

СТВОРЕННЯ ТА ТЕСТУВАННЯ КРОСПЛАТФОРМНОГО ДОДАТКУ ДЛЯ АВТОМАТИЗАЦІЇ СКЛАДСЬКОГО ОБЛІКУ У СФЕРІ МАЛОГО ПІДПРИЄМНИЦТВА

Ю.Ю. Козіна¹, Д.Р. Горпенко², Д.О. Гріднєв

Національний університет «Одеська політехніка»

1, пр. Шевченка, Одеса, 65044, Україна

Emails: yuliyakc21@gmail.com¹, horpenko@op.edu.ua²

Розглянуто ключові принципи та переваги застосування WMS-систем в управлінні складськими процесами. Проаналізовано сучасні підходи до організації інвентаризації в роздрібній торгівлі та запропоновано власний кросплатформний застосунок, реалізований з використанням технологій C# та MAUI. Розроблений інструмент підтримує роботу як на мобільних пристроях, так і на стаціонарних комп'ютерах, забезпечує функції додавання, зберігання та пошуку товарів, а також підвищує ефективність управління запасами. У рамках розробки реалізовано функціонал моніторингу товарних залишків у реальному часі, визначення їх розміщення на складі та автоматизації процесів інвентаризації. Представлені результати можуть бути впроваджені в практичну діяльність торгових підприємств для підвищення продуктивності персоналу та зменшення впливу людського фактору.

Ключові слова: WMS-система, моніторинг, інвентаризація, кросплатформність.

Вступ. Раціональне управління складськими запасами відіграє важливу роль у забезпеченні успішної діяльності підприємств, особливо в роздрібній торгівлі, де безперерйна наявність продукції на полицях є критичною. Використання застарілих методів обліку, таких як паперові носії чи електронні таблиці, часто спричиняє помилки та ускладнює процес контролю товарних залишків. Це свідчить про актуальність впровадження сучасних цифрових інструментів, здатних оптимізувати інвентаризацію та підвищити ефективність складських операцій.

Мета роботи. Вимоги практики до розробки кросплатформних додатків, що функціонують як на десктопних, так і на мобільних платформах для підтримки працівників під час інвентаризації товарів у магазині, є досить високими. Метою даної роботи є створення кросплатформного додатку з низьким рівнем ресурсоемності на основі принципів WMS-систем (Warehouse Management System) для ефективного управління запасами та контролю товарних залишків.

Основна частина. Проведений аналіз WMS-систем [1], аналогічних програмних рішень та кросплатформних фреймворків дозволив виявити ключові переваги й недоліки кожного підходу, що варто враховувати при розробці або виборі системи управління складом. Серед WMS-систем спостерігається чіткий розподіл за рівнем адаптивності та складності впровадження. Коробкові рішення є простими у встановленні, проте обмежені в налаштуванні. Адаптовані системи є найкращим вибором для середнього бізнесу, забезпечуючи гнучкість та розширюваність. Замовні системи, попри високу вартість і тривалий цикл впровадження, забезпечують найвищу ефективність для великих підприємств із складними логістичними потребами. Водночас, хмарні рішення забезпечують швидкий старт і зменшення експлуатаційних витрат, однак залежать від сторонніх провайдерів. У рамках порівняльного аналізу існуючих рішень, зокрема Odoo та DNTrade [2], виявлено, що обидва продукти частково задовольняють потреби управління складом, однак мають суттєві обмеження. Odoo пропонує тісну інтеграцію з іншими бізнес-процесами, але не забезпечує детального управління простором складу, а також вимагає додаткових налаштувань і витрат. DNTrade є доступнішим, проте менш функціональним – відсутні звіти та

повноцінне управління персоналом, що обмежує можливості масштабування. Щодо кросплатформних фреймворків, MAUI відзначається високою продуктивністю і стабільністю, проте має обмежену екосистему та велику вагу додатків [3]. React Native забезпечує швидку розробку й активну підтримку спільноти, однак не підходить для складної графіки й потребує додаткових рішень для оптимізації інтерфейсу під iOS. Flutter, хоча й потребує вивчення мови Dart, вирізняється потужними інструментами для створення функціональних інтерфейсів та високою продуктивністю, але має певні технічні обмеження через ще не повністю стабілізовану екосистему.

Таким чином, ефективне впровадження WMS-системи потребує збалансованого вибору між функціональністю, масштабованістю та вартістю. А вибір фреймворку для розробки безпосередньо залежить від складності продукту, строків реалізації та наявних технічних ресурсів.

Для оптимізації інвентаризаційних процесів було вирішено створити кросплатформний застосунок, що поєднає можливості WMS-систем і враховує потреби малого бізнесу. Він дозволить швидко знаходити товари за назвою чи штрих-кодом, переглядати залишки та місцезнаходження продукції. Ключовою особливістю є система акаунтів із розподілом ролей і захищеним зберіганням паролів. Застосунок буде підтримувати базові операції з товарами – додавання, редагування, видалення, а також формування звітів. Адміністратор бути мати повний доступ до керування системою. Завдяки локальній базі даних додаток буде працювати навіть без інтернету, що особливо важливо за умов нестабільного зв'язку.

Для зберігання даних у системі інвентаризації пропонується реляційна база даних, яка містить основні сутності, необхідні для функціонування додатку. Структура бази побудована таким чином, щоб забезпечити ефективне управління користувачами та товарами, а також спростити процеси пошуку, редагування та оновлення інформації.

У базі даних визначено дві основні сутності: User (Користувач) та Product (Товар). Розглянемо схему бази даних та її елементи рис. 1.

Сутність User (Користувач) відповідає за управління обліковими записами співробітників і контроль доступу до функцій додатку. Містить унікальний ID (int, первинний ключ з автоінкрементом), ім'я (Name), унікальний логін (Login), зашифрований пароль (Password) та роль (Role), наприклад, «Адміністратор» або «Працівник». Розподіл ролей забезпечує різні рівні доступу для підвищення безпеки.

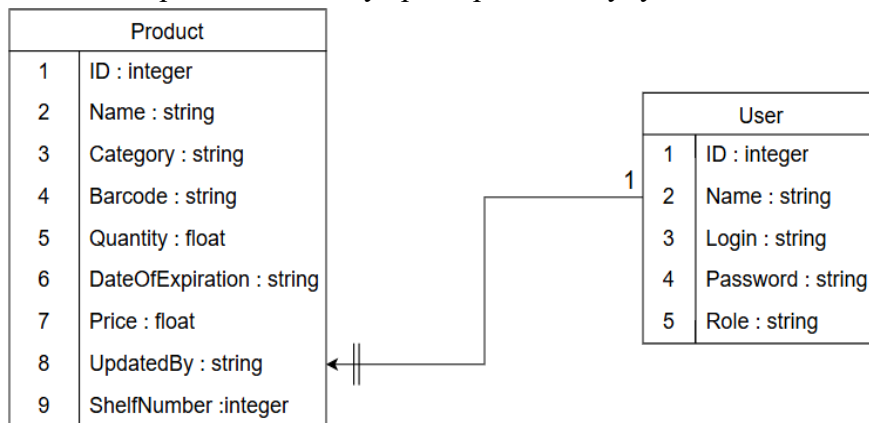


Рис. 1. Схема зв'язку бази даних та її елементи

Сутність Product (Товар) зберігає інформацію про товари на складі та в магазині: унікальний ID, назву (Name), категорію (Category), штрих-код (Barcode), кількість (Quantity), термін придатності (DateOfExpiration), ціну (Price), логін останнього, хто оновлював товар (UpdatedBy) та номер полиці (ShelfNumber). Така структура дозволяє швидко шукати товари, контролювати залишки та оновлювати дані. База даних працює локально, що забезпечує доступ до інформації навіть без інтернету.

Запропоновано зберігати у зашифрованому вигляді за допомогою алгоритму BCrypt, що підвищує безпеку системи. Унікальність логінів виключає дублювання облікових записів. Наявність даних про термін придатності товарів дозволяє автоматизувати контроль прострочених позицій. Алгоритм BCrypt використовує сольове хешування та регульований фактор складності, що ускладнює атаки методом перебору (brute-force) і захищає від відомих вразливостей стандартних хеш-функцій (SHA-256, MD5). Завдяки цьому паролі надійно захищені навіть при зломі бази даних. База даних запроєктована для швидкої обробки запитів, ефективного збереження інформації і можливості подальшого розширення функціоналу системи. Загальна блок-схема функції хешування показана на рис. 2. Під час генерації ключа BCrypt застосовує функцію розширення ключа (key expansion), що приймає сіль і пароль як вхідні дані та виконує велику кількість ітерацій для уповільнення процесу ця система ітерацій є основною захисною рисою алгоритму. Далі відбувається інтенсивне ключове розтягнення (key stretching): алгоритм виконує 2^{cost} раундів (де cost це налаштовуваний параметр складності), і чим вищий цей параметр, тим довше формується хеш, що значно ускладнює brute-force атаки. Готовий хеш кодується у спеціальному форматі, яке містить ідентифікатор алгоритму (наприклад, \$2a\$), сіль, рівень складності (cost) і власне *фінальний* хеш пароля [4].



Рис.2. Алгоритм хешування

Ключові функції системи охоплюють основні процеси, зокрема управління товарами, роботу з обліковими записами, перевірку паролів і створення звітів. Розробка функцій орієнтована на зручність використання, безпеку та продуктивність.

Основні з них:

- реєстрація, видалення та вхід користувачів із перевіркою пароля за допомогою BCrypt;
- додавання, редагування й пошук товарів у базі даних;
- моніторинг термінів придатності та залишків під час формування звітів;
- контроль доступу до функцій залежно від ролі користувача.

Функції системи поділено за призначенням. Зокрема, функції, що стосуються користувачів, згруповані в категорії "Акаунт" і реалізовані з використанням моделі User та запитів до бази даних. Функція входу (аутентифікації) забезпечує доступ до системи лише авторизованим користувачам відповідно до їхніх прав доступу.

У роботі було використано запропонований в [5] алгоритм, що базується на парадоксу днів народжень. При розробці системи, нам треба генерувати унікальні ідентифікатори, які використовуються для аутентифікації користувачів.

Так як ідентифікатор, по суті, дає повний доступ до аккаунту користувача, до його генерації слід підійти серйозно, головна вимога до гарантії його унікальності, так як при співпаданні ідентифікаторів у різних користувачів, один з користувачів отримає доступ до чужого аккаунту. Виходячи з цього ймовірність дворазового генерування одного і того ж ідентифікатора – виникнення колізії – повинна бути вкрай мала.

Так само ідентифікатор по можливості не повинен бути сильно довгим в цілях економії трафіку. На ймовірність виникнення колізії впливають два фактори:

- 1) розмір ідентифікаційного простору – кількість можливих унікальних ідентифікаторів;
- 2) метод генерування ідентифікаторів – яким чином ідентифікатор вибирається із загального простору. В ідеалі нам потрібно великий простір (щодо наших потреб), з якого випадково обираються рівномірно розподілені ідентифікатори. Тому для генерації випадкових рівномірно розподілених ідентифікаторів ми будемо конвертувати результат функції `crypto.randomBytes(N)`, де N кількість повернених випадкових байт, у шістнадцятиричну строку. Ця функція, із пакету інструментів OpenSSL, є реалізацією криптографічно стійкого псевдовипадкового алгоритму який базується на "Вихрі Мерсена" [6]. Залишилося підрахувати скільки випадкових байт нас, влаштує, щоб ймовірність колізії була вкрай мала. Представимо нашу задачу в наступному абстрактному вигляді: Дано n випадкових чисел з дискретного рівномірного розподілу з діапазоном $[1, H]$. Яка ймовірність $p(n, H)$, що, принаймні два числа збігаються?

Дана задача є узагальненням парадоксу днів народжень. Знайдемо зворотну ймовірність $\bar{p}(n, H)$ при $n < H$, таку що всі числа будуть різними:

$$\bar{p}(n, H) = 1 \times \left(1 - \frac{1}{H}\right) \times \left(1 - \frac{2}{H}\right) \times \dots \times \left(1 - \frac{n-1}{H}\right). \quad (1)$$

H – буде позначати розмір ідентифікаційного простору, P – ймовірність колізії, n – кількість згенерованих ідентифікаторів. Розмір ідентифікаційного простору обчислюється таким чином:

$$H = 2^{8N}, \quad (2)$$

де N – довжина ідентифікатора в байтах.

Виходячи з досліджень, зроблених на базі цього підходу, можна прийняти довжину ідентифікатора рівну 8 байтам, що гарантує, що ми досягнемо ймовірності колізії в 0,1%.

Проведено тестування розробленої системи. На головній сторінці розташовані елементи для пошуку товарів зі списку, список товарів, та навігаційна панель на якій розташовані кнопки для переходу до інших сторінок. На рис. 3 зображено вигляд інтерфейсу сторінки складу. У додатку можна здійснити пошук за назвою товару або ж його штрих кодом. Для цього потрібно ввести в поле пошуку назву або номер штрих коду та натиснути кнопку "Пошук". Реалізована можливість додати товар.

Введіть назву або штрихкод:		Пошук	Всі продукти
Назва	Категорія	Штрих-код	Кількість
Томат черрі "Амор"	Овочі	4567999774123	70,5
Чипси Nice зі смаком паприки	Снеки	1462481884621	100
Хліб житній "Житній" міні	Випічка	12531515532	40
Сметана Minister 10%	Молочні продукти	12165126562	100
Корм для котяг ShebaBlack&Gold	Товари для тварин	125175255514	150
Лимонад "Fresh" апельсин-персик	Напої	125361747	200
Bicki Jacson	Алкоголь	12456371788	50
Цукерки "Tutti"	Кондитерські товари	1241471641	70,5
Креветка "Ванамей"	Морепродукти	1266723761	5,5
Картопля фри "Western"	Закуски	12521521562	100
Пиво BeerMan	Алкоголь	1231774718841	124
Почив "Хрушка"	Кондитерські вироби	58667638867	44,7

Рис.3. Вигляд інтерфейсу сторінки складу

В полях для інформації можна вписати деталі про товар такі як: назва, категорія, штрих-код, кількість, ціну, дату закінчення терміну придатності та номер полиці (Рис. 4)

Додати продукт

Заповніть всі поля знизу:

Назва продукту

Сухий сніданок "Finish" з полуницею

Категорія

Бакалія

Штрих-код

5323313121231

Кількість

125

Ціна за шт.

63,90

Дата закінчення терміну придатності (dd-MM-yyyy)

21-05-2025 21-05-2025

Номер полиці

5

Додати продукт

Рис. 4. Вигляд інтерфейсу сторінки додавання товару

Можна сформувати звіт по терміну придатності. Для цього потрібно обрати діапазон дат в яких закінчиться термін придатності товару. На рис. 5 зображено приклад створеного звіту по терміну придатності.

Оберіть тип звіту:

Звіт по терміну придатності

Дата початку:

11-02-2025

Дата кінця:

19-02-2025

Сформувати звіт Копіювати

Звіт по терміну придатності(З 11-02-2025 по 19-02-2025) за 14.04.2025 21:26:57
 Кількість продуктів які було виділено: (233,0), їх ціна становить: 19951,00 грн.
 1. Томат черрі "Амор"; Штрих-код: 4567999774123; Термін сплине: 19-02-2025
 2. Хліб житній "Житній" міні; Штрих-код: 12531515532; Термін сплине: 14-02-2025
 3. Сметана Minister 10%; Штрих-код: 12165126562; Термін сплине: 18-02-2025
 4. Креветка "Ванамей"; Штрих-код: 1266723761; Термін сплине: 15-02-2025
 5. Чипси "З крабом"; Штрих-код: 24214142; Термін сплине: 18-02-2025

Рис. 5. Приклад створеного звіту по терміну придатності

Після завершення функціонального тестування розроблений застосунок, який ґрунтується на принципах WMS-систем, було порівняно з DNTrade за такими ключовими параметрами: адаптивність інтерфейсу, точність введення та обробки даних, розширені функції інвентаризації, рівень безпеки доступу, стійкість до нестабільного інтернет-з'єднання та модель розповсюдження з економічної точки зору.

Розроблений додаток вирізняється справжньою кросплатформенністю – інтерфейс автоматично оптимізується під будь-який тип пристрою, зберігаючи зручність користування та чіткість відображення. У випадку з DNTrade, який переважно орієнтований на використання з десктопів, перехід на мобільні пристрої призводить до зменшення елементів інтерфейсу або їх згортання в списки, що ускладнює доступ до потрібної інформації та уповільнює процес вибору товарів.

Висновки. Запропонований кросплатформений додаток, розроблений за принципами WMS-систем не лише покращує деякі функції у DNTrade такі як адаптивність, точність обліку, безпека паролів та стабільність роботи без інтернету, а й надає бізнесу додаткову економію завдяки безкоштовному розповсюдженню. Результати його функціонування мають прикладне значення та відкривають перспективи подальшого вдосконалення через інтеграцію з іншими системами й використання розширених аналітичних інструментів.

Список літератури

1. Implementing Warehouse Management Systems (WMS) in Logistics URL: <https://www.researchgate.net/publication/330293478>
2. Oddo ERP and CR. URL: <https://www.odoo.com>
3. Roger Y. .NET MAUI Cross-Platform Application Development: навчальний посібник. Birmingham: Packt Publishing Ltd., 2023. 400 p.
4. Fast Software Encryption, Description of a New Variable-Length Key, 64-Bit Block Cipher URL: https://www.schneier.com/academic/archives/1994/09/description_of_a_new.html
5. Birthday problem URL: https://en.wikipedia.org/wiki/Birthday_problem
6. Mersenne Twister Home Page URL: <http://www.math.sci.hiroshima-u.ac.jp/~m-mat/MT/emt.html>

Ю.Ю. Козіна, Д.Р. Горпенко, Д.О. Гріднєв

**DEVELOPMENT AND TESTING OF A CROSS-PLATFORM APPLICATION
FOR WAREHOUSE MANAGEMENT AUTOMATION IN THE FIELD OF
SMALL BUSINESS**

Yu.Yu. Kozina¹, D.R. Gorpenko², D.O. Gridnev

National Odesa Polytechnic University
1, Shevchenko Ave., Odesa, 65044, Ukraine
Emails: yuliyakc21@gmail.com¹, horpenko@op.edu.ua²

The key principles and advantages of using WMS systems in warehouse process management have been examined. Modern approaches to inventory organization in retail have been analyzed, and a custom cross-platform application developed using C# and MAUI technologies has been proposed. The developed tool supports operation on both mobile devices and desktop computers, providing functionality for adding, storing, and searching for products, while also improving inventory management efficiency. The application includes features for real-time stock level monitoring, determining product placement within the warehouse, and automating inventory processes. The presented results can be applied in the practical operations of retail businesses to increase staff productivity and reduce the impact of human error.

Keywords: WMS system, monitoring, inventory, cross-platform.