

РЕФЕРАТ

Пояснювальна записка до дипломного проекту: 88 с., 14 рис., 10 табл., 20 джерел, 17 електронних плакатів.

Об'єкт дослідження: комп'ютерна інформаційна система "Складський термінал".

Мета роботи: вибір апаратних та розроблення програмних засобів комп'ютерної системи автоматизації складського терміналу регіонального представництва ТОВ "Марс Беверіджиз".

У дипломному проекті розроблений алгоритм роботи інформаційної системи в цілому та окремих її підсистем, розроблена база даних інформаційної системи з теоретичним обґрунтуванням на основі правил ER – декомпозиції.

Для нормального функціонування системи необхідна наявність наступного набору програмних засобів: операційна система Windows 98/XP; драйвера БД, Microsoft Access.

БАЗА ДАНИХ, ТАБЛИЦЯ, ЗАПИТ, ПРОГРАМА

Умови одержання дипломного проекту

93400 м. Сєвєродонецьк, пр.Центральний 59«А», СНУ ім. В.Даля

ЗМІСТ

ПЕРЕЛІК СКОРОЧЕНЬ ТА УМОВНИХ ПОЗНАК	6
ВСТУП	7
1 АНАЛІЗ ПРЕДМЕТНОЇ ГАЛУЗІ І ПОСТАНОВКА ЗАДАЧІ	9
1.1 Територіально-адміністративна структура підприємства	9
1.2 Організація існуючої системи управління підприємством.....	13
1.2.1 Топологія існуючої корпоративної комп'ютерної мережі	13
1.2.2. Логічна організація корпоративної комп'ютерної мережі.....	16
1.3 Постановка технічного завдання.....	19
1.3.1 Найменування, галузь застосування та мета створення системи .	19
1.3.2 Задачі, що мають вирішуватися за допомогою системи.....	21
1.3.3 Вимоги до захисту інформації у комп'ютерній мережі підрозділу	22
1.3.4 Характеристика об'єктів обробки.....	22
1.3.5 Функціональні вимоги до системи.....	23
1.3.6 Характеристика процесів обробки	25
1.3.7 Вихідна інформація ІС	26
1.3.8 Вимоги до інформаційної системи.....	29
2 ВИБІР ПРОГРАМНО-АПАРАТНИХ ЗАСОБІВ РОЗРОБЛЕННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ	31
2.1 Розроблення функціональної структури інформаційної системи складського терміналу	31
2.2 Вибір програмних засобів для розроблення інформаційної системи.	42
2.2.1 СУБД Microsoft Access.....	43
2.2.2 СУБД Interbase	56
2.2.3 Обґрунтування вибору програмного забезпечення.....	60
2.3 Вибір апаратного забезпечення	60
3 ПРОЕКТУВАННЯ "ІНФОРМАЦІЙНОЇ СИСТЕМИ КОМПЛЕКСУ СКЛАДСЬКИХ ТЕРМІНАЛІВ"	65
3.1 Розроблення алгоритму функціонування	65
3.2 Інформаційна структура бази даних	68
4 ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ .	77

4.1 Аналіз потенційних небезпечних і шкідливих виробничих чинників проектованого об'єкту, що мають вплив на персонал	77
4.2 Заходи щодо техніки безпеки	79
4.3 Рекомендації по пожежній безпеці	82
ВИСНОВКИ.....	86
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ.....	87

ПЕРЕЛІК СКОРОЧЕНЬ ТА УМОВНИХ ПОЗНАК

ОС – операційна система;

ПЕОМ – персональна електронна обчислювальна машина;

ПЗ – програмне забезпечення;

БД – база даних;

СУБД – система управління базами даних;

НМА – нематеріальні активи;

ІС – інформаційна система.

ВСТУП

Прогрес, досягнутий за останні кілька років у всіх аспектах обчислювальної техніки, включаючи теорію, технологію і додатка, привели до значного розширення області застосування комп'ютерів і росту числа їх користувачів. Істотною частиною сучасного суспільства є різноманітні системи доступу і зберігання інформації, які є невід'ємної складової сучасного науково-технічного прогресу. Існує багато вагомих причин перекладу існуючої інформації на комп'ютерну основу, тому що більш швидка обробка даних і централізація їх зберігання з використанням клієнт/серверних технологій дозволяють зберегти значні засоби, а головне і час для одержання необхідної інформації. Також значно спрощується доступ до більших обсягів інформації та ведення баз даних.

У будь-якій організації, як великий, так і маленької, виникає проблема такої організації керування даними, яка забезпечила б найбільш ефективну роботу. Деякі організації використовують для цього шафи з папками, але більшість віддають перевагу комп'ютеризованим СУБД, що дозволяють ефективно зберігати, добувати інформацію і управляти більшими обсягами даних. Сучасні СУБД - багатокористувацькі системи керування базою даних, які спеціалізуються на керуванні масивом інформації, одним або безліччю одночасно працюючих користувачів.

Системи керування базами даних становлять у цей час основу комп'ютерного забезпечення інформаційних процесів, що входять практично в усі сфери людської діяльності.

Процес створення повнофункціональної системи керування базами даних, як правило, містить у собі наступні етапи:

- визначення завдань, виконуваних створюваної СУБД;
- розробка;
- створення запитів;

- побудова форм для введення/виводу даних і перегляду інформації, що зберігаються в таблицях і запитах;
- створення необхідних звітів.

Метою дипломного проекту є розробка інформаційної системи обліку складського терміналу регіонального торговельного підприємства ТОВ "Марс Беверіджиз".

1 АНАЛІЗ ПРЕДМЕТНОЇ ГАЛУЗІ І ПОСТАНОВКА ЗАДАЧІ

1.1 Територіально-адміністративна структура підприємства

Компанія ТОВ "Марс Беверіджиз" перебуває на українському ринку близько десяти років і має на даний момент величезні площі для виробництва та розповсюдженню продуктів харчування. Номенклатура вироблених товарів становить кілька сотень найменувань і постійно збільшується. Підприємства компанії розташовані в різних регіонах країни і працюють незалежно друг від друга під керівництвом головного офісу в м. Луганськ. Розподіл готової продукції здійснюється через регіональні офіси компанії, які розташовуються в різних регіонах країни. Безпосереднє просування товарів від виробництва до покупця здійснюється логістичною компанією "Торгівельний дім "Марс". Даний посередник здійснює транспортування готової продукції з виробництва по регіонах, її складування, зберігання і відвантаження клієнтам. Використання посередника в цьому випадку представляє певні вигоди. Різко зменшується число контактів виробника із імовірними покупцями товару (рис. 1.1).

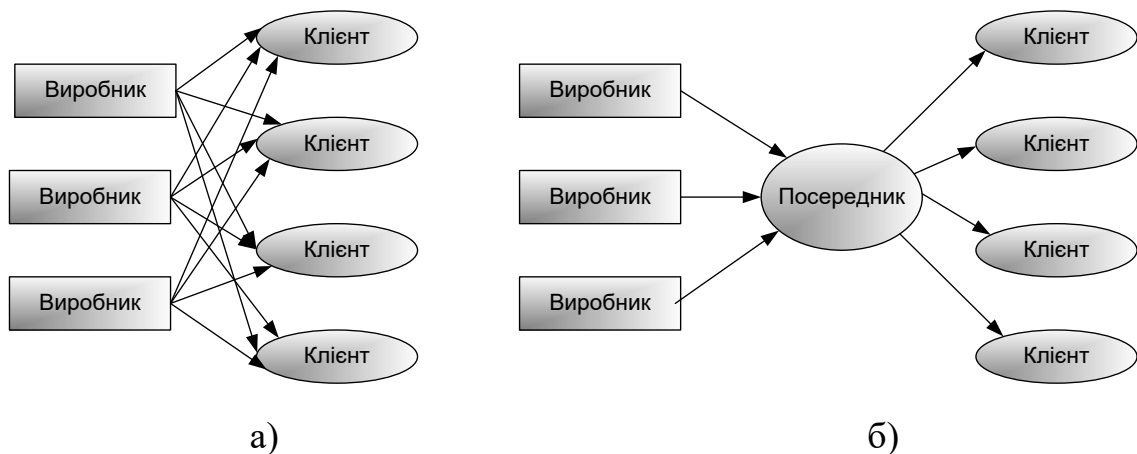


Рисунок 1.1 – Схема організації руху товарів: а) індивідуальна, б) з посередником

Процес одержання товару клієнтом наступний: Клієнт приїжджає в регіональний офіс компанії «Марс Беверіджиз», оплачує необхідні товари і після оформлення відповідних документів відправляється на склад посередника, де по оформлених документах безпосередньо проводиться відвантаження продукції клієнтові.

У цей час у компанії «Марс Беверіджиз» існує розгорнута мережа виробників. Компанія є власником декількох фабрик виробляючих товари народного споживання. Фабрики перебувають у наступних містах (рис.1.2):

- м. Біловодськ;
- м. Кремінне;
- м. Сватове;
- м. Лисичанськ.



Рисунок 1.2 – Територіальне розташування філій ТОВ "Марс Беверіджиз"

Основний офіс компанії розташовується в місті Сєвєродонецьк. Збут продукції здійснюється через регіональні офіси компанії. Регіональні офіси компанії перебувають у містах:

- м. Сватове;
- м. Сєвєродонецьк⁴
- м. Троїцьке;
- м. Старобільськ;
- м. Лисичанськ.

Територіальне розташування фабрик компанії, а також регіональних офісів показано на рис. 1.2. У зв'язку з тим, що підприємства фірми розміщені в різних районах, що і спеціалізуються на випуску одного або декількох видів продукції використовується децентралізована форма керування. При варіанті децентралізованого керування типова схема її організації передбачає окрему відповідальність за постачання виробництва і збут на кожному підприємстві фірми, що спеціалізується на випуску певних видів продукції. Ця схема припускає концентрацію функцій матеріально-технічного забезпечення в рамках однієї служби кожного підприємства, відповідального за той або інший вид продукції.

На рис. 1.3. представлена структурна схема децентралізованого керування виробництвом і матеріально-технічним забезпеченням. Система складається із трьох рівнів. У безпосередньому підпорядкуванні вищого керівництва фірми перебувають чотири підрозділи:, що відають фінансовими активами фірми і цінними паперами, юридичною службою, кадровими питаннями і зв'язками із громадськістю (вони не показані на рис. 1.3.).

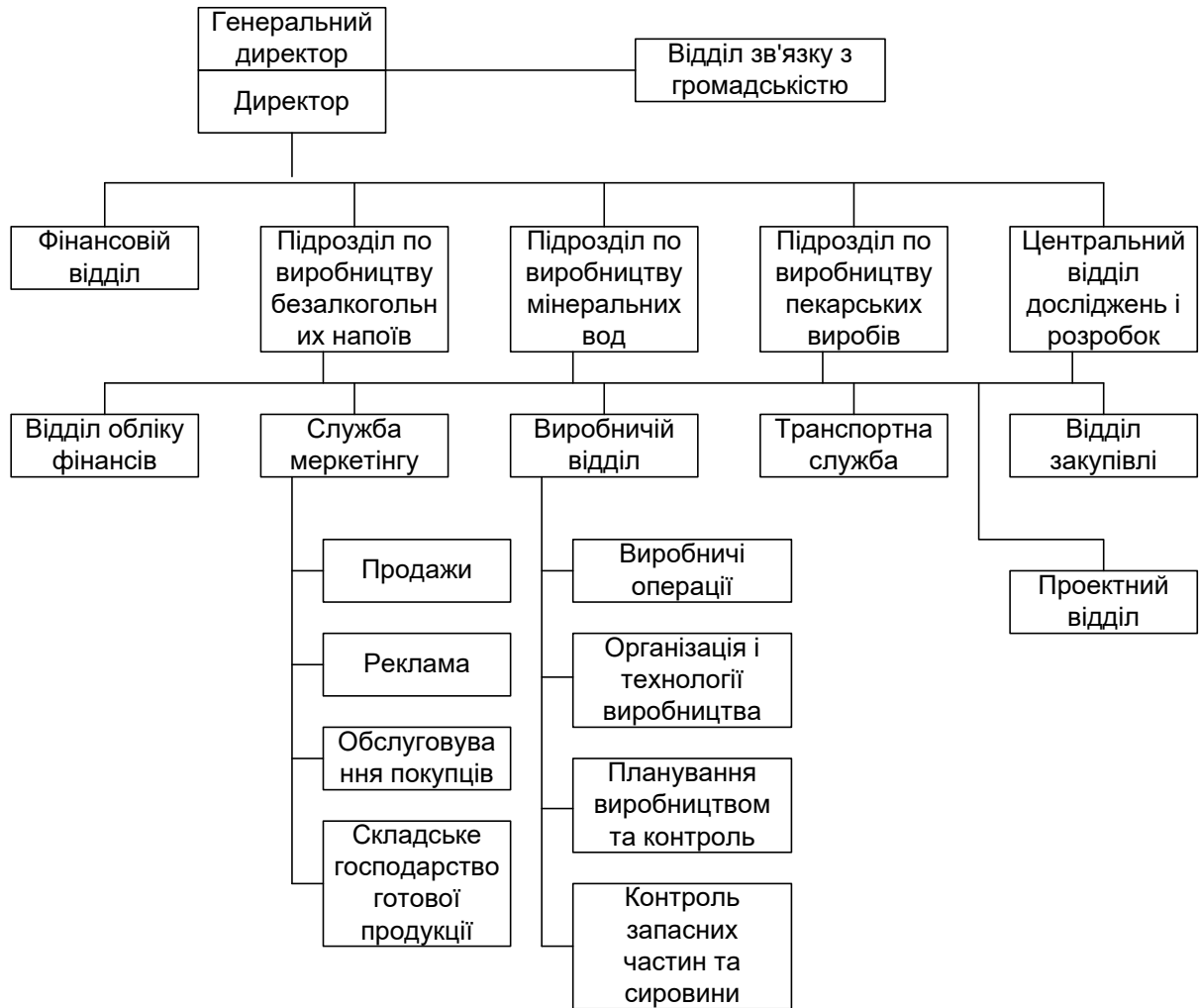


Рисунок 1.3 – Структурна схема децентралізованого керування матеріально-технічним забезпеченням при розподіленій організації підрозділів

Ці підрозділи відображають функції, покладені на апарат керівництва. Другий рівень керування становлять функціональні підрозділи, що безпосередньо організують питання фінансування виробництва, дослідницькі і дослідно-конструкторські розробки, а також увесь комплекс питань, пов'язаних з випуском окремих видів продукції. Третя ланка керування перебуває в безпосередньому підпорядкуванні цих підрозділів і відповідає за дослідження ринку і збутову політику (маркетинг); воно пов'язане з підрозділом, що займається контролем фінансової діяльності і статичним обліком. У функції підрозділу матеріально-технічного забезпечення входить увесь технологічний

ланцюжок, включаючи транспортування і складування матеріалів, а також обслуговування покупців готової продукції. Лінійна служба підрозділу, що відає контролем над запасами, здійснює функції планування виробництва спільно зі службами виробничого підрозділу. При цьому вона походить із наявності і номенклатури запасів на складі фірми.

1.2 Організація існуючої системи управління підприємством

1.2.1 Топологія існуючої корпоративної комп'ютерної мережі

Термін "топологія мережі" відноситься до шляху, по якому дані переміщаються по мережі. Існують три основні види топологій: "загальна шина", "зірка" і "кільце". ЛОМ підприємства побудована по топології «зірка» рис.1.4. У якості середовища передачі сигналів використовується кручена пари (Twisted Pair) п'ятої категорії, що дозволяє передавати інформацію на швидкості до 100 Мбіт/с. Мережа побудована по самій популярній технології Ethernet, і являє собою архітектуру мереж з поділюваним середовищем і широкомовною передачею. Це означає, що всі вузли сегмента мережі одержують пакет одночасно.

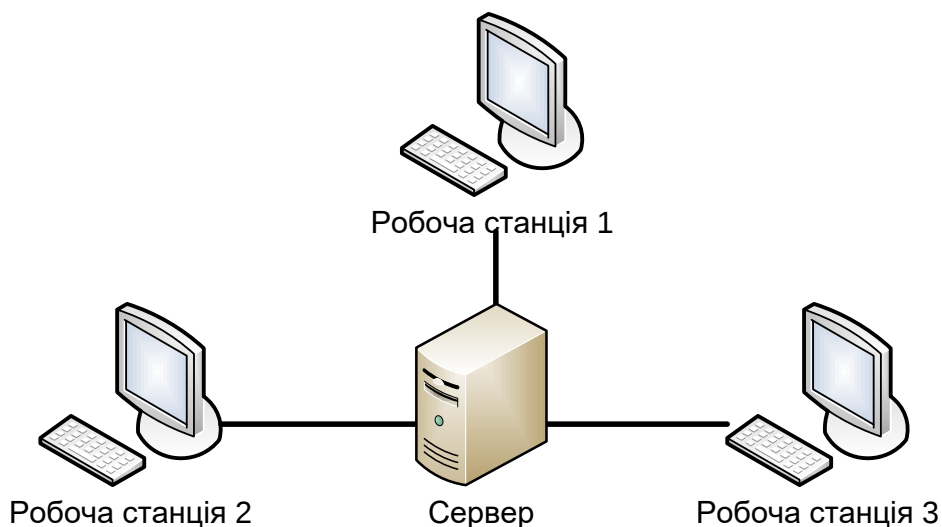


Рисунок 1.4 – Мережна топологія типу "зірка"

У класичному варіанті архітектури із шинною топологією використовується метод множинного доступу із прослуховуванням несучої і виявленням колізій – CSMA/CD (Carrier Sense Multiple Access/Collision Detect). Суть цього методу полягає в тому, що будь-який абонент може намагатися одержати доступ до середовища (почати передачу пакета) у будь-який момент часу, але буде робити це обачно. Якщо в процесі передачі передавальний вузол виявляє колізію (зіткнення з роботою іншого передавача), то він припинить передачу, і буде вичікувати випадковий інтервал часу до поновлення спроби передачі. Такий метод доступу належить до класу недетермінованих з децентралізованим керуванням (усі вузли рівноправні). Як було сказано вище, для реалізації Ethernet на крученому парі застосовується зіркоподібна фізична топологія. Логічно всі вузли виявляються об'єднаними в шину [3]. На рис. 1.5. представлена схема інформаційних потоків у розглянутому підрозділі.

Найбільше докладно показані інформаційні потоки в межах одного регіонального представництва у м. Сєвєродонецьк. В інших представництвах картина потоків інформації аналогічна.

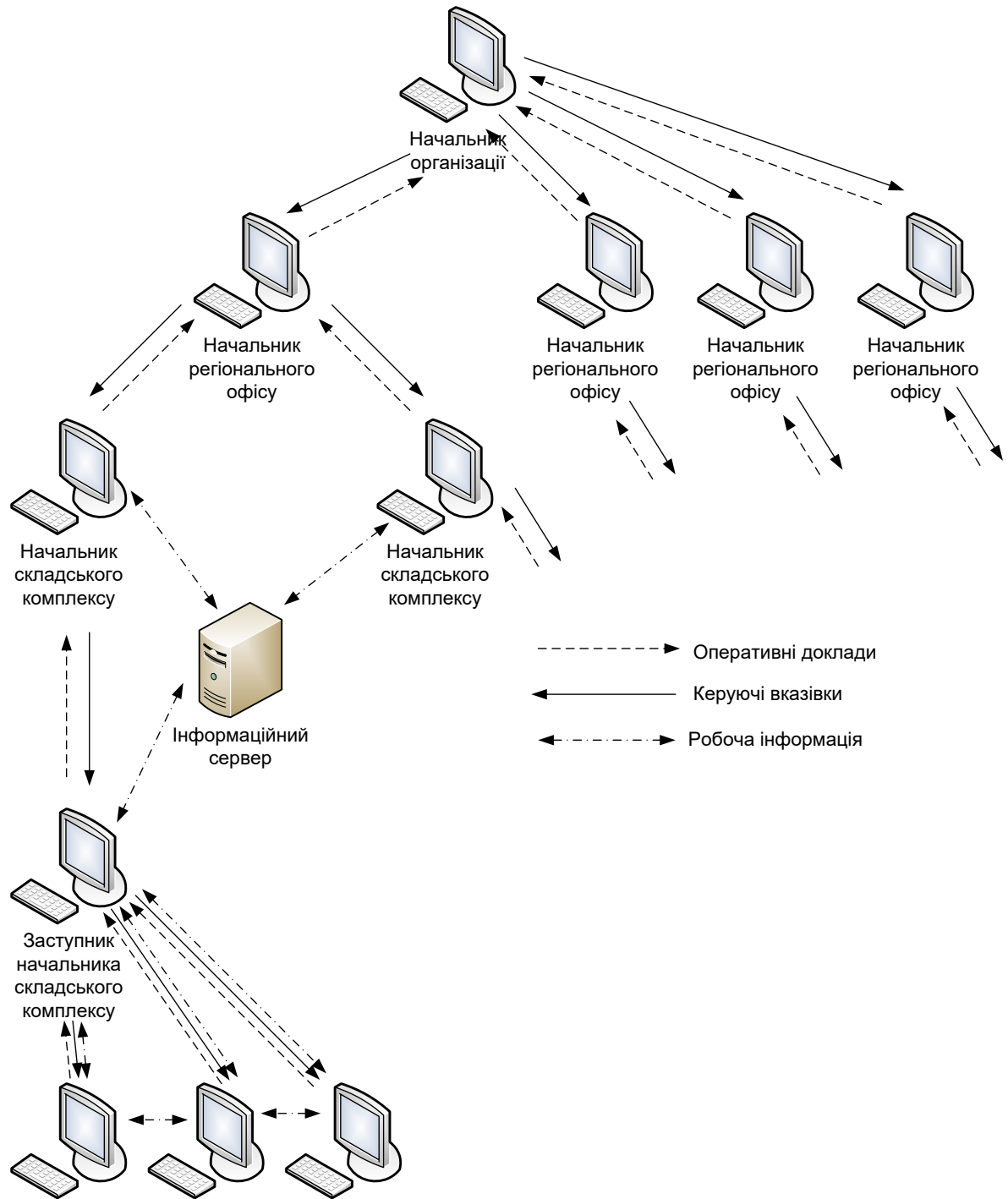


Рисунок 1.5 – Схема інформаційних потоків в комп'ютерній мережі ТОВ "Марс Беверіджиз"

1.2.2. Логічна організація корпоративної комп'ютерної мережі

Основним елементом централізованого адміністрування в Windows NT Server є домен. Домен - це група серверів, що працюють під керуванням Windows NT Server, яка функціонує, як одна система. Усі сервери Windows NT у домені використовують той самий набір облікових карток користувача, тому досить заповнити облікову картку користувача тільки на одному сервері домену, щоб вона розпізнавалася усіма серверами цього домену. Групування комп'ютерів у домени дає дві важливі переваги мережним адміністраторам і користувачам. Найбільш важливе - сервери домену складають (формують) єдиний адміністративний блок, що спільно використовує службу безпеки і інформацію облікових карток користувача. Кожний домен має одну базу даних, що містить облікові картки користувача і груп, а також настановні параметри політики безпеки. Усі сервери домену функціонують або як первинний контролер домену, або як резервний контролер домену, що містить копію цієї бази даних. Це означає, що адміністраторам потрібно управляти тільки однією обліковою картою для кожного користувача, і кожний користувач повинен використовувати (і пам'ятати) пароль тільки однієї облікової картки. Розширюючи адміністративний блок з єдиного комп'ютера на цілий домен, Windows NT Server зберігає зусилля адміністраторів і час користувачів. Друга перевага доменів зроблена для зручності користувачів: коли користувачі переглядають мережу в пошуках доступних ресурсів, вони бачать мережу, згруповану в домени, а не розкидані по всій мережі сервери і принтери. Мінімальна вимога для домену - один сервер, що працює під керуванням Windows NT Server, який служить як первинний контролер домену і зберігає оригінал бази даних облікових карток користувача й груп домену. На додаток до сказаного, домен може також мати інші сервери, що працюють під керуванням

Windows NT Server і службовці як резервні контролери домену, а також комп'ютери, що служать у якості стандартних серверів, серверів LAN Manager 2.x, клієнтів Windows NT Workstation і інших клієнтів, як наприклад, що працюють із MS-DOS (рис.1.6.).

Первинний контролер домену повинен бути сервером, що працюють під керуванням Windows NT Server. Усі зміни бази даних, облікових карток користувача і груп домену повинні виконуватися в базі даних первинного контролера домену. Резервні контролери домену, що працюють під керуванням Windows NT Server, зберігають копію бази даних облікових карток домену. База даних облікових карток копіюється в усі резервні контролери домену. Усі резервні контролери домену доповнюють первинний контролер і можуть обробляти запити на початки сеансу від користувачів облікових карток домену. Якщо домен одержує запит на початок сеансу, первинний контролер домену або кожної з резервних контролерів домену може ідентифікувати спробу початку сеансу.

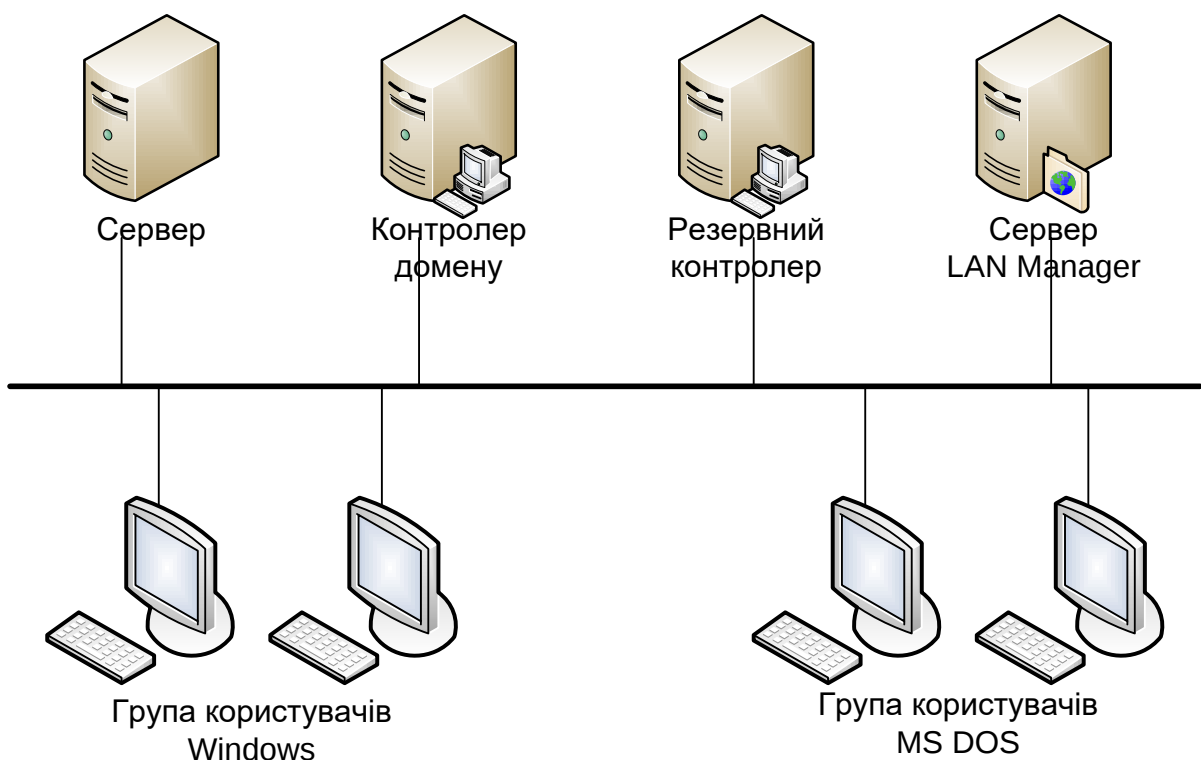


Рисунок 1.6 – Структура домену

Додатково до первинних і резервних контролерів домену, що працюють під керуванням Windows NT Server, є інший тип серверів. Під час установки Windows NT вони визначаються, як “сервери”, а не контролери домену. Сервер, який входить у домен, не одержує копію бази даних користувачів домену. Для організації мережі існує чотири моделі: модель єдиного домену, модель основного домену, модель численних основних доменів і модель повної довіри. Для побудови мережі підприємства використана модель основного домену. Ця модель дає централізоване керування і організаційні переваги керування багатьма доменами. У цій моделі один домен - основний домен, у якому реєструються всі користувачі і глобальні групи. Усі інші домени мережі довіряють цьому домену і у такий спосіб можна використовувати користувачів і глобальні групи, зареєстровані в них. Основна мета головного домену - керування мережними обліковими картками користувача. Інші домени в мережі - домени ресурсу; вони не зберігають облікові картки користувача і не управляють ними, а тільки забезпечують ресурси (як наприклад, файли і принтери колективного використання) мережі. У цій моделі тільки первинні і резервні контролери домену в основному домені мають копії облікових карток користувачів мережі [4]. Логічна структура мережі показана на рис. 1.7.

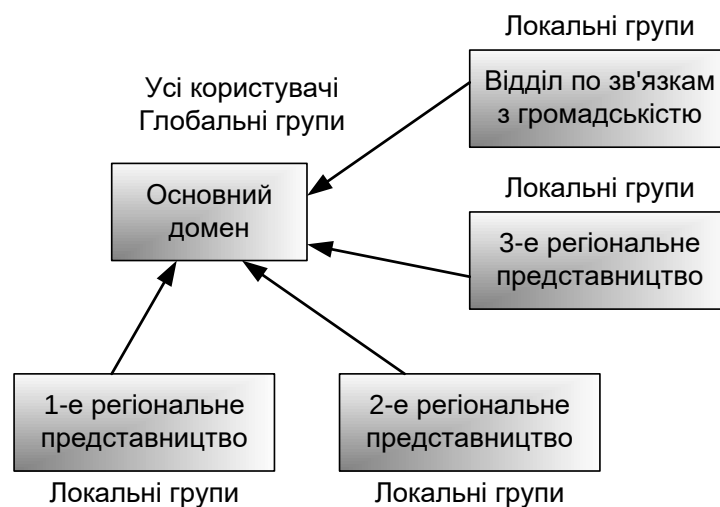


Рисунок 1.7 – Логічна структура мережі

Функціональна схема підрозділу наведена на рис.1.8.

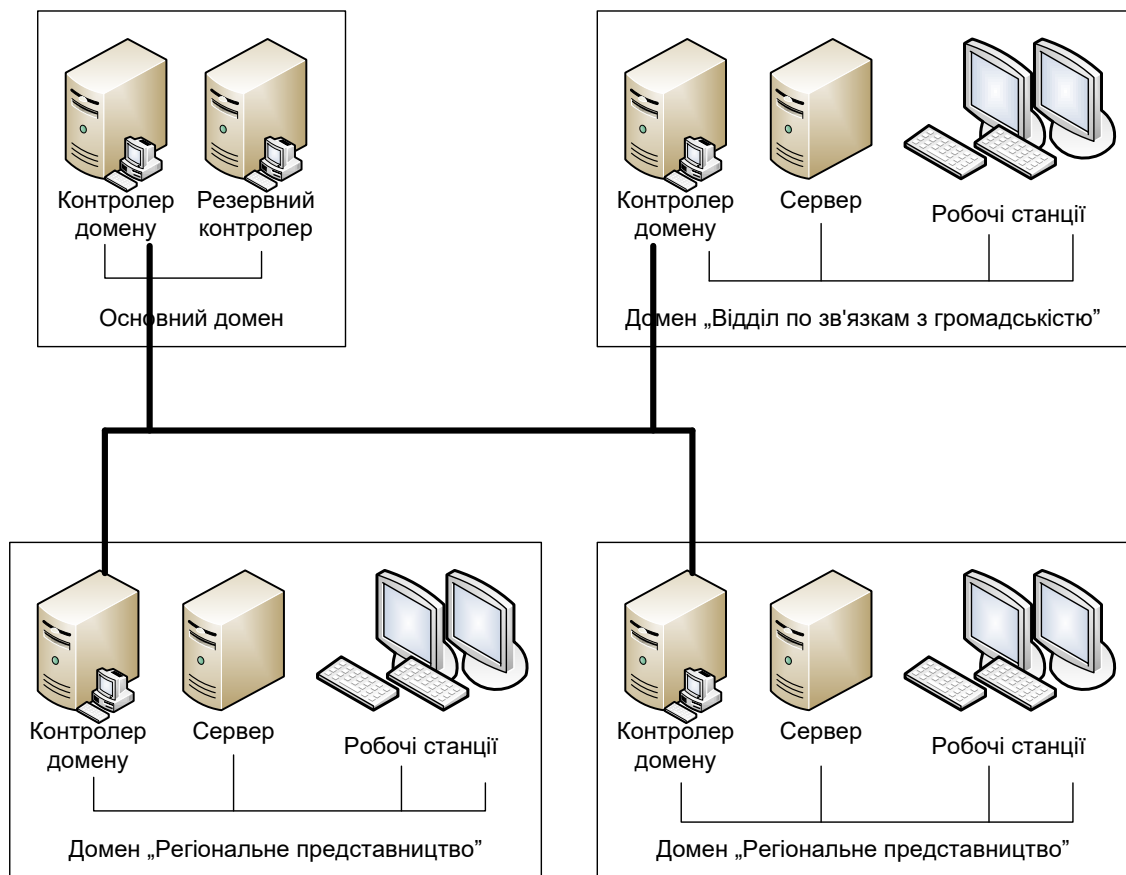


Рисунок 1.8 – Функціональна схема комп'ютерної мережі ТОВ "Марс Беверіджиз"

1.3 Постановка технічного завдання

1.3.1 Найменування, галузь застосування та мета створення системи

Розроблювальна система, називана «Інформаційна система комплексу складських терміналів» призначена для автоматизації процесу керування рухом матеріальних потоків складського комплексу. Ціль розробки – переклад застарілої СУБД на нову архітектуру клієнт-сервер, виправлення виявлених недоліків існуючої системи. Існуюча програма

«Складський термінал» працює під керуванням операційною системою MS-DOS. При цьому вона «вішає» машину при спробі запуску з-під Windows'7, тому, для роботи із програмою доводиться перезавантажувати комп'ютер у режимі командного рядка. Той факт, що вона написана під DOS, уже свідчить про незручний інтерфейс користувача. Відсутність підтримки мишки, складність, заплутаність і незрозумілість призначення деяких діалогових вікон, відсутність системи допомоги (не говорячи вже про гнучку систему контекстної підказки), незручність введення інформації та багато чого іншого ще менше приваблює до програми. Система керування базою даних побудована на технології файл-сервер. При цьому програма може працювати як з локальної, так і з мережною базою даних. Помітимо, що при відсутності доступу до мережної бази, програма автоматично перемикається на локальну базу, не видаючи при цьому ніяких попереджень і повідомлень. Ще кілька років назад, серед СУБД найбільшою популярністю користувалися СУБД dbase, Paradox, Rbase, що одержали загальну назву Xbase (створених на технології файл-сервер), а в якості інструментальних засобів найпоширенішими були Clipper і Foxpro. Зараз на ринку цих СУБД розповсюджені Access, Foxpro, Paradox, dbase. У ході експлуатації таких систем були виявлені загальні недоліки архітектури файл-сервер, які полягають у наступному:

- уся вага обчислювальної роботи лягає на комп'ютер клієнта; наприклад, якщо в результаті запиту клієнт повинен одержати 2 записи з таблиці обсягом 100000 записів, усі 100000 записів будуть скопійовані з файл-сервера на клієнтський комп'ютер; у результаті зростає завантаження мережі (мережний трафік) і збільшуються вимоги до апаратних потужностей користувацького комп'ютера; помітимо, що потреби в постійному збільшенні обчислювальних потужностей клієнтського комп'ютера обумовлюються постійно зростаючим обсягом інформації, що накопичується і оброблюється;

- оскільки БД являє собою набір файлів на мережному сервері, доступ до таблиць регулюється тільки мережною операційною системою, що робить такі БД по суті беззахисними від випадкового або навмисного викривлення інформації, що зберігається в ній, знищення або розкрадання;

- недостатньо розвитий апарат транзакцій локальних СУБД служить потенційним джерелом помилок як при одночасному внесенні змін у ту саму запись, так і при реалізації відкату результатів серії об'єднаних за змістом у єдине ціле операцій над БД, коли деякі з них завершилися успішно, а деякі – ні; це може порушувати посиальну і значеннєву цілісність БД.

Але найголовніший недолік таких СУБД, це те, що тільки дана конкретна програма здатна правильно робити зміни в БД, зберігаючи їх цілісність. Будь-яке стороннє втручання в базу даних може привести до повного руйнування даних і втраті всієї інформації.

1.3.2 Задачі, що мають вирішуватися за допомогою системи

Із усіх задач, розв'язуваних даною системою можна виділити ряд основних завдань – необхідність яких обов'язкова для будь-якої інформаційної системи подібного роду. До таких задач відносяться:

- облік продукції, яка надходить в зону складського комплексу;
- облік продукції відвантаженої із зони складського комплексу;
- можливість постійного контролю стану складського комплексу (наявність вільних, зайнятих комірок і т.д.);
- ідентифікація користувачів системи;
- формування необхідної документації.

1.3.3 Вимоги до захисту інформації у комп'ютерній мережі підрозділу

Щоб забезпечити необхідний рівень безпеки інформації у комп'ютерній мережі підрозділу, система безпеки повинна мати наступні засоби:

- засоби ідентифікації й перевірки повноважень;
- засоби забезпечення захисту файлів;
- засоби захисту ОС і програм користувачів;
- засоби шифрування/дешифрування трафіка мережі;
- засоби знищення залишків інформації в системі;
- засоби реєстрації звертань до системи.

1.3.4 Характеристика об'єктів обробки

Відповідно до перерахованих задач, реалізованих системою, дамо характеристику відповідних об'єктів обробки інформації:

Приймання продукції на склад. Продукцію на склад доставляють автомобільним або залізничним транспортом. У процесі розвантаження продукція укладається на дерев'яні піддони за певними правилами. На один піддон укладається продукція одного найменування з тим самим строком придатності. Максимальна кількість коробів, розташовуване на одному піддоні, визначається виходячи з їхнього розміру і ваги. Після формування піддони впаковують плівкою (палетизують). Упакований піддон із продукцією називається палетою. Після цього кожної палеті визначається адреса на складі. Водії електронавантажувачів розвозять палети по адресах.

Відвантаження продукції клієнтам. З відділу продажів завідувач складу одержує накладну, у якій зазначений асортименти продукції необхідний клієнтові, а також його кількість. Система визначає адреси на складі, де перебуває необхідна продукція. Видає до друку накази водіям електронавантажувачів на переміщення продукції з адрес у певне місце (місце замовлення). Після переміщення продукції з адреси перевіряється наявність усіх найменувань продукції, а також їх кількість. По прибуттю клієнта замовлення відвантажується із зони замовлення в машину або вагон.

1.3.5 Функціональні вимоги до системи

Інформаційна система повинна виконувати уведення, запис і зберігання інформації про прихід або відвантаження продукції на складі.

При прийманні продукції на склад система повинна забезпечувати два режими роботи:

- приймання продукції в ручному режимі;
- приймання продукції в автоматичному режимі.

При прийманні продукції в ручному режимі адреси, у яких буде розташована продукція, вибираються оператором самостійно і заводяться вручну. При прийманні продукції в автоматичному режимі система повинна сама вибирати зручне розташування для продукції, що надходить на склад. Оператор заводить у базу даних тільки код і строк придатності товару. Адреса для нього система визначає автоматично.

При відвантаженні товару система також повинна забезпечувати два режими:

- формування замовлення в автоматичному режимі;
- формування замовлення в ручному режимі.

При формуванні замовлення в ручному режимі оператор заводить у базу даних усю інформацію – код продукції, яка необхідна клієнтові, кількість коробів, адреса з якої їх потрібно завантажити. При формуванні замовлення в автоматичному режимі оператор заводить тільки асортименти продукції та кількість коробів, що необхідна клієнтові. Адреси, з яких необхідно буде їх завантажити система повинна прораховувати автоматично. При автоматичному прорахунку замовлення система повинна включати в замовлення код продукту необхідний клієнтові із самим раннім строком придатності.

Доступ до системи повинен бути розмежований на два рівні:

- адміністратор бази даних;
- оператор бази даних;

Для операторів бази даних повинні бути недоступні деякі функції, які доступні адміністраторові бази даних. Оператор не повинен мати доступу до редагування і доповненню даних "картка товару", "оператори", "клієнти". Адміністратор бази даних повинен мати можливість редагування цих даних (додати товар, оператора, вилучити, відредагувати наявні записи). Редагування всіх збережених документів у системі повинне бути заборонене для всіх працівників, що мають допуск. Усі документи, коли-небудь створені в системі не повинні зазнати жодних корекцій або видалення. Система повинна дозволяти операторам одержувати інформацію про поточний стан складу (стані комірок; товарі, що знаходиться в комірках), а також забезпечувати можливість блокування, розблокування комірок складу. Це необхідно для того, щоб система при прийманні продукції на склад в автоматичному режимі не використовувала заблоковані комірки.

1.3.6 Характеристика процесів обробки

У проєктованій інформаційній системі кожна із представлених функцій буде реалізована відповідним програмним модулем, кожний з яких буде мати можливість виконання ряду операцій, таких як коректування інформації, пошук даних, формування звітів. Процеси заповнення, коректування та пошуку даних будуть реалізоване однотипним способом у всіх модулях. При додаванні та редагуванні даних буде застосовуватися контроль правильності введення користувачем інформації, шляхом її порівняння з необхідним типом, діапазоном зміни можливих значень, маскою введення і т.д. При заповненні некоректної інформації буде використана система повідомлень, що повідомляють, що автоматично видаються для попередження та допомоги користувачеві. Для додавання і редагування даних будуть використані екранні форми з усіма необхідними полями введення, а також поясненнями та керуючими елементами (наприклад, кнопками), призначеними для вироблення керуючих впливів (збереження, скасування змін) і навігації (переміщенню) по БД. Для забезпечення пошуку даних також будуть використані екранні форми, що дозволяють задавати різні значення (діапазони значень) інформації, що цікавить, з контролем допустимості значень умов пошуку. Пошук інформації можна буде робити як по окремих полях таблиць, так і по сукупності полів, на частковий, або повний збіг умов пошуку, без обліку регістру букв для зручності користувача, для чисельних полів і полів типу «дата» буде реалізована можливість вказівки діапазонів зміни значень і т.д. У кожному із програмних модулів системи передбачається наявність усіх необхідних звітних форм для формування і друку документів установленної форми. Усі звіти генеруються автоматично, використовуючи вибірки інформації із БД.

1.3.7 Вихідна інформація ІС

Вихідна інформація ІС класифікується на наступні види:

- інформація, пов'язана із прийманням продукції на склад;
- інформація, пов'язана з відвантаженням продукції зі складу;
- довідкова інформація, що відображає стан на складі.

Інформація, пов'язана із прийманням продукції на склад, складається з наступних документів:

- акт про розвантаження;
- журнал приходу;

Ці документи підтверджують приймання продукції на склад. Акт про розвантаження містить наступні дані:

- назва складу, що робить приймання продукції;
- номер акту розвантаження;
- дата приймання товару на склад;
- місце приймання товару та складання даного акту;
- інформація про состав комісії, що робила приймання продукції на склад, а саме: оператор (зав. складом), що робив приймання продукції на склад; водій автомобіля, що доправив продукцію на склад; номер автомобіля, що доправив продукцію на склад;
- назва організації, звідки доставлений вантаж;
- дата і час відправлення вантажу;
- час доставки на склад;
- час закінчення розвантаження;
- номер товарно-транспортної накладної, по якій товар був доставлений на склад;
- кількість прийнятих місць (коробів) із вказівкою дати реалізації продукту;
- стан пломби на причепі автомобіля;

– наявність рекламних матеріалів, обладнання та т.і., не зазначених у супровідних документах.

Журнал приходу містить наступні дані:

- номер журналу приходу (збігається з номером акту про розвантаження);
- дата приходу продукції на склад;
- найменування складу, що робив приймання продукції;
- найменування організації, звідки була доставлена продукція;
- інформацію про прихід товару на склад з урахуванням поділу товару на палети, а саме: код товару; число коробів на палеті;
- строк придатності продукції;
- адреса місцезнаходження продукції на складі;
- інформація про оператора (зав. складом), що робив приймання товару;

Форми вихідних документів приходу (акт про розвантаження, журнал приходу) наведені в додатку А. Інформація, пов'язана з відвантаженням продукції на склад, складається з наступних документів:

- лист відвантаження зі складу;
- журнал видатку.

Ці документи підтверджують факт відвантаження продукції зі складу. Лист відвантаження зі складу містить наступні дані:

- назва складу, з якого була відвантажена продукція;
- номер замовлення;
- код клієнта, що одержує продукцію;
- назва клієнта;
- дата і час відвантаження;
- код продукції, що відвантажується;
- назва продукції, що відвантажується;
- строк придатності, що відвантажується продукції;
- кількість коробів, що відвантажуються;

- інформація про оператора, що робив відвантаження.

Журнал видатку містить наступні дані:

- номер журналу видатку;
- назва складу, що робив відвантаження;
- інформацію про видаток товару з урахуванням поділу товару на паллети, а саме:

- код товару;
- число коробів на палеті;
- строк придатності продукції;
- адреса місцезнаходження продукції на складі;
- інформація про оператора (зав. складом), що робив відвантаження товару;

Форми вихідних документів видатку (лист відвантаження зі складу, журнал видатку) наведені в додатку А. Крім зазначених вище документів, під час приймання продукції на склад повинні формуватися наклейки, які наклеюються на паллети і містять:

- код продукту;
- найменування продукту;
- строк придатності продукту;
- кількість коробів на палеті;
- адреса, у яку необхідно помістити паллету.

Приклад наклейки наведений у додатку А. При добірці замовлення клієнтові, система повинна формувати і виводити на друк накази на переміщення паллет з адрес у зону замовлення, які видаються водіям електронавантажувачів. Наказ на переміщення повинен містити:

- найменування переміщуваного продукту;
- адреса, з якої необхідно перемістити продукцію;
- кількість коробів, що перебувають у даній адресі;
- строк придатності переміщуваної продукції;
- номер замовлення, для якого проводиться переміщення;

– кількість коробів, яка необхідно перемістити з адреси в зону замовлення.

Приклад наказу на переміщення представлений у додатку А. Форми вихідної довідкової інформації досить різноманітні та повинні формуватися системою самостійно, таким чином, як це може спостерігати користувач на своєму моніторі.

1.3.8 Вимоги до інформаційної системи

Загальні вимоги. Розроблювальна система повинна являти собою закінчений програмний продукт, що реалізує автоматизацію процесу керування складським комплексом у встановленому обсязі задач. Для забезпечення ефективної роботи система повинна мати дружній графічний інтерфейс із користувачем, зрозуміле призначення функцій і наочний результат обробки інформації, надійно функціонувати в існуючому програмному оточенні та умовах експлуатації, мати засоби захисту від збоїв і відновлення. Система повинна бути відкритою для подальшого масштабування та нарощування функціональних можливостей окремих програмних модулів. Необхідним і обов'язковим є наявність вбудованої інтерактивної довідкової інформації з роботи із системою і програмної документації на систему: технічне завдання, опис програми, вихідний текст програми, керівництва системного програміста і користувача.

Вимоги до видів забезпечення і компонентам. По технічному забезпеченню: персональний комп'ютер, з достатнім обсягом оперативної пам'яті, наявність дискових накопичувачів (НЖМД, НГМД) для зберігання БД, документів і їх архівних копій, монітор, принтер для друку звітної документації. По програмному забезпеченню: системне: операційна система, драйвера для вбудованих і зовнішніх пристроїв; прикладне:

програми діагностики апаратних засобів і антивірусного захисту. По інформаційному забезпеченню: інформаційне забезпечення процесу керування рухом матеріальних потоків складського комплексу можна представити наступною схемою.

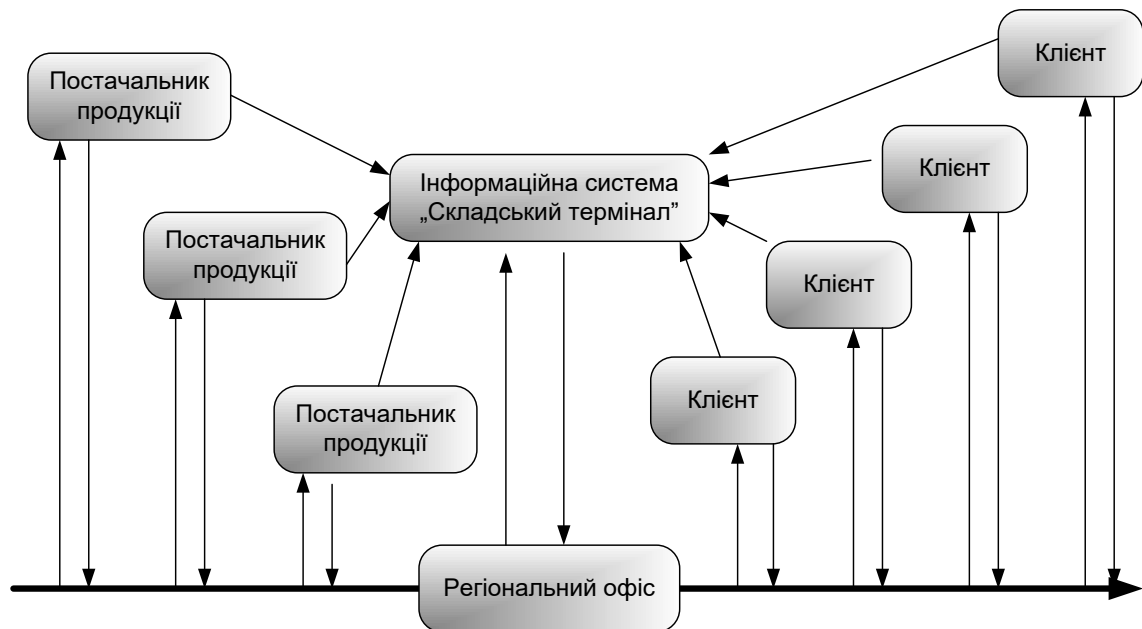


Рисунок 1.9 – Схема інформаційного забезпечення процесу управління рухом матеріальних потоків складського комплексу

2 ВИБІР ПРОГРАМНО-АПАРАТНИХ ЗАСОБІВ РОЗРОБЛЕННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ

2.1 Розроблення функціональної структури інформаційної системи складського терміналу

Відповідно до технічного завдання структуру задач, що вирішуються системою можна представити наступною схемою (рис. 2.1).



Рисунок 2.1 – Структурна схема задач інформаційної системи

Для виконання першої задачі – ідентифікації користувачів, необхідно щоб система могла:

- дозволяти роботу із системою тільки зареєстрованим користувачам;
- забезпечувати гнучке налаштування доступності процедур системи для кожного окремого користувача;
- забезпечувати збереження інформації в базі даних системи про користувача і зміни, їм зроблених.

Реалізація описаних вище процедур системи забезпечується роботою першої функції <Ідентифікація користувачів>, виконання якої

забезпечують чотири підфункції: <Уведення інформації про користувачів системи>, <Перегляд даних>, <Обробка інформації>, <Збереження інформації про зроблені користувачем операції>.

Підфункція системи <Уведення інформації про користувачів системи> реалізується за допомогою підфункцій другого рівня <Додавання даних>, <Перевірка правильності введення інформації> і <Збереження інформації>.

Підфункція <Обробка інформації> може бути реалізовано трьома підфункціями другого рівня <Пошук інформації>, <Сортування інформації>, <Зміна інформації>, які у свою чергу реалізуються підфункціями третього рівня. Для підфункції <Пошук інформації> це:

- <Завдання умов пошуку>;
- <Виконання умов пошуку>;
- <Перегляд результатів пошуку>;
- <Друк результатів пошуку>;

Для підфункції <Сортування інформації> це:

- <Завдання умов сортування>;
- <Виконання умов сортування>;
- <Перегляд результатів сортування>;
- <Друк результатів сортування>;

Для підфункції <Зміна інформації> це:

- <Додавання інформації>;
- <Виправлення інформації>;
- <Перевірка правильності введення інформації>;
- <Видалення інформації>;
- <Перегляд результатів>;
- <Збереження змін>.

Підфункція системи <Збереження інформації про зроблені користувачем операції> має важливе значення, і тому виділена в окрему підфункцію. Призначення даної підфункції полягає в збереженні

ідентифікатора користувача і часу в службових полях бази даних при добутку користувачем, будь-яких дій з базою даних системи. Це необхідно для відновлення інформації про того – хто, коли і які операції робив. Структурну схему функції системи <Ідентифікація користувачів> можна представити схемою, представленої на рис. 2.2.

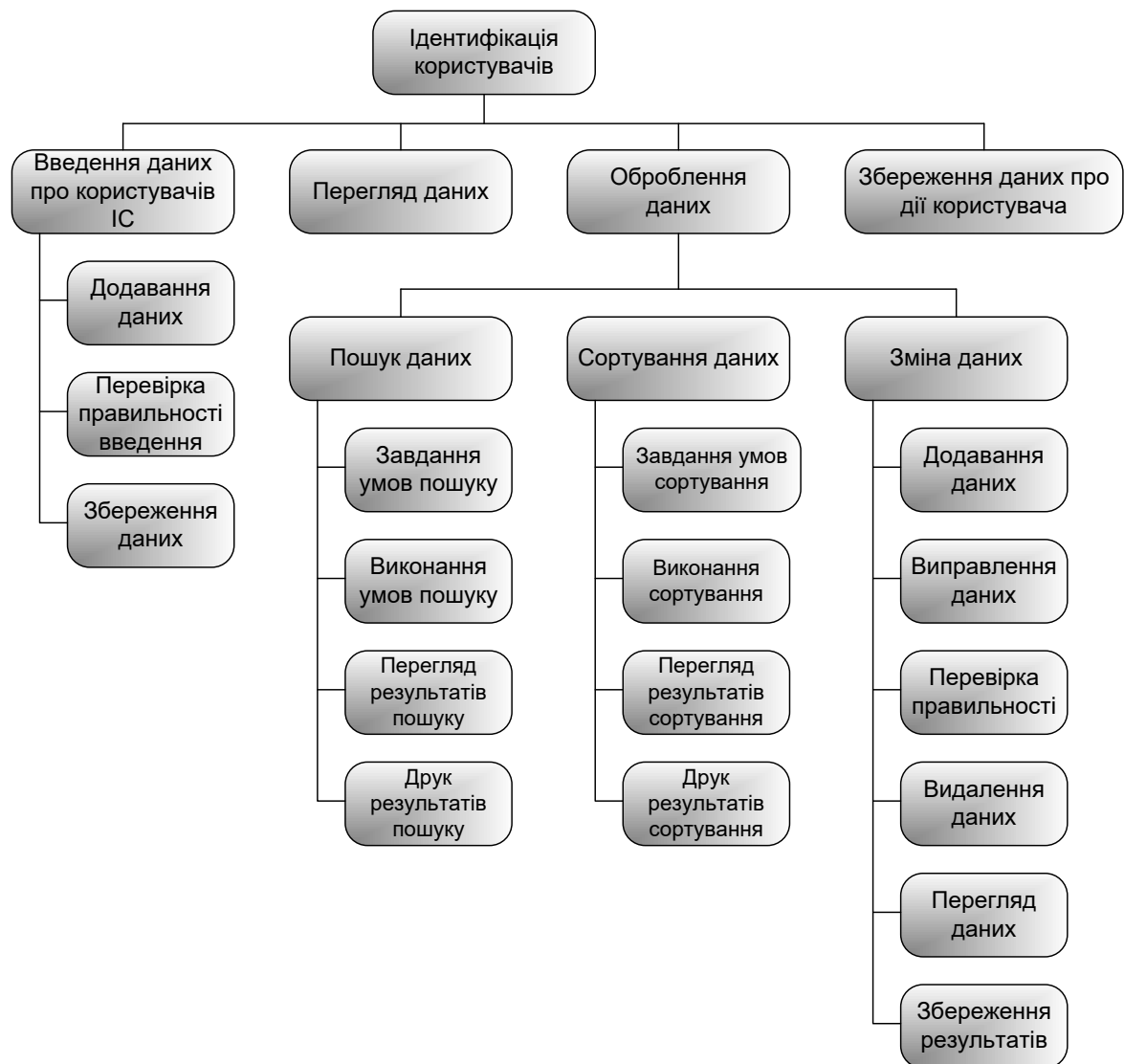


Рисунок 2.2 – Структурна схема функцій системи <Ідентифікація користувачів>

Для виконання другої задачі – обліку продукції, що надходить в зону складського комплексу, необхідно щоб система могла:

– забезпечувати можливість введення інформації про прихід продукції на склад у двох режимах:

а) автоматичний режим – режим, при якому на систему лягає задача вибору оптимального розташування продукції на складі. Оператор уводить у систему прийняту продукцію, система визначає – де її розташувати.

б) ручний режим – режим, при якому оператор вручну визначає розташування прийнятої продукції на складі.

- забезпечувати можливість коректування введеної інформації;
- забезпечувати перегляд наявної інформації;
- контролювати правильність введення інформації;
- автоматично зберігати введену інформацію;
- забезпечувати можливість видалення інформації;
- забезпечувати можливість пошуку необхідних даних;
- забезпечувати можливість сортування наявної інформації.
- забезпечувати формування необхідної документації та вивід її на друк.

Реалізація описаних вище процедур системи забезпечується роботою другої функції <Облік продукції, що надходить на склад>, виконання якої забезпечують такі підфункції як <Введення інформації про надходження продукції на склад>, <Перегляд усіх наявних даних>, <Обробка наявної інформації>, <Формування необхідної документації>.

Підфункція системи <Введення інформації про надходження продукції на склад> реалізується за допомогою підфункцій другого рівня <Автоматичний режим> і <Ручний режим>. Дані підфункції, у свою чергу, реалізуються за допомогою підфункцій третього рівня:

- <Додавання даних>;
- <Перевірка правильності введення даних>;
- <Внесення змін>;
- <Видалення інформації>;

- <Перегляд результату>;
- <Видача необхідної інформації на печатку>;
- <Автоматичне збереження інформації>.

Для реалізації підфункцій <Автоматичний режим> також необхідна підфункція третього рівня <Автоматичне визначення необхідної інформації та занесення її в базу даних системи>. У задачі даної підфункції входить визначення найбільш оптимального розташування продукції на складі і видача відповідних адрес складу для прийнятої продукції.

Підфункція системи <Перегляд усіх наявних даних> реалізується за допомогою підфункцій другого рівня <Перегляд документів> і <Друк документів>. Підфункція системи <Обробка наявної інформації> може бути реалізовано за допомогою двох підфункцій другого рівня <Пошук інформації>, <Сортування інформації>. Виконання задач підфункції <Пошук інформації> здійснюється підфункціями третього рівня:

- <Завдання умов пошуку інформації>;
- <Виконання пошуку інформації>;
- <Перегляд результатів пошуку>;
- <Друк результатів пошуку>;

а підфункція <Сортування інформації> у свою чергу підфункціями:

- <Завдання умов сортування>;
- <Виконання сортування>;
- <Перегляд результатів сортування>;
- <Друк результатів сортування>.

Підфункція системи <Формування необхідної документації> реалізується чотирма підфункціями другого рівня:

- <Формування звітних форм>;
- <Формування звітів>;
- <Перегляд і друк звітів>;
- <Збереження звітів>.

На підставі вищевикладеного, реалізацію функції <Облік продукції. Що надходить на склад> можна описати схемою представленої на рис. 2.3.

Для виконання третього задачі - обліку продукції відвантаженої із зони складського комплексу, необхідно щоб система могла:

- забезпечувати можливість введення інформації про відвантаження продукції зі складу у двох режимах:

- а) автоматичний режим – режим, при якому на систему лягає задача оптимального вибору адреси складу, з яких буде проводитися відвантаження продукції клієнтам. Оператор уводить у систему, що відвантажуються продукцію, система визначає – де її взяти;

- б) ручний режим – режим, при якому оператор вручну визначає розташування продукції, що відвантажуються, на складі;

- забезпечувати можливість коректування введеної інформації;
 - забезпечувати перегляд наявної інформації;
 - контролювати правильність введення інформації;
 - автоматично зберігати введену інформацію;
 - забезпечувати можливість видалення інформації;
 - забезпечувати можливість пошуку необхідних даних;
 - забезпечувати можливість сортування наявної інформації;
 - забезпечувати формування необхідної документації та вивід її на друк.

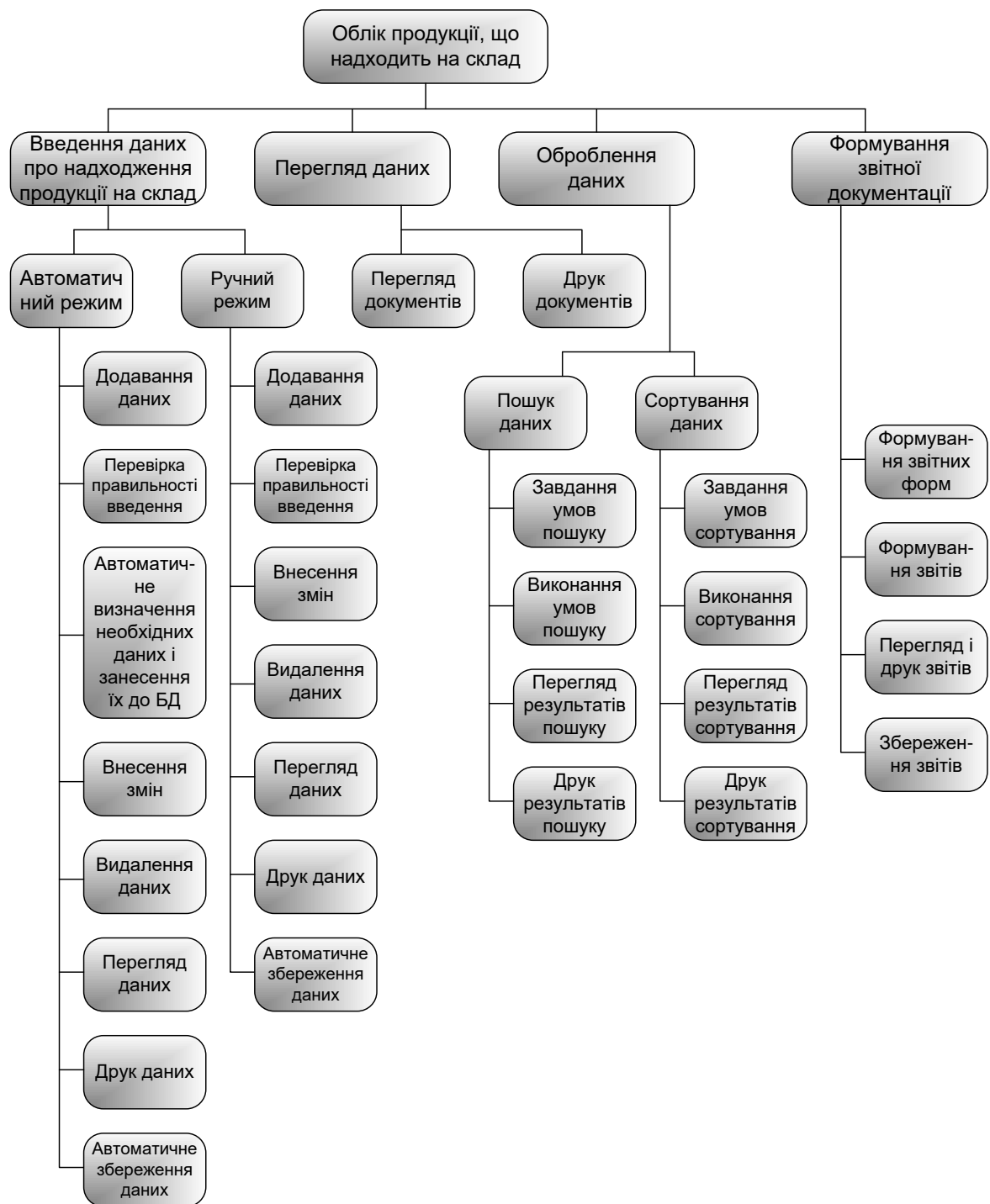


Рисунок 2.3 – Структурна схема функції системи <Облік продукції, що надходить на склад>

Реалізація описаних вище процедур системи забезпечується роботою другої функції <Облік продукції відвантаженої зі складу>, виконання якої забезпечують такі підфункції як <Введення інформації про відвантаження продукції зі складу>, <Перегляд усіх наявних даних>, <Обробка наявної

інформації>, <Формування необхідної документації>. Як видно, перелік виконуваних даної функцією процедур аналогічний списку процедур попередньої функції, тому буде доцільно використовувати структуру функції <Облік продукції, що надходить на склад>, з різницею лише в тому, що процедури будуть виконуватися для інформації з бази даних документів відвантаження зі складу. Так само, для реалізації функції <Облік продукції відвантаженої зі складу> зміниться призначення підфункції третього рівня <Автоматичне визначення необхідної інформації та занесення її в базу даних системи>. У задачі даної підфункції входить визначення найбільш оптимального розташування продукції на складі та видача відповідних адрес складу для продукції, що відвантажується. Структурна схема функції системи <Облік продукції відвантаженої зі складу> буде аналогічна схемі функції <Облік продукції, що надходить на склад> (рис. 2.3).

Для виконання четвертої задачі – контроль над станом складського комплексу, необхідно щоб система могла вести:

- облік постачальників продукції;
- облік клієнтів складського комплексу;
- облік найменувань продукції складського комплексу;
- можливість добутку операцій з віртуальними комірками складу.

Реалізація описаних вище процедур системи забезпечується роботою четвертої функції <Контроль над станом складського комплексу>, виконання якої здійснюється підфункціями: <Перегляд комірок складу>, <Операції із комірками складу>, <Облік постачальників продукції>, <Облік клієнтів>, <Облік найменувань продукції>. Підфункція <Операції із комірками складу> реалізується чотирма підфункціями другого рівня:

- <Створення віртуальних комірок>
- <Видалення віртуальних комірок>
- <Редагування інформації про комірки>
- <Блокування комірок>

Підфункція <Облік постачальників продукції> також реалізується чотирма підфункціями другого рівня:

- <Додавання постачальника>
- <Перегляд наявних постачальників>
- <Видалення постачальника>
- <Редагування інформації про постачальника>

Підфункції <Облік клієнтів> і <Облік найменувань продукції> реалізовані також як і попередня підфункція, і мають по чотири підфункції, відповідальні за додавання інформації, перегляд наявних даних, видалення інформації та редагування інформації. Структурна схема функції системи <Контроль над станом складського комплексу> представлена на рис.2.4.

Для виконання п'ятої задачі – формування необхідної документації, необхідно щоб система могла:

- пошук інформації для складених звітів;
- перегляд сформованих звітів;
- створення запитів для зв'язку звітів з даними;
- створення форм звітів;
- генерацію звітів;
- настроювання представлення звітів;
- друк і збереження звітів у різних форматах.



Рисунок 2.4 – Структурна схема функцій системи <Контроль над станом складського комплексу>

Реалізація описаних вище процедур системи забезпечується роботою п'ятої функції <Формування необхідної документації>, виконання якої здійснюється підфункціями <Створення запитів для зв'язку звітів з даними>, <Створення форм звітів>, <Формування звітів> і <Обробка звітів>. Підфункція <Створення запитів для зв'язку звітів з даними> призначена для створення запитів до бази даних. У запитах вказуються таблиці даних і необхідні умови вибірки даних. Кожний запит має своє ім'я, посилаючись на яке проводиться формування даних для звіту, що викликав даний запит. Функціонування даної підфункції здійснюється п'ятьма підфункціями другого рівня:

- <Перегляд списку запитів>;
- <Додавання запиту>;
- <Редагування запиту>;
- <Видалення запиту>;
- <Перегляд результатів запиту>.

Підфункція <Створення форм звітів> призначена для створення шапок звітів. У формі звіту створюються необхідні написи (заголовок звіту, заголовки стовпців і т.д.) а також вказуються використовувані запити, необхідні для формування даного звіту. Функціонування даної підфункції здійснюється п'ятьма підфункціями другого рівня:

- <Перегляд списку форм>;
- <Редагування існуючих форм>;
- <Додавання форми>;
- <Видалення форми>;
- <Перегляд форми>.

Підфункція <Формування звітів> реалізується в системі за допомогою підфункцій другого рівня: <Пошук даних для звіту>, <Вибір форми звіту>, <Вибір запиту для звіту>, <Генерація звіту>. Підфункція <Обробка звітів> реалізується в системі за допомогою підфункцій другого рівня: <Перегляд і налаштування звітів> <Друк звітів> <Завантаження звітів> <Збереження звітів у необхідному форматі>. Складемо структурну схему функції системи <Формування необхідної документації> рис. 2.5.

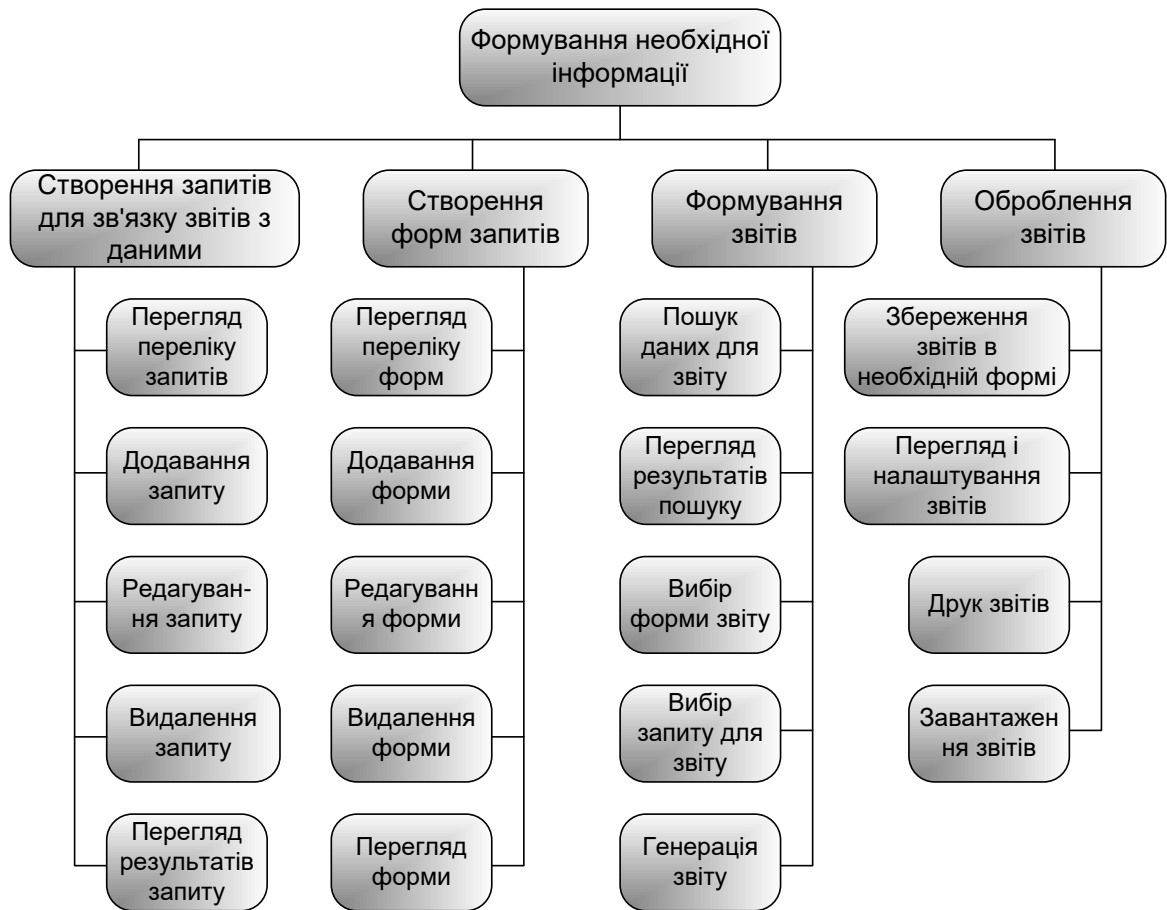


Рисунок 2.5 – Структурна схема функцій системи <Формування необхідної документації>

2.2 Вибір програмних засобів для розроблення інформаційної системи

Розглянемо сучасні програмні засоби, за допомогою яких можливо створювати сучасні інформаційні системи.

2.2.1 СУБД Microsoft Access

«Microsoft Access» (повна назва Microsoft Office Access) — система управління базами даних від компанії Майкрософт, програма, що входить до складу пакету офісних програм Microsoft Office. Має широкий спектр функцій, включаючи зв'язані запити, сортування по різних полях, зв'язок із зовнішніми таблицями і базами даних. Завдяки вбудованій мові VBA, в самому Access можна писати підпрограми, що працюють з базами даних. Версії: 1993 Access 2.0 для Windows (Office 4.3) 1995 Access 7 для Windows 95 (Office 95) 1997 Access 97 (Office 97) 1999 Access 2000 (Office 2000) 2001 Access 2002 (Office XP) 2003 Access 2003 (із комплекта програм Microsoft Office 2003) 2007 Microsoft Office Access 2007 (із комплекта програм Microsoft Office 2007) 2010 Microsoft Office Access 2010 (із комплекта програм Microsoft Office 2010).

Основні компоненти MS Access:

- конструктор таблиць;
- конструктор екранних форм;
- конструктор SQL-запитів (мова SQL в MS Access не відповідає стандарту ANSI);
- конструктор звітів, що виводяться на друк.

Таблиця — це основний об'єкт бази даних, призначений для збереження даних. Запит — вибирає дані з таблиць згідно з умовами, що задаються. Форма — відображає дані з таблиць або запитів відповідно до форматів, описаних користувачем. Форма дозволяє переглядати, редагувати та друкувати дані. Звіт — відображає і друкує дані з таблиць або запитів згідно з описаним користувачем форматом. У звіті дані редагувати не можна.

Вбудовані засоби взаємодії MS Access зі зовнішніми СУБД з використанням інтерфейсу ODBC знімають обмеження, властиві Microsoft

Jet Database Engine. Інструменти MS Access, які дозволяють реалізувати таку взаємодію називаються "пов'язані таблиці" (зв'язок з таблицею СУБД) і "запити до сервера" (запит на діалекті SQL, який "розуміє" СУБД). Корпорація Microsoft для побудови повноцінних клієнт-серверних додатків на базі MS Access рекомендує використовувати в якості рушія бази даних СУБД MS SQL Server. При цьому є можливість поєднати з властивою MS Access простотою інструменти для управління БД і засоби розробки. Відомі також реалізації

Таблиця 2.1 – Сумісність Access зі сторонніми джерелами даних

СУБД	Версія Access	Драйвер	Обновлювані запити
Файли Excel	всі	вбудований	Ні
SQLite			Так
MySQL	2000-2003	MyODBC v.3.51.X, 5.1.X	Так
PostgreSQL			Так
Firebird			Так
1C v.7.7 (dbf)	2003	Visual FoxPro ODBC driver v.6.01.8629.01	Ні
Paradox			
Oracle			
Текстові файли	всі	вбудований	Ні
Таблиці html	всі	вбудований	Ні

Оскільки Access являє собою СКБД, що складається з багатьох частин, у табл. 4 описуються компоненти чи об'єкти, що складають базу даних Access.

Таблиця 2.2 – Компоненти Access

Об'єкт	Опис
Таблиця	Об'єкт в якому зберігаються дані. Виглядає схожим до електронної таблиці
Форма	Шаблон відображення даних, що полегшує читання і сприйняття даних на екрані
Отчет	Шаблон даних для друку, використовується для друку даних, що зберігаються в таблиці
Запрос	Вибирає дані з таблиць на основі критеріїв заданих користувачем
Макрос	Автоматизує роботу об'єктів бази даних за допомогою команд чи подій
Модуль	Процедура мовою VBA для автоматичного виконання більш складних операцій, що не можуть виконати макроси

Спільна робота всіх цих об'єктів бази даних створює базу даних Access. Проте у базі даних Access необов'язково використовувати всі ці об'єкти. Наприклад, макроси і модулі використовують, тільки якщо в базі даних необхідна яка-небудь автоматизація.

При створенні баз даних без використання майстра баз даних, варто керуватися деякими корисними основними принципами. Важливий порядок, у якому створюються об'єкти бази даних. Базу даних рекомендується створювати в наступному порядку:

Таблиці - в таблицях містяться поля, у які поміщені дані бази даних, і на них посилаються всі об'єкти бази даних. Таблиці є фундаментом бази даних. Усі зв'язки між таблицями повинні бути створені на цьому етапі,

Запити - додатково з'єднують інформацію в таблицях і відфільтровують дані, які повинні відображатися на екрані чи в звіті.

Форми - служать екранним чи інтерфейсом користувача бази даних.

Звіти - після завершення конструювання інтерфейсу користувача, що звичайно працює як частина бази даних, призначена для введення

користувачем, можна сконструювати звіти, що служать вихідними даними для друку.

Макроси і модулі призначені для автоматизації роботи елементів керування й об'єктів бази даних.

Ознайомившись з основними концепціями баз даних роботу Access розпочинають у головному вікні рис. 1 на закладці Таблицы зі створення структури. Опишемо інтерфейс програми та застосуємо Access для створення таблиць.



Рисунок 2.1 – Головне вікно програми Access

Щоб створити нову таблицю, насамперед слід відкрити базу даних, для якої вона призначена. Потім клацніть на вкладці Таблиці, а в ній клацніть на кнопці Створити. Тоді Access відобразить діалогове вікно Нова таблиця (рисунок 2.2).

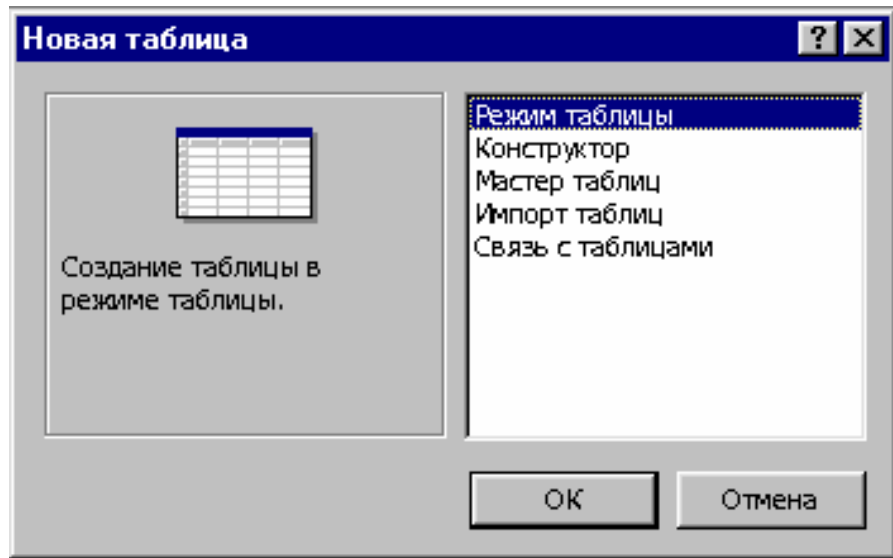


Рисунок 2.2 – Вибір методу створення таблиці

У Access застосовують наступні методи створення таблиць:

- *Режим таблиці.* Нову таблицю створюють шляхом введення даних безпосередньо в порожню базу даних, подібно введенню даних в електронну таблицю. Після завершення введення даних і збереження таблиці Access автоматично переглядає дані, а потім створює поля відповідних типів і форматів. Якщо користувач раніше не створював таблиць або дані уже готові до застосування, цей метод дуже корисний.

- *Конструктор.* У цьому режимі таблицю створюють із самого початку, вибираючи імена, типи і формати полів. Метод максимально гнучкий і надає можливість повного контролю над створенням таблиці.

- *Майстер таблиць.* Майстер автоматично створює таблиці, опираючись на попередньо заданий список типових таблиць. Користувач надає майстру інформацію про потрібну таблицю, і майстер допомагає її створити.

- *Імпорт таблиць.* Якщо потрібні дані вже зберігаються в іншій програмі чи іншій базі даних Access, але бажано постійно зберігати їх у новій таблиці, імпортування даних - найкраще рішення. При імпортуванні Access копіює дані у вихідному місцеположенні і автоматично створює

Неприпустимо заносити в поле дані несумісного з ним типу. Не можна, наприклад, розміщати текст у числовому полі. У Access можна використовувати дані наступних типів:

- *Текстовий*: алфавітно-цифрові символи;
- *Поле МЕМО*: алфавітно-цифрові символи (до 65 535 символів);
- *Числовий*: будь-які числові дані;
- *Дата/час*: дата і час (до 8 байтів);
- *Грошовий*: округлені числові дані для грошових значень;
- *Лічильник*: унікальні, послідовно зростаючі (на 1) чи випадкові числа, використовувані Access для автоматичної нумерації кожного запису, що вводиться;
- *Логічний*: логічні значення, що можуть містити одне з двох можливих значень, такі як True/False і On/Off;
- *Поле об'єкта OLE*: Об'єкти OLE, графічні чи зображення інші дані в війковому форматі;
- *Гіперпосилання*: рядок з букв і цифр, що визначає шлях доступу до документа, сторінці Web чи конкретному місцеві в документі;
- *Майстер підстановок*: майстер, що створює поле зі списком, що відображає список припустимих значень, що розкривається, з іншої таблиці.

По замовчуванню Access автоматично призначає всім новоствореним полям тип даних текстовий.

В стовпчику опис можна зберігати інформацію про поле.

Створюване поле, подібно іншим об'єктам, має властивості. Ці властивості можна встановити, щоб задати метод збереження і відображення даних цього поля. Поля, які створюються в режимі Конструктор, використовуються іншими об'єктами бази даних, що працюють з таблицями, наприклад формами, звітами і запитами.

– Властивості полей в області Властивості поля змінюють на вкладках Общие і Подстановка. Нижче коротко описані властивості полів на вкладці Загальні, залежні від типу обраних даних:

– *Размер поля.* Задає максимальне число символів текстового поля й обмежує діапазон значень числових полів.

– *Новые значения.* Визначає, як будуть встановлюватись нові значення для полів типу Лічильник (як Послідовні — зростаючі з інкрементом 1 для кожного нового чи запису як Випадкові — ті, що отримують випадкові значення).

– *Формат поля.* Задає формат відображення даних поля, наприклад, дат, чисел і т.д.

– *Число десятичных знаков.* Задає число десяткових знаків у полях типу Числовий і Грошовий.

– *Маска ввода.* Використовується тільки для полів типу Текстовий і Дата/час.

– *Надпись.* Надає для поля напис, що буде використовуватись у формах і звітах, а також у заголовках таблиць, що містять це поле.

– *Значение по умолчанию.* Задає стандартне значення, вводиться автоматично в нові поля.

– *Условие на значение.* Обмежує дані, що вводяться, значеннями, що задовольняють критеріям вірогідності (максимально можливе число, достовірна дата не пізніше сьогоднішнього дня і т.д.).

– *Пустые строки.* Дозволяє вводити порожній рядок у поля типу Текстова, поле МЕМО і Гіперпосилання.

– *Индексированное поле.* Створює індекс, що базується на даному полі.

На вкладці Підстановка містяться додатково установлені властивості. Ці установки властивостей залежать від типу даних, використовуваних у полі, і значення першої властивості поля на вкладці Підстановка, який називається Тип елемента керування.

Нижче приведений опис властивостей елемента керування типу Список на вкладці Підстановка:

- *Тип елемента управління.* Задає тип елемента керування, використовуваного для відображення поля на формі. Елементи керування можуть відноситися до наступного типам: Поле (стандартне), Список і Поле зі списком.

- *Тип источника строк.* Визначає тип джерела даних, що надходять у Список чи Поле зі списком: з таблиці, запиту, з даних у введеному списку або зі списку імен полів в таблиці чи запиті, зазначених у значенні властивості Источник строк.

- *Приєднаний стовпець.* Задає число стовпців у списку або в поле зі списком, які з'являються в Источник строк.

- *Число стовпців.* Задає число стовпців, відображуваних у Поле со списком чи в Списку.

- *Заглавия столбцов.* Відображає імена перших полів списку/поля зі списком з наявних у полі Источник строк.

- *Ширина стовпців.* Вказує ширину списку, що розкривається, чи стовпця списку.

Щоб задати властивості полів у режимі Конструктор, виділіть поле, для якого необхідно задати властивості. Тоді в області панелі Свойства поля відображається вкладка Общие. У ній представлені властивості даного поля.

Потім клацніть на тій властивості, який необхідно задати. Введіть значення властивості (чи вкажіть його в списку, що розкривається.). Точно так само задайте інші властивості.

Крім установок, описаних у попередньому розділі, у режимі Конструктор можна змінювати й інші установки і вносити інші модифікації в таблиці. Нижче представлений короткий огляд цих установок:

– *Ключове поле.* Корисне, хоча і необов'язкове, створювати Ключове поле в кожній таблиці. Ключове поле містить значення первинного індексу, що допомагає Access ефективно обробляти дані, а також полегшує створення відношень і зв'язків з іншими таблицями. Щоб задати Ключове поле, клацніть на панелі інструментів Конструктор таблиць на кнопці Ключове поле або виберіть у меню команду Правка ► Ключевое поле.

– *Индексированное поле.* Подібно ключовим полям, індексування допомагає Access знаходити значення й організовувати дані. Ключові поля Access індексує автоматично. Щоб індексувати інші поля, виберіть індексируемое поле в області панелі Бланк, в області панелі Властивості поля виділіть властивість Индексированное поле, а потім виберіть тип індексу в списку, що розкривається.

Запит - один з найбільш потужних об'єктів MS Access, який дозволяє ефективно представити інформацію, що містять таблиці, з певними властивостями. В деякому розумінні запит подібний до фільтрів, коли з таблиць будується виборка за певною умовою. Але на відміну від фільтру запит дозволяє отримати більш змістовний результат. Перш за все, це пояснюється тим, що фільтр дає інформацію для перегляду (друку), але, на відміну від запиту автоматично не зберігається, як окремий об'єкт бази даних. Запити, маючи таку властивість, дозволяють динамічно поновлювати інформацію у своїх таблицях, якщо у таблицях бази даних виникла зміна інформації. Крім цього, запит має і зворотню дію: якщо змінювати інформацію у його таблицях, то таблиці бази даних, на базі яких побудований запит, будуть адекватно змінювати свою інформацію.

Бази даних можуть містити тисячі і десятки тисяч записів. Типову задачу опрацювання даних формують так: серед записів відшукати в БД ті, які задовольняють деякий критерій (умову) і вивести їх на екран, папір чи створити з них іншу таблицю.

Для відшукування потрібних даних використовують фільтри і запити. *Фільтри* застосовують у випадку нескладних умов пошуку і виведення даних на екран чи папір.

SQL (Structured Query Language – структурована мова запитів) – це стандартна мова програмування, яка має свої корені у 70-х роках минулого століття, коли фірма IBM створювала програмне забезпечення для роботи з реляційними базами даних. Сьогодні існує міжнародний стандарт мови SQL, якого дотримуються розробники програмного забезпечення з обробки реляційних баз даних. У MS Access вбудовано деякий діалект мови SQL, який використовується в таких діях, як побудова запитів на вибірку, підсумовуючих запитів, перехресних запитів, запитів на побудову таблиць, запитів на оновлення, додавання або видалення записів таблиці. Мову SQL MS Access використовує при роботі з базою даних, в тому числі неявно, коли користувач діє в рамках звичайного запиту – кожний запит, незалежно від того, як він створювався, зберігається у вигляді інструкції SQL. Необхідність використання мови SQL виникає в різних випадках. Наприклад, коли є необхідність побудувати так званий *підлеглий запит*, результати якого використовуються в якості умов в інших запитах. Синтаксис діалекту MS SQL необхідно знати, щоб безпосередньо на цій мові виконувати ті чи інші SQL-запити. Для початку цікаво познайомитись, як в термінах SQL зберігається будь-який запит. Для цього необхідно лише скористатись дією меню MS Access *Вид - Режим SQL*, коли стає активним той чи інший запит.

Ядром мови SQL є інструкція SELECT, яка використовується для відбору полів і записів таблиць бази даних. Синтаксис інструкції SELECT використовує п'ять основних речень і його можна представити в наступному вигляді:

```
SELECT <список полів>
FROM <список таблиць>
[WHERE <специфікація вибору записів>]
[GROUP BY <специфікація групування>]
[HAVING <специфікація вибору груп>]
```

[ORDER BY <специфікація сортування]

Тут речення FROM визначає таблиці або запити, які є джерелом даних для запиту, що створюється. Речення WHERE та HAVING визначають умови відбору записів і груп записів відповідно. Речення GROUP BY визначає поле (поля) для подальшого формування запиту відносно груп даних з однаковими значеннями цього поля (полів). Речення ORDER BY визначає сортування записів. Крім цих речень в кожному діалекті мови SQL можуть зустрітись інші речення стандарту SQL. У MS Access реалізовані чотири з них:

- TRANSFORM – для побудови перехресного запиту;
- IN – для зв'язку з віддаленою базою даних;
- DISTINCTROW – для відбору унікальних записів;
- WITH OWNERACCESS OPTION – для побудови запиту користувачами, що не мають прав доступу до відповідних таблиць цього запиту.

Приклад Простого SQL запиту на об'єднання:

```
SELECT [Сума] as [залишок сумарний] from [доходи]
UNION ALL select [ціна]*(-1) from [витрати];
```

Є два основні способи відображення даних з БД для візуального огляду: 1) у вигляді таблиці; 2) у вигляді форми.

Форма подібна до бланка чи карточки (рис. 2.4).

ЗАГАЛЬНА : форма

Січень | лютий | Березень | Квітень | Травень | Червень | Липень | Серпень | **Загальні витрати та доходи**

Доходи

Дата	Сума	Джерело
► 01.05.03	5,00 грн.	Зарплата
*	0,00 грн.	

Записи: 14 | 1 | ► | ►► | із 1

Всього доходи: 5,00 грн.

Витрати

Дата	Ціна	Мета	Статті витрат	Примітка
► 05.05.03	40,00 грн.	ремонт	машина	
* 06.02.03	0,00 грн.			

Записи: 14 | 1 | ► | ►► | із 1

Всього витрат: 40,00 грн.

Записи: 14 | 1 | ► | ►► | із 1

Залишок: 1 473,00 грн.

Рисунок 2.4 – Приклад оформлення інтерфейсу за допомогою форми.

Прикладами форми є будь-який бланк, сторінки з паспорта, карточка з досьє чи бібліографічна карточка на книжку в бібліотеці. Одна форма містить дані лише з одного запису.

Перехід до форм, окрім кращої візуалізації даних, дає низку додаткових можливостей. Виявляється, що на формі можна зручно розташувати:

- поля типу OLE з картинками, фотографіями тощо;
- елементи керування: кнопки, перемикачі тощо.
- надписи: заголовки форми, рубрик (а також розрисувати форму чи задати фоновий рисунок-заставку: сутінки, глобус, хмари тощо);
- обчислювальні поля (це також елементи керування) для відображення результатів обчислень, виконаних на базі наявних полів;
- закладки (багатосторінкові форми, де поля групують за змістом на різних закладках).

Є декілька способів створення форм, а саме:

- автоматично за допомогою команди Автоформа;

- за допомогою майстра форм;
- вручну за допомогою конструктора форм;
- комбінованим способом.

Наприклад, виконавши команду Автоформа ► У стовпець, отримаємо форму, в якій усі поля з таблиці будуть розташовані в стовпець і вирівняні до лівого краю.

Переваги:

- 1) в MS Access є стандартна кнопкова форма, за допомогою якої можна легко створити оболонку для реалізованої ІС;
- 2) в MS Access є макроси, за допомогою яких можна самим створити підходящу для вас кнопку форму;
- 3) в MS Access можна легко створювати потрібні для роботи реалізованої ІС бази даних, що зберігають потрібну інформацію;
- 4) в MS Access автоматично формуються звіти;
- 5) в MS Access є вбудовані формати "типи даних", які не дозволяють користувачеві вводити неправильні "типи даних".

Недоліки:

До недоліків цієї концепції можна віднести маленький набір функціональних можливостей СУБД Access, у порівнянні з високим рівнем мовою програмування C#, C++, Delphi.

2.2.2 СУБД Interbase

Інший розповсюдженої СУБД є система від компанії Interbase. Дана СУБД також є клієнт-серверної, отже вона призначена для компаній, що мають інформаційну мережу з потужним сервером. Ця СУБД використовує теж реляційну модель даних, але містить елементи об'єктно-орієнтованої моделі даних. На 2013 рік самої нової є версія Interbase 13g Release 2.

Модуль Interbase Real Application Clusters наступне покоління продукту Interbase Parallel Server, забезпечує прозору масштабованість додатків за рахунок швидкого і ефективного спільного використання кластерного кешу для погодженого доступу до даних.

Аварійне відновлення Interbase Data Guard забезпечує клієнтів системою швидкого відновлення з аварійних ситуацій. Операції відновлення в Data Guard передбачають можливі випадки, прості у використанні та повністю автоматизовані. Фізичне резервування бази даних здійснюється шляхом передачі журналів операцій з основного сервера на резервний. Операції із цих журналів потім фізично виконуються засобами відновлення (цей процес називається «довиконанням»), щоб погодити зміст резервного і основного серверів. Резервна база даних ідентична основній за фізичною структурою і навіть може бути використана при операціях читання.

В Interbase Data Guard з'явився новий механізм резервування бази даних – логічна резервна база даних (Logical Standby Database), що відрізняється від фізичної способом реалізації. Матеріалом для обох резервних баз даних служать журнали операцій, але замість прямого їхнього застосування логічна база даних екстраполює з них оператори SQL і виконує їх на резервному сервері, як звичайні Sql-Команди. У підсумку резервна база даних доступна для нормальних операцій читання і запису. Interbase Data Guard має здатність синхронно або асинхронно переписувати відновлення журналів операцій прямо з основної бази даних у резервну. Це дозволяє запропонувати всеосяжне вирішення аварійного відновлення без опори на сторонні продукти для зберігання оперативних копій журналів відновлення. Таким чином, у будь-якій ситуації клієнти можуть бути впевнені, що їх резервна база даних збереже всі операції, зроблені аж до моменту збою.

Interbase Fail Safe – чотирьохвузлова відмовостійка конфігурація для Windows Interbase Fail Safe забезпечує необхідну для електронного бізнесу

безперервну доступність і захист від системних збоїв у кластерній архітектурі на базі Windows NT – 2008. Interbase Fail Safe виконує аварійне відновлення для бази даних і серверів додатків як у двовузлових, так і в чотирьохвузлових кластерах Windows NT – 2008.

База даних Interbase забезпечує захист даних за рахунок удосконаленого керування доступом, шифрування бази даних і застосування нової технології тонкого аудита. Віртуальна приватна база даних Interbase (Virtual Private Database, VPD) забезпечує керування доступом користувачів на рівні рядків даних (записів БД). Механізм Interbase Label Security розширює функціональність VPD, управляючи доступом до даних на основі міток. Ця технологія підходить для ASP, тому що з її допомогою в одній і тій же базі даних може зберігатися інформація декількох компаній. У базі даних Interbase передбачені засоби шифрування даних на основі технології Interbase Advanced Security при передачі даних з бази і назад. Вбудовані можливості шифрування захищають найціннішу інформацію навіть від привілейованих користувачів, які можуть перевищити свої повноваження, а також від зловмисних користувачів, що намагаються прочитати файли даних з операційної системи. Тонкий аудит відслідковує дії бази даних, включаючи оператори, виконувані користувачами інформацію, що їй повертається. Це застерігає користувачів від перевищення своїх повноважень, тому що вони знають, що всі їх дії відслідковуються. Широкі можливості аудита також допомагають виявити проломи в безпеці. Наприклад, при звертанні до секретних областей бази даних оброблювач подій може відправити попередження адміністраторові. Якщо дії користувача будуть визнана небезпечними, сеанс роботи з базою даних буде негайно перерваний.

Механізми безпеки повинні враховувати розміри Інтернету: підтримувати роботу з більшою кількістю користувачів і бути простими в керуванні. Interbase пропонує кілька вирішень для побудови Інтернет-Додатків, що забезпечують безпеку доступу користувачів: поліпшену

аутентифікацію проксі-бази даних, підтримку інфраструктури відкритих ключів (Public Key Infrastructure, PKI) і нові засоби однократної наскрізної реєстрації для доступу до серверів додатків. Сукупність цих функцій дозволяє підприємствам точно ідентифікувати користувачів на всіх рівнях мережі. Робота ПЗ проміжного рівня більше не обмежується простим підключенням Web-Користувача до бази даних. Засоби аутентифікації на рівні проксі формують безліч масштабованих малих сесій для роботи з базою даних, беручи на себе навантаження по перевірці дійсності, що забезпечує тонке керування доступом і докладний аудит.

Однократна аутентифікація забезпечується компонентом Login Server для служб порталів на сервері додатків. У підсумку для доступу до всіх Web-Додаткам підприємства користувачеві буде потрібно лише один обліковий запис. Для керування повноваженнями користувачів системним адміністраторам досить підтримувати один центральний каталог LDAP (Інтернет-Каталог Interbase). Централізація відомостей про повноваження користувачів не тільки підвищує безпеку, але і знижує витрати на адміністрування. База даних Interbase пропонує механізми спільного використання апаратного і програмного забезпечення декількома групами користувачів. Роздільне зберігання даних кожної групи забезпечують Virtual Private Database і Interbase Label Security.

Крім того, СУБД Interbase можуть працювати під керуванням будь-якої операційної системи: як Windows Server, так і Unix, і інші. Це робить її більш гнучкою і адаптируємою до будь-якої інформаційної системи. Настільки розвинені засоби забезпечення надійності і безпеки також сприяють її поширеності. По статистиці, близько 53% інформаційних систем підприємств базуються на Interbase.

2.2.3 Обґрунтування вибору програмного забезпечення

Для створення інформаційної системи "Облік готової продукції" оптимальним вибором є третій варіант розглянутого програмного забезпечення – використання Microsoft Access для розробки оболонки і робочих модулів, що дозволить автоматизувати основні функції обліку складського терміналу.

2.3 Вибір апаратного забезпечення

Існує можливість вибору між готовими серверами, пропонованими виробниками і постачальниками комп'ютерної техніки, і серверами самостійного складання. При наявності певного досвіду зібраний під замовлення сервер може скласти альтернативу готовому продукту. Вибираючи сервер, не завжди варто гнатися за продуктивністю кожного з компонентів - якщо необхідно файловий сервер, йому не потрібен надпотужний процесор, і збільшити дискову підсистему. А от якщо сервер, конфігурацію якого вибираємо, багатоцільовий, тоді варто подумати саме про продуктивність процесора і обсяг оперативної пам'яті. При виборі конфігурації також корисно знати мінімальні системні вимоги програмного забезпечення, що планується встановити на сервері.

Велика розмаїтість компонентів не дає можливості назвати конкретні види апаратного забезпечення для закупівлі і збирання. Тому варто звернути увагу на наступні моменти. За завданням дипломного проекту, необхідно вибрати конфігурацію сервера відповідним заданим умовам.

– необхідний сервер із двома процесорами, що повинен забезпечити роботу певної кількості користувачів. Процесор є «серцем» будь-якої

комп'ютерної техніки. Очевидно, що від його продуктивності прямо залежить продуктивність сервера в цілому. Тому, зупинимося на моделях із двоядерним процесором Xeon;

- оскільки вимоги сучасних програм до системних ресурсів зростають, для сервера є важливим такий показник, як максимальний обсяг оперативної пам'яті - він характеризує здатність сервера до розширюваності. Обсяг оперативної пам'яті повинен бути не менш ніж 4 Гб;

- у сервері повинні використовуватися, як мінімум, вінчестери SCSI з підтримкою RAID-Технологій. Розширення можливостей сервера можливо і за рахунок організації RAID-масиву, що являє собою надлишкову безліч незалежних дисків, що забезпечують надійне зберігання даних. Для організації RAID-масиву необхідні SCSI диски і RAID контролер. Дані вінчестери являють собою ідеальний пристрій для високопродуктивного сервера, у тому числі і для мереж з інтенсивним обміном даними;

- якщо сервер буде оснащений двома або більше жорсткими дисками, необхідно подумати про його додаткове охолодження. Для цього встановлюють спеціальні вентилятори, які можна додатково встановити в системний блок;

- швидкісний привід CD-ROM не тільки заощадить час при установці ОС і прикладного ПО, але і виявиться надзвичайно корисним при роботі із централізованою довідковою системою;

- тому що всі підключені до мережі робочі станції будуть постійно звертатися до сервера, одним з його найважливіших компонентів є продуктивна 32-бітна мережна карта. Вона повинна ефективно управляти інформаційним обміном;

- продуктивність серверної плати є інтегральною характеристикою. Одним з основних показників швидкодії системної плати є тактова частота системної шини. Вона характеризує швидкість, з якої різні пристрої на

серверній платі обмінюються даними. Очевидно, чим більше цей показник - тим швидше працює плата. Тому, слідом за новими процесорами з'явилися серверні плати, що працюють на тактовій частоті 1066 МГц, які й пропоную використовувати на розроблювальному сервері. У цьому випадку оптимальним варіантом є серверна плата Intel Core 2 Duo LGA 775 1066 FSB BOX.

Для дискової підсистеми сервера основними показниками є: обсяг збережених даних, надійність і швидкість роботи вінчестерів. Для підвищення надійності в серверах використовують як мінімум два вінчестери: один - під операційну систему, іншої - під даних користувачів. Обсяг вінчестера з операційною системою вибирається виходячи із системних вимог програмного забезпечення - кожний продукт займає певну кількість дискового простору, про що завжди пишеться в супровідній документації. Надійність роботи дискової підсистеми сервера забезпечується використанням вінчестерів або безпосередньо від виробника, або марок, що зарекомендували себе. При цьому установка жорстких дисків у сервери відомих виробників має ряд особливостей, зокрема, необхідно застосовувати жорсткі диски тільки тих марок, використання яких сертифіковане виробником сервера.

Продуктивність - важливий показник роботи дискової підсистеми. Вінчестер не може працювати сам по собі, для його функціонування необхідний контролер, - пристрій, що виконує роль своєрідного перекладача між форматом даних, у якому оперує системна плата, і форматом даних вінчестера. Найбільше поширення одержали два інтерфейси: IDE і SCSI.

Крім швидкості роботи, IDE і SCSI мають і інші розходження: IDE при спілкуванні з вінчестером активно використовує ресурси процесора, у той час як SCSI роботу з жорстким диском бере на себе, тому продуктивність систем з SCSI-Контролером вище. Крім цього, SCSI дозволяє підключати більша кількість периферійних пристроїв - до 15 на

канал, а IDE - лише до 2 пристроїв на канал. Існують як багатоканальні SCSI-контролери, що дозволяють підключати більшу кількість пристроїв, так і багатоканальні IDE-контролери які через свою дешевину, дуже поширені в комп'ютерах. У серверах же, навпаки - завдяки своїй продуктивності основним інтерфейсом є SCSI.

Тому, у розроблювальному сервері, беручи до уваги вищевикладене, пропоную встановити жорсткі диски SCSI марки Segate 300Gb, що зарекомендували себе гарантованою надійністю.

Для розроблювального сервера пропоную використовувати серверний корпус Intel 6U IPC спеціально розроблений для обраної серверної плати Intel Core 2 Duo. Цей серверний корпус має ряд особливостей і переваг, представлених у таблиці 2.3:

Таблиця 2.3 - Технічні показники корпусу

Особливості	Переваги
1	2
Конфігурація з одним джерелом живлення 400 Вт або із двома (основний + резервний)	Досить для системи будь-якої складності; надійність резервування
5 відсіків для дискових пристроїв стандарту Ultra320 SCSI з підтримкою швидкості обертання дисків до 10 тис. обертів/хв.	Понад 300 Гб дисковий простір
Модифікація з можливістю "гарячої заміни" підтримує п'ять 1" SCSI пристроїв "гарячої заміни" і два пристрої, що не знімаються	Скорочує час простою, викликаного заміною дисків, що вийшли з ладу; збільшує повний розмір дискового простору.
Три відсіки 5.25" для установлення периферії	Універсальність конфігурації системи
Модифікація для установлення в стійку згідно норм і стандартів	Зручність монтажу й налагодження
Механічний захист від несанкціонованого доступу та утиліта контролю стану сервера Intel Server Control (ISC)	Забезпечення безпеки, можливість автоматичного контролю життєво важливих систем сервера, розсилання попереджень і діагностика відмов

Таким чином, виходячи з вищесказаного, сформована конфігурація сервера представлена в таблиці 2.4:

Таблиця 2.4 - Конфігурація сервера

Найменування	Тип пристрою	Кіл-В	Ціна \$
Процесор:	Intel Core 2 Duo LGA 775 2.66G/4Mb/1066 FSB BOX	1	127
Жорсткий диск:	500Gb Western Digital Buffer 16Mb Special RAID Edition	2	100
Пам'ять:	DIMM 2Gb DDR3 ECC Kingston	2	50
Плата:	Intel Core 2 Duo LGA 775 1066 FSB BOX	1	130
Привод DVD-RW:	ND-3550 Dual Layer, DVD 16x/8x/16x, CD 48x/32x/48x	1	40
Контролер RAID:	інтегрований	1	179
Корпус серверний:	Intel 6U IPC CASE w/ SAFETY P4	1	110
Мережна карта:	GEMBIRD PCMCIA Card 10/100 32bit	1	15
Відеоадаптер:	інтегрований		
Разом:			901

3 ПРОЕКТУВАННЯ "ІНФОРМАЦІЙНОЇ СИСТЕМИ КОМПЛЕКСУ СКЛАДСЬКИХ ТЕРМІНАЛІВ"

3.1 Розроблення алгоритму функціонування

Функціонування ІС здійснюється згідно зі структурним алгоритмом, побудованим виходячи з вимог, пропонованих до виконуваних програмою функціям (розділи 1.3 і 2.1). Роботу всієї системи можна представити алгоритмом, зображеним на рис. 3.1.

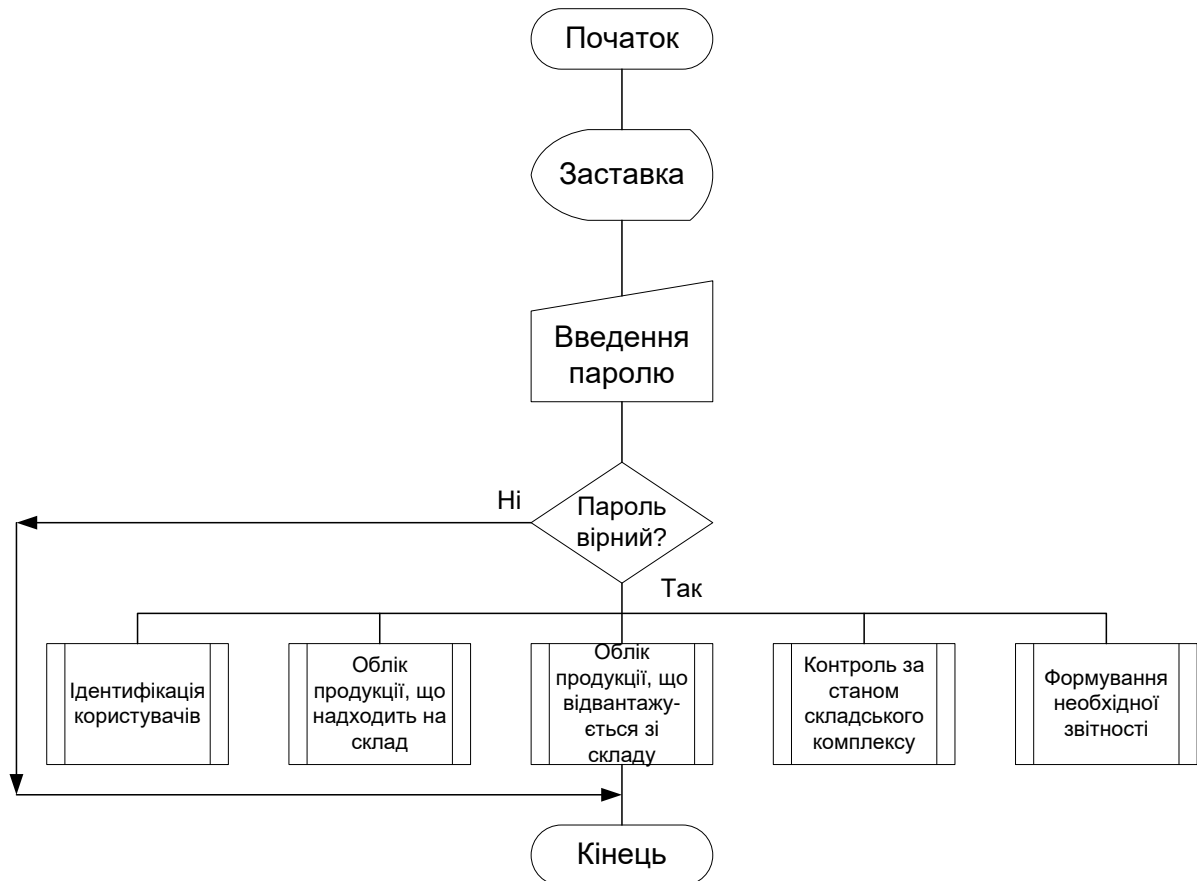


Рисунок 3.1 – Схема алгоритму роботи інформаційної системи

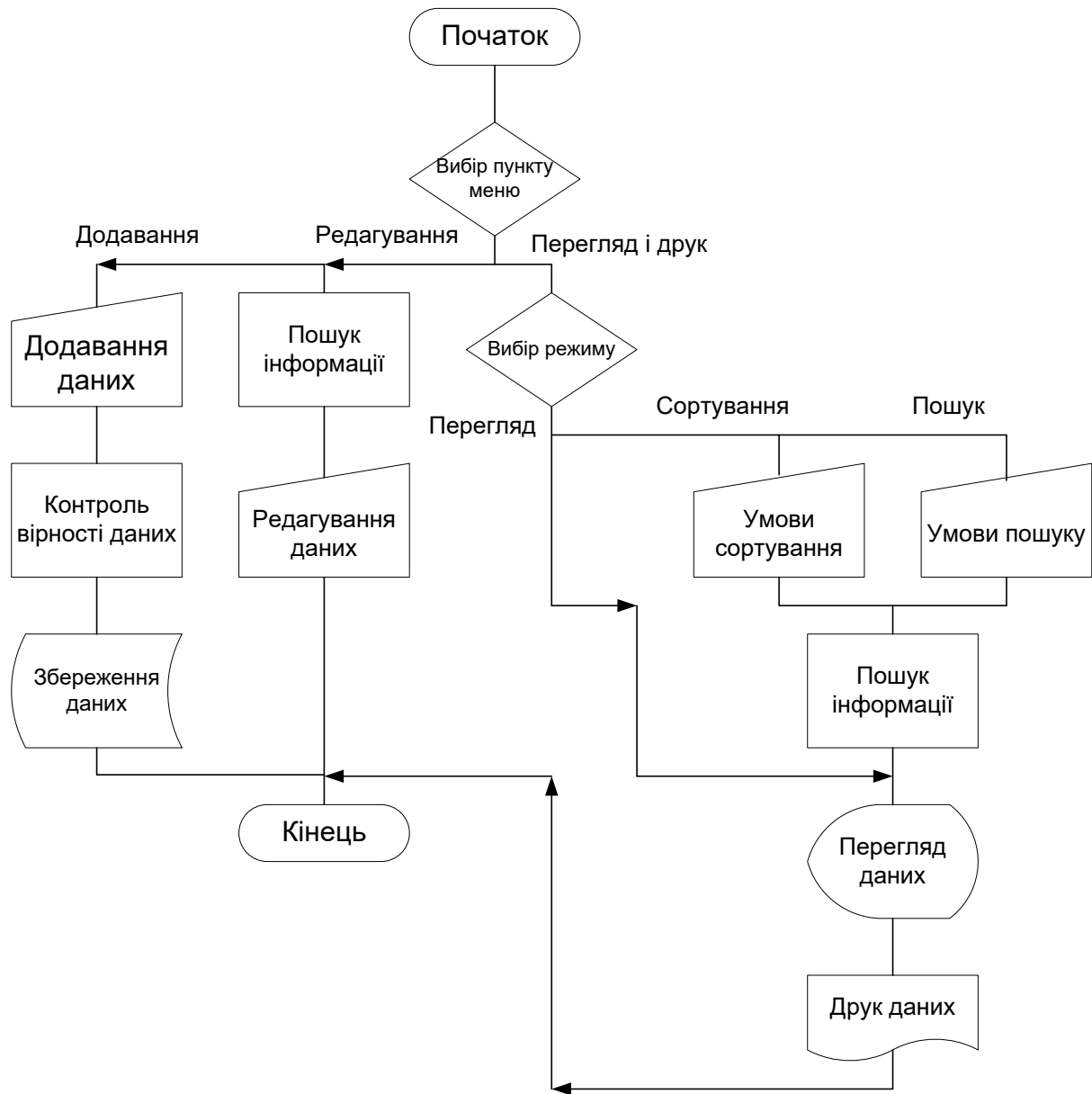


Рисунок 3.2 – Схема алгоритму обробки даних користувача

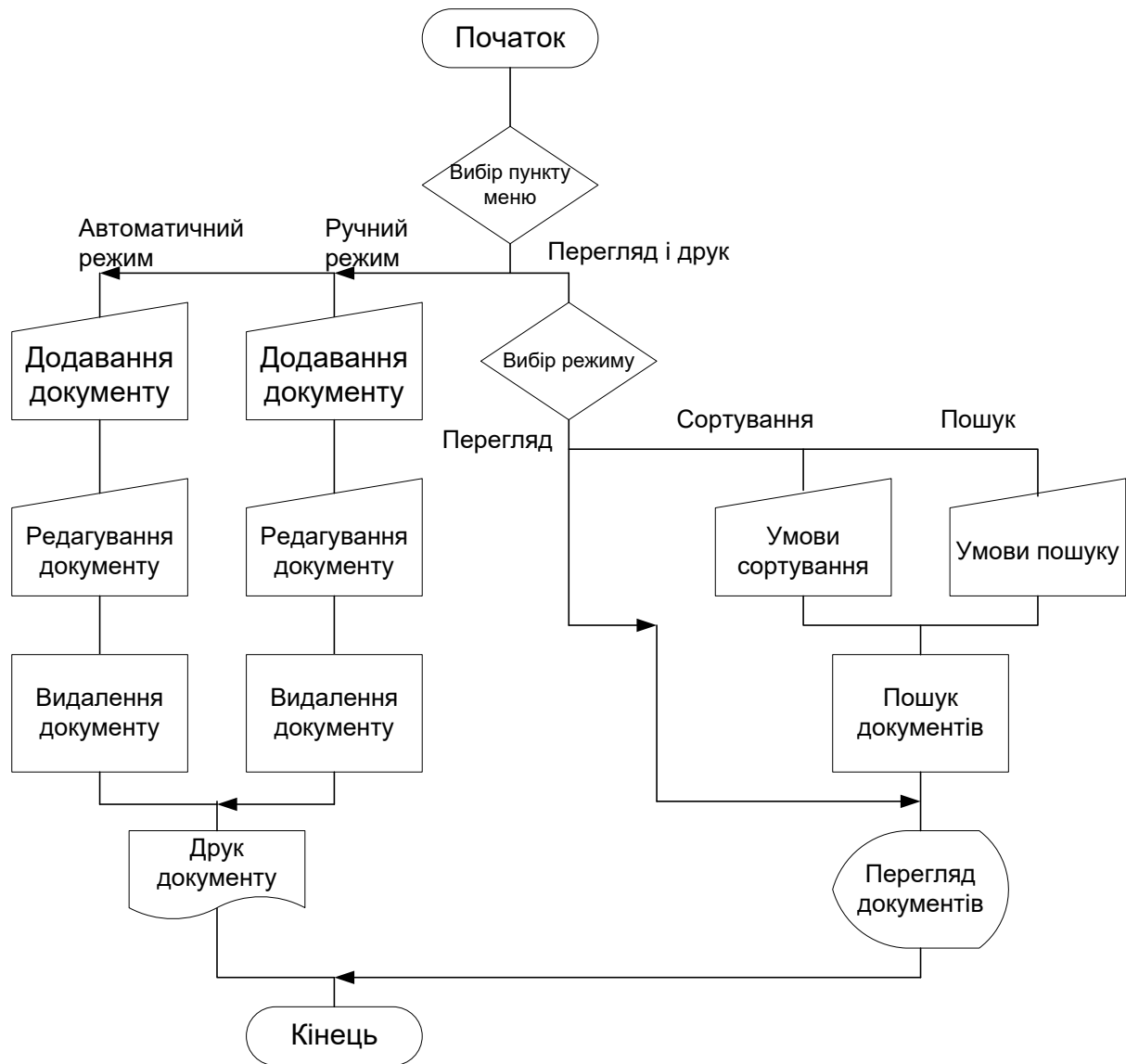


Рисунок 3.3 – Схема алгоритму роботи з документами



Рисунок 3.4 – Схема алгоритму контролю стану складського комплексу

3.2 Інформаційна структура бази даних

Проектування бази даних починається з виявлення атрибутів і підбора даних. Проектована база даних буде містити об'єктне відношення документів приходу та об'єктне відношення документів відвантаження зі

складу. Аналіз інформації, яка повинна втримуватися в акті про прихід продукції на склад, показує, що слід виділити наступні атрибути об'єктного відношення документів приходу:

- 1) № акту про розвантаження;
- 2) оператор, що робить приймання продукції на склад (зав. складом);
- 3) № товарно-транспортної накладної, по якій продукція прибула на склад;
- 4) дата створення акту про розвантаження;
- 5) час створення акту про розвантаження;
- 6) № машини, з якої прибула продукція;
- 7) постачальник продукції;
- 8) водій машини;
- 9) дата розвантаження;
- 10) час розвантаження;
- 11) код продукту;
- 12) найменування продукту;
- 13) строк придатності продукту;
- 14) кількість коробів продукції;
- 15) вага короба продукції;
- 16) ціна короба продукції;
- 17) адреса розвантаженої продукції на складі.

Дане об'єктне відношення також повинне містити інформацію про постачальника продукції (адреса, телефон і т.д.). Використовуючи дане об'єктне відношення, ми одержимо занадто громіздку базу даних, з величезною надмірністю. Тому що прийнята продукція буде мати певну кількість різних адрес на складі для кожного коду продукції окремо, то ми одержимо велику кількість рядків, у яких буде повторюватися інформація про постачальників, продукцію, операторів. Виходячи з даного аналізу доцільно буде розбити об'єктне відношення документів приходу на кілька

окремих об'єктних відносин: документи приходу, картка товару, постачальники, оператори, розташування.

Визначимо атрибути об'єктного відношення «Картка товару»:

- 1) найменування товарної одиниці;
- 2) виробник товарної одиниці;
- 3) код продукту;
- 4) вага короба продукції;
- 5) висота короба продукції;
- 6) ширина короба продукції;
- 7) довжина короба продукції;
- 8) ціна короба продукції.

Визначимо атрибути об'єктного відношення «Постачальники»:

- 1) код постачальника;
- 2) назва постачальника;
- 3) адреса постачальника;
- 4) телефон постачальника;
- 5) розрахунковий рахунок постачальника;
- 6) № договору з постачальником.

Визначимо атрибути об'єктного відношення «Оператори»:

- 1) прізвище оператора;
- 2) ім'я оператора;
- 3) по батькові оператора;
- 4) адреса оператора;
- 5) телефон оператора.

Визначимо атрибути об'єктного відношення «Документи приходу»:

- 1) № акту розвантаження;
- 2) оператор;
- 3) № товарно-транспортної накладній;
- 4) час створення акту розвантаження;
- 5) дата створення акту розвантаження;

- 6) № машини, з якої прибула продукція;
- 7) постачальник;
- 8) водій машини;
- 9) дата розвантаження;
- 10) час розвантаження.

Визначимо атрибути об'єктного відношення «Розташування»:

- 1) № акту розвантаження;
- 2) код продукту;
- 3) кількість коробів;
- 4) строк придатності продукції;
- 5) адреса.

Інформація про товари буде розташовуватися у таблиці з іменем "tovar" з наступною структурою (таблиця 3.1.):

Таблиця 3.1 – Структура таблиці "tovar"

Назва	Ім'я поля	Тип поля	Довжина
Назва товару	Naim_tov	текстовий	30
Виробник товарної одиниці	Naim_proizvod	текстовий	15
Код продукту	Kod_prod	числовий	6
Вага короба продукції	Ves_prod	числовий	4
Ширина короба продукції	Shir_prod	числовий	3
Висота короба продукції	Visot_prod	числовий	3
Довжина короба продукції	Dlin_prod	числовий	3
Ціна короба продукції	Cena_prod	числовий	4

Інформація про постачальників буде розташовуватися у таблиці з іменем "postav" з наступною структурою (таблиця 3.2):

Таблиця 3.2 – Структура таблиці "postav"

Назва	Ім'я поля	Тип поля	Довжина
код постачальника	Kod_post	числовий	5
назва постачальника	Naim_post	текстовий	15
адреса постачальника	Adres_post	текстовий	30

телефон постачальника	Telef_post	числовий	6
розрахунковий рахунок постачальника	Ras_shet	числовий	30
№ договору з постачальником	№_dogov	числовий	10

Інформація про операторів буде розташовуватися у таблиці з іменем "operators" з наступною структурою (таблиця 3.3):

Таблиця 3.3 – Структура таблиці "operators"

Назва	Ім'я поля	Тип поля	Довжина
Прізвище оператора	FIO1_oper	текстовий	10
Ім'я оператора	FIO2_oper	текстовий	8
По батькові оператора	FIO3_oper	текстовий	10
Адреса оператора	Adres_oper	текстовий	30
Телефон оператора	Telef_oper	числовий	6

Інформація про документи приходу буде розташовуватися у таблиці з іменем "prihod" з наступною структурою (таблиця 3.4):

Таблиця 3.4 – Структура таблиці "prihod"

Назва	Ім'я поля	Тип поля	Довжина
№ акту розвантаження	№_akt	числовий	10
Оператор	operator	текстовий	10
№ товарно-транспортної накладної	№_TTN	числовий	5
Час створення акту про розвантаження	Time	time	8
Дата створення акту про розвантаження	Data	data	10
№ машини, з якої прибула продукція	№_cars	загальний	10
Код постачальник	Kod_post	текстовий	15
Водій машини	Voditel	текстовий	10
Дата розвантаження	Data1	data	10
Час розвантаження	Time1	time	8

Інформація про розташування буде розташовуватися у файлі з іменем "adress" з наступною структурою (таблиця 3.5):

Таблиця 3.5 – Структура таблиці "adress"

Назва	Ім'я поля	Тип поля	Довжина
№ акту розвантаження	№_acts	числовий	10
Код продукту	Kod_prod	числовий	6
Кількість коробів	Kol_case	числовий	3
Строк придатності продукції	BBD	загальний	15
Адреса	Adress	загальний	15

Визначимо необхідні атрибути об'єктного відношення документів відвантаження. Аналіз інформації, яка повинна втримуватися в акті про відвантаження продукції зі складу, показує, що слід виділити наступні атрибути об'єктного відношення документів відвантаження зі складу:

- 1) № акту про відвантаження;
- 2) № замовлення;
- 3) оператор;
- 4) час створення акту;
- 5) дата створення акту;
- 6) код клієнта;
- 7) назва клієнта;
- 8) адреса клієнта;
- 9) телефон клієнта;
- 10) дата відвантаження;
- 11) час відвантаження;
- 12) код продукції;
- 13) адреса продукції на складі;
- 14) кількість коробів;
- 15) строк придатності продукції.

Використовуючи дане об'єктне відношення, ми одержимо занадто громіздку базу даних, з величезною надмірністю. Тому що продукція, що відвантажувється, буде мати певну кількість різних адрес на складі для кожного коду продукції окремо, то ми одержимо велику кількість рядків, у

яких буде повторюватися інформація про клієнтів, продукцію, операторів. Виходячи з даного аналізу доцільно буде розбити об'єктне відношення документів відвантаження на кілька окремих об'єктних відносин: документи відвантаження, картка товару, клієнти, оператори, адреса відвантаження.

Об'єктні відносини картка товару і оператори представлені вище. Визначимо атрибути об'єктного відношення «Клієнти»:

- 1) код клієнта;
- 2) назва клієнта;
- 3) адреса клієнта;
- 4) телефон клієнта.

Визначимо атрибути об'єктного відношення «Документи відвантаження»:

- 1) № акту відвантаження;
- 2) № замовлення;
- 3) оператор;
- 4) час створення акту відвантаження;
- 5) дата створення акту відвантаження;
- 6) код клієнта;
- 7) дата відвантаження;
- 8) час відвантаження.

Визначимо атрибути об'єктного відношення «адреса відвантаження»:

- 1) код продукту;
- 2) кількість коробів;
- 3) строк придатності продукції;
- 4) адреса продукції.

Інформація про клієнтів буде розташовуватися у таблиці з іменем "klient" з наступною структурою (таблиця 3.6):

Таблиця 3.6 – Структура таблиці "klient"

Назва	Ім'я поля	Тип поля	Довжина
код клієнта	Kod_klien	числовий	5
назва клієнта	Naim_klien	текстовий	15
адреса клієнта	Adres_klien	текстовий	30
телефон клієнта	Telef_klien	числовий	6

Інформація про документи відвантаження буде розташовуватися у таблиці з іменем "otgryska" з наступною структурою (таблиця 3.7):

Таблиця 3.7 – Структура таблиці "otgryska"

Назва	Ім'я поля	Тип поля	Довжина
№ акту відвантаження	№_akt1	числовий	10
№ замовлення	№_zakaz	числовий	10
Оператор	operator	текстовий	10
Час створення акту про відвантаження	Time2	time	8
Дата створення акту про відвантаження	Data2	data	10
Код клієнта	Kod_klien	загальний	5
Дата відвантаження	Data3	data	10
Час відвантаження	Time3	time	8

Інформація про розташування буде розташовуватися у таблиці з іменем "adress1" з наступною структурою (таблиця 3.8):

Таблиця 3.8 – Структура таблиці "adress1"

Назва	Ім'я поля	Тип поля	Довжина
№ акту відгрузки	№_acts	числовий	10
Код продукту	Kod_prod	числовий	6
Кількість коробів	Kol_case	числовий	3
Строк придатності продукції	BBD	загальний	15
Адреса	Adress	загальний	15

Інфологічна модель бази даних "Прихід", "Догляд" побудована за допомогою мови "таблиці-зв'язку" представлена на рис. 3.5.

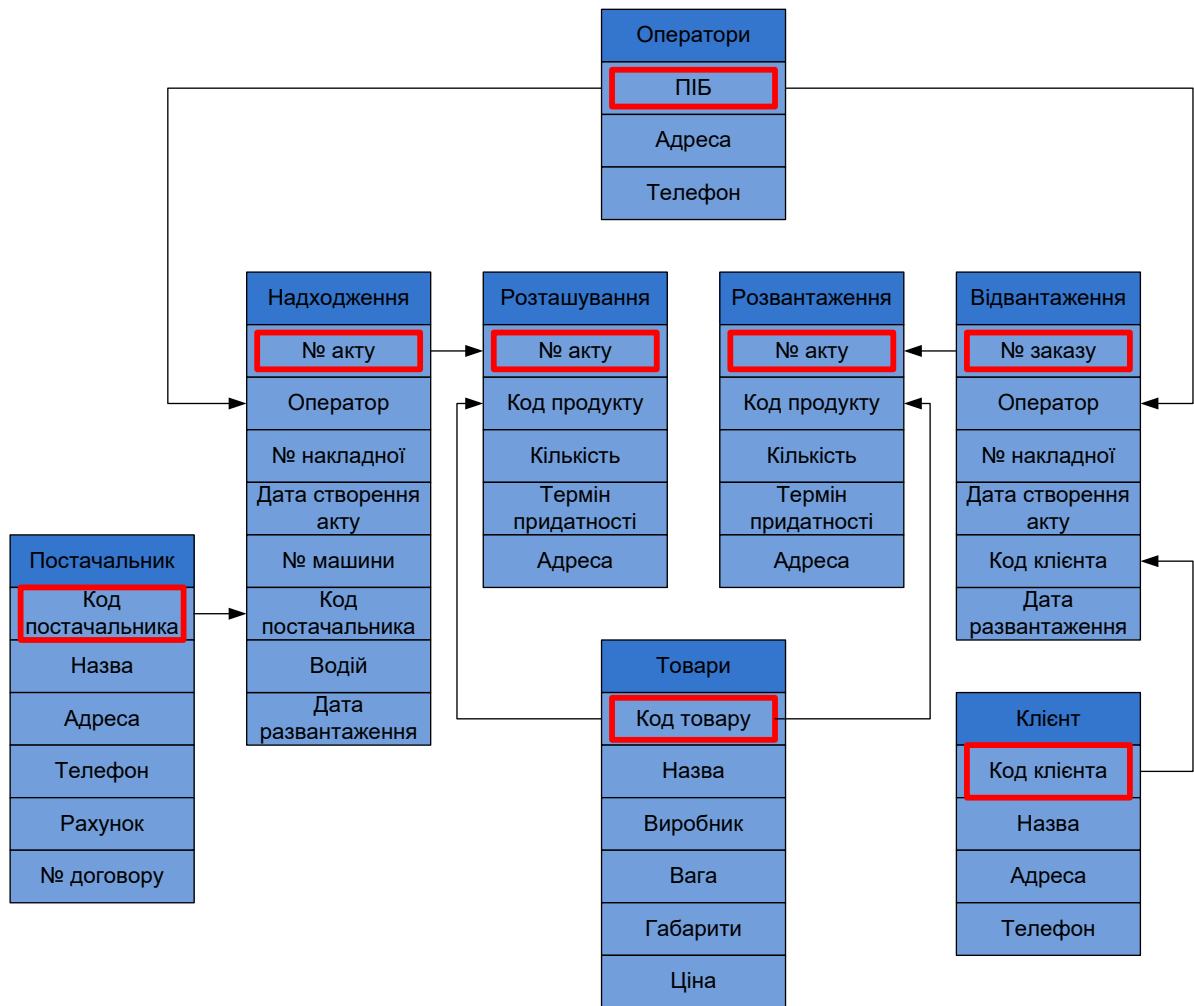


Рисунок 3.5 – Інфологічна модель бази даних

4 ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ

4.1 Аналіз потенційних небезпечних і шкідливих виробничих чинників проєктованого об'єкту, що мають вплив на персонал

У даному дипломному проєкті розробляється програмне забезпечення.

Розроблене програмне забезпечення орієнтоване на роботу з персональним комп'ютером. Експлуатовані для вирішення внутрішньовиробничих завдань ПЕОМ типу IBM PC мають наступні характеристики:

споживана потужність	220 Вт;
робоча напруга	220 В;
напруга джерел живлення	+12 В; - 12 В; +5 В;
робоча частота	50 Гц.

Виходячи з приведених характеристик, вочевидь, що для людини існує небезпека поразки електричним струмом, унаслідок недбалого поводження з комп'ютером і порушення правил експлуатації, залишення частин ПЕОМ, що знаходяться під напругою, відкритими або знятих для ремонту вузлів.

Відповідно до ГОСТ 12.1.005-88 до легкої фізичної роботи відносяться всі види діяльності, виконувані сидячи і ті, що не потребують фізичної напруги. Робота користувача ПК відноситься до категорії 1а.

При роботі на ПЕОМ користувач піддається ряду потенційних небезпек. Унаслідок недотримання правил техніки безпеки при роботі з машиною (невиконання огляду відкритих частин ПЕОМ, що знаходяться під напругою або знятих для ремонту вузлів) для користувача існує небезпека поразки електричним струмом.

Джерелами підвищеної небезпеки можуть служити наступні елементи:

- розподільний щит;
- джерела живлення;
- блоки ПЕОМ і друку, що знаходяться в ремонті.

Ще одна проблема полягає у тому, що спектр випромінювання комп'ютерного монітора включає рентгенівську, ультрафіолетову і інфрачервону області, а також широкий діапазон хвиль інших частот. Небезпека рентгенівського проміння мала, оскільки цей вид випромінювання поглинається речовиною екрану. Проте велику увагу слід приділяти біологічним ефектам низькочастотних електромагнітних полів(аж до порушення ДНК).

Відповідно до ГОСТ 12.1.003-74, при обслуговуванні ПЕОМ мають місце фізичні і психофізичні небезпечні, а також шкідливі виробничі чинники:

- підвищене значення напруги в електричному ланцюзі, замикання якої може відбутися через тіло людини;
- підвищений рівень статичної електрики;
- підвищений рівень електромагнітних випромінювань;
- підвищена або знижена температура повітря робочої зони;
- підвищений або знижений рух повітря;
- підвищена або знижена вологість повітря;
- відсутність або недостатність природного світла;
- підвищена пульсація світлового потоку;
- недостатня освітленість робочого місця;
- підвищений рівень шуму на робочому місці;
- розумове перенапруження;
- емоційні навантаження;
- монотонність праці.

4.2 Заходи щодо техніки безпеки

Основним небезпечним чинником при роботі з ЕОМ є небезпека поразки людини електричним струмом, яка посилюється тим, що органи чуття людини не можуть на відстані знайти наявності електричної напруги на устаткуванні.

Проходячи через тіло людини, електричний струм чинить на нього складну дію, що є сукупністю термічної(нагрів тканин і біологічних середовищ), електролітичної(розкладання крові і плазми) і біологічної(роздратування і збудження нервових волокон і інших органів тканин організму) дій.

Тяжкість поразки людини електричним струмом залежить від цілого ряду чинників:

- значення сили струму;
- електричного опору тіла людини і тривалості протікання через нього струму;
- роду і частоти струму;
- індивідуальних властивостей людини і навколишнього середовища.

Розроблений дипломний проект передбачає наступні технічні способи і засоби, що застерігають людину від ураження електричним струмом:

- заземлення електроустановок;
- занулення;
- захисне відключення;
- електричне розділення ятерів;
- використання малої напруги;
- ізоляція частин, що проводять струм;
- огорожа електроустановок.

Занулення зменшує напругу дотику і обмежує година, протягом якого людина, ткнувшись до корпусу, може потрапити під дію напруги.

Струм однофазного короткого замикання визначається по наближеній формулі:

$$I_K = \frac{U_{\phi}}{Z_{\Pi} + \frac{Z_T}{3}}, \quad (4.1)$$

де U_{ϕ} - номінальна фазна напруга мережі, В;

Z_{Π} - повний опір петлі, створене фазними і нульовими дротами, Ом;

Z_T - повний опір струму короткого замикання на корпус, Ом.

Згідно таблиці 4 ГОСТ 12.1.009-76: $Z_T/3 = 0,1$ Ом.

Для провідників і жил кабелю для розрахунку повного опору петлі використовуємо формулу(4.2.) :

$$Z_{\Pi} = \sqrt{R_{\Pi}^2 + X_{\Pi}^2}, \quad (5.2)$$

де $R_{\Pi} = R_{\phi} + R_o$ - сумарний активний опір фазного R_{ϕ} і нульового R_o дротів, Ом;

X_{Π} - індуктивний опір паяння дротів, Ом.

Перетин 1 км мідного дроту $S = 2.5$ мм, тоді згідно таблицям 5 і 6 ГОСТ 12.1.009-76 [13], має такий опір:

$$X_{\Pi} = 0,11 \text{ Ом};$$

$$R_{\phi} = 7,55 \text{ Ом};$$

$$R_o = 7,55 \text{ Ом}.$$

$$\text{Отже, } R_{\Pi} = 7,55 + 7,55 = 15,1 \text{ Ом}.$$

Тоді по формулі (4.2) знаходимо повний опір петлі :

$$Z_{\Pi} = \sqrt{15,1^2 + 0,11^2} \approx 15,1 \text{ (Ом)}.$$

Струм однофазного короткого замикання рівний:

$$I_k = \frac{220}{15,1 + 0,1} = 14,47 \text{ (A)}.$$

Дія плавкої вставки на ПЕОМ забезпечується, якщо виконується співвідношення:

$$I_k \geq k * I_n, \quad (4.3)$$

де I_n - номінальний струм спрацьовування плавкої вставки, А;

k - коефіцієнт кратності нелінійного струму I_n , А.

Коефіцієнт кратності нелінійного струму I_n розраховується по формулі (4.4.) :

$$I_n = P / U, \quad (4.4)$$

де $P = 220$ Вт - споживана потужність;

$U = 220$ В - робоча напруга;

$k = 3$ А - для плавких вставок.

Отже, $I_n = 220 / 220 = 1$ А.

Підставивши значення у вираз (4.3), одержимо:

$$14,47 > 3 * 1.$$

Таким чином, доведено, що апарат забезпечить спрацьовування(і захист) при підвищенні номінального струму.

4.3 Рекомендації по пожежній безпеці

Пожежі в приміщеннях, де встановлена обчислювальна техніка, представляють небезпеку для життя людини. Пожежі також пов'язані як з матеріальними втратами, так і з відмовою засобів обчислювальної техніки, що у свою чергу спричиняє за собою порушення ходу технологічного процесу.

Пожежа може виникнути при наявності горючої речовини та внесення джерела запалювання в горюче середовище. Пальними матеріалами в приміщеннях, де розташовані ПЕОМ, є:

- поліамід - матеріал корпусу мікросхеми, горюча речовина, температура самозаймання аерогелю 420 З ;
- полівінілхлорид - ізоляційний матеріал, горюча речовина, температура запалювання 335 З, температура самозаймання 530 З, кількість енергії, що виділяється при згоранні - 18000 - 20700 кДж/кг;
- стеклотекстоліт ДЦ - матеріал друкарських плат, важкозаймистий матеріал, показник горючості 1.74, не схильний до температурного самозаймання;
- пластика кабельний №489 - матеріал ізоляції кабелю, горючий матеріал, показник горючості більш 2.1;
- деревина - будівельний і обробний матеріал, матеріал з якого виготовлені меблі, горючий матеріал, показник горючості більше 2.1, теплота згорання 18731 - 20853 кДж/кг, температура запалювання 399 З, схильна до самозаймання [17].

Згідно НАПБ.Б. 03.002-2007 приміщення відносяться до категорії В(пожежовибухонебезпечним) і згідно правилам побудови електроустановок простір усередині приміщення відноситься до вогнебезпечної зони класу П - Па (зони, розташовані в приміщеннях, в яких зберігаються тверді горючі речовини).

Потенційними джерелами запалення при роботі ПЕОМ є:

- іскри при замиканні і розмиканні ланцюгів;
- іскри і дуги коротких замикань;
- перегріву від тривалого перевантаження і наявності перехідного опору.

Продуктами згорання, що виділяються при пожежі, є : оксид вуглецю, сірчистий газ, оксид азоту, синильна кислота, акропеїн, фосген, хлор та ін. При горінні пластмас, окрім звичайних продуктів згорання, виділяються різні продукти термічного розкладання: хлорангідридні кислоти, формальдегіди, хлористий водень, фосген, синильна кислота, аміак, фенол, ацетон, стирол та ін., що шкідливо впливають на організм людини.

Для захисту персоналу від дії небезпечних і шкідливих чинників пожежі проектом передбачається застосування промислового протигаза з коробкою марки В(жовта).

Пожежна безпека об'єктів народного господарства регламентується ГОСТ 12.1.004-91 і забезпечується системами запобігання пожежам і протипожежному захисту. Для успішного гасіння пожеж вирішальне значення має швидке виявлення пожежі і своєчасний виклик пожежних підрозділів до місця пожежі.

Зменшити горюче навантаження не представляється можливим, тому проектом передбачається застосувати наступні способи і їх комбінації для запобігання утворенню(внесення) джерел запалення :

- застосування устаткування, що задовольняє вимогам електростатичної безпеки;
- застосування в конструкції швидкодіючих засобів захисного відключення можливих джерел запалення;
- виключення можливості появи іскрового заряду статичної електрики в горючому середовищі з енергією, рівної і вище мінімальної енергії запалення;

- підтримка температури нагріву поверхні машин, механізмів, устаткування, пристроїв, речовин і матеріалів, які можуть увійти до контакту з палим середовищем, нижче гранично допустимої, становить 80% якнайменшої температури самозаймання пального.

- заміна небезпечних технологічних операцій більш безпечними;
- ізолюване розташування небезпечних технологічних установок і устаткування;

- зменшення кількості палих і вибухонебезпечних речовин, що знаходяться у виробничих приміщеннях;

- запобігання можливості утворення палих сумішей на лінії, вентиляційних системах і ін.;

- механізація, автоматизація та справність(потокова) виробництва;

- суворе дотримання стандартів і точне виконання встановленого технологічного режиму;

- запобігання можливості появи в небезпечних місцях джерел запалення;

- запобігання розповсюдженню пожеж і вибухів;

- використання устаткування і пристроїв, при роботі яких не виникає джерел запалення;

- виконання вимог сумісного зберігання речовин і матеріалів;

- наявність громовідводу;

- ліквідація можливості самозаймання речовин і матеріалів .

Для запобігання пожежі в обчислювальних центрах проектом пропонується виконання наступних вимог :

- електроживлення ЕОМ повинно мати автоматичне блокування відключення електроенергії на випадок зупинки системи охолодження і кондиціонування;

- система вентиляції обчислювальних центрів повинна бути обладнана блокуючими пристроями, що забезпечують її відключення на випадок пожежі;

– робочі місця повинні бути оснащені пожежними щитами, сигналізацією, засобами для сповіщення про пожежну небезпеку (телефонами), медичними аптечками для надання першої медичної допомоги, розробленим планом евакуації.

Для зниження пожежної небезпеки в приміщеннях використовуються первинні засоби гасіння пожеж, а також система автоматичної пожежної сигналізації, яка дозволяє знайти початкову стадію загоряння, швидко і точно оповістити службу пожежної охорони про час і місце виникнення пожежі.

Відповідно до правил пожежної безпеки для промислових підприємств приміщення категорії В підлягають устаткуванню системами автоматичної пожежної сигналізації. Проектом передбачається застосування датчика типу ІДФ - 1(димовий фотоелектричний датчик), оскільки специфікою пожеж обчислювальної техніки і радіоапаратури є, в першу чергу, виділення диму, а потім - підвищення температури.

При виникненні пожежі в робочому приміщенні обслуговуючий персонал зобов'язаний негайно вжити заходи по ліквідації пожежі. Для ліквідації пожежі використовують вогнегасники (хімічно-пінні, пінні для повітря ОП-5, ОП-6, ОП-9, вуглекислотні ОУ-5), пісок, пожежний інвентар(сокири, ломы, багри, шерстяну або азбестову ковдри). Як засіб індивідуального захисту проектом передбачається використання промислового протигаза з маскою, фільтруючої коробки В.

В якості організаційно-технічних заходів рекомендується проводити навчання робочого персоналу правилам пожежної безпеки.

У розділі «Охорона праці» виконано аналіз потенційних небезпек при роботі із засобами обчислювальної техніки і механізмами, розроблені заходи щодо техніки безпеки, заходи, які забезпечують виробничу санітарію і гігієну праці, виконані рекомендації по пожежній безпеці.

ВИСНОВКИ

У дипломному проекті представлена розробка інформаційної системи складського термінала. Впровадження даного проекту дозволить у значній мірі підвищити якість і продуктивність роботи персоналу складського термінала.

У дипломному проекті розроблений алгоритм роботи інформаційної системи в цілому та окремих її підсистем, розроблена база даних інформаційної системи з теоретичним обґрунтуванням на основі правил ER – декомпозиції.

У розділі «Охорона праці» виконаний аналіз небезпечних і шкідливих виробничих факторів, причин пожеж. На основі аналізу розроблені заходи щодо техніки безпеки і рекомендації з пожежної профілактики. Виконаний розрахунок захисного заземлення, розрахунок кількості світильників у приміщенні, імовірності виникнення пожежі при виникненні короткого замикання від транзистора блоку живлення монітора.

Розроблена інформаційна система складського комплексу задовольняє всім вимогам технічного завдання.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

- 1) А.А. Смехов Введение в логику: - М: Транспорт, 1993. – 110 с.
- 2) Б.А. Аникин Логистика: Учебник. – М.: ИНФРА-М, 2000. – 352 с.
- 3) М. Гук Аппаратные средства IBM PC: Энциклопедия. – СПб: Питер Ком, 1999. – 816 с.
- 4) А.Юдин. "Концепции и руководство по планированию Microsoft Windows NT Server".
- 5) В.В. Кириллов CitForum «Основы проектирования реляционных баз данных»
- 6) Дейт К. Дж. Введение в системы баз данных.: Пер. с. англ. – 6–е изд.– К.: Диалектика, 1998. – 784 с.: ил.
- 7) Джексон Г. Проектирование реляционных БД для использования с микроЭВМ.–М.: МИР, 1991.– 252 с.
- 8) Михеева В.Д., Харитонов И.А. Microsoft Access 2002. – СПб.: БХВ – Петербург, 2007. – 1040 с.
- 9) Охрана труда в вычислительных центрах. Ю.Г. Сибаров, Н.Н, Сколотнев, В.К. Васин и др. – Машиностроение, 1990 – 192 с.
- 10) ГОСТ 12.1.003-83. ССБТ. Шум. Общие требования безопасности.
- 11) ГОСТ 10.1.004-85. ССБТ. Пожарная безопасность. Общие требования.
- 12) ГОСТ 12.1.005-88. ССБТ. Общие санитарно-гигиенические правила к воздуху рабочей зоны.
- 13) ГОСТ 12.1.006-76. ССБТ. Электромагнитные поля радиочастот. Общие требования безопасности.
- 14) ГОСТ 12.1.009-76. ССБТ. Электробезопасность. Требования и определения.
- 15) ГОСТ 12.1.004-91. ССБТ. Пожарная безопасность. Общие требования.

16) ГОСТ 12.1.019-79. Электробезопасность. Общие требования и номенклатура видов защиты.

17) ГОСТ 12.1.029-80. ССБТ. Средства защиты от шума. Классификация.

18) ГОСТ 12.1.030-81. ССБТ. Электробезопасность. Защитное заземление, зануление.

19) СНиП 23-05-95. Естественное и искусственное освещение. Нормы проектирования. – М.:Стройиздат, 1980 г.

20) СнпП 2.04.05.-86. Отопление, вентиляция и кондиционирование. – М.: ЦИТП Госстроя СССР, 1987 г.

21) ГОСТ 12.0.002-80. ССБТ. Основные понятия. Термины и определения