

Міністерство освіти і науки України

Тернопільський національний технічний університет імені Івана Пулюя

Комп'ютерно-інформаційних систем і програмної інженерії

(повна назва факультету)

Програмної інженерії

(повна назва кафедри)

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття освітнього ступеня

бакалавр

(назва освітнього ступеня)

на тему: Розробка програмного забезпечення для автоматизації складських процесів на базі web-технологій

Виконав(ла): студент(ка) 4 курсу, групи СПс-42 спеціальності 121 Програмна інженерія

(шифр і назва спеціальності)

	(підпис)	Гаврас Л.П.	(прізвище та ініціали)
Керівник	(підпис)	Коноваленко І.В.	(прізвище та ініціали)
Нормоконтроль	(підпис)	Стоянов Ю.М.	(прізвище та ініціали)
Завідувач кафедри	(підпис)	Петрик М.Р.	(прізвище та ініціали)
Рецензент	(підпис)		(прізвище та ініціали)

Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя

Факультет Комп'ютерно-інформаційних систем і програмної інженерії
(повна назва факультету)

Кафедра Програмної інженерії
(повна назва кафедри)

ЗАТВЕРДЖУЮ

Завідувач кафедри

Петрик М.Р.
(підпис) (прізвище та ініціали)
« » 2024 р.

**З А В Д А Н Н Я
НА КВАЛІФІКАЦІЙНУ РОБОТУ**

на здобуття освітнього ступеня бакалавр
(назва освітнього ступеня)

за спеціальністю 121 Програмна інженерія
(шифр і назва спеціальності)

студенту Гаврасу Любомиру Петровичу
(прізвище, ім'я, по батькові)

1. Тема роботи Розробка програмного забезпечення для автоматизації
складських процесів на базі web-технологій

Керівник роботи Коноваленко Ігор Володимирович, к.т.н., доц.
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

Затверджені наказом ректора від « 15 » квітня 2024 року № 4/7-344

2. Термін подання студентом завершеної роботи _____

3. Вихідні дані до роботи Вимоги замовника, технічні умови, технічне завдання

4. Зміст роботи (перелік питань, які потрібно розробити)

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень, слайдів)

6. Консультанти розділів роботи

[illegible]

7. Дата видачі завдання

КАЛЕНДАРНИЙ ПЛАН

[illegible]

Студент _____ Гаврас Л.П.
(підпис) (прізвище та ініціали)

Керівник роботи

(підпис)

Коноваленко І.В.

(прізвище та ініціали)

РЕФЕРАТ

Кваліфікаційна робота бакалавра містить:

с. 47, рис. 10, табл. 17, джер. 28

Ключові слова: розробка, програмний засіб, складське приміщення, сервіс, управління, датчик, база даних.

В даній роботі розглянуто варіант покращення автоматизації складського приміщення. Система спроектована з урахуванням результатів аналізу дослідження предметної області та аналізу вже існуючих на ринку варіантів управління, що дало змогу виділення ключові моменти для впровадження у власну систему управління.

ABSTRACT

Key words: development, software, warehouse, service, management, sensor, database.

In this work, the option of improving the automation of the warehouse is considered. The system is designed taking into account the results of the analysis of the research of the subject area and the analysis of management options already existing on the market, which made it possible to highlight the key points for implementation in your own management system.

ЗМІСТ

ВСТУП

1 РОЗРОБКА ПРОГРАМНОЇ СИСТЕМИ

1.1 Загальна характеристика підприємства

1.2. Опис процесу відстеження складських залишків

1.3. Аналіз рівня технічної та програмної оснащеності підприємства

1.4. Обґрунтування необхідності розробки web-сервісу відстеження складських залишків

1.5 Проектування структури бази даних

1.6 Огляд програмних аналогів

1.6.1. Архітектура платформ Java та .Net

1.6.2. MS Visual Studio.

1.6.3. Axis і Apache Tomcat.

1.6.4. Сервери Apache та IIS

1.7 Постановка комплексу завдань розробки web-сервіса

1.8 Вибір інструментальних програмних засобів розробки

1.9. Висновок до розділу 1

2 ТЕСТУВАННЯ ПРОГРАМНОГО ЗАСОБУ

3 БЕЗПЕКА ЖИТТЄДІЯЛЬНОСТІ, ОСНОВИ ОХОРОНИ ПРАЦІ

3.1. Ергономічні проблеми безпеки життєдіяльності

3.2 Гігієнічні вимоги до організації та обладнання робочих місць з ВДТ

ЗАГАЛЬНІ ВИСНОВКИ

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

ДОДАТКИ

ВСТУП

У світі інформація є найважливішим об'єктом взаємодії у суспільстві, а інформаційні технології використовуються практично у всіх сферах людської діяльності. Розвиток торгівлі та послуг досяг того рівня, коли без застосування сучасних технологій складно встояти в конкурентній боротьбі.

Багато компаній сьогодні замислюються про автоматизацію складських процесів, оскільки існуючі схеми їх роботи, особливо ручну працю, перестають справлятися зі зростаючими оборотами. Співробітникам доводиться стикатися з великими потоками інформації та тримати все у пам'яті, через це відбуваються помилки, а люди перебувають у нервовій напрузі.

Впровадження автоматизованої системи на підприємствах сприяють зменшенню штату, надає можливість залучити кваліфікований персонал, а також ефективно керувати підприємством, швидко позбавлятися проблем і негативних напрямків, що виникають.

У цій кваліфікаційній роботі буде розроблено web-сервіс для ведення складського обліку на підприємстві.

Web-додаток – має клієнт-серверну архітектуру. Тут клієнтом є браузер, а сервером – веб-сервер. Його логіка розподіляється між клієнтом і сервером, Логіка web-додатка розподілена між сервером, на якому зберігаються дані, та клієнтом.

Переваги даного підходу:

1. Можливість запустити web-додаток абсолютно з будь-якого комп'ютера підключеного до інтернету, так як web-додаток буде працювати через браузер, не тільки windows комп'ютера, а й linux та Mac, мобільні пристрої. Web-додаток дає крос-платформність.

2. Можливість викласти web-додаток на сайті компанії.

3. Web-додаток не треба встановлювати та завантажувати, багато людей бояться завантажувати програми з мережі та обґрунтовано це роблять. Web-

додаток дозволяє їм не відчувати страху і допомагає спростити запуск програми в мережі.

4. Web - додатки це сучасна тенденція. Якщо зважати на все вищеописане, то можна зрозуміти, що актуальність створення інформаційної системи для обліку руху товарів підприємства дуже висока на сьогоднішній день.

Об'єктом дослідження є web-сервіс для відстеження складських залишків.

Метою роботи є розробка web-сервісу для відстеження складських залишків з метою надання оперативної інформації для керівництва та співробітникам підприємства.

Досягнення поставленої мети вимагає:

- вивчення обліку руху товарів складі;
 - обґрунтування та вибір засобу розробки для створення web-сервісу;
 - формування вимог до проєктованого web-сервісу;
 - розроблення web-сервісу для відстеження складських залишків
- «WEBСклад»: розробка контенту та дизайну сайту, проєктування структури бази даних;
- тестування розробленого web-сервісу «WEBСклад»;
 - розроблення програмної документації.

1. РОЗРОБКА ПРОГРАМНОЇ СИСТЕМИ

1.1 Загальна характеристика підприємства

Для виконання цієї роботи розглянемо підприємство основним видом діяльності якого є оптовий та дрібнооптовий продаж канцтоварів.

Підприємство займається продажем та доставкою канцелярії, паперу, картриджів.

Головне завдання підприємства – створити комфортні умови для роботи офісу. Звертаючись до компанії, покупець отримує:

- Зручний спосіб замовлення товару, за допомогою інтернет-каталогу. Покупець може переглянути фотографії товару на екрані комп'ютера і відразу оформити замовлення.

- Весь товар, що доставляється, акуратно упакований у коробки, це допоможе заощадити час покупця при перевірці замовлення.

- Швидку обробку замовлень надісланих через інтернет каталог, обробку таких замовлень автоматизовано і тому вони обробляються насамперед.

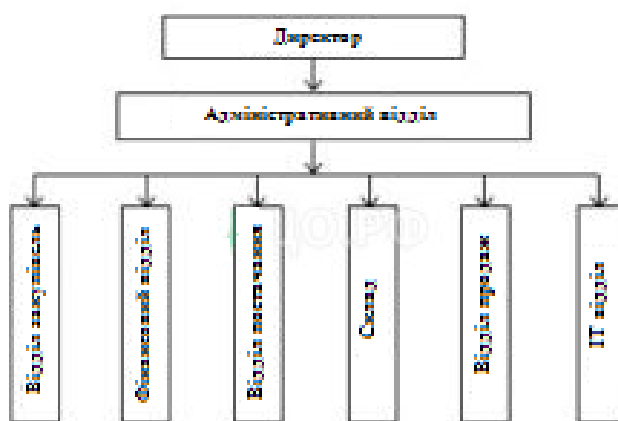


Рисунок 1.1 – Організаційна структура оптово-роздрібного підприємства

Адміністративний відділ. Завданнями відділу є організація всіх відділів підприємства, їх злагодженої та сталої взаємодії. Адміністративним відділом здійснюється загальне керівництво фірмою, що передбачає безпосередній контроль над діяльністю підприємства, а також загальне стратегічне керівництво.

Фінансовий відділ – окремий структурний підрозділ. Його очолює головний бухгалтер, адміністративно підпорядкований директору.

Відділ продажів До нього входять менеджери з продажу, які приймають замовлення від покупців. Головним обов'язком керуючого відділом продажів є керівництво, контроль та координація робіт із продажу товарів. Керуючий розробляє посадові інструкції для кожного працівника, складанням плану продажу та проведенням заходів з метою розширення ринку збуту та збільшення обсягів продажу.

Менеджер з продажу здійснює організацію навчання працюючих та надає їм усіляку допомогу у здійсненні їхньої поточної діяльності.

Відділ закупівель – окремий підрозділ, це вхідна ланка у логістиці системи.

Склад - це структурний підрозділ, що займається складуванням, розміщенням та зберіганням товарів.

Відділ постачання. Це структурний підрозділ, основною функцією якого є доставка продукції до клієнтів, які потребують доставки придбаної продукції.

ІТ-відділ. Керівництво відділом покладено на системного адміністратора. Призначення – підтримка автоматизації бізнес-процесів шляхом забезпечення необхідними технічними та програмними засобами.

1.2. Опис процесу відстеження складських залишків

Склади — це будівлі, споруди та різноманітні пристрої, призначені для приймання, розміщення та зберігання товарів, що надійшли на них, підготовки їх до споживання та відпуски споживачеві.

Основне призначення складу – накопичення запасів, їх збереження та забезпечення безперебійного та розподіленого рівномірно в часі виконання замовлень клієнтів [1].

У цілому до складських операцій відносять: розгрузка, прийом товару, розміщення та зберігання його, видача товару, комплектування, пакування, переміщення всередині складу. На складі обслуговуються три напрями: вхід, вихід, внутрішній.

При підтримці вхідного потоку виконуються операції з розвантаження транспорту, контролю кількості та якості вантажу, що прибув. Обслуговування вихідного потоку включає завантаження транспорту.

Внутрішній потік пересувається всередині складу [2].

Технологічний процес руху товарів на складі складається з наступних основних операцій: розвантаження, прийом на зберігання, перевірка якості, розміщення по місцях зберігання, відпуск товарів з відділів зберігання (Додаток 1) [3].

Розвантаження. Це операції зі звільнення транспортного засобу від вантажу у приймальному відділенні. Технологія виконання вантажних робіт на складі залежна від типу наявних засобів, характеру вантажу, а також від виду механізованості процесу. У приймальному відділі рекомендується передбачити зони: розвантаження, розпакування та сортування товару, що прибув; тимчасове зберігання товару; руху та розміщення засобів механізації та транспортування; перевірки товарів відсутність механічних домішок; карантинна; зберігання тари.

Прийом товарів здійснюється у приймальному відділенні відповідно до нормативної документації. Для приймання і передачі продукції, що надійшла на складі створюється приймальна комісія, що складається не менше ніж з трьох осіб. Головою приймальної комісії є завідувач складу [1]. Особи, здійснюють приймання, зобов'язані добре знати правила приймання продукції за кількістю, якістю та вартістю, основні та особливі умови поставки товару, фізичні властивості товару.

Основними функціями ділянки приймання є:

- прийом вантажів за кількістю, комплектністю та якістю;
- перевірка супровідної документації;
- ведення претензійної роботи;
- розподіл вантажу за місцес зберігання способами, які використовуються складі та умовами зберігання окремих груп товарів.

Крім того, на ділянку приймання можуть бути покладені функції відбору проб товару на аналіз якості; пакетування вантажів, комплектація укрупнених одиниць для збереження на складі, а також розукомплектування останніх із тією самою метою; тимчасового зберігання (накопичення) вантажу для оперативного розподілу на основних складських площах [4]. Усі поставки повинні супроводжуватися документами, що дозволяють встановити: дату відвантаження, найменування товару, поставлену кількість, ціну відпущеного препарату або вартість препарату, назву та адресу постачальника та одержувача. Розміщення на місцях зберігання дозволяє реалізувати функції складу, залежить від дотримання умов зберігання, контроль за термінами придатності, цілісністю вторинної упаковки. Вантажна ємність ділянки зберігання залежить від розмірів, а й від обраного способу зберігання – стелажного, на піддонах, в контейнерах. Залежно від обраного способу зберігання та застосування того чи іншого обладнання можна досягти оптимального використання ділянки зберігання вантажів з урахуванням їхньої подальшої обробки. Як правило, на ділянках зберігання складу здійснюється також комплектація замовлень та передача їх у відділ експедиції [5].

Відпуск товару з відділу зберігання різним організаціям виходячи з замовлень з допомогою різних засобів зв'язку. Для цього створюється ділянка експедиції, яка є окремим приміщенням, призначеним для: обліку вантажів, що відправляються, тимчасового складання, формування супровідної документації, що містить достовірну інформацію: про постачальника, покупця, дату поставки, найменування товару, виробника товару, кількість.

Такими документами може бути товарно-транспортні накладні, рахунки – фактури.

1.3. Аналіз рівня технічної та програмної оснащеності підприємства

У компанії використовується наступне технічне оснащення: міська телефонна лінія, IP-телефони/факси, МФУ — 30 ПК, 2 сервери під керуванням Windows Server 2012, мережевий зв'язок.

Таблиця 1.1 – Конфігурація ПК підприємства

Атрибут	Опис
ПК директора	
Операційна система Windows 7 Professional SP1 x64	
Процесор	Intel Core i7-4790 16 Гб Б, 7200
RAM	16 Гб (DDR3-1600DDR3 SDRAM)
Відеоадаптер	NVIDIA GeForce GT 640 (2048 Мб GDDR5 128 bit)
HDD	1024 Гб, 7200 RPM, SATA-II
ПК менеджера у напрямку	
Операційна система Windows 7 Professional SP1 x64	
Процесор	Intel Core i5-3470 Ivy Bridge 3.20GHz
RAM	4 Гб (DDR3-1333 DDR3 SDRAM)
Відеоадаптер	PALIT GeForce GT 630 (1024 Мб GDDR5 128 bit)
HDD	500 Гб, 7200 RPM, SATA-II

Використання технологій, які забезпечать ефективну, зручну та оперативну роботу: оперативні пошукові системи для знаходження товарів on-line, графічний редактор, ведення звітних документів та листування електронною поштою.

Отже, на момент обслуговування клієнтів використано достатній рівень автоматизації. Загалом рівень технічної та програмної оснащеності дозволяє розгорнути досить потужну інформаційну систему.

1.4. Обґрунтування необхідності розробки web-сервісу відстеження складських залишків

Автоматизація обліку на складі забезпечує більшу точність та оперативність виконання процесів на складі. Система автоматизації складу дає підприємству низку переваг, які дозволять суттєво скоротити собівартість складських операцій та підвищити ефективність складу:

- контроль залишків на складі та нових надходжень
- виключення можливості подвійного введення даних, і як наслідок скорочення помилок;
- повне та швидке отримання інформації про нинішній стан бізнесу;
- зменшення часу обробки та аналізу даних;
- наявність повної та достовірної інформаційної моделі компанії, яка дозволяє швидко визначати проблеми організації обліку та вживати певних заходів їх ліквідації;
- зменшення помилок введення і зниження ролі людського фактора;
- підвищення швидкості роботи;
- здатність обмежувати доступ персоналу до зміни чи перегляду певних даних;
- дозволяє проводити аналіз та планування діяльності підприємства або компанії;
- зниження витрат.

Система автоматизації складу дозволить координувати всі складські операції в момент їх здійснення: приймання та розміщення товару, комплектація замовлень, відвантаження. Важлива перевага автоматизованого складу полягає у скороченні частки впливу людського фактора на складські процеси та збільшення інтенсивності складських операцій. Незважаючи на підвищення темпу роботи, кількість помилок скорочується до мінімуму. Система автоматизації складу дозволяє підвищити якість, точність та оперативність виконання операцій: за

допомогою автоматизованої системи управління складом можна відстежити залишки товару на складі та отримати звіти щодо залишків на будь-яку дату [6]. Впровадження автоматизованої системи управління складом «WEBСклад» як web-сервіс, дозволить підприємству в найкоротші терміни отримувати інформацію про поточні залишки, оптимізувати процес оформлення документів з будь-якого комп'ютера, підключеного до інтернету, підвищити ефективність складського обліку в цілому.

Проектована автоматизована система «WEBСклад» є SaaS додатком.

ПЗ як послуга (SaaS) — це форма хмарних обчислень, у якій провайдер пропонує клієнту використання прикладного програмного забезпечення та керує всіма фізичними та програмними ресурсами, що використовуються програмою. Відмінною рисою SaaS порівняно з іншими моделями доставки програмного забезпечення є те, що він відокремлює «володіння програмним забезпеченням від його використання».

Доступ до SaaS зазвичай здійснюється через веб-додаток. На відміну від більшості самостійних програмних продуктів, існує лише одна версія програмного забезпечення та підтримується лише одна операційна система та конфігурація. Продукти SaaS зазвичай працюють на основі орендованої інфраструктури як послуги (IaaS) або системи платформи як послуги (PaaS), включаючи апаратне забезпечення, а іноді й операційні системи та проміжне програмне забезпечення, щоб відповідати швидкому зростанню використання, забезпечуючи миттєву та постійну доступність, яку очікують клієнти. Клієнтам SaaS надається абстракція безмежних обчислювальних ресурсів, а економія масштабу знижує вартість.

Немає конкретних практик розробки програмного забезпечення, які б відрізняли SaaS від розробки інших додатків, хоча часто акцент робиться на частому тестуванні та випусках. Архітектури SaaS, як правило, багатокористувацькі; Хоча зазвичай вони досягають високої ефективності, спільним використанням принаймні деяких ресурсів між клієнтами, нерідко пропонують повністю ізольоване середовище за додаткову плату. Загальні моделі

доходу SaaS включають безкоштовну плату, плату за підписку та плату за використання. На відміну від традиційного програмного забезпечення, рідко можна придбати постійну ліцензію на певну версію програмного забезпечення.

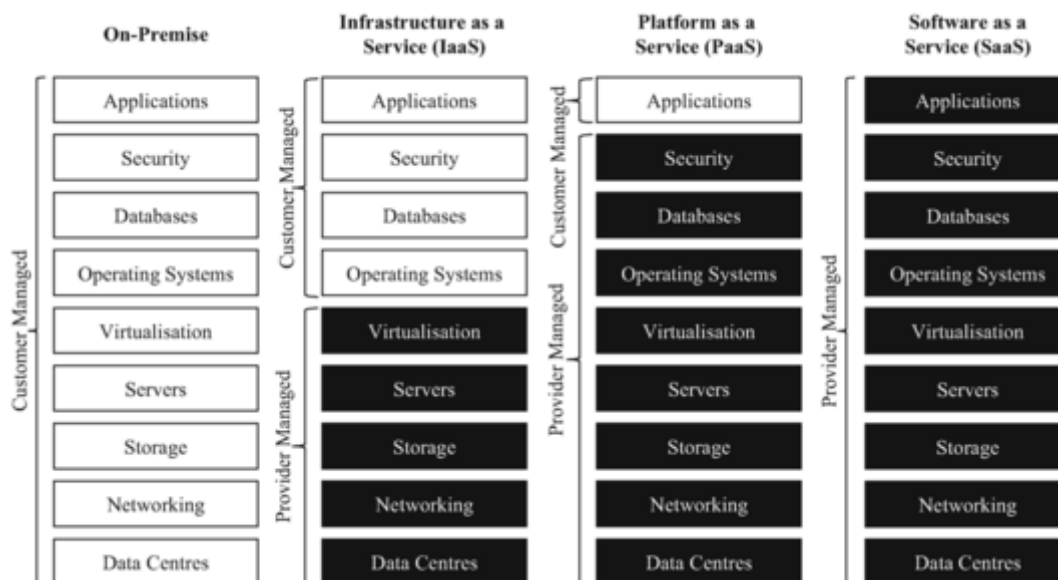


Рисунок 1.2 – Порівняння локальних, IaaS, PaaS і SaaS

Інфраструктура як послуга (IaaS) є найпростішою формою хмарних обчислень, де інфраструктурні ресурси, такі як фізичні комп'ютери, не є власністю користувача, а натомість орендуються у хмарного постачальника. Завдяки цьому ресурси інфраструктури можна швидко збільшити, а не чекати тижнями, поки комп'ютери будуть доставлені та налаштовані. IaaS потребує часу та досвіду, щоб використовувати інфраструктуру у вигляді операційних систем і програм. Платформа як послуга (PaaS) включає ОС та проміжне програмне забезпечення, але не програми. Провайдери SaaS зазвичай використовують служби PaaS або IaaS, щоб запустити свої програми.

Відсутність IaaS ускладнює масштабування SaaS для різної кількості користувачів, одночасно забезпечуючи миттєвий та постійний доступ клієнта. Здебільшого кінцевий користувач використовує тільки SaaS і, йому не потрібно турбуватися щодо технічної складності фізичного обладнання та операційної системи. Оскільки, до хмарних ресурсів можна отримати доступ без впливу

людини, клієнтам SaaS надається абстракція безмежних обчислювальних ресурсів, а економія масштабу знижує вартість. Ще одна ключова особливість хмарних обчислень полягає в тому, що оновлення програмного забезпечення можна розгортати та робити доступними для всіх клієнтів майже миттєво.

Поширені моделі прибутку SaaS включають підписку та плату за користування. Для клієнтів переваги включають зниження початкових витрат, підвищену гнучкість і нижчу загальну вартість порівняно з традиційним програмним забезпеченням із безстроковими ліцензіями на програмне забезпечення. У деяких випадках високі одноразові витрати, яких вимагали продавці традиційного програмного забезпечення, були недосяжними для менших компаній, але моделі SaaS з оплатою за використання роблять програмне забезпечення доступним. За використання може стягуватися плата на основі кількості користувачів, транзакцій, обсягу використаної пам'яті або інших показників. Багато покупців віддають перевагу оплаті за користування, тому що вони вважають, що вони відносно малокористуються ПЗ, і продавець отримує вигоду, охоплюючи випадкових користувачів, які в іншому випадку не купували б програмне забезпечення. Однак це може спричинити невизначеність прибутку для продавця та збільшити накладні витрати на виставлення рахунків.

Доступ до продуктів SaaS зазвичай здійснюється через веб-браузер як загальнодоступний веб-додаток. Це означає, що клієнти можуть отримати доступ до програми будь-де з будь-якого пристрою без необхідності встановлювати чи оновлювати її.

Програми SaaS переважно пропонують протоколи інтеграції та інтерфейси прикладного програмування (API), які працюють у глобальній мережі.

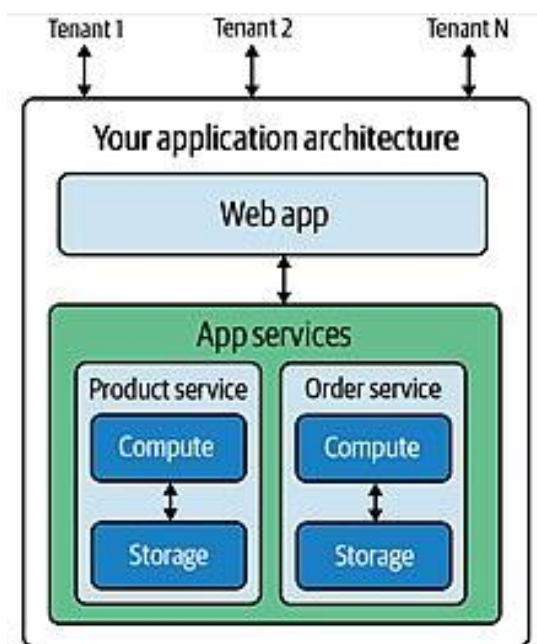


Рисунок 1.3 – Архітектура SaaS SaaS.

Усі клієнти використовують ту саму версію програмного забезпечення на одній платформі

Архітектура SaaS значно відрізняється від продукту до продукту. Незважаючи на це, більшість провайдерів SaaS пропонують мультитенантну архітектуру. У цій моделі єдина версія програми з єдиною конфігурацією (апаратне забезпечення, мережа, операційна система) використовується для всіх клієнтів («орендарів»). Це означає, що компанії не потрібно підтримувати кілька версій і конфігурацій. Зміна архітектури від того, що кожен клієнт використовує свою власну версію програмного забезпечення на власному апаратному забезпеченні, впливає на багато аспектів дизайну програми та функцій безпеки. У мультитенантній архітектурі багато ресурсів можуть використовуватися різними орендарями або спільно використовуватися кількома орендарями.

Структуру типової програми SaaS можна розділити на площину програми та керування. Продукти SaaS відрізняються тим, як ці площини розділені, які можуть бути тісно інтегровані або слабо пов'язані в моделі, керованій подіями чи повідомленнями. Площина керування відповідає за керування системою та охоплює такі функціональні можливості, як підключення орендарів, виставлення

рахунків і показники, а також систему, яку використовує провайдер SaaS для налаштування, керування та експлуатації послуги. Багато продуктів SaaS пропонуються на різних рівнях обслуговування за різними цінами, які називаються рівнями. Це також може вплинути на архітектуру для обох площин, хоча зазвичай розміщується в площині керування. На відміну від прикладної площини, послуги в контрольній площині не призначені для багатокористування. [9].

1.5 Проектування структури бази даних

База даних web-сервісу «WebСклад» представлена на СУБД MySQL. Всі таблиці бази даних сайту створено web-додатки phpMyAdmin.

PHPMyAdmin — веб-додаток з відкритим кодом, написаний мовою PHP і web-інтерфейс для адміністрування СУБД MySQL. Він дозволяє через браузер здійснювати адміністрування сервера MySQL, запускати команди SQL та переглядати вміст таблиць та баз даних. Dodatok користується великою популярністю у web-розробників, оскільки дозволяє керувати СУБД MySQL без безпосереднього введення команд SQL, надаючи дружній інтерфейс [19].

Під час створення бази даних web-сервісу «WebСклад» було створено 15 таблиць:

1. «Agent» — містить інформацію про контрагентів: постачальників та покупців.

Таблиця 1.2 – Структура таблиці «Agent»

Ім'я поля	Тип	Тип індексу	Припустимість значення Null	Призначення
Key	int(11)	BTREE	Відсутнє	Ідентифікатор контрагента
Ownership	int(11)	BTREE	Відсутнє	Ідентифікатор форми власності
Name	char(30)		Відсутнє	Назва контрагента
Address	char(50)		Відсутнє	Юридична адреса
INN	char(10)		Відсутнє	ІНН
KPP	char(9)		Відсутнє	КПП
Person	int(11)	BTREE	Відсутнє	Ідентифікатор керівника

"Document" – електронна форма складського ордера.

Таблиця 1.3 – Структура таблиці "Document"

Ім'я поля	Тип	Тип індексу	Припустимість значення Null	Призначення
Key	int(11)	BTREE	Відсутнє	Ідентифікатор складського ордера
NumberD	char(15)		Відсутнє	Реєстраційний номер документа
DateD	date		Відсутнє	Дата документа
Agent1	int(11)	BTREE	Відсутнє	Контрагент/склад відпускаючий
Agent2	int(11)	BTREE	Відсутнє	Контрагент/склад приймаючий
Operat	tinyint(1)		Відсутнє	Операція приходу/вирати
Status	int(11)		Відсутнє	Статус документа
Person	int(11)	BTREE	Відсутнє	Ідентифікатор відповідального

3. "DocumentTable" – паблична форма складського ордера.

Таблиця 1.4 – Структура таблиці "DocumentTable"

Ім'я поля	Тип	Тип індексу	Припустимість значення Null	Призначення
Key	int(11)	BTREE	Відсутнє	Ідентифікатор запису табличної частини складського ордера
Document	int(11)	BTREE	Відсутнє	Шапка складського документа
Product	int(11)	BTREE	Відсутнє	Товар
CountD	int(11)		Відсутнє	Кількість

4. "Dogovir" – зберігає список укладених договорів.

Таблиця 1.5 – Структура таблиці "Dogovir"

Ім'я поля	Тип	Тип індексу	Припустимість значення Null	Призначення
Key	int(11)	BTREE	Відсутнє	Ідентифікатор договору
Number	char(100)		Відсутнє	Реєстраційний номер договору
Date	date		Відсутнє	Дата договору
Dogovir	int(11)	BTREE	Відсутнє	Договір
Agent	int(11)	BTREE	Відсутнє	Контрагент
TypePay	int(11)	BTREE	Відсутнє	Форма оплати
PercentPay	decimal(10,0)	BTREE		% передоплати

5. "GroupProduct" – містить список укладених договорів.

Таблиця 1.6 - Структура таблиці "GroupProduct"

Ім'я поля	Тип	Тип індексу	Припустимість значення Null	Призначення
Key	int(11)	BTREE	Ні	Ідентифікатор групи товару
GroupProduct	char(150)		Ні	Назва групи товару

6. "Oborot" – використовується відображення руху товару складі – кількісні обороти.

Таблиця 1.7 - Структура таблиці "Oborot"

Ім'я поля	Тип	Тип індексу	Припустимість значення Null	Призначення
Key	int(11)	BTREE	Ні	Ідентифікатор запису обігу товару
DateO	date		Ні	Дата руху товару
Operat	tinyint(1)		Ні	Операція руху: прихід/витрата
Agent	int(11)	BTREE	Ні	Контрагент/склад
Product	int(11)	BTREE	Ні	Товар
CountO	int(11)	BTREE	Ні	Кількість
Document	int(11)	BTREE	Ні	Виконавчий документ

7. "Ownership" – форми власності організацій.

Таблиця 1.8 - Структура таблиці "Ownership"

Ім'я поля	Тип	Тип індексу	Припустимість значення Null	Призначення
Key	int(11)	BTREE	Hi	Ідентифікатор форми власності
Ownership	char(150)		Hi	Назва форм власності

8. "Payment" – використовується для відображення оплати за договором.

Таблиця 1.9 - Структура таблиці "Payment"

Ім'я поля	Тип	Тип індексу	Припустимість значення Null	Призначення
Key	int(11)	BTREE	Hi	Ідентифікатор платежу
NumberDoc	char(150)		Так	Номер платіжного документа
DateDoc	date		Hi	Дата платежу
Dogovir	int(11)	BTREE	Hi	Ідентифікатор договору

9. "Person". Таблиця містить співробітники організацій/фізосіб.

Таблиця 1.10 - Структура таблиці "Person"

Ім'я поля	Тип	Тип індексу	Припустимість значення Null	Призначення
Key	int(11)	BTREE	Hi	Ідентифікатор співробітника
TabNumber	char(5)		Hi	Табельний номер співробітника
PIB	char(50)		Hi	Прізвище Ім'я По батькові
Vacant	int(11)	BTREE	Hi	Ідентифікатор посади
Telefon	char(250)		Так	Ідентифікатор посади

10. "Product". Містить список товарів на складі.

Таблиця 1.11 - Структура таблиці "Product"

Ім'я поля	Тип	Тип індексу	Припустимість значення Null	Призначення
Key	int(11)	BTREE	Ні	Ідентифікатор товару
Articke	char(15)		Так	Артикул товару
Name	char(50)		Ні	Назва товару
Unit	int(11)	BTREE	Ні	Одиниця вимірювання
GroupProduct	int(11)	BTREE	Ні	Ідентифікатор групи товару

11. "Registration" – зареєстровані користувачі системи.

Таблиця 1.12 - Структура таблиці "Registration"

Ім'я поля	Тип	Тип індексу	Припустимість значення Null	Призначення
Key	int(11)	BTREE	Ні	Ідентифікатор користувача
Login	char(10)		Ні	Логін
Parol	char(10)		Ні	Пароль

12. "Saldo" – зберігає дані про залишки товарів початку кожного місяця року.

Таблиця 1.13 - Структура таблиці "Saldo"

Ім'я поля	Тип	Тип індексу	Припустимість значення Null	Призначення
Key	int(11)	BTREE	Ні	Ідентифікатор запису про залишок
DateS	int(11)		Ні	Дата початку місяця
Agent	int(11)	BTREE	Ні	Контрагент/склад
Product	int(11)	BTREE	Ні	Товар
CountS	int(11)		Ні	Залишок товару

13. "TypePay" – містить список видів оплат.

Таблиця 1.14 - Структура таблиці "TypePay"

Ім'я поля	Тип	Тип індексу	Припустимість значення Null	Призначення
Key	int(11)	BTREE	Ні	Ідентифікатор виду оплати
TypePay	char(1505)		Ні	Назва виду оплати

14. "Unit" – містить одиниці виміру товару.

Таблиця 1.15 - Структура таблиці "Unit"

Ім'я поля	Тип	Тип індексу	Припустимість значення Null	Призначення
Key	int(11)	BTREE	Ні	Ідентифікатор запису про залишок
OKEI	char(5)		Відсутнє	Дата початку місяця
Unit	char(15)		Ні	Товар
UnitShort	char(10)		Ні	Залишок товару

15. "Vacant" – містить список посад.

Таблиця 1.16 - Структура таблиці "Vacant"

Ім'я поля	Тип	Тип індексу	Припустимість значення Null	Призначення
Key	int(11)	BTREE	Ні	Ідентифікатор посади
Vacant	char(150)		Ні	Назва посади

Опис зв'язків таблиць бази даних наведено у таблиці.

Таблиця 1.17 – Зв'язки таблиць бази даних системи

Перша таблиця	Ім'я первинного ключа	Тип зв'язку	Друга таблиця	Ім'я вторинного ключа
"GroupProduct"	Key	Один до багатьох	"Product"	GroupProduct
"Vacant"	Key	Один до багатьох	"Person"	Vacant
"Person"	Key	Один до багатьох	«Document»	Person
"Person"	Key	Один до багатьох	"Agent"	Person
"Ownership"	Key	Один до багатьох	"Agent"	Ownership
"TypePay"	Key	Один до багатьох	"Dogovir"	TypePay
"Dogovir"	Key	Один до багатьох	"Payment"	Dogovir
"Dogovir"	Key	Один до багатьох	«Document»	Dogovir
"Agent"	Key	Один до багатьох	"Dogovir"	Agent
"Agent"	Key	Один до багатьох	«Document»	Agent1
"Agent"	Key	Один до багатьох	«Document»	Agent2
"Agent"	Key	Один до багатьох	"Oborot"	Agent
"Agent"	Key	Один до багатьох	"Saldo"	Agent
"Product"	Key	Один до багатьох	"DocumentTable"	Product
"Product"	Key	Один до багатьох	"Oborot"	Product
"Product"	Key	Один до багатьох	"Saldo"	Product
"Unit"	Key	Один до багатьох	"Product"	Unit
«Document»	Key	Один до багатьох	"DocumentTable"	Document
«Document»	Key	Один до багатьох	"Oborot"	Document

Відобразимо схему БД MySQL web-сервісу «Склад».

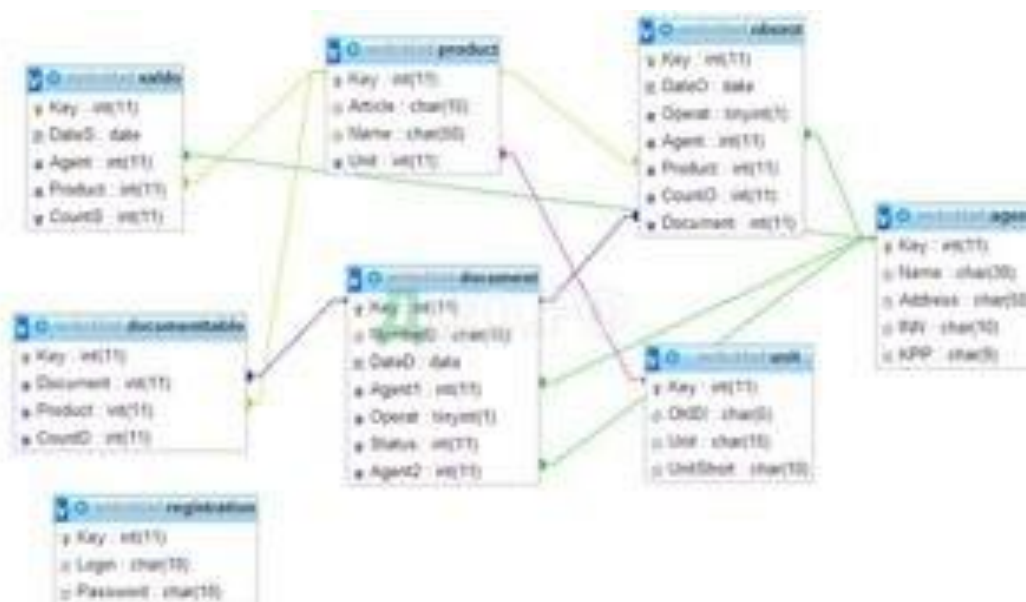


Рисунок 1.3 – Схема бази даних web-сервісу «WebСклад»

1.6 Огляд програмних аналогів

1.6.1. Архітектура платформ Java та .Net.

Java: є кросплатформовим середовищем. Якщо для необхідної платформи існує JDK, то Java буде функціонувати. .NET: повністю розроблена Microsoft і може працювати тільки на операційних системах Microsoft. На сьогоднішній день єдиною платформою є лише Windows. Єдиною мовною основою є мова програмування Java. Платформа є, якщо так можна сказати, мовонезалежною. Якщо для цієї мови програмування створено механізм відображення в MSIL, його можна сміливо застосовувати в розробці. Вихідні тексти Java -програми компілюються в байт-код, який інтерпретується (виконується) віртуально Java-машиною щоразу під час виконання. Для роботи програми Java потрібно поставити на клієнта JRE (Java Runtime Environment) .

Схема роботи .NET-платформи багато в чому схожа. Виконанням програми управляє CLR (аналог JVM). При цьому програма повинна бути попередньо скомпільована в MS IntermediateLanguage - MSIL. На вибір засобів розробки впливають їх поширеність, доступність, простота розгортання, уподобання програміста.

1.6.2. MS Visual Studio.

Це платний продукт встановленню якого передують довга інсталяція. Він займає багато місця на диску. Забезпечує швидке та легке розгортання web-сервісів. специфікацій WSSecurity, WSTrust, WSSecureConversation та WSAddressing. у частині визначення функції. При виборі команди . Project I Build - Visual Studio .NET . компілює надсилає його на web-сервер.

1.6.3. Axis і Apache Tomcat.

Apache Tomcat широко поширений та доступний безкоштовно (<http://tomcat.apache.org/>) і швидко ставиться. Працює на будь-якій платформі, де є Java. Є надійним, має підтримку сучасних стандартів. Для підтримки web-сервісів треба встановити пакет ApacheAxis. Всю установку також можна звести до простої розархівзації дистрибутива та копіювання бібліотек Axis з каталогу lib/ в common/lib/ і каталогу webapps/axis/ в каталог webapps/ webapps/ сервера Tomcat. Після цього він працюватиме ще й як SOAP і WSDL -сервер.

Відмінною особливістю Axis є наявність у ньому утиліт (WSDL2java, java2WSDL), що дозволяють автоматизувати деякі стадії створення веб-сервісу та клієнта до нього, а також велику супровідну документацію та приклади.

Завдання розгортання web-сервісів на будь-якій платформі швидко вирішується за допомогою Apache Axis та Java. Можна легко написати обгортку у вигляді jws-класу для програми та таким чином забезпечити доступ до нього, як до web-сервісу. Залежно від вимог і можливостей організувати звернення до web-сервісів на Java з клієнтських додатків можна різними способами.

1.6.4. Сервери Apache та IIS.

Найпопулярнішим програмним забезпеченням визнано безкоштовне ПЗ Apache, яке займає майже 50% ринку — близько 77 мільйонів сайтів, потім йде продукція компанії Microsoft.

Перший критерій вибору — апаратне забезпечення сервера. Існує два основних рішення - виділений сервер та розміщення на вже існуючому сервері, налаштувавши маршрутизацію трафіку на брандмауері.

IIS може бути встановлений лише на ОС MS Windows. Apache може бути встановлений як на ОС сімейства UNIX, і на MS Windows. Доступність. Apache розповсюджується безкоштовно. MS Internet Information Server розповсюджується за ліцензією.

Знаючи технологію Java і технологію .Net програмісту знадобиться приблизно однакову кількість часу розробки web-сервісу. Але, якщо розробник не знайомий з будь-якої з цих технологій, то, звичайно, краще розробляти в знайомій вже технології. Так як знайомство з новою технологією може зайняти більшу частину всього часу створення web-сервісу. Якщо ж розробник не знайомий з жодною з цих технологій, то на мій погляд, простіше і швидше розібратися і створити web-сервіс використовуючи MS Visual Studio.Net.

На основі написаних web-сервісів видно, що число рядків коду сервісу, написаного засобами MS Visual Studio та web-сервісу, написаного за допомогою Axis, що виконують однакові дії, практично однаково. Простий сервіс, що повертає список даних з однієї таблиці бази даних, займає 29 рядків у MS Visual Studio.Net і 33 рядки, написаний за допомогою Axis.

1.7 Постановка комплексу завдань розробки web-сервіса

Метою кваліфікаційної роботи є розробка web-сервісу для відстеження складських залишків.

Web-сервіс призначений для автоматизації організації та управління рухом товару на складі.

Система забезпечує автоматизацію складського обліку: - ведення даних про постачальників у системі;

- ведення даних про товар;
- прийом товару від постачальника та відвантаження товару покупцю;
- формування даних про наявність товару на складі;
- формування даних про рух товару на складі;
- виписку складських документів: прибуткові та видаткові ордери.

Система дозволяє:

- отримувати щоденні та щомісячні звіти;

- вести щоденний облік стану товарних запасів у кількісному вираженні;
- отримувати довідки про рух та наявність товару, постачальників, покупців.

Діаграма визначає взаємозв'язку процесів у системі обліку товарів з вихідними даними, виконавцями, керуючими командами та результатами виконання.

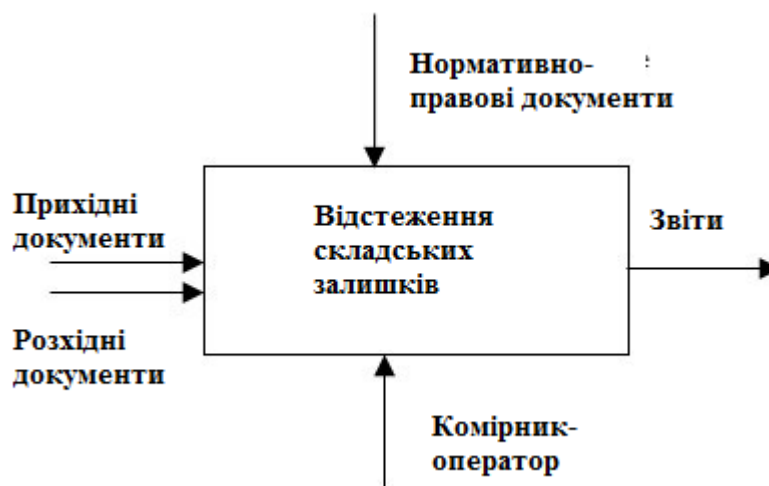


Рисунок 1.2 – Контекстна діаграма

Вихідними даними у діаграмі є:

Прихідні (прибуткові) документи: товарно-транспортні та товарні накладні, рахунки-фактури, прибутковий ордер;

Розхідні (видаткові) документи: товарно-транспортні та товарні накладні, рахунки-фактури, видатковий ордер.

Управляючою інформацією є нормативно-правові документи.

Результати системи – звіти: про наявність товару на складі на дату, рух товару на складі за період.

1.8 Вибір інструментальних програмних засобів розробки

Для розробки сервісу відстеження складських залишків WEBСклад необхідні такі додаткові інструментальні програмні засоби: мова програмування PHP5 та його розширення, локальний сервер Apache, СУБД MySQL.

PHP - це мова серверу для створення загальних сценаріїв, яка використовується для розробки web-додатків. Її використовують більшість хостинг-провайдерів для розробки динамічних web-сайтів. Популярність він завоював завдяки високій швидкості виконання, широкому функціоналу, поширенню свого відкритого коду та простоті. Має підтримку розширень - модулів, що підключаються, для роботи в різних умовах - наприклад, з PDF-документами, криптографічними бібліотеками або динамічною графікою, причому кожен бажаючий може створити своє власне розширення. В даний час мову використовують сотні тисяч розробників, і про роботу з цією мовою повідомляє п'ята частина інтернет-ресурсів. Однією з найважливіших переваг роботи PHP у порівнянні з іншими мовами є можливість впроваджувати команди цієї мови у HTML-документ. Розробкою мови займається безліч однодумців, які добровільно працюють над удосконаленням як ядра мови, так і її розширень, складають документацію та займаються іншими суміжними проектами [10].

MySQL AB - комерційна компанія, заснована розробниками MySQL, будує свій бізнес на представленні послуг, так чи інакше пов'язаних із СУБД MySQL.

СУБД MySQL є структурованим набором даних. Містить різного роду інформацію - від звичайного списку покупок до великого об'єму даних, які використовуються у корпоративних мережах. Також її розглядають як систему курування реляційними БД, де зберігається інформація у табличному виді, що підвищує продуктивність і гнучкість.

Стандарт SQL було прийнято 1986 року і нині існує кілька його версій.

Це система з відкритим кодом. Відкритість вихідного коду означає, що будь-хто має можливість використовувати і модифікувати це ПЗMySQL можна з

Internet, причому абсолютно безкоштовно. Кожен користувач, за бажанням, може вивчити вихідні тексти та змінити їх відповідно до своїх потреб.

Сервер MySQL працює в клієнт-серверних та вбудованих системах. СУБД MySQL є клієнт-серверною системою, що включає багатопотоковий SQL-сервер, що підтримує різні платформи, кілька клієнтських програм та бібліотек, інструменти адміністрування та широкий діапазон програмних інтерфейсів додатків (API-інтерфейсів).

Доступний величезний обсяг MySQL, написаного незалежними розробниками. [11, 22, 25, 26]. Використання локального сервера Apache викликане необхідністю можливості попереднього перегляду та тестування розробленого веб-сайту до його розміщення на хостингу.

Apache HTTP-сервер - це так званий вільний web-сервер, що є кроссплатформенним програмним забезпеченням. Він підтримує наступні ОС: BSD, Microsoft Windows, Linux, Mac OS, BeOS, Novell NetWare. Надійність та гнучкість конфігурації – головні переваги Apache.

Завдяки цьому програмному забезпеченню можна здійснювати підключення зовнішніх модулів, що використовуються для надання даних, створювати повідомлення щодо помилки, за допомогою СУБД автентифікувати користувачів.

Apache має вбудований механізм віртуальних хостів, кожен за однією IP-адресою обслуговується декілька web-проектів (доменних імен). Для кожного передбачено налаштовувати модулі та ядра та зробити обмеження доступу до сайту чи до окремих файлів.

Додаткові модулі Apache підтримують системи розробки і мов програмування, це Python, PHP, Perl тощо. Також, Apache здійснює підтримку FastCGI та CGI, які дають можливість програмування різними мовами, зокрема JS, SH, C, C ++ [12, 23, 26, 28].

Всі перелічені інструментальні засоби мають вільну ліцензію та поширюються безкоштовно, що дозволяє уникнути додаткових витрат на розробку.

Щоб швидко розгорнути програму використаю локальний сервер Denwer, який не вимагає встановлення окремо серверу Apache, СУБД MySQL та адмін для бази даних phpMyAdmin.

Після вивчення теоретичного матеріалу, ми перейшли безпосередньо до проектування та створення Web-сервісу. Насамперед встановимо локальний сервер Denwer.

Після запуску програми установки Denwer на першому етапі потрібно вказати каталог, на який встановлювати локальний сервер. За замовчуванням запропонують встановити Denwer на диск C у папку WebServers. шлях - C: WebServers. Рекомендовано встановлення комплексу у C-каталог. WebServers, Тоді не виникне проблем із встановленням пактів розширень. Натискаєте клавішу Enter.

Далі потрібно ввести ім'я віртуального диска, який буде пов'язаний із щойно зазначеною директорією. Рекомендується погодитися зі стандартним значенням (Z:). Важливо, що диск з таким ім'ям не повинен утримуватися в системі

Потім потрібно вказати, в якому режимі запускати Denwer. За замовчуванням стоїть 1 режим, в якому установка віртуального диска відбувається одразу із завантаженням Windows, завантаження серверів відбувається за натисканням ярлика, як і у 2 режимі, в якому віртуальний диск включається під час запуску сервером та відключення після зупинки Denwer.

Деякі ОС Windows хибно відключають диски та потребують перезавантаження. Тому, після Enter на робочому столі появляться ярлики. Папка denwer містить файли для запуску, перевантаження, зупинки та ін.. У папці home є електронні сторінки та скрипти сайтів, а також файли адміністрування в локальному сервері localhost. У папці tmp знаходяться тимчасові файли. У папці usr знаходяться файли баз даних MySQL, PHP, BIN, файли Apache та інші [13].

1.9. Висновок до розділу 1

В результаті проведеного аналізу ведення складського обліку на підприємствах, що реалізують товари зі складу, можна виділити найчастіше проблеми обліку.

- Багаторазове введення інформації. При оформленні руху товару всередині підприємства бухгалтер і комірник вручну заповнюють уніфіковані форми на товари, що постійно використовуються. Дублює рух товару у складських картках та журналах реєстрації складських документів. Це помітно збільшує трудовитрати персоналу та ймовірність виникнення помилок.

- Уповільнення роботи. Ця проблема тісно пов'язана із попередньою. Через багаторазове ручне заповнення уніфікованих форм (накладні, акти, ордери), відомостей робота компанії уповільнюється. Багато часу витрачається на ручне заповнення документів та відображення руху товару у складських картках.

- Велика ймовірність втрати інформації або помилки. При формуванні звітів (оборотно-сальдові та оборотні відомості, звіт про наявність товарів та інші) співробітникам складів доводиться обробляти великі обсяги даних, що неминуче призводить до помилок, або навіть втрати інформації.

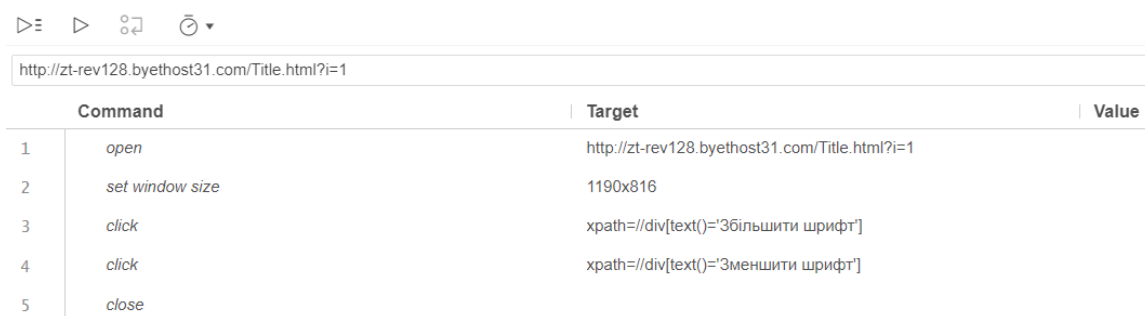
Розділ присвячений розробці системи web-сервісу. Для розробки системи обрано мову програмування PHP5 та його розширення, локальний сервер Apache, систему управління БД MySQL. Розроблено контент та дизайн web-сервісу «WebСклад». Спроектовано три типи форм інтерфейсу системи для введення, редагування та перегляду даних системи. Розроблена логічна структура бази даних MySQL web-сервісу «WebСклад» була описана та реалізована у вигляді таблиць та відносин між таблицями.

2. ТЕСТУВАННЯ СИСТЕМИ

Для забезпечення якості розробленого програмного продукту проведемо автоматизоване тестування. При цьому було створено 7 тест-кейсів, 5 позитивних та 2 негативних [22 , 26 , 27].

Позитивні тест-кейси:

1) Збільшення та зменшення тексту на сторінці через користувацьке меню.

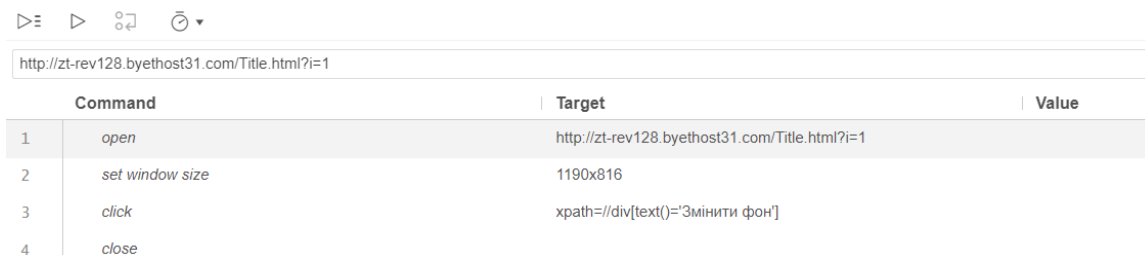


The screenshot shows a Selenium IDE interface with a URL bar containing 'http://zt-rev128.byethost31.com/Title.html?i=1'. Below the URL bar is a table with 3 columns: 'Command', 'Target', and 'Value'. The table contains 5 rows of test steps.

	Command	Target	Value
1	open	http://zt-rev128.byethost31.com/Title.html?i=1	
2	set window size	1190x816	
3	click	xpath=//div[text()='Збільшити шрифт']	
4	click	xpath=//div[text()='Зменшити шрифт']	
5	close		

Рисунок 2.1 – Команди для перевірки дії збільшення і зменшення тексту на сторінці

2) Зміна фону сторінки через користувацьке меню

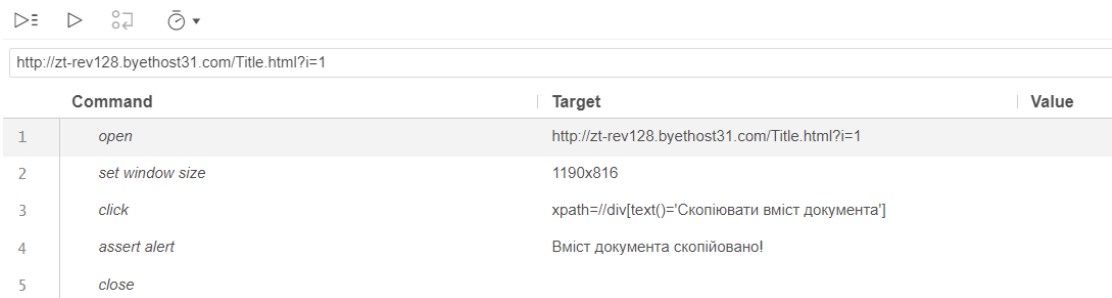


The screenshot shows a Selenium IDE interface with a URL bar containing 'http://zt-rev128.byethost31.com/Title.html?i=1'. Below the URL bar is a table with 3 columns: 'Command', 'Target', and 'Value'. The table contains 4 rows of test steps.

	Command	Target	Value
1	open	http://zt-rev128.byethost31.com/Title.html?i=1	
2	set window size	1190x816	
3	click	xpath=//div[text()='Змінити фон']	
4	close		

Рисунок 2.2 – Команди для перевірки дії зміни фону на сторінці

3) Копіювання вмісту сторінки через користувацьке меню



	Command	Target	Value
1	open	http://zt-rev128.byethost31.com/Title.html?i=1	
2	set window size	1190x816	
3	click	xpath=//div[text()='Скопіювати вміст документа']	
4	assert alert	Вміст документа скопійовано!	
5	close		

Рисунок 2.3 – Команди для перевірки дії копіювання вмісту сторінки

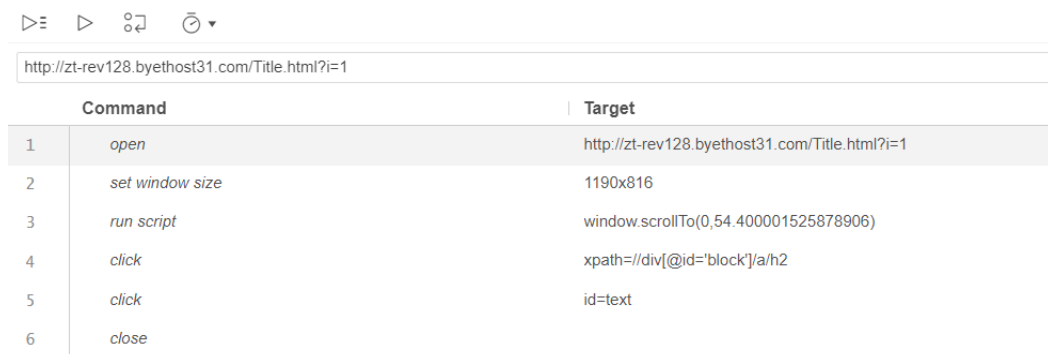
4) Перевірка умов зберігання



	Command	Target
1	open	http://zt-rev128.byethost31.com/Title.html?i=1
2	set window size	1552x832
3	click	css=a:nth-child(5) > img
4	store window handle	root
5	select window	handle=\${win8861}
6	run script	window.scrollTo(0,700)
7	close	
8	select window	handle=\${root}
9	close	

Рисунок 2.4 – Команди для перевірки роботи умов зберігання

5) Перехід на сторінку "Персонал"

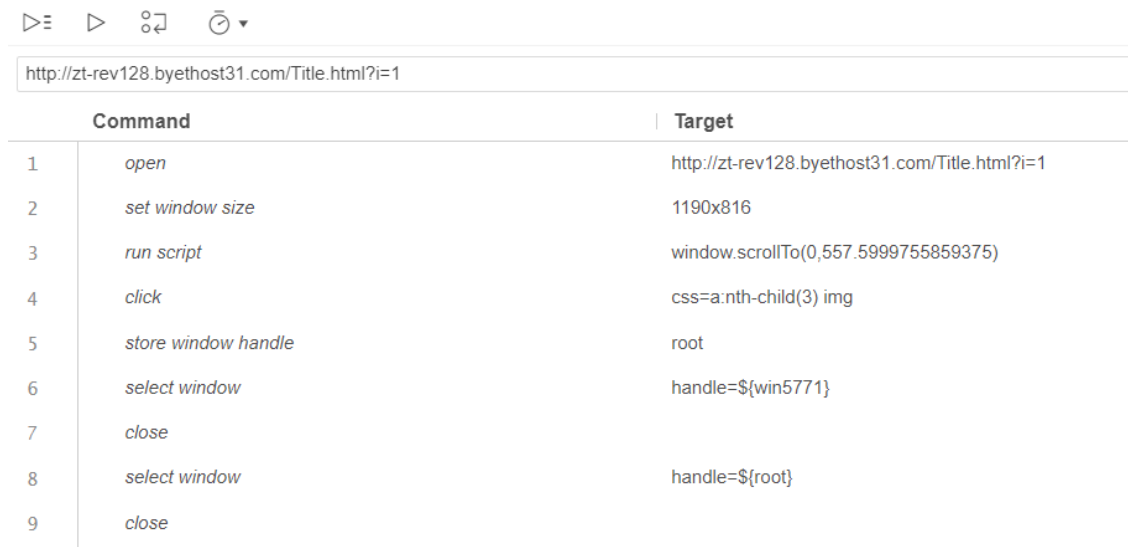


	Command	Target
1	open	http://zt-rev128.byethost31.com/Title.html?i=1
2	set window size	1190x816
3	run script	window.scrollTo(0,54.400001525878906)
4	click	xpath=//div[@id='block']/a/h2
5	click	id=text
6	close	

Рисунок 2.5 – Команди для перевірки переходу на сторінку "Персонал"

Негативні тест-кейси:

1) Перевірка роботи фінансового віджету



	Command	Target
1	open	http://zt-rev128.byethost31.com/Title.html?i=1
2	set window size	1190x816
3	run script	window.scrollTo(0,557.5999755859375)
4	click	css=a:nth-child(3) img
5	store window handle	root
6	select window	handle=\${win5771}
7	close	
8	select window	handle=\${root}
9	close	

Рисунок 2.6 – Команди для перевірки некоректного посилання на сторінку фінансового віджету

Після того, як були створені та описані команди для тест-кейсів, запускаємо перевірку всіх тестів:

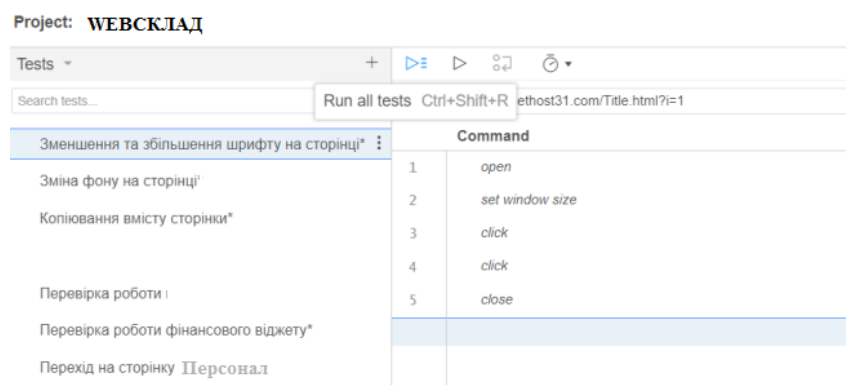


Рисунок 2.7 – Запуск всіх описаних тестів

Після виконання всіх описаних тестів отримуємо їх результати:

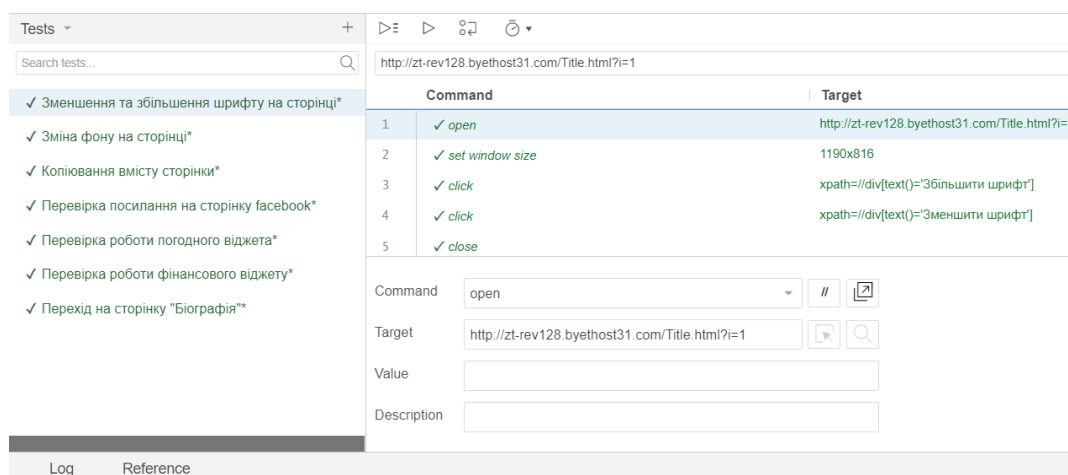


Рисунок 2.8 – Результати тестів

Автоматизоване тестування програмного забезпечення показало свою ефективність у забезпеченні якості та стабільності роботи веб-сайту. Використовуючи Selenium IDE, було створено і виконано декілька тест-кейсів, які перевірили ключові функціональності сайту, такі як зміна розміру шрифту, зміна фону, копіювання вмісту сторінки, перевірка посилань на соціальні мережі, а також робота погодного та фінансового віджетів. Усі тести були успішно виконані, що свідчить про коректну роботу перевічених елементів інтерфейсу.

Результати автоматизованого тестування показали, що веб-сайт відповідає основним функціональним вимогам і працює стабільно під час виконання базових операцій. Автоматизація процесу тестування дозволила значно скоротити час перевірки і підвищити точність виявлення помилок.

3 БЕЗПЕКА ЖИТТЄДІЯЛЬНОСТІ, ОСНОВИ ОХОРОНИ ПРАЦІ

3.1 Ергономічні проблеми безпеки життєдіяльності

При написанні, кваліфікаційної роботи на здобуття освітнього ступеня “бакалавр” я дійшов висновку, що працюючи над проєктами будь-яких систем, дуже важливим є уникнення ергономічних проблем. В ході застосування мого додатку, користувач повинен дотримуватися ергономічних вимог щодо робочого місця, часу та відпочинку. А використання функціоналу цілої системи, а саме: перегляд дописів, додавання фотографій та опису подорожей, їх редагування та видалення, передбачає перебування споживача в одній позиції впродовж тривалого часу, що може погано впливати на його здоров'я та життєдіяльність. Щоб уникнути цих негативних наслідків від перегляду застосунків та користування мобільними пристроями чи ПК, потрібно проаналізувати і вивчити ергономічні проблеми та вимоги безпеки життєдіяльності.

Під ергономікою розуміють галузь знань, що комплексно вивчає трудову діяльність людини у системах людина-машина-середовище (ЛМС) з метою забезпечення її ефективності, безпеки та комфорту[11].

Мета ергономіки - підвищення ефективності системи ЛМС, забезпечення безпеки праці.

Ергономічні проблеми безпеки життєдіяльності – це проблеми, пов'язані з проектуванням та організацією праці та середовища, які впливають на безпеку та здоров'я людей та включають: тривалість і інтервали роботи і відпочинку; організація та планування робочого місця; фізичні і розумові навантаження і стреси.

Продуктивність праці, працездатність людини в багатьох випадках визначаються правильним встановленням режиму праці та відпочинку, що означає зміну періодів праці та відпочинку протягом доби, тижня та довшого

терміну. Реалізація основних ергономічних вимог до режимів праці та відпочинку дає змогу забезпечити необхідний рівень працездатності, зменшити втому, зберегти здоров'я людей. При розробленні режимів праці та відпочинку необхідно встановити [13]:

- тривалість періодів безперервної праці протягом доби (тривалість робочої зміни);
- інтервали між періодами безперервної праці (між змінами);
- кількість змін, які забезпечують чергування;
- тривалість та форму відпочинку.

У системі ЛМС завжди є 3 елементи: предмет праці, засоби праці та суб'єкт праці [11]. Найменшою цільною одиницею, де наявні вказані елементи, є місце праці. Місце праці — це зона, де є необхідні технічні засоби, де відбувається трудова діяльність людини. Місце праці обладнане засобами відображення інформації, органами керування та допоміжним обладнанням.

Організацією місця праці називається проведення системи заходів щодо його обладнання засобами та предметами праці і їх розташуванням у визначеному порядку з метою досягнення:

- оптимізації умов трудової діяльності;
- безпеки праці; – максимальної ефективності;
- комфортності роботи людини.

До робочого місця ставляться такі вимоги:

- достатній робочий простір, який дає змогу працюючій людині здійснювати необхідні рухи та переміщення;
- достатні фізичні, зорові та слухові зв'язки між людиною та обладнанням, а також між людьми

Під час виконання спільного трудового завдання;

- необхідний рівень освітлення;
- допустимий рівень шуму, і вібрацій та інших шкідливих чинників, які генерує обладнання місця праці та інші джерела;
- наявність необхідних засобів захисту;

– оптимальне розташування робочих місць, а також безпечні та достатні проходи для працюючих людей.

Засоби відображення інформації забезпечують своєчасність отримання людиною потрібної інформації для її аналізу, логічної обробки та прийняття погрібного рішення. Використання застосунків вимагає концентрації, аналізу та вирішення складних завдань. Це призводить до психологічного напруження та втоми.

Для зменшення розумових навантажень рекомендується дотримуватися правил раціональної організації робочого процесу, встановлення чітких цілей та пріоритетів, а також надання користувачам необхідних інструментів та підтримки. Високі очікування, термінові завдання та обмежений час створюють стресові ситуації для користувачів. Стрес може негативно впливати на якість роботи та здоров'я. Для зменшення стресу важливо забезпечити раціональне планування роботи, розподіл завдань та належну комунікацію між учасниками проєкту. Звернення уваги на фізичні і розумові навантаження, а також на стресові ситуації під час розробки та використання інтерактивного додатку допоможе забезпечити безпечні умови праці та покращити ефективність та задоволення користувачів. Щоб вирішити ці проблеми, слід скористатися ергономічними принципами та стандартами для оптимізації взаємодії між людьми, машинами та навколишнім середовищем.

Ергономічні рекомендації до роботи за комп'ютером [12]:

- перед увімкненням живлення користувач ПК має переконатися у наявності заземлення, передбаченого інструкцією, а також перевірити справність шнура живлення і шнура зв'язку клавіатури із блоком живлення;
- положення тулуба користувача ПК є таким, щоб його погляд був спрямований прямо на монітор; - нижній край екрана монітора знаходиться на 20 см нижче від рівня очей користувача ПК;
- верхній край екрана монітора є на рівні чола користувача ПК;
- екран монітора розташований на відстані не менше 75—120 см від очей користувача ПК;

- робоче крісло користувача ПК відрегульоване так, щоб кут між його стегнами і тулубом становив 90° ;

- при восьмигодинному робочому дні кількість опрацьованих (шляхом введення даних або їх зчитуванням з екрана монітора) символів (знаків) не повинна перевищувати 30 000 за 4 години роботи.

Надання належної уваги ергономічним аспектам роботи за комп'ютером є важливим завданням для користувача. Інвестиція в налагодження комфортних умов праці та дотримання ергономічних принципів розглядається, як важлива складова частина створення здорового та продуктивного робочого середовища.

3.2 Гігієнічні вимоги до організації та обладнання робочих місць з ВДТ

Гігієнічні вимоги до організації обладнання робочих місць з ВДТ стають необхідністю для забезпечення оптимальних умов праці. Правильно організоване робоче місце забезпечує ефективність та продуктивність працівника, а також допомагає уникнути можливих проблем зі здоров'ям, пов'язаних з тривалим використанням ВДТ. Метою даного розділу є дослідження та опис гігієнічних вимог до організації обладнання робочих місць з ВДТ. Розглянемо такі аспекти, як розташування екрану, ергономічність робочого місця, освітлення, перерви та відпочинок. Дослідження цих вимог допомагає встановити оптимальні умови роботи з ВДТ та забезпечити здоров'я та комфорт працівників.

1. Екран ВДТ має розташовуватися на оптимальній відстані від очей користувача, що становить 600...700 мм, але не ближче ніж за 600 мм з урахуванням розміру літеро-цифрових знаків і символів [14].

Розташування екрана ВДТ повинен забезпечувати зручність зорового спостереження у вертикальній площині під кутом $+30^{\circ}$ до нормальної лінії погляду працюючого. Монітори потрібно розміщувати на висоті рівня очей (висота від підлоги до нижнього краю екрана має становити 95...100 см) на

відстані 60...70 см від оператора (відстань від краю столу – 50...70 см). Кут зору працюючого щодо екрана має дорівнювати 10...20°, але не більше 40°, кут між верхнім краєм монітора і рівнем очей користувача має становити менш як 10°. Найдоцільніше розміщувати екран перпендикулярно до лінії погляду користувача. Кут нахилу екрана по вертикалі має становити 0...30°. З цією метою сучасні монітори комплектують підставкою з поворотним кронштейном, що дає змогу регулювати кут нахилу монітора і горизонтально обертати його навколо вертикальної осі. Висоту екрана від поверхні підлоги регулюють змінюючи висоту робочої поверхні столу. Іноді монітори встановлюють на спеціальні підставки, що уможливорює його переміщення у просторі у вертикальному та горизонтальному напрямках. З метою зменшення напруження очей потрібно, щоб відстань між краями сусідніх точок зображення на моніторі не перевищувала Г. Оптимальний розмір літеро-цифрових знаків – 16...20, складних знаків – 35...40. Оптимальні співвідношення параметрів літер і цифр такі: ширина значка – 0,75 їх висоти, товщина ліній при зворотному контрасті – 1/6-1/8, відстань між знаками 0,25...0,5 висоти значка, між словами – 0,75...1, між рядками – 0,5...1.

2. Приміщення для роботи з ВДТ повинні мати природне та штучне освітлення відповідно до СНиП II-4-79[16]. Природне освітлення має здійснюватися через світлові прорізи, орієнтовані переважно на північ чи північний схід і забезпечувати коефіцієнт природної освітленості (КПО) не нижче ніж 1,5 %. Розраховується КПО за методикою, викладеною в СНиП II-4-79. Штучне освітлення в приміщеннях з робочими місцями, обладнаними ВДТ ЕОМ та ПЕОМ, має здійснюватися системою загального рівномірного освітлення. У виробничих та адміністративно-громадських приміщеннях, у разі переважної роботи з документами, допускається застосування системи комбінованого освітлення. Зазначення освітлення освітленості на поверхні робочого столу в зоні розміщення документів має становити 300 – 500 лк. Якщо ці значення освітленості неможливо забезпечити системою загального освітлення, допускається використовувати місцеве освітлення. При цьому світильники місцевого освітлення слід встановлювати таким чином, щоб не створювати блиск

на поверхні екрана, а освітленість екрана має не перевищувати 300 лк. Джерела світла в разі штучного освітлення мають застосовуватись переважно люмінесцентні лампи типу ЛБ. У разі влаштування відбитого освітлення у виробничих та адміністративно-громадських приміщеннях допускається застосування метало галогенних ламп потужністю 250 Вт. Дозволяється застосування ламп розжарювання у світильниках місцевого освітлення. Система загального освітлення має становити суцільні або переривчасті лінії світильників, розташовані збоку від робочих місць (переважно ліворуч), паралельно лінії зору працюючих. Дозволяється використання світильників таких класів світлорозподілу: – прямого світла – П; – переважно прямого світла – Н; – переважно відбитого світла – В. Для загального освітлення слід застосовувати світильники серії ЛПО 36 із дзеркальними ґратами, укомплектовані високочастотними пускорегулювальними апаратами (ВЧ ПРА). Дозволяється застосовувати світильники цієї серії без ВЧ ПРА тільки в модифікації “Косо світло”. Застосування світильників без розсіювачів та екрануючих ґрат заборонено.

3. При організації праці, що пов'язана з використанням ВДТ ЕОМ і ПЕОМ, для збереження здоров'я працюючих, запобігання професійним захворювання і підтримки працездатності слід передбачити внутрішньо змінні регламентовані перерви для відпочинку [15]. Внутрішньо змінні режими праці і відпочинку мають передбачати додаткові нетривалі перерви в періоди, що передують появі об'єктивних і суб'єктивних ознак стомлення і зниження працездатності. При виконанні протягом дня робіт, що належать до різних видів трудової діяльності, за основну роботу з ВДТ ЕОМ і ПЕОМ слід вважати таку, що займає не менше 50 % часу впродовж робочої зміни мають передбачатися: – перерви для відпочинку і вживання їжі (обідні перерви); – перерви для відпочинку і особистих потреб (згідно з трудовими нормами); – додаткові перерви, що вводяться для окремих професій з урахуванням особливостей трудової діяльності. Тривалість обідньої перерви визначається чинним законодавством про працю і Правилами внутрішнього трудового розпорядку підприємства (Організації, установи).

ВИСНОВОК

Внаслідок написання роботи мною було розроблено web-сервіс відстеження складських залишків «WebСклад».

Для виконання були потрібні такі інструментальні програмні засоби:

- Мова програмування PHP5 та його розширення,
- локальний сервер Apache,
- Система управління базами даних MySQL.

Система автоматизації складу дає підприємству низку переваг, які дозволять суттєво скоротити собівартість складських операцій та підвищити ефективність складу:

- контроль залишків на складі та нових надходжень;
- виключення можливості подвійного введення даних, і як наслідок скорочення помилок;
- повне та швидке отримання інформації про нинішній стан бізнесу;
- зменшення часу обробки та аналізу даних;
- наявність повної та достовірної інформаційної моделі компанії, яка дозволяє швидко визначати проблеми організації обліку та вживати певних заходів їх ліквідації;
- зменшення помилок введення і зниження ролі людського фактора;
- підвищення швидкості роботи;
- здатність обмежувати доступ персоналу до зміни або перегляду певних даних.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Automation Will Enhance Worker Performance Rather Than Replace Workers: Zebra Study. (дата звернення: 09.06.2024). <https://www.scrabbl.com/automation-will-enhance-worker-performance-ratherthan-replace-workers-zebra-study/amp/>
2. Warehouse automation: technology adoption 2019-2030 | Statista. (дата звернення: 09.06.2024). Statista. <https://www.statista.com/statistics/1169394/global-warehouseautomation-technology-adoption/>
3. Warehouse Rental Costs Expected to Increase by 5% in 2022 – CARGONOW. (дата звернення: 09.06.2024). CARGONOW – Connecting the Logistics Eco-System. <https://cargonow.world/warehouse-rental-costs-expected-to-increase-by-5-in-2022/>
4. Новый робот від Amazon. Компанія прагне повної автоматизації? | Haski.ua. (дата звернення: 09.06.2024). Haski.ua. <https://haski.ua/blog/novyy-robot-otamazon-mozhet-zamenit-lyudey>
5. Walmart to acquire ASRS maker Alert Innovation. (дата звернення: 09.06.2024). The Robot Report. <https://www.therobotreport.com/walmart-toacquire-asrs-maker-alert-innovation/>
6. Smart Warehouses: How IoT Improves Warehouse Management — Digiteum. (дата звернення: 09.06.2024). Digiteum. <https://www.digiteum.com/internet-ofthings-for-smart-warehouses/>
7. IoT in Warehouse Management: 6 Ways To Transform Operations. (дата звернення: 09.06.2024). Hopstack | Fulfillment Automation and Visibility for Warehouse Management. <https://www.hopstack.io/blog/iot-can-transformwarehouse-Operations>
8. Michaelis, J. (дата звернення: 09.06.2024). Warehouse Insights – The planning and optimization application for modern warehouses. All Blog Posts | SAP Community. <https://blogs.sap.com/2020/04/16/warehouse-insights-the-planningand-optimization-application-for-modern-warehouses/>

9. Manhattan Active® Solutions Overview | Manhattan Associates. (дата звернення: 09.06.2024). Omnichannel Fulfillment and Supply Chain Leader |Manhattan Associates. <https://www.manh.com/active>

10. Warehouse Management | Oracle. (дата звернення: 09.06.2024). Oracle Cloud Applications and Cloud Platform. <https://www.oracle.com/scm/logistics/warehouse-management/>

11. Taylor, J. (дата звернення: 09.06.2024). RFID & Warehouse Management: Pros& Cons To Consider. Omnichannel Fulfillment Software for 3PLs & Brands |Extensiv | Extensiv. <https://www.extensiv.com/blog/rfid-warehouse-management>

13. How to setup a Raspberry Pi RFID RC522 Chip. (дата звернення: 09.06.2024). Pi My Life Up. <https://pimylifeup.com/raspberry-pi-rfid-rc522/>

14. ESP32 Wi-Fi & Bluetooth MCU I Espressif Systems. (дата звернення: 09.06.2024). Wi-Fi & Bluetooth MCUs and AIoT Solutions I Espressif Systems. <https://www.espressif.com/en/products/socs/esp32>

15. Raspberry Pi Documentation. (дата звернення: 09.06.2024). Raspberry Pi. <https://www.raspberrypi.com/documentation/>

18. DevDocs – JavaScript documentation. (дата звернення: 09.06.2024). DevDocs API Documentation. <https://devdocs.io/javascript/>

19. Documentation | Node.js. (дата звернення: 09.06.2024). Node.js. <https://nodejs.org/en/docs>

20. Angular. (дата звернення: 09.06.2024). Angular. <https://angular.io/docs>

21. PostgreSQL: Documentation. (дата звернення: 09.06.2024). PostgreSQL: The world's most advanced open source database. <https://www.postgresql.org/docs/>

22. Методичні вказівки до виконання дипломної роботи освітнього рівня —бакалавр студентами усіх форм навчання для напряму підготовки 121 – Інженерія програмного забезпечення / Укладачі : Петрик М.Р., Михалик Д.М., Кінах Я.І., Гладь С.В., Цуприк Г.Б. – Тернопіль : Вид-во ТНТУ імені Івана Пулюя, 2016 – 28 с.

23. М.Р. Петрик, Д.М. Михалик, О.Ю. Петрик, Г.Б. Цуприк. Методичні вказівки до виконання атестаційної роботи магістра за спеціальністю 121 –

“Інженерія програмного забезпечення” для усіх форм навчання [Текст] – Тернопіль : Тернопільський національний технічний університет імені Івана Пулюя – 2020 – 27 с.

25. Бойко І.В., М.Р. Петрик, Г.Б. Цуприк. Інформаційні технології видобутку даних (Data mining, високопродуктивні обчислення у складних системах): навчальний посібник. Тернопіль: : ТНТУ 2020 – 62 с.

26. Бойко І.В., М.Р. Петрик, Г.Б. Цуприк. Моделювання та видобуток даних (високопродуктивні обчислення у великих алгебраїчних та числових системах, комбінаторному аналізі): навчальний посібник. Тернопіль: : ТНТУ 2019 – 62 с.

27. Нога В.В., Цуприк Г.Б. Інформаційні технології при створенні автоматизованих систем для задач збору та збереження інформації. Збірник тез доповідей VII Міжнародної науково-технічної конференції молодих учених та студентів. Актуальні задачі сучасних технологій – Тернопіль 28-29 листопада 2018. — Т. : ТНТУ, 2018. — Том 2. — С. 131.

28. І.В. Бойко, М.Р. Петрик, Г.Б. Цуприк. Дискретні структури (Алгебраїчні та числові системи, комбінаторний аналіз). Навчально-методичний посібник для студентів спеціальності 121 «Інженерія програмного забезпечення», аспірантів та викладачів вищих навчальних закладів. Конспект лекцій : Навчальний посібник / І.В. Бойко, М.Р. Петрик, Г.Б. Цуприк – Тернопіль : ТНТУ , 2017. – 64 с.