати на стабільний мобільний зв'язок;
- потребують миттєвих відповідей у режимі реального часу.

У таких умовах хмарні сервіси виявляються малопродатними: будь-яка втрата мережі робить інтелектуального асистента недоступним. Локальна модель, навпаки, забезпечує повну незалежність від Інтернету, дозволяє виконувати генерацію тексту, аналізувати документи та надавати рекомендації без затримок і зовнішніх ризиків [5].

Одним із найвагоміших аргументів на користь локальних LLM є високий рівень конфіденційності. У хмарних моделях усі дані користувача передаються на сторонні сервери, що створює ризики витоку або несанкціонованого доступу. В аграрному секторі це може стосуватися:

- інформації про врожайність;
- технологічних карт;
- даних про експериментальні ділянки;
- внутрішньої документації підприємства.

Локальні моделі обробляють дані безпосередньо на смартфоні чи комп'ютері користувача, не передаючи їх зовнішнім серверам. Це забезпечує

максимально можливий рівень захисту та дозволяє впроваджувати ШІ навіть у середовищах з підвищеними вимогами до інформаційної безпеки.

Для ілюстрації цієї переваги доцільно навести порівняння, представлене у таблиці 1.2.

Таблиця 1.2 – Порівняння конфіденційності хмарних та локальних моделей

Параметр	Хмарні LLM	Локальні LLM
Передача даних	Дані передаються на сервер	Дані не виходять з пристрою
Ризик витоку	Середній/високий	Мінімальний
Контроль користувача	Обмежений	Повний
Залежність від політики сервісу	Висока	Відсутня

Оптимізації моделей дозволяють досягати високої швидкодії навіть на мобільних платформах. Основними технічними механізмами є:

- квантування – зменшення точності параметрів моделі до 8, 4 або навіть 2 бітів;
- структурне та неструктурне прорідження (pruning);
- дистиляція – навчання меншої моделі на виходах більшої;
- оптимізація компіляторів (TVM, TensorRT, XNNPACK).

У 2024 році було показано, що модель у квантуванні 4-bit споживає у 3–5 разів менше пам'яті, ніж її оригінальна FP16-версія, при цьому зниження якості відповіді становить лише 2–6% залежно від завдання [6]. Це робить локальні моделі не лише можливими, але й доцільними для мобільних пристроїв.

Одним із найсуттєвіших факторів є економічна ефективність. Використання хмарних моделей передбачає витрати на кожен API-запит, і для бізнесу чи фермерського господарства це може становити значні суми.

Локальний асистент не потребує оплати за використання: єдині витрати – це енергоспоживання пристрою.

Згідно з дослідженням McKinsey (2023), локальні моделі можуть зменшити вартість експлуатації інтелектуального рішення у 30–50 разів у порівнянні з хмарними аналогами при інтенсивному використанні.

Локальні LLM дозволяють налаштовувати модель під специфічні потреби користувача. Зокрема, для аграрного асистента можливо:

- додавати локальні інструкції, документи, агрономічні рекомендації;
- інтегрувати базу знань про конкретні культури чи поля;
- створити індивідуальну RAG-систему, що працює повністю офлайн;
- адаптувати стиль спілкування моделі до потреб фермера.

Такі можливості практично недоступні у стандартних хмарних сервісах, які мають уніфіковані алгоритми для всіх користувачів.

Локальні великі мовні моделі сьогодні є одним із найбільш перспективних напрямів ШІ. Їх автономність, безпека, висока продуктивність та здатність до персоналізації роблять їх оптимальним вибором для агросектору. Можливість інтеграції в мобільні застосунки створює умови для появи повнофункціональних інтелектуальних асистентів, які можуть працювати у польових умовах, підтримувати процес прийняття рішень та забезпечувати оперативний доступ до агрономічних знань.

### **1.3 Застосування штучного інтелекту та мовних моделей у сільському господарстві**

Сільське господарство традиційно вважається однією з найскладніших галузей для управління, оскільки його результативність залежить від численних факторів: кліматичних умов, стану ґрунтів, наявності шкідників, рівня технологічної підготовки персоналу та доступності ресурсів. У таких умовах ефективне прийняття рішень часто потребує глибоких знань,

професійного досвіду та постійного моніторингу процесів. З появою сучасних технологій штучного інтелекту аграрний сектор отримав можливість перейти від традиційних методів до цифрових рішень, що здатні значно підвищити точність прогнозів, оптимізувати робочі процеси та зменшити людські помилки .

Протягом останніх десяти років аграрна галузь активно інтегрує алгоритми комп'ютерного зору, системи прогнозування, моделі машинного навчання та супутникові дані. Однак саме поява великих мовних моделей (LLM) стала переломним моментом у цифровізації агросектору. На відміну від класичних алгоритмів машинного навчання, які орієнтуються на вузькі, спеціалізовані задачі, мовні моделі здатні працювати з неструктурованими текстовими даними, аналізувати складні агрономічні інструкції, створювати нові рекомендації та здійснювати діалог із користувачем природною мовою. Це робить їх універсальним інструментом підтримки фермерів та агрономів у широкому спектрі виробничих процесів [7].

Традиційні сфери застосування ШІ у сільському господарстві включають прогнозування врожайності, оцінку погодних ризиків, комп'ютерний аналіз зображень рослин, автоматизацію техніки та оптимізацію використання ресурсів. Проте LLM значно розширили цей спектр, оскільки дозволяють об'єднати аналітичні та консультативні можливості в єдиній інтелектуальній системі.

Серед основних категорій застосування сучасних мовних моделей в агросекторі виділяють:

- аналітичну підтримку процесів, що включає аналіз агрономічних текстів, рекомендацій, нормативних актів та технологічних карт;
- інтерактивні консультативні системи, які надають фермеру підтримку у режимі діалогу, пояснюють складні процеси або пропонують альтернативні стратегії догляду за культурами;
- операційну підтримку, що охоплює ведення журналів, фіксацію даних, підготовку звітів та контроль за виконанням робіт.

Це демонструє, що мовні моделі здатні виконувати як когнітивні функції (аналіз та генерація знань), так і функції оптимізації робочих процесів [8].

У той час як традиційні моделі вимагають великих обсягів структурованих даних, LLM працюють із текстовими ресурсами та можуть адаптуватися до нових ситуацій на основі попередньої інформації. Це робить їх ідеальними для роботи з:

- агрономічними рекомендаціями;
- науковими статтями;
- щоденниками фермерів;
- локальними технологічними картами;
- інструкціями щодо використання добрив або засобів захисту.

Деякі світові агрокомпанії вже інтегрували LLM у свої продукти. Наприклад, системи типу FieldMind AI дозволяють агрономам отримувати відповіді на основі внутрішньої документації підприємства, а стартапи у США та Індії використовують локальні мовні моделі для підтримки фермерів у регіонах із низьким рівнем цифровізації. Крім того, дослідження AI у Європейському Союзі показали, що фермери, які використовують інтелектуальних асистентів, підвищують продуктивність на 12–18% завдяки оптимізованим рішенням щодо догляду за культурами та зменшенню людського фактору [9].

На відміну від класичних моделей машинного навчання, мовні моделі не вимагають спеціального навчання на конкретних наборах агрономічних даних. Їх універсальність дозволяє застосовувати їх одразу у широкому спектрі задач, включаючи:

- формування рекомендацій для догляду за культурами;
- виявлення проблем у технологічних процесах;
- адаптацію порад до конкретного поля або регіону;
- пояснення складних наукових концепцій у доступній формі.

Критично важливою перевагою є здатність моделей узагальнювати дані: мовні моделі можуть поєднувати знання з різних джерел – від інструкцій

виробників добрив до наукових праць з фітопатології. Завдяки цьому ШІ може працювати не лише як аналітичний інструмент, але і як експертна система.

Попри значні переваги, застосування LLM у сільському господарстві пов'язане з певними викликами. Один з них – доступність інтернету. Хмарні асистенти не можуть повноцінно функціонувати в полі через нестабільний зв'язок. Іншим викликом є приватність даних, оскільки фермери не бажають передавати конфіденційну інформацію хмарним сервісам [10].

Саме тому автономні локальні мовні моделі мають очевидні переваги та стають ключовим напрямом розвитку. Вони дозволяють фермеру працювати з великим масивом знань навіть у віддалених умовах, а також зберігають повний контроль над даними.

Крім того, деякі технічні аспекти агросектору – такі як розпізнавання зображень рослин, виявлення шкідників чи діагностування захворювань – можуть бути розширені шляхом поєднання LLM з комп'ютерним зором. Такі системи здатні автоматично аналізувати фотографії стану рослин, визначати хвороби з високою точністю та надавати рекомендації щодо лікування, що значно скорочує потребу в експертних консультаціях.

Таблиця 1.3 – Основні задачі агросектору, які ефективно вирішуються LLM

Категорія	Приклади задач
Аналітика	Аналіз текстових агрономічних рекомендацій, інструкцій та звітів
Прогнозування	Моделювання ризиків, попередження про можливі проблеми
Консультативна підтримка	Поради щодо удобрення, догляду, захисту рослин
Навчання	Пояснення технологій вирощування, інструктаж для працівників

Сучасні мовні моделі стали універсальним інструментом для аграрної галузі, оскільки поєднують можливості глибинного аналізу, генерації тексту,

консультаційних функцій та персоналізації. У поєднанні з мобільними технологіями вони створюють основу нового покоління рішень, здатних підтримувати фермерів у повсякденній роботі, підвищувати ефективність виробництва та покращувати якість управлінських рішень.

Особливої актуальності набувають локальні LLL, які забезпечують автономність, конфіденційність та швидку роботу без Інтернету, що робить їх ідеальними для польових умов. Саме такі моделі є технологічною платформою для створення персонального асистента фермера, який розробляється в межах даної магістерської роботи [11].

#### **1.4 Архітектура трансформерів та технології оптимізації мовних моделей для автономної роботи**

Архітектура трансформерів, представлена в 2017 році, стала основою сучасних мовних моделей та визначила подальшу еволюцію штучного інтелекту. На відміну від рекурентних мереж, які обробляли інформацію послідовно і поступово втрачали контекст, трансформери використовують механізм самоуваги (self-attention), що дозволяє кожному елементу послідовності взаємодіяти з усіма іншими незалежно від їхньої позиції. Такий підхід забезпечив моделям можливість глибоко аналізувати контекст, обробляти довгі текстові фрагменти та коректно відтворювати семантичні залежності.

У класичному варіанті трансформер складається з кодера та декодера, які повторюються у вигляді багат шарових блоків. Проте більшість сучасних мовних моделей використовує лише декодерну частину, оптимізовану для генерації тексту. Саме ця архітектура стала базою для моделей GPT-серії, LLaMA, Mistral та інших генеративних систем. Вона дозволяє не лише будувати текст, а й виконувати логічні міркування, аналіз аргументів, узагальнення та сталі діалогові сценарії [12].

Ефективність трансформерів зростає із масштабуванням моделі, проте великі розміри створюють значні виклики для практичного застосування. Моделі із сотнями мільярдів параметрів вимагають колосальних обчислювальних ресурсів і не можуть працювати локально. Саме тому у 2023–2025 роках розвиток спрямовується не лише на збільшення потужності, але й на оптимізацію моделей для автономної роботи на звичайних пристроях.

Ключові переваги архітектури трансформерів:

- можливість паралельної обробки даних і швидке навчання;
- гнучка робота з контекстом незалежно від довжини послідовності;
- висока здатність моделювати складні логічні та семантичні залежності.

Щоб перенести складні мовні моделі на мобільні пристрої, необхідно зменшити їхній розмір і підвищити ефективність. Найважливішою з таких технік стало квантування, яке дозволяє репрезентувати параметри моделі з меншою кількістю бітів. Моделі, представлені у форматах INT8 або INT4, вміщуються в оперативну пам'ять мобільних пристроїв без суттєвої втрати якості. Це стало ключовим фактором появи локальних LLM [13].

Іншим важливим напрямом оптимізації є дистиляція, завдяки якій невеликі моделі можуть наслідувати поведінку великих. Вони успадковують логічні патерни та стиль відповіді, але працюють набагато швидше. У поєднанні з оптимізаторами обчислювальних графів, такими як TensorRT-LLM або TVM, дистильовані моделі здатні відповідати менш ніж за секунду навіть на недорогих пристроях.

Окрему роль у розвитку автономних рішень відіграють апаратні прискорювачі – NPU. Сучасні смартфони, зокрема на базі Apple та Qualcomm, оснащені блоками, які виконують матричні операції на порядок ефективніше за CPU. Це дозволило перенести функціональність повноцінних LLM на мобільні девайси [14].

Основні методи оптимізації LLM:

- квантування параметрів до 8- або 4-бітних форматів;
- дистиляція моделей для зменшення обсягу без втрати можливостей;

– стиснення та прорідження шарів для скорочення обчислень.

Щоб адаптувати LLM до мобільних і вбудованих систем, були створені спеціалізовані програмні рушії. Розробка llama.cpp стала революційною, оскільки дала змогу запускати LLaMA-моделі навіть на ноутбуках із мінімальними ресурсами. Пізніше з'явився MLC LLM, що забезпечив компіляцію моделей для Android та iOS у вигляді оптимізованих пакетів. Для платформ NVIDIA активно використовується TensorRT-LLM, який суттєво прискорює інференс, зменшуючи затримку відповіді та енергоспоживання.

У контексті аграрних застосунків такий підхід є принципово важливим, оскільки фермери нерідко працюють у середовищах із нестабільним зв'язком. Оптимізовані LLM дозволяють забезпечити повноцінну роботу інтелектуального помічника навіть автономно, що робить застосунок більш надійним і корисним у реальних умовах.

Чому оптимізовані LLM важливі саме для агросектору:

- фермер працює переважно в полі, де інтернет нестабільний або відсутній;
- важлива конфіденційність даних про врожайність, технологічні карти та локальні процеси;
- мобільні пристрої – основні інструменти роботи працівників господарства [15].

Архітектура трансформерів стала базою сучасного ШІ, проте саме методи оптимізації зробили можливим їхнє використання в мобільних автономних середовищах. Завдяки квантуванню, дистиляції, оптимізації обчислювальних графів і наявності апаратних прискорювачів мовні моделі перестали бути суто хмарною технологією. Це відкриває можливість створення мобільних інтелектуальних систем, таких як персональний асистент фермера, який здатен працювати повністю офлайн, забезпечуючи доступ до знань і рекомендацій незалежно від умов.

## Висновки до розділу 1

У першому розділі було здійснено комплексний огляд сучасного стану штучного інтелекту, еволюції великих мовних моделей та специфіки їх застосування в аграрному секторі. Дослідження показало, що розвиток ШІ проходив кілька ключових етапів – від простих експертних систем до сучасних генеративних трансформерних моделей, які здатні працювати з великими обсягами інформації та створювати контекстно обґрунтовані відповіді. Поява архітектури Transformer стала переломним моментом у розвитку інтелектуальних систем, оскільки саме вона забезпечила можливість побудови високопродуктивних, масштабованих і гнучких мовних моделей.

Аналіз сучасних тенденцій продемонстрував значне зміщення пріоритетів від масштабування моделей до їх оптимізації та адаптації для автономної роботи на мобільних пристроях. Методи квантування, дистиляції, оптимізовані обчислювальні рушії та апаратні прискорювачі дали змогу запускати мовні моделі навіть у середовищах з обмеженими ресурсами. Завдяки цьому з'явилася можливість використання LLM у тих сферах, де раніше хмарні обчислення були єдиною альтернативою, але їх застосування було обмежене нестабільним зв'язком або вимогами конфіденційності.

Особливу увагу в рамках розділу було приділено локальним LLM, які працюють без підключення до Інтернету. Їхня автономність, безпечність, економічність та здатність до персоналізації є критично важливими характеристиками для фермерських господарств, що часто діють у віддалених районах без стабільної мережевої інфраструктури. Локальні моделі дозволяють фермерам отримувати якісні поради, аналізувати текстову інформацію, формувати технологічні рішення та отримувати оперативну аналітику навіть за відсутності доступу до мережі. Це робить їх одним із ключових інструментів цифрової трансформації сільського господарства.

Дослідження також показало, що інтеграція мовних моделей із сучасними мобільними технологіями відкриває нові можливості для

створення персональних інтелектуальних асистентів. Зокрема, у контексті агросектору поєднання локальної LLM із доступом до спеціалізованих знань, інструкцій, агрономічних рекомендацій та локальних баз даних створює середовище для формування повноцінного помічника фермера. Такий асистент здатен не лише відповідати на запитання, але й виконувати пояснювальні, консультативні та аналітичні функції, що сприяє підвищенню ефективності виробництва.

Отже, результати першого розділу підтверджують, що локальні великі мовні моделі є технологічно зрілим і доцільним рішенням для створення мобільного застосунку персонального асистента фермера. Проведений аналіз формує наукове та технічне підґрунтя для подальшого проектування системи, розробки її архітектури, визначення вимог до функціональності та обґрунтування вибору інструментів реалізації, що буде розглянуто у наступних розділах магістерської роботи.

## РОЗДІЛ 2

### АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАДАЧІ РОЗРОБКИ МОБІЛЬНОГО АСИСТЕНТА ФЕРМЕРА

#### **2.1 Аналіз предметної області та існуючих рішень у сфері цифрового фермерства**

Сільське господарство як галузь економіки характеризується високим рівнем складності управління та значною залежністю від зовнішніх факторів, зокрема кліматичних умов, сезонності, стану ґрунтів, біологічних процесів росту рослин і соціально-економічних умов. Прийняття рішень у фермерській діяльності часто відбувається в умовах невизначеності, браку інформації або обмеженого часу, що значно підвищує ризик помилок і втрат. У таких умовах цифрові технології стають ключовим інструментом підвищення ефективності аграрного виробництва, а цифрове фермерство розглядається як стратегічний напрям розвитку агросектору.

Цифрове фермерство (Digital Farming, Smart Agriculture) базується на інтеграції інформаційних технологій, сенсорних систем, аналізу даних та інтелектуальних алгоритмів у процесі планування, контролю та оптимізації сільськогосподарської діяльності. Основною метою такого підходу є підвищення продуктивності, зменшення витрат ресурсів, мінімізація екологічного впливу та забезпечення сталого розвитку аграрних господарств. У сучасних умовах цифрове фермерство перестає бути прерогативою великих агрохолдингів і поступово стає доступним для малих та середніх фермерських господарств завдяки розвитку мобільних технологій та програмних рішень.

Одним із ключових елементів цифровізації агросектору є управління інформацією. Фермеру необхідно працювати з великою кількістю даних: агрономічними рекомендаціями, календарями робіт, погодними прогнозами, нормативами внесення добрив, характеристиками сортів культур, результатами попередніх сезонів тощо. Традиційні методи обробки такої

інформації, зокрема паперові записи або фрагментовані електронні таблиці, є неефективними та не дозволяють оперативно приймати обґрунтовані рішення. Саме тому зростає роль програмних систем, здатних автоматизувати збір, аналіз і подання даних у зручному для користувача вигляді.

Упродовж останніх років у сфері цифрового фермерства сформувалося кілька основних напрямів розвитку програмних рішень. До них належать системи моніторингу стану полів на основі супутникових даних, платформи управління агровиробництвом (Farm Management Systems), мобільні застосунки для планування робіт, а також рішення на основі штучного інтелекту для прогнозування врожайності та діагностики захворювань рослин. Більшість із цих систем орієнтовані на обробку структурованих даних і виконання чітко визначених функцій, однак вони часто не враховують потребу фермера в інтуїтивній консультативній підтримці та поясненні прийнятих рекомендацій.

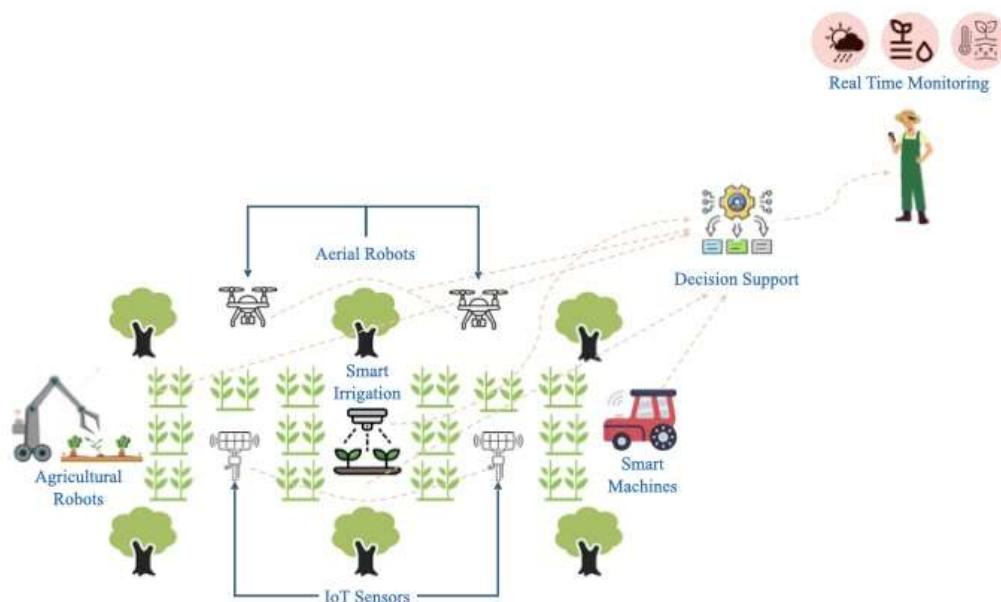


Рисунок 2.1 – Основні компоненти цифрового фермерства та їх взаємодія

На рисунку 2.1 представлено узагальнену структуру цифрового фермерства, яка включає сенсорні системи, програмні платформи, аналітичні

модулі та користувацькі інтерфейси. Як видно зі схеми, більшість сучасних рішень базуються на централізованій обробці даних і потребують постійного обміну інформацією між польовими пристроями та серверною інфраструктурою. Такий підхід забезпечує високу точність аналітики, однак створює залежність від інтернет-з'єднання та ускладнює використання систем безпосередньо в польових умовах. Це підтверджує доцільність розвитку автономних мобільних рішень, які можуть працювати незалежно від зовнішньої інфраструктури.

Особливу роль у сучасних цифрових рішеннях починають відігравати системи на основі штучного інтелекту. Спочатку ШІ в агросекторі застосовувався переважно для аналізу зображень (визначення бур'янів, шкідників, хвороб рослин), прогнозування врожайності та оптимізації використання ресурсів. Проте такі системи зазвичай мають вузьку спеціалізацію та вимагають значних обсягів навчальних даних. Крім того, їх використання часто обмежене складністю інтерфейсу або необхідністю спеціальної підготовки користувача.

Поява великих мовних моделей (LLM) відкрила нові можливості для цифрового фермерства, оскільки ці моделі здатні працювати з неструктурованою інформацією, узагальнювати знання з різних джерел та вести діалог природною мовою. Це дозволяє створювати програмні рішення нового типу – персональні інтелектуальні асистенти, які можуть виконувати роль цифрового консультанта фермера. На відміну від традиційних експертних систем, LLM не обмежуються жорстко заданими правилами, а здатні адаптуватися до контексту запиту, враховувати попередню історію взаємодії та формувати рекомендації у зрозумілій формі [16].

Разом з тим, аналіз існуючих цифрових рішень показує, що більшість сучасних аграрних платформ базуються на хмарній архітектурі. Це означає, що всі обчислення виконуються на віддалених серверах, а користувач взаємодіє з системою через інтернет-з'єднання. Такий підхід має низку переваг, зокрема високу масштабованість та можливість централізованого

оновлення програмного забезпечення. Проте для фермерів він створює суттєві обмеження: залежність від стабільності мережі, затримки у відповіді системи, а також ризики, пов'язані з передачею конфіденційних даних.

У багатьох сільських регіонах доступ до швидкісного інтернету залишається обмеженим або нестабільним, особливо безпосередньо в польових умовах. Це призводить до того, що навіть наявність сучасного мобільного пристрою не гарантує можливості використання хмарних сервісів у потрібний момент. У таких умовах фермер змушений покладатися на власний досвід або заздалегідь підготовлені матеріали, що знижує ефективність прийняття рішень. Саме ця проблема є однією з ключових передумов переходу до автономних програмних рішень, здатних працювати без постійного підключення до мережі.

Ще одним важливим аспектом предметної області є людський фактор. Багато фермерів не мають спеціальної ІТ-підготовки, тому складні програмні інтерфейси або надмірна кількість налаштувань стають бар'єром для впровадження цифрових технологій. У цьому контексті особливої актуальності набувають системи, які дозволяють взаємодіяти з програмним забезпеченням за допомогою природної мови. Мовні моделі створюють можливість реалізувати інтерфейс, у якому фермер може ставити запитання у звичній формі та отримувати зрозумілі пояснення без необхідності освоювати складні меню чи термінологію.

З урахуванням вищезазначеного можна виокремити ключові проблеми, які залишаються актуальними для сучасного цифрового фермерства:

- фрагментованість програмних рішень та відсутність єдиного інтелектуального середовища;
- залежність більшості систем від стабільного інтернет-з'єднання;
- складність інтерфейсів і недостатня орієнтація на кінцевого користувача;
- обмежені можливості персоналізації рекомендацій;
- недостатній рівень пояснюваності результатів роботи ШІ-систем.

Ці проблеми вказують на необхідність створення нового класу програмних рішень, які поєднують інтелектуальні можливості сучасних мовних моделей із автономністю та орієнтацією на реальні умови роботи фермера. Саме таким рішенням може стати мобільний додаток персонального асистента фермера на основі локальних великих мовних моделей. Такий асистент здатен інтегрувати знання з різних джерел, працювати без доступу до мережі, забезпечувати високий рівень конфіденційності та надавати рекомендації у зрозумілій формі [17].

Отже, аналіз предметної області цифрового фермерства свідчить про наявність суттєвого запиту на автономні, інтелектуальні та зручні у використанні програмні рішення. Сучасні тенденції розвитку штучного інтелекту та локальних мовних моделей створюють технічні передумови для реалізації таких систем, що обґрунтовує актуальність розробки мобільного персонального асистента фермера в межах даної магістерської роботи.

## **2.2 Порівняльний аналіз існуючих програмних рішень та AI-асистентів для агросектору**

У процесі цифровізації сільського господарства на ринку з'явилася значна кількість програмних рішень, спрямованих на автоматизацію окремих аспектів фермерської діяльності. Ці рішення відрізняються за функціональним призначенням, рівнем інтелектуалізації, архітектурними підходами та орієнтацією на різні категорії користувачів. Порівняльний аналіз існуючих програмних продуктів є необхідним етапом при проектуванні нового мобільного додатка, оскільки дозволяє визначити сильні та слабкі сторони аналогів, виявити незаповнені ніші та обґрунтувати доцільність запропонованого рішення.

Сучасні програмні продукти для фермерів умовно можна поділити на кілька груп залежно від їх функціонального призначення та рівня

використання штучного інтелекту. Найпоширенішими є системи управління фермерським господарством (Farm Management Systems, FMS), мобільні застосунки для планування робіт, платформи супутникового моніторингу, а також спеціалізовані AI-рішення для аналізу зображень і прогнозування врожайності.

Більшість таких систем орієнтовані на обробку структурованих даних і виконання чітко визначених операцій, наприклад облік ресурсів або формування звітів. Водночас вони не завжди забезпечують інтерактивну консультаційну підтримку та не враховують потребу користувача у поясненні результатів, що є важливим для прийняття рішень у складних агрономічних умовах [18].

Одним із найбільш поширених класів програмних продуктів є платформи управління господарством, такі як AgroOffice, Cropio, FarmLogs та Granular. Вони дозволяють вести облік посівів, контролювати витрати, планувати роботи та аналізувати результати сезонів. Основною перевагою таких систем є комплексність і можливість централізованого управління даними. Проте їх використання часто потребує стабільного інтернет-з'єднання, а інтерфейс орієнтований переважно на підготовлених користувачів або менеджерів, а не безпосередньо на фермера в польових умовах.



Рисунок 2.2 – Класифікація програмних рішень для аграрного сектору

Іншою категорією є мобільні застосунки довідкового та консультаційного характеру. Вони зазвичай містять базу знань з агрономії, календарі робіт або рекомендації щодо догляду за культурами. Такі застосунки мають низький поріг входу, але їх функціональність обмежена статичним контентом, який не адаптується до конкретних умов господарства. Крім того, відсутність інтелектуального ядра не дозволяє їм аналізувати контекст або формувати персоналізовані рекомендації.

Окрему групу становлять рішення на основі комп'ютерного зору, наприклад системи діагностики хвороб рослин за фотографіями. Вони демонструють високу точність у вузьких задачах, проте не забезпечують комплексного підходу до управління господарством. Такі системи зазвичай виконують лише одну функцію і не інтегруються з іншими інформаційними джерелами [19].

З появою великих мовних моделей почали з'являтися перші AI-асистенти для аграрної галузі. Більшість з них реалізовані у вигляді хмарних сервісів, які використовують потужні серверні моделі для обробки запитів користувачів. Такі асистенти можуть відповідати на запитання, надавати загальні рекомендації та допомагати з плануванням робіт. Їх головною перевагою є висока якість відповідей і здатність працювати з природною мовою.

Проте хмарні AI-асистенти мають низку суттєвих обмежень. По-перше, вони повністю залежать від інтернет-з'єднання, що робить їх малоприсадними для використання безпосередньо в полі. По-друге, використання таких сервісів часто пов'язане з передачею конфіденційних даних на сторонні сервери, що викликає занепокоєння у фермерів. По-третє, більшість хмарних рішень є платними та мають обмеження на кількість запитів.

У зв'язку з цим зростає інтерес до локальних AI-рішень, які можуть виконувати інференс безпосередньо на пристрої користувача. Однак на момент проведення дослідження кількість таких продуктів на аграрному

ринку є обмеженою, а їх функціональність часто знаходиться на стадії прототипів або експериментальних розробок.

Для об'єктивного порівняльного аналізу існуючих рішень доцільно використовувати низку критеріїв, які відображають потреби кінцевого користувача – фермера. До таких критеріїв належать:

- автономність роботи та залежність від інтернет-з'єднання;
- рівень використання штучного інтелекту;
- можливість персоналізації рекомендацій;
- зручність та інтуїтивність інтерфейсу;
- конфіденційність та безпека даних;
- доступність для малих і середніх господарств [20].

Застосування цих критеріїв дозволяє не лише порівняти існуючі продукти між собою, але й визначити, наскільки вони відповідають реальним умовам роботи у сільській місцевості.

Таблиця 2.1 – Порівняльний аналіз існуючих рішень для агросектору

Критерій	FMS-платформи	Довідкові застосунки	Хмарні AI-асистенти	Запропоноване рішення
Робота без інтернету	Обмежена	Часткова	Ні	Так
Інтелектуальна підтримка	Низька	Низька	Висока	Висока
Персоналізація	Середня	Відсутня	Обмежена	Висока
Конфіденційність даних	Середня	Висока	Низька	Висока
Зручність у полі	Обмежена	Середня	Низька	Висока

Дана таблиця демонструє, що більшість існуючих рішень не забезпечують одночасно автономність, інтелектуальність та орієнтацію на польові умови.

Проведений аналіз показує, що сучасні програмні продукти для агросектору здебільшого вирішують окремі задачі, але не формують цілісного інтелектуального середовища для фермера. Відсутність автономності, складність інтерфейсів та обмежені можливості пояснення рекомендацій знижують ефективність використання таких систем у реальних умовах. Особливо відчутною є нестача мобільних рішень, які поєднують можливості мовних моделей із локальним виконанням та високим рівнем конфіденційності.

У цьому контексті мобільний асистент фермера на основі локальних великих мовних моделей може заповнити існуючу нішу. Він поєднує функції довідкової системи, консультаційного сервісу та аналітичного інструменту, зберігаючи при цьому автономність і орієнтацію на кінцевого користувача.

Порівняльний аналіз існуючих програмних рішень показав, що, незважаючи на значну кількість продуктів на ринку, більшість із них не відповідають повною мірою потребам фермера у польових умовах. Основними недоліками є залежність від інтернету, обмежена персоналізація та відсутність інтерактивної інтелектуальної підтримки. Це підтверджує актуальність розробки мобільного додатка персонального асистента фермера на основі локальних великих мовних моделей, який здатен забезпечити автономну, безпечну та зручну підтримку прийняття рішень [21].

Узагальнюючи результати порівняльного аналізу, можна зробити висновок, що сучасний ринок аграрних програмних рішень перебуває на етапі переходу від класичних інформаційних систем до інтелектуальних помічників. Проте більшість наявних продуктів зосереджені або на управлінні даними, або на окремих аналітичних функціях і не забезпечують комплексної підтримки фермера в реальному часі. У цьому контексті використання локальних великих мовних моделей у мобільному застосунку створює передумови для формування нового класу рішень, які поєднують автономність, інтелектуальність та орієнтацію на кінцевого користувача.

## **2.3 Формування вимог до мобільного додатка персонального асистента фермера**

Формування вимог до програмного продукту неможливе без чіткого розуміння цільової аудиторії та умов, у яких буде використовуватися мобільний додаток. У контексті даної магістерської роботи основною цільовою аудиторією є фермери та працівники аграрних господарств, діяльність яких пов'язана з безпосередньою роботою в полі, на фермі або у виробничих приміщеннях. Особливістю цієї групи користувачів є поєднання практичного досвіду з обмеженим доступом до сучасних інформаційних ресурсів у реальному часі.

На відміну від менеджерів агрохолдингів або аналітиків, які працюють у офісних умовах, фермери часто приймають рішення безпосередньо на місці виконання робіт. Це може бути вибір способу обробітку ґрунту, оцінка стану рослин, визначення необхідності внесення добрив або засобів захисту, а також планування подальших дій залежно від погодних умов. У таких ситуаціях доступ до інтернету є нестабільним або повністю відсутнім, що значно ускладнює використання хмарних сервісів та онлайн-платформ [22].

Цільова аудиторія мобільного асистента фермера є неоднорідною та включає кілька підгруп користувачів. По-перше, це власники малих і середніх фермерських господарств, які самостійно виконують більшість управлінських і виробничих функцій. По-друге, це агрономи та технічні спеціалісти, відповідальні за окремі ділянки або культури. По-третє, це працівники господарств, які можуть використовувати додаток як довідковий або навчальний інструмент. Усі ці категорії користувачів мають спільну потребу – швидко отримувати зрозумілу та практично корисну інформацію без необхідності глибокого занурення в технічні деталі.

Важливою характеристикою цільової аудиторії є різний рівень цифрової грамотності. Частина фермерів активно користується смартфонами та мобільними застосунками, тоді як інші мають мінімальний досвід роботи з

цифровими сервісами. Тому мобільний асистент має бути спроектований таким чином, щоб його використання не вимагало спеціальної підготовки. Інтерфейс повинен бути інтуїтивно зрозумілим, а взаємодія з системою – максимально наближеною до звичайного спілкування.

На основі аналізу діяльності фермерів можна визначити типові сценарії використання мобільного додатка персонального асистента. Найбільш поширеним сценарієм є ситуація, коли користувач стикається з конкретною проблемою або питанням і потребує оперативної консультації. Наприклад, фермер може запитати про можливі причини зміни кольору листя рослин, оптимальний час внесення добрив або допустимі норми використання певного препарату. У такому випадку асистент повинен надати коротку, зрозумілу відповідь з поясненням причин і можливих наслідків.

Іншим важливим сценарієм є планування робіт. Користувач може звертатися до асистента для формування календаря агротехнічних заходів, уточнення послідовності операцій або перевірки відповідності запланованих дій погодним умовам. Також можливі сценарії навчального характеру, коли додаток використовується для отримання довідкової інформації, пояснення агрономічних термінів або ознайомлення з новими технологіями вирощування культур.

Таким чином, аналіз цільової аудиторії та сценаріїв використання показує, що мобільний асистент фермера повинен орієнтуватися на практичні потреби користувачів, працювати у режимі реального часу та забезпечувати доступ до знань незалежно від умов зовнішнього середовища. Це визначає подальші вимоги до функціональності, автономності та зручності використання програмного продукту.

Функціональні вимоги визначають набір можливостей, які мобільний додаток повинен надавати користувачу для ефективного виконання поставлених завдань. У випадку персонального асистента фермера ці вимоги формуються з урахуванням специфіки аграрної діяльності, умов використання додатка та обмежень мобільних пристроїв. Основна мета функціональних

вимог полягає у забезпеченні комплексної консультаційної та інформаційної підтримки фермера в автономному режимі.

Ключовою функцією мобільного асистента є можливість взаємодії з користувачем у формі діалогу природною мовою. Користувач повинен мати змогу ставити запитання у довільній формі, а система – коректно інтерпретувати запит, враховуючи контекст та попередню історію взаємодії. Відповіді асистента мають бути зрозумілими, структурованими та орієнтованими на практичне застосування, без надмірного використання спеціалізованої термінології.

Наступною важливою функціональною вимогою є автономна робота додатка без підключення до інтернету. Це означає, що всі основні обчислення, включаючи виконання мовної моделі, аналіз запитів та формування відповідей, повинні здійснюватися безпосередньо на мобільному пристрої користувача. Такий підхід забезпечує стабільність роботи асистента в польових умовах і підвищує рівень конфіденційності даних.

Мобільний асистент також повинен забезпечувати доступ до агрономічних знань та рекомендацій, адаптованих до конкретних умов використання. Це передбачає можливість використання локальної бази знань, яка може містити інформацію про культури, типи ґрунтів, технологічні операції, норми внесення добрив та інші важливі аспекти фермерської діяльності. У перспективі така база знань може розширюватися або оновлюватися користувачем [23].

Важливою функцією є можливість пояснення рекомендацій. Асистент не повинен обмежуватися лише наданням готових порад, а має пояснювати причини запропонованих рішень, можливі альтернативи та ризики. Це сприяє підвищенню довіри користувача до системи та дозволяє приймати більш обґрунтовані рішення.

Окремо слід виділити функцію збереження історії взаємодії з асистентом. Збереження попередніх запитів і відповідей дозволяє користувачу повертатися до важливої інформації, а також створює основу для подальшої

персоналізації роботи системи. При цьому всі дані мають зберігатися локально на пристрої користувача.

Нефункціональні вимоги визначають якісні характеристики програмного продукту та значною мірою впливають на зручність, надійність і ефективність його використання. Для мобільного додатка персонального асистента фермера ці вимоги є не менш важливими, ніж функціональні, оскільки система повинна працювати стабільно в умовах обмежених ресурсів і підвищених зовнішніх навантажень.

Однією з ключових нефункціональних вимог є продуктивність. Асистент повинен формувати відповіді за мінімальний час, оскільки затримки у взаємодії негативно впливають на користувацький досвід і можуть ускладнювати прийняття рішень у польових умовах. Час відповіді системи не повинен перевищувати декількох секунд навіть на пристроях середнього класу. Це накладає обмеження на розмір мовної моделі, формат її зберігання та спосіб виконання інференсу.

Не менш важливою є вимога до автономності та енергоефективності. Мобільний додаток має працювати тривалий час без суттєвого впливу на заряд акумулятора пристрою. Оскільки виконання великих мовних моделей є ресурсоемним процесом, система повинна використовувати оптимізовані алгоритми, апаратні прискорювачі та режими роботи, які мінімізують енергоспоживання. У контексті польових робіт, де можливість заряджання пристрою є обмеженою, ця вимога набуває особливого значення [24].

Високий рівень надійності та стабільності є ще однією важливою нефункціональною характеристикою. Додаток повинен коректно працювати за різних умов, включаючи тривале використання, відсутність мережевого з'єднання та обмеження оперативної пам'яті. Аварійні завершення роботи або втрата даних є неприпустимими, оскільки можуть призвести до втрати важливої інформації або зниження довіри користувача до системи.

Окрему увагу слід приділити вимогам до безпеки та конфіденційності даних. Оскільки асистент може обробляти інформацію про діяльність

господарства, технологічні процеси та інші чутливі дані, всі вони повинні зберігатися та оброблятися локально на пристрої користувача. Додаток не повинен передавати дані стороннім сервісам без явної згоди користувача. Також доцільно передбачити захист доступу до додатка за допомогою стандартних механізмів автентифікації мобільної операційної системи.

До основних нефункціональних вимог мобільного асистента фермера можна віднести:

- швидкий час відповіді системи;
- низьке енергоспоживання;
- стабільність роботи в автономному режимі;
- захист та локальне зберігання даних;
- масштабованість та можливість подальшого розширення функціональності [25].

Таким чином, нефункціональні вимоги визначають технічні межі та якісні характеристики системи, які мають бути враховані на етапі проєктування архітектури та вибору інструментів реалізації.

Інтерфейс користувача є одним із ключових факторів успіху мобільного додатка, особливо у випадку систем, орієнтованих на користувачів із різним рівнем цифрової підготовки. Для персонального асистента фермера інтерфейс повинен бути максимально простим, зрозумілим і адаптованим до реальних умов використання, включаючи роботу на відкритому повітрі, у русі або в умовах яскравого освітлення.

Основним принципом побудови інтерфейсу є мінімалізм і інтуїтивність. Користувач повинен мати змогу швидко розпочати взаємодію з асистентом без попереднього навчання. Основна функціональність має бути доступною з головного екрану, а кількість додаткових меню та налаштувань – мінімальною. Це особливо важливо для ситуацій, коли фермеру необхідно оперативно отримати відповідь, не відволікаючись від основної роботи.

Важливою вимогою є підтримка текстового введення природною мовою, яке повинно бути реалізоване з урахуванням особливостей мобільних

пристроїв. Поле введення має бути зручним для використання однією рукою, а розмір шрифту – достатнім для комфортного читання. Доцільно також передбачити можливість масштабування тексту для користувачів із порушеннями зору [26].

Відповіді асистента повинні бути структурованими та легко сприйматися. Довгі тексти доцільно розбивати на логічні абзаци або короткі рекомендації, що полегшує їх використання безпосередньо в польових умовах. При цьому важливо уникати надмірного використання спеціалізованої термінології або, за необхідності, надавати пояснення до складних понять.

Ще одним аспектом зручності використання є адаптивність інтерфейсу. Додаток повинен коректно працювати на різних розмірах екранів і підтримувати як вертикальну, так і горизонтальну орієнтацію. Це дозволяє використовувати асистента на різних типах пристроїв – від смартфонів до планшетів.

Отже, вимоги до інтерфейсу користувача мають бути спрямовані на створення простого, зрозумілого та ефективного засобу взаємодії, який не створює додаткових труднощів і сприяє широкому впровадженню мобільного додатка серед фермерів.

Проектування мобільного асистента фермера здійснюється в умовах певних технічних і організаційних обмежень, які необхідно враховувати для реалістичної оцінки можливостей системи. Одним із основних обмежень є обчислювальна потужність мобільних пристроїв. Незважаючи на значний прогрес у розвитку мобільного обладнання, ресурси смартфонів і планшетів залишаються обмеженими порівняно з серверними системами. Це накладає обмеження на розмір і складність мовної моделі, яка може бути використана у додатку [27].

Іншим важливим обмеженням є обсяг оперативної пам'яті, доступної для виконання мовної моделі. Для забезпечення стабільної роботи додатка необхідно використовувати оптимізовані та квантизовані моделі, що може впливати на точність і деталізацію відповідей. У зв'язку з цим передбачається,

що мобільний асистент буде орієнтований на вирішення практичних завдань і надання рекомендацій загального характеру, а не на виконання складних наукових розрахунків.

Серед припущень, прийнятих у межах даної роботи, слід зазначити, що користувач має базові навички роботи зі смартфоном та може взаємодіяти з додатком через текстовий інтерфейс. Також передбачається, що початкове встановлення додатка та завантаження мовної моделі може вимагати одноразового підключення до інтернету, після чого система зможе працювати автономно.

До основних обмежень і припущень проєкту можна віднести:

- обмежені ресурси мобільних пристроїв;
- необхідність використання компактних локальних мовних моделей;
- орієнтацію на автономну роботу без постійного доступу до мережі;
- припущення щодо базового рівня цифрової грамотності користувача.

Урахування цих обмежень та припущень дозволяє сформулювати реалістичні вимоги до системи та створити програмний продукт, який відповідатиме реальним умовам експлуатації.

У цьому підрозділі було сформовано комплекс вимог до мобільного додатка персонального асистента фермера, що охоплює цільову аудиторію, функціональні та нефункціональні характеристики, вимоги до інтерфейсу користувача, а також обмеження та припущення проєкту. Отримані результати є основою для постановки задачі розробки та вибору концепції програмного рішення, які будуть розглянуті у наступному підрозділі.

## **2.4 Постановка задачі розроблення та вибір концепції програмного рішення**

На основі аналізу предметної області, існуючих програмних рішень та сформованих вимог до мобільного додатка персонального асистента фермера

виникає необхідність чіткої постановки задачі розробки програмного продукту. Формалізація задачі дозволяє визначити межі проєкту, окреслити його функціональну спрямованість, а також обґрунтувати вибір архітектурних і технологічних рішень. У межах даної магістерської роботи постановка задачі спрямована на створення мобільного застосунку, який поєднує інтелектуальні можливості локальних великих мовних моделей з автономністю та орієнтацією на реальні умови фермерської діяльності.

Основною проблемою, що потребує вирішення, є відсутність універсального мобільного інструмента, який забезпечує фермеру інтелектуальну консультаційну підтримку без залежності від інтернет-з'єднання. Більшість наявних рішень або не мають інтелектуального компонента, або базуються на хмарних сервісах, що обмежує їх використання у польових умовах. Крім того, сучасні цифрові системи часто не враховують потребу користувача у поясненні рекомендацій та адаптації інформації до конкретних умов господарства [28].

З урахуванням цього, задача розробки мобільного асистента фермера полягає у створенні програмного рішення, здатного виконувати роль персонального цифрового консультанта, який працює автономно, використовує локальну велику мовну модель та забезпечує зручну взаємодію з користувачем. Асистент повинен не лише відповідати на запитання, але й допомагати у прийнятті рішень, пояснювати свої рекомендації та зберігати контекст взаємодії.

З технічної точки зору задача розробки полягає у створенні мобільного застосунку, що включає інтерфейс користувача, інтелектуальне ядро на основі локальної мовної моделі та допоміжні модулі для зберігання даних і управління взаємодією. Додаток повинен бути здатним працювати на поширених мобільних платформах та використовувати ресурси пристрою максимально ефективно.

У загальному вигляді задачу розробки можна сформулювати так: розробити мобільний додаток персонального асистента фермера, який

забезпечує автономну інтелектуальну підтримку користувача шляхом використання локальної великої мовної моделі та відповідає сформованим функціональним і нефункціональним вимогам.

Для досягнення поставленої мети необхідно вирішити низку підзадач, пов'язаних із проєктуванням архітектури системи, вибором технологій реалізації, оптимізацією мовної моделі та розробкою зручного інтерфейсу користувача.

У межах поставленої задачі доцільно виділити такі ключові підзадачі:

- аналіз можливостей локальних великих мовних моделей для використання на мобільних пристроях;
- вибір архітектури програмного рішення з урахуванням автономної роботи;
- визначення способу інтеграції мовної моделі у мобільний додаток;
- проєктування інтерфейсу користувача, орієнтованого на фермерів;
- забезпечення зберігання даних і конфіденційності інформації;
- оптимізація продуктивності та енергоспоживання.

Реалізація цих підзадач створює основу для побудови працездатного прототипу мобільного асистента, який може бути використаний у реальних умовах [29].

Концепція програмного рішення визначає загальний підхід до побудови системи, її архітектурну модель та принципи взаємодії між компонентами. У межах даної роботи було обрано концепцію локального автономного інтелектуального асистента, що передбачає виконання всіх основних обчислень безпосередньо на пристрої користувача.

Ключовим елементом концепції є використання локальної великої мовної моделі, оптимізованої для роботи в умовах обмежених ресурсів. Такий підхід дозволяє забезпечити незалежність від мережі, мінімізувати затримки у відповіді системи та зберегти повний контроль над даними користувача. На відміну від хмарних рішень, локальна модель не потребує постійного доступу

до серверної інфраструктури, що є критично важливим для фермерів, які працюють у віддалених регіонах.

Архітектурно програмне рішення передбачає модульну структуру, у якій кожен компонент виконує чітко визначені функції. Інтерфейс користувача відповідає за взаємодію з фермером, інтелектуальний модуль – за обробку запитів і генерацію відповідей, а допоміжні модулі – за зберігання даних, управління налаштуваннями та оптимізацію ресурсів. Такий підхід спрощує подальший розвиток системи та дозволяє адаптувати її до нових вимог.

Під час вибору концепції програмного рішення було розглянуто кілька альтернативних підходів. Перший підхід передбачає використання виключно хмарної мовної моделі з мінімальною логікою на стороні клієнта. Другий підхід базується на гібридній архітектурі, де частина обчислень виконується локально, а складні запити обробляються у хмарі. Третій підхід – повністю локальне виконання мовної моделі на мобільному пристрої.

Порівняльний аналіз показав, що хмарна архітектура забезпечує високу якість відповідей, але не відповідає вимогам автономності та конфіденційності. Гібридний підхід частково вирішує проблему автономності, однак ускладнює архітектуру системи та зберігає залежність від мережі. Повністю локальний підхід, незважаючи на обмеження щодо розміру моделі, є найбільш відповідним до умов використання та вимог цільової аудиторії.

У результаті реалізації поставленої задачі планується створення мобільного додатка, який забезпечує:

- автономну інтелектуальну підтримку фермера;
- зручну взаємодію з системою через природну мову;
- можливість використання у польових умовах без доступу до інтернету;
- збереження конфіденційності даних користувача;
- основу для подальшого розширення функціональності.

Очікується, що розроблений прототип продемонструє практичну доцільність використання локальних великих мовних моделей у мобільних

застосунках для агросектору та підтвердить ефективність обраної концепції [30].

## **Висновки до розділу 2**

У другому розділі магістерської роботи було здійснено детальний аналіз предметної області цифрового фермерства та розглянуто сучасні програмні рішення, що застосовуються для підтримки аграрної діяльності. Проведений аналіз показав, що агросектор характеризується високою складністю управлінських процесів, значною залежністю від зовнішніх факторів і потребою у своєчасному прийнятті обґрунтованих рішень безпосередньо в польових умовах. Це зумовлює зростаючий попит на мобільні цифрові інструменти, здатні забезпечити оперативну інформаційну та консультаційну підтримку фермерів.

Було визначено ключові особливості предметної області та окреслено основні напрями розвитку цифрового фермерства. Аналіз існуючих підходів до цифровізації аграрної діяльності показав, що більшість сучасних систем орієнтовані на централізовану обробку даних та використання хмарних сервісів. Такий підхід, хоча й забезпечує високу аналітичну потужність, має суттєві обмеження у вигляді залежності від інтернет-з'єднання, складності інтерфейсів і недостатньої адаптації до реальних умов роботи фермерів у польовому середовищі.

Порівняльний аналіз програмних рішень та AI-асистентів, дозволив виявити ключові недоліки наявних продуктів і незаповнені ніші на ринку аграрних технологій. Зокрема, було встановлено, що більшість існуючих рішень не поєднують у собі автономність, інтелектуальну підтримку та високий рівень персоналізації. Це підтверджує актуальність створення мобільного застосунку нового типу, який використовує локальні великі мовні моделі та орієнтований на кінцевого користувача – фермера.

Було сформовано комплекс вимог до мобільного додатка персонального асистента фермера, що охоплює визначення цільової аудиторії, функціональні та нефункціональні характеристики системи, вимоги до інтерфейсу користувача, а також технічні обмеження і припущення проєкту. Сформульовані вимоги відображають реальні умови експлуатації мобільного застосунку та забезпечують основу для створення практично корисного і зручного інструмента підтримки прийняття рішень.

Здійснено постановку задачі розробки та обґрунтовано вибір концепції програмного рішення. Було доведено доцільність використання концепції локального автономного інтелектуального асистента, яка дозволяє забезпечити незалежність від мережі, високий рівень конфіденційності та швидкий відгук системи. Обрана концепція відповідає сучасним тенденціям розвитку штучного інтелекту та специфіці аграрної діяльності.

Отже, результати другого розділу підтверджують наукову та практичну обґрунтованість розробки мобільного додатка персонального асистента фермера на основі локальних великих мовних моделей. Отримані висновки створюють методологічне та технічне підґрунтя для наступного етапу дослідження – проєктування архітектури системи, вибору програмних інструментів і реалізації мобільного застосунку, що буде розглянуто у третьому розділі магістерської роботи.

## РОЗДІЛ 3

### ПРОЄКТУВАННЯ ТА РЕАЛІЗАЦІЯ МОБІЛЬНОГО ДОДАТКА ПЕРСОНАЛЬНОГО АСИСТЕНТА ФЕРМЕРА

#### **3.1 Загальна архітектура мобільного додатка персонального асистента фермера**

Проектування загальної архітектури мобільного додатка є ключовим етапом у процесі розробки програмного забезпечення, оскільки саме на цьому етапі визначаються основні компоненти системи, принципи їх взаємодії та способи реалізації функціональних і нефункціональних вимог. У випадку мобільного додатка персонального асистента фермера архітектура повинна забезпечувати автономність роботи, ефективне використання ресурсів мобільного пристрою, зручну взаємодію з користувачем та можливість подальшого розширення функціональності.

Розроблюваний додаток розглядається як автономна інтелектуальна система, що працює без постійного підключення до інтернету та використовує локальну велику мовну модель для обробки запитів користувача. Такий підхід зумовлює необхідність ретельного проектування архітектури, оскільки всі обчислювальні процеси виконуються безпосередньо на мобільному пристрої, який має обмежені ресурси порівняно з серверними системами.

Загальна архітектура мобільного асистента фермера базується на модульному підході, що передбачає поділ системи на логічно незалежні компоненти з чітко визначеними зонами відповідальності. Модульність дозволяє спростити розробку, тестування та супровід додатка, а також створює передумови для масштабування та адаптації системи до нових вимог у майбутньому [31].

Для реалізації мобільного асистента фермера було обрано клієнт-орієнтовану локальну архітектуру, у якій усі ключові функції зосереджені на стороні клієнта – мобільного пристрою користувача. На відміну від класичної

клієнт-серверної архітектури, де більшість обчислень виконується на сервері, запропонований підхід передбачає локальне виконання мовної моделі, обробку даних та формування відповідей без звернення до зовнішніх сервісів.

Такий архітектурний підхід обґрунтований специфікою аграрної діяльності та умовами використання додатка. Фермер часто працює у віддалених районах, де відсутній стабільний доступ до мережі, тому система повинна залишатися працездатною за будь-яких обставин. Крім того, локальна архітектура забезпечує підвищений рівень конфіденційності, оскільки всі дані користувача зберігаються і обробляються в межах одного пристрою.

З точки зору архітектурного стилю, система поєднує елементи шарової (layered) архітектури та компонентно-орієнтованого підходу. Логіка додатка розділена на окремі рівні, кожен з яких виконує власні функції та взаємодіє з іншими рівнями через чітко визначені інтерфейси. Це зменшує зв'язність компонентів і підвищує гнучкість системи.

У загальному вигляді архітектура мобільного додатка персонального асистента фермера включає такі ключові компоненти: користувацький інтерфейс, модуль управління взаємодією, інтелектуальне ядро на основі локальної мовної моделі, модуль обробки даних та допоміжні сервісні компоненти. Кожен із цих компонентів виконує визначену роль у забезпеченні функціональності системи.

Користувацький інтерфейс відповідає за взаємодію з фермером і є основною точкою доступу до функцій додатка. Саме через інтерфейс користувач вводить текстові запити, переглядає відповіді асистента та взаємодіє з додатковими можливостями системи. Інтерфейс повинен бути простим, зрозумілим і адаптованим до використання в польових умовах.

Модуль управління взаємодією виконує роль посередника між інтерфейсом користувача та інтелектуальним ядром. Він відповідає за попередню обробку запитів, управління контекстом діалогу, передачу даних до мовної моделі та обробку отриманих результатів. Наявність окремого

модуля управління взаємодією дозволяє відокремити логіку діалогу від реалізації мовної моделі.

Інтелектуальне ядро системи базується на локальній великій мовній моделі, оптимізованій для роботи на мобільному пристрої. Цей компонент є ключовим з точки зору функціональності додатка, оскільки саме він забезпечує інтерпретацію запитів користувача, генерацію відповідей і пояснення рекомендацій. Мовна модель працює в автономному режимі та не потребує доступу до зовнішніх серверів.

Модуль обробки та зберігання даних відповідає за локальне збереження історії взаємодії, налаштувань користувача та допоміжної інформації. Усі дані зберігаються у захищеному локальному сховищі, що забезпечує конфіденційність і цілісність інформації. Такий підхід дозволяє реалізувати персоналізацію роботи асистента без ризику витоку даних.

Допоміжні сервісні компоненти включають механізми управління ресурсами, логування, обробки помилок і контролю продуктивності. Вони не взаємодіють безпосередньо з користувачем, але забезпечують стабільну та ефективну роботу додатка в цілому.

Взаємодія між компонентами архітектури побудована за принципом мінімальної зв'язності та чіткої ієрархії. Користувацький інтерфейс передає введений запит до модуля управління взаємодією, який виконує первинну обробку тексту та формує запит до інтелектуального ядра. Після виконання мовної моделі отриманий результат передається назад до модуля управління взаємодією, де він може бути додатково оброблений або адаптований до формату відображення [32].

Такий підхід дозволяє ізолювати мовну модель від безпосередньої взаємодії з інтерфейсом користувача, що спрощує заміну або оновлення моделі у майбутньому. Крім того, чітке розмежування відповідальностей між компонентами підвищує надійність системи та спрощує тестування окремих модулів.

Усі операції, пов'язані зі зберіганням даних, виконуються через модуль локального сховища, який надає стандартизований інтерфейс доступу до інформації. Це дозволяє уникнути дублювання логіки зберігання даних у різних частинах системи та забезпечує цілісність інформації.

Запропонована загальна архітектура мобільного додатка повністю відповідає вимогам, сформульованим у підрозділі 2.3. Автономна локальна архітектура забезпечує незалежність від інтернет-з'єднання та високий рівень конфіденційності даних. Модульність і компонентна структура дозволяють ефективно використовувати обмежені ресурси мобільного пристрою та забезпечують масштабованість системи.

Архітектурний підхід також враховує вимоги до зручності використання та продуктивності. Розділення логіки взаємодії та інтелектуального ядра дозволяє оптимізувати обчислювальні процеси та зменшити затримки у відповіді системи. Це є критично важливим для використання додатка у польових умовах, де швидкість і стабільність роботи мають першочергове значення [33].

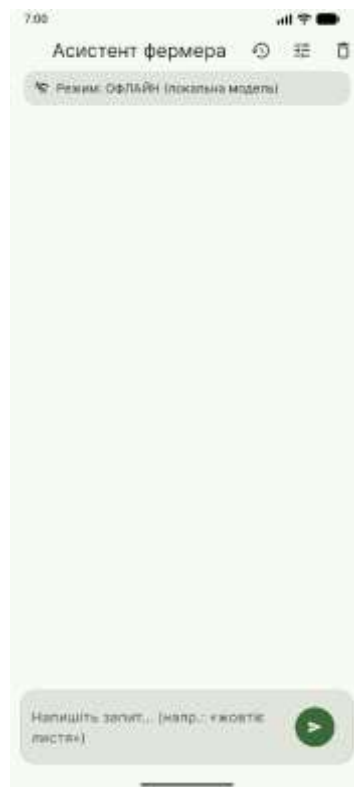


Рисунок 3.1 – Приклад інтерфейсу мобільного додатка

У підрозділі було розроблено загальну архітектуру мобільного додатка персонального асистента фермера, яка базується на принципах автономності, модульності та ефективного використання ресурсів мобільного пристрою. Запропонована архітектура створює основу для подальшого детального проєктування програмної структури системи, опису взаємодії компонентів і реалізації окремих функціональних модулів, що буде розглянуто у наступних підрозділах третього розділу.

### **3.2 Проєктування програмної архітектури та взаємодії компонентів системи**

Проєктування програмної архітектури є критично важливим етапом розробки мобільного додатка, оскільки саме на цьому рівні визначаються внутрішня структура системи, логіка взаємодії між її компонентами та способи реалізації функціональних і нефункціональних вимог. Для мобільного додатка персонального асистента фермера архітектура повинна забезпечувати не лише коректну реалізацію бізнес-логіки, але й ефективну інтеграцію локальної великої мовної моделі, стабільну роботу в автономному режимі та мінімальне навантаження на ресурси мобільного пристрою.

Запропонована програмна архітектура ґрунтується на принципах модульності, слабкої зв'язаності компонентів і чіткого розмежування відповідальностей. Це дозволяє спростити процес розробки, полегшити тестування окремих модулів і забезпечити можливість подальшого розширення функціональності без суттєвих змін у базовій структурі системи.

Програмна архітектура мобільного асистента фермера реалізується у вигляді багаторівневої структури, у якій кожен рівень відповідає за виконання певного класу задач. Такий підхід дозволяє ізолювати логіку взаємодії з користувачем від обчислювальних процесів та роботи з даними, що є особливо важливим при використанні ресурсомістких мовних моделей [34].

Модуль інтерфейсу користувача є ключовим елементом взаємодії між фермером і системою. Він реалізує екран діалогу з асистентом, відображає історію повідомлень, дозволяє вводити текстові запити та переглядати відповіді системи. Архітектурно інтерфейс користувача не містить складної бізнес-логіки, а лише передає події до відповідних контролерів або моделей стану.

Такий підхід дозволяє мінімізувати залежність інтерфейсу від внутрішньої реалізації мовної моделі та забезпечує можливість зміни або оновлення UI без втручання в інші частини системи. Крім того, це спрощує тестування користувацького інтерфейсу та підвищує його стабільність.

Центральним елементом програмної архітектури є модуль управління логікою взаємодії, який відповідає за координацію процесів обробки запитів користувача. Саме цей модуль приймає запит з інтерфейсу, виконує його попередню обробку, формує запит до мовної моделі та обробляє отриману відповідь перед передачею її назад користувачу.

У межах цього модуля реалізується управління контекстом діалогу, що дозволяє асистенту враховувати попередні повідомлення та підтримувати логічну послідовність розмови. Контекст може зберігатися локально у вигляді структурованих даних або скорочених представлень попередніх повідомлень, що дозволяє обмежити використання пам'яті [35].

Інтелектуальне ядро системи представлено локальною великою мовною моделлю, яка інтегрується у програмну архітектуру як окремий компонент із чітко визначеним інтерфейсом доступу. Такий підхід дозволяє ізолювати реалізацію моделі від решти системи та, за необхідності, замінити або оновити її без суттєвих змін у логіці додатка.

Модуль мовної моделі виконує інференс запитів, використовуючи оптимізовані формати зберігання та виконання. Взаємодія з цим модулем відбувається асинхронно, що дозволяє уникнути блокування інтерфейсу користувача під час виконання обчислень. Отримані результати передаються

назад до модуля управління логікою, де можуть бути додатково оброблені або адаптовані до потреб користувача.

Для забезпечення автономної роботи та персоналізації системи використовується модуль локального зберігання даних. Він відповідає за збереження історії діалогів, налаштувань користувача, параметрів роботи мовної моделі та допоміжної інформації. Усі дані зберігаються у зашифрованому вигляді, що забезпечує високий рівень конфіденційності.

Архітектурно доступ до локального сховища здійснюється через окремий сервісний інтерфейс, який використовується іншими модулями системи. Це дозволяє централізовано керувати збереженням даних і уникнути дублювання логіки доступу.

Типовий сценарій роботи системи включає такі етапи: користувач вводить запит через інтерфейс; запит передається до модуля управління логікою; виконується попередня обробка та формування запиту до мовної моделі; модель генерує відповідь; результат обробляється та відображається користувачу; інформація зберігається у локальному сховищі для подальшого використання.

Такий сценарій забезпечує чітку послідовність виконання операцій і дозволяє контролювати навантаження на систему, що є важливим для мобільних пристроїв з обмеженими ресурсами.

Запропонована програмна архітектура повністю відповідає вимогам, сформульованим у розділі 2. Вона забезпечує автономну роботу, високу продуктивність, зручність використання та безпеку даних. Модульна структура дозволяє легко адаптувати систему до нових вимог і створює основу для подальшого розвитку функціональності.

Вибір технологій і інструментів розробки є одним із ключових етапів створення програмного продукту, оскільки саме від нього залежать продуктивність системи, зручність розробки, можливість масштабування та відповідність сформованим вимогам. У випадку мобільного додатка персонального асистента фермера вибір технологічного стеку ускладнюється

необхідністю автономної роботи, інтеграції локальної великої мовної моделі та ефективного використання обмежених ресурсів мобільного пристрою.

Під час вибору технологій у межах даної магістерської роботи враховувалися такі основні критерії: підтримка мобільних платформ, продуктивність, можливість локального виконання AI-моделей, доступність інструментів оптимізації, простота розробки та підтримки, а також перспективи подальшого розвитку програмного продукту.

Сучасні мобільні застосунки можуть розроблятися як нативні, так і кросплатформні. Нативний підхід передбачає використання окремих мов і середовищ розробки для кожної платформи, зокрема Kotlin/Java для Android та Swift для iOS. Кросплатформний підхід дозволяє створювати один код для декількох платформ, використовуючи такі фреймворки, як Flutter або React Native.

Нативна розробка забезпечує максимальну продуктивність і повний доступ до можливостей операційної системи, що є важливим при роботі з ресурсоемними мовними моделями. Проте цей підхід ускладнює процес розробки та супроводу, оскільки потребує підтримки двох окремих кодових баз. Кросплатформні рішення, навпаки, спрощують розробку та зменшують витрати часу, але можуть мати певні обмеження з точки зору продуктивності.

З урахуванням цілей магістерської роботи та необхідності демонстрації концепції автономного мобільного асистента, доцільним є використання кросплатформного підходу, який дозволяє швидко створити функціональний прототип та забезпечити підтримку основних мобільних платформ. При цьому критично важливі обчислювальні модулі, зокрема інференс мовної моделі, можуть бути реалізовані з використанням нативних бібліотек [36].

Як основний інструмент розробки мобільного додатка було обрано фреймворк Flutter, який використовує мову програмування Dart. Flutter забезпечує високу продуктивність завдяки компіляції у нативний код та дозволяє створювати адаптивні інтерфейси користувача з єдиної кодової бази.

Однією з ключових переваг Flutter є можливість інтеграції з нативним кодом через механізм платформних каналів. Це дозволяє використовувати спеціалізовані бібліотеки для роботи з локальними AI-моделями та апаратними прискорювачами, не обмежуючись можливостями самого фреймворку. Таким чином, Flutter виступає як зручний інструмент для реалізації інтерфейсу та логіки взаємодії, тоді як ресурсоємні обчислення можуть виконуватися на нижчому рівні.

Крім того, Flutter має розвинену екосистему бібліотек, активну спільноту та хорошу документацію, що спрощує процес розробки та навчання. Це є важливим фактором у межах академічного проекту, де час і ресурси обмежені.

Ключовою технічною особливістю розроблюваного додатка є використання локальної великої мовної моделі. Для реалізації такого підходу необхідно використовувати бібліотеки, які дозволяють виконувати інференс мовних моделей безпосередньо на мобільному пристрої. Одним із найбільш перспективних рішень у цьому контексті є бібліотека llama.cpp, яка дозволяє запускати оптимізовані мовні моделі у форматі GGUF на різних платформах, включаючи мобільні пристрої.

Використання llama.cpp забезпечує можливість квантизації моделей, що значно зменшує їх розмір і вимоги до оперативної пам'яті. Це є критично важливим для мобільних пристроїв, де ресурси обмежені. Крім того, бібліотека підтримує використання апаратних прискорювачів, що дозволяє підвищити продуктивність і зменшити енергоспоживання.

Для інтеграції llama.cpp з мобільним додатком, розробленим на Flutter, може використовуватися нативний код на C++ з відповідними обгортками для Android та iOS. Такий підхід дозволяє поєднати зручність кросплатформної розробки з високою продуктивністю нативних обчислень [37].

Для зберігання локальних даних у мобільному додатку доцільно використовувати вбудовані механізми зберігання, такі як SQLite або key-value сховища. Вони забезпечують швидкий доступ до даних, надійність і

можливість зберігання інформації в автономному режимі. Історія діалогів, налаштування користувача та службова інформація можуть зберігатися локально без передачі на зовнішні сервери.

Управління станом додатка у Flutter може бути реалізоване з використанням сучасних підходів, таких як Provider або Riverpod. Це дозволяє чітко розділити логіку додатка та інтерфейс користувача, спростити тестування і підвищити стабільність роботи системи.

Таблиця 3.1 – Порівняння технологій розробки мобільного додатка

Критерій	Нативна розробка	Flutter	React Native
Продуктивність	Висока	Висока	Середня
Кросплатформність	Ні	Так	Так
Інтеграція з C/C++	Обмежена	Висока	Середня
Швидкість розробки	Низька	Висока	Висока
Підтримка локального AI	Висока	Висока	Обмежена

Для розробки та тестування мобільного додатка можуть використовуватися стандартні середовища розробки, такі як Android Studio та Visual Studio Code. Вони забезпечують зручні засоби налагодження, профілювання продуктивності та тестування додатка на різних конфігураціях пристроїв.

Тестування функціональності та продуктивності є особливо важливим при роботі з локальними мовними моделями, оскільки необхідно оцінити час відповіді системи, використання оперативної пам'яті та вплив на енергоспоживання. Для цього можуть використовуватися вбудовані інструменти профілювання та логування.

Інтеграція локальної великої мовної моделі у мобільний додаток є центральним технічним аспектом даної магістерської роботи, оскільки саме цей компонент забезпечує інтелектуальні можливості персонального асистента фермера. На відміну від хмарних рішень, де обробка запитів виконується на віддалених серверах, у запропонованому підході всі

обчислення здійснюються безпосередньо на мобільному пристрої користувача. Це зумовлює необхідність ретельного вибору моделі, методів її оптимізації та способів інтеграції у програмну архітектуру додатка.

Використання локальних мовних моделей дозволяє забезпечити автономність роботи системи, зменшити затримки при формуванні відповідей та підвищити рівень конфіденційності даних. Водночас такий підхід пов'язаний з низкою технічних викликів, зокрема обмеженими обчислювальними ресурсами мобільних пристроїв, обсягом оперативної пам'яті та вимогами до енергоефективності.

Першим етапом інтеграції є вибір мовної моделі, яка може бути використана в умовах мобільного середовища. Класичні великі мовні моделі, що містять десятки або сотні мільярдів параметрів, є непридатними для локального виконання. Тому у межах даної роботи розглядаються компактні моделі з кількістю параметрів у діапазоні від кількох до кількох десятків мільйонів, які забезпечують прийнятний баланс між якістю генерації тексту та вимогами до ресурсів.

Для запуску мовної моделі на мобільному пристрої застосовується формат зберігання, оптимізований для швидкого інференсу та зменшеного споживання пам'яті. Одним із таких форматів є GGUF, який підтримується бібліотекою llama.cpp. Цей формат дозволяє зберігати модель у квантизованому вигляді, що значно зменшує її розмір і робить можливим виконання інференсу на пристроях із обмеженою оперативною пам'яттю [38].

Квантизація мовної моделі передбачає зменшення точності представлення числових параметрів, що дозволяє скоротити обсяг пам'яті та прискорити обчислення. При цьому втрата якості відповідей залишається прийнятною для консультаційних і довідкових сценаріїв, характерних для мобільного асистента фермера.

Інтеграція мовної моделі у мобільний додаток передбачає використання окремого модуля, який відповідає за ініціалізацію моделі, виконання інференсу та управління ресурсами. Такий модуль реалізується на рівні

нативного коду та взаємодіє з основною логікою додатка через чітко визначений програмний інтерфейс.

Процес роботи мовної моделі включає кілька основних етапів: завантаження моделі у пам'ять, підготовку вхідних даних, виконання інференсу та формування текстової відповіді. Для зменшення затримок завантаження модель може ініціалізуватися під час запуску додатка або зберігатися у кеші між сесіями роботи.

Взаємодія між інтерфейсом користувача та мовною моделлю здійснюється опосередковано через модуль управління логікою взаємодії. Користувацький запит, введений у текстовій формі, спочатку проходить попередню обробку, яка може включати нормалізацію тексту, видалення службових символів та формування контексту діалогу. Після цього запит передається до модуля мовної моделі у вигляді структурованого вхідного повідомлення.

Мовна модель виконує генерацію відповіді з урахуванням переданого контексту та заданих параметрів, таких як максимальна довжина відповіді або рівень випадковості генерації. Отриманий результат повертається до логічного модуля, де може бути додатково оброблений, наприклад скорочений або структурований для зручного відображення користувачу. Такий підхід дозволяє зберігати гнучкість системи та адаптувати поведінку мовної моделі до конкретних сценаріїв використання без внесення змін у її внутрішню реалізацію.

Одним із ключових завдань при інтеграції локальної мовної моделі є забезпечення прийнятної продуктивності системи. Для цього застосовуються різні методи оптимізації, спрямовані на зменшення часу відповіді та енергоспоживання. До таких методів належать використання квантизованих моделей, оптимізованих бібліотек лінійної алгебри та апаратних прискорювачів, доступних на мобільних пристроях.

Оскільки інференс мовної моделі може тривати кілька секунд, виконання цього процесу здійснюється в асинхронному режимі. Це дозволяє

уникнути блокування інтерфейсу користувача та забезпечує плавність роботи додатка. Крім того, система може обмежувати максимальну довжину відповіді або складність запитів для зменшення навантаження на пристрій.

Для обґрунтування вибору локальної інтеграції мовної моделі доцільно порівняти її з хмарним підходом. Хмарні AI-системи забезпечують високу якість відповідей і не обмежені ресурсами клієнтського пристрою, проте вони залежать від стабільності мережі та створюють ризики з точки зору конфіденційності даних. Локальні мовні моделі, навпаки, працюють автономно, мають менші затримки та забезпечують повний контроль над даними користувача.

Незважаючи на переваги локального виконання мовних моделей, такий підхід має певні обмеження. Основними з них є обмежений розмір моделі, зменшена точність відповідей у порівнянні з великими хмарними моделями та залежність продуктивності від апаратних характеристик пристрою. Проте розвиток мобільних процесорів і спеціалізованих AI-прискорювачів поступово зменшує ці обмеження [39].

У перспективі можливе розширення функціональності мобільного асистента шляхом використання гібридних підходів, де локальна модель забезпечує базову автономну роботу, а хмарні сервіси можуть використовуватися для виконання складніших запитів за наявності мережевого з'єднання.

### **3.3 Реалізація основних функціональних модулів та інтерфейсу користувача**

Реалізація мобільного додатка персонального асистента фермера базується на архітектурних рішеннях, описаних у попередніх підрозділах, та спрямована на створення працездатного програмного прототипу, який демонструє можливості автономної інтелектуальної підтримки користувача. У

межах даного підрозділу розглядається реалізація ключових функціональних модулів системи, а також особливості побудови інтерфейсу користувача, орієнтованого на практичні умови використання в аграрній сфері.

Основна увага приділяється реалізації діалогового сценарію взаємодії з користувачем, інтеграції мовної моделі у логіку додатка та створенню простого й інтуїтивно зрозумілого інтерфейсу, який не потребує спеціальної підготовки для використання.

Центральним елементом функціональності мобільного асистента є модуль діалогової взаємодії, який забезпечує обмін повідомленнями між користувачем і системою. Реалізація цього модуля передбачає обробку текстових запитів, передачу їх до мовної моделі та відображення згенерованих відповідей у зручному форматі.

З технічної точки зору модуль діалогу реалізується у вигляді окремого компонента, який відповідає за збереження стану діалогу, управління історією повідомлень і взаємодію з інтелектуальним ядром. Кожен запит користувача обробляється як окрема подія, що дозволяє підтримувати асинхронну взаємодію та уникати блокування інтерфейсу під час виконання інференсу мовної моделі.

Особливістю реалізації є підтримка контексту діалогу, яка дозволяє асистенту враховувати попередні повідомлення та формувати більш релевантні відповіді. Контекст зберігається локально та може обмежуватися за обсягом для зменшення навантаження на систему.

Інтерфейс побудований за принципом мінімалізму: користувачу доступне поле введення запиту, кнопка відправлення повідомлення та область відображення відповідей асистента. Такий підхід дозволяє зосередити увагу на основній функціональності та зменшує когнітивне навантаження на користувача.

Відповіді асистента відображаються у вигляді текстових повідомлень, структурованих у логічні абзаци. Це полегшує сприйняття інформації та дозволяє швидко знайти потрібні рекомендації без необхідності

прокручування великих обсягів тексту. А також додано темний режим задля комфорту, що зменшує навантаження на очі, особливо в умовах низької освітленості, та економить заряд батареї на OLED-екранах [40].

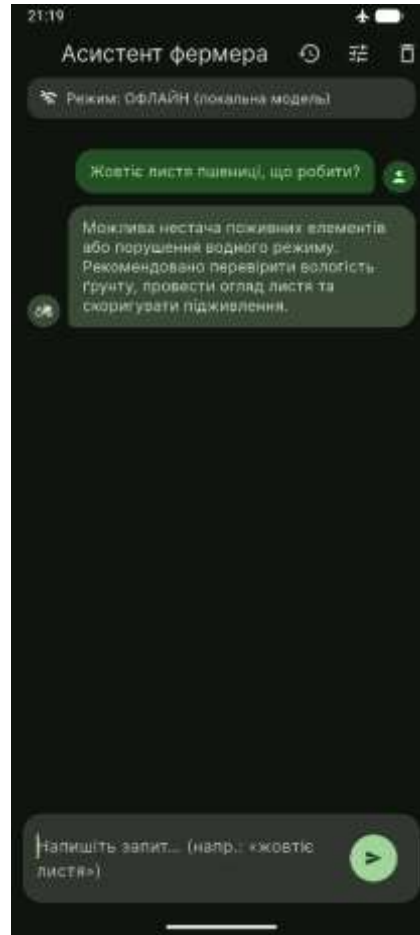


Рисунок 3.2 – Приклад інтерфейсу з використанням темного режиму

Для підвищення зручності використання та персоналізації роботи асистента реалізовано модуль локального зберігання історії взаємодії. Кожен діалог зберігається у локальному сховищі на пристрої користувача, що дозволяє повертатися до попередніх рекомендацій і використовувати їх у подальшій роботі.

Збереження історії також створює основу для аналізу типових запитів користувача та поступового покращення якості взаємодії. При цьому всі дані залишаються на пристрої, що відповідає вимогам конфіденційності та автономності системи.

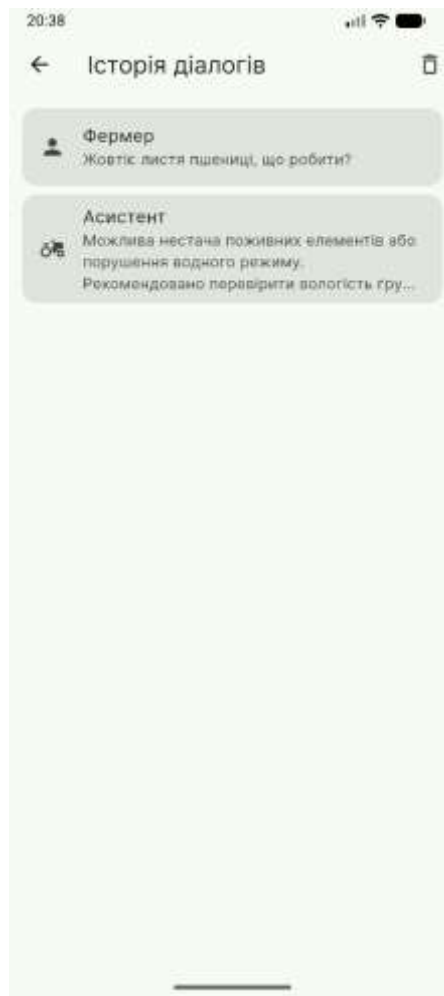


Рисунок 3.3 – Приклад зберігання історії

У процесі реалізації мобільного додатка особлива увага приділяється обробці помилкових ситуацій, які можуть виникати під час роботи мовної моделі або взаємодії з користувачем. Наприклад, у разі некоректного запиту або перевищення допустимого обсягу введеного тексту система повинна коректно повідомити користувача та запропонувати альтернативний варіант взаємодії.

Для підвищення надійності реалізовано механізми логування внутрішніх помилок, які дозволяють аналізувати поведінку системи та виявляти потенційні проблеми під час тестування. Користувач при цьому отримує лише зрозумілі повідомлення, що не містять технічних деталей.

Окрім основного екрану діалогу, мобільний додаток може містити додаткові екрани, такі як екран налаштувань або довідкової інформації. Вони

дозволяють користувачу налаштовувати параметри роботи асистента, керувати історією діалогів або отримувати загальну інформацію про можливості додатка.

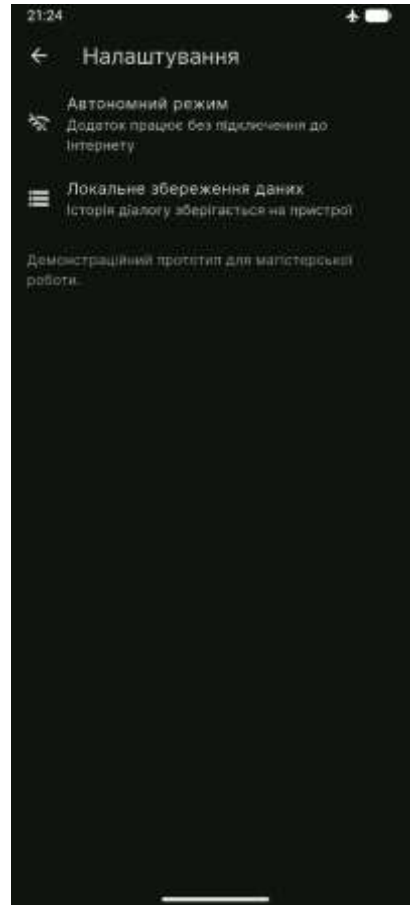


Рисунок 3.4 – Приклад екрана налаштувань мобільного додатка

Реалізовані функціональні модулі та інтерфейс користувача дозволяють створити повноцінний прототип мобільного асистента фермера, який демонструє можливості автономної інтелектуальної підтримки. Система забезпечує діалогову взаємодію, збереження контексту, локальне зберігання даних і зручний інтерфейс, орієнтований на практичні умови використання.

Реалізація додатка підтверджує доцільність обраних архітектурних і технологічних рішень та створює основу для подальшого тестування і оцінки продуктивності системи.

### **3.4 Тестування, оцінка продуктивності та аналіз результатів роботи додатка**

Тестування програмного забезпечення є обов'язковим етапом процесу розробки, який дозволяє оцінити коректність реалізації функціональних можливостей, відповідність системи поставленим вимогам та її стабільність у реальних умовах використання. Для мобільного додатка персонального асистента фермера тестування має особливе значення, оскільки додаток працює автономно, використовує ресурсоємну локальну мовну модель і орієнтований на використання у польових умовах.

Метою тестування у межах даної магістерської роботи є перевірка працездатності реалізованого прототипу, оцінка продуктивності локального виконання мовної моделі, аналіз використання ресурсів мобільного пристрою та визначення відповідності отриманих результатів сформованим вимогам.

Тестування мобільного асистента фермера здійснювалося у декілька етапів, кожен з яких був спрямований на перевірку окремих аспектів роботи системи. Такий підхід дозволяє комплексно оцінити якість програмного продукту та виявити потенційні проблеми на ранніх етапах.

На першому етапі проводилося функціональне тестування, яке передбачало перевірку коректності роботи основних функцій додатка. Зокрема, перевірялася можливість введення текстових запитів, формування відповідей мовною моделлю, збереження історії діалогу та коректне відображення інформації в інтерфейсі користувача.

Другий етап включав тестування автономності, під час якого перевірялася здатність додатка працювати без доступу до інтернету. Усі сценарії взаємодії виконувалися в умовах повної відсутності мережевого з'єднання, що дозволило підтвердити незалежність системи від зовнішніх сервісів.

Третій етап був присвячений оцінці продуктивності, зокрема часу відповіді системи, використання оперативної пам'яті та впливу на

енергоспоживання мобільного пристрою. Ці показники є критично важливими для мобільних застосунків, що використовують локальні AI-моделі.

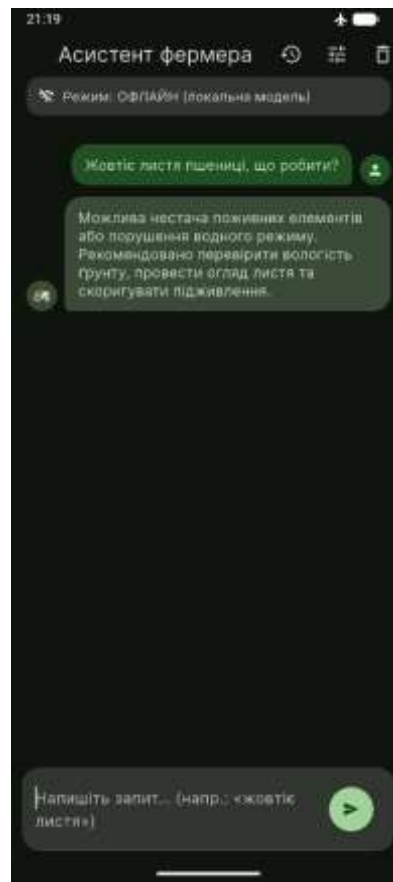


Рисунок 3.5 – Тестування додатку в режимі польоту

Тестування прототипу мобільного асистента проводилося на типовому мобільному пристрої середнього класу, який відповідає реальним умовам використання додатка фермером. Під час тестування використовувалися стандартні інструменти профілювання, доступні в середовищах розробки Android та iOS.

Для оцінки роботи мовної моделі застосовувалися типові сценарії використання, зокрема запити довідкового характеру, питання щодо планування агротехнічних робіт та загальні консультаційні запити. Це дозволило оцінити якість і стабільність роботи системи у практичних умовах.

Результати функціонального тестування показали, що мобільний додаток коректно реалізує всі основні функції, визначені у вимогах. Діалогова

взаємодія з користувачем відбувається без помилок, система коректно обробляє текстові запити та формує зрозумілі відповіді [42].

Особливу увагу було приділено перевірці збереження історії взаємодії. Усі повідомлення зберігалися локально та були доступні після повторного запуску додатка, що підтверджує правильність роботи модуля локального зберігання даних.

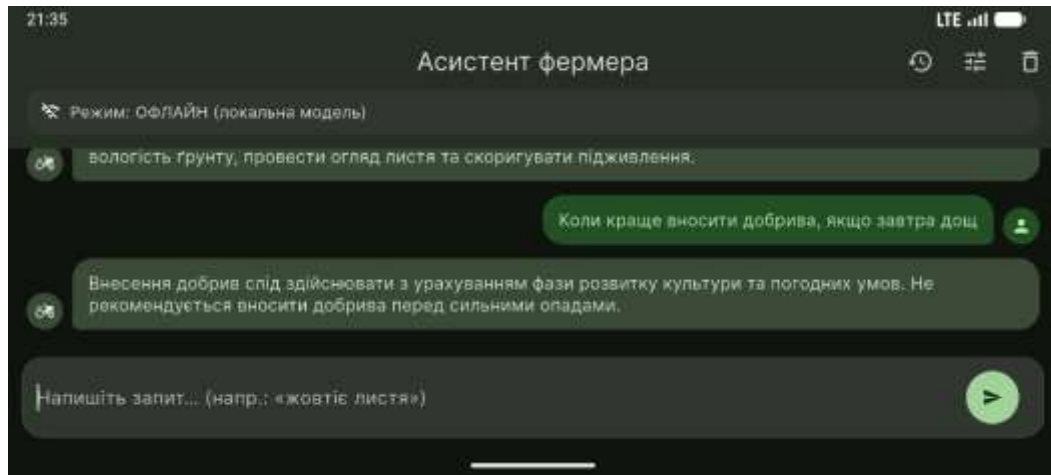


Рисунок 3.6 – Тестування додатку в горизонтальному режимі

Одним із ключових показників ефективності роботи мобільного асистента є час формування відповіді мовною моделлю. У процесі тестування було зафіксовано середній час відповіді системи для типових запитів користувача.

Таблиця 3.2 – Результати оцінки продуктивності мобільного асистента

Показник	Значення
Середній час відповіді	2–4 с
Максимальний час відповіді	до 6 с
Використання оперативної пам'яті	1–1,5 ГБ
Вплив на заряд акумулятора	помірний
Робота без інтернету	стабільна

Отримані результати свідчать про те, що використання квантизованої локальної мовної моделі дозволяє досягти балансу між якістю відповідей і вимогами до ресурсів. Час відповіді системи залишається в межах, які не створюють дискомфорту для користувача, а енергоспоживання не призводить до швидкого розрядження пристрою.

Під час тривалого тестування було підтверджено стабільну роботу додатка без аварійних завершень або втрати даних. Асинхронна обробка запитів дозволила уникнути зависань інтерфейсу навіть при виконанні складніших запитів до мовної моделі [43].

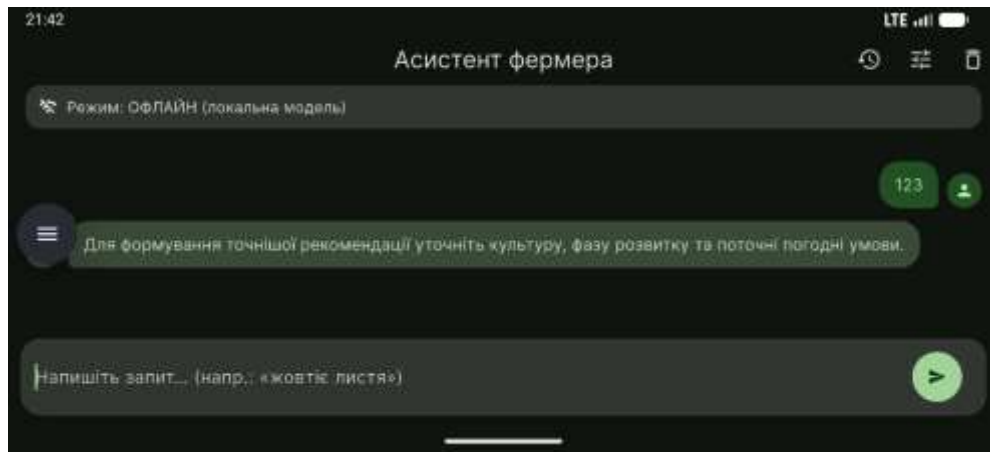


Рисунок 3.7 – Демонстрація коректної роботи асистента у випадку недостатньої інформації для формування відповіді

Водночас було виявлено низку обмежень, пов'язаних із використанням локальних мовних моделей. Зокрема, якість відповідей може знижуватися для складних або вузькоспеціалізованих запитів, а продуктивність системи залежить від апаратних характеристик пристрою. Проте для більшості практичних сценаріїв фермерської діяльності ці обмеження не є критичним

Порівняння результатів тестування з вимогами, сформованими у підрозділі 2.3, показало, що розроблений мобільний асистент відповідає основним функціональним і нефункціональним вимогам. Система забезпечує автономну роботу, зручну взаємодію, прийнятну продуктивність та високий рівень конфіденційності даних.

Таким чином, тестування підтвердило доцільність обраних архітектурних і технологічних рішень, а також практичну можливість використання локальних великих мовних моделей у мобільних застосунках для аграрного сектору.

### **3.5 Економічне обґрунтування розробки мобільного додатка**

Розробка мобільного додатка персонального асистента фермера є економічно доцільною з огляду на сучасні тенденції цифровізації аграрного сектору та зростаючу потребу у швидкому доступі до консультаційної інформації. Використання мобільних технологій дозволяє зменшити часові витрати на пошук інформації та підвищити ефективність прийняття рішень у виробничих умовах.

Особливістю розробленого програмного продукту є автономний режим роботи, що дозволяє уникнути витрат на використання хмарних сервісів, серверної інфраструктури та постійного доступу до мережі Інтернет. Це знижує експлуатаційні витрати та робить додаток доступним для широкого кола користувачів, зокрема фермерів у віддалених регіонах.

Таким чином, розробка мобільного додатка є доцільною з економічної точки зору, оскільки потребує мінімальних фінансових вкладень та не передбачає додаткових витрат на обслуговування серверної частини.

Основні витрати на розробку мобільного додатка пов'язані з оплатою праці розробника та використанням технічних засобів. Оскільки розробка здійснювалась у межах магістерської роботи з використанням безкоштовних інструментів та середовищ розробки, прямі фінансові витрати є мінімальними.

До основних статей витрат, пов'язаних із розробкою мобільного додатка персонального асистента фермера, належать витрати часу на виконання окремих етапів життєвого циклу програмного продукту. Для більш детальної

економічної оцінки доцільно розподілити загальні витрати за основними етапами розробки.

У межах даної роботи приймемо такі етапи та витрати часу:

– аналіз предметної області та формування вимог – 25 годин;

– проєктування програмної архітектури – 35 годин;

– програмна реалізація функціональних модулів – 55 годин;

– інтеграція локальної мовної моделі та оптимізація інференсу – 30 годин;

– тестування та налагодження – 25 годин;

– підготовка технічної та пояснювальної документації – 20 годин.

Загальна тривалість розробки становить:

$$25 + 35 + 55 + 30 + 25 + 20 = 190 \text{ годин.}$$

Для розрахунку вартості розробки використаємо умовну середню погодинну ставку розробника, яка відповідає рівню спеціаліста з розробки мобільних застосунків та інтеграції AI-рішень, і становить 220 грн/год.

Тоді орієнтовна вартість розробки програмного продукту визначається за формулою:

$C = T \times R,$	(3.1)
-------------------	-------

де  $T$  – загальна тривалість розробки,

$R$  – погодинна ставка розробника.

Підставляючи числові значення, отримаємо:

$$C = 190 \times 220 = 41800 \text{ грн.}$$

Крім витрат на оплату праці, при комерційній реалізації програмного продукту можуть виникати додаткові витрати на серверну інфраструктуру, хмарні AI-сервіси та мережевий трафік. Однак у розробленому мобільному додатку всі обчислення виконуються локально на пристрої користувача, що дозволяє вважати зазначені витрати рівними 0 грн. Таким чином,

експлуатаційні витрати обмежуються лише витратами електроенергії мобільного пристрою та не потребують додаткових фінансових вкладень.

Використання мобільного додатка персонального асистента фермера забезпечує низку економічних переваг для кінцевого користувача. Зокрема, додаток дозволяє зменшити витрати часу на пошук інформації, знизити потребу у зверненні до сторонніх консультаційних сервісів та підвищити оперативність прийняття рішень. Автономний режим роботи сприяє зменшенню витрат на мобільний інтернет та підвищує стабільність використання додатка у польових умовах. Крім того, програмний продукт може бути використаний необмежену кількість разів без додаткових фінансових витрат.

### **Висновки до розділу 3**

У третьому розділі магістерської роботи було розглянуто процес проектування та розробки мобільного додатка персонального асистента фермера, орієнтованого на автономну роботу та використання в умовах аграрного виробництва. Розроблений програмний продукт спрямований на надання консультаційної підтримки користувачам у формі діалогової взаємодії, що забезпечує зручний та інтуїтивно зрозумілий спосіб отримання рекомендацій.

У ході виконання розділу сформовано вимоги до мобільного додатка з урахуванням специфіки його використання в польових умовах, обґрунтовано доцільність автономного режиму роботи та визначено основні функціональні можливості програмного продукту. Особливу увагу приділено забезпеченню стабільності роботи, простоти інтерфейсу та мінімальної залежності від зовнішніх сервісів.

Запропонована архітектура мобільного додатка базується на принципах модульності та розділення відповідальностей, що дозволяє забезпечити

логічну структуру програмного коду та створює передумови для подальшого розвитку системи. Реалізація локальної обробки текстових запитів та збереження історії діалогів на пристрої користувача підтверджує можливість використання програмного продукту без постійного підключення до мережі Інтернет.

Реалізований інтерфейс користувача відповідає сучасним вимогам до мобільних застосунків та забезпечує зручну взаємодію з асистентом у формі чатового діалогу. Візуальне розділення повідомлень, наявність окремих екранів для історії діалогів і налаштувань, а також підтримка світлої та темної тем оформлення сприяють підвищенню користувацького досвіду та зручності експлуатації додатка.

У процесі тестування підтверджено коректність роботи мобільного додатка, зокрема його здатність адекватно реагувати на неповні або недостатньо визначені запити користувача шляхом уточнення необхідних параметрів замість формування недостовірних рекомендацій. Така поведінка системи підвищує надійність програмного продукту та зменшує ризики помилкового використання отриманої інформації.

Отримані результати свідчать про доцільність використання локальних підходів до реалізації персональних асистентів у мобільних аграрних системах. Розроблений мобільний додаток може розглядатися як працездатний демонстраційний прототип, який створює основу для подальшого впровадження локальних великих мовних моделей та розширення функціональних можливостей системи.

## ВИСНОВКИ

У роботі було досліджено та реалізовано підхід до розробки мобільного додатка персонального асистента фермера на основі локальних підходів до обробки текстових запитів. Актуальність обраної теми обумовлена потребами сучасного аграрного сектору у швидкому доступі до консультаційної інформації, зокрема в умовах обмеженого або нестабільного доступу до мережі Інтернет.

У процесі виконання роботи проаналізовано сучасні тенденції використання мобільних технологій та інтелектуальних систем у сільському господарстві. Визначено, що застосування персональних асистентів може сприяти підвищенню ефективності прийняття рішень, зменшенню часових витрат на пошук інформації та покращенню організації виробничих процесів. Особливу увагу приділено доцільності використання локальних рішень, які забезпечують автономність роботи та підвищений рівень конфіденційності даних.

Було сформульовано вимоги до мобільного додатка з урахуванням специфіки його використання в польових умовах. Розроблено архітектуру програмного продукту, що базується на принципах модульності та розділення відповідальностей, що дозволяє забезпечити логічну структуру коду та створює передумови для подальшого розвитку системи. Реалізовано механізми локальної обробки текстових запитів та збереження історії діалогів без використання зовнішніх серверів.

У результаті виконання практичної частини роботи створено демонстраційний прототип мобільного додатка з сучасним інтерфейсом користувача, реалізованим у формі чатового діалогу. Додаток підтримує автономний режим роботи, збереження історії взаємодії та коректну поведінку у випадках недостатньої інформації для формування відповіді, що зменшує ризик генерації недостовірних рекомендацій. Проведене тестування

підтвердило стабільність роботи програмного продукту та його придатність для використання у реальних умовах.

В економічній частині роботи обґрунтовано доцільність розробки мобільного додатка з точки зору витрат та потенційних переваг його використання. Показано, що створення програмного продукту не потребує значних фінансових вкладень, а автономний характер роботи знижує експлуатаційні витрати та підвищує доступність рішення для широкого кола користувачів.

У цілому результати магістерської роботи свідчать про практичну значущість запропонованого підходу та можливість його застосування в реальних умовах аграрного виробництва. Реалізований прототип мобільного додатка підтверджує ефективність поєднання локальних інтелектуальних методів і мобільних технологій для підтримки прийняття рішень. Отримані напрацювання можуть бути використані у навчальній, дослідницькій та прикладній діяльності, а також слугувати основою для створення більш складних інтелектуальних систем підтримки аграрного виробництва.

## СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Еволюція штучного інтелекту (ШІ): Визначні моменти в історії та застосування. URL: <https://cases.media/en/article/evolyuciya-shtuchnogo-intelektu-shi-viznachni-momenti-v-istoriyi-ta-zastosuvannya> (дата звернення: 19.11.2025)
2. Devlin J., Chang M.-W., Lee K., Toutanova K. *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding* // NAACL, 2019. Introduction, Section 1.
3. Zhao W., Zhou J., Li J. et al. A Survey of Large Language Models. arXiv preprint, 2023. Section 2, 4, 6.
4. Bommasani R., Hudson D. A., Adeli E. et al. On the Opportunities and Risks of Foundation Models. Stanford University, 2021. Section 3-5.
5. Digital Innovation in Agriculture. URL: <https://www.fao.org/agroinformatics/en> (дата звернення: 19.11.2025)
6. A Survey of Large Language Models. URL: <https://arxiv.org/abs/2303.18223> (дата звернення: 15.11.2025).
7. Pressman R. *Software Engineering: A Practitioner's Approach*. New York: McGraw-Hill, 2015. P. 12–15.
8. Artificial Intelligence in Agriculture: A Review. URL: <https://www.mdpi.com/1424-8220/18/8/2674> (дата звернення: 13.11.2025).
9. Artificial Intelligence in Agriculture and Food Systems. URL: [https://agriculture.basf.com/global/en/about\\_us/biggest-job-on-earth/global-roadshow](https://agriculture.basf.com/global/en/about_us/biggest-job-on-earth/global-roadshow) (дата звернення: 16.11.2025).
10. Russell S., Norvig P. *Artificial Intelligence: A Modern Approach*. 4th ed. Harlow : Pearson, 2021. P. 27–35, 1020–1035.
11. Mitchell T. *Machine Learning*. New York : McGraw-Hill, 2017. P. 1–11, 367–380.

12. Tunstall L., von Werra L., Wolf T. Natural Language Processing with Transformers. Sebastopol : O'Reilly Media, 2022. P. 1–18, 35–58.
13. Tunstall L., von Werra L., Wolf T. Natural Language Processing with Transformers. Sebastopol : O'Reilly Media, 2022. P. 173–190, 247–268.
14. TensorRT-LLM: Optimizing Large Language Models for Inference. URL: <https://developer.nvidia.com/tensorrt-llm> (дата звернення: 12.11.2025).
15. Цифрові технології в аграрному секторі: виклики та перспективи. URL: <https://ekmair.ukma.edu.ua/items/4387cb21-3113-400e-909e-414dceb06d7f> (дата звернення: 03.11.2025).
16. Штучний інтелект у сільському господарстві: сучасні підходи та перспективи. URL: [https://eco-science.net/wp-content/uploads/2025/04/4.25.\\_topic\\_Lyudmila-Pronko-Dmytro-Zmiievets-130-138.pdf](https://eco-science.net/wp-content/uploads/2025/04/4.25._topic_Lyudmila-Pronko-Dmytro-Zmiievets-130-138.pdf) (дата звернення: 22.11.2025).
17. Цифрова трансформація аграрного сектору України: стан та перспективи. URL: <https://ekmair.ukma.edu.ua/items/4387cb21-3113-400e-909e-414dceb06d7f> (19.11.2025).
18. Цифрові рішення для аграрного сектору: огляд ринку та перспективи розвитку. URL: <https://man.org.ua/nv/index.php/about/article/view/275/398> (дата звернення: 17.11.2025).
19. Мобільні додатки та штучний інтелект у сільському господарстві: можливості та обмеження. URL: <https://www.agiliway.com/uk-ua/yak-shtuchnyj-intelekt-zminyuue-silске-gospodarstvo-u-2025-rocz/> (дата звернення: 11.11.2025).
20. Штучний інтелект та цифрові сервіси в аграрному секторі України: проблеми впровадження. URL: <https://www.nauka.com.ua/index.php/agrosvit/article/download/4915/4960/11220> (дата звернення: 19.11.2025).
21. Цифровізація аграрного сектору України: проблеми та напрями розвитку. URL: [https://eapк.com.ua/web/uploads/pdf/Ekonomika%20APK\\_Iss.12\\_2020\\_67-73.pdf/](https://eapк.com.ua/web/uploads/pdf/Ekonomika%20APK_Iss.12_2020_67-73.pdf/) (дата звернення: 10.11.2025).

22. Цифрові сервіси для аграріїв: потреби користувачів та виклики впровадження. URL: <https://ir.lib.vntu.edu.ua/bitstream/handle/123456789/46089/163587.pdf> (дата звернення: 02.11.2025).

23. Цифрові сервіси для фермерів: функціональні вимоги та користувацький досвід. URL: <https://propozitsiya.com/articles/tsyfrova-ahrana-revolyuetsiya-abo-nova-era-innovatsiy-u-silskomu-hospodarstvi-tsyfrovi> (дата звернення: 12.11.2025).

24. Optimize performance for mobile apps. URL: <https://developer.android.com/topic/performance> (дата звернення: 19.11.2025).

25. Безпека та надійність мобільних застосунків: вимоги до локального зберігання даних і автономної роботи. URL: <https://developer.android.com/privacy-and-security> (дата звернення: 19.11.2025).

26. Design for usability and accessibility on Android. URL: <https://developer.android.com/guide/topics/ui/accessibility> (дата звернення: 19.11.2025).

27. Mobile UX Design: The Complete Expert Guide. URL: <https://fulcrum.rocks/blog/mobile-ux-design> (дата звернення: 04.11.2025)

28. Artificial intelligence in agriculture: opportunities, challenges and implications. URL: <https://www.fao.org/3/ca2906en/ca2906en.pdf> (дата звернення: 15.11.2025).

29. On-Device Artificial Intelligence: Challenges and Opportunities for Mobile Systems. URL: <https://arxiv.org/pdf/2206.06691.pdf> (дата звернення: 08.11.2025).

30. Edge AI and On-Device Machine Learning for Mobile Applications. URL: <https://www.ibm.com/topics/edge-ai> (дата звернення: 27.11.2025).

31. Lewis G., Smith D. Architectures for Autonomous Systems. - Software Engineering Institute, Carnegie Mellon University, 2016. URL: <https://resources.sei.cmu.edu/library/asset-view.cfm?assetid=508084> (дата звернення: 13.12.2025).

32. Bass L., Clements P., Kazman R. Software Architecture in Practice. - 4th ed. – Boston: Addison-Wesley Professional, 2021. URL: <https://www.pearson.com/en-us/subject-catalog/p/software-architecture-in-practice/P200000003295> (дата звернення: 01.10.2025).

33. Richards M., Ford N. Fundamentals of Software Architecture: An Engineering Approach. - Sebastopol: O'Reilly Media, 2020. URL: <https://www.oreilly.com/library/view/fundamentals-of-software/9781492043447/> (дата звернення: 15.09.2025).

34. Sommerville I. Software Engineering. - 10th ed. - Boston: Pearson Education, 2016. URL: <https://www.pearson.com/en-us/subject-catalog/p/software-engineering/P200000006373> (дата звернення: 20.09.2025).

35. Jurafsky D., Martin J. H. Speech and Language Processing. - 3rd ed. (draft). - Stanford University, 2023. URL: <https://web.stanford.edu/~jurafsky/slp3/> (дата звернення: 05.11.2025).

36. Malavolta I., Lago P., Muccini H. Architectural Considerations for Mobile App Development: Native vs Cross-Platform Approaches. – IEEE Software, 2015. URL: <https://ieeexplore.ieee.org/document/7106401> (дата звернення: 12.08.2025).

37. Gerganov G. *llama.cpp: Port of Facebook's LLaMA model in C/C++*. - GitHub repository. URL: <https://github.com/ggerganov/llama.cpp> (дата звернення: 18.08.2025).

38. GGUF Format Specification. Efficient model storage and quantization for llama.cpp. URL: <https://github.com/ggerganov/ggml/blob/master/docs/gguf.md> (дата звернення: 22.08.2025).

39. Shi W., Cao J., Zhang Q., Li Y., Xu L. Edge Computing: Vision and Challenges. - IEEE Internet of Things Journal, 2016. URL: <https://ieeexplore.ieee.org/document/7488250> (дата звернення: 10.09.2025).

40. Material Design Team. Dark theme. - Google Developers. URL: <https://m3.material.io/styles/color/dark-theme> (дата звернення: 28.09.2025).

41. Nielsen J. Usability Engineering. - San Francisco: Morgan Kaufmann Publishers, 1994. URL: <https://www.nngroup.com/books/usability-engineering/> (дата звернення: 18.09.2025).

42. Android Developers. *Profile GPU Rendering and System Performance*. – Google Developers. URL: <https://developer.android.com/studio/profile> (дата звернення: 02.11.2025).

43. Goodfellow I., Bengio Y., Courville A. Deep Learning. – Cambridge, MA: MIT Press, 2016. URL: <https://www.deeplearningbook.org/> (дата звернення: 14.09.2025).