

ОСОБЛИВОСТІ РОЗРОБЛЕННЯ ВЕБДОДАТКУ ДЛЯ СИСТЕМИ УПРАВЛІННЯ СКЛАДСЬКИМ ПРИМІЩЕННЯМ

М. Я. Ковалишин, Я. С. Парамуд

Національний університет “Львівська політехніка”,

кафедра електронних обчислювальних машин

E-mail: marian.kovalyshyn.mkisp.2023@lpnu.ua, yaroslav.s.paramud@lpnu.ua

© Ковалишин М. Я., Парамуд Я. С., 2024

У статті розглянуто особливості загального процесу проєктування системи управління складським приміщенням, орієнтованої на максимальну ефективність його роботи. Основна увага приділена методам моделювання основних етапів функціонування системи для ефективної роботи основних операцій складу, а також детально оглянуто структуру бажаної бази даних, що слугувала б сховищем для зберігання різноманітних даних про товари, їх рух та операції над ними. Здійснено аналітичний огляд різних аналогічних рішень в царині управління складськими приміщеннями, описано їхні переваги та недоліки, які використані для формування функціоналу системи.

Надано опис функціональних можливостей програмного забезпечення, що б забезпечило автоматизацію роботи складу, включаючи обробку отриманих замовлень, керування запасами та моніторинг змін між операціями. Це було досягнуто завдяки текстовому опису та реалізації діаграм різних нотацій, що найкраще доповнюють одна одну. Основні результати дослідження показують важливість впровадження якісного та сучасного вебдодатку для управління складськими приміщеннями для підвищення продуктивності, уникнення помилок та збільшення точності під час виконання операцій. Це допомагає скоротити час обробки замовлень на 30 % завдяки автоматизації процесів і оптимізації інтерфейсу. Додатково система знижує операційні витрати на 15 % завдяки автоматичному обліку та зменшенню помилок, а також покращує синхронізацію даних на 20 %, забезпечуючи їхнє оновлення в реальному часі.

Ключові слова: вебдодаток, база даних, управління запасами, управління складом.

Вступ

Розглядаючи особливості керування складськими приміщеннями, можна визначити, що важливим елементом успішного функціонування підприємств цього типу є операції з товарними запасами. Серед базових операцій можна визначити такі: приймання, опрацювання, зберігання різноманітних товарів, встановлення маршрутів їхнього переміщення. До основних завдань таких систем належить відстеження залишків товарів, облік переміщення обраних товарів між складськими зонами, створення звітів за різними критеріями та проведення аналітики для прийняття рішень щодо ефективного використання складської території. Відповідно важливо забезпечити безперебійність цих операцій та достовірність результатів їх виконання. На великих підприємствах, коли кількість та обсяг товарів швидко змінюється, створюються умови для ускладнення логістики та виникнення помилкових операцій. Через це автоматизація виконання базових операцій та використання інформаційних систем стає невіддільною (частина/ознака) складовою сучасних складських приміщень.

Вебдодатки, що розширюють функціональні можливості управління складом, суттєво зменшують кількість помилок, які виникають під час виконання складських робіт вручну, покращують якість виконання базових операцій, підвищують швидкість та ефективність опрацювання замовлень, забезпечують зручний доступ до даних. Відповідно дослідження в царині розроблення спеціалізованих вебдодатків є актуальним завданням.

Огляд літературних джерел

Ефективне управління складом є важливим для хорошого функціонування логістичної системи. На сьогодні робота на складі стає все складнішою, оскільки підприємства прагнуть заощадити витрати, пришвидшити обробку замовлень та мінімізувати наявність помилок. Однією з найбільших проблем є неналежна автоматизація процесів, яка спричиняє помилки при обліку товарів, втраті часу та підвищенні небажаних витрат на підтримку операцій.

Посилаючись на дослідження [1], підприємства стикаються з проблемою чіткої видимості процесів складських операцій. Виконання операцій вручну та недостатня імплементація систем між складами призводять до втрати або неправильного розподілу ресурсів. Це підтверджує невідворотність впровадження інформаційних систем для управління складом, що приведе керування складом на високий рівень ефективності.

Наступне дослідження [2] показує, що проблемою, з якої можна часто стикнутись, є погана координація роботи між відділами, що відповідають за керування ресурсами. Автоматизована система дозволить вести відстеження переміщень товарів у реальному часі, що спростить пошук бажаних товарів та зменшить час пошуку, який інакше б проводився через документацію або самостійний пошук на складі.

Наступною проблемою, з якою можна стикнутись під час керування, є розбіжність між можливостями поточних систем управління складом та збільшенням обсягів даних та операцій у них. Дослідження [3] показує, що саме сучасні технології обробки великих масивів даних допомагають складам швидко адаптуватися до зміни функціональних задач, полегшуючи роботу системи на всіх рівнях роботи та для всіх користувачів.

Однак, незважаючи на значні досягнення в автоматизації, деякі проблеми залишаються нерозв'язаними. Наприклад, дослідження [4] надає висновок, що більшість складських систем орієнтовані лише на відстежування товарів і не використовують дані для прийняття рішень щодо керування складом. Це відбувається через те, що розробка більш інтелектуальної системи, яка могла б не лише автоматизувати облік бажаних товарів, але й проводити аналіз даних для прогнозування бажаної кількості товарів, є більш складною.

При розробці вебдодатку для системи управління складським приміщенням варто насамперед створити загальну схему системи з демонстрацією бізнес-процесів. Оглянувши дослідження [5], можна виділити основні методи створення діаграми IDEF0, яка дозволить покращити аналіз і структурувати систему управління складом шляхом коригування діаграм відповідно до вимог балансування та згуртованості. Проте слід зауважити, що цей процес складний і вимагає розуміння методології IDEF0, яка може бути неінтуїтивно зрозумілою і призвести до надмірного ускладнення системи, якщо застосувати всі принципи, описані у цій статті.

Крім IDEF0 діаграм, важливим кроком є створення діаграми UML. У статті [6] проведено огляд, який мав на меті зрозуміти найпоширеніші діаграми UML, які використовують, їх призначення та області застосування. Недоліком цього дослідження є те, що була обрана лише один сервіс Google Scholar, який потенційно обмежує сферу застосування. Більш наочний приклад застосування таких діаграм можна побачити у цій статті [7]. Проте варто зауважити, що описаний в ній процес створення діаграм є неповним і може бути не до кінця зрозумілим людям, які мають на меті створити складну систему, не маючи великого досвіду застосування UML діаграм.

Незважаючи на те, якими корисними діаграми діяльності є для бізнес-моделювання [8], коли їх використовують для деталізації процесів, пов'язаних з бізнес-діяльністю, найважливішим в

управлінні складським приміщенням є зберігання товару. Для розробки вебдодатку, в якого основна мета – це якраз обробка та збереження великої кількості даних, найкращим рішенням може бути MySQL [9]. MySQL є системою керування реляційними базами даних (RDBMS), яка розроблена всесвітньо відомою компанією Oracle. Для проведення операцій використовують мову структурованих запитів (SQL). При цьому вона не є універсальною, тому не може бути впроваджена у всіх сферах застосування. Отже, перед розробкою додатку варто ретельно дослідити інші, аналогічні рішення.

Огляд поточних досліджень показує, що важливість автоматизації процесів та інтеграція систем для покращення ефективності керування запасами є необхідністю для кожного підприємства. Та навіть існують деякі сфери, як аналітична підтримка та інтеграція з іншими системами, які потребують додаткових досліджень та удосконалень.

Постановка завдання

Система управління складським приміщенням є системою, яка повинна допомагати забезпечувати ефективну роботу із запасами товарів, включаючи їх облік, зберігання, переміщення та доставку. Традиційне керування складів часто стикається з численними проблемами, такими як облікові помилки, нечіткі процеси та нестача аналітичної підтримки під час прийняття рішень. Ці проблеми мають бути вирішені завдяки створенню якісної системи, яка зробить автоматизацію цих операцій та значно підвищить загальну ефективність складу.

Метою цього дослідження є розроблення особливостей створення та впровадження вебдодатку інформаційної системи для управління складським приміщенням, орієнтованому на зменшення часу опрацювання замовлень, зменшення кількості помилок та покращення процесу керування запасами.

Основними завданнями, які необхідно вирішити для виконання поставлених цілей, є:

1. Аналіз існуючих систем управління складом та перегляд їхніх недоліків.
2. Розробка архітектури ІС для автоматизації основних процесів на складі.
3. Підбір СУБД і впровадження бази даних для швидкого доступу до інформації про деталі кожного товару та їх розміщення або транспортування.
4. Покращення функціоналу для автоматизації складських завдань, як отримання товарів, їх відстеження та переміщення.
5. Створення зручного інтерфейсу, який зміг би запропонувати зручний доступ до основних функцій системи.
6. Тестування розробленої системи для можливості оцінки її продуктивності.

Формування вимог до системи з управління складу

Функціональні вимоги:

1. Система має забезпечувати поточне відстеження товарів, їх знаходження, історію переміщень та їхню кількість в певний момент, а також додатково оновлювати ці дані за допомогою запитів при додаванні або видаленні товару.
2. Необхідно надати можливість обліку з надходжень та відправлень запасів товарів із використанням кодів для мінімізації помилок.
3. Функція прогнозування попиту має попереджувати про зменшення кількості певних продуктів на складі. Що порівняно з ABC Inventory надає перевагу, бо в цій системі така функція відсутня.
4. Вебсайт повинен надати повний доступ до обробки замовлень, сортування товарів за критеріями, розподіл запасів на складі та відправку до відвантаження.
5. Система повинна підтримувати інтеграцію з іншими системами для майбутньої інтеграції даних і процесів. Наприклад, LMN Warehouse Manager, часто стикається з проблемою поганої синхронізації даних між різними системами. Пропонована система при інтеграції дозволить скоротити час обробки даних на 20%, автоматично синхронізуючи їх у реальному часі з іншими платформами.

Нефункціональні вимоги:

1. Необхідно забезпечити безпеку даних через використання шифрування для захисту конфіденційних даних та створення додаткового контролю доступу на основі ролей для різних користувачів.
2. Система мусить забезпечувати безперервну роботу не дивлячись на збої, зі збереженням резервних копій для резервного відновлення.
3. Вебсайт повинен адаптуватися при змінах до розміру складу та обсягах проведених операцій між товарами.
4. Система повинна надавати можливість швидкої обробки при занесенні великої кількості запитів, для реалізації ефективності роботи. Це буде реалізовано завдяки використанню простих функцій, що пришвидшить виконання завдань учасників. На відміну від Oracle WMS, інтерфейс якого часто критикують за складність, проєктована система пропонує інтуїтивно зрозумілий дизайн, що дозволяє скоротити час навчання персоналу на 40 %.
5. Інтерфейс повинен бути інтуїтивно зрозумілим для всіх користувачів.

Моделювання функціональних процесів за допомогою діаграм IDEF0

Реалізувавши опис обраних вимог проєкту, можна перейти до важливого етапу розбору бізнес-процесів обраної предметної області. Найкращим способом для візуалізації наступних дій буде створення IDEF0 діаграм.

Діаграма IDEF0 (є абревіатурою від Integrated Computer Aided Manufacturing) представляє інтегроване зображення входів, керування, виходів і механізмів (ICOM) для декомпозиції певних функцій. Діаграма IDEF0, яка є частиною поведінкового набору представлень, показує багато контекстної інформації про взаємозв'язки декомпозиції, при цьому не відображаючи фактичну керуючу логіку або структуру декомпозиції. Вона представляє різноманітні процеси у вигляді функцій (блоків), які взаємодіють із різними даними.

Контекстна діаграма (A-0) є найвищим рівнем діаграми, її основною ціллю є опис загальної функції системи. Вона показує систему як єдину функцію, визначаючи додатково її стрілки входів, виходів, механізмів та управління.

Контекстна та наступні декомпозиції діаграм нотації IDEF0 для будь-якої системи повинні включати такі типи стрілок:

Вхід (Input) – уособлюють собою певні дані чи матеріали, які входять до процесу на першочерговому етапі та змінюються в ході виконання системи.

Управління (Control) – правила, обмеження та плани, які визначають та скеровують, як саме повинна виконувати певна функція.

Механізми (Mechanisms) – ресурси, інструменти або люди, які входять до виконання функції.

Вихід (Output) – результат виконання функції.

Для вебдодатку, що розробляється, вигляд контекстної діаграми буде таким (рис. 1).

Під час розробки контекстної діаграми її вигляд, обмежений однією функцією, може бути недостатнім для повноцінного огляду системи. Тому існує декомпозиція першого рівня, яка розбиває функції на певну кількість підфункцій. Кожна з цих функцій показує різноманітні процеси, що входять до розглянутого раніше блоку. Цей тип діаграми також повинен містити чотири типи стрілок.

Для вебдодатку, що розробляється, вигляд першого рівня декомпозиції буде таким (рис. 2).

Навіть виконавши діаграму першого рівня її деталізованість може бути недостатньою для повноцінного опису діаграми, тоді можна застосувати такий же метод та провести декомпозиції на другому рівні, де одна з функцій розглядається ще детальніше. Декомпозиція цього рівня (A1, A2 тощо) розкриває структуру кожної з розглянутих підфункцій.

Для вебдодатку, що розробляється, вигляд другого рівня декомпозиції буде таким (рис. 3).

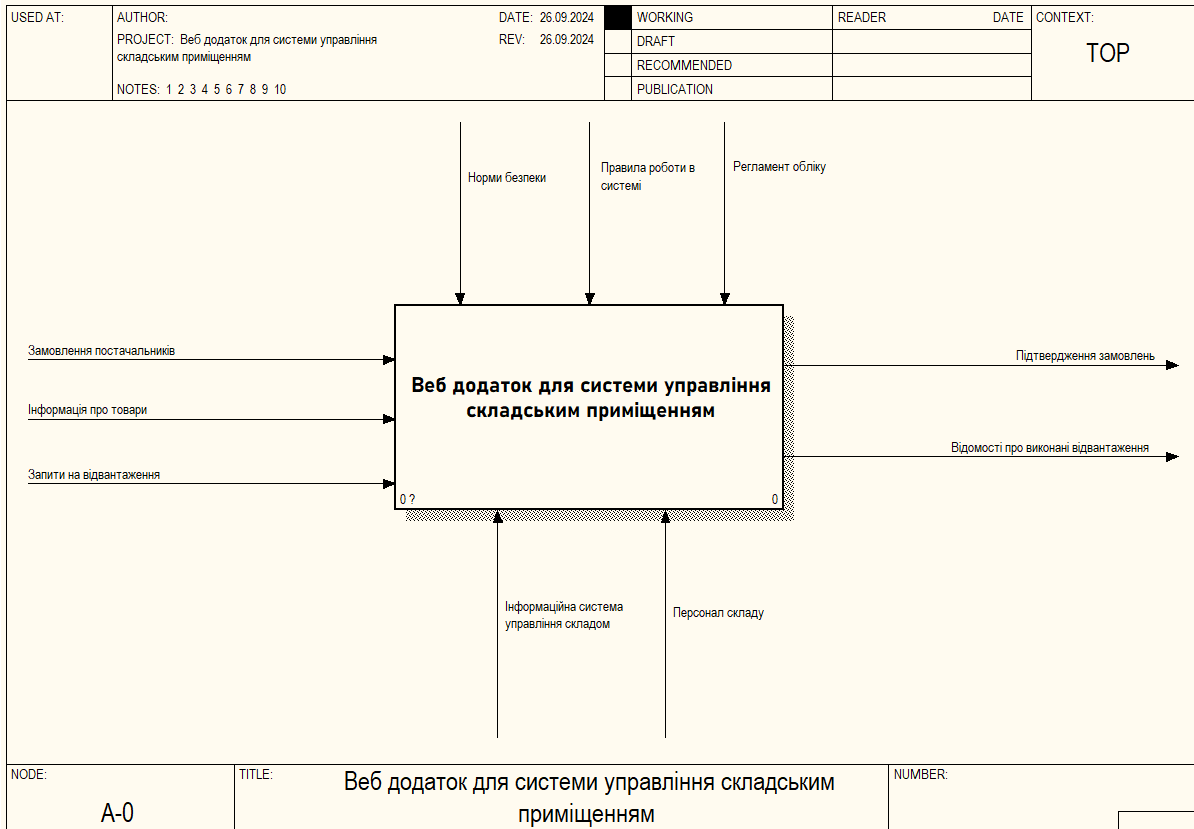


Рис. 1. Контекстна діаграма системи управління складським приміщенням

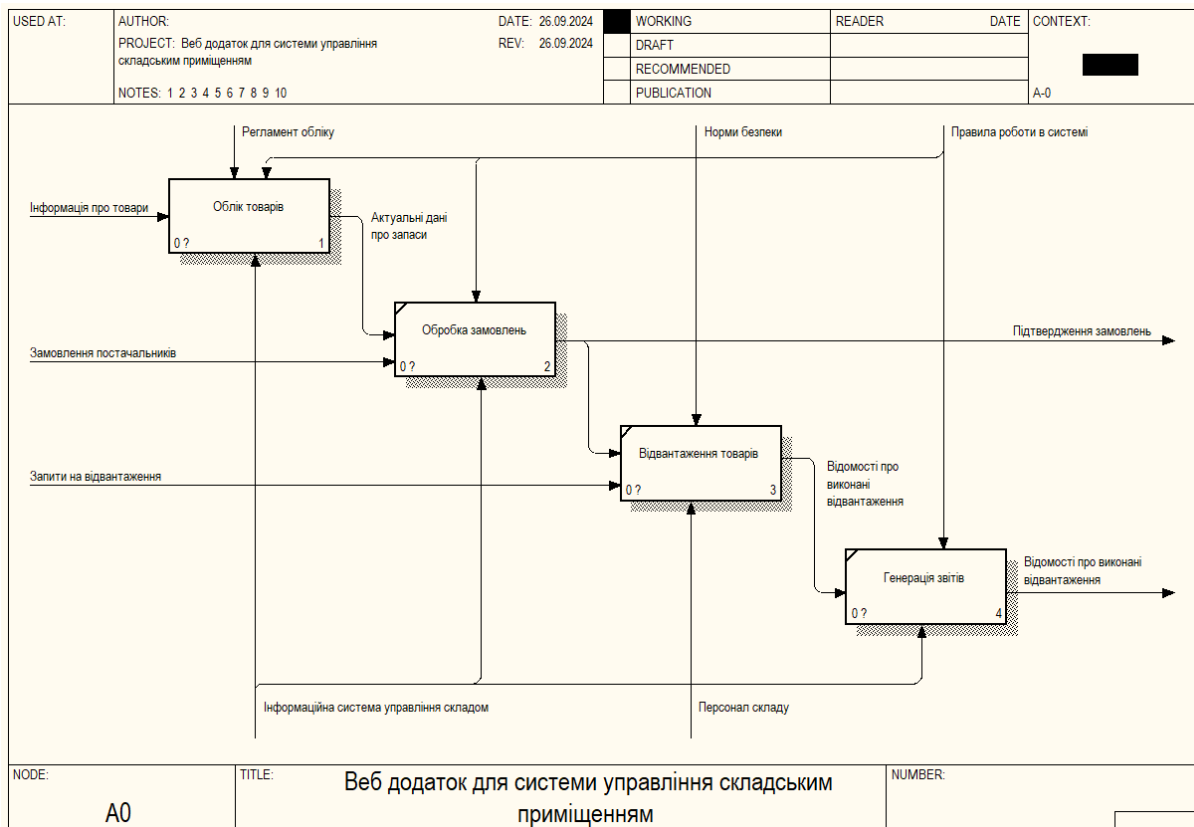


Рис. 2. Діаграма першого рівня декомпозиції системи управління складським приміщенням

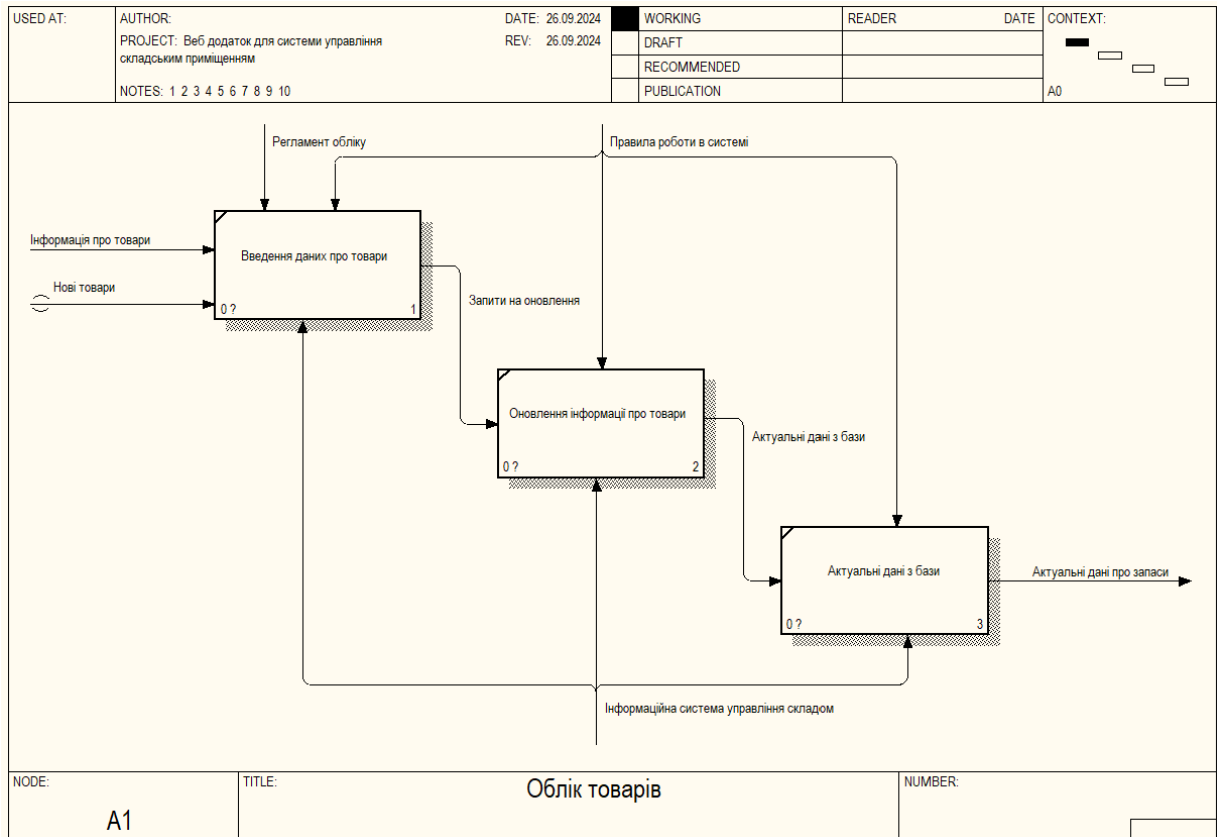


Рис. 3. Діаграма другого рівня декомпозиції системи управління складським приміщенням

У цьому підрозділі представлено функціональне моделювання процесів управління складом за допомогою діаграм IDEF0. Описано основні функціональні блоки, взаємозв'язки між ними та інформаційні потоки.

Проектування архітектури системи в нотації UML

Діаграма UML – це спосіб візуалізації систем і програмного забезпечення за допомогою уніфікованої мови моделювання (UML). Інженери програмного забезпечення створюють діаграми UML, щоб зрозуміти конструкції, архітектуру коду та запропоновану реалізацію складних програмних систем. Діаграми UML також використовують для моделювання робочих процесів і бізнес-процесів.

Написання коду може бути складним процесом із багатьма взаємопов'язаними елементами. Часто існують тисячі рядків мови програмування, які важко зрозуміти з першого погляду. Діаграма UML спрощує цю інформацію у візуальне посилання, яке легше сприймати.

Цей вид діаграм використовують для відображення динамічної поведінки системи. Вона демонструє функціональні можливості системи, об'єднуючи варіанти використання (use case) та акторів (actors). Вона моделює служби, функції та завдання, необхідні системі чи підсистемі програми. Така діаграма описує функціональність системи на високому рівні, а також те, як користувач взаємодіє із системою. Основна мета таких діаграм – це відобразити динамічний аспект системи. Вона деталізує внутрішні та зовнішні дані, які входять до вимог системи. Діаграма представляє осіб, варіанти використання та інші речі, які використовують акторів і елементи, відповідальні за реалізацію діаграм варіантів використання.

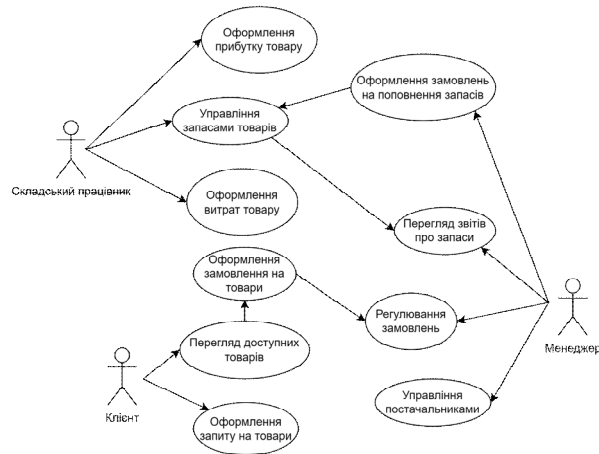


Рис. 4. UML-діаграма варіантів використання системи управління складським приміщенням

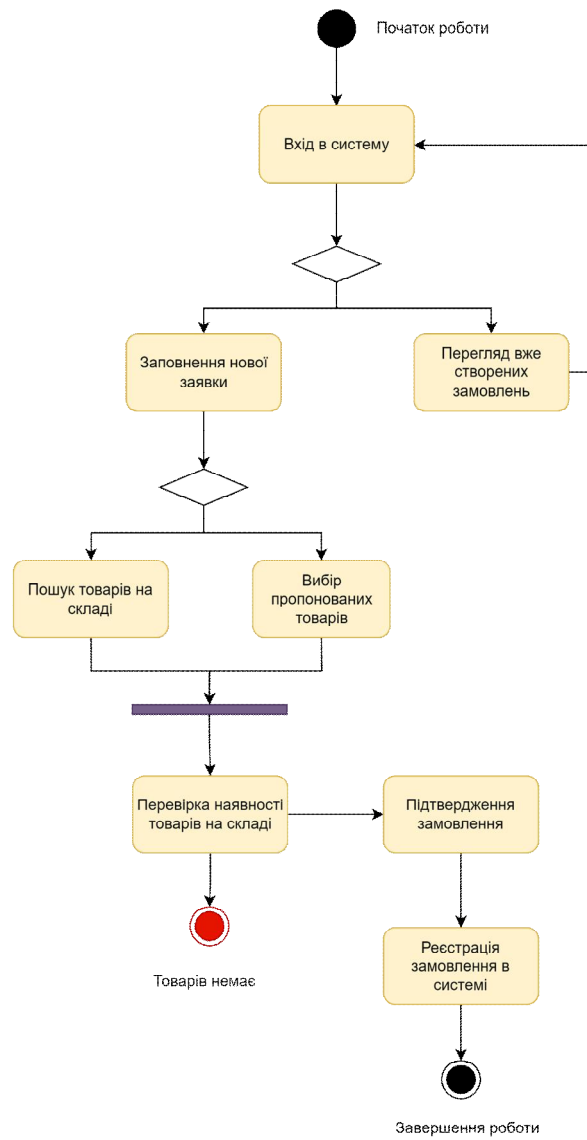


Рис. 5. Діаграма діяльності системи управління складським приміщенням

В UML для відображення послідовності дій використовують діаграму діяльності. Діаграми діяльності відображають робочий процес від початку до кінця, детально описуючи різні шляхи прийняття рішень, які виникають у ході розвитку подій, що входять до складу діяльності. Вона може використовуватися для деталізації ситуацій, коли під час виконання деяких дій може відбуватися паралельна обробка (рис. 5).

Діаграма розгортання – це тип діаграми уніфікованої мови моделювання, який використовують для представлення фізичного розгортання компонентів системи. Вона показує, як деталі програмного забезпечення розподіляються між апаратними компонентами, та ілюструє фізичні конфігурації програмного та апаратного забезпечення, які є критичними для виконання, роботи та масштабування системи. Ці діаграми використовують інженери програмного забезпечення та системні інженери для детального опису та розуміння складної інфраструктури та мережних налаштувань, де система має бути розгорнута (рис. 6).

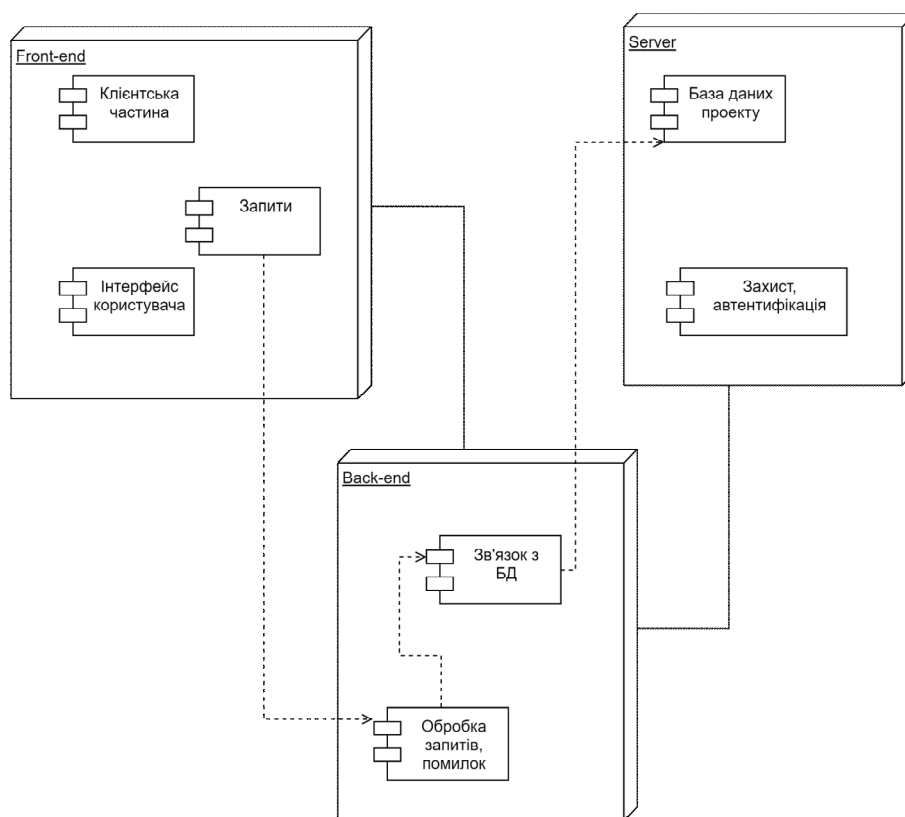


Рис. 6. Діаграма розгортання системи управління складським приміщенням

Структурне проектування бази даних інформаційної системи

Розглядаючи різні етапи планування та розробки, можна зробити висновок, що створення бази даних є найважливішим етапом. Це виходить з того, що БД зберігає всю інформацію системи, яка є головною частиною проекту, бо якщо база буде некоректно спроектована, збережена або заповнена, то робота всієї системи буде під загрозою.

Обравши MySQL для розробки вебдодатку, варто виділити, що вона є однією з найбільш впізнаваних технологій у сучасній розробці застосунків. Часто можна почути, що вона є найпопулярнішою базою даних, яку використовує велика кількість професіоналів.

Базу даних загалом можна описати як структурований набір даних. Наприклад, реляційна база даних – це звичайне цифрове сховище, що опрацьовує дані, зберігаючи їх відповідно до реляційної моделі. У такій моделі дані організовані в таблиці, що містять рядки та стовпці.

MySQL можна описати як невід’ємну частину багатьох популярних програмних забезпечень для створення та підтримки. Її використовують такі популярні для Інтернету вебсайти, як Facebook, Wikipedia, Twitter та YouTube.

Оскільки MySQL в технічному плані існує доволі довго, сьогодні її не можна вважати новинкою серед розробників. Вона стикається із сильною конкуренцією з боку Microsoft SQL Server і PostgreSQL, але продовжує виконувати роботу, через свою стабільність та роки справної роботи.

Згідно з опитуванням розробників Stack Overflow 2020 (одним з найбільш популярних сайтів серед програмістів), MySQL все ще вважалася найпопулярнішою технологією баз даних серед професійних розробників.

Довговічність і популярність означають, що MySQL можна вважати відмінним варіантом для невеликих підприємств або організацій, які не мають великої чи складної команди обробки даних. Що найкраще підходить для обраного проекту.

Обравши СУБД розробки, далі для проектування БД потрібно обрати сутності, з якими буде вестись робота. Вони виводяться за допомогою оглядів та досліджень, створених у попередніх пунктах:

- 1) suppliers
- 2) products
- 3) inventory_logs
- 4) orders
- 5) order_items
- 6) shipments

Тоді для показу моделей зазвичай використовують ER-діаграми трьох типів: концептуальна, логічна та фізична. Далі буде показано кожен з них [10].

Концептуальна модель – це неформальне представлення бізнесу, подане у спосіб, зрозумілий користувачам. Вона складатиметься з класів сутностей. Найчастіше її вигляд подається як діаграми із сутностями та зв’язками.

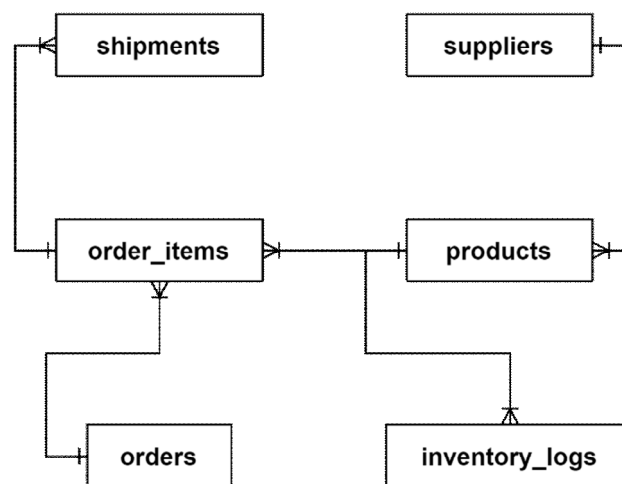


Рис. 7. Діаграма із сутностями та зв'язками

Логічна модель (LM) формалізує КМ у структури даних і обмеження цілісності. Вона повинна включати всі структури даних і обмеження цілісності для даних (всі обмеження, а не лише ту підмножину обмежень, які легко визначити в більшості доступних систем керування базами даних) [11].

LM може бути представлена як реляційна модель даних (RDM). На цьому етапі розробки доречно описати атрибути та їх приблизні типи даних.

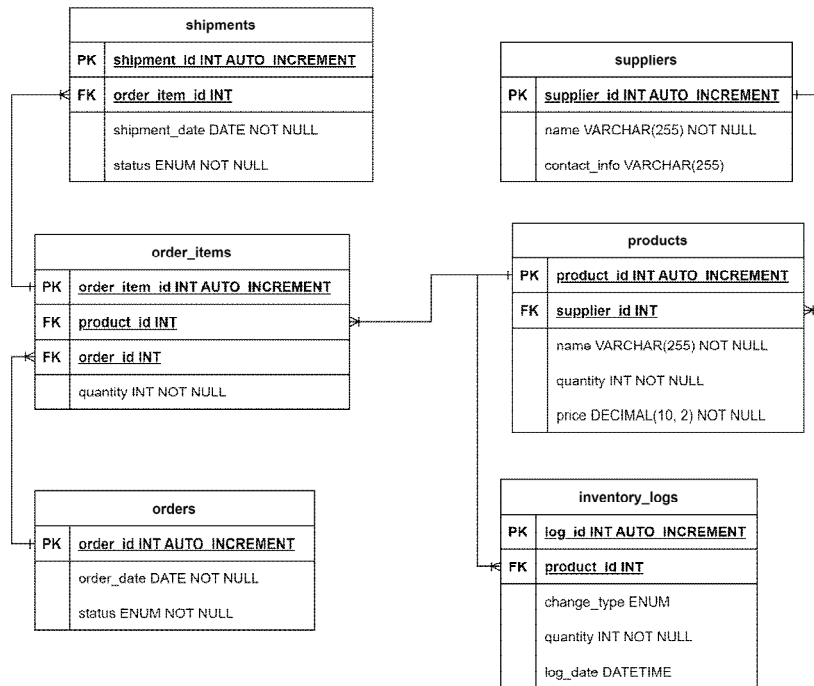


Рис. 8. Реляційна модель даних

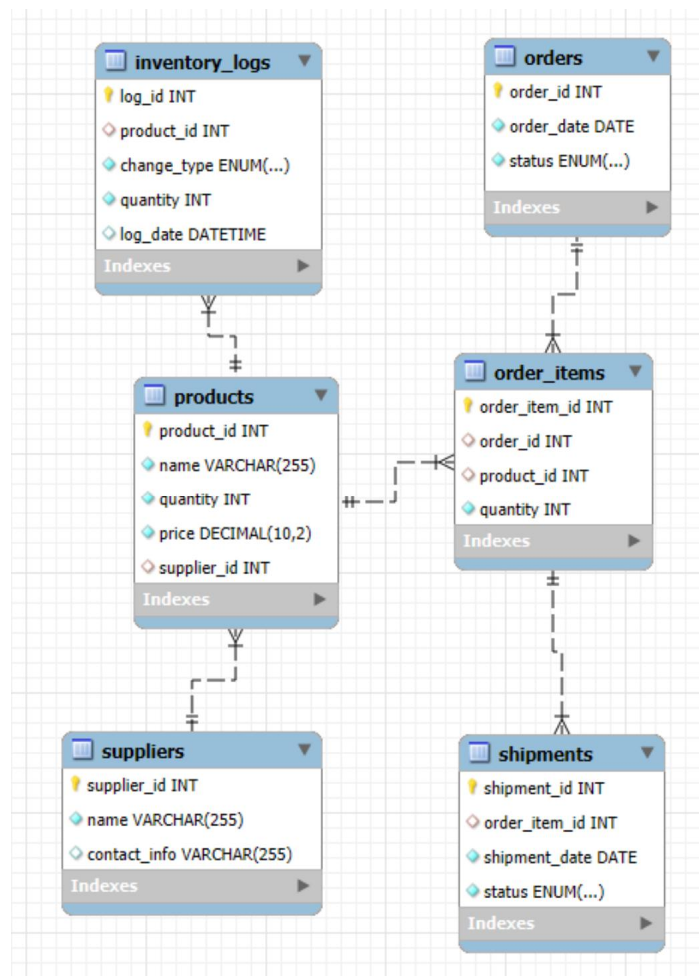


Рис. 9. Вигляд фізичної моделі бази даних

Останньою моделлю є фізична модель (PM) – це представлення LM на певному обладнанні та системі управління базою даних. Тобто є діаграмою, що показує вигляд спроектованої бази даних саме з СУБД [11].

Зазвичай цю БД можна скласти завдяки внутрішнім інструментам та звідти вивести сформовану модель бази даних. Або навпаки сформувати SQL-код та тоді завдяки інструменту Reverse Engineer отримати фізичну діаграму.

Реалізована БД є ідеальною для управління складським приміщенням. Зберігаючи в собі різні сутності та атрибути, що в повноті описують всі деталі предметної області та служитимуть для якісного зберігання даних системи.

Порівняльний аналіз альтернативних систем

Загалом можна знайти різноманітні системи для управління складом, але надалі будуть оглянуті тільки найпопулярніші з них, кожна з яких має певні переваги та недоліки.

Зазвичай компанії обирають такі системи: Oracle WMS, ABC Inventory, LMN Warehouse Manager. Далі, провівши ретельний аналіз цих систем, можна виявити їхні особливості.

Oracle WMS є одним з кращих додатків, який є створеним з орієнтацією на великі підприємства з великою кількістю операцій. Але для малих та середніх компаній його налаштування зазвичай було б надто дорогим через складність його налаштування та тривалу інтеграцію. Також велика кількість користувачів мають проблеми з нестачею гнучкості у функціях системи, що перешкоджає його налаштуванню для конкретних бізнес-вимог. Наприклад, відсутність налаштування звітів або окремих процесів може призвести до додаткових витрат часу та ресурсів. У пропонованій системі це буде покращено завдяки вбудованим підказкам та спрощеному інтерфейсі, де можна обирати, які саме функції потрібні користувачу і яку будуть відображатися на екрані.

Враховуючи масштабність системи Oracle WMS та кількість її функцій, які постійно працюють для підтримки роботи цієї системи, було вирішено провести аналіз продуктивності, який включав порівняння використання ресурсів під час виконання стандартних операцій, таких як облік товарів і моніторинг запасів. Було виявлено, що Oracle WMS споживає в середньому на 25 % більше оперативної пам'яті при обробці великих баз даних, що призводить до більшого навантаження на сервер, яке збільшує вартість обслуговування такої системи. Завдяки меншій кількості функцій, які більшість підприємств не використовують, та запобігання роботи функцій на фоні пропонована система забезпечує до 25 % менше використання ресурсів сервера, підтримуючи швидкість обробки навіть при зростанні обсягу даних.

Також був проведений аналіз продуктивності користувачів у сценаріях реальних завдань: було проведено тестові сценарії, зокрема, для виконання операцій обліку запасів, приймання та переміщення товарів. Наприклад, система ABC Inventory часто має складні меню, які збільшують кількість кліків для доступу до базових функцій. У пропонованій системі передбачено спрощене меню з єдиним доступом до важливих інструментів, що знижує кількість необхідних дій на 30 % і, таким чином, прискорює роботу персоналу.

Хоча ABC Inventory забезпечує більшу гнучкість і є загалом простішим у використанні, ніж Oracle WMS, але в ньому відсутні багато функцій автоматизації та аналітики. Часто можна стикнутись з проблемою, що системі бракує інструментів для прогнозування потреб певних товарів, що робить її менш ефективною при роботі з великими даними. Це може стати проблемою у підприємств, які прагнуть покращення складських процесів та вдосконалення фінансового обліку.

LMN Warehouse Manager можна також назвати одним з популярних додатків при пошуку системи складських приміщень. Додаток містить широкий функціональний спектр, до якого входить перегляд зміни запасів та робота з доставкою. Проте недоліком системи є порівняно складний інтерфейс, через який у користувачів можуть виникнути проблеми з виконанням найпростіших дій, особливо в період впровадження системи на підприємстві. Часто зазначається необхідність довгого навчання для того, щоб повністю освоїти можливості системи. Розроблена система буде надавати шаблони з налаштуваннями для простішого та швидшого налаштування, впровадження та використання такої системи. Ці шаблони будуть додаватися за допомогою спільноти користувачів та враховуючи потреби і вимоги підприємств.

Всі перераховані системи вимагають великих фінансових і ресурсних затрат для їх впровадження та постійного обслуговування. Розроблена система не тільки забезпечує приблизно скорочення 20 % часу обробки основних операцій, але й допоможе значно зменшити ціну за оплату системи, через вигідніші пропозиції для купівель, в яких підприємство платить лише за потрібні йому функції.

Загалом після проведення порівняльного аналізу можна дійти висновку, що теперішні альтернативні рішення мають низку недоліків, які впливають на вибір серед продуктів. До головних недоліків можна віднести такі:

- висока вартість впровадження;
- обмежені можливості для автоматизації;
- недостатня гнучкість у налаштуваннях;
- складний інтерфейс, що ускладнює їх використання.

Беручи до уваги вищезазначене, можна впевнено зробити висновок, що для реалізації рішень цих проблем проєктована система була б розроблена з акцентом на проблемні моменти, які були знайдені в альтернативах.

Обґрунтування вибору технологій розробки вебсайту

При створенні настільки серйозної системи управління складом є важливим рішення про використання технологій. Тому для створюваного вебдодатку важливо обрати найбільш якісні та сучасні технології, в цьому випадку це буде Node.js, Vue, і MySQL. Вибір цих технологій найперше ґрунтувався на їх ефективності у вирішенні основних цілей проєкту.

Node.js ефективно обробляє різні за обсягом дані за допомогою обробки запитів на основі прописаних подій. Він є ідеальним рішенням для систем, які працюють з великою кількістю користувачів та їх запитів, що дозволяє підвищити загальну продуктивність системи.

Для створення інтерфейсу найкраще підходить Vue, зважаючи на його здатність створювати швидкі та інтерактивні сторінки для бажаного проєкту. Завдяки компонентній структурі Vue стало можливим створити зручний для користувача інтерфейс, який буде зручним для масштабування та підтримки в майбутньому.

Та на останок був обраний СУБД для проєкту – MySQL є однією з надійніших систем управління базами даних, що надає можливість зберігати велику кількість даних та проводити з ними бажану обробку. MySQL допомагає проводити складні запити на вибірку, оновлення, видалення та додатково має високу продуктивність, що є ледве не найважливішим критерієм при виборі СУБД для роботи складських систем.

Отже, вибрані технології дозволять реалізувати задуману систему в повному обсязі, що відповідатиме всім вимогам сучасного складського бізнесу: надійність, швидкість та можливість масштабування.

Опис функціональних можливостей та роботи системи

Передбачена система управління складом має багато функцій, які допоможуть керувати товарними запасами та проводити потрібні операції над ними. З основних функцій системи можна перелічити такі:

1. Облік товарів та запасів – система повинна дозволити відстежувати наявні товари, що доступні в режимі реального часу, відзначати зміни в його русі (надходження, переміщення, списання) і вести чіткий облік кожної операції;

2. Контроль запасів – система моніторингу повинна автоматично повідомляти про необхідність поповнення певних товарів, що мінімізує їхню нестачу. Реалізувати цю функцію найлегше завдяки використанню SQL-запитів;

3. Автоматизація процесів – автоматичні процеси повинні бути оброблені для виконання функцій обробки, тобто приймання та відправки товарів, зводячи до мінімуму людський фактор і підвищуючи точність операцій;

4. Управління замовленнями – користувачі мають мати можливість легко створювати та керувати замовленнями на доставку або відвантаження товарів. Система автоматично створює звіти та оновлює поточний статус обраних замовлень.

Крім цього, в майбутньому система повинна дозволити інтеграцію з іншими бізнес-системами для можливості обміну даними, що дозволить покращити систему. Кожна частина системи повинна працювати разом з іншими, забезпечуючи безперервність роботи та не сповільнюючи час, необхідний для обробки інформації.

Основною перевагою цієї системи є її високий ступінь автоматизації. Порівняно з іншими системами вона дозволить автоматично виконувати більшу частину операцій, таких як відстеження запасів, попередження про брак товару та управління замовленнями. Це значно полегшує навантаження на персонал і знижує загальні витрати підприємства.

Другою важливою перевагою є модульна архітектура. Завдяки їй систему можна легко адаптувати під необхідні потреби користувача, дозволивши додавати або змінювати модулі без необхідності перероблювання великої частини програмного коду. Це робить систему гнучкішою та економічно вигіднішою, порівнюючи з конкурентами на ринку.

Крім того, система повинна бути побудована на основі сучасних технологій, що також дозволить досягти кращої продуктивності та стабільності роботи. Сучасні інструменти управління та керування за допомогою інтерфейсу зробиють її зручною та простою для використання, що значно скорочує час на навчання.

Результати дослідження

Під час проведення дослідження можна дійти висновку, що впровадження нової системи управління із вебдодатком будуть надавати суттєві покращення у складських процесах. Зокрема:

- час обробки замовлень скоротився від 20 до 13–14 хвилин, що свідчить про підвищення швидкості роботи системи на 25 % порівняно з попередніми системами. Цього було досягнуто завдяки оптимізованим алгоритмам обробки замовлень та інтуїтивно зрозумілому інтерфейсу;
- зниження операційних витрат на 15 % стало можливим через автоматизовані процеси обліку та управління інвентарем, які зменшують потребу в ручній праці та мінімізують витрати на корекцію помилок;
- покращення синхронізації даних на 20 % досягається завдяки інтеграції з іншими системами та реальному оновленню інформації про запаси, що полегшує пошук товарів та прийняття швидких рішень.

Наведемо основні результати.

Аналіз проблематики: під час виконання цього етапу було виявлено головні недоліки ігнорування використання системи для автоматизації, з них можна виділити зайві витрати часу на обробку даних, імовірність отримання помилок при веденні обліку. Логічно зрозуміти, що розв'язання перерахованих проблем потребує сучасної системи, що могла б автоматизувати наступні процеси.

Постановка завдання: були сформовані основні цілі, що включатимуть огляд готових рішень для додатку, аналіз архітектури проєкту завдяки створенню діаграм IDEF0 та UML, проєктування бази даних для збереження всіх даних, вибір технологій для розробки та вибір функціоналу, що потребує реалізації.

Формування вимог: були визначені основні функціональні та нефункціональні вимоги до системи.

Моделювання процесів в нотації IDEF0: за допомогою цих діаграм були показані основні бізнес-процеси під час використання додатку. Цей етап дозволив визначити кроки під час обробки товарів на складі, відстежуючи основні кроки під час користування вебсайтом.

Проєктування архітектури через UML: побудова UML-діаграм допомогли відобразити загальну архітектуру системи. Були створені діаграми: варіантів використання, послідовності, розгортання. Створення цих діаграм для програмного забезпечення дозволило показати взаємодію між різними компонентами теоретичної системи.

Структурна реалізація бази даних: далі була створена модель для БД, що підтримує у собі всі необхідні дані для збереження та обліку товарів на складі. Основні таблиці включають інформацію про товари, замовлення, постачальників.

Порівняльний аналіз сучасних систем: проведено огляд кількох систем управління складом. Виявлені ключові недоліки для майбутньої розробки покращеного додатку.

Огляд технологій розробки вебсайту: було проведено детальний аналіз стеку для побудови системи. Були обрано відповідні технології для серверної – node.js та клієнтської – vue.js частин. Це було виконано з урахуванням вимог до загальної продуктивності, масштабованості вебсайту.

Опис функціональних можливостей: наприкінці було надано опис можливих функцій майбутньої системи, що включатимуть автоматизацію обліку замовлень та товарів, контроль за кількістю запасів, інтеграцію з іншими системами, а також імовірно розширення системи для створення нових функцій, як прогнозування попиту та управління доставками.

Під час проведеного дослідження було змодельовано загальну архітектуру додатку для управління складськими приміщеннями, що відповідають сучасним вимогам до автоматизації. Визначено головні аспекти функціонування цих систем та описано вибір технологій для розробки вебсайту.

Висновки

За результатами дослідження особливостей розроблення вебдодатку для нової системи управління складом були враховані основні аспекти, необхідні для підвищення ефективності складських операцій. Під час дослідження було виконано аналіз функціональних вимог, оглянути альтернативні додатки доступні на ринку, спроектовано базу даних, створено діаграми IDEF0 та UML, що описують, як система працює і як її частини взаємодіють між собою.

У процесі опису аналізу функціональних можливостей були враховані основні елементи, необхідні для підвищення ефективності складських операцій. З таких завдань можна виділити автоматизацію обліку товарів, керування запасами, автоматичне створення звітів та теоретичну інтеграцію з іншими бізнес-платформами. Завдяки добре структурованій базі даних з таблицями, що передбачають зберігання інформації про товари, постачання, замовлення та клієнтів, можна забезпечити точний облік та швидкий доступ до бажаної інформації.

Одним з найважливіших моментів проєктування системи була розробка моделей в нотатції UML, зокрема, діаграма варіантів використання та діаграма послідовності, що дозволило відобразити взаємозв'язки між теоретичними компонентами системи та вивести основні сценарії для використання системи. Діаграми IDEF0 водночас допомогли більш детально оглянути процеси управління запасами та самої обробки замовлень в системі, що показало можливості майбутньої автоматизації цих процесів.

Проєктування системи надало можливість визначити потрібні для її реалізації компоненти та взаємодію між ними. Дослідження показали, що система має потенціал для суттєвого покращення управління складськими процесами через додаткові налаштування, автоматизацію одноманітних операцій і можливості інтеграції із сучасними технологіями.

Порівняно із системою Oracle WMS, пропонується система скорочує час обробки замовлень приблизно на 30 %. Якщо Oracle WMS зазвичай потребує близько 20 хвилин для обробки одного замовлення, проєктована система дозволяє зменшити цей час до 13–14 хвилин завдяки автоматизованим алгоритмам обробки та покращеному користувацькому інтерфейсу, що зекономить підприємству час та кошти, покращивши загальну ефективність бізнес-процесів.

Завдяки отриманим результатам можна зробити висновок, що запропонований вебдодаток відповідає заданим вимогам сучасних засобів, що мають справу з керуванням складом. Після завершення розробки та впровадження вебсистема може стати якісним та корисним інструментом для покращення бухгалтерського обліку, зробивши його більш точним, замовлення оброблятиметься швидше, що зекономить підприємству час та кошти, покращить загальну ефективність бізнес-процесів.

Загалом система показує покращення для обробки замовлень та відстеження запасів порівняно з іншими програмами, як Oracle WMS, ABC Inventory та LMN Warehouse Manager. Вона прискорює обробку замовлень орієнтовно на 30 %, скорочує час очікування через синхронізацію даних на 20 % і знижує операційні витрати на 15 % завдяки автоматизації та спрощеному інтерфейсу.

Список літератури

1. Tikwayo L. N., Mathaba T. N. D. *Applications of Industry 4.0 Technologies in Warehouse Management: A Systematic Literature Review. Logistics* 2023, 7, 24. URL: <https://doi.org/10.3390/logistics7020024>

2. Choy K. L., Ho G. T. S. & Lee C. K. H. A RFID-based storage assignment system for enhancing the efficiency of order picking. *J Intell Manuf* 28, 111–129 (2017). URL: <https://doi.org/10.1007/s10845-014-0965-9>
3. Custodio L., Machado R. Flexible automated warehouse: a literature review and an innovative framework. *Int J Adv Manuf Technol* 106, 533–558 (2020). URL: <https://doi.org/10.1007/s00170-019-04588-z>
4. Bal A., Satoglu S. I. (2018). *Advances in Virtual Factory Research and Applications*. In: *Industry 4.0: Managing The Digital Transformation*. Springer Series in Advanced Manufacturing. Springer, Cham. URL: https://doi.org/10.1007/978-3-319-57870-5_14
5. Godlevskiy M. D., Orlovskiy D. L. & Kopp A. M. (2018). STRUCTURAL ANALYSIS AND OPTIMIZATION OF IDEF0 FUNCTIONAL BUSINESS PROCESS MODELS. *Radio Electronics, Computer Science, Control*, (3). URL: <https://doi.org/10.15588/1607-3274-2018-3-6>
6. Koç H., Erdoğan A. M., Barjakly Y., Peker S. UML Diagrams in Software Engineering Research: A Systematic Literature Review. *Proceedings 2021*, 74, 13. URL: <https://doi.org/10.3390/proceedings2021074013>
7. GeeksforGeeks. Use Case Diagram. GeeksforGeeks. URL: <https://www.geeksforgeeks.org/use-case-diagram/>
8. Ambler Scott W. “UML Activity Diagrams”. Chapter. In *The Elements of UMLTM 2.0 Style*, 113–31. URL: <https://doi.org/10.1017/CBO9780511817533.011>
9. TechTarget. MySQL. TechTarget. URL: <https://www.techtarget.com/searchoracle/definition/MySQL>.
10. Frantiska J. (2018). Entity-Relationship Diagrams. In: *Visualization Tools for Learning Environment Development*. SpringerBriefs in Educational Communications and Technology. Springer, Cham. URL: https://doi.org/10.1007/978-3-319-67440-7_4
11. GoodData. (2020). Physical vs. Logical Data Model. *GoodData Blog*. URL: <https://www.gooddata.com/blog/physical-vs-logical-data-model/>

WEB APPLICATION FOR WAREHOUSE MANAGEMENT SYSTEM

M. Kovalyshyn, Y. Paramud

Lviv Polytechnic National University,
Department of Computer Engineering

E-mail: marian.kovalyshyn.mkisp.2023@lpnu.ua, yaroslav.s.paramud@lpnu.ua

© Kovalyshyn M., Paramud Y., 2024

The article examines the overall process of designing and future development of a warehouse management system, which is being developed with a focus on improving its operational efficiency. The main emphasis was placed on modeling methods of key business processes to optimize the main operations of the warehouse, as well as a detailed review of the desired database structure, which would serve as a datastore for storing various data about goods, their movement, and operations involving them. Various alternative systems on similar topics were also reviewed, describing their advantages and disadvantages, from which the necessary functionalities of the system can be derived.

Next, a description of the software’s functional capabilities was provided, which ensured the automation of warehouse operations, including order processing, inventory management, and monitoring changes between operations. This was achieved through textual descriptions and the implementation of diagrams from various notations. The main results of the research demonstrate the importance of implementing a high-quality and modern web application for warehouse management to increase productivity, avoid errors, and ensure accuracy during operations.

Keywords: business processes, database, inventory management, warehouse management, web application, UML, IDEF0, ER.