

Міністерство освіти і науки України
Криворізький національний університет
Кафедра моделювання та програмного забезпечення

КВАЛІФІКАЦІЙНА РОБОТА
на здобуття ступеня вищої освіти бакалавра
зі спеціальності 121 – Інженерія програмного забезпечення

На тему: Розробка інформаційної системи для обліку роботи аграрної компанії

Засвідчую, що в цій
кваліфікаційній роботі немає
запозичень із праць інших
авторів без відповідних
посилань.

Студент гр. ІПЗ–21–1

_____ / Д. Д. Опарастюк /

Керівник кваліфікаційної
роботи

_____ / Н. О. Карабут /

Завідувач кафедри

_____ / А. М. Стрюк /

Кривий Ріг

2025

Криворізький національний університет

Факультет: Інформаційних технологій

Кафедра: Моделювання та програмного забезпечення

Ступінь вищої освіти: бакалавр

Спеціальність: 121 – Інженерія програмного забезпечення

ЗАТВЕРДЖУЮ:

Зав. кафедри

_____ А. М. Стрюк

«_____» _____ 2025 р.

ЗАВДАННЯ

на кваліфікаційну роботу

студенту групи ІПЗ–21–1 Опарастюку Даніілу Денисовичу

1. Тема: Розробка інформаційної системи для обліку роботи аграрної компанії
затверджена наказом по КНУ № 199с від «10» квітня 2025р.
2. Термін подання студентом закінченої роботи: «03» червня 2025р.
3. Вихідні дані по роботі: розроблені програмні засоби мають реалізувати облік інформації в аграрних компаніях.
4. Зміст пояснювальної записки (перелік питань, що їх треба розробити):
проаналізувати існуючі програми з подібною функціональністю, обґрунтувати необхідні функції розроблюваної системи, спроектувати програмне забезпечення та виконати його розробку, здійснити тестування розробленої системи.
5. Перелік ілюстративного матеріалу: функціональна схема, блок–схема алгоритму, зображення екранних форм додатку.

Календарний план:

№	Найменування етапів кваліфікаційної роботи	Термін виконання етапів роботи
1	Розгляд літературних джерел та пошук інтернет-ресурсів з заданої тематики	11.01.25 – 20.01.25
2	Аналіз існуючих методів вирішення проблеми	21.01.25 – 02.02.25
3	Формулювання актуальності роботи і постановка завдань	03.02.25 – 19.02.25
4	Оформлення матеріалів першого розділу роботи	20.02.25 – 02.03.25
5	Створення функціональної системи та алгоритму додатку	03.03.25 – 19.03.25
6	Оформлення матеріалів другого розділу роботи	20.04.25 – 09.04.25
7	Розробка баз даних, інтерфейсу програмного забезпечення, програмних модулів	10.04.25 – 01.05.25
8	Оформлення матеріалів третього розділу роботи	02.05.25 – 07.05.25
9	Тестування розробленої програми	08.05.25 – 16.05.25
10	Оформлення пояснювальної записки	17.05.25 – 30.05.25

Дата видачі завдання: «02» січня 2025 р.

Студент: _____ / Д. Д. Опарастюк /

Керівник роботи: _____ / Н. О. Карабут /

РЕФЕРАТ

СІЛЬСЬКОГОСПОДАРСЬКЕ ПІДПРИЄМСТВО, АГРАРНА КОМПАНІЯ, ОБЛІК РОБОТИ, ІНФОРМАЦІЙНА СИСТЕМА.

Пояснювальна записка: 70 с., 5 табл., 18 рис., 1 дод., 16 джерел.

Мета кваліфікаційної роботи: розробка інформаційної системи для обліку роботи аграрної компанії.

Об'єкт проектування: інформаційна система для обліку роботи аграрної компанії.

У теоретичній частині роботи проаналізовано наявні на сьогодні програмні рішення для автоматизації аграрних компаній. Визначені переваги та недоліки існуючих програмних продуктів. Обґрунтовані актуальність роботи, мета та сформульовані завдання для подальшого дослідження.

У практичній частині кваліфікаційної роботи реалізовано функціональну схему розроблюваної інформаційної системи, структуру даних, що використовуються в додатку та алгоритм його роботи. Виконані проектування та розробка баз даних, розроблено інтерфейс програмного продукту та програмну логіку інформаційної системи для обліку роботи аграрної компанії. Проведено тестування розробленого програмного забезпечення.

Інформаційна система для обліку роботи аграрної компанії може бути використана як в умовах невеликих фермерських господарств, так і у великих агрохолдингах.

ABSTRACT

FAST LENDING, LOANS, FINANCIAL ACCOUNTING,
INFORMATION SYSTEM, APPLICATION.

Explanatory note: 70 p., 5 tables, 18 pics, 1 pp., 16 sources.

The aim of the work: development of an information system for accounting for the work of an agricultural company.

Design object: the information system for accounting for the work of an agricultural company.

In the theoretical part of the qualifying work, the current software solutions for accounting for the work of an agricultural company have analyzed. The advantages and disadvantages of existing software products have identified. The work's relevance, purpose, and objectives for further research have formulated.

In the practical part of the qualification work, the functional scheme of the product has developed, the data structure used in the application and the algorithm of the application have implemented. The design and development of databases have carried out; the software product interface and the program logic of the system for accounting for the work of an agricultural company have developed. The developed software was tested.

The information system for accounting for an agricultural company's work can be used in small farms and extensive agricultural holdings.

ЗМІСТ

ВСТУП.....	8
1 РОЗГЛЯД ПИТАННЯ СТВОРЕННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ ДЛЯ ОБЛІКУ РОБОТИ АГРАРНОЇ КОМПАНІЇ.....	9
1.1 Короткий аналіз сільськогосподарської галузі України	9
1.2 Актуальність здійснення обліку інформаційних потоків в сільськогосподарських компаніях.....	12
1.3 Огляд продуктів-аналогів	14
1.3.1 AgriWebb	14
1.3.2 Trimble Ag Software	15
1.3.3 Agworld.....	17
1.4 Результати порівнянь продуктів для обліку роботи аграрного підприємства.....	19
2 ПРОЕКТУВАННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ ДЛЯ ОБЛІКУ РОБОТИ АГРАРНОЇ КОМПАНІЇ ТА ВИЗНАЧЕННЯ ЗАВДАНЬ	21
2.1 Постановка задачі на створення інформаційної системи для обліку роботи аграрної компанії.....	21
2.2 Вимоги інформаційної системи для обліку роботи аграрної компанії до апаратного забезпечення.....	23
2.3 Програмні засоби для розробки інформаційної системи для обліку роботи аграрної компанії.....	24
3 РОЗРОБКА ІНФОРМАЦІЙНОЇ СИСТЕМИ	26
3.1 Функціональна схема взаємодії компонентів інформаційної системи для обліку роботи аграрної компанії	26
3.2 Структура бази даних інформаційної системи для обліку роботи аграрної компанії.....	28

3.3 Алгоритм функціонування інформаційної системи	31
3.4 Програмна реалізація інформаційної системи для обліку роботи аграрної компанії.....	33
ВИСНОВКИ	40
ПЕРЕЛІК ПОСИЛАНЬ	41
Додаток А	43

ВСТУП

Аграрний сектор традиційно відіграє важливу роль в економіці України, забезпечуючи продовольчу безпеку країни, значну частину валютних надходжень та зайнятість великої кількості громадян. Україна по праву вважається одним із провідних світових експортерів зернових, олійних культур та іншої сільськогосподарської продукції. Однак останні роки стали справжнім випробуванням для вітчизняного аграрного бізнесу. Повномасштабна війна, що триває на території нашої держави, призвела до руйнування інфраструктури, втрати частини посівних площ, ускладнення логістики та зростання виробничих ризиків.

У цих складних умовах аграрії змушені працювати в умовах невизначеності, постійно адаптуватися до виникаючих проблем, шукати шляхи оптимізації витрат і підвищення ефективності виробництва. Відсутність стабільного доступу до ресурсів, перебої з постачанням техніки, добрив, пального, а також ризики втрати врожаю через військові дії — усе це вимагає від керівників агропідприємств максимальної оперативності у прийнятті рішень та точності в обліку.

Водночас, саме у кризові часи особливо гостро проявляється необхідність впровадження сучасних інформаційних технологій у сфері аграрного менеджменту. Автоматизація облікових і управлінських процесів дозволяє не лише зменшити вплив людського фактору на робочі процеси, а й забезпечити збереження та цілісність підприємницьких даних, швидко аналізувати поточну виробничу ситуацію, своєчасно реагувати на зміни та мінімізувати можливі втрати. Використання програмних рішень для централізованого зберігання даних про поля, техніку, врожай тощо, стає запорукою виживання і розвитку аграрного бізнесу за умов військового стану.

Саме тому розробка та впровадження сучасних автоматизованих систем управління аграрним підприємством є не просто актуальним, а стратегічно важливим завданням для України сьогодні.

1 РОЗГЛЯД ПИТАННЯ СТВОРЕННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ ДЛЯ ОБЛІКУ РОБОТИ АГРАРНОЇ КОМПАНІЇ

1.1 Короткий аналіз сільськогосподарської галузі України

Аграрний сектор України є складною системою, у якій поєднуються різноманітні напрями діяльності. В основі цієї системи лежить рослинництво, що забезпечує країну та світ продовольством, кормами, сировиною для промисловості. Українські поля славляться родючістю, а завдяки сприятливому клімату та сучасним технологіям тут вирощують широкий спектр культур — від зернових і олійних до овочів і фруктів. Завдяки багатим ґрунтам і працьовитості аграріїв, рослинництво стало справжньою візитівкою держави, а зібраний врожай часто знаходить шлях до споживачів далеко за межами країни.

Дослідниками зазначається [1], що Україна має понад 42 млн гектар земель сільськогосподарського призначення, що складають приблизно 70% площі нашої держави. Структура відсоткового розподілу цих земель за призначенням наведена на рисунку 1.1.

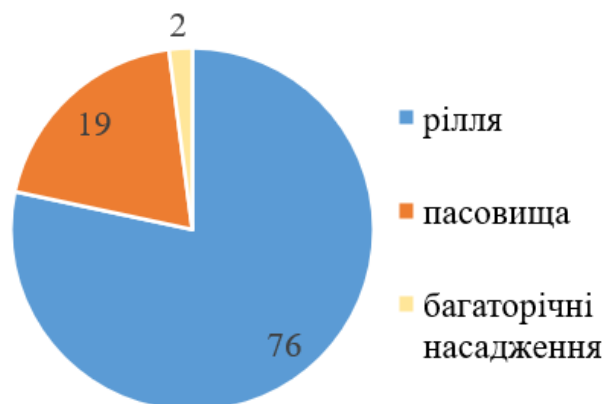


Рисунок 1.1 – Розподіл сільськогосподарських земель за призначенням (відповідно до [1])

На основі даних у роботі [2] структурний розподіл площ посівів можна візуалізувати діаграмою 1.2, з якої видно, що більше половини земель усіх

державних господарств зайняті зерновими культурами, близько третини посівів припадає на технічні культури, і решта (13%) складає площі, засіяні кормовими, овоче-баштаними культурами та картоплею (статистика наведена за 2022 р.).



Рисунок 1.2 – Структурний розподіл посівних площ (відповідно до даних [2])

Аграрна галузь України вже багато років залишається одним із найбільших виробників сільськогосподарської продукції у Європі. Щороку українські поля дають врожай, який вражає своїми обсягами навіть у складних умовах. Наприклад, у 2024 році, попри всі виклики, українським аграріям вдалося зібрати понад 45 мільйонів тонн зернових і зернобобових культур, зокрема кукурудзи — понад 22 млн тонн, а пшениці — понад 18 млн тонн [3].

Велика частина вирощеної продукції знаходить свого споживача за кордоном. Україна вже давно закріпила за собою статус одного з провідних експортерів зерна, а також соняшникової олії, яка користується попитом у багатьох країнах світу. За даними Українського клубу аграрного бізнесу [4], соняшникова олія залишається одним із ключових експортних продуктів: у 2023 році Україна експортувала понад 5 млн тонн цієї продукції.

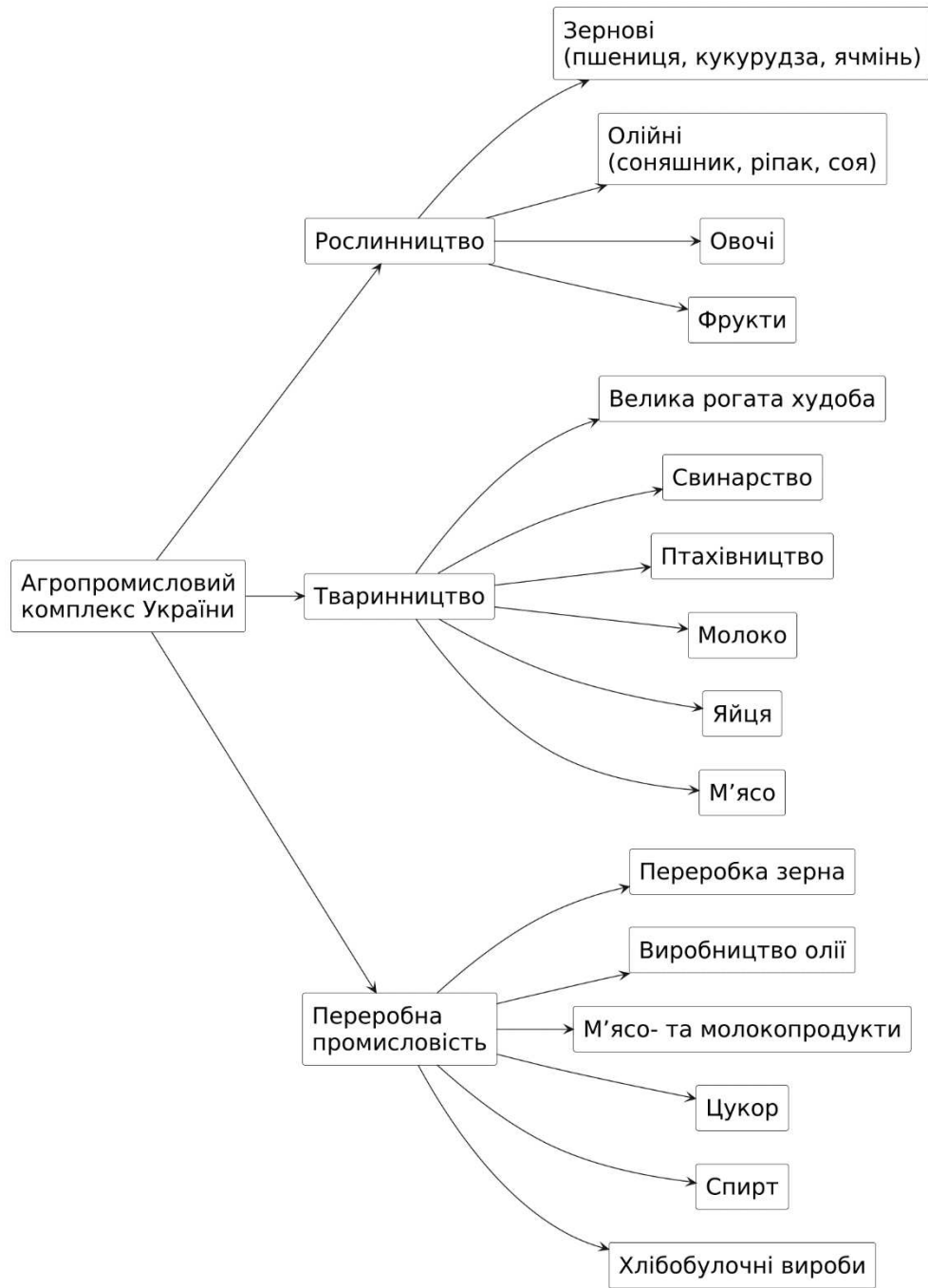


Рисунок 1.3 – Схема компонентів агропромислового комплексу України

Експорт агропродукції забезпечує значну частку валютних надходжень до бюджету та зміцнює економічну стабільність держави. Навіть у періоди логістичних труднощів, пов'язаних із війною та блокуванням портів, українські виробники знаходять нові шляхи для доставки продукції до споживачів — зокрема, через дунайські порти та залізничні маршрути до країн ЄС.

Щодо імпорту, то він відіграє допоміжну роль. Український агросектор потребує сучасного насіння, засобів захисту рослин, добрив і сільськогосподарської техніки, більшість з яких надходить із-за кордону. Саме завдяки імпорту інноваційних технологій та матеріалів вітчизняні виробники можуть підтримувати високу врожайність і якість продукції, що відповідає міжнародним стандартам.

Тваринництво, у свою чергу, доповнює цю картину, забезпечуючи внутрішній ринок м'ясом, молоком, яйцями та іншими продуктами тваринного походження. Ця підгалузь тісно пов'язана з рослинництвом, оскільки значна частина врожаю зернових і бобових використовується як корми для тварин. Розвиток тваринництва сприяє збереженню сільських територій, створенню робочих місць і підтримці соціальної стабільності в регіонах.

Важливим елементом аграрної системи є переробна промисловість, яка додає цінності сільськогосподарській продукції. В Україні активно функціонують підприємства, що займаються виробництвом олії, борошна, цукру, м'ясних і молочних виробів, а також іншої харчової продукції. Завдяки розвитку переробної промисловості, наша держава може не тільки експортувати сировину, а й пропонувати світу продукти з доданою вартістю, що підвищує конкурентоспроможність на міжнародних ринках [5,6].

Таким чином, аграрний сектор України — це взаємопов'язана система, де кожна підгалузь підтримує іншу, формуючи базу для економічного розвитку, забезпечення продовольчої безпеки та інтеграції у світові ринки.

1.2 Актуальність здійснення обліку інформаційних потоків в сільськогосподарських компаніях

Сьогоденний аграрний сектор переживає етап розвитку, що характеризуються ускладненням організаційної структури підприємств та впровадженням новітніх технологій. В умовах конкуренції та постійних змін ринкової кон'юнктури особливої ваги набуває питання ефективного

управління ресурсами аграрної компанії, оптимізації її виробничих процесів і забезпечення прозорості обліку аграрної продукції, розходних матеріалів, насіння тощо.

На практиці багато аграрних підприємств нашої держави досі використовують застарілі методи ведення обліку своїх справ — паперові журнали, розрізнені електронні таблиці або навіть просто усні домовленості між контрагентами. Такий підхід не дає можливості оперативно отримувати достовірну інформацію про стан справ на аграрному підприємстві, ускладнює аналіз виробничих показників, та навіть створює ризики втрати або спотворення важливих даних про роботу компанії. Крім того, ручне ведення обліку призводить до значних часових витрат працівників, підвищує ймовірність появи помилок та ускладнює формування звітності для представлення керівництву чи контролюючим органам.

Водночас сучасні інформаційні технології відкривають нові можливості для автоматизації облікових процесів сільськогосподарських підприємств, централізованого зберігання та обробки важливих даних, швидкого пошуку потрібної інформації та проведення її аналізу. Впровадження спеціалізованого програмного забезпечення може дозволити не лише підвищити ефективність управління аграрним підприємством, а й забезпечити його конкурентоспроможність на вітчизняному і світовому ринках, зменшити вплив людського фактору, своєчасно реагувати на зміни у виробничому процесі [7,8].

Розробка інформаційної системи для обліку роботи аграрної компанії є актуальною відповіддю на ці виклики. Вона покликана забезпечити комплексну автоматизацію обліку основних ресурсів аграрного підприємства — земельних ділянок, техніки, врожаю, складу та персоналу. Завдяки впровадженню такої системи підприємство отримує можливість працювати з єдиною базою даних, оперативно отримувати аналітичну інформацію для прийняття управлінських рішень, спрощувати підготовку звітності та підвищувати загальний рівень організації роботи.

Таким чином, створення і впровадження програмного продукту інформаційної системи для обліку роботи аграрної компанії є не лише актуальним, а й необхідним кроком для підвищення ефективності діяльності сучасного аграрного підприємства в умовах цифрової трансформації галузі.

1.3 Огляд продуктів-аналогів

1.3.1 AgriWebb

AgriWebb [9] — це цифрова платформа, яка допомагає фермерам організовувати всі основні процеси управління господарством у єдиному електронному середовищі.

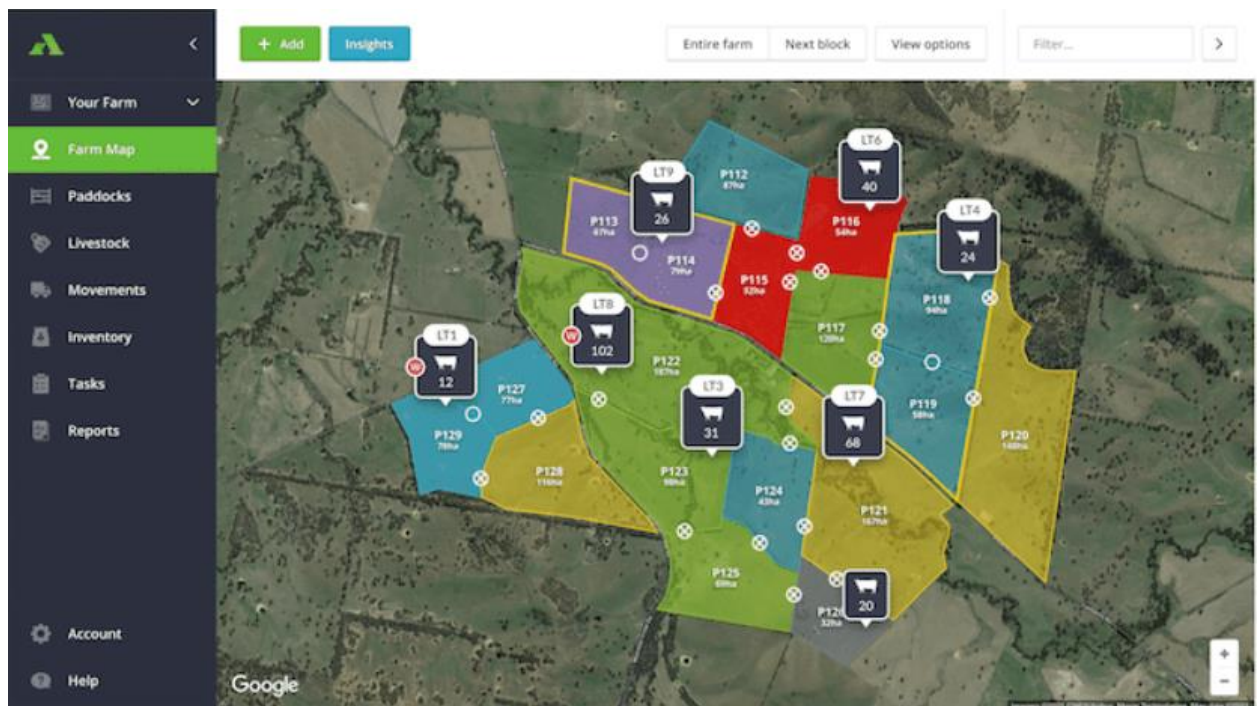


Рисунок 1.4 – Скріншот сервісу AgriWebb

Вона дозволяє вести облік тварин і полів, планувати агротехнічні заходи, контролювати використання техніки та ресурсів, а також аналізувати ефективність виробництва за допомогою інтегрованої аналітики. Користувачі можуть працювати як через веб-інтерфейс, так і з мобільного додатку, що особливо зручно під час роботи безпосередньо на полі чи фермі. Завдяки

інтерактивній карті господарства, фермери отримують уявлення про структуру своїх земель, розташування пасовищ, полів і об'єктів інфраструктури. Усі дії та зміни фіксуються в системі, що забезпечує прозорість операцій і полегшує командну роботу. Програма також дозволяє формувати звіти, які можна використовувати як для внутрішнього аналізу, так і для подання у державні органи.

Попри широкий функціонал, AgriWebb має і недоліки. По-перше, ця система орієнтована переважно на великі та середні фермерські господарства, тому для невеликих підприємств її використання може бути економічно недоцільним через високу вартість підписки. Деякі користувачі відзначають складність початкового налаштування, оскільки інтерфейс і логіка роботи можуть бути незвичними для тих, хто раніше не користувався подібними рішеннями. Також варто враховувати, що повноцінна робота з системою потребує інтернет-з'єднання, а це не завжди можливо у віддалених сільських районах. Нарешті, частина функцій та інтеграцій доступна лише у дорожчих тарифних планах, що може обмежувати можливості окремих користувачів.

1.3.2 Trimble Ag Software

Trimble Ag Software [10] — це комплексне рішення для цифрового управління сільськогосподарським виробництвом, яка поєднує в собі сучасні інструменти планування, моніторингу та аналізу агробізнесу. Вона створена для того, щоб фермери могли максимально ефективно використовувати свої ресурси, оптимізувати виробничі процеси та приймати обґрунтовані рішення на основі точних даних. Програмне забезпечення Trimble охоплює всі ключові аспекти аграрної діяльності: від планування посівних площ і внесення добрив до контролю за станом техніки, управління запасами на складах і ведення фінансової звітності.

Однією з особливостей Trimble Ag Software є тісна інтеграція з GPS-обладнанням, агродронами та системами точного землеробства. Завдяки цьому фермери отримують можливість у режимі реального часу відстежувати

виконання польових робіт, аналізувати карти врожайності, контролювати дотримання технологічних операцій і оперативно реагувати на зміни погодних умов чи стану посівів. Вся інформація зберігається у хмарі, що дозволяє мати до неї доступ із будь-якого пристрою — як з офісу, так і безпосередньо з поля.



Рисунок 1.5 – Trimble Ag Software

Trimble Ag Software пропонує гнучкий інтерфейс, який адаптується до потреб різних користувачів: як агрономів і менеджерів, так і механізаторів чи бухгалтерів. Програма дозволяє планувати роботи, призначати завдання співробітникам, вести облік виконаних операцій, формувати докладні звіти та аналізувати економічну ефективність кожного етапу виробництва. Особливу увагу розробники приділили питанням аналітики: система надає інструменти для порівняння врожайності на різних полях, аналізу витрат і доходів, прогнозування майбутніх результатів на основі історичних даних.

Trimble Ag Software підтримує командну роботу, дозволяючи декільком користувачам одночасно працювати з однією базою даних, обмінюватися інформацією та координувати дії. Завдяки широким можливостям інтеграції з іншими програмними продуктами та обладнанням, ця платформа може стати центром управління агропідприємством.

Серед недоліків Trimble Ag Software користувачі відзначають складність налаштування та необхідність адаптації до великої кількості функцій, що може вимагати додаткових навчальних активностей персоналу аграрної компанії. Також повідомляється про дещо високу вартість використання, особливо з урахуванням того, що система орієнтується на західні ринки.

1.3.3 Agworld

Agworld [11] — це хмарна платформа, розроблена для комплексного управління сільськогосподарськими підприємствами різного масштабу. Система орієнтована на створення єдиного інформаційного простору, де всі учасники виробничого процесу — від агрономів і фермерів до консультантів і менеджерів — можуть ефективно співпрацювати, обмінюватися даними й приймати обґрунтовані рішення.

Головна ідея Agworld полягає у централізації всієї агрономічної інформації. Платформа дозволяє детально планувати всі виробничі операції: від підготовки ґрунту, вибору культур і розрахунку потреб у насінні чи добривах до реєстрації фактичних робіт і контролю виконання завдань. Користувачі можуть створювати інтерактивні карти полів, фіксувати агрохімічні параметри, вести історію обробітку та спостерігати за змінами стану посівів у динаміці. Усі дані зберігаються у хмарі, що забезпечує доступ до них із будь-якого пристрою — як з комп'ютера в офісі, так і зі смартфона безпосередньо на полі.

Особливістю Agworld є можливість залучати до роботи консультантів, які можуть дистанційно переглядати дані, надавати рекомендації та допомагати у прийнятті рішень. Система підтримує гнучке управління

завданнями: менеджери можуть призначати роботи співробітникам, відстежувати їх виконання й отримувати зворотний зв'язок у режимі реального часу. Вбудовані аналітичні інструменти дозволяють аналізувати ефективність агротехнічних заходів, порівнювати врожайність на різних полях, планувати бюджети й формувати звітність для внутрішнього користування чи зовнішніх організацій.

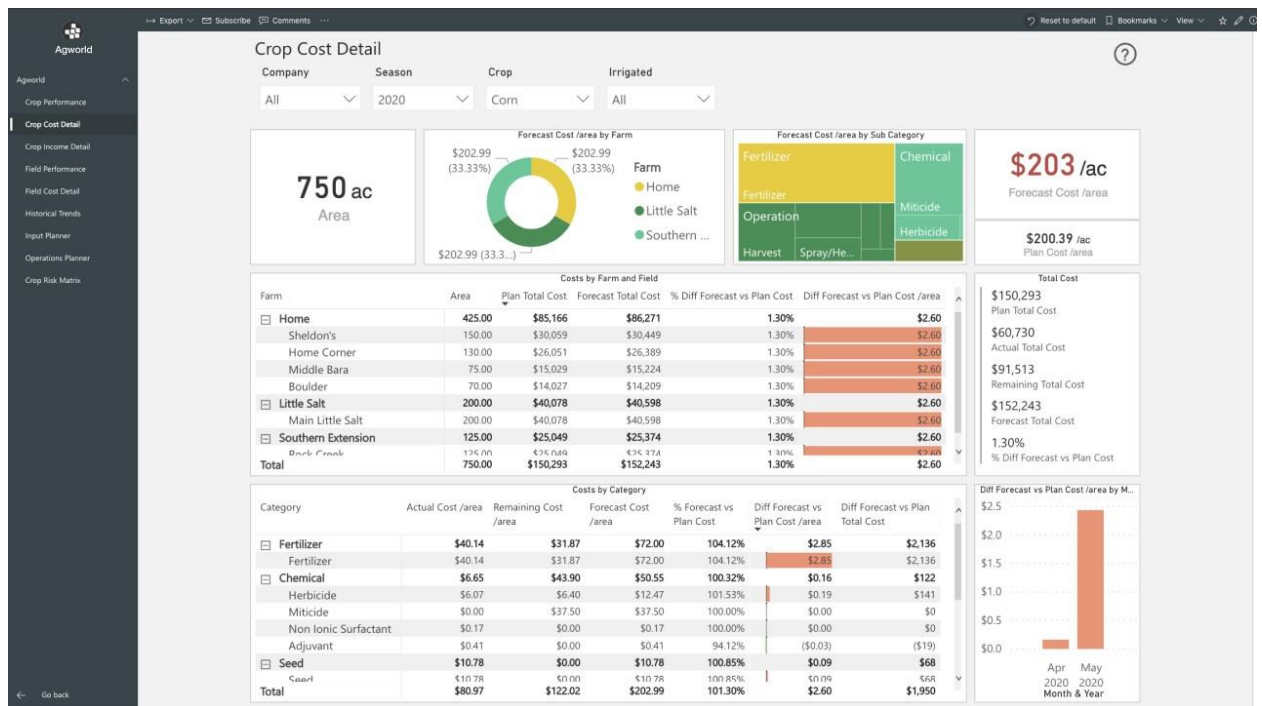


Рисунок 1.6 – Agworld

Agworld відзначається інтуїтивно зрозумілим інтерфейсом і високим рівнем адаптивності під потреби конкретного господарства. Платформа легко інтегрується з іншими цифровими рішеннями, що використовуються в агробізнесі, й активно підтримує командну роботу.

Щодо недоліків, то повідомляється, що для повноцінного використання всіх можливостей Agworld потрібне стабільне інтернет-з'єднання і певний час на навчання персоналу через складність швидкого опанування функціоналу системи.

1.4 Результати порівнянь продуктів для обліку роботи аграрного підприємства

Виконаний аналіз сучасних інформаційних систем для обліку роботи аграрного сектору показав, що більшість аналогів орієнтовані на максимальну автоматизацію виробничих процесів, аналітику та командну роботу в реальному часі. Проте такі рішення здебільшого реалізовані у вигляді хмарних сервісів, що, безумовно, відкриває широкі можливості для мобільності та спільного доступу до даних з будь-якої точки світу. Це особливо зручно для великих агрохолдингів із розгалуженою структурою, де важливо забезпечити оперативний обмін інформацією між різними підрозділами та спеціалістами.

Водночас, залежність від стабільного інтернет-з'єднання може стати суттєвою перешкодою для фермерських господарств, що працюють у віддалених регіонах, де якість мережі не завжди відповідає сучасним вимогам. У разі відсутності доступу до мережі користувачі ризикують втратити можливість оперативно вносити зміни або переглядати важливі дані. Ще одним аспектом, який викликає занепокоєння, є централізоване зберігання інформації на сторонніх серверах. Хоча це спрощує адміністрування та резервне копіювання, водночас виникають ризики несанкціонованого доступу, втрати конфіденційності або навіть витоку інформації через вразливості серверного ПЗ чи людський фактор. В той же час варто зауважувати на сьогоденні реалії та брати до уваги можливі відключення електроенергії та зв'язку, що може паралізувати використання веб-додатків.

Щодо функціональності, багато систем пропонують розширені можливості інтеграції з агродронами, GPS, фінансовими модулями й аналітичними інструментами. Однак така багатофункціональність часто ускладнює інтерфейс, роблячи його менш інтуїтивним для нових користувачів. Деякі продукти мають перевантажені меню, дрібні шрифти та велику кількість налаштувань, що може відлякувати людей без досвіду роботи з подібними системами. Окрім цього, більшість розглянутих популярних

платформ не підтримують українську мову, що створює додаткові труднощі для користувачів.

Враховуючи ці особливості, при створенні аграрної інформаційної системи доцільно зробити акцент на простоті та зрозумілості інтерфейсу, забезпечити підтримку української мови та надати можливість автономної роботи без постійного підключення до інтернету. Локальне зберігання даних підвищить рівень захисту інформації та дозволить уникнути ризиків, пов'язаних із зовнішніми серверами. Також важливо уникати перевантаження програми зайвими функціями, зосередившись на найнеобхідніших інструментах для обліку земель, техніки, врожаю, складу та персоналу. Такий підхід дозволить створити доступний і надійний продукт, який відповідатиме реальним потребам аграрних підприємств різного масштабу, а також забезпечить зручність і безпеку роботи для кінцевих користувачів.

Окремо варто зазначити, що, на відміну від багатьох розглянутих аналогів, впровадження гнучкої ліцензійної політики з можливістю одноразової покупки програми замість щомісячної підписки зробить продукт більш привабливим для невеликих господарств.

2 ПРОЕКТУВАННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ ДЛЯ ОБЛІКУ РОБОТИ АГРАРНОЇ КОМПАНІЇ ТА ВИЗНАЧЕННЯ ЗАВДАНЬ

2.1 Постановка задачі на створення інформаційної системи для обліку роботи аграрної компанії

Завданням на кваліфікаційну роботу є розробка інформаційної системи для обліку роботи аграрної компанії.

В межах цієї роботи планується розробити сучасну інформаційну систему для підтримки діяльності фермерських господарств та аграрних підприємств. Попередньо було проведено аналіз предметної області та досліджено специфіку роботи фермерських господарств нашої держави, окрему увагу було приділено особливостям обліку збору сільськогосподарської продукції. Також був проведений огляд вже існуючих комерційних інформаційних систем, що використовуються в аграрному секторі.

Для досягнення поставленої мети необхідно виконати такі основні етапи проектування:

а) розробка структурної схеми інформаційної системи, яка відобразить логічну організацію даних та взаємозв'язки між основними сутностями, що використовуватимуться в програмі;

б) проектування бази даних, що полягатиме в розробці її структури, реалізації забезпечення цілісності, узгодженості та максимізації швидкості доступу до інформації;

в) вибір системи керування базами даних, що буде використано в інформаційній системі;

г) реалізація бази даних інформаційної системи, тобто створення її фізичної моделі;

д) дослідження та вибір найбільш ефективних способів зберігання, пошуку, фільтрації та аналізу інформації, що забезпечать високу продуктивність системи;

е) вибір технологічного стеку, що полягає у визначенні оптимальної мови програмування, виборі середовища розробки (IDE), враховуючи специфіку поставлених при проектуванні завдань та існуючі актуальні вимоги до програмного забезпечення;

ж) формулювання вимог з боку інформаційної системи до апаратного забезпечення комп'ютерної техніки, на якій буде відбуватися її розгортання та використання;

з) проектування архітектури програми та розробка основних класів модулів, які відповідатимуть за роботу з різними сутностями системи;

і) складання блок-схеми алгоритму роботи інформаційної системи аграрної компанії та з відображенням її функцій для наочності та полегшення подальшої реалізації;

к) розробка користувацького інтерфейсу, що полягатиме у проектуванні та реалізації інтуїтивно зрозумілого та ергономічного інтерфейсу, який забезпечить зручну взаємодію користувача зі створюваною інформаційною системою;

л) створення функціональних модулів для введення, редагування, видалення, пошуку та виведення інформації, а також експорту даних у документи в форматі pdf;

м) наповнення інформаційної системи синтетичними даними, які відображатимуть діяльність фермерських господарств та використання агроугідь нашої країни;

н) підготовка посібника користувача із покроковими інструкціями по використанню.

Таким чином, виконання зазначених етапів проектування дозволить створити повноцінну – зручну та функціональну – інформаційну систему, яка

відповідатиме вимогам аграрного бізнесу та сприятиме підвищенню ефективності діяльності фермерських господарств.

2.2 Вимоги інформаційної системи для обліку роботи аграрної компанії до апаратного забезпечення

Для ефективної роботи інформаційної системи обліку роботи аграрної компанії користувачу рекомендується мати комп'ютер із наступною конфігурацією:

а) багатоядерний процесор із тактовою частотою від 2 ГГц (на кшталт Intel Core i3/i5 або AMD Ryzen 3 і вище) для забезпечення швидкої обробки команд;

б) оперативна пам'ять (RAM) у розмірі 4 ГБ, що дозволить одночасно працювати з іншими додатками операційної системи без зниження продуктивності;

в) монітор із роздільною здатністю від 1366×768 пікселів для комфортного перегляду таблиць, звітів та роботи з користувацьким інтерфейсом;

г) графічна підсистема на базі інтегрованої або дискретної відеокарти з об'ємом пам'яті від 64 МБ, чого достатньо для коректного відображення інтерфейсу та звітів;

д) мінімум 50 МБ вільного простору на жорсткому диску для встановлення інформаційної системи, зберігання бази даних та резервних копій (обсяг може збільшуватись залежно від кількості наявних записів у базі даних);

е) наявність пристроїв введення – стандартної клавіатури та миші для зручного введення та редагування даних;

ж) операційна система Windows 10 або 11 (як 32, так і 64-розрядна версія системи);

з) наявність програми для відтворення pdf файлів (на кшталт Adobe Acrobat Reader [12]) для перегляду експортованих звітів.

2.3 Програмні засоби для розробки інформаційної системи для обліку роботи аграрної компанії

При розробці інформаційної системи для обліку роботи аграрної компанії буде застосовано бібліотеки Microsoft .NET Framework [13]. Робота з нею дозволяє використовувати різні мови програмування. Втім, однією з найбільш розповсюджених та затребуваних на сьогодні є мова С# [14]. Вона може бути вдало застосована для розробки з використанням бібліотеки Microsoft .NET Framework. Мова С#, в свою чергу, є доволі зручною для розробки, та дозволяє швидко та ефективно розробляти додатки з візуальним інтерфейсом. Застосування комплексу із бібліотеки Microsoft .NET Framework та мови програмування С# дозволить пришвидшити процес створення системи за рахунок наявності великої кількості довідкових матеріалів та розвинутого ком'юніті.

Для реалізації даного комплексу програмних засобів доцільно використовувати сучасне інтегроване середовище розробки Visual Studio 2022 [15]. Visual Studio 2022 є інструментом, який забезпечує розробників широким спектром можливостей для створення, налагодження та тестування програмного забезпечення на платформі .NET Framework. Однією з головних переваг Visual Studio 2022 є підтримка нею новітніх стандартів та технологій, що дає змогу створювати сучасні додатки. Інтерфейс інтегрованого середовища інтуїтивно зрозумілий, а наявність шаблонів і майстрів проектів значно прискорює процес створення нових програмних рішень.

Крім того, Visual Studio 2022 має розвинені засоби для роботи з мовою програмування С#, включаючи інтелектуальне автодоповнення коду (IntelliSense [16]), інтегровану систему контролю версій, а також інструменти для профілювання та оптимізації продуктивності додатків. Вбудований відлагоджувач дозволяє ефективно знаходити та виправляти помилки на різних етапах розробки, що сприяє підвищенню якості створюваного продукту. Завдяки активній підтримці з боку спільноти та регулярним

оновленням, Visual Studio 2022 залишається одним з найкращих виборів для професійної розробки програмного забезпечення під платформу .NET.

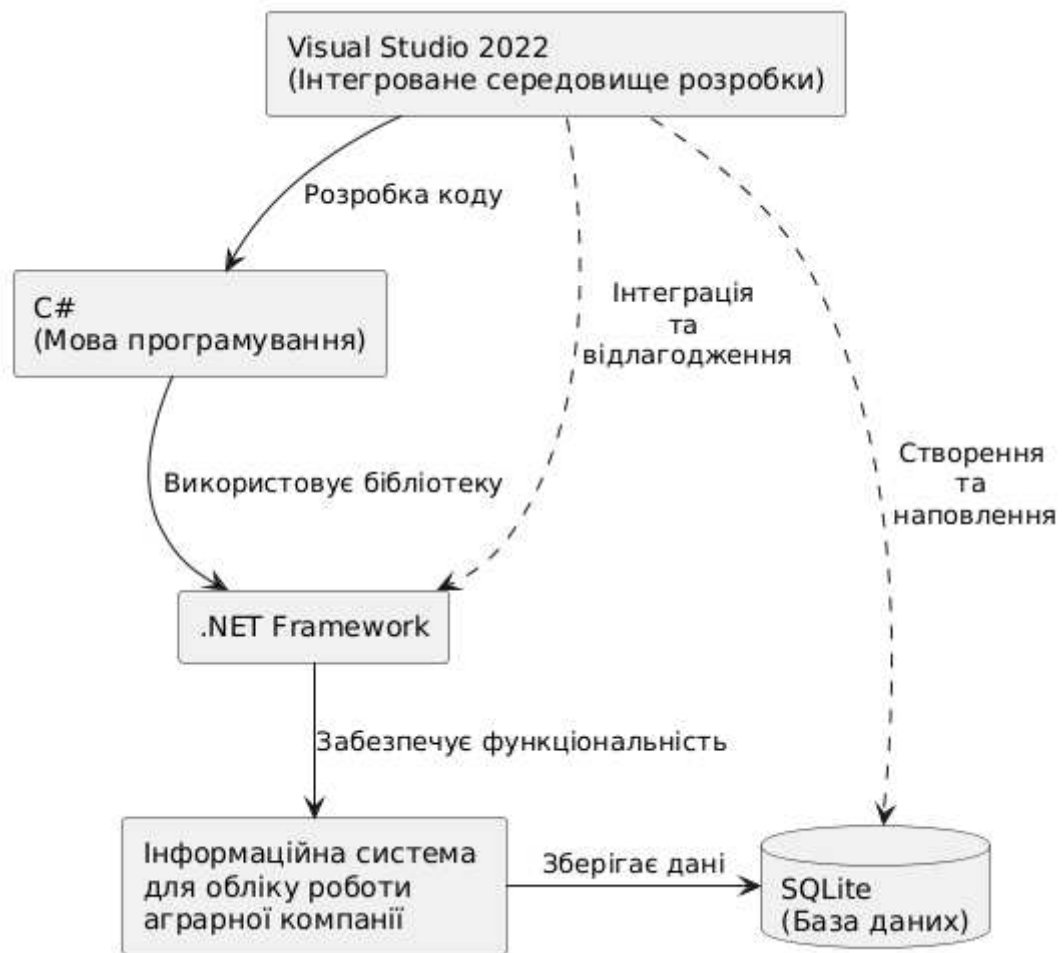


Рисунок 2.1 – Візуалізація взаємозв’язку технологій, що використовуються для розробки інформаційної системи для обліку роботи аграрної компанії

Таким чином, використання бібліотеки Microsoft .NET Framework у поєднанні з мовою програмування C# та інтегрованим середовищем розробки Visual Studio 2022 забезпечить ефективність та високу якість розробки інформаційної системи для обліку роботи аграрної компанії. Це дозволить скоротити терміни впровадження проекту та забезпечити його підтримку в процесі експлуатації.

3 РОЗРОБКА ІНФОРМАЦІЙНОЇ СИСТЕМИ

3.1 Функціональна схема взаємодії компонентів інформаційної системи для обліку роботи аграрної компанії

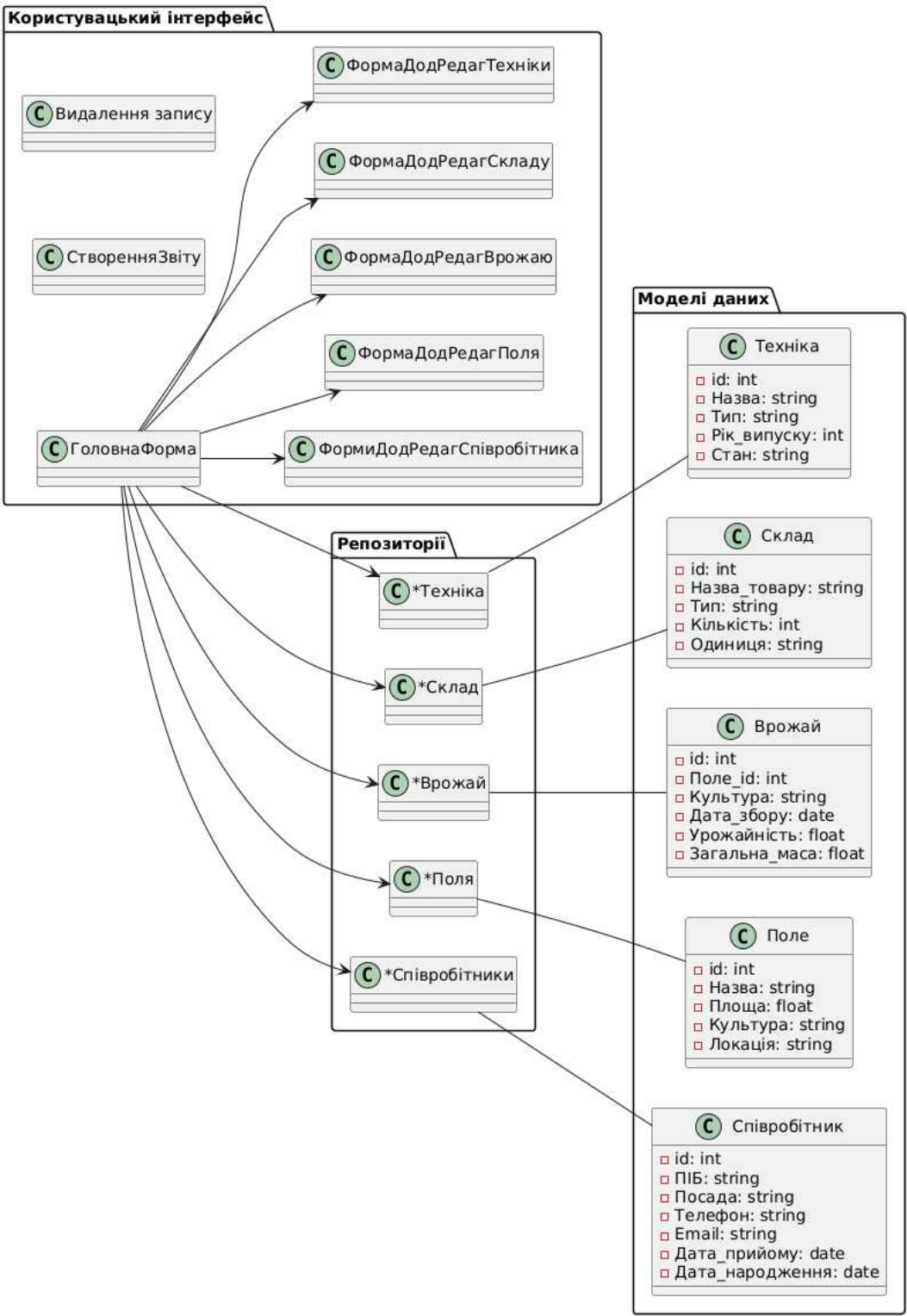


Рисунок 3.1 – Функціональна схема створюваної інформаційної системи

Створена функціональна схема інформаційної системи для обліку роботи аграрної компанії структуровано відображає основні компоненти та взаємозв'язки між ними, що забезпечують повний цикл автоматизації обліку діяльності агропідприємства.

З правої сторони схеми розташовані основні моделі даних, які відповідають ключовим сутностям аграрного бізнесу: "Співробітник", "Поле", "Врожай", "Склад" та "Техніка". Кожна з цих сутностей містить набір характеристик, що відображають реальні властивості відповідних об'єктів у господарстві.

Між моделями даних і користувацьким інтерфейсом розташований шар репозиторіїв. Репозиторії виконують роль посередників між інтерфейсом та базою даних, забезпечуючи реалізацію збереження, оновлення, пошуку та видалення інформації про відповідні сутності.

Користувацький інтерфейс представлений блоками, що відповідають основним функціональним можливостям системи.

Центральне місце в інтерфейсі займає "Головна Форма", яка є основним робочим вікном програми. З неї користувач отримує доступ до всіх функцій: може переходити до додавання, редагування, видалення даних, а також до модуля "Створення Звіту".

Останній модуль відповідає за формування звітної документації за вибраними критеріями, що дозволяє швидко отримувати аналітичну інформацію про діяльність підприємства.

Завдяки такій структурі система є логічно цілісною та зручною для користувача. Всі основні процеси агропідприємства охоплені окремими функціональними блоками, а взаємозв'язки між ними забезпечують ефективну роботу з даними. Схема також дозволяє легко розширювати функціонал у майбутньому — наприклад, додавати нові сутності чи модулі, не порушуючи загальної запропонованої архітектури системи.

3.2 Структура бази даних інформаційної системи для обліку роботи аграрної компанії

Програмне забезпечення, що створюється, має включати в свій склад базу даних, яка надасть змогу зберігати дані та реалізовувати доступ до них. База даних інформаційної системи складається відповідно з п'яти таблиць, що забезпечують функціонування відповідних модулів і виконання ними необхідних для роботи програми завдань:

- а) Fields (Поля);
- б) Equipment (Техніка);
- в) Harvest (Врожай);
- г) Warehouse (Склад);
- д) Employees (Співробітники).

Відомості про наявні в розпорядженні аграрного підприємства поля містяться в таблиці 3.1.

Таблиця 3.1 – Поля

Назва поля	Сутність	Тип даних	Ключове
id	Ідентифікатор	INTEGER	true
name	Назва поля	TEXT	false
area	Площа (га)	REAL	false
crop	Культура	TEXT	false
sowing_date	Дата посіву	TEXT	false
harvest_date	Дата збору врожаю	TEXT	false

Інформація про техніку та обладнання, наявні в розпорядженні агрофірми, внесена в таблицю 3.2.

Таблиця 3.2 – Техніка

Назва поля	Сутність	Тип даних	Ключове
id	Ідентифікатор	INTEGER	true
name	Назва техніки	TEXT	false

model	Марка/модель	TEXT	false
year	Рік випуску	INTEGER	false
condition	Стан	TEXT	false
last_service_date	Дата останнього обслуговування технічного засобу	TEXT	false

Дані про зібраний врожай та дельну інформацію по ньому містить таблиця 3.3.

Таблиця 3.3 – Врожай

Назва поля	Сутність	Тип даних	Ключове
id	Ідентифікатор	INTEGER	true
field_id	Ідентифікатор поля (зовн. ключ)	INTEGER	true
crop	Культура	TEXT	false
harvest_date	Дата збору врожаю	TEXT	false
yield	Урожайність (т/га)	REAL	false
total_weight	Загальний збір (тонн)	REAL	false

Таблиця 3.4 містить інформацію про поточний стан справ на складі агропідприємства як в контексті сільськогосподарських матеріалів (мінеральні добрива, фунгіциди, інсектициди, насіння тощо), так і з боку технічних і організаційних потреб (запчастини, паливно-мастильні матеріали, тара для зберігання продукції).

Таблиця 3.4 – Склад

Назва поля	Сутність	Тип даних	Ключове
id	Ідентифікатор	INTEGER	true

item_name	Найменування матеріалу	TEXT	false
item_type	Тип (насіння, добрива, ПММ тощо)	TEXT	false
quantity	Кількість (залишок)	REAL	false
unit	Одиниця виміру	TEXT	false

В таблиці 3.5 подано дані про робітників аграрної фірми, їх посади, контакти для зв'язку з ними та додаткову інформацію.

Таблиця 3.5 – Співробітники

Назва поля	Сутність	Тип даних	Ключове
id	Ідентифікатор	INTEGER	true
full_name	ПІБ	TEXT	false
position	Посада	TEXT	false
phone	Телефон	TEXT	false
email	Email	TEXT	false
hire_date	Дата прийняття на роботу	TEXT	false

Таким чином, за рахунок реалізації бази даних з зазначеними таблицями буде реалізовано механізми зберігання внесених даних в інформаційну систему для обліку роботи аграрної компанії.

3.3 Алгоритм функціонування інформаційної системи

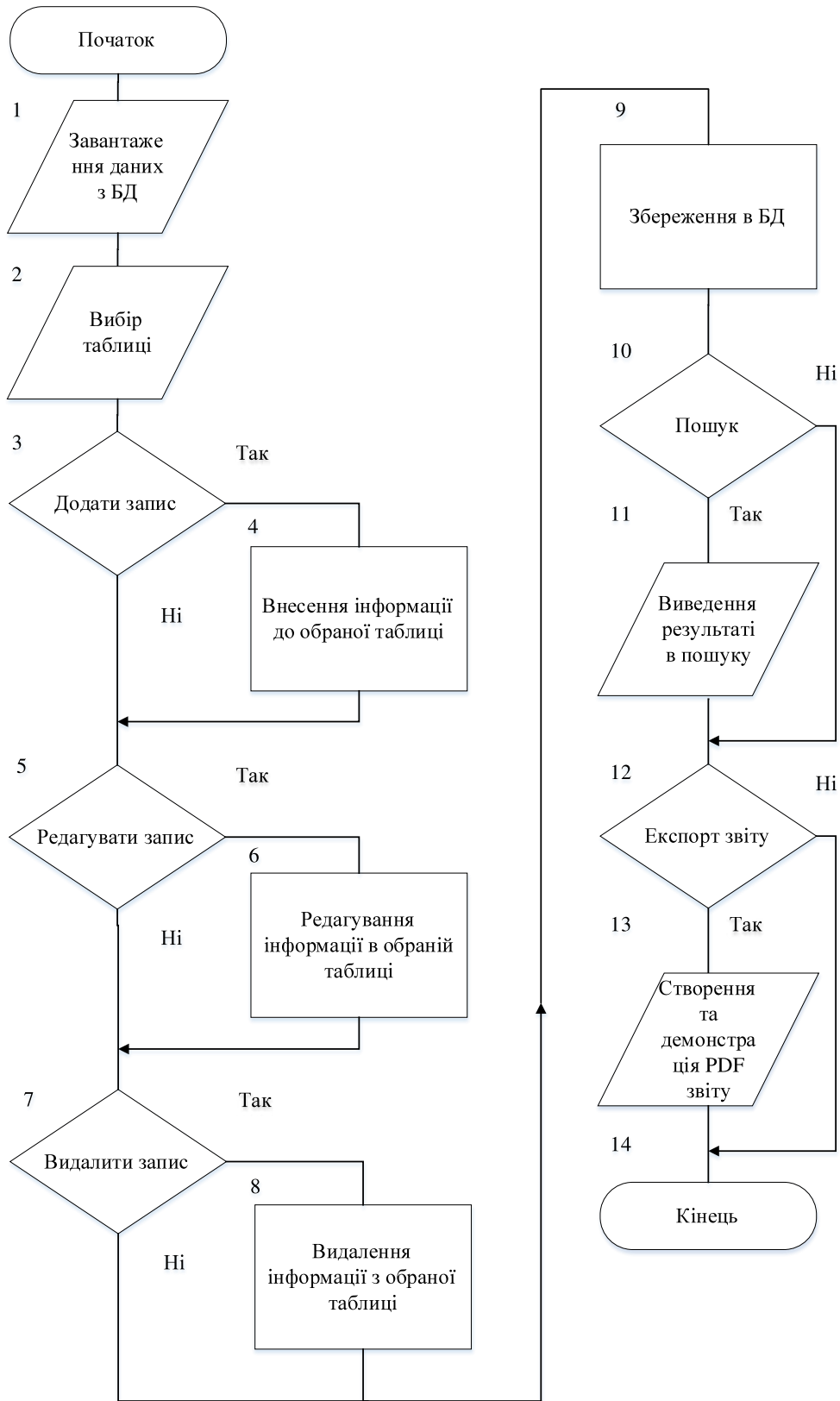


Рисунок 3.2 – Алгоритм роботи інформаційної системи для обліку роботи аграрної компанії

На рисунку 3.2 зображено алгоритм функціонування інформаційної системи для обліку роботи аграрної компанії. Після запуску системи відбувається завантаження наявної інформації з бази даних, що дозволяє отримати доступ до всіх наявних записів. Користувач починає роботу з вибору потрібної таблиці, наприклад, це може бути облік співробітників, полів, врожаю, складу чи техніки.

Далі система надає можливість виконати одну з ключових дій: додати новий запис в таблицю, відредагувати вже існуючий або видалити непотрібний. У разі додавання нового запису відкривається форма для введення даних, після чого інформація зберігається у відповідній таблиці. Якщо необхідно оновити вже наявну інформацію, користувач обирає потрібний запис, відкривається вікно редагування з відповідними даними таблиці полями, він вносить зміни, і система знову фіксує оновлені дані у базі. Аналогічно, при необхідності видалення, обраний запис видаляється з таблиці, що дозволяє підтримувати актуальність інформації.

Після виконання будь-якої з цих операцій система образу оновлює базу даних, забезпечує збереження змін у ній, та гарантує цілісність і актуальність наявної в ній інформації. Окрім роботи з окремими записами, користувач може скористатися функцією пошуку для знаходження потрібної інформації за заданими критеріями. Результати пошуку відображаються одразу у поточній таблиці, що дозволяє швидко аналізувати отримані дані без перемикання на інші вікна.

Додатково передбачена можливість експорту інформації у вигляді звіту. Якщо користувач бажає сформувати звіт, система генерує його у форматі PDF, який можна одразу переглянути, зберегти або роздрукувати.

Завдяки такій організації взаємодії з інформаційною системою для обліку роботи аграрної компанії та використанню вищеописаного принципу її функціонування, робота з програмним продуктом є інтуїтивно зрозумілою і дозволяє співробітникам ефективно виконувати всі основні завдання аграрного підприємства.

3.4 Програмна реалізація інформаційної системи для обліку роботи аграрної компанії

В даному підрозділі описана функціональність створеної інформаційної системи та наведені скріншоти екранних форм, що демонструють основні принципи функціонування додатку. Тому даний підрозділ можна одночасно розглядати і як інструкцію користувача.

Після запуску користувач потрапляє у головне вікно, де у верхній частині розташовано панель навігації. Ця панель дозволяє швидко перемикатися між основними розділами системи: облік полів, техніки, врожаю, складу та співробітників. Завдяки такій структурі користувач завжди має під рукою доступ до даних з будь-якого аспекту діяльності підприємства. Головне вікно програми наведено на рисунку 3.3.

	Id	Назва	Площа	Культура	Дата посіву	Дата збору
	10	Поле Сонячне	24,5	Пшениця	15 березня 2024	10 серпня 2024
	11	Долина Миру	15,2	Кукурудза	01 квітня 2024	05 вересня 2024
	12	Зелений Луг	30	Соняшник	20 березня 2024	25 серпня 2024
▶	13	Вишневий Сад	12,8	Ячмінь	12 березня 2024	30 липня 2024
	14	Східний Масив	18,7	Ріпак	25 березня 2024	15 серпня 2024
	15	Зоряний Край	22,1	Соя	05 квітня 2024	10 вересня 2024
	16	Лісова Галявина	16,4	Гречка	28 березня 2024	20 серпня 2024
	17	Річковий Берег	20,3	Овес	10 квітня 2024	30 серпня 2024
	18	Західне Поле	25	Пшениця	18 березня 2024	12 серпня 2024
	19	Південна Долина	14,9	Кукурудза	03 квітня 2024	07 вересня 2024
	20	Пшеничний Лан	27,6	Пшениця	22 березня 2024	18 серпня 2024
	21	Ясний Ранок	19,5	Соняшник	27 березня 2024	26 серпня 2024
	22	Сонячна Долина	23,2	Ріпак	30 березня 2024	22 серпня 2024
	23	Гірська Галявина	11,7	Гречка	07 квітня 2024	12 вересня 2024
	24	Луг Веселий	13,8	Овес	12 квітня 2024	28 серпня 2024
	25	Північний Масив	21,4	Соя	08 квітня 2024	15 вересня 2024

Пошук:

Рисунок 3.3 – Головне вікно інформаційної системи для обліку роботи аграрної компанії після запуску

Центральне місце інтерфейсу займає таблиця, яка відображає дані відповідного розділу. Стовпці таблиці мають заголовки, які виділені кольором для кращої читабельності. Користувач може сортувати дані за будь-яким стовпцем кліком лівої клавіші миші по його заголовку, що суттєво полегшує аналіз таблиць з великою кількістю записів.

Для виконання основних операцій із записами передбачені CRUD-кнопки (додавання, редагування та видалення записів у базі даних). Вони розташовані так, щоб бути завжди під рукою, і відкривають відповідні діалогові вікна для введення або коригування інформації (рисунок 3.4).

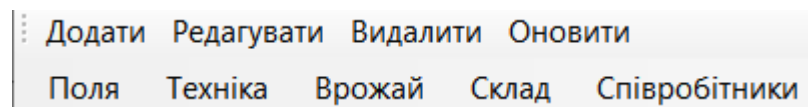


Рисунок 3.4 – CRUD-кнопки інформаційної системи

Форми введення та редагування даних спроектовані з урахуванням юзабіліті та використовують додаткові елементи для зручності – календарі, комбобокси та лічильники. Вони наведені на рисунку 3.5.

Рисунок 3.5 – Форми додавання та редагування полів таблиць аграрної фірми

Під таблицею розташована панель фільтрів — тут можна знайти потрібні дані відповідно до різних критеріїв, наприклад, обрати культуру,

посаду чи діапазон дат. Приділено увагу в програмі також і експорту інформації. Користувач може у будь-який момент експортувати поточну таблицю у формат PDF, причому у фінальному документі зберігаються всі застосовані фільтри, а підписи стовпців залишаються українською мовою. Для коректного відображення кирилиці у PDF використовується спеціальний шрифт, що гарантує якість друкованих звітів.

На рисунку 3.6 наведено панель пошуку та експорту звітів.

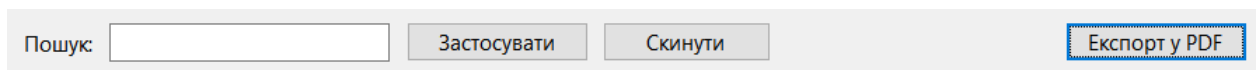


Рисунок 3.6 – Панель пошуку та експорту звітів

Рисунок 3.7, відповідно демонструє таблицю обліку техніки аграрного підприємства.

Інформаційна система аграрної компанії						
Додати Редагувати Видалити Оновити						
Поля Техніка Врожай Склад Співробітники						
	Id	Назва	Марка/модель	Рік випуску	Стан	Дата
	6	Трактор	John Deere 6135B	2019	Відмінний	15 лютого 2024
	7	Комбайн	Case IH Axial-Flow	2020	Добрий	10 березня 2024
	8	Сівалка	Amazone D9	2018	Задовільний	20 листопада 2023
▶	9	Обприскувач	Berthoud Tracker	2021	Відмінний	18 січня 2024
	10	Плуг	Lemken Juwel	2017	Добрий	05 жовтня 2023
	11	Культиватор	Horsch Tiger	2016	Потребує ремон...	14 вересня 2023
	12	Борона	KUHN Discover	2019	Добрий	28 лютого 2024
	13	Зерновоз	MAN TGS	2020	Відмінний	05 березня 2024
	14	Трактор	MT3-82	2015	Задовільний	12 грудня 2023
	15	Комбайн	New Holland CX5	2021	Відмінний	12 березня 2024
	16	Сівалка	Great Plains 3P60...	2018	Добрий	25 листопада 2023
	17	Обприскувач	John Deere R4030	2019	Добрий	30 січня 2024
	18	Плуг	Gregoire Besson	2016	Потребує ремон...	18 жовтня 2023
	19	Культиватор	Case IH Tiger-Mate	2017	Добрий	20 вересня 2023
	20	Борона	Vaderstad Carrier	2020	Відмінний	22 лютого 2024
	21	Зерновоз	DAF CF	2018	Задовільний	20 грудня 2023

Рисунок 3.7 – Таблиця «Техніка»

Таким чином, інформація про всю сільськогосподарську техніку зберігається у відповідній таблиці. Користувачі мають змогу вести перелік машин і контролювати життєвий цикл кожної одиниці техніки від моменту придбання до списання. Приділяється увага обліку технічного стану та датам останнього обслуговування, що дозволяє своєчасно проводити ремонтні роботи та мінімізувати ризики простоїв.

Таблиця «Врожай» створена для систематизації результатів збору продукції сільськогосподарською компанією. Тут фіксуються дані про те, з якого саме поля і яку культуру було зібрано в ході робіт, коли відбувався збір, яка була урожайність і загальна вага зібраного урожаю. Це надає можливість переглядати статистику за різними періодами, порівнювати урожайність по культурах чи ділянках. Зазначена таблиця представлена на скріншоті (рисунок 3.8).

	Id	Назва поля	Культура	Дата збору	Урожайність (т/га)	Загальна вага (т)
▶	1	Поле Західне1	Кукурудза	10 серпня 2025	5,19	65,0
	2	Поле Східне	Кукурудза	05 вересня 2025	7,1	59,0
	3	Поле Південне	Соняшник	20 вересня 2025	2,8	42,0
	5	Поле Сонячне	Пшениця	10 серпня 2024	6,2	151,9
	6	Долина Миру	Кукурудза	05 вересня 2024	8,1	123,1
	7	Зелений Луг	Соняшник	25 серпня 2024	2,9	87,0
	8	Вишневий Сад	Ячмінь	30 липня 2024	5,5	70,4
	9	Східний Масив	Ріпак	15 серпня 2024	3,4	63,6
	10	Зоряний Край	Соя	10 вересня 2024	2,7	59,7
	11	Лісова Галявина	Гречка	20 серпня 2024	1,8	29,5
	12	Річковий Берег	Овес	30 серпня 2024	3,2	64,9
	13	Західне Поле	Пшениця	12 серпня 2024	6,0	150,0
	14	Південна Долина	Кукурудза	07 вересня 2024	8,3	123,7
	15	Пшеничний Лан	Пшениця	18 серпня 2024	6,4	176,6
	16	Ясний Ранок	Соняшник	26 серпня 2024	3,0	58,5
	17	Сонячна Долина	Ріпак	22 серпня 2024	3,6	83,5

Рисунок 3.8 – Таблиця «Врожай»

Облік матеріальних ресурсів здійснюється через роботу з таблицею «Склад». На відміну від інших розділів, акцент зроблено на контролі залишків за для реалізації можливості своєчасного поповнення запасів. Програма дозволяє швидко дізнатися, скільки матеріалів є у наявності, та які з них потребують закупівлі. Це дає змогу уникати перебоїв у забезпеченні виробничого процесу (рисунок 3.9).

Інформаційна система аграрної компанії

Додати Редагувати Видалити Оновити

Поля Техніка Врожай **Склад** Співробітники

	Id	Назва матеріалу	Тип матеріалу	Кількість	Од. виміру
▶	1	Добрива NPK	ЗЗР (гербіцид)	1200,0	кг
	2	Добрива NPK	Запасні частини	501,0	шт
	3	Дизельне паливо	ПММ	800,0	л
	5	Насіння пшениці	Насіння	1200,0	кг
	6	Карбамід	Добриво	800,0	кг
	7	Гербіцид Гранстар	ЗЗР (гербіцид)	50,0	л
	8	Масило Shell	Масильні матеріали	200,0	л
	9	Дизельне паливо	Паливо	1500,0	л
	10	Запасні частини до трак...	Запасні частини	30,0	шт
	11	Тара для зерна	Тара	100,0	шт
	12	Насіння соняшника	Насіння	900,0	кг
	13	Кас-32	Добриво	600,0	л
	14	Фунгіцид Тілт	ЗЗР (фунгіцид)	40,0	л
	15	Гербіцид Пріма	ЗЗР (гербіцид)	60,0	л
	16	Інсектицид Актара	ЗЗР (інсектицид)	25,0	л
	17	Масило Total	Масильні матеріали	180,0	л

Пошук: Застосувати Скинути Експорт у PDF

Рисунок 3.9 – Таблиця «Склад»

Кадровий облік у програмі реалізовано через таблицю «Співробітники». Вона містить контактні дані й посади працівників і дозволяє фіксувати дати прийому на роботу і планувати розподіл обов'язків. Завдяки цьому керівництво аграрної фірми має під рукою актуальні дані про кадровий потенціал свого підприємства та має змогу ефективно управляти людськими ресурсами (рисунок 3.10).

Інформаційна система аграрної компанії							
Додати Редагувати Видалити Оновити							
Поля Техніка Врожай Склад Співробітники							
	Id	ПІБ	Посада	Телефон	Email	Дата	Дата
	1	Іваненко Іван ...	Агроном	+380502234568	ivanenko@agr...	10 березня 20...	21 квітня 1993
▶	2	Петренко Пет...	Механік	+380671234567	petrenko@agr...	22 травня 2021	02 червня 2002
	3	Сидоренко Си...	Водій	+380931234569	syderenko@ag...	15 січня 2023	21 квітня 1995
	5	Іваненко Оле...	Агроном	+380501234567	ivanenko@agr...	01 березня 20...	15 липня 1985
	6	Петренко Ма...	Бухгалтер	+380671112233	petrenko@agr...	12 квітня 2021	22 березня 19...
	7	Сидоренко А...	Механізатор	+380931234567	sydorenko@ag...	20 травня 2020	10 вересня 19...
	8	Коваленко Ол...	Менеджер	+380991234567	kovalenko@ag...	15 січня 2023	05 грудня 1993
	9	Гриценко Віта...	Водій	+380671234567	hrytsenko@agr...	10 серпня 2019	18 червня 1979
	10	Мельник Ната...	Агроном	+380661234567	melnyk@agro...	01 вересня 20...	30 листопада ...
	11	Ткаченко Сер...	Механізатор	+380501112233	tkachenko@ag...	25 жовтня 2020	22 квітня 1983
	12	Шевченко Ок...	Бухгалтер	+380731234567	shevchenko@a...	05 липня 2021	14 лютого 1991
	13	Демченко Ол...	Водій	+380991112233	demchenko@a...	18 грудня 2018	09 жовтня 1980
	14	Кравченко Ю...	Менеджер	+380931112233	kravchenko@a...	10 квітня 2023	27 серпня 1995
	15	Лисенко Вікто...	Агроном	+380671234568	lysenko@agro...	15 червня 2022	19 січня 1987
	16	Бондаренко Т...	Бухгалтер	+380501234568	bondarenko@...	22 вересня 20...	11 травня 1992
	17	Захарченко П...	Механізатор	+380661234568	zakharchenko...	07 березня 20...	16 березня 19...

Пошук:
Застосувати
Скинути
Експорт у PDF

Рисунок 3.10 – Таблиця «Співробітники»

Як зазначалося вище, інформаційна система дає користувачеві змогу формувати та зберігати вибрані дані у вигляді PDF-документів. Технічно ця можливість реалізована за допомогою бібліотеки iTextSharp, яка забезпечує генерацію PDF-файлів із підтримкою української мови та коректним відображенням кирилиці. Приклад сформованого звіту наведено на рисунку 3.11.

Загалом, інтерфейс інформаційної системи для обліку роботи аграрної компанії забезпечує простий і швидкий доступ до всіх функцій системи, дозволяє ефективно керувати аграрними ресурсами, вести облік персоналу, техніки, врожаю та складу, а також формувати звіти для подальшого аналізу та прийняття управлінських рішень. Елементи інтерфейсу гармонійно поєднані, що робить роботу з програмою приємною та простою навіть для користувачів з невеликим досвідом взаємодії з комп'ютером.

Експорт даних: Employees

ID	ПІБ	Посада	Телефон	Email	Дата прийняття	Дата народження
1	Іваненко Іван Іванович	Агроном	+380502234568	ivanenko@agro.ua	10 березня 2022	21 квітня 1993
2	Петренко Петро Петрович	Механік	+380671234567	petrenko@agro.ua	22 травня 2021	02 червня 2002
3	Сидоренко Сидір Сидорович	Водій	+380931234569	syderenko@agro.ua	15 січня 2023	21 квітня 1995
5	Іваненко Олександр Сергійович	Агроном	+380501234567	ivanenko@agro.ua	01 березня 2022	15 липня 1985
6	Петренко Марія Василівна	Бухгалтер	+380671112233	petrenko@agro.ua	12 квітня 2021	22 березня 1990
7	Сидоренко Андрій Миколайович	Механізатор	+380931234567	sydorenko@agro.ua	20 травня 2020	10 вересня 1982

Рисунок 3.11 – Фрагмент експортованого звіту

Таким чином, розроблена інформаційна система для обліку роботи аграрної компанії належним чином реалізує вирішення поставлених під час проектування завдань.

ВИСНОВКИ

У процесі розробки інформаційної системи для обліку роботи аграрної компанії було реалізовано сучасний додаток для автоматизації обліку сільськогосподарського підприємства. Розроблена програма дозволяє ефективно управляти даними про співробітників, поля, техніку, врожай та складські ресурси, що значно підвищує прозорість і оперативність управлінських процесів.

В ході роботи було створено інтуїтивно зрозумілий інтерфейс, який забезпечує швидкий доступ до основних функцій системи. Особлива увага приділялася зручності користування: всі елементи інтерфейсу мають українські підписи, реалізовано кольорове виділення заголовків таблиць, підтримується сортування та фільтрація даних, а також експорт у PDF із коректним відображенням кирилиці. Це робить програму доступною для широкого кола користувачів, незалежно від їхнього рівня підготовки.

З технічної точки зору, було розроблено надійну структуру бази даних з урахуванням зв'язків між сутностями, що забезпечує цілісність та узгодженість даних.

В результаті виконаної роботи вдалося створити універсальний інструмент, який може бути використаний як у невеликих фермерських господарствах, так і на великих агрохолдингах. Програма легко масштабується та може бути доповнена новими модулями відповідно до потреб підприємства.

Загалом, розробка інформаційної системи для обліку роботи аграрної компанії підтвердила доцільність використання сучасних інформаційних технологій для оптимізації аграрного бізнесу, а також продемонструвала можливість створення якісного програмного продукту з урахуванням вимог кінцевого користувача.

ПЕРЕЛІК ПОСИЛАНЬ

1. Mykhailenko O., Khilchenko I. Agricultural Sector of Ukraine: the Modern State, Problems and Prospects of Reform in the Field. *Modern Economics*. 2019. Vol. 15, no. 1. P. 148–152. URL: [https://doi.org/10.31521/modecon.v15\(2019\)-21](https://doi.org/10.31521/modecon.v15(2019)-21) (date of access: 08.02.2025).
2. Солоп А. П. Аграрний сектор України: сучасні виклики та можливості. *Таврійський науковий вісник. Серія: Економіка*. 2024. № 20. С. 147–158. URL: <https://doi.org/10.32782/2708-0366/2024.20.17> (дата звернення: 08.02.2025).
3. Площі, валові збори та урожайність сільськогосподарських культур за їх видами та по регіонах у 2024 році [Електронний ресурс] – Режим доступу до ресурсу: <https://www.ukrstat.gov.ua/>
4. Український клуб аграрного бізнесу [Електронний ресурс] – Режим доступу до ресурсу: <https://ucab.ua/ua>
5. Ковбаса О. Компоненти стратегічного розвитку аграрного підприємництва в Україні. *Економіка та суспільство*. 2024. № 67. URL: <https://doi.org/10.32782/2524-0072/2024-67-123> (дата звернення: 09.02.2025).
6. Беженар І. М., Грищенко О. Ю. Фермерські господарства в Україні: стан та перспективи розвитку. *Проблеми сучасних трансформацій. Серія: економіка та управління*. 2023. № 9. URL: <https://doi.org/10.54929/2786-5738-2023-9-04-14> (дата звернення: 10.02.2025).
7. Чернявський І. Ю. Стратегічні орієнтири повоєнного відновлення економіки підприємств аграрної сфери. *Зелена економіка та низьковуглецевий розвиток: дорожня карта для України*. 2024. URL: <https://doi.org/10.36059/978-966-397-448-4-28> (дата звернення: 11.02.2025).
8. Бойко Л. О. Виклики та проблеми фермерських господарств у період невизначеності. *Проблеми сучасних трансформацій. Серія: економіка та управління*. 2024. № 11. URL: <https://doi.org/10.54929/2786-5738-2024-11-04-02> (дата звернення: 13.02.2025).

9. AgriWebb. Livestock management software that moves your farm forward [Электронный ресурс] – Режим доступа до ресурсу: <https://www.agriwebb.com/>
10. Trimble Ag Software [Электронный ресурс] – Режим доступа до ресурсу: <https://ptxtrimble.com/en/products/software/trimble-agriculture-software>
11. Agworld. Agriculture's preferred independent farm information management ecosystem [Электронный ресурс] – Режим доступа до ресурсу: <https://www.agworld.com/us/>
12. Adobe Acrobat Reader [Электронный ресурс] – Режим доступа до ресурсу: <https://get.adobe.com/ru/reader/>
13. .NET_Framework [Электронный ресурс] – Режим доступа до ресурсу: <https://dotnet.microsoft.com/>
14. Albahari B., Albahari J. C# 7.0 in a Nutshell: The Definitive Reference. O'Reilly Media, 2017. 1088 p.
15. Visual Studio 2022 [Электронный ресурс] – Режим доступа до ресурсу: <https://visualstudio.microsoft.com/>
16. IntelliSense in Visual Studio [Электронный ресурс] – Режим доступа до ресурсу: <https://learn.microsoft.com/ru-ru/visualstudio/ide/using-intellisense?view=vs-2022>

Додаток А

Фрагмент програмного коду

AgroHoldingMainForm.cs

```
using System;
using System;
using System.Collections.Generic;
using System.Globalization;
using System.Linq;
using System.Windows.Forms;
using iTextSharp.text;
using iTextSharp.text.pdf;
using System.IO;

namespace AgroHolding
{
    public partial class AgroHoldingMainForm : Form
    {
        private MenuStrip mainMenu;
        private DataGridView mainGrid;
        private ToolStrip mainToolStrip;
        private string currentSection = "Fields"; // Відстеження
активного розділу

        // --- Елементи фільтрації
        private Panel panelFilters;
        private TextBox textSearch;
        private Button buttonApply;
        private Button buttonReset;

        public AgroHoldingMainForm()
        {
            this.Text = "Інформаційна система аграрної компанії";
            this.ClientSize = new System.Drawing.Size(900, 600);

            // Ініціалізація бази та тестових даних
            DatabaseInitializer.InitializeDatabase();
            DatabaseInitializer.SeedTestData();

            // Створення інтерфейсу
            CreateMainMenu();
            CreateToolStrip();
            CreateMainGrid();
            CreateFilterPanel();
        }

        // Створення головного меню
        private void CreateMainMenu()
        {

```

```

        mainMenu = new MenuStrip();

        var itemFields = new ToolStripMenuItem("Поля");
        itemFields.Click += (s, e) => LoadFields();
        mainMenu.Items.Add(itemFields);

        var itemEquipment = new ToolStripMenuItem("Техніка");
        itemEquipment.Click += (s, e) => LoadEquipment();
        mainMenu.Items.Add(itemEquipment);

        var itemHarvest = new ToolStripMenuItem("Врожай");
        itemHarvest.Click += (s, e) => LoadHarvest();
        mainMenu.Items.Add(itemHarvest);

        var itemWarehouse = new ToolStripMenuItem("Склад");
        itemWarehouse.Click += (s, e) => LoadWarehouse();
        mainMenu.Items.Add(itemWarehouse);

        var itemEmployees = new ToolStripMenuItem("Співробітники");
        itemEmployees.Click += (s, e) => LoadEmployees();
        mainMenu.Items.Add(itemEmployees);

        //var itemReports = new ToolStripMenuItem("Звіти");
        //itemReports.Click += (s, e) => ShowReports();
        //mainMenu.Items.Add(itemReports);

        this.MainMenuStrip = mainMenu;
        this.Controls.Add(mainMenu);
    }

    // Створення панелі інструментів
    private void CreateToolStrip()
    {
        mainToolStrip = new ToolStrip();
        mainToolStrip.Items.Add("Додати").Click += (s, e) =>
AddRecord();
        mainToolStrip.Items.Add("Редагувати").Click += (s, e)
=> EditRecord();
        mainToolStrip.Items.Add("Видалити").Click += (s, e)
=> DeleteRecord();
        mainToolStrip.Items.Add("Оновити").Click += (s, e) =>
RefreshData();

        mainToolStrip.Dock = DockStyle.Top;
        this.Controls.Add(mainToolStrip);
    }

    // Створення таблиці для відображення даних
    private void CreateMainGrid()
    {
        mainGrid = new DataGridView();
    }

```



```

        mainGrid.Dock = DockStyle.Fill;
        mainGrid.ReadOnly = true;
        mainGrid.SelectionMode
DataGridViewSelectionMode.FullRowSelect;
        mainGrid.AutoSizeColumnsMode
DataGridViewAutoSizeColumnsMode.Fill;

        //сортування
        mainGrid.ColumnHeadersDefaultCellStyle.Alignment
DataGridViewContentAlignment.MiddleCenter;
        mainGrid.AllowUserToOrderColumns = true;
        mainGrid.ColumnHeaderMouseClick
MainGrid_ColumnHeaderMouseClick;

        mainGrid.EnableHeadersVisualStyles = false;
        mainGrid.ColumnHeadersDefaultCellStyle.BackColor
Color.LightSteelBlue;
        mainGrid.ColumnHeadersDefaultCellStyle.ForeColor
Color.DarkBlue;
        mainGrid.ColumnHeadersDefaultCellStyle.Alignment
DataGridViewContentAlignment.MiddleCenter;

        this.Controls.Add(mainGrid);
        this.Controls.SetChildIndex(mainGrid, 0);

        // Завантажити поля за замовчуванням
        LoadFields();
    }

    //сортування
    private string lastSortColumn = null;
    private bool lastSortAscending = true;

    private void MainGrid_ColumnHeaderMouseClick(object
sender, DataGridViewCellEventArgs e)
    {
        string
columnName
mainGrid.Columns[e.ColumnIndex].DataPropertyName;

        // Получаем текущий источник данных
        if (mainGrid.DataSource is IEnumerable<object> data)
        {
            bool ascending = true;
            if (lastSortColumn == columnName)
                ascending = !lastSortAscending;

            var sorted = ascending
                ? data.OrderBy(x
x.GetType().GetProperty(columnName).GetValue(x, null)).ToList()
                : data.OrderByDescending(x
x.GetType().GetProperty(columnName).GetValue(x, null)).ToList();
=>
=>

```

```

        mainGrid.DataSource = null;
        mainGrid.DataSource = sorted;
        RenameColumnsForCurrentSection();

        lastSortColumn = columnName;
        lastSortAscending = ascending;
    }
}

// панель фільтрації
private void CreateFilterPanel()
{
    panelFilters = new Panel()
    {
        Height = 50,
        Dock = DockStyle.Bottom
    };

    // "Пошук:"
    var labelSearch = new Label()
    {
        Text = "Пошук:",
        Left = 10,
        Top = 13,
        Width = 60,
        Height = 24,
        TextAlign =
System.Drawing.ContentAlignment.MiddleLeft
    };
    panelFilters.Controls.Add(labelSearch);

    textSearch = new TextBox()
    {
        Name = "textSearch",
        Left = labelSearch.Right + 5, // справа от лейбла
        Top = 11,
        Width = 200,
        Anchor = AnchorStyles.Left | AnchorStyles.Top
    };
    panelFilters.Controls.Add(textSearch);

    buttonApply = new Button()
    {
        Text = "Застосувати фільтр",
        Left = textSearch.Right + 12,
        Top = 10,
        Width = 130,
        Height = 28,
        Anchor = AnchorStyles.Left | AnchorStyles.Top
    };
    buttonApply.Click += ButtonApply_Click;
}

```

```

panelFilters.Controls.Add(buttonApply);

buttonReset = new Button()
{
    Text = "Скинути фільтр",
    Left = buttonApply.Right + 10,
    Top = 10,
    Width = 120,
    Height = 28,
    Anchor = AnchorStyles.Left | AnchorStyles.Top
};
buttonReset.Click += ButtonReset_Click;
panelFilters.Controls.Add(buttonReset);

// Кнопка "Експорт у PDF"
var buttonExportPdf = new Button()
{
    Text = "Експорт у PDF",
    Left = buttonReset.Right + 210,
    Top = 10,
    Width = 130,
    Height = 28,
    Anchor = AnchorStyles.Left | AnchorStyles.Top
};
buttonExportPdf.Click += ButtonExportPdf_Click;
panelFilters.Controls.Add(buttonExportPdf);

this.Controls.Add(panelFilters);
}

```

```

private void ExportToPdf(List<object> data, string
filePath)
{
    string fontPath =
Path.Combine(Application.StartupPath, "Fonts", "DejaVuSans.ttf");
    BaseFont baseFont = BaseFont.CreateFont(fontPath,
BaseFont.IDENTITY_H, BaseFont.EMBEDDED);
    var font = new iTextSharp.text.Font(baseFont, 11);
    var fontBold = new iTextSharp.text.Font(baseFont, 11);

    using (var fs = new FileStream(filePath,
FileMode.Create, FileAccess.Write, FileShare.None))
        using (var doc = new Document(PageSize.A4, 10, 10, 20,
20))
            using (var writer = PdfWriter.GetInstance(doc, fs))
            {
                doc.Open();

                var titleFont = new
iTextSharp.text.Font(baseFont, 16);

```

```

        doc.Add(new Paragraph("Экспорт даних: " +
currentSection, titleFont));
        doc.Add(new Paragraph(" "));

        var props = data[0].GetType().GetProperties();
        PdfPTable table = new PdfPTable(props.Length);
        table.WidthPercentage = 100;

        // Заголовки стовпців
        var headers = GetColumnHeaders();
        foreach (var prop in props)
        {
            string header =
headers.ContainsKey(prop.Name) ? headers[prop.Name] : prop.Name;
            table.AddCell(new Phrase(header, fontBold));
        }

        // Дані
        foreach (var item in data)
        {
            foreach (var prop in props)
            {
                var value = prop.GetValue(item);
                table.AddCell(new
Phrase(value?.ToString() ?? "", font));
            }
        }

        doc.Add(table);
        doc.Close();
    }

    // Фільтрація
    private void ButtonApply_Click(object sender, EventArgs e)
    {
        string search = textSearch.Text.Trim().ToLower();

        // Работает только если источник данных — List<object>
        if (mainGrid.DataSource is IEnumerable<object>
enumerable)
        {
            var filtered = enumerable
                .Where(item => ItemContainsSearch(item,
search))
                .ToList();
            mainGrid.DataSource = null;
            mainGrid.DataSource = filtered;
            RenameColumnsForCurrentSection();
        }
    }

```

```

private void ButtonReset_Click(object sender, EventArgs e)
{
    textSearch.Text = "";
    RefreshData();
}

private void ButtonExportPdf_Click(object sender,
EventArgs e)
{
    // Отримуємо поточний відфільтрований список
    var exportList = mainGrid.DataSource as
IEnumerable<object>;
    if (exportList == null || !exportList.Any())
    {
        MessageBox.Show("Немає даних для експорту.",
"Експорт у PDF", MessageBoxButtons.OK,
MessageBoxIcon.Information);
        return;
    }

    using (var sfd = new SaveFileDialog())
    {
        Filter = "PDF файли (*.pdf)|*.pdf",
        Title = "Зберегти як PDF",
        FileName =
$" {currentSection}_{DateTime.Now:yyyyMMdd_HH:mm}.pdf"
    })
    {
        if (sfd.ShowDialog() == DialogResult.OK)
        {
            try
            {
                ExportToPdf(exportList.ToList(),
sfd.FileName);
                MessageBox.Show("Експорт завершено
успішно!", "Експорт у PDF", MessageBoxButtons.OK,
MessageBoxIcon.Information);
            }
            catch (Exception ex)
            {
                MessageBox.Show("Помилка експорту: " +
ex.Message, "Експорт у PDF", MessageBoxButtons.OK,
MessageBoxIcon.Error);
            }
        }
    }

private bool ItemContainsSearch(object item, string
search)
{
    if (string.IsNullOrEmpty(search)) return true;

```

```

        if (item == null) return false;
        foreach (var prop in item.GetType().GetProperties())
        {
            var value = prop.GetValue(item);
            if (value != null &&
value.ToString().ToLower().Contains(search))
                return true;
        }
        return false;
    }

// Завантаження даних з таблиці Fields з форматуванням дат
private void LoadFields()
{
    currentSection = "Fields";
    var fields = FieldRepository.GetAllFields();

    var displayList = new List<object>();
    foreach (var f in fields)
    {
        displayList.Add(new
        {
            f.Id,
            Name = f.Name,
            Area = f.Area,
            Crop = f.Crop,
            SowingDate = FormatDate(f.SowingDate),
            HarvestDate = FormatDate(f.HarvestDate)
        });
    }

    mainGrid.DataSource = displayList;
    RenameColumnsForCurrentSection();
}

private void LoadEquipment()
{
    currentSection = "Equipment";
    var equipment =
EquipmentRepository.GetAllEquipment();

    var displayList = new List<object>();
    foreach (var eq in equipment)
    {
        displayList.Add(new
        {
            eq.Id,
            Name = eq.Name,
            Model = eq.Model,
            Year = eq.Year,
            Condition = eq.Condition,

```

```

        LastServiceDate
FormatDate(eq.LastServiceDate)
    });
    }
    mainGrid.DataSource = displayList;
    RenameColumnsForCurrentSection();
}

private void LoadHarvest()
{
    currentSection = "Harvest";
    var harvests
HarvestRepository.GetAllHarvestDisplay();

    var displayList = new List<object>();
    foreach (var h in harvests)
    {
        displayList.Add(new
        {
            h.Id,
            FieldName = h.FieldName,
            Crop = h.Crop,
            HarvestDate = FormatDate(h.HarvestDate),
            Yield = h.Yield,
            TotalWeight = h.TotalWeight
        });
    }
    mainGrid.DataSource = displayList;
    RenameColumnsForCurrentSection();
}

private void LoadWarehouse()
{
    currentSection = "Warehouse";
    var items = WarehouseRepository.GetAllWarehouse();

    var displayList = new List<object>();
    foreach (var w in items)
    {
        displayList.Add(new
        {
            w.Id,
            ItemName = w.ItemName,
            ItemType = w.ItemType,
            Quantity = w.Quantity,
            Unit = w.Unit
        });
    }
    mainGrid.DataSource = displayList;
    RenameColumnsForCurrentSection();
}

```

```

private void LoadEmployees()
{
    currentSection = "Employees";
    var items = EmployeesRepository.GetAllEmployees();

    var displayList = new List<object>();
    foreach (var e in items)
    {
        displayList.Add(new
        {
            e.Id,
            FullName = e.FullName,
            Position = e.Position,
            Phone = e.Phone,
            Email = e.Email,
            HireDate = FormatDate(e.HireDate),
            DateOfBirth = FormatDate(e.DateOfBirth)
        });
    }
    mainGrid.DataSource = displayList;
    RenameColumnsForCurrentSection();
}

private void RenameColumnsForCurrentSection()
{
    if (mainGrid.Columns == null) return;

    if (currentSection == "Fields")
    {
        if (mainGrid.Columns["Name"] != null)
            mainGrid.Columns["Name"].HeaderText = "Назва";
        if (mainGrid.Columns["Area"] != null)
            mainGrid.Columns["Area"].HeaderText = "Площа";
        if (mainGrid.Columns["Crop"] != null)
            mainGrid.Columns["Crop"].HeaderText = "Культура";
        if (mainGrid.Columns["SowingDate"] != null)
            mainGrid.Columns["SowingDate"].HeaderText = "Дата посіву";
        if (mainGrid.Columns["HarvestDate"] != null)
            mainGrid.Columns["HarvestDate"].HeaderText = "Дата збору";
    }
    else if (currentSection == "Equipment")
    {
        if (mainGrid.Columns["Name"] != null)
            mainGrid.Columns["Name"].HeaderText = "Назва";
        if (mainGrid.Columns["Model"] != null)
            mainGrid.Columns["Model"].HeaderText = "Марка/модель";
        if (mainGrid.Columns["Year"] != null)
            mainGrid.Columns["Year"].HeaderText = "Рік випуску";
        if (mainGrid.Columns["Condition"] != null)
            mainGrid.Columns["Condition"].HeaderText = "Стан";
    }
}

```



```

        if (mainGrid.Columns["LastServiceDate"] != null)
mainGrid.Columns["LastServiceDate"].HeaderText = "Дата
обслуговування";
    }
    else if (currentSection == "Harvest")
    {
        if (mainGrid.Columns["FieldName"] != null)
mainGrid.Columns["FieldName"].HeaderText = "Назва поля";
        if (mainGrid.Columns["Crop"] != null)
mainGrid.Columns["Crop"].HeaderText = "Культура";
        if (mainGrid.Columns["HarvestDate"] != null)
mainGrid.Columns["HarvestDate"].HeaderText = "Дата збору";
        if (mainGrid.Columns["Yield"] != null)
mainGrid.Columns["Yield"].HeaderText = "Урожайність (т/га)";
        if (mainGrid.Columns["TotalWeight"] != null)
mainGrid.Columns["TotalWeight"].HeaderText = "Загальна вага (т)";
    }
    else if (currentSection == "Warehouse")
    {
        if (mainGrid.Columns["ItemName"] != null)
mainGrid.Columns["ItemName"].HeaderText = "Назва матеріалу";
        if (mainGrid.Columns["ItemType"] != null)
mainGrid.Columns["ItemType"].HeaderText = "Тип матеріалу";
        if (mainGrid.Columns["Quantity"] != null)
mainGrid.Columns["Quantity"].HeaderText = "Кількість";
        if (mainGrid.Columns["Unit"] != null)
mainGrid.Columns["Unit"].HeaderText = "Од. виміру";
    }
    else if (currentSection == "Employees")
    {
        if (mainGrid.Columns["FullName"] != null)
mainGrid.Columns["FullName"].HeaderText = "ПІБ";
        if (mainGrid.Columns["Position"] != null)
mainGrid.Columns["Position"].HeaderText = "Посада";
        if (mainGrid.Columns["Phone"] != null)
mainGrid.Columns["Phone"].HeaderText = "Телефон";
        if (mainGrid.Columns["Email"] != null)
mainGrid.Columns["Email"].HeaderText = "Email";
        if (mainGrid.Columns["HireDate"] != null)
mainGrid.Columns["HireDate"].HeaderText = "Дата прийняття";
        if (mainGrid.Columns["DateOfBirth"] != null)
mainGrid.Columns["DateOfBirth"].HeaderText = "Дата народження";
    }
}

// Форматування дати
private string FormatDate(string dateStr)
{
    if (string.IsNullOrEmpty(dateStr)) return "";
    if (DateTime.TryParse(dateStr, out DateTime dt))
        return dt.ToString("dd MMMM yyyy", new
CultureInfo("uk-UA"));
}

```

```

        return dateStr;
    }

    private void ShowReports()
    {
        using (var form = new ReportForm())
        {
            form.ShowDialog();
        }
    }

    // Додати запис для активного розділу
    private void AddRecord()
    {
        if (currentSection == "Fields")
        {
            using (var form = new FieldAddForm())
            {
                if (form.ShowDialog() == DialogResult.OK &&
form.IsAdded)
                {
                    LoadFields();
                }
            }
        }
        else if (currentSection == "Equipment")
        {
            using (var form = new EquipmentAddForm())
            {
                if (form.ShowDialog() == DialogResult.OK &&
form.IsAdded)
                {
                    LoadEquipment();
                }
            }
        }
        else if (currentSection == "Harvest")
        {
            using (var form = new HarvestAddForm())
            {
                if (form.ShowDialog() == DialogResult.OK &&
form.IsAdded)
                {
                    LoadHarvest();
                }
            }
        }
        else if (currentSection == "Warehouse")
        {
            using (var form = new WarehouseAddForm())
            {

```

```

        if (form.ShowDialog() == DialogResult.OK &&
form.IsAdded)
        {
            LoadWarehouse();
        }
    }
    else if (currentSection == "Employees")
    {
        using (var form = new EmployeesAddForm())
        {
            if (form.ShowDialog() == DialogResult.OK &&
form.IsAdded)
            {
                LoadEmployees();
            }
        }
    }
}

private void EditRecord()
{
    if (mainGrid.SelectedRows.Count == 0)
    {
        MessageBox.Show("Будь ласка, виберіть запис для
редагування.", "Редагування", MessageBoxButtons.OK,
MessageBoxIcon.Information);
        return;
    }

    var selected =
mainGrid.SelectedRows[0].DataBoundItem;
    if (selected == null)
        return;

    if (currentSection == "Fields")
    {
        var idProp =
selected.GetType().GetProperty("Id");
        if (idProp == null)
        {
            MessageBox.Show("Не вдалося визначити Id
запису.", "Помилка", MessageBoxButtons.OK, MessageBoxIcon.Error);
            return;
        }
        int fieldId = (int)idProp.GetValue(selected);

        var field =
FieldRepository.GetFieldById(fieldId);
        if (field == null)

```

```

        {
            MessageBox.Show("Не вдалося знайти запис для
редагування.", "Помилка", MessageBoxButtons.OK,
MessageBoxIcon.Error);
            return;
        }

        using (var form = new FieldAddForm(field))
        {
            if (form.ShowDialog() == DialogResult.OK &&
form.IsAdded)
            {
                LoadFields();
            }
        }
    }
    else if (currentSection == "Equipment")
    {
        if (mainGrid.SelectedRows.Count == 0)
        {
            MessageBox.Show("Будь ласка, виберіть запис
для редагування.", "Редагування", MessageBoxButtons.OK,
MessageBoxIcon.Information);
            return;
        }

        selected =
mainGrid.SelectedRows[0].DataBoundItem;
        if (selected == null)
            return;

        var idProp =
selected.GetType().GetProperty("Id");
        if (idProp == null)
        {
            MessageBox.Show("Не вдалося визначити Id
запису.", "Помилка", MessageBoxButtons.OK, MessageBoxIcon.Error);
            return;
        }
        int equipmentId = (int)idProp.GetValue(selected);

        var equipment =
EquipmentRepository.GetEquipmentById(equipmentId);
        if (equipment == null)
        {
            MessageBox.Show("Не вдалося знайти запис для
редагування.", "Помилка", MessageBoxButtons.OK,
MessageBoxIcon.Error);
            return;
        }

        using (var form = new EquipmentAddForm(equipment))

```

```

        {
            if (form.ShowDialog() == DialogResult.OK &&
form.IsAdded)
            {
                LoadEquipment();
            }
        }
    }
    else if (currentSection == "Harvest")
    {
        var idProp =
selected.GetType().GetProperty("Id");
        if (idProp == null)
        {
            MessageBox.Show("Не вдалося визначити Id
запису.", "Помилка", MessageBoxButtons.OK, MessageBoxIcon.Error);
            return;
        }
        int harvestId = (int)idProp.GetValue(selected);

        var harvest =
HarvestRepository.GetHarvestById(harvestId);
        if (harvest == null)
        {
            MessageBox.Show("Не вдалося знайти запис для
редагування.", "Помилка", MessageBoxButtons.OK,
MessageBoxIcon.Error);
            return;
        }

        using (var form = new HarvestAddForm(harvest))
        {
            if (form.ShowDialog() == DialogResult.OK &&
form.IsAdded)
                LoadHarvest();
        }
    }
    else if (currentSection == "Warehouse")
    {
        var idProp =
selected.GetType().GetProperty("Id");
        if (idProp == null)
        {
            MessageBox.Show("Не вдалося визначити Id
запису.", "Помилка", MessageBoxButtons.OK, MessageBoxIcon.Error);
            return;
        }
        int warehouseId = (int)idProp.GetValue(selected);

        var warehouseItem =
WarehouseRepository.GetWarehouseById(warehouseId);
        if (warehouseItem == null)

```

```

        {
            MessageBox.Show("Не вдалося знайти запис для
редагування.", "Помилка", MessageBoxButtons.OK,
MessageBoxIcon.Error);
            return;
        }

        using (var form = new
WarehouseAddForm(warehouseItem))
        {
            if (form.ShowDialog() == DialogResult.OK &&
form.IsAdded)
                LoadWarehouse();
        }
    }
    else if (currentSection == "Employees")
    {
        var idProp =
selected.GetType().GetProperty("Id");
        if (idProp == null)
        {
            MessageBox.Show("Не вдалося визначити Id
запису.", "Помилка", MessageBoxButtons.OK, MessageBoxIcon.Error);
            return;
        }
        int employeeId = (int)idProp.GetValue(selected);

        var employee =
EmployeesRepository.GetEmployeeById(employeeId);
        if (employee == null)
        {
            MessageBox.Show("Не вдалося знайти запис для
редагування.", "Помилка", MessageBoxButtons.OK,
MessageBoxIcon.Error);
            return;
        }

        using (var form = new EmployeesAddForm(employee))
        {
            if (form.ShowDialog() == DialogResult.OK &&
form.IsAdded)
                LoadEmployees();
        }
    }
}

private void DeleteRecord()
{
    // Перевірка, чи вибрано рядок
    if (mainGrid.SelectedRows.Count == 0)
    {

```

```

        MessageBox.Show("Будь ласка, виберіть запис для
видалення.", "Видалення", MessageBoxButtons.OK,
MessageBoxIcon.Information);
        return;
    }

    var selected =
mainGrid.SelectedRows[0].DataBoundItem;
    if (selected == null)
        return;

    // Підтвердження видалення
    var confirm = MessageBox.Show("Ви дійсно бажаєте
видалити цей запис?", "Підтвердження видалення",
MessageBoxButtons.YesNo, MessageBoxIcon.Question);
    if (confirm != DialogResult.Yes)
        return;

    // Визначаємо Id запису
    var idProp = selected.GetType().GetProperty("Id");
    if (idProp == null)
    {
        MessageBox.Show("Не вдалося визначити Id
запису.", "Помилка", MessageBoxButtons.OK, MessageBoxIcon.Error);
        return;
    }
    int id = (int)idProp.GetValue(selected);

    // Видалення відповідно до розділу
    try
    {
        if (currentSection == "Fields")
        {
            FieldRepository.DeleteField(id);
            LoadFields();
        }
        else if (currentSection == "Equipment")
        {
            EquipmentRepository.DeleteEquipment(id);
            LoadEquipment();
        }
        else if (currentSection == "Harvest")
        {
            HarvestRepository.DeleteHarvest(id);
            LoadHarvest();
        }
        else if (currentSection == "Warehouse")
        {
            WarehouseRepository.DeleteWarehouseItem(id);
            LoadWarehouse();
        }
        else if (currentSection == "Employees")

```

```

        {
            EmployeesRepository.DeleteEmployee(id);
            LoadEmployees();
        }
    }
    catch (Exception ex)
    {
        MessageBox.Show("Виникла помилка при видаленні: "
+ ex.Message, "Помилка", MessageBoxButtons.OK,
MessageBoxIcon.Error);
    }
}

private Dictionary<string, string> GetColumnHeaders()
{
    if (currentSection == "Fields")
    {
        return new Dictionary<string, string>
        {
            { "Id", "ID" },
            { "Name", "Назва" },
            { "Area", "Площа" },
            { "Crop", "Культура" },
            { "SowingDate", "Дата посіву" },
            { "HarvestDate", "Дата збору" }
        };
    }
    else if (currentSection == "Equipment")
    {
        return new Dictionary<string, string>
        {
            { "Id", "ID" },
            { "Name", "Назва" },
            { "Model", "Марка/модель" },
            { "Year", "Рік випуску" },
            { "Condition", "Стан" },
            { "LastServiceDate", "Дата обслуговування" }
        };
    }
    else if (currentSection == "Harvest")
    {
        return new Dictionary<string, string>
        {
            { "Id", "ID" },
            { "FieldName", "Назва поля" },
            { "Crop", "Культура" },
            { "HarvestDate", "Дата збору" },
            { "Yield", "Урожайність (т/га)" },
            { "TotalWeight", "Загальна вага (т)" }
        };
    }
    else if (currentSection == "Warehouse")

```



```

        {
            return new Dictionary<string, string>
        {
            { "Id", "ID" },
            { "ItemName", "Назва матеріалу" },
            { "ItemType", "Тип матеріалу" },
            { "Quantity", "Кількість" },
            { "Unit", "Од. виміру" }
        };
    }
    else if (currentSection == "Employees")
    {
        return new Dictionary<string, string>
        {
            { "Id", "ID" },
            { "FullName", "ПІБ" },
            { "Position", "Посада" },
            { "Phone", "Телефон" },
            { "Email", "Email" },
            { "HireDate", "Дата прийняття" },
            { "DateOfBirth", "Дата народження" }
        };
    }
    return new Dictionary<string, string>();
}

private void RefreshData()
{
    // Оновити дані для поточного розділу
    switch (currentSection)
    {
        case "Fields":
            LoadFields();
            break;
        case "Equipment":
            LoadEquipment();
            break;
        case "Harvest":
            LoadHarvest();
            break;
        case "Warehouse":
            LoadWarehouse();
            break;
        case "Employees":
            LoadEmployees();
            break;
        case "Reports":
            ShowReports();
            break;
    }
}
}
}

```

FieldRepository.cs

```
using System.Collections.Generic;
using System.Data.SQLite;

namespace AgroHolding
{
    // Клас для роботи з таблицею Fields (Поля)
    public static class FieldRepository
    {
        private static string dbFile = "agri_company.db";

        // Метод для отримання всіх полів
        public static List<Field> GetAllFields()
        {
            var fields = new List<Field>();
            using (var connection = new SQLiteConnection($"Data
Source={dbFile};Version=3;"))
            {
                connection.Open();
                string query = "SELECT id, name, area, crop,
sowing_date, harvest_date FROM Fields";
                using (var command = new SQLiteCommand(query,
connection))
                using (var reader = command.ExecuteReader())
                {
                    while (reader.Read())
                    {
                        fields.Add(new Field
                        {
                            Id = reader.GetInt32(0),
                            Name = reader.GetString(1),
                            Area = reader.GetDouble(2),
                            Crop = reader.GetString(3),
                            SowingDate = reader.IsDBNull(4) ?
null : reader.GetString(4),
                            HarvestDate = reader.IsDBNull(5) ?
null : reader.GetString(5)
                        });
                    }
                }
            }
            return fields;
        }

        public static void UpdateField(Field field)
        {
            using (var conn = new SQLiteConnection("Data
Source=agri_company.db;Version=3;"))
            {
                conn.Open();
                string query = @"UPDATE Fields SET
```

```

        name = @name,
        area = @area,
        crop = @crop,
        sowing_date = @sowing,
        harvest_date = @harvest
        WHERE id = @id";
using (var cmd = new SQLiteCommand(query, conn))
{
    cmd.Parameters.AddWithValue("@name",
field.Name);
    cmd.Parameters.AddWithValue("@area",
field.Area);
    cmd.Parameters.AddWithValue("@crop",
field.Crop);
    cmd.Parameters.AddWithValue("@sowing",
field.SowingDate);
    cmd.Parameters.AddWithValue("@harvest",
field.HarvestDate);
    cmd.Parameters.AddWithValue("@id", field.Id);
    cmd.ExecuteNonQuery();
}
}

public static void DeleteField(int id)
{
    using (var conn = new SQLiteConnection("Data
Source=agri_company.db;Version=3;"))
    {
        conn.Open();
        string query = "DELETE FROM Fields WHERE id = @id";
        using (var cmd = new SQLiteCommand(query, conn))
        {
            cmd.Parameters.AddWithValue("@id", id);
            cmd.ExecuteNonQuery();
        }
    }
}

public static Field GetFieldById(int id)
{
    using (var conn = new SQLiteConnection("Data
Source=agri_company.db;Version=3;"))
    {
        conn.Open();
        string query = "SELECT id, name, area, crop,
sowing_date, harvest_date FROM Fields WHERE id = @id";
        using (var cmd = new SQLiteCommand(query, conn))
        {
            cmd.Parameters.AddWithValue("@id", id);
            using (var reader = cmd.ExecuteReader())
            {

```

```

        if (reader.Read())
        {
            return new Field
            {
                Id = reader.GetInt32(0),
                Name = reader.GetString(1),
                Area = reader.GetDouble(2),
                Crop = reader.GetString(3),
                SowingDate = reader.GetString(4),
                HarvestDate = reader.GetString(5)
            };
        }
    }
}
return null;
}
}

```

```

// Клас-модель для поля
public class Field
{
    public int Id { get; set; }
    public string Name { get; set; }
    public double Area { get; set; }
    public string Crop { get; set; }
    public string SowingDate { get; set; }
    public string HarvestDate { get; set; }
}

```

FieldAddForm.cs

```

using System;
using System.Windows.Forms;
using System.Data.SQLite;

namespace AgroHolding
{
    public class FieldAddForm : Form
    {
        public bool IsAdded { get; private set; } = false;
        private Field editingField = null;

        private TextBox textName;
        private NumericUpDown numericArea;
        private ComboBox comboCrop;
        private DateTimePicker dtSowing;
        private DateTimePicker dtHarvest;
        private Button buttonAdd;
    }
}

```

```

        private Button buttonCancel;

        public FieldAddForm()
        {
            System.Threading.Thread.CurrentThread.CurrentCulture
= new System.Globalization.CultureInfo("uk-UA");

            System.Threading.Thread.CurrentThread.CurrentUICulture    =    new
            System.Globalization.CultureInfo("uk-UA");

            this.Text = "Додавання нового поля";
            InitializeForm();
        }

        public FieldAddForm(Field fieldToEdit) : this()
        {
            editingField = fieldToEdit;
            if (editingField != null)
            {
                this.Text = "Редагування поля";
                textName.Text = editingField.Name;
                numericArea.Value = (decimal)editingField.Area;
                comboCrop.Text = editingField.Crop ?? "";

                if
                (!string.IsNullOrWhiteSpace(editingField.SowingDate)                &&
                DateTime.TryParse(editingField.SowingDate, out DateTime sowing))
                {
                    dtSowing.Value = sowing;
                    dtSowing.Checked = true;
                }
                else
                {
                    dtSowing.Checked = false;
                }
                if
                (!string.IsNullOrWhiteSpace(editingField.HarvestDate)                &&
                DateTime.TryParse(editingField.HarvestDate, out DateTime
                harvest))
                {
                    dtHarvest.Value = harvest;
                    dtHarvest.Checked = true;
                }
                else
                {
                    dtHarvest.Checked = false;
                }
                buttonAdd.Text = "Зберегти";
            }
        }
    }

```

```

private void InitializeForm()
{
    int inputWidth = 215;

    var labelName = new System.Windows.Forms.Label() {
Text = "Назва поля:", Left = 10, Top = 15, Width = 100 };

    textName = new System.Windows.Forms.TextBox() { Name
= "textName", Left = 120, Top = 12, Width = inputWidth };

    var labelArea = new System.Windows.Forms.Label() {
Text = "Площа (га):", Left = 10, Top = 45, Width = 100 };

    numericArea = new
System.Windows.Forms.NumericUpDown()
    {
        Name = "numericArea",
        Left = 120,
        Top = 42,
        Width = inputWidth,
        DecimalPlaces = 2,
        Minimum = 0,
        Maximum = 10000,
        Increment = 0.1M
    };

    var labelCrop = new System.Windows.Forms.Label() {
Text = "Культура:", Left = 10, Top = 75, Width = 100 };

    comboCrop = new System.Windows.Forms.ComboBox()
    {
        Name = "comboCrop",
        Left = 120,
        Top = 72,
        Width = inputWidth,
        DropDownStyle = ComboBoxStyle.DropDown
    };

    comboCrop.Items.AddRange(new string[]
    {
        "Пшениця", "Кукурудза", "Соняшник", "Ячмінь",
        "Ріпак", "Соя", "Гречка", "Овес"
    });

    var labelSowing = new System.Windows.Forms.Label() {
Text = "Дата посіву:", Left = 10, Top = 105, Width = 100 };

    dtSowing = new System.Windows.Forms.DateTimePicker()
    {
        Name = "dtSowing",
        Left = 120,

```

```

        Top = 102,
        Width = inputWidth,
        Format = DateTimePickerFormat.Custom,
        CustomFormat = "dd MMMM yyyy",
        ShowCheckBox = true
    };

    var labelHarvest = new System.Windows.Forms.Label() {
Text = "Дата збору:", Left = 10, Top = 135, Width = 100 };

    dtHarvest = new System.Windows.Forms.DateTimePicker()
    {
        Name = "dtHarvest",
        Left = 120,
        Top = 132,
        Width = inputWidth,
        Format = DateTimePickerFormat.Custom,
        CustomFormat = "dd MMMM yyyy",
        ShowCheckBox = true
    };

    int buttonWidth = 100;
    int buttonHeight = 40;
    int buttonTop = 175;
    int buttonLeftAdd = 65;
    int buttonLeftCancel = 185;

    buttonAdd = new System.Windows.Forms.Button()
    {
        Text = "Додати",
        Left = buttonLeftAdd,
        Top = buttonTop,
        Width = buttonWidth,
        Height = buttonHeight
    };
    buttonAdd.Click += ButtonAdd_Click;

    buttonCancel = new System.Windows.Forms.Button()
    {
        Text = "Відмінити",
        Left = buttonLeftCancel,
        Top = buttonTop,
        Width = buttonWidth,
        Height = buttonHeight
    };
    buttonCancel.Click += (s, e) =>
    {
        this.DialogResult = DialogResult.Cancel;
        this.Close();
    };

    this.Controls.AddRange(new Control[]

```

```

        {
            labelName, textName,
            labelArea, numericArea,
            labelCrop, comboCrop,
            labelSowing, dtSowing,
            labelHarvest, dtHarvest,
            buttonAdd, buttonCancel
        });

        this.ClientSize = new System.Drawing.Size(350, 230);
        this.FormBorderStyle = FormBorderStyle.FixedDialog;
        this.MaximizeBox = false;
        this.MinimizeBox = false;
        this.StartPosition = FormStartPosition.CenterParent;
    }

    private void ButtonAdd_Click(object sender, EventArgs e)
    {
        try
        {
            if (string.IsNullOrEmpty(textName.Text) ||
                string.IsNullOrEmpty(comboCrop.Text))
            {
                MessageBox.Show("Будь ласка, заповніть всі обов'язкові поля!", "Помилка", MessageBoxButtons.OK, MessageBoxIcon.Error);
                return;
            }

            double area = (double)numericArea.Value;
            string sowingDate = dtSowing.Checked ? dtSowing.Value.ToString("yyyy-MM-dd") : null;
            string harvestDate = dtHarvest.Checked ? dtHarvest.Value.ToString("yyyy-MM-dd") : null;

            if (editingField == null)
            {
                using (var connection = new SQLiteConnection("Data Source=agri_company.db;Version=3;"))
                {
                    connection.Open();
                    string query = "INSERT INTO Fields (name, area, crop, sowing_date, harvest_date) VALUES (@name, @area, @crop, @sowing, @harvest)";
                    using (var cmd = new SQLiteCommand(query, connection))
                    {
                        cmd.Parameters.AddWithValue("@name", textName.Text);

```



```

        cmd.Parameters.AddWithValue("@area",
area);

        cmd.Parameters.AddWithValue("@crop",
comboCrop.Text);

cmd.Parameters.AddWithValue("@sowing",    (object)sowingDate    ??
DBNull.Value);

cmd.Parameters.AddWithValue("@harvest",    (object)harvestDate    ??
DBNull.Value);

        cmd.ExecuteNonQuery();
    }
}
else
{
    using (var connection = new
SQLiteConnection("Data Source=agri_company.db;Version=3;"))
    {
        connection.Open();
        string query = @"UPDATE Fields SET
                                name = @name,
                                area = @area,
                                crop = @crop,
@sowing,                                sowing_date    =
@harvest                                harvest_date    =
                                WHERE id = @id";
        using (var cmd = new SQLiteCommand(query,
connection))
        {
            cmd.Parameters.AddWithValue("@name",
textName.Text);

            cmd.Parameters.AddWithValue("@area",
area);

            cmd.Parameters.AddWithValue("@crop",
comboCrop.Text);

cmd.Parameters.AddWithValue("@sowing",    (object)sowingDate    ??
DBNull.Value);

cmd.Parameters.AddWithValue("@harvest",    (object)harvestDate    ??
DBNull.Value);

            cmd.Parameters.AddWithValue("@id",
editingField.Id);

            cmd.ExecuteNonQuery();
        }
    }
}

```

```
        IsAdded = true;
        this.DialogResult = DialogResult.OK;
        this.Close();
    }
    catch (Exception ex)
    {
        MessageBox.Show("Виникла помилка: " + ex.Message,
"Помилка", MessageBoxButtons.OK, MessageBoxIcon.Error);
    }
}
}
```