

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ЖИТОМИРСЬКИЙ АГРОТЕХНІЧНИЙ ФАХОВИЙ КОЛЕДЖ

ВІДДІЛЕННЯ
«ІНЖЕНЕРНА ІНФРАСТРУКТУРА ТА КОМП'ЮТЕРНІ НАУКИ»

ЦИКЛОВА КОМІСІЯ
«ІНЖЕНЕРНА ІНФРАСТРУКТУРА ТА КОМП'ЮТЕРНІ НАУКИ»

ПОЯСНЮВАЛЬНА ЗАПИСКА

до дипломної роботи освітнього ступеня «фаховий молодший бакалавр»
за спеціальністю 122 «Комп'ютерні науки»

на тему:

«Система управління персоналом на підприємстві»

Виконав здобувач освіти групи П-42
Харчук Іван Олексійович

Керівник роботи:
Устименко Ярослав Іванович

Рецензент:
Устименко Людмила Миколаївна

Зміст

Вступ	6
РОЗДІЛ 1. ОГЛЯД ТА АНАЛІЗ ТЕХНОЛОГІЙ РОЗРОБКИ	7
1.1. Постановка основної задачі розробки	7
1.2. Огляд та порівняння аналогічних систем.	8
1.3. Вибір та обґрунтування технологій розробки	11
Висновки до розділу 1	13
РОЗДІЛ 2. ПРОЕКТУВАННЯ І РОЗРОБКА	14
2.1. Вибір алгоритмів та їх ефективність	14
2.2. База даних	16
2.3. Спроектовані класи програми	20
2.4. Програмна реалізація	22
Висновки до розділу 2	31
РОЗДІЛ 3. КЕРІВНИЦТВО КОРИСТУВАННЯ ПРОГРАМОЮ	33
3.1. Мінімальні вимоги до системи	33
3.2. Інтерфейс користувача	35
3.3. Інструкція для роботи з програмою	43
Висновки до розділу 3	48
ВИСНОВКИ	50
Перелік літературних джерел	52
Додаток А.	55

					ДР.122.042.034.ПЗ		
Змн.	Арк.	№ докум.	Підпис	Дата	Система управління персоналом на підприємстві		
Розробив		Харчук І. О.					
Перевірив		Устименко Я.І.					
Рецензент		Устименко Л.М.					
Н. Контр.							
Затвердив.		Лавріщев О.О.			ЖАТФК		
					Літ.	Арк.	Аркушів
						3	58

РЕФЕРАТ

Записка: 58 стор., 24 рис., 1 додаток, 20 джерел.

Ключові слова: управління персоналом, автоматизація, SQLite, .NET 8, Windows Forms, Entity Framework Core, база даних, інтерфейс користувача.

Дипломна робота присвячена розробці системи управління персоналом на підприємстві з використанням Windows Forms, .NET 8, SQLite та Entity Framework Core. Мета роботи – створення ефективного інструменту для автоматизації кадрового обліку, управління користувачами, посадами, відділами та навичками.

У першому розділі проведено аналіз предметної області, визначено вимоги до системи та обґрунтовано вибір технологій: SQLite для портативності, .NET 8 для продуктивності, Windows Forms для зручного інтерфейсу. Другий розділ описує проектування системи, включаючи алгоритми асинхронних запитів, хешування паролів (SHA-256) і транзакційну обробку. Структура бази даних охоплює таблиці для співробітників, користувачів, посад, відділів і навичок. Реалізація включає форми для автентифікації, головного меню та каталогів із модульною архітектурою. Третій розділ містить керівництво користувача: мінімальні системні вимоги (2 ГБ ОЗП, Windows 10/11), опис інтерфейсу та інструкцію для роботи.

Система забезпечує безпечне управління даними, валідацію введення. Вона готова до використання в малих і середніх організаціях і може бути розширена для додаткових функцій.

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

СУП – Система управління персоналом.

ІС – Інформаційна система.

БД – База даних.

SQLite – Легка реляційна система управління базами даних, використана в системі.

.NET 8 – Платформа розробки від Microsoft, основа програмного забезпечення системи.

EF Core – Entity Framework Core, інструмент для роботи з даними в .NET.

SHA-256 – Алгоритм криптографічного хешування, використаний для захисту паролів.

LINQ – language Integrated Query, технологія для створення запитів до даних у .NET.

UI – User Interface (інтерфейс користувача).

HR – Human Resources (управління людськими ресурсами).

CRUD – Create, Read, Update, Delete (операції створення, читання, оновлення, видалення даних).

SQL – Structured Query Language (мова структурованих запитів).

DBMS – Database Management System (система управління базами даних).

					ДР.122.042.034.ПЗ	Арк.
						5
Змн.	Арк.	№ докум.	Підпис	Дата		

ВСТУП

У сучасних умовах ефективне управління персоналом є ключовим фактором успіху будь-якого підприємства. Організація процесів, пов'язаних із наймом, обліком, розвитком та оцінкою працівників, вимагає використання автоматизованих систем, які дозволяють оптимізувати роботу кадрових служб, підвищувати продуктивність та забезпечувати точність даних. Розробка інформаційних систем для управління персоналом є актуальним завданням, оскільки такі системи сприяють зменшенню витрат часу, підвищенню прозорості процесів та покращенню прийняття управлінських рішень.

Дипломна робота присвячена розробці системи управління персоналом на підприємстві, яка реалізує основні функції кадрового обліку, включаючи управління даними про співробітників, користувачів системи, їхні ролі та дії. Система розроблена з використанням технології Windows Forms на платформі .NET 8, що забезпечує зручний та інтуїтивно зрозумілий інтерфейс користувача. Як базу даних обрано SQLite, що дозволяє створювати легкі, портативні та ефективні рішення для зберігання даних без необхідності встановлення складних серверних систем.

Метою роботи є створення функціональної системи, яка автоматизує облік персоналу, забезпечує безпечне управління користувачами та їхніми діями, а також відповідає вимогам сучасних інформаційних технологій. У процесі розробки було реалізовано форми для управління співробітниками та користувачами, які дозволяють додавати, редагувати, видаляти записи, а також вести облік дій користувачів у системі. Використання асинхронного програмування та Entity Framework Core забезпечило високу продуктивність і надійність роботи з базою даних.

Розроблена система може бути використана на підприємствах різного масштабу для оптимізації кадрових процесів, забезпечення безпеки даних і спрощення доступу до інформації.

					ДР.122.042.034.ПЗ	Арк.
						6
Змн.	Арк.	№ докум.	Підпис	Дата		

РОЗДІЛ 1. ОГЛЯД ТА АНАЛІЗ ТЕХНОЛОГІЙ РОЗРОБКИ

1.1. Постановка основної задачі розробки

Розробка системи управління персоналом на підприємстві спрямована на створення ефективного інструменту для автоматизації процесів кадрового обліку, управління даними про співробітників, користувачів системи та їхні дії. Основна задача полягає у забезпеченні зручного, безпечного та надійного програмного рішення, яке дозволить оптимізувати роботу кадрових служб, зменшити час на обробку інформації та підвищити точність ведення обліку. Система повинна надавати функціонал для додавання, редагування, видалення та перегляду даних про працівників, а також управління користувачами з різними рівнями доступу, включаючи адміністраторів, менеджерів та звичайних користувачів.

Ключовим аспектом є створення інтуїтивно зрозумілого інтерфейсу, який забезпечить простоту використання для працівників із різним рівнем технічної підготовки. Програма має підтримувати асинхронну роботу з базою даних для підвищення продуктивності, а також включати механізми валідації даних і обробки помилок, щоб запобігти некоректному введенню інформації. Важливим завданням є забезпечення безпеки даних, зокрема через хешування паролів користувачів і обмеження доступу до функціоналу залежно від ролі.

Для реалізації системи обрано технологію Windows Forms на платформі .NET 8, яка забезпечує швидку розробку настільних додатків із багатим набором компонентів для створення графічного інтерфейсу. Як базу даних використано SQLite, що дозволяє створювати портативні рішення без необхідності встановлення серверного програмного забезпечення. Розробка передбачає використання Entity Framework Core для спрощення взаємодії з базою даних, а також асинхронного програмування для оптимізації операцій із даними. Таким чином, основна задача розробки полягає у створенні комплексної системи, яка відповідає вимогам сучасного управління персоналом і забезпечує ефективну взаємодію між користувачами та даними підприємства.

					ДР.122.042.034.ПЗ	Арк.
						7
Змн.	Арк.	№ докум.	Підпис	Дата		

1.2. Огляд та порівняння аналогічних систем.

Сучасний ринок інформаційних систем пропонує різноманітні рішення для управління персоналом, які варіюються за функціоналом, технологічною базою та цільовою аудиторією. Для визначення оптимального підходу до розробки системи управління персоналом на підприємстві було проведено огляд та порівняння кількох популярних аналогічних систем, таких як SAP SuccessFactors, BambooHR та Hurma System. Кожна з цих систем має свої особливості, переваги та недоліки, що впливають на вибір технологій і функціональних можливостей для реалізації власного рішення.

SAP SuccessFactors є однією з провідних хмарних платформ для управління персоналом, яка підтримує широкий спектр функцій, включаючи облік співробітників, управління продуктивністю, навчання, рекрутинг та аналітику. Система орієнтована на великі корпорації з розгалуженою організаційною структурою. Її перевагами є висока масштабованість, інтеграція з іншими модулями SAP та потужні аналітичні інструменти. Проте складність налаштування, висока вартість ліцензії та необхідність постійного доступу до інтернету роблять її менш доступною для малих і середніх підприємств. Крім того, хмарна архітектура може викликати занепокоєння щодо безпеки даних у деяких організаціях.

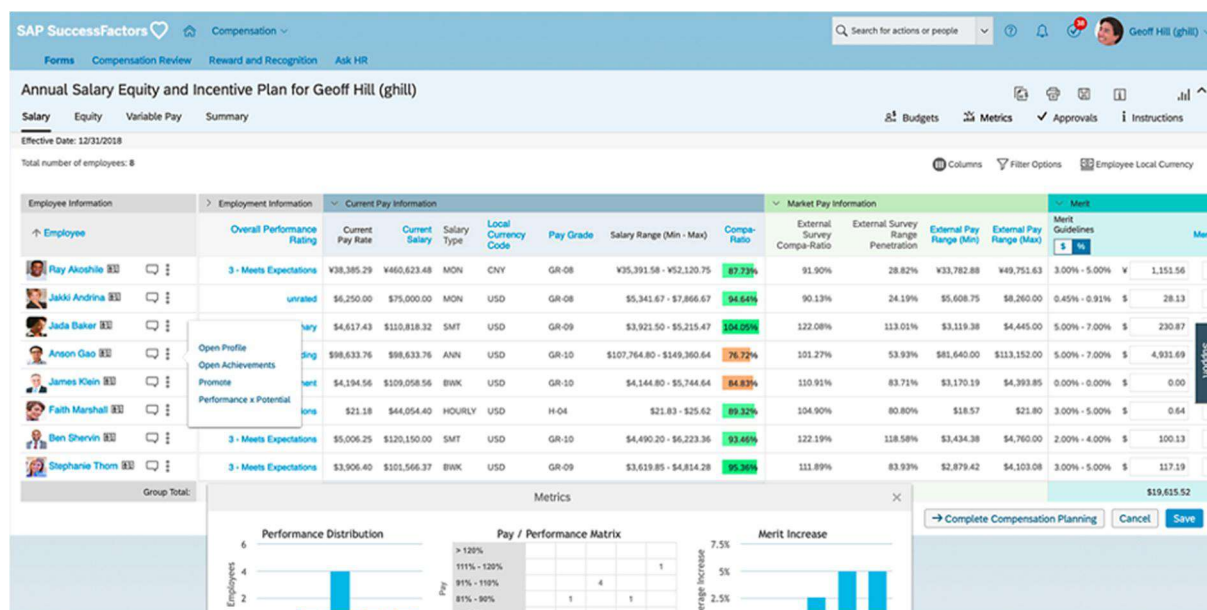


Рисунок 1.2.1 – SAP SuccessFactors

BambooHR — це хмарне рішення, призначене для малих і середніх компаній. Воно пропонує зручний інтерфейс для управління даними співробітників, відстеження робочого часу, створення звітів та автоматизації процесів найму. Система вирізняється простотою використання та швидким впровадженням, що робить її популярною серед компаній із обмеженими ресурсами для ІТ. Однак BambooHR має обмежені можливості кастомізації, що може бути недоліком для організацій із специфічними вимогами. Також залежність від хмарної інфраструктури створює ризики, пов'язані з доступністю даних у разі збоїв у мережі.

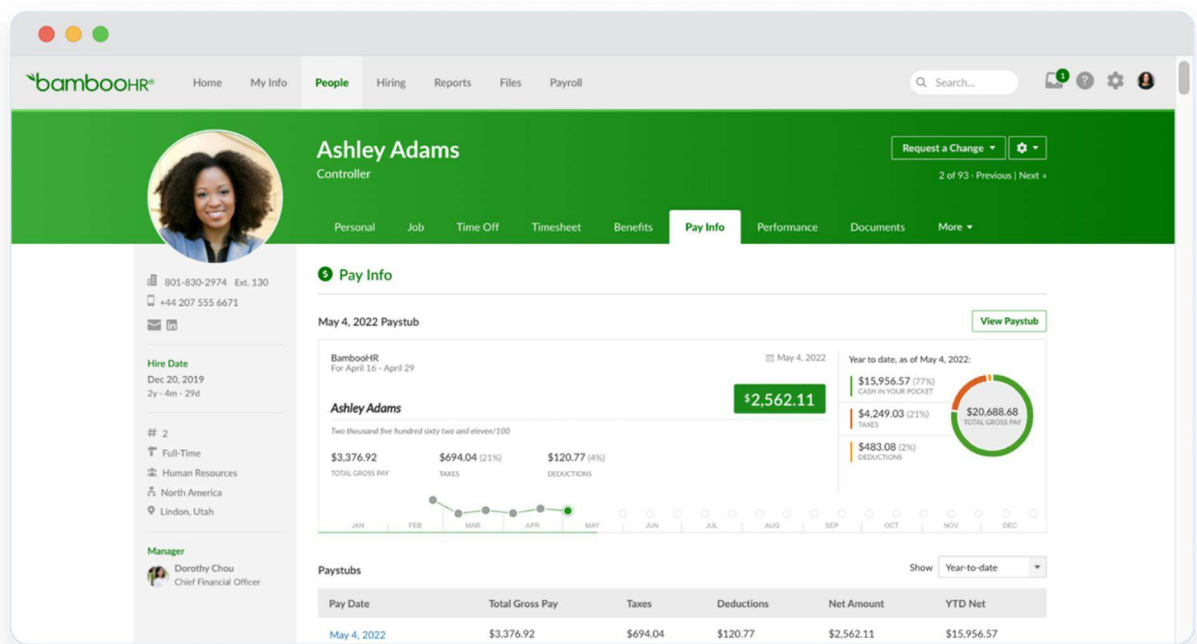


Рисунок 1.2.2 – BambooHR

Hurma System — це сучасна HRM-система, розроблена з акцентом на автоматизацію кадрових процесів, таких як рекрутинг, адаптація, оцінка персоналу та управління відпустками. Вона підтримує як хмарну, так і локальну інсталяцію, що забезпечує гнучкість у виборі моделі розгортання. Hurma System пропонує інтуїтивний інтерфейс і модульну структуру, що дозволяє підключати лише необхідні функції. Проте система має відносно обмежену підтримку складних аналітичних звітів порівняно з SAP

SuccessFactors, а також може вимагати додаткових налаштувань для інтеграції з іншими корпоративними системами.

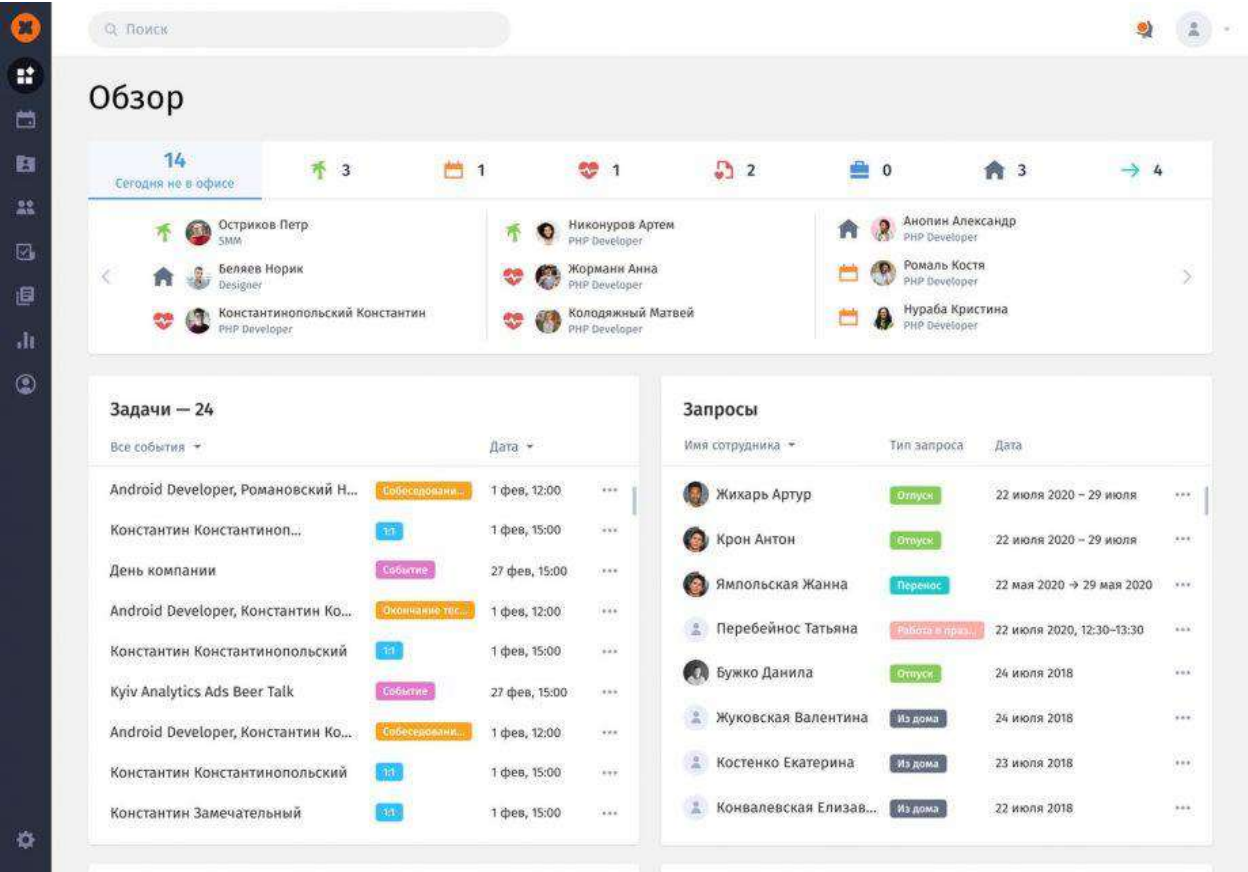


Рисунок 1.2.3 – Hurma System

Порівняння цих систем показало, що хмарні рішення, такі як SAP SuccessFactors і BambooHR, є зручними для компаній із розподіленими командами, але їхня залежність від інтернету та висока вартість можуть бути бар'єрами для малих підприємств. Hurma System пропонує баланс між гнучкістю та простотою, однак її функціонал може бути недостатнім для великих організацій із комплексними вимогами. Усі розглянуті системи використовують веб-технології, що забезпечують крос-платформений доступ, але для локального використання на підприємствах із обмеженим бюджетом настільні додатки залишаються актуальними.

На основі аналізу було вирішено розробити настільну систему управління персоналом на базі Windows Forms і .NET 8 з використанням SQLite як локальної бази даних. Таке рішення дозволяє створити портативний додаток, який не залежить від інтернет-з'єднання, має низькі вимоги до

апаратного забезпечення та забезпечує достатній рівень безпеки даних. На відміну від хмарних аналогів, розроблена система орієнтована на малі та середні підприємства, які потребують простого, але функціонального інструменту для кадрового обліку. Вона підтримує управління співробітниками, користувачами, їхніми ролями та діями, а також включає механізми валідації даних і логування, що відповідає основним вимогам до HRM-систем.

1.3. Вибір та обґрунтування технологій розробки

Вибір технологій для розробки системи управління персоналом на підприємстві базується на потребах забезпечення простоти використання, продуктивності, портативності та економічної доцільності для малих і середніх організацій. Після аналізу вимог до системи, таких як автоматизація кадрового обліку, управління користувачами, безпечне зберігання даних і зручний інтерфейс, було обрано технологію Windows Forms на платформі .NET 8 у поєднанні з базою даних SQLite та інструментами Entity Framework Core для взаємодії з даними.

Windows Forms обрано як основну технологію для створення графічного інтерфейсу через її зрілість, простоту розробки та широкий набір компонентів для побудови настільних додатків. Ця технологія дозволяє швидко створювати форми з інтуїтивно зрозумілим дизайном, що є важливим для користувачів із різним рівнем технічної підготовки, таких як працівники кадрових служб. Windows Forms забезпечує стабільну роботу на платформах Windows, які є стандартними для більшості офісних середовищ, і не вимагає додаткових витрат на розгортання серверів чи хмарних сервісів. Крім того, інтеграція з .NET 8 надає доступ до сучасних можливостей платформи, таких як покращена продуктивність, підтримка асинхронного програмування та розширена бібліотека класів.

Платформа .NET 8 обрана через її універсальність, високу продуктивність і активну підтримку спільнотою розробників. Вона забезпечує

					ДР.122.042.034.ПЗ	Арк.
						11
Змн.	Арк.	№ докум.	Підпис	Дата		

надійне виконання коду, ефективне управління пам'яттю та вбудовані механізми безпеки, що є критично важливим для захисту даних користувачів, зокрема паролів, які зберігаються у хешованому вигляді. Використання асинхронного програмування через ключові слова `async/await` дозволяє оптимізувати операції з базою даних, зменшуючи час очікування для користувача під час виконання запитів. .NET 8 також підтримує крос-платформні бібліотеки, що залишає можливість для майбутнього розширення системи, якщо з'явиться потреба в адаптації до інших операційних систем.

Для зберігання даних обрано SQLite як легку та портативну базу даних, яка не потребує встановлення серверного програмного забезпечення. SQLite зберігає всі дані в одному файлі, що спрощує розгортання системи на комп'ютерах підприємства та забезпечує легке резервне копіювання. Ця база даних підтримує стандартні SQL-запити, включаючи зовнішні ключі, що дозволяє реалізувати зв'язки між таблицями, наприклад, між таблицями `Users` і `Employees`. SQLite є оптимальним вибором для малих і середніх організацій, оскільки не потребує значних апаратних ресурсів і забезпечує достатню швидкодію для обробки даних про персонал.

Entity Framework Core використано як інструмент для роботи з базою даних через об'єктно-реляційне відображення (ORM). Цей інструмент спрощує операції з даними, дозволяючи розробнику працювати з об'єктами .NET замість прямих SQL-запитів, що зменшує ймовірність помилок і прискорює розробку. Entity Framework Core підтримує асинхронні запити, що підвищує продуктивність системи, особливо під час виконання операцій із великими наборами даних. Крім того, він забезпечує гнучке налаштування моделей даних, що дозволило точно відтворити структуру таблиць, таких як `Users` і `Employees`, із урахуванням обмежень унікальності та зовнішніх ключів.

Обґрунтування вибору цих технологій також пов'язане з їхньою економічною доцільністю. Windows Forms, .NET 8 і SQLite є безкоштовними для використання, що знижує витрати на розробку та впровадження системи. Entity Framework Core, як частина екосистеми .NET, не потребує додаткових

					ДР.122.042.034.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		12

ліцензій. Поєднання цих технологій забезпечує баланс між функціональністю, продуктивністю та доступністю, що відповідає потребам цільової аудиторії – малих і середніх підприємств, які прагнуть автоматизувати кадрові процеси без значних інвестицій у хмарні чи серверні рішення. Таким чином, обрані технології дозволяють створити надійну, портативну та зручну систему управління персоналом, яка відповідає сучасним вимогам до інформаційних систем.

Висновки до розділу 1

Проведений аналіз технологій розробки системи управління персоналом на підприємстві дозволив сформулювати чіткі вимоги до програмного забезпечення та обґрунтувати вибір інструментів для її реалізації. Основна задача розробки полягає в створенні зручного, безпечного та ефективного настільного додатка, який автоматизує кадровий облік, управління користувачами та їхніми діями, забезпечуючи простоту використання для працівників із різним рівнем технічної підготовки.

Огляд аналогічних систем, таких як SAP SuccessFactors, BambooHR та Hurma System, показав, що хмарні рішення мають переваги в масштабованості та крос-платформеності, але їхня висока вартість і залежність від інтернет-з'єднання роблять їх менш придатними для малих і середніх підприємств. Натомість настільні додатки, реалізовані за допомогою локальних баз даних, забезпечують портативність і економічну доцільність, що є критично важливим для цільової аудиторії.

Вибір технологій, зокрема Windows Forms на платформі .NET 8, SQLite та Entity Framework Core, обґрунтовано їхньою здатністю забезпечити швидку розробку, високу продуктивність, безпеку даних і низькі витрати на впровадження. Windows Forms дозволяє створювати інтуїтивно зрозумілий інтерфейс, .NET 8 підтримує сучасні можливості асинхронного програмування, SQLite забезпечує легке та портативне зберігання даних, а Entity Framework Core спрощує взаємодію з базою даних.

					ДР.122.042.034.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		13

РОЗДІЛ 2. ПРОЕКТУВАННЯ І РОЗРОБКА

2.1. Вибір алгоритмів та їх ефективність

Проектування системи управління персоналом на підприємстві передбачає вибір оптимальних алгоритмів для реалізації основних функцій, таких як завантаження даних, їх обробка, збереження, валідація та логування дій користувачів. Ефективність алгоритмів є критично важливою для забезпечення швидкодії системи, особливо при роботі з базою даних, обробці великих обсягів інформації та забезпеченні безпеки даних. У процесі розробки системи були обрані алгоритми, які поєднують простоту реалізації, високу продуктивність і відповідність вимогам до настільних додатків на основі Windows Forms, .NET 8 та SQLite.

Для завантаження даних із бази даних, зокрема списків співробітників і користувачів, використано асинхронні запити через Entity Framework Core. Алгоритм асинхронного завантаження даних дозволяє виконувати операції введення-виведення без блокування основного потоку інтерфейсу користувача, що забезпечує плавну роботу програми. Наприклад, у методі LoadUsers форми frmUsersCatalog використовується LINQ-запит із включенням пов'язаних даних (Include(u => u.Employee)), що мінімізує кількість звернень до бази даних. Ефективність цього алгоритму залежить від оптимізації запитів: вибірка лише необхідних полів через Select зменшує обсяг даних, що передаються, а асинхронне виконання (ToListAsync) скорочує час очікування. Часова складність такого алгоритму наближається до $O(n)$, де n – кількість записів у таблиці, оскільки SQLite виконує послідовне сканування таблиць для малих наборів даних.

Для збереження та оновлення даних, як у методі btnSave_Click форм frmEmployeesCatalog і frmUsersCatalog, застосовується алгоритм транзакційної обробки. Entity Framework Core автоматично керує транзакціями під час виклику SaveChangesAsync, що гарантує атомарність операцій. Алгоритм включає валідацію введених даних (перевірка унікальності, формату, обмежень довжини) перед збереженням, що запобігає

					ДР.122.042.034.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		14

некоректним записам. Ефективність цього алгоритму забезпечується мінімальною кількістю операцій із базою даних: одноразове звернення для перевірки унікальності (AnyAsync) і одне для збереження. Часова складність залежить від розміру даних і становить $O(1)$ для операцій із одним записом, оскільки SQLite оптимізує вставку та оновлення за первинним ключем.

Хешування паролів у формі frmUsersCatalog реалізовано за допомогою алгоритму SHA-256, який є стандартним для забезпечення безпеки даних. Метод ComputeSHA256Hash перетворює пароль у байтовий масив, обчислює хеш і повертає його у шістнадцятковому форматі. SHA-256 має фіксовану часову складність $O(k)$, де k – довжина вхідного рядка, що забезпечує швидке виконання навіть для довгих паролів. Використання цього алгоритму гарантує захист паролів від зворотного відновлення, а також відповідає сучасним стандартам безпеки.

Для обробки вибору записів у DataGridView (методи dgvEmployees_CellClick і dgvUsers_CellClick) застосовується алгоритм асинхронного завантаження пов'язаних даних. Використання BeginInvoke для оновлення елементів інтерфейсу в основному потоці забезпечує синхронізацію даних між асинхронними запитами та графічним інтерфейсом. Ефективність цього алгоритму полягає в одноразовому зверненні до бази даних для отримання повного об'єкта (FirstOrDefaultAsync) із включенням пов'язаних записів, що має часову складність $O(1)$ для пошуку за первинним ключем.

Обрані алгоритми є оптимальними для настільного додатка з локальною базою даних, оскільки вони враховують обмеження SQLite, такі як послідовний доступ до даних, і мінімізують кількість звернень до бази. Використання асинхронного програмування підвищує швидкодію інтерфейсу, а оптимізовані LINQ-запити зменшують обсяг оброблюваних даних. Алгоритми валідації та хешування забезпечують надійність і безпеку, що відповідає вимогам до системи управління персоналом.

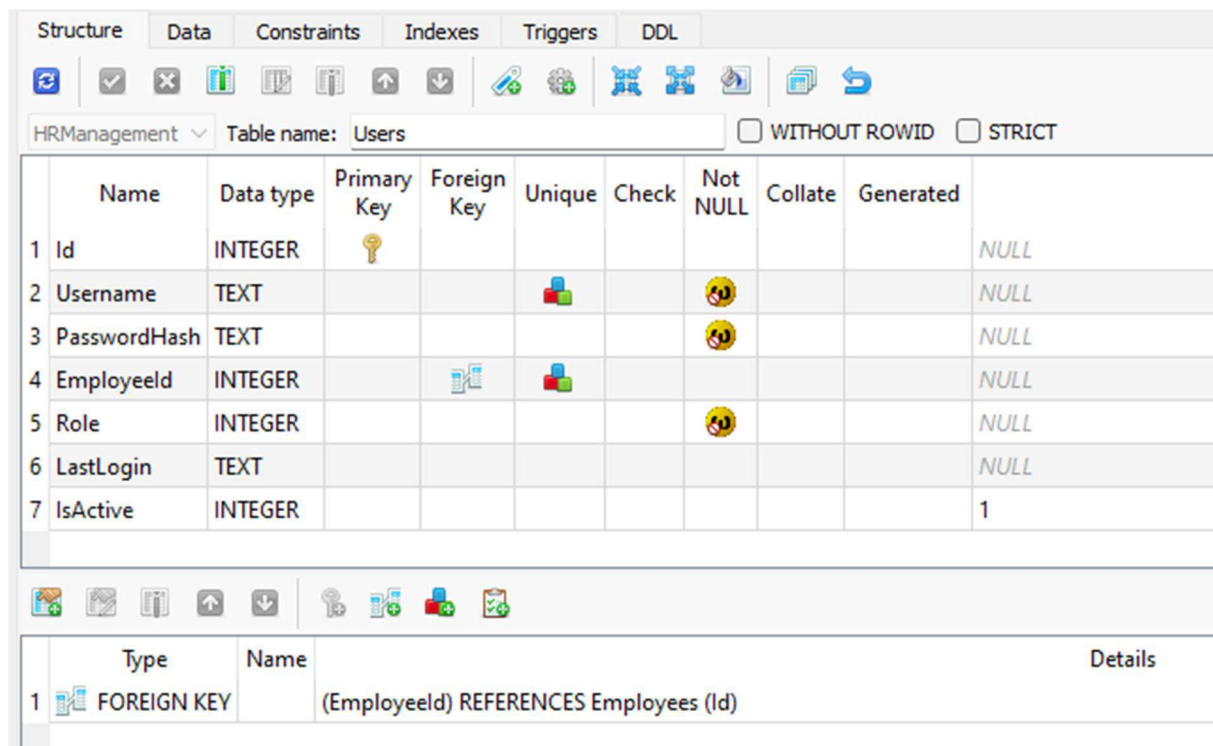
					ДР.122.042.034.ПЗ	Арк.
						15
Змн.	Арк.	№ докум.	Підпис	Дата		

2.2. База даних

База даних є ключовим компонентом системи управління персоналом на підприємстві, забезпечуючи структуроване зберігання, швидкий доступ та цілісність даних про співробітників, користувачів, їхні дії та пов'язані об'єкти. Для реалізації системи обрано SQLite як легку, портативну та ефективну реляційну базу даних, яка відповідає потребам малих і середніх організацій. Структура бази даних розроблена з урахуванням вимог до управління персоналом, безпеки та підтримки зв'язків між об'єктами, що забезпечує надійну основу для функціонування системи.

База даних складається з кількох основних таблиць, які відображають ключові сутності системи.

Таблиця Users містить дані про користувачів системи, такі як ідентифікатор (Id), ім'я користувача (Username), хеш пароля (PasswordHash), ідентифікатор пов'язаного співробітника (EmployeeId), роль (Role), час останнього входу (LastLogin) та статус активності (IsActive). Унікальність забезпечується полями Username та EmployeeId, а також зовнішнім ключем, що пов'язує EmployeeId із таблицею Employees.



	Name	Data type	Primary Key	Foreign Key	Unique	Check	Not NULL	Collate	Generated
1	Id	INTEGER	✓						NULL
2	Username	TEXT			✓		✗		NULL
3	PasswordHash	TEXT					✗		NULL
4	EmployeeId	INTEGER		✓	✓				NULL
5	Role	INTEGER					✗		NULL
6	LastLogin	TEXT							NULL
7	IsActive	INTEGER							1

Type	Name	Details
FOREIGN KEY	(EmployeeId)	REFERENCES Employees (Id)

Рисунок 2.2.1 – Таблиця Users

Таблиця Employees зберігає інформацію про співробітників, включаючи ідентифікатор (Id), ім'я, прізвище, по батькові, дату народження, дату прийому, ідентифікатори посади та департаменту, зарплату, контактні дані, фотографію та статус активності. Унікальність записів забезпечується первинним ключем Id з автоінкрементом.

Structure Data Constraints Indexes Triggers DDL										
HRManagement Table name: Employees ☐ WITHOUT ROWID ☐ STRICT										
	Name	Data type	Primary Key	Foreign Key	Unique	Check	Not NULL	Collate	Generated	
1	Id	INTEGER								NULL
2	FirstName	TEXT								NULL
3	LastName	TEXT								NULL
4	MiddleName	TEXT								NULL
5	BirthDate	TEXT								NULL
6	HireDate	TEXT								NULL
7	PositionId	INTEGER								NULL
8	DepartmentId	INTEGER								NULL
9	Salary	REAL								NULL
10	Email	TEXT								NULL
11	Phone	TEXT								NULL
12	Address	TEXT								NULL
13	Photo	BLOB								NULL
14	Status	INTEGER								1
	Type	Name								
1	FOREIGN KEY	(PositionId) REFERENCES Positions (Id)								
2	FOREIGN KEY	(DepartmentId) REFERENCES Departments (Id)								

Рисунок 2.2.2 – Таблиця Employees

Structure Data Constraints Indexes Triggers DDL										
HRManagement Table name: Positions ☐ WITHOUT ROWID ☐ STRICT										
	Name	Data type	Primary Key	Foreign Key	Unique	Check	Not NULL	Collate	Generated	
1	Id	INTEGER								NULL
2	Title	TEXT								NULL
3	Code	TEXT								NULL
4	BaseSalary	REAL								NULL
5	Description	TEXT								NULL
6	IsManagerial	INTEGER								0

Рисунок 2.2.3 – Таблица Positions

Structure Data Constraints Indexes Triggers DDL										
HRManagement Table name: Departments ☐ WITHOUT ROWID ☐ STRICT										
	Name	Data type	Primary Key	Foreign Key	Unique	Check	Not NULL	Collate	Generated	
1	Id	INTEGER								NULL
2	Name	TEXT								NULL
3	Code	TEXT								NULL
4	ParentDepartmentId	INTEGER								NULL
5	ManagerId	INTEGER								NULL
6	Description	TEXT								NULL

Рисунок 2.2.4 – Таблица Departments

Structure Data Constraints Indexes Triggers DDL										
HRManagement Table name: Skills ☐ WITHOUT ROWID ☐ STRICT										
	Name	Data type	Primary Key	Foreign Key	Unique	Check	Not NULL	Collate	Generated	
1	Id	INTEGER								NULL
2	Name	TEXT								NULL
3	Description	TEXT								NULL

Рисунок 2.2.5 – Таблица Skills

Structure Data Constraints Indexes Triggers DDL										
HRManagement		Table name: EmployeeSkills		☐ WITHOUT ROWID		☐ STRICT				
	Name	Data type	Primary Key	Foreign Key	Unique	Check	Not NULL	Collate	Generated	
1	EmployeeId	INTEGER								NULL
2	SkillId	INTEGER								NULL
3	ProficiencyLevel	INTEGER								NULL

Type	Name	Details
1 PRIMARY KEY		(EmployeeId, SkillId)
2 FOREIGN KEY		(EmployeeId) REFERENCES Employees (Id)
3 FOREIGN KEY		(SkillId) REFERENCES Skills (Id)

Рисунок 2.2.6 – Таблиця EmployeeSkills

Додаткові таблиці включають Positions для зберігання інформації про посади (ідентифікатор і назву), Departments для департаментів (ідентифікатор і назву), Skills для навичок співробітників та EmployeeSkills для зв'язування співробітників із навичками та рівнем володіння.

Зовнішні ключі в таблицях Employees, Users і EmployeeSkills забезпечують цілісність даних, запобігаючи видаленню або модифікації пов'язаних записів без відповідних перевірок.

Для взаємодії з базою даних використано Entity Framework Core, який забезпечує об'єктно-реляційне відображення (ORM) і дозволяє працювати з даними через об'єкти .NET. Моделі даних, такі як Employee і User, точно відображають структуру таблиць, а налаштування через DbContext включають визначення первинних і зовнішніх ключів, унікальних індексів та обмежень. Наприклад, у таблиці Users поле Username позначено як унікальне, а EmployeeId має умовну унікальність для ненульових значень. Entity Framework Core також підтримує асинхронні операції, що підвищує швидкодію системи під час виконання запитів, таких як вибірка, вставка чи оновлення.

Оптимізація бази даних досягнута завдяки створенню індексів для часто використовуваних полів, таких як Username і EmployeeId у таблиці Users, що прискорює пошук і перевірку унікальності. Використання зовнішніх ключів із обмеженням RESTRICT запобігає видаленню співробітників, які пов'язані з користувачами, забезпечуючи цілісність даних. SQLite, як локальна база даних, зберігає всі дані в одному файлі, що спрощує резервне копіювання та перенесення системи між комп'ютерами. Обсяг даних, типовий для малих і середніх організацій (сотні або тисячі записів), не створює значного навантаження на SQLite, що забезпечує швидкий доступ до інформації.

Розроблена структура бази даних є гнучкою і дозволяє розширювати систему в майбутньому, наприклад, шляхом додавання таблиць для управління відпустками чи оцінки продуктивності. Використання SQLite та Entity Framework Core забезпечує баланс між простотою розгортання, ефективністю та підтримкою реляційних зв'язків, що відповідає вимогам до системи управління персоналом. Завдяки цьому база даних надійно підтримує всі операції системи, від завантаження списків до збереження та валідації даних.

2.3. Спроектовані класи програми

Розробка системи управління персоналом на підприємстві передбачала створення низки класів, які забезпечують взаємодію з базою даних, обробку даних, реалізацію логіки програми та відображення інформації через графічний інтерфейс. Програма побудована на основі технології Windows Forms, платформи .NET 8, SQLite та Entity Framework Core, що вплинуло на структуру класів. Спроектовані класи поділяються на моделі даних, контекст бази даних, класи форм і допоміжні класи, кожен із яких виконує чітко визначену роль у системі.

Основними класами моделей даних є Employee, User, Position, Department, Skill, EmployeeSkill та UserAction, які відображають відповідні

					ДР.122.042.034.ПЗ	Арк.
						20
Змн.	Арк.	№ докум.	Підпис	Дата		

таблиці бази даних. Клас Employee містить властивості для зберігання інформації про співробітника, такі як Id (ціле число, первинний ключ), FirstName, LastName, MiddleName (рядки), BirthDate, HireDate (рядки у форматі дати), PositionId, DepartmentId (цілі числа, зовнішні ключі), Salary (десяткове число), Email, Phone, Address (рядки), Photo (масив байтів) та Status (ціле число). Клас User включає властивості Id, Username, PasswordHash (рядки), EmployeeId (ціле число, зовнішній ключ, може бути null), Role, IsActive (цілі числа) та LastLogin (рядок). Решта моделей, такі як Position і Department, мають простішу структуру з полями Id і Title або Name. Клас EmployeeSkill реалізує зв'язок багато-до-багатьох між співробітниками та навичками, а UserAction зберігає лог дій користувачів із полями Id, UserId, Action та ActionTime.

Клас HRDbContext, похідний від DbContext, є центральним для взаємодії з базою даних. Він містить властивості типу DbSet для кожної моделі (DbSet<Employee> Employees, DbSet<User> Users тощо) і налаштування конфігурації через метод OnModelCreating. У цьому методі визначено первинні ключі, унікальні індекси (наприклад, для Username у User), зовнішні ключі з обмеженням RESTRICT і умовну унікальність для EmployeeId. Клас також ініціалізує підключення до SQLite за допомогою рядка з'єднання, що вказує на локальний файл бази даних.

Класи форм відповідають за графічний інтерфейс і логіку обробки подій. Клас frmEmployeesCatalog реалізує управління співробітниками, включаючи завантаження списків, додавання, редагування, видалення та збереження даних. Він містить методи, такі як LoadEmployees, LoadComboBoxes, btnSave_Click і dgvEmployees_CellClick, які обробляють асинхронні запити до бази даних і оновлення елементів керування (DataGridView, TextBox, ComboBox). Аналогічно, клас frmUsersCatalog відповідає за управління користувачами, включаючи методи для завантаження даних (LoadUsers), валідації (btnSave_Click) і хешування паролів (ComputeSHA256Hash). Клас

					ДР.122.042.034.ПЗ	Арк.
						21
Змн.	Арк.	№ докум.	Підпис	Дата		

frmLogin забезпечує автентифікацію користувачів, перевіряючи введені дані через порівняння хешу пароля з даними в таблиці Users.

Допоміжні класи включають статичний клас frmLogin із властивістю CurrentUser, яка зберігає дані поточного користувача після автентифікації. Ця властивість використовується в інших формах для обмеження доступу та логування дій. Наприклад, у frmUsersCatalog доступ до форми дозволено лише користувачам із роллю адміністратора (Role = 0).

Усі класи спроектовані з урахуванням принципів об'єктно-орієнтованого програмування, таких як інкапсуляція та модульність. Моделі даних ізольовано від логіки інтерфейсу, що спрощує їх повторне використання. Використання асинхронних методів у класах форм (async/await) підвищує продуктивність, а обробка винятків забезпечує стійкість до помилок. Класи форм взаємодіють із HRDbContext через асинхронні запити Entity Framework Core, що мінімізує затримки при роботі з базою даних. Таким чином, спроектована структура класів забезпечує чіткий розподіл обов'язків, ефективну обробку даних і зручний інтерфейс, що відповідає вимогам системи управління персоналом.

2.4. Програмна реалізація

Програмна реалізація системи управління персоналом на підприємстві виконана з використанням технології Windows Forms на платформі .NET 8, бази даних SQLite та Entity Framework Core для взаємодії з даними. Проект структуровано таким чином, щоб забезпечити модульність, зручність підтримки та чіткий розподіл функціональних компонентів. Система включає набір форм для взаємодії з користувачем, моделі даних, контекст бази даних і допоміжні класи, які разом реалізують функції кадрового обліку, управління користувачами та логування їхніх дій. Нижче описано склад проекту та детально розглянуто форми, які складають основу інтерфейсу користувача.

					ДР.122.042.034.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		22

Склад проекту

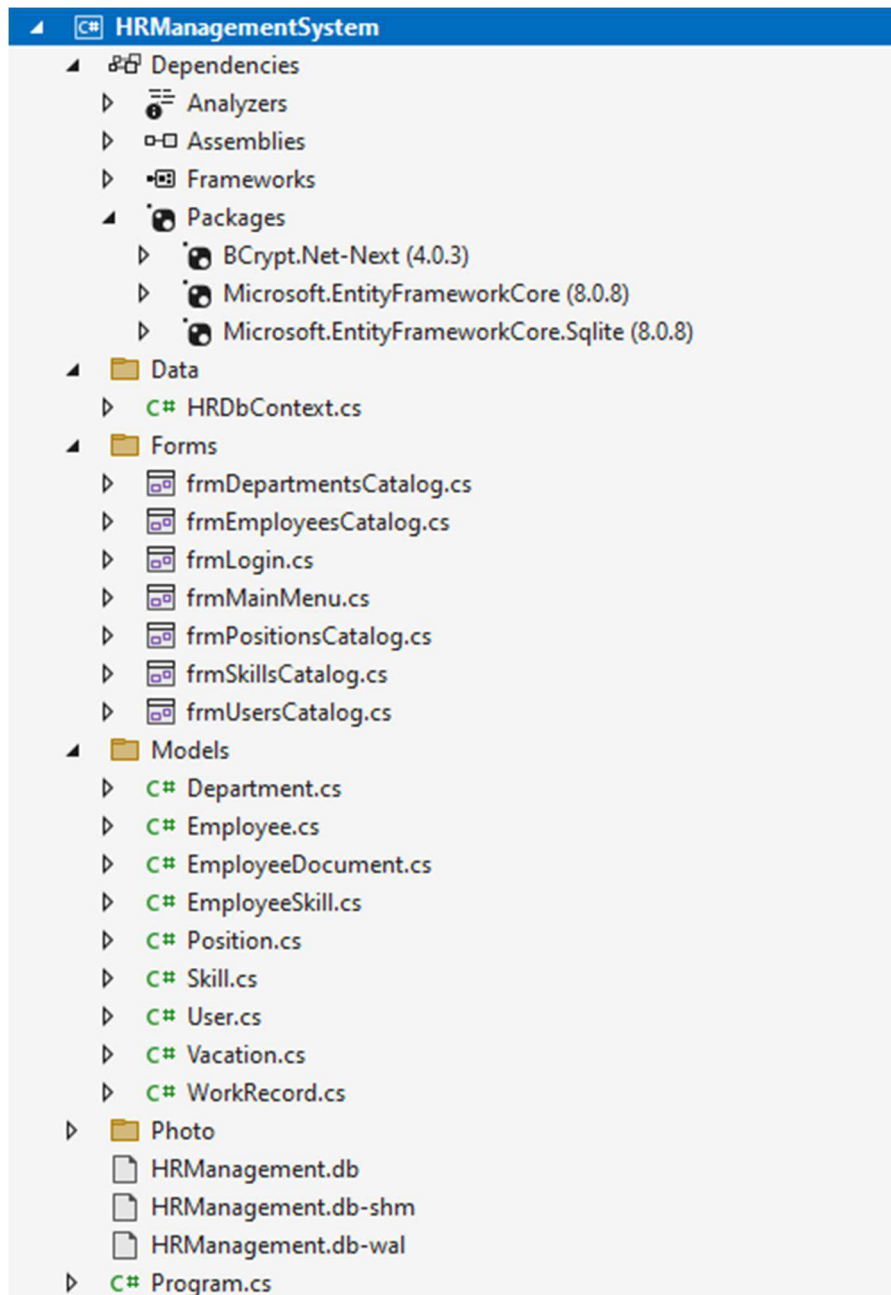


Рисунок 2.4.1 – Склад проекту

Проект складається з кількох основних компонентів, організованих у Visual Studio як рішення з одним проектом типу Windows Forms App (.NET). Структура проекту включає наступні елементи:

1. **Моделі даних:** Класи, які відображають таблиці бази даних, розташовані в папці Models.

Employee: Модель для таблиці Employees, що містить дані про співробітників (ідентифікатор, ім'я, прізвище, дату народження, зарплату, посаду, департамент тощо).

User: Модель для таблиці Users, що включає дані про користувачів (ідентифікатор, ім'я користувача, хеш пароля, роль, пов'язаний співробітник).

Position, Department, Skill, EmployeeSkill, UserAction: Моделі для відповідних таблиць, що зберігають посади, департаменти, навички, зв'язки співробітників із навичками та лог дій.

2. **Контекст бази даних:** Клас HRDbContext, розташований у папці Data, успадкований від DbContext. Він містить набори DbSet для кожної моделі та конфігурацію зв'язків, індексів і обмежень через метод OnModelCreating. Клас налаштовує підключення до SQLite за допомогою рядка з'єднання, що вказує на локальний файл HRManagement.db.
3. **Форми:** Класи форм, розташовані в корені проекту, реалізують графічний інтерфейс і логіку обробки подій. Кожна форма відповідає за певний аспект управління даними. Основні форми описано нижче.
4. **Допоміжні класи:** Статичний клас frmLogin із властивістю CurrentUser для зберігання даних поточного користувача після автентифікації. Ця властивість використовується для обмеження доступу та логування дій.
5. **Залежності:** Проект включає NuGet-пакети Microsoft.EntityFrameworkCore і Microsoft.EntityFrameworkCore.Sqlite (версія 8.0.x) для роботи з базою даних. Файл бази даних HRManagement.db створюється автоматично в директорії виконання програми.

Проект компілюється у вигляді виконуваного файлу (.exe), який містить усі необхідні бібліотеки та файл бази даних, що забезпечує портативність системи.

					ДР.122.042.034.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		24

Форми системи

Система включає три основні форми, які реалізують ключовий функціонал. Кожна форма спроектована в дизайнері Visual Studio та має відповідний клас із логікою обробки подій.

frmLogin Форма автентифікації (**frmLogin**) користувача, яка відображається при запуску програми. Вона містить текстові поля для введення імені користувача (txtUsername) і пароля (txtPassword), а також кнопки "Увійти" (btnLogin) і "Вийти" (btnCancel).

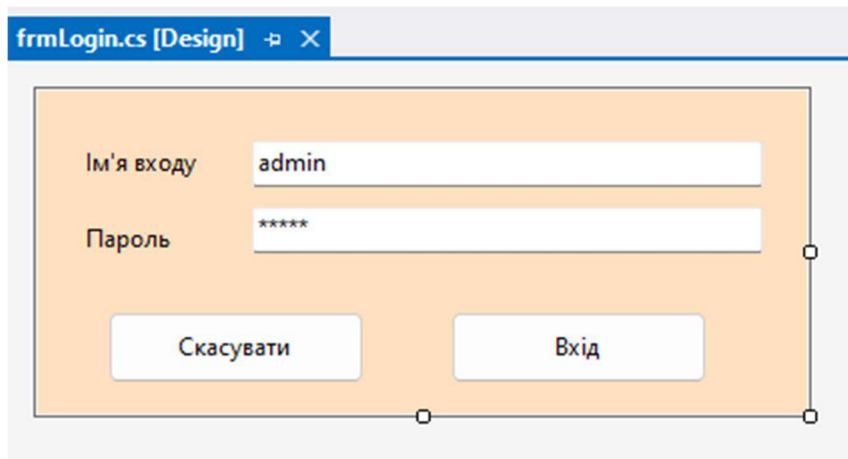


Рисунок 2.4.2 – Форма автентифікації

Перевіряє введені дані, обчислюючи SHA-256 хеш пароля та порівнюючи його з даними в таблиці Users. У разі успішної автентифікації зберігає дані користувача у статичній властивості CurrentUser і відкриває головне меню або форму frmEmployeesCatalog. У разі невдачі відображає повідомлення про помилку.

Основні методи:

- btnLogin_Click: Виконує асинхронну перевірку даних через HRDbContext і оновлює поле LastLogin у таблиці Users.
- ComputeSHA256Hash: Хешує пароль для порівняння.

Використовує асинхронні запити для швидкої перевірки даних і обробляє винятки для відображення зрозумілих повідомлень про помилки.

Форма для управління даними співробітників (**frmEmployeesCatalog**), яка дозволяє переглядати, додавати, редагувати та видаляти записи. Вона

					ДР.122.042.034.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		25

містить DataGridView (dgvEmployees) для відображення списку співробітників, текстові поля для введення особистих даних (txtFirstName, txtLastName тощо), комбіновані списки (cmbPosition, cmbDepartment, cmbSkill) для вибору посад, департаментів і навичок, а також кнопки для керування (btnAdd, btnEdit, btnSave, btnCancel, btnDelete).

Рисунок 2.4.3 – Каталог співробітників

Завантажує список співробітників із пов'язаними даними (посади, департаменти, навички), підтримує асинхронне збереження та валідацію даних (унікальність email, обмеження довжини, формат дати). Дозволяє додавати фотографії співробітників і керувати їхніми навичками через dgvSkills.

Основні методи:

- LoadEmployees: Асинхронно завантажує список співробітників у dgvEmployees.
- LoadComboBoxes: Заповнює комбіновані списки даними з таблиць Positions, Departments і Skills.

- btnSave_Click: Валідує та зберігає дані співробітника, включаючи навички, із логуванням дії.
- dgvEmployees_CellClick: Завантажує дані вибраного співробітника у поля форми.

Використовує BeginInvoke для синхронізації асинхронних запитів із інтерфейсом, обробляє помилки через винятки та забезпечує транзакційне збереження.

Форма для управління користувачами системи (frmUsersCatalog), яка дозволяє створювати, редагувати, видаляти та переглядати облікові записи. Вона включає DataGridView (dgvUsers) для списку користувачів, текстові поля (txtUsername, txtPassword), комбіновані списки (cmbEmployee, cmbRole) для вибору співробітника та ролі, прапорець (chkIsActive) для статусу активності та кнопки керування (btnAdd, btnEdit, btnSave, btnCancel, btnDelete).

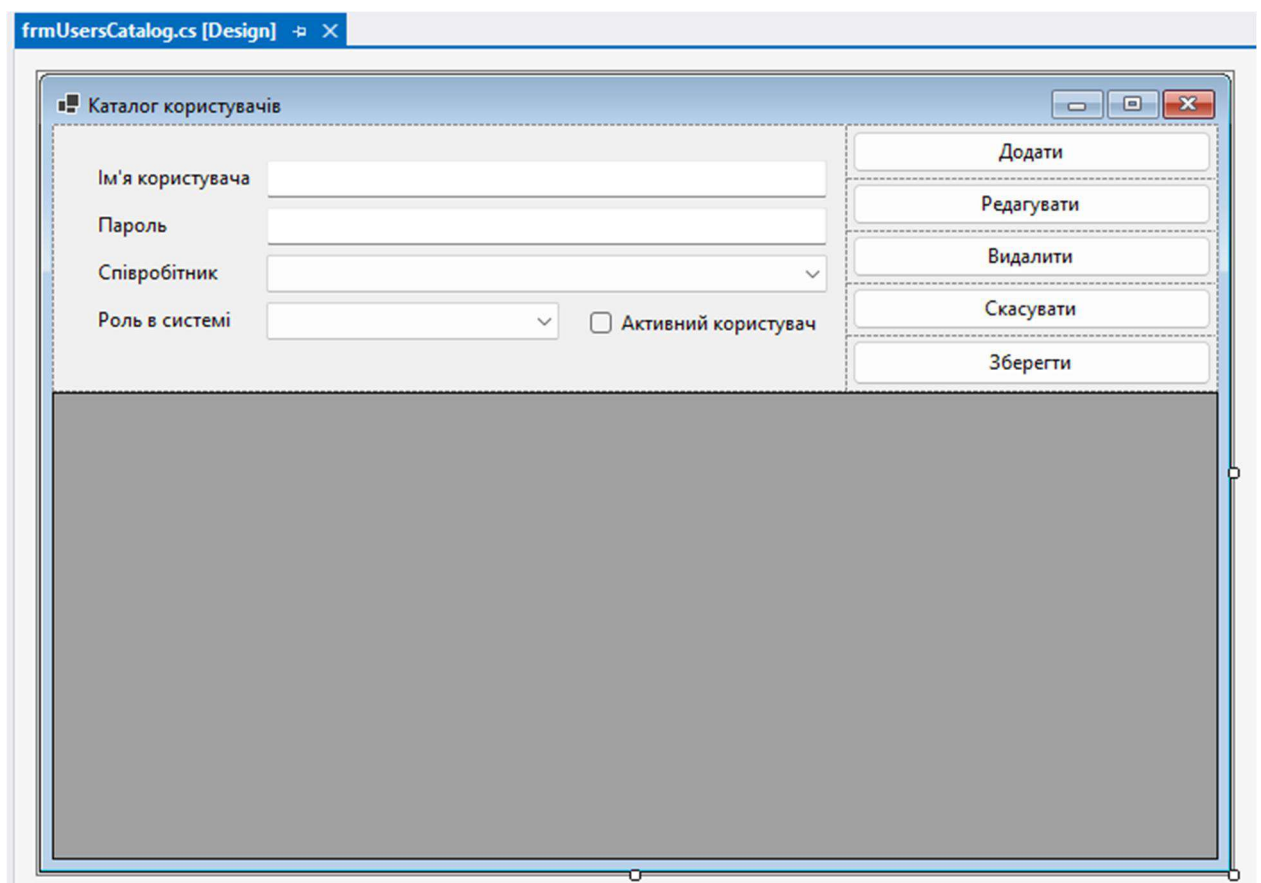


Рисунок 2.4.4 – Каталог користувачів системи

Завантажує список користувачів із пов'язаними співробітниками, підтримує хешування паролів за допомогою SHA-256, валідує унікальність Username і EmployeeId. Доступ до форми обмежено для користувачів із роллю адміністратора (Role = 0).

Основні методи:

- LoadUsers: Асинхронно завантажує список користувачів із відображенням імен співробітників і ролей.
- LoadComboBoxes: Заповнює cmbEmployee активними співробітниками та cmbRole статичним списком ролей.
- btnSave_Click: Валідує та зберігає дані користувача, включаючи хешування пароля, із логуванням дії.
- dgvUsers_CellClick: Завантажує дані вибраного користувача, очищаючи поле пароля для безпеки.
- ComputeSHA256Hash: Обчислює хеш пароля.

Забороняє видалення поточного користувача, використовує асинхронні запити та обробляє винятки для надійності.

Програмна реалізація базується на асинхронному програмуванні (async/await) для всіх операцій із базою даних, що забезпечує швидкодію та чуйність інтерфейсу. Використання Entity Framework Core спрощує роботу з даними, дозволяючи оперувати об'єктами замість прямих SQL-запитів. Кожна форма містить обробку винятків із відображенням зрозумілих повідомлень для користувача та логуванням у консоль для діагностики.

Інтерфейс форм розроблено з урахуванням ергономіки: елементи керування згруповано логічно, а валідація даних виконується перед збереженням, щоб запобігти помилкам. Наприклад, у frmEmployeesCatalog перевірка формату email і дат народження/прийому виконується до збереження, а в frmUsersCatalog додатково перевіряється мінімальна довжина пароля. Файл бази даних HRManagement.db автоматично створюється в директорії програми, що спрощує розгортання.

					ДР.122.042.034.ПЗ	Арк.
						28
Змн.	Арк.	№ докум.	Підпис	Дата		

Таким чином, програмна реалізація системи включає чітко структурований проект із модульними компонентами, які забезпечують управління персоналом і користувачами. Форми frmLogin, frmEmployeesCatalog і frmUsersCatalog охоплюють основний функціонал, забезпечуючи зручний інтерфейс, безпечну обробку даних і надійну роботу з базою даних.

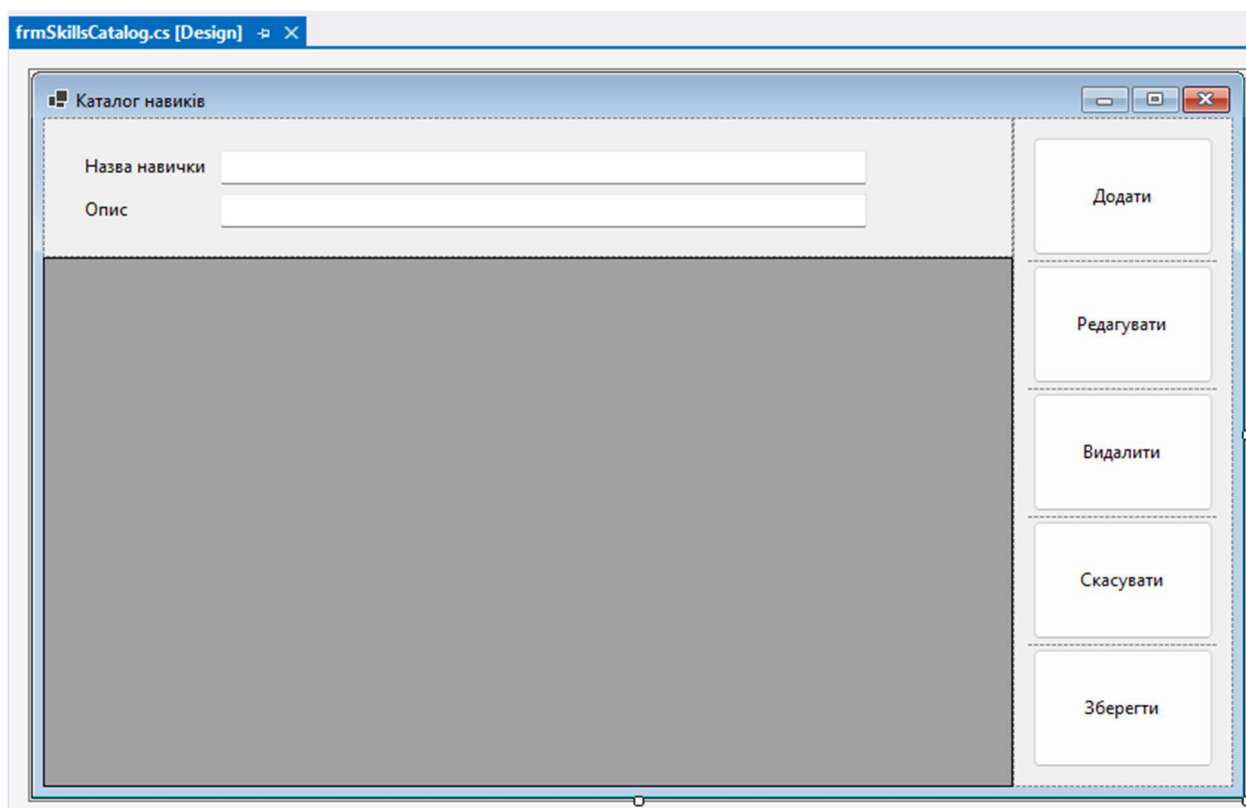


Рисунок 2.4.5 – Каталог навиків

frmPositionsC...og.cs [Design] ➤ ✕

Каталог посад

Назва посади

Код Ставка ☐ менеджер

Опис

Рисунок 2.4.6 – Каталог посад

frmDepartmen...g.cs [Design] ➤ ✕

Каталог відділів

Назва

Код

Опис

Відділ

Менеджер

Рисунок 2.4.7 – Каталог відділів

Висновки до розділу 2

Розробка системи управління персоналом на підприємстві, описана в другому розділі, дозволила створити функціональний програмний продукт, який відповідає вимогам автоматизації кадрового обліку та управління користувачами. Проектування системи базувалося на чіткому виборі алгоритмів, структурі бази даних і класів, що забезпечило ефективність, надійність і зручність використання.

Вибір алгоритмів, таких як асинхронні запити через Entity Framework Core, хешування SHA-256 для паролів і транзакційна обробка даних, забезпечив оптимальну швидкість та безпеку. Асинхронне програмування дозволило уникнути блокування інтерфейсу користувача, а оптимізовані LINQ-запити зменшили навантаження на базу даних SQLite, що є особливо важливим для локальних додатків із обмеженими ресурсами.

База даних, спроектована з використанням SQLite, включає таблиці для співробітників, користувачів, посад, департаментів, навичок і логування дій, із підтримкою зовнішніх ключів та унікальних індексів. Використання Entity Framework Core спростило взаємодію з даними, забезпечило цілісність і дозволило гнучко налаштовувати зв'язки між таблицями. Легка та портативна природа SQLite відповідає потребам малих і середніх організацій, забезпечуючи простоту розгортання та резервного копіювання.

Спроектовані класи, включаючи моделі даних (Employee, User, UserAction тощо), контекст бази даних (HRDbContext) і класи форм (frmLogin, frmEmployeesCatalog, frmUsersCatalog), реалізують чіткий розподіл обов'язків. Моделі даних ізольовано від логіки інтерфейсу, що підвищує модульність, а асинхронні методи в класах форм забезпечують продуктивність. Допоміжні класи, такі як frmLogin із властивістю CurrentUser, підтримують автентифікацію та обмеження доступу.

Програмна реалізація включає три основні форми, які охоплюють автентифікацію, управління співробітниками та користувачами. Кожна форма містить валідацію даних, обробку винятків і логування дій, що підвищує

					ДР.122.042.034.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		31

надійність і зручність. Структура проекту, побудована на Windows Forms і .NET 8, забезпечує портативність і економічну доцільність, а інтеграція з SQLite дозволяє працювати без додаткового серверного програмного забезпечення.

Другий розділ демонструє комплексний підхід до проектування та розробки системи, що поєднує ефективні алгоритми, продуману структуру бази даних і модульну архітектуру класів. Це створює надійну основу для автоматизації кадрових процесів, яка може бути розширена в майбутньому для підтримки додаткових функцій.

					ДР.122.042.034.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		32

РОЗДІЛ 3. КЕРІВНИЦТВО КОРИСТУВАННЯ ПРОГРАМОЮ

3.1. Мінімальні вимоги до системи

Для коректної роботи системи управління персоналом на підприємстві, розробленої з використанням Windows Forms, .NET 8 та SQLite, необхідно забезпечити відповідність апаратного та програмного забезпечення мінімальним системним вимогам. Оскільки система є настільним додатком із локальною базою даних, вона має низькі вимоги до ресурсів, що робить її доступною для використання на більшості сучасних офісних комп'ютерів, характерних для малих і середніх організацій.

Апаратні вимоги

Система може працювати на комп'ютерах із мінімальними апаратними характеристиками, що відповідають базовим офісним конфігураціям. Рекомендується:

- Процесор: Одноядерний процесор із частотою 1 ГГц або вище. Сучасні дво- чи чотирядерні процесори (наприклад, Intel Core i3 або AMD Ryzen 3) забезпечать оптимальну продуктивність.
- Оперативна пам'ять: 2 ГБ для базового використання. Для комфортної роботи з великими наборами даних (тисячі записів) рекомендується 4 ГБ або більше.
- Вільне місце на диску: 100 МБ для виконуваного файлу програми, бібліотек .NET та файлу бази даних SQLite (HRManagement.db). Додаткове місце може знадобитися для резервних копій бази даних.
- Монітор: Роздільна здатність 1024x768 або вище для комфортного відображення інтерфейсу Windows Forms.

Програмні вимоги

Система розроблена для роботи в середовищі операційної системи Windows і залежить від наявності певних програмних компонентів:

- Операційна система: Windows 10 (версія 1803 або новіша) або Windows 11. 64-розрядні версії рекомендуються для кращої продуктивності, але 32-розрядні також підтримуються.

					ДР.122.042.034.ПЗ	Арк.
						33
Змн.	Арк.	№ докум.	Підпис	Дата		

- .NET Runtime: .NET 8 Runtime (версія 8.0 або новіша), необхідний для виконання програми. Цей компонент автоматично встановлюється разом із програмою або може бути завантажений із офіційного сайту Microsoft.
- Драйвери SQLite: Вбудовані в бібліотеку Microsoft.EntityFrameworkCore.Sqlite, тому додаткове встановлення не потрібне.

Додаткові вимоги

- Права доступу: Для запуску програми та створення/модифікації файлу бази даних (HRManagement.db) необхідні права на запис у директорію, де розташована програма. Зазвичай це директорія виконання (наприклад, C:\Program Files\HRManagementSystem).
- Мережа: Система працює автономно і не вимагає підключення до Інтернету, що робить її придатною для використання в ізольованих офісних середовищах.
- Периферійні пристрої: Миша та клавіатура для взаємодії з інтерфейсом. Для завантаження фотографій співробітників може знадобитися сканер або цифровий пристрій для отримання зображень у форматі JPEG/PNG.

Рекомендації для оптимальної роботи

Для забезпечення максимальної продуктивності, особливо при роботі з великими обсягами даних (понад 1000 співробітників), рекомендується використовувати комп'ютер із чотириядерним процесором (Intel Core i5 або еквівалент), 8 ГБ оперативної пам'яті та SSD-диском для швидшого доступу до файлу бази даних. Оновлення операційної системи до останніх версій Windows 10/11 і .NET 8 забезпечить сумісність із майбутніми оновленнями програми.

					ДР.122.042.034.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		34

3.2. Інтерфейс користувача

Інтерфейс користувача системи управління персоналом на підприємстві розроблено з використанням технології Windows Forms, що забезпечує зручність, інтуїтивність і простоту взаємодії для працівників кадрових служб із різним рівнем технічної підготовки. Інтерфейс програми складається з кількох форм, кожна з яких відповідає за певний аспект управління даними: автентифікацію, доступ до основного меню, управління співробітниками, користувачами, посадами, відділами та навичками. Всі форми мають уніфікований дизайн із логічно згрупованими елементами керування, чіткими написами та підтримкою української мови, що відповідає вимогам локалізації. Нижче наведено детальний опис інтерфейсу кожної форми.

Форма автентифікації є першою, яку бачить користувач при запуску програми. Вона має компактний розмір (384x161 пікселів) і оформлена у світло-помаранчевому кольорі (BackColor: RGB(255, 224, 192)), що створює дружній вигляд. Форма не має рамки (FormBorderStyle: None), що надає їй сучасного вигляду.

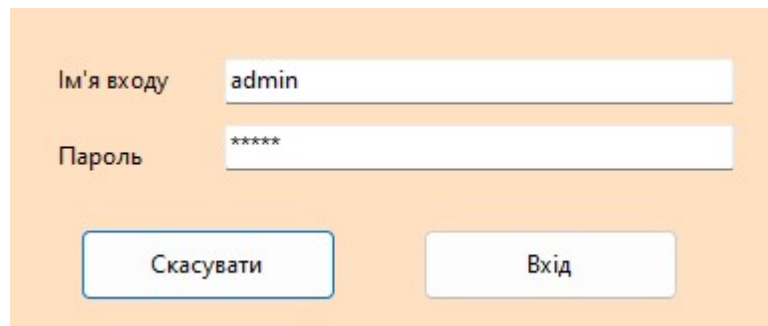


Рисунок 3.2.1 – Форма автентифікації

Два текстові поля: txtUsername для введення імені користувача та txtPassword (з символом маскування "*") для пароля.

Два написи: "Ім'я входу" та "Пароль" для позначення полів.

Дві кнопки: btnLogin ("Вхід") для автентифікації та btnCancel ("Скасувати") для закриття програми.

Форма дозволяє користувачу увійти в систему, вводячи облікові дані. У разі помилки автентифікації відображається повідомлення, а при успішному

вході відкривається головне меню (frmMainMenu). Форма розташована по центру екрана (StartPosition: CenterScreen) для зручності.

Форма головного меню слугує центральним навігаційним елементом, надаючи доступ до всіх каталогів системи. Її розмір становить 741x317 пікселів, а заголовок форми містить назву системи та ім'я розробника ("Система управління персоналом на підприємстві (Харчук Іван Олексійович)").

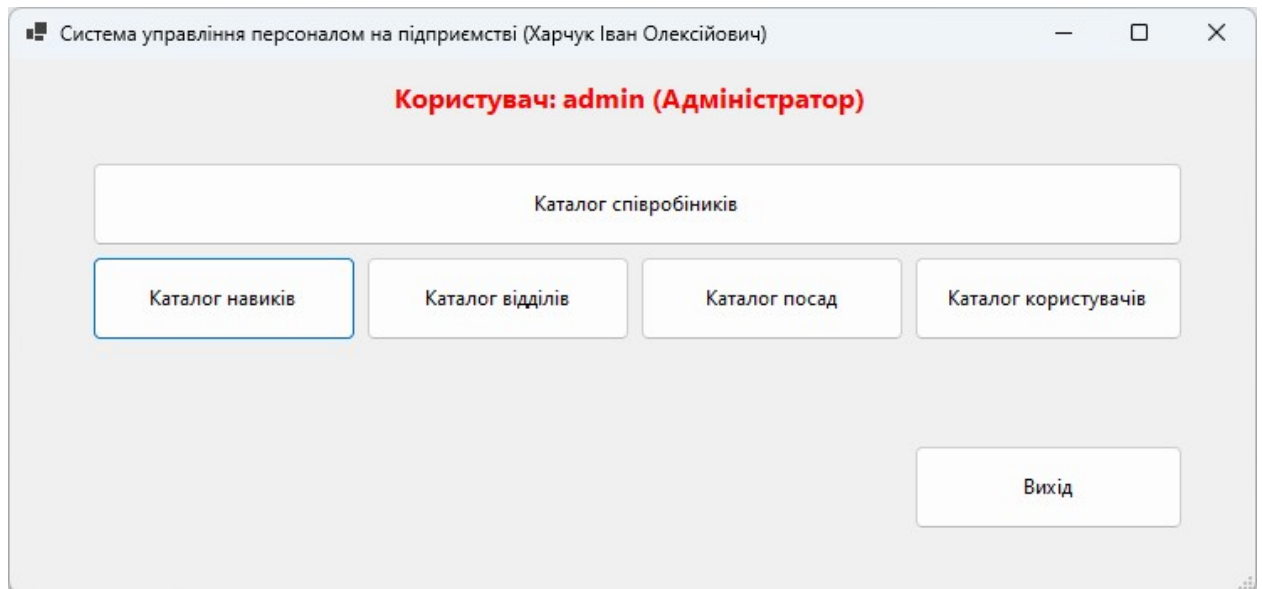


Рисунок 3.2.2 – Головного меню

Елементи керування:

- Напис відображає роль поточного користувача (наприклад, "Адміністратор") у верхній частині форми.
- Панель із сіткою 4x4 містить шість кнопок:
- "Каталог співробітників"– займає всю ширину першого рядка для швидкого доступу.
- "Каталог навиків", "Каталог відділів", "Каталог посад", "Каталог користувачів" – розташовані в другому рядку.
- "Вихід"– у правому нижньому куті для завершення сеансу.

Кожна кнопка відкриває відповідну форму каталогу, а кнопка виходу повертає користувача до форми автентифікації. Форма розташована по центру екрана, а доступ до "Каталог користувачів" обмежено для адміністраторів.

Форма управління співробітниками є основною для кадрового обліку. Її розмір становить 1093x605 пікселів, що забезпечує достатньо простору для введення даних. Форма поділена на вкладки для анкетних даних і навичок.

Каталог співробітників

Анкетні дані

Навики співробітника

Прізвище

Коваленко

Адреса

м. Київ, вул. Хрещатик, 25

Ім'я

Андрій

Електронна пошта

andriy.kovalenko@company.com

По батькові

Олександрович

Дата народження

30 листопада 1978 р.

Телефон

+380503456789

Відділ

IT-відділ

Дата прийому

20 травня 2011 р.

Посада

Технічний директор

Зарплата

45000,00

Статус активності

☒

Додати

Редагувати

Видалити

Завантажити фото

Скасувати

Зберегти

ПІБ

Дата народження

Дата прийому

Посада

Департамент

Зарплата

Електронна пошта

Телефон

Іван Петренко Іванович

1975-05-15

2010-01-10

Генеральний директор

Керівництво

50000,0

ivan.petrenko@company.com

+3805012

Олена Сидорова Миколаївна

1980-08-22

2012-03-15

Фінансовий директор

Фінансовий відділ

40000,0

olena.sydorova@company.com

+3805023

Андрій Коваленко Олександрович

1978-11-30

2011-05-20

Технічний директор

IT-відділ

45000,0

andriy.kovalenko@company.com

+3805034

Марк Шевченко Володимирович

1985-04-18

2015-07-01

Менеджер з продажу

Відділ продажу

25000,0

maria.shevchenko@company.com

+3805045

Олексій Бондаренко Сергійович

1990-02-28

2018-09-10

Розробник програмного забезпечення

Розробка

22000,0

oleksiy.bondarenko@company.com

+3805056

Назар Коваль Андрійович

1987-07-12

2016-11-05

Маркетолог

Відділ маркетингу

19000,0

nataliya.koval@company.com

+3805067

Віктор Мельник Ігорович

1983-09-25

2014-02-15

Бухгалтер

Фінансовий відділ

18000,0

viktor.melnyk@company.com

+3805078

Тетяна Лисенко Олегівна

1988-12-05

2017-04-01

HR-менеджер

HR-відділ

20000,0

tetiana.lysenko@company.com

+3805089

Сергій Павленко Вікторович

1992-03-20

2019-08-15

Системний адміністратор

IT-відділ

21000,0

serhiy.pavlenko@company.com

+3805090

Анна Савченко Дмитрівна

1991-06-14

2020-01-20

Аналітик даних

Розробка

23000,0

anna.savchenko@company.com

+3805011

wefwef qefdwe wefw

1995-04-19

2025-04-19

Системний адміністратор

IT-відділ

4444,0

wd@fgg.net

103

Рисунок 3.2.3 – Перша вкладка вікна «Каталог співробітників»

Каталог співробітників

Анкетні дані

Навики співробітника

Навик

Digital Marketing

Рівень володіння

1

Додати навик

Видалити навик

Навичка

Рівень володіння

Керування проектами

5

Бази даних

4

Програмування на C#

4

ПІБ

Дата народження

Дата прийому

Посада

Департамент

Зарплата

Електронна пошта

Телефон

Іван Петренко Іванович

1975-05-15

2010-01-10

Генеральний директор

Керівництво

50000,0

ivan.petrenko@company.com

+3805012

Олена Сидорова Миколаївна

1980-08-22

2012-03-15

Фінансовий директор

Фінансовий відділ

40000,0

olena.sydorova@company.com

+3805023

Андрій Коваленко Олександрович

1978-11-30

2011-05-20

Технічний директор

IT-відділ

45000,0

andriy.kovalenko@company.com

+3805034

Марк Шевченко Володимирович

1985-04-18

2015-07-01

Менеджер з продажу

Відділ продажу

25000,0

maria.shevchenko@company.com

+3805045

Олексій Бондаренко Сергійович

1990-02-28

2018-09-10

Розробник програмного забезпечення

Розробка

22000,0

oleksiy.bondarenko@company.com

+3805056

Назар Коваль Андрійович

1987-07-12

2016-11-05

Маркетолог

Відділ маркетингу

19000,0

nataliya.koval@company.com

+3805067

Віктор Мельник Ігорович

1983-09-25

2014-02-15

Бухгалтер

Фінансовий відділ

18000,0

viktor.melnyk@company.com

+3805078

Тетяна Лисенко Олегівна

1988-12-05

2017-04-01

HR-менеджер

HR-відділ

20000,0

tetiana.lysenko@company.com

+3805089

Сергій Павленко Вікторович

1992-03-20

2019-08-15

Системний адміністратор

IT-відділ

21000,0

serhiy.pavlenko@company.com

+3805090

Анна Савченко Дмитрівна

1991-06-14

2020-01-20

Аналітик даних

Розробка

23000,0

anna.savchenko@company.com

+3805011

wefwef qefdwe wefw

1995-04-19

2025-04-19

Системний адміністратор

IT-відділ

4444,0

wd@fgg.net

103

Рисунок 3.2.4 – Друга вкладка вікна «Каталог співробітників»

Змн.

Арк.

№ докум.

Підпис

Дата

ДР.122.042.034.ПЗ

Арк.

37

Вкладка "Анкетні дані":

- Текстові поля для введення особистих даних, контактів і зарплати.
- Комбіновані списки для вибору посади та відділу.
- Елементи вибору дати для дати народження та прийому.
- Прапорець для позначення активності співробітника.
- Зображення для відображення фотографії та кнопка для її завантаження.
- Панель із кнопками btnAdd, btnEdit, btnDelete, btnCancel, btnSave і btnLoadPhoto.

Вкладка "Навики співробітника":

- Таблиця для відображення навичок співробітника.
- Панель із комбінованим списком, числовим полем (від 1 до 5) для рівня володіння, кнопками btnAddSkill і btnRemoveSkill.
- Таблиця у нижній частині форми для відображення списку співробітників.

Форма дозволяє додавати, редагувати, видаляти співробітників, завантажувати їхні фотографії та керувати навичками. Валідація даних (наприклад, формат email, унікальність телефону) виконується перед збереженням. Таблиця підтримує вибір запису через клік (CellClick).

Форма управління користувачами (713x450 пікселів) призначена для адміністраторів і дозволяє керувати обліковими записами.

Ім'я користувача	Співробітник	Роль	Останній вхід	Статус
admin	Іван Петренко	Адміністратор	19.04.2025 17:06	Активний
hr_manager	Тетяна Лисенко	HR	17.04.2025 20:24	Активний
finance_manager	Олена Сидорова	Менеджер	17.04.2025 20:24	Активний
it_manager	Андрій Коваленко	Менеджер	17.04.2025 20:24	Активний
sales_manager	Марк Шевченко	Менеджер	17.04.2025 20:24	Активний
dev1	Олексій Бондаренко	Співробітник	17.04.2025 20:24	Активний
marketing1	Назар Коваль	Співробітник	17.04.2025 20:24	Активний
accountant1	Віктор Мельник	Співробітник	17.04.2025 20:24	Активний
sysadmin1	Сергій Павленко	Співробітник	17.04.2025 20:24	Активний
analyst1	Анна Савченко	Співробітник	17.04.2025 20:24	Активний

Рисунок 3.2.5 – Форма управління користувачами

Елементи керування:

- Текстовими полями txtUsername і txtPassword (з маскуванням "*") для імені користувача та пароля.
- Комбінованими списками cmbEmployee (співробітники) і cmbRole (полі: адміністратор, менеджер, користувач).
- Прапорцем chkIsActive для статусу активності.
- Написами для позначення полів.
- Панель tableLayoutPanel1 із кнопками btnAdd, btnEdit, btnDelete, btnCancel, btnSave.
- Таблиця dgvUsers для відображення списку користувачів.

Форма підтримує створення, редагування, видалення користувачів із валідацією унікальності імені та хешуванням паролів (SHA-256). Вибір запису в таблиці оновлює поля форми.

Форма управління відділами (800x450 пікселів) дозволяє керувати структурою організації.

Назва	Код	Опис	Відділ	Менеджер відділу
Керівництво	MANAGEMENT	Вище керівництво компанії	Немає	Іван Петренко
Фінансовий відділ	FINANCE	Фінанси та бухгалтерія	Керівництво	Олена Сидорова
ІТ-відділ	IT	Інформаційні технології	Керівництво	Андрій Коваленко
Відділ продажу	SALES	Продажі та обслуговування клієнтів	Керівництво	Марк Шевченко
Відділ маркетингу	MARKETING	Маркетинг та реклама	Керівництво	Назар Коваль
HR-відділ	HR	Управління персоналом	Керівництво	Тетяна Лисенко
Підтримка	SUPPORT	Технічна підтримка клієнтів	ІТ-відділ	Сергій Павленко
Розробка	DEV	Розробка програмного забезпечення	ІТ-відділ	Андрій Коваленко

Рисунок 3.2.6 – Форма управління відділами

Елементи керування:

- Текстовими полями для назви, коду та опису відділу.
- Комбінованими списками відділів і менеджер.
- Написами для позначення полів.
- Панель із кнопками btnAdd, btnEdit, btnDelete, btnCancel, btnSave.
- Таблиця для списку відділів.

Дозволяє створювати ієрархію відділів, призначати менеджерів і зберігати описи. Вибір запису в таблиці оновлює поля.

Форма управління посадами (800x450 пікселів) призначена для ведення каталогу посад.

Каталог посад

Назва посади

Код Ставка ☐ менеджер

Опис

Назва	Код	Ставка (грн)	Опис	Менеджер
Генеральний директор	CEO	50000,0	Керівник компанії	Так
Фінансовий директор	CFO	40000,0	Керівник фінансового відділу	Так
Технічний директор	CTO	45000,0	Керівник технічного відділу	Так
Менеджер з продажу	SM	25000,0	Керівник відділу продажу	Так
Розробник програмного забезпечення	DEV	20000,0	Розробка програмного забезпечення	Ні
Маркетолог	MKT	18000,0	Проведення маркетингових кампаній	Ні
Бухгалтер	ACC	17000,0	Ведення бухгалтерського обліку	Ні
HR-менеджер	HR	19000,0	Управління персоналом	Ні
Системний адміністратор	SYS	21000,0	Обслуговування IT-інфраструктури	Ні
Аналітик даних	DA	22000,0	Аналіз бізнес-даних	Ні

Додати

Редагувати

Видалити

Скасувати

Зберегти

Рисунок 3.2.7 – Форма управління посадами

Елементи керування:

- Текстовими полями для назви, коду, опису та базової ставки.
- Прапорцем для позначення управлінської посади.
- Написами для позначення полів.
- Панель із кнопками btnAdd, btnEdit, btnDelete, btnCancel, btnSave.
- Таблиця для списку посад.

Підтримує створення, редагування, видалення посад із валідацією унікальності коду та ставки.

Форма управління навичками (800x450 пікселів) дозволяє вести каталог навичок.

Каталог навичок

Назва навички: Програмування на C#

Опис: Розробка програмного забезпечення на C#

ID	Назва навички	Опис
1	Програмування на C#	Розробка програмного забезпечення на C#
2	Керування проектами	Планування та управління проектами
3	Фінансовий аналіз	Аналіз фінансової звітності
4	Веб-розробка	Розробка веб-додатків
5	Бази даних	Робота з реляційними базами даних
6	Digital Marketing	Інтернет-маркетинг та соціальні мережі
7	HR-менеджмент	Управління персоналом
8	Бухгалтерія	Ведення бухгалтерського обліку
9	Клієнтська підтримка	Робота з клієнтами
10	Аналіз даних	Аналіз та візуалізація даних

Додати

Редагувати

Видалити

Скасувати

Зберегти

Рисунок 3.2.8 – Форма управління навичками

Елементи керування:

- Текстовими полями для назви та опису навички.
- Написами для позначення полів.
- Панель із кнопками btnAdd, btnEdit, btnDelete, btnCancel, btnSave.
- Таблиця для списку навичок.

Дозволяє додавати, редагувати, видаляти навички, які потім використовуються в frmEmployeesCatalog.

Загальні особливості інтерфейсу

- Усі форми мають стандартний розмір шрифту (7 pt), розташовані по центру екрана, із чіткими написами українською мовою. Кнопки керування (Додати, Редагувати, Видалити, Зберегти, Скасувати) розміщені в панелях TableLayoutPanel для логічного групування.
- DataGridView у всіх каталогах підтримують вибір повного рядка (FullRowSelect), що полегшує навігацію. Колонки автоматично адаптуються до вмісту.

- Поля вводу мають валідацію (наприклад, формат email, мінімальна довжина пароля), а помилки відображаються через спливні повідомлення. Паролі маскуються символом "*".
- Елементи керування розташовані для зручного доступу, із достатніми відступами та логічним порядком табуляції. Форми підтримують обробку подій закриття (FormClosing) для збереження даних.

Інтерфейс користувача є інтуїтивно зрозумілим, з мінімальною кількістю елементів, що зменшує час навчання. Уніфікована структура форм і чітке позначення функцій сприяють ефективному управлінню персоналом, а підтримка української мови забезпечує зручність для локальних користувачів.

3.3. Інструкція для роботи з програмою

Система управління персоналом на підприємстві, розроблена з використанням Windows Forms, .NET 8 та SQLite, призначена для автоматизації кадрового обліку, управління користувачами, посадами, відділами та навичками. Ця інструкція описує послідовність дій для роботи з програмою, включаючи автентифікацію, навігацію, управління даними та завершення роботи. Інтерфейс програми є інтуїтивно зрозумілим і розрахований на користувачів із базовими навичками роботи з комп'ютером.

1. Запуск програми та автентифікація

1. Запуск програми:

Двічі клацніть на виконуваному файлі HRManagementSystem.exe, розташованому в директорії встановлення (наприклад, C:\Program Files\HRManagementSystem).

Програма відкриється з формою автентифікації.

2. Вхід у систему:

У полі для імені користувача введіть ваше ім'я (наприклад, admin для адміністратора).

					ДР.122.042.034.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		43

У полі для пароля введіть пароль (за замовчуванням admin для адміністратора). Пароль відображається зірочками для безпеки.

Натисніть кнопку Вхід.

У разі успішної автентифікації відкриється головне меню. Якщо дані введено неправильно, з'явиться повідомлення про помилку.

Для виходу з програми натисніть Скасувати, що закrije додаток.

Якщо ви забули пароль або не маєте облікового запису, зверніться до адміністратора системи для створення нового користувача.

2. Навігація в головному меню

Головне меню містить напис із роллю поточного користувача (наприклад, "Адміністратор") у верхній частині та панель із кнопками для доступу до каталогів.

1. Доступ до каталогів:

Натисніть Каталог співробітників для управління даними співробітників.

Натисніть Каталог навиків, Каталог відділів або Каталог посад для управління відповідними довідниками.

Натисніть Каталог користувачів для управління обліковими записами (доступно лише адміністраторам).

Натисніть Вихід для повернення до форми автентифікації.

Доступ до певних функцій залежить від ролі користувача (адміністратор, менеджер, звичайний користувач).

3. Робота з каталогом співробітників

Форма поділена на вкладки: Анкетні дані (для особистих даних) і Навики співробітника (для управління навичками).

У нижній частині розташовано таблицю зі списком співробітників.

Панель із кнопками Додати, Редагувати, Видалити, Зберегти, Скасувати, Завантажити фото розташована внизу вкладки Анкетні дані.

1. Перегляд даних:

					ДР.122.042.034.ПЗ	Арк.
						44
Змн.	Арк.	№ докум.	Підпис	Дата		

У таблиці клацніть на рядок, щоб вибрати співробітника. Його дані автоматично відобразяться у полях вкладки Анкетні дані та таблиці навичок на вкладці Навики співробітника.

2. Додавання співробітника:

Натисніть Додати.

Заповніть поля на вкладці Анкетні дані:

- Прізвище, ім'я, по батькові.
- Дати народження та прийому.
- Відділ і посаду з відповідних списків.
- Зарплату, email, телефон, адресу.
- Позначте статус активності (за замовчуванням увімкнено).

Для додавання фото натисніть Завантажити фото, виберіть файл (JPEG/PNG) і підтвердіть.

Перейдіть на вкладку Навики співробітника, виберіть навичку зі списку, вкажіть рівень володіння (від 1 до 5) і натисніть Додати навик.

Натисніть Зберегти. У разі помилок (наприклад, некоректний email) з'явиться повідомлення.

3. Редагування співробітника:

Виберіть співробітника в таблиці, натисніть Редагувати.

Змініть потрібні поля, додайте/видаліть навички.

Натисніть Зберегти або Скасувати для відміни змін.

4. Видалення співробітника:

Виберіть співробітника, натисніть Видалити і підтвердіть дію. Якщо співробітник пов'язаний із користувачем, видалення буде заблоковано.

4. Робота з каталогом користувачів

Форма доступна лише адміністраторам. Містить таблицю зі списком користувачів і панель із полями для введення імені, пароля, співробітника, ролі та статусу активності.

					ДР.122.042.034.ПЗ	Арк.
						45
Змн.	Арк.	№ докум.	Підпис	Дата		

1. Додавання користувача:

Натисніть Додати.

Введіть ім'я користувача, пароль (мінімум 6 символів), виберіть співробітника, роль (адміністратор, менеджер, користувач) і статус активності.

Натисніть Зберегти. Пароль автоматично захищається.

2. Редагування користувача:

Виберіть користувача в таблиці, натисніть Редагувати.

Змініть потрібні поля (поле пароля очищається для безпеки, введіть новий за потреби).

Натисніть Зберегти або Скасувати.

3. Видалення користувача:

Виберіть користувача, натисніть Видалити і підтвердіть. Поточний користувач не може бути видалений.

Ім'я користувача має бути унікальним.

5. Робота з каталогом відділів

Містить таблицю зі списком відділів і панель із полями для назви, коду, опису, батьківського відділу та менеджера.

1. Додавання відділу:

Натисніть Додати.

Введіть назву, код, опис, виберіть батьківський відділ (необов'язково) і менеджера (необов'язково).

Натисніть Зберегти.

2. Редагування та видалення:

Виберіть відділ, натисніть Редагувати для зміни або Видалити для видалення. Видалення заборонено, якщо відділ пов'язаний із співробітниками.

Примітка: Код відділу має бути унікальним.

6. Робота з каталогом посад

Містить таблицю зі списком посад і панель із полями для назви, коду, опису, базової ставки та позначки управлінської посади.

					ДР.122.042.034.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		46

1. Додавання посади:

Натисніть Додати.

Введіть назву, код, опис, базову ставку та позначте, чи є посада управлінською.

Натисніть Зберегти.

2. Редагування та видалення:

Аналогічно до відділів, використовуйте Редагувати або Видалити. Видалення заборонено для посад, пов'язаних із співробітниками.

Код посади має бути унікальним, а ставка – числовим значенням.

7. Робота з каталогом навичок

Містить таблицю зі списком навичок і панель із полями для назви та опису.

1. Додавання навички:

Натисніть Додати.

Введіть назву та опис (необов'язково).

Натисніть Зберегти.

2. Редагування та видалення:

Виберіть навичку, натисніть Редагувати або Видалити. Видалення заборонено для навичок, пов'язаних із співробітниками.

Примітка: Назва навички має бути унікальною.

8. Завершення роботи

1. Вихід із каталогу:

У будь-якій формі каталогу натисніть кнопку закриття (хрестик у правому верхньому куті), щоб повернутися до головного меню.

2. Вихід із системи:

У головному меню натисніть Вихід, щоб повернутися до форми автентифікації.

3. Закриття програми:

На формі автентифікації натисніть Скасувати або закрийте форму через хрестик.

					ДР.122.042.034.ПЗ	Арк.
						47
Змн.	Арк.	№ докум.	Підпис	Дата		

9. Рекомендації та застереження

Регулярно створюйте копії файлу бази даних (HRManagement.db), розташованого в директорії програми, для запобігання втрати даних.

Не передавайте паролі стороннім особам. Адміністратори повинні періодично змінювати паролі та деактивувати невикористовувані облікові записи.

Звичайні користувачі не мають доступу до каталогу користувачів, а менеджери можуть мати обмежений доступ до редагування даних співробітників.

У разі повідомлень про помилки (наприклад, некоректний формат email або спроба видалення пов'язаного запису) перевірте введені дані або зверніться до адміністратора.

Після внесення змін у каталоги посад, відділів чи навичок, оновіть відповідні списки в каталозі співробітників, закривши та відкривши форму.

Ця інструкція охоплює основні сценарії роботи з програмою. Для додаткових налаштувань, таких як створення звітів чи інтеграція з іншими системами, зверніться до адміністратора або розробника. Програма забезпечує ефективне управління персоналом завдяки простому інтерфейсу та надійному збереженню даних.

Висновки до розділу 3

Розділ 3 містить детальне керівництво користування програмою управління персоналом на підприємстві, яке охоплює мінімальні системні вимоги, опис інтерфейсу користувача та інструкцію для роботи з програмою. Ці матеріали забезпечують повне розуміння процесу впровадження та використання системи, що є важливим для її ефективного застосування в організаціях.

Мінімальні системні вимоги демонструють доступність програми для більшості офісних комп'ютерів. Завдяки використанню SQLite як локальної бази даних і Windows Forms на платформі .NET 8, система не потребує

					ДР.122.042.034.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		48

потужного обладнання чи складного серверного забезпечення. Вимоги, такі як 2 ГБ оперативної пам'яті, 100 МБ дискового простору та Windows 10/11, дозволяють розгорнути програму навіть на застарілих конфігураціях, що робить її економічно вигідною для малих і середніх організацій.

Інтерфейс користувача, побудований на принципах простоти та інтуїтивності, забезпечує зручну взаємодію для користувачів із різним рівнем технічної підготовки. Уніфікований дизайн форм (frmLogin, frmMainMenu, frmEmployeesCatalog, frmUsersCatalog, frmDepartmentsCatalog, frmPositionsCatalog, frmSkillsCatalog) із логічно згрупованими елементами керування, чіткими написами українською мовою та підтримкою валідації даних сприяє швидкому освоєнню системи. Використання таблиць (DataGridView), комбінованих списків і вкладок оптимізує роботу з великими обсягами інформації, а обробка помилок через спливні повідомлення підвищує надійність.

Інструкція для роботи з програмою детально описує всі етапи взаємодії: від автентифікації до управління каталогами та завершення роботи. Покрокові вказівки для додавання, редагування та видалення даних, а також рекомендації щодо безпеки та резервного копіювання забезпечують безперебійне використання системи. Обмеження доступу за ролями (адміністратор, менеджер, користувач) гарантують захист конфіденційної інформації, а логування дій підвищує контроль над змінами.

Розділ 3 підтверджує, що система управління персоналом є зручним, надійним і доступним інструментом для автоматизації кадрових процесів. Інтерфейс і інструкція адаптовані до потреб користувачів, а низькі системні вимоги забезпечують широкі можливості для впровадження. Програма готова до використання в організаціях і може бути розширена для додаткових функцій у майбутньому.

					ДР.122.042.034.ПЗ	Арк.
						49
Змн.	Арк.	№ докум.	Підпис	Дата		

ВИСНОВКИ

Розроблена в рамках дипломної роботи система управління персоналом на підприємстві є ефективним інструментом для автоматизації кадрового обліку, управління користувачами, посадами, відділами та навичками. Використання технологій Windows Forms, .NET 8, SQLite та Entity Framework Core дозволило створити портативний, швидкодіючий і економічно вигідний програмний продукт, який відповідає потребам малих і середніх організацій.

У першому розділі було проведено аналіз предметної області, визначено основні вимоги до системи та обґрунтовано вибір технологій. SQLite забезпечує легкість розгортання та портативність, .NET 8 гарантує сучасну продуктивність, а Windows Forms надає зручний і знайомий інтерфейс. Ці рішення оптимізують витрати на впровадження та підтримку системи.

Другий розділ присвячено проектуванню та реалізації системи. Розроблені алгоритми, включаючи асинхронні запити, хешування паролів (SHA-256) і транзакційну обробку, забезпечують швидкодію та безпеку. Структура бази даних із таблицями для співробітників, користувачів, посад, відділів, навичок і логів дій підтримує цілісність і гнучкість. Спроектовані класи (Employee, User, HRDbContext, класи форм) відображають модульний підхід, а асинхронна взаємодія з базою даних через Entity Framework Core мінімізує затримки. Програмна реалізація охоплює форми для автентифікації, головного меню та каталогів, які інтегрують усі функції системи.

Третій розділ містить керівництво користувача, яке включає мінімальні системні вимоги, опис інтерфейсу та інструкцію для роботи. Низькі вимоги до апаратного забезпечення (2 ГБ ОЗП, 100 МБ дискового простору, Windows 10/11) забезпечують доступність системи. Інтерфейс, побудований на принципах простоти та уніфікованості, з українською локалізацією та валідацією даних, є зручним для користувачів із різним рівнем підготовки. Інструкція детально описує всі етапи роботи, від автентифікації до управління даними, із рекомендаціями щодо безпеки та резервного копіювання.

					ДР.122.042.034.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		50

Розроблена система успішно виконує поставлені завдання: автоматизує облік персоналу, забезпечує безпечне управління користувачами та підтримує гнучке налаштування довідників. Вона готова до використання в реальних організаціях і може бути розширена для підтримки додаткових функцій, таких як звіти чи інтеграція з іншими системами. Робота підтвердила ефективність обраних технологій і методів проектування, а також практичну цінність створеного продукту для автоматизації кадрових процесів.

					ДР.122.042.034.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		51

ПЕРЕЛІК ЛІТЕРАТУРНИХ ДЖЕРЕЛ

1. Андерсон, Е. Управління персоналом: стратегії, методи, технології: монографія / Е. Андерсон. – Київ: Видавничий дім "Професіонал", 2020. – 320 с.
2. Бойко, О. В. Інформаційні системи управління персоналом: підручник / О. В. Бойко, Н. П. Шевченко. – Харків: ХНУ імені В. Н. Каразіна, 2019. – 280 с.
3. Власенко, Т. А. Автоматизація бізнес-процесів на підприємстві: навч. посіб. / Т. А. Власенко. – Львів: ЛНУ імені Івана Франка, 2021. – 200 с.
4. Гнатів, М. Р. Проектування баз даних: теорія та практика: монографія / М. Р. Гнатів. – Київ: КНЕУ, 2018. – 360 с.
5. Григор'єва, Л. І. Сучасні інформаційні технології в управлінні персоналом: навч. посіб. / Л. І. Григор'єва, О. М. Коваль. – Одеса: ОНУ імені І. І. Мечникова, 2020. – 240 с.
6. Дем'яненко, В. М. Програмування на C# з використанням .NET Framework: підручник / В. М. Дем'яненко. – Київ: Видавництво "Техніка", 2022. – 400 с.
7. ДСТУ 8302:2015. Інформація та документація. Бібліографічний запис. Загальні вимоги та правила складання. – Київ: УкрНДНЦ, 2016. – 16 с.
8. Ємельянова, О. П. Управління людськими ресурсами: сучасні підходи: монографія / О. П. Ємельянова. – Дніпро: ДНУ імені Олеся Гончара, 2019. – 290 с.
9. Журавльов, П. В. Розробка інформаційних систем: методологія та інструменти: навч. посіб. / П. В. Журавльов, С. О. Петренко. – Київ: КПІ ім. Ігоря Сікорського, 2020. – 312 с.
10. Ковальчук, І. М. Технології баз даних: SQLite у прикладних системах: монографія / І. М. Ковальчук. – Львів: Видавництво "Магнолія", 2018. – 256 с.

					ДР.122.042.034.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		52

11. Козлов, А. С. Автоматизація управління персоналом: світовий досвід та українські реалії: монографія / А. С. Козлов. – Харків: ХНЕУ ім. С. Кузнеця, 2021. – 340 с.
12. Левицька, О. В. Проектування інтерфейсів користувача: навч. посіб. / О. В. Левицька. – Київ: Видавництво "Ліра-К", 2019. – 220 с.
13. Мельник, В. Г. Entity Framework Core: розробка сучасних додатків: підручник / В. Г. Мельник. – Київ: Видавництво "Наукова думка", 2023. – 380 с.
14. Олійник, Н. П. Управління персоналом у цифрову епоху: монографія / Н. П. Олійник. – Київ: КНЕУ, 2020. – 300 с.
15. Петров, С. І. Windows Forms для розробки настільних додатків: навч. посіб. / С. І. Петров. – Одеса: Видавництво "Фенікс", 2021. – 260 с.
16. Сидоренко, О. М. Безпека інформаційних систем: методи та засоби: монографія / О. М. Сидоренко. – Київ: Видавництво "Політехніка", 2019. – 320 с.
17. Ткачук, Р. В. Проектування програмного забезпечення: сучасні підходи: підручник / Р. В. Ткачук, Л. О. Шевчук. – Львів: ЛНУ імені Івана Франка, 2020. – 350 с.
18. Шаповал, Ю. А. Автоматизація управління персоналом: інструменти та технології: навч. посіб. / Ю. А. Шаповал. – Дніпро: Видавництво "Акцент", 2021. – 280 с.
19. Щербак, Л. М. Інформаційні системи в управлінні підприємством: монографія / Л. М. Щербак. – Харків: ХНУ імені В. Н. Каразіна, 2018. – 310 с.
20. Microsoft Docs. .NET 8 Documentation [Електронний ресурс]. – Режим доступу: <https://docs.microsoft.com/en-us/dotnet/core/whats-new/dotnet-8>. – Дата доступу: 15.04.2025.

ДОДАТКИ

Програмний код модулів

```

using HRManagementSystem.Models;
using Microsoft.EntityFrameworkCore;
using System;
using System.IO;

namespace HRManagementSystem
{
    public class HRDbContext : DbContext
    {
        public DbSet<Employee> Employees { get; set; }
        public DbSet<Department> Departments { get; set; }
        public DbSet<Position> Positions { get; set; }
        public DbSet<Skill> Skills { get; set; }
        public DbSet<EmployeeSkill> EmployeeSkills { get; set; }
        public DbSet<EmployeeDocument> EmployeeDocuments { get; set; }
        public DbSet<User> Users { get; set; }

        public HRDbContext(DbContextOptions<HRDbContext> options) : base(options)
        {
        }

        protected override void OnConfiguring(DbContextOptionsBuilder optionsBuilder)
        {
            if (!optionsBuilder.IsConfigured)
            {
                string databasePath =
                Path.Combine(AppDomain.CurrentDomain.BaseDirectory, "HRManagement.db");
                optionsBuilder.UseSqlite($"Data Source={databasePath};Foreign Keys=True")
                    .EnableSensitiveDataLogging()
                    .EnableDetailedErrors();
            }
        }
    }
}

```