

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»

Інститут інформатики та радіоелектроніки,
Факультет комп'ютерних наук і технологій
(повне найменування інституту, назва факультету)

Кафедра програмних засобів
(повне найменування кафедри)

Пояснювальна записка

до дипломного проєкту (роботи)
магістр
(ступінь вищої освіти)

на тему ДОСЛІДЖЕННЯ ТА ПРОГРАМНА РЕАЛІЗАЦІЯ МОДУЛІВ
ERP-СИСТЕМИ
RESEARCH AND SOFTWARE IMPLEMENTATION OF ERP SYSTEM
MODULES

Виконав: студент(ка) 2 курсу, групи КНТ-219м
Спеціальності 122 Комп'ютерні науки
(код і найменування спеціальності)

Освітня програма (спеціалізація)
Системи штучного інтелекту

Музика О.І.
(прізвище та ініціали)

Керівник Олійник А.О.
(прізвище та ініціали)

Рецензент Скрупський С.Ю.
(прізвище та ініціали)

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»
(повне найменування закладу вищої освіти)

Інститут, факультет ІРЕ, ФКНТ
Кафедра програмних засобів
Ступінь вищої освіти магістр
Спеціальність 122 Комп'ютерні науки
(код і найменування)
Освітня програма (спеціалізація) Системи штучного інтелекту
(назва освітньої програми (спеціалізації))

ЗАТВЕРДЖУЮ

Завідувач кафедри ПЗ, д.т.н, проф.
С.О. Субботін
“ ” 20__ року

З А В Д А Н Н Я

НА ДИПЛОМНИЙ ПРОЄКТ (РОБОТУ) СТУДЕНТА(КИ)

Музики Олександра Ігоровича
(прізвище, ім'я, по батькові)

1. Тема проєкту (роботи) Дослідження та програмна реалізація модулів ERP - системи. Research and Software Implementation of ERP System Modules

керівник проєкту (роботи) Олійник Андрій Олександрович, к.т.н., доцент,
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом закладу вищої освіти від “03” листопада 2020 року № 301

2. Строк подання студентом проєкту (роботи) 10 грудня 2020 року

3. Вихідні дані до проєкту (роботи) рекомендована література, технічне завдання

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) 1. Аналіз предметної області. 2. Вибір та обґрунтування структури системи, яка проєктується. 3. Основні рішення щодо реалізації компонентів системи. 4. Керівництво програміста. 5. Керівництво оператора. 6. Економіко-організаційна частина. 7. Охорона праці та безпека в надзвичайних ситуаціях.

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень) Слайди презентації

6. Консультанти розділів проєкту (роботи)

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	прийняв виконане завдання
1-6 Основна частина	Олійник А.О., доцент		
Економіко-організаційна частина	Пожуєва Т.О., професор		
Охорона праці та безпека в надзвичайних ситуаціях	Коробко О.В., ст. викладач		
Нормоконтролер	Калініна М.В., асистент		

7. Дата видачі завдання “ 02 ” жовтня 2020 року.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного проєкту (роботи)	Строк виконання етапів проєкту (роботи)	Примітка
1	Аналіз предметної області	1 тиждень	
2	Розробка та затвердження технічного завдання	2 тиждень	Технічне завдання
3	Проектування структури системи	3–4 тиждень	Структурна схема
4	Розробка схеми функціонування системи та модулів програми	5–6 тиждень	Функціональна схема. Модулі
5	Створення програмного коду системи	7–8 тижні	Текст програми
6	Тестування та відлагодження програми	9 тиждень	Працездатна система
7	Розробка розділів пояснювальної записки	10 тиждень	ПЗ
8	Розробка слайдів презентації.	11–12 тиждень	Слайди презентації
9	Захист роботи.	13 тиждень	

Студент(ка)

(підпис)

Музика О.І.

(прізвище та ініціали)

Керівник проєкту (роботи)

(підпис)

Олійник А.О.

(прізвище та ініціали)

РЕФЕРАТ

Пояснювальна записка до дипломної кваліфікаційної роботи магістра:
162 с., 27 рис., 4 додатки, 32 джерела.

ОПТИМІЗАЦІЯ, ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ, СИСТЕМА
КЕРУВАННЯ БАЗАМИ ДАНИХ, ERP СИСТЕМА, SAP R/3, SQL.

Об'єкт дослідження – процес програмної реалізації складних систем.

Предмет дослідження – програмне забезпечення ERP систем.

Мета роботи – дослідження ERP систем та програмна реалізація модулів ERP системи корпоративного підприємства.

Матеріали, методи та технічні засоби: структурне та об'єктно-орієнтоване програмування, СКБД Progress, IBM-сумісний комп'ютер з монітором, клавіатурою та маніпулятором типу «миша», що функціонує під управлінням операційної системи Microsoft Windows.

Результати. Створено модель модуля ERP системи територіально розподіленого підприємства. Розроблено програмні модулі ERP системи територіально розподіленого підприємства, що забезпечує підтримку внутрішніх бізнес-процесів підприємства та що надає можливості для взаємодії з зовнішніми об'єктами.

Висновки. Розроблено програмні модулі ERP системи територіально розподіленого підприємства. Проаналізовано предметну область, розроблено технічне завдання для реалізації програми, обрано середовище розробки, розроблено структурну схему програми, розроблено схему бази даних, виконано конструювання програмного забезпечення ERP системи територіально розподіленого підприємства, здійснено тестування розробленого програмного забезпечення.

Галузь використання – підтримка внутрішніх бізнес-процесів підприємств різної форми власності.

ABSTRACT

Explanatory note to the diploma qualifying work of the master: 162 pages, 27 figures, 4 appendixes, 32 sources.

OPTIMIZATION, SOFTWARE, DATABASE MANAGEMENT SYSTEM, ERP SYSTEM, SAP R / 3, SQL.

Object of study is a process of complex systems software implementation.

Subject of study is an ERP systems software.

The purpose of the work is to research of ERP systems and software implementation of ERP systems of the corporate enterprise.

Materials, methods, and tools: structured and object-oriented programming, DBMS Progress, computer with a monitor, keyboard and mouse manipulator operating under the control of the operating system.

Results. The model of the ERP module of the system of the territorially distributed enterprise is created. Developed software modules of the ERP system of a geographically distributed enterprise, which provides support for internal business processes of the enterprise and provides opportunities for interaction with external objects.

Conclusions. ERP software modules of the territorially distributed enterprise system have been developed. The subject area is analyzed, the technical task for program realization is developed, the development environment is chosen, the structural scheme of the program is developed, the scheme of the database is developed, the ERP software design of the territorially distributed enterprise is performed, the developed software is tested.

The field of use is a support of internal business processes of enterprises of various forms of ownership.

ЗМІСТ

	С.
Перелік скорочень та умовних познач	8
Вступ	9
1 Аналіз предметної області	11
1.1 Аналіз корпоративних інформаційних систем.....	11
1.2 Загальні переваги та недоліки ERP систем	26
1.3 Дослідження ERP системи SAP R/3	31
1.4 Функціональні можливості SAP/R3	40
1.5 Висновки за розділом 1	44
2 Вибір та обґрунтування структури системи, яка проєктується	46
2.1 Аналіз та вибір СКБД корпоративної інформаційної системи	46
2.2 Вибір мови програмування	55
2.3 Перенесення даних із зовнішніх джерел в SAP систему	57
2.4 Розробка моделі модуля SAP R/3 територіально розподіленого підприємства.....	58
2.5 Висновки за розділом 2	68
3 Основні рішення щодо реалізації компонентів системи	69
3.1 Функціональна структура ERP системи підприємства	69
3.2 Структура програмного забезпечення	71
3.3 Висновки за розділом 3	76
4 Керівництво програміста	77
4.1 Призначення й умови застосування програми.....	77
4.2 Характеристики програми.....	78
4.3 Звернення до програми.....	78
4.4 Вхідні і вихідні дані	79
4.5 Повідомлення	79
5 Керівництво оператора.....	81
5.1 Призначення програми	81
5.2 Умови виконання програми	81

5.3 Виконання програми.....	82
5.4 Приклад роботи з програмою	82
5.5 Повідомлення оператору	86
6 Економіко—організаційна частина.....	88
6.1 Планування розробки програмного продукту.....	88
6.2 Визначення витрат на розробку програми	89
6.3 Розрахунок економічної ефективності програмного продукту.....	97
7 Охорона праці та безпека у надзвичайних ситуаціях	100
7.1 Аналіз потенційних небезпек.....	100
7.2 Заходи щодо забезпечення безпеки.....	101
7.3 Заходи щодо виробничої санітарії та гігієни праці	103
7.4 Заходи з пожежної безпеки	108
7.5 Заходи щодо забезпечення безпеки в надзвичайних ситуаціях	110
Висновки.....	116
Перелік джерел посилання	118
Додаток А Технічне завдання	121
Додаток Б Опис програми	125
Додаток В Текст програми	128
Додаток Д Слайди презентації.....	154

ПЕРЕЛІК СКОРОЧЕНЬ ТА УМОВНИХ ПОЗНАК

ERP	– Enterprise Resource Planning;
MRP	– Material requirements planning;
RAD	– Rapid Application Development – швидка розробка додатків;
VCL	– Visual Component Library – бібліотека візуальних компонентів;
KIC	– корпоративні інформаційні системи;
ОЗП	– оперативно-запам'ятовуючий пристрій;
ОС	– операційна система;
ПЗ	– програмне забезпечення;
ПК	– персональний комп'ютер;
ПП	– програмний продукт.

ВСТУП

У наш час в умовах ринкової економіки та конкуренції для багатьох підприємств актуальним питання стало підвищення ефективності управління. У сучасних умовах одним з основних напрямків підвищення ефективності управління є використання програмного забезпечення для автоматизації виробничих та інших процесів, що виконуються на підприємстві [1]–[18]. Процес автоматизації починається з аналізу діяльності підприємства і вироблення основних рекомендацій до майбутньої інформаційної системи. Тільки після цього вирішується питання з вибору тієї чи іншої готової системи або розробки власної. В Україні набагато частіше, ніж за кордоном, приймається рішення про проведення розробки власної інформаційної системи, що обумовлено, з одного боку, обмеженістю фінансових ресурсів підприємства, а, з іншого боку, – відносно дешевою робочою силою. В цьому випадку, доводиться вирішувати цілий ряд проблем таких як, вибір базового програмного і апаратного забезпечення, проектування функціональної структури інформаційної системи, проектування розподіленої бази даних і розрахунок параметрів її функціонування

Останнім часом активізувалися процеси інтеграції дрібних та середніх підприємств у корпорації. Це, у свою чергу, обумовлює необхідність розробки корпоративних інформаційних систем, які повинні забезпечувати роботу декількох територіально розподілених підрозділів. У зв'язку з цим стає неможливим застосування централізованих баз даних та програмного забезпечення. Тому актуальною є розробка нових моделей та модулів програмного забезпечення для автоматизації діяльності підприємства.

Метою роботи є дослідження ERP систем та програмна реалізація модулів ERP системи корпоративного підприємства.

Для досягнення поставленої мети необхідно було розв'язати такі завдання:

- аналіз корпоративних інформаційних систем;
- розробка технічного завдання для реалізації програми;

- вибір засобів розробки програмного забезпечення;
- розробка моделі модуля ERP системи територіально розподіленого підприємства
- розробка структурної схеми програми;
- конструювання програмного забезпечення;
- тестування розробленого програмного забезпечення.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

Як відзначено вище, для підвищення ефективності процесів управління складними територіально розподіленими підприємствами у наш час широко використовується програмне забезпечення (ПЗ) корпоративних інформаційних систем (KIC), модулі якого призначені для автоматизації виробничих та інших процесів, що виконуються на підприємстві [1]–[4].

На ринку корпоративних інформаційних систем представлені різні програмні рішення, зокрема: SAP R/3, BAAN IV, Renaissance CS, Syte Line, Concorde XAL, Oracle Applications, «Галактика», «Парус-Корпорація», «БОС-Корпорація» та ін. Хоча в останні роки і спостерігається тенденція збільшення частки українсько-російських розробок, обсяг українського ринку KIC поки не можна порівняти з європейським і північноамериканським. З усіх компаній, що використовують KIC лише близько чверті використовують для цього вітчизняні рішення. Певною мірою це пов'язано з тим, що значення використання KIC в роботі краще усвідомлюють підприємства, які мають частку іноземного капіталу, вибір яких найчастіше зупиняється на іноземних рішеннях, що добре знайомі інвесторам [5]–[17].

1.1 Аналіз корпоративних інформаційних систем

Відомі KIC з'явилися на перспективному українському ринку на початку 1990-х рр. Були відкриті представництва та укладені партнерські угоди з низкою українських компаній. Однак швидких успіхів з впровадженні цих KIC на українських підприємствах не відбулося. Перша проблема полягає в тому, що корпоративна та інформаційна система не може існувати сама по собі. Вона вимагає певної культури виробництва і управління підприємством. Тому при спробі впровадження KIC доводиться говорити про реінжиніринг бізнес-процесів або реструктуризацію

виробництва. Друга причина впровадження КІС полягає у відмінності між українською і західною системою бухгалтерського обліку. Додаткові проблеми створює постійні зміни законодавства. Крім цього, при поточному рівні заробітної плати на вітчизняних підприємствах економічно не вигідно витрачати великі кошти на її облік, купуючи західні підсистеми для автоматизації обліку заробітної плати та кадрів. Незважаючи на всі ці проблеми, практично всі серйозні західні розробники корпоративного програмного забезпечення закінчили процес адаптації до українських умов. В даний час в Україні представлено доволі багато КІС для різних типів підприємств. Найбільш відомими та широко використовуваними є системи SAP R/3, Oracle Applications, BAAN IV [1]–[17].

Компанія Baan офіційно вийшла на український ринок в 1997 році через компанію «БААН-Євразія». Головний напрямок діяльності компанії «Баан-Євразія» по локалізації системи – це розвиток і підтримка української версії BAAN IV. Випущені нові релізи української версії BAAN в базовій та мультивалютній модифікаціях. Випущений також новий реліз української версії з істотним розширенням базової функціональності з певних напрямків управлінського та бухгалтерського обліку, розробляються додаткові засоби для генерації звітів. Також використовується галузеве розширення BAAN для аерокосмічної і оборонної промисловості (Baan Aerospace & Defense), а також модуль, що відповідає за інформацію про виріб (Baan PDM) [1]–[17].

Oracle Application Express (Oracle Apex, APEX) є вільним середовищем швидкої розробки прикладного програмного забезпечення на основі СКБД Oracle Database, реалізована як веб-додаток. Всі елементи, що виникають в циклі розробки програми в даному середовищі зберігаються безпосередньо в інфраструктурі Oracle Database, тим самим забезпечується спільна робота розробників і контроль версій без використання файлів і додаткових систем управління версіями. Додатки можуть бути розгорнуті на Oracle Database без додаткового програмного забезпечення (використовуючи вбудований в СКБД веб-сервер), або веб-сервер може бути винесений на

зовнішній сервер на основі Apache httpd з модулем mod_plsql. Також існує Java-додаток Oracle APEX Listener, який може бути розгорнуто на J2EE-серверах додатків для забезпечення функцій веб-сервера для APEX-додатків з можливостями кешування на стороні веб-сервера, конфігурації через веб-інтерфейс, додатковими функціями з моніторингу та забезпечення безпеки [1]–[17].

ERP-система для середніх підприємств Concorde XAL складається з трьох елементів: процедур обробки даних, що складають його функціональність, ядра, виконуючого ці процедури, і засобів розробки з доступом до вихідних текстів і функціональності. Для більшості замовників ці процедури забезпечують 80-90% функціональних вимог, що пред'являються ними до системи. Це стандартний рівень відповідності для систем такого класу. Функціональність якої не вистачає додається за допомогою використання вбудованих в "Конкорд XAL" засобів розробки. Ядро системи написано на C++ і доступно тільки виробнику, а вся бізнес-функціональність системи реалізована на мові XAL і доступна клієнтам. Така архітектура дозволяє підлаштовувати систему під бізнес-процеси клієнта. Ще одне завдання, яке вирішується завдяки такому підходу – це захист системного ядра в разі некоректно реалізованої модифікації. На відміну від рішень, в яких доопрацювання системи ведеться на рівні ядра на мові типу C, Кобола і т. п., зіпсувати системне ядро в "Конкорді XAL" неможливо. Технологію модифікації і розвитку продукту можна назвати технологією «шарового пирога». Компанія Damgaard Data надає базовий та стандартний варіант системи, який утворює її серцевину (SYS). Далі розташовується дистриб'юторський шар (DIS) – локалізований варіант продукту. Наступний шар – дилерський. Він може складатися з двох підшарів. Перший прошарок (VAR) – орієнтований на варіанти системи в конкретній галузі, допрацьовані дилером, другий (BUS) – варіанти системи, допрацьовані дилером з урахуванням потреб клієнтів і встановлені у конкретних замовників.

Нарешті, останній шар, який також може включати два прошарки – це варіанти системи з доопрацюваннями, створеними самим замовником [1]–[17].

Велика компанія може купити систему, виконати її доопрацювання та передає своїм філіям. Ця розробка утворює перший призначений для користувача прошарок (CUS). Другий призначений для користувача прошарок утворюють інсталяції з доопрацюваннями, виконаними на конкретних підприємствах (USR). Побудова системи за принципом «шарового пирога» полегшує також оновлення її версій. Політика розвитку продукту така, що розробники в Данії основний акцент роблять на підвищення гнучкості ядра системи, тобто на ті її можливості, які впливають на швидкість, зручність і безпеку розвитку системи дистриб'юторами, дилерами і користувачами. Нові функціональні можливості з'являються, але їх обсяг непорівнянний із змінами, що зачіпають ядро. Damgaard Data випускає нові версії приблизно раз на півтора року [1]–[17].

KIS Ахарта, розроблена датською компанією Damgaard містить такі модулі: «Фінанси і бухгалтерський облік», «Продажі», «Закупівлі», «Логістика і склад», «Виробництво», «Управління проектами», «Електронна комерція», «Управління відносинами з клієнтами», «Управління персоналом». В системі реалізовані можливості роботи з територіально розподіленими структурами з підключенням до 1000 користувачів. Особливу увагу приділено програмам для ведення електронного бізнесу (як B2B, так і B2C). Система може працювати з MS SQL Server і з Oracle. Вона є об'єктно-орієнтованою і побудована на основі відкритих технологій (стандартів COM/DCOM). Ахарта реалізована в клієнт-серверній архітектурі і може працювати як в дворівневому, так і в трьохрівневому варіанті. В клієнтській частині може бути використаний як Windows-, так і Web-інтерфейс (або обидва одночасно) [1]–[17].

Використання KIC ERP-класу iRenaissance корпорації ROSS Systems дозволяє досягти найбільш високих показників при використанні на середніх і великих підприємствах з процесним типом виробництва, до якого можна віднести такі галузі промисловості: харчова; хімічна; металургійна; нафтопереробна; целюлозно-паперова; фармацевтична. Крім того, iRenaissance успішно використовується для автоматизації управління в медичних, адміністративних та освітніх установах, в бібліотеках, в сфері збуту.

Система iRenaissance має низку яскравих відмінних рис: порівняно швидке впровадження - власна методологія FrontRunner дозволяє скоротити терміни впровадження системи до 3-6 місяців; відносно низька вартість експлуатації; легкість адміністрування системи; велика кількість успішних використання в галузях з процесним типом виробництва [1]–[17].

В основу iRenaissance закладені передові концепції управління ресурсами підприємства ERP (Enterprise Resource Planning) та управління ланцюгами поставок SCM (Supply Chain Management). Вони дозволяють раціонально управляти підприємством та звести до мінімуму витрати виробництва і домогтися швидкого повернення інвестицій. Основні завдання, які вирішуються iRenaissance: планування обсягів виробництва; облік готової продукції; зберігання і пошук управлінської інформації (постачальники, матеріали, обладнання тощо); розрахунок витрат, калькуляція собівартості продукції; організація і планування виробничого процесу; управління фінансами та контроль взаєморозрахунків з постачальниками і споживачами; бухгалтерський облік на підприємстві та його підрозділах; аналіз і прогноз продажів; контроль за використанням кредитів; планування і аналіз бюджету підприємства і його підрозділів; розрахунок, аналіз, узагальнення та прогноз фінансових показників підприємства; розрахунок грошових потоків на основі бухгалтерських звітів; управління перевезеннями і оптимізація транспортних витрат; управління експлуатацією обладнання; організація електронного документообігу; ведення електронного архіву підприємства; облік кадрів,

управління персоналом; облік трудовитрат і розрахунок заробітної плати; управління проектами.

Система складається з ряду модулів, починаючи з виробничого та фінансового і закінчуючи модулями управління персоналом і розрахунку зарплати. Ці модулі можуть використовуватися незалежно один від одного для автоматизації окремих напрямків діяльності підприємства. Таким чином, компанія, що розвивається може бути автоматизована відповідно до своїх поточних вимог і надалі нарощувати функціональність системи за рахунок впровадження додаткових модулів у міру розвитку нових напрямків діяльності організації.

Впровадження iRenaissance дозволяє оптимізувати роботу підприємства – позбутися від непотрібних і повторюваних операцій, запасів, тобто скоротити витрати і збільшити прибуток. Крім того, впровадження iRenaissance – це спосіб сертифікації виробництва у відповідності з ISO 9000 – системи міжнародних стандартів якості.

Популярність системи iRenaissance пояснюється, зокрема, можливістю її налаштування в залежності з офіційним вимогами документообігу та звітності різних країн. Система має вбудований репозиторій знань (бізнес-процедур), в якому тільки з самого початку зберігаються 1400 готових бізнес-прецедур. iRenaissance повністю локалізована, що підтверджується успішно завершеними проектами по впровадженню і успішної експлуатацією системи на російських підприємствах. iRenaissance є відкритою системою - користувач може додавати в неї необхідні функції, використовуючи вбудовані CASE-засоби і мову четвертого покоління (4GL).

Система SyteLine ERP представляє собою серйозну і гнучку основу для розширення за рахунок додаткових комплементарних програмних продуктів, що в комплексі забезпечують широку функціональність і підтримку передових технологій, дозволяючи ефективно керувати бізнесом в

різних галузях промисловості. Система SyteLine ERP володіє повним набором функцій для забезпечення роботи в наступних функціональних областях: обслуговування клієнтів; керування витратами; обробка замовлень; керування якістю; прогнозування і керування збутом; керування запасами і закупівлями; планування потреб у сировині і матеріалах; керування виробничими процесами; планування та керування виробничими потужностями; керування цехами і диспетчеризація; бухгалтерський і податковий облік; керування персоналом; керування фінансами; керування проектами; зв'язок з САПР і АСУ ТП [1]–[17].

За допомогою SyteLine ERP можна легко спланувати завантаження устаткування, трудовитрати і потреби в сировині, необхідні для виробництва продукції під замовлення клієнта або масовим способом. Система призначена для управління запасами як у кожній структурній одиниці підприємства, так і між цими одиницями. При розрахунку потреб в матеріалах проводиться зіставлення замовлень клієнтів з наявними ресурсами підприємства і визначається, які комплектуючі повинні бути закуплені або вироблені для виконання взятих зобов'язань. Функції планування реалізують мережеве планування з урахуванням обмежених і необмежених потужностей. Потужності можуть розраховуватися як по робочим центрам, так і по виробничим лініях. В системі SyteLine ERP реалізована підтримка бази постачальників, що дозволяє оперативно виявити найбільш вигідних постачальників за цінами, якістю послуг і термінів поставки [1]–[17].

Характеристики SyteLine ERP: ергономічний інтерфейс користувача; можливість управління територіально-розподіленими майданчиками; повнофункціональна головна книга; технічна документація онлайн; підтримка прийняття управлінських рішень; керування документообігом і повідомленнями; високий рівень захисту інформації та розмежування доступу;

Система SyteLine ERP дозволяє відстежувати провадження за допомогою методу замовлення-наряд, виробничим графіками або етикетками

KANBAN. Метод замовлення-нарядів дозволяє відстежувати вартість на самому детальному рівні, тобто за кожним виробом, що випускається. Для підприємств з нежорсткими вимогами з контролю собівартості (масовий або крупносерійний випуски) можна використовувати виробничі графіки, що скорочують кількість інформації, що вводиться. Цехи, організовані за методом KANBAN, потребують лише в реєстрації тільки виробленої кількості готової продукції, і SyteLine ERP списує витрати сировини і витрачений час безпосередньо на базі специфікацій або рецептур. В системі SyteLine ERP такі комбінації методів можуть реалізовані і на рівні виробів [1]–[17].

SyteLine ERP дозволяє: збільшити продажі; автоматизувати бізнес-процеси; підвищити конкурентоспроможність; організувати ефективний логістичний ланцюжок; організувати ефективну взаємодію з клієнтами; оцінювати замовлення точно і швидко; скоротити час виконання замовлення; знизити терміни виведення на ринок нової продукції; знизити операційні витрати; підвищити ефективність праці фахівців; краще планувати і використовувати виробничі потужності; знизити комерційні витрати і собівартість; мінімізувати повернення продукції і переробку браку; оцінювати ступінь досягнення поставлених цілей; максимізувати темпи зростання; підвищити рентабельність.

Система SyteLine ERP дозволяє обробляти замовлення клієнтів швидко, гарантуючи, що пропозиції, зроблені клієнтам, будуть реалістичними та виконаними. При цьому, швидкість відгуку на запит така, що дозволяє протягом декількох хвилин відповісти на більшість питань, що виникають: про наявність товару, стан замовлення, розмір заборгованості по оплаті або знижках, що надаються. До того ж система дозволяє обробляти непрогнозовані, швидко мінливі запити клієнтів.

SyteLine ERP взаємодіє з багатьма системами, серед яких: електронні торгові майданчики; автоматизовані системи збору даних; система розрахунку заробітної плати; система керування персоналом; система по

спільним розробкам нових продуктів; системи автоматизованого проектування (САПР/АСУ ТП); генератори звітів; системи планування маршрутів та оптимізації доставки; системи управління документообігом; системи електронного обміну даними; облікові та бухгалтерські системи російських розробників; системи управління життєвим циклом продукту (Product Life Cycle), управління взаємовідносинами з клієнтами (Customer Relationship Management), синхронного планування та оптимізації (Advanced Planning & Scheduling - APS), управління ланцюгами поставок (Supply Chain Management - SCM) [1]–[17].

В наш час на українському ринку пропонуються різні вітчизняні програмні продукти управління підприємствами. Найбільш відомими КІС вітчизняної розробки є: «Галактика»; «Парус-Корпорація»; «БОС-Корпорація»; NS2000; «1С»; «БЕСТ-ПРО»; IBS Trade House; «Акорд»; «Альфа»; «Еталон»; «Флагман»; «Супер-Менеджер»; «Інфо-Бухгалтер»; «Турбо-Бухгалтер»; «Ресурс», VRsystem; «Скат»; «Еверест»; «Компас»; «Моноліт»; GESTORY; ABACUS Financial та ін.

Система «Галактика» – ERP система, складова частина комплексу бізнес-рішень корпорації Галактика. Система адресована середнім і великим підприємствам і володіє широким функціоналом для інформаційної підтримки задач стратегічного планування та оперативного управління. Галактика ERP розроблена для застосування в умовах вітчизняної економіки з її яскраво вираженою специфікою, постійними змінами законодавства. До плюсів «Галактики» можна віднести досить глибоку інтегрованість і чітку підтримку законодавчої бази. Система Галактика ERP має компонентну структуру і складається з функціональних модулів, які об'єднуються в контури, що дозволяють системі бути гнучкою і вирішувати специфічні завдання окремих галузей промисловості. Апробовані технології розгортання проектів автоматизації сприяють тому, що впровадження ERP системи проходить в стислі терміни, з фіксованим бюджетом і малими ризиками. Розробка і впровадження системи Галактика ERP ведеться з використанням

передових технологій і інструментів. Додаткові можливості для інтеграції Галактики ERP з продуктами сторонніх виробників надають реалізацію в системі сервіс-орієнтованої архітектури (SOA) і технологій web-сервісів.

Парус – серія програмних продуктів, призначених для автоматизації діяльності організацій сектора державного і муніципального управління, а також комерційних підприємств. Система "ПАРУС – Підприємство 7" призначена для малих і середніх підприємств різної галузевої приналежності (торгівля, сфера послуг, елементарне виробництво, реклама та ЗМІ, громадське харчування, туризм, іноземні компанії та ін.). Це проста, зручна, але в той же час потужна повнофункціональна система, що дозволяє автоматизувати бухгалтерський облік, основні торговельні процеси і складський облік, розрахунок заробітної плати і кадровий облік. Може експлуатуватися як на одному, так і на декількох об'єднаних в локальну мережу комп'ютерах. Користувачами системи можуть бути: головні бухгалтери та бухгалтери по розділах обліку; менеджери зі збуту і складські працівники; фахівці фінансово-економічних відділів.

Система побудована за модульним принципом і являє собою набір модулів, кожен з яких призначений для автоматизації одного з основних видів діяльності підприємства і може працювати як в автономному режимі, так і спільно з іншими модулями комплексу, створюючи єдиний інформаційно-управлінський простір масштабу підприємства: бухгалтерія; реалізація і склад; комплекс; облік договорів; комплектування; заробітна плата; кадри. Кожен модуль може працювати як самостійний додаток, але в повній мірі переваги модулів реалізуються при використанні їх в якості єдиного програмного комплексу із загальною базою даних. Бухгалтерська частина комплексу являє собою універсальну повнофункціональну бухгалтерську систему з новими можливостями, наданими Windows-інтерфейсом. Система увібрала в себе весь багаторічний досвід Паруса і його численних клієнтів. Призначена для широкого кола підприємств. Торгово-складська частина комплексу являє собою систему для автоматизації

торгівельної діяльності. Тут управління торгівлею розглядається як самостійний вид діяльності, виведений за рамки бухгалтерії. Торгова частина комплексу являє собою окремий повноцінний програмний продукт, що відповідає вимогам менеджерів і орієнтований на них і інших працівників торгівлі. У ній можна враховувати і контролювати не тільки рух товарів, але і стан фінансів, і взаєморозрахунки з автоматичним контролем ліміту кредитування. Модуль "Комплекс" об'єднує в собі можливості бухгалтерської та товарно-складської частин програми – це класика комплексних рішень Корпорації «Парус» з використанням нових, прогресивних технологій. Модуль «Зарплата» дозволяє проводити повний цикл розрахунку заробітної плати і отримувати звіти для Податкової інспекції і Пенсійного Фонду. Кадрова частина дозволяє автоматизувати процес кадрового обліку на підприємстві від ведення штатного розкладу і наказів по прийому і руху кадрів до отримання різних статистичних звітів. Облік договорів є доповненням в складській частини комплексу для автоматизації обліку договорів. Дана комплектація дозволяє враховувати і контролювати договори про купівлю продаж, надання послуг і т. д. Можна простежити рух товарів, а також стан фінансів і взаєморозрахунки з автоматичним контролем ліміту кредитування.

Перевагами системи є: простота освоєння; широкі функціональні можливості; висока надійність функціонування; типові налаштування на різні типи підприємств; скорочення витрат на автоматизацію шляхом вибору оптимальної для вас комплектації з можливістю її подальшого нарощування; можливість аналізу облікових даних і інформації в базі даних.

KIC NS2000 базується на СКБД Progress, працює на різних типах комп'ютерів і під різними ОС: всіх UNIX-платформах; MS Windows і Windows NT; Novell Netware; OS/2; VAX/VMS. Для розробки NS2000 використана мова Progress 4 GL. У багатокористувацьких системах (UNIX, VMS і ін.) NS2000 дозволяє реалізувати як традиційну клієнт-серверну архітектуру, так і термінальний режим. Для одного користувача систем

(Windows) підтримується доступ до серверу через мережу і автономний режим роботи з БД. Введення даних розділене на багатьох користувачів та реалізований через автоматичну асинхронну реплікацію даних: через електронну пошту по звичайних телефонних лініях із застосуванням модемного зв'язку, через виділені оптоволоконні канали зв'язку або супутникові системи зв'язку, через мобільні віддалені робочі місця. Система побудована за модульним принципом. Її основу складають універсальний конструктор і набір модулів, які можуть працювати незалежно. До складу NS2000 входять наступні групи модулів: «Управління фінансами», «Логістика», «Управління персоналом», «Управління виробництвом» і «Аналіз» [1]–[17].

KIC VRsystem (розробки компанії V-Real) побудована за модульним принципом з застосуванням технологій об'єктно-орієнтованого програмування і реалізована на багаторівневій клієнт-серверній архітектурі. Система працює на різних апаратних платформах під управлінням MS Windows NT або UNIX. В якості СКБД використаний сервер Adaptive Server Enterprise компанії Sybase. В основу VRsystem покладено такі технології: об'єктно-реляційну; самовизначення (універсальних) документів; модель розподіленої обробки Open Client - Open Server компанії Sybase. Ядро VRsystem – це сервер бізнес-логіки, що за допомогою серверної бібліотеки розбирає потік даних, який приходить від клієнта, який виконує обробку бізнес-правил, і прозора переносить пропозиції мови Transact-SQL на вказаний SQL Server. VRsystem складається з більш ніж 100 функціональних модулів, які групуються за такими напрямками: «Управління запасами», «Управління закупівлями», «Управління фінансами», «Адміністрування».

KIC «Еверест» призначена для автоматизації управління матеріальними, фінансовими і людськими ресурсами промислових і торгових підприємств. Вона побудована за модульним принципом і реалізована в клієнт-серверній архітектурі з використанням Borland Delphi в якості клієнта і Oracle – як сервер. Система працює з найбільш поширеними ОС: MS

Windows NT, Novell Netware і різними версіями Unix. При необхідності вона інтегрується через відповідні шлюзи з ПЗ розробки третіх фірм. В систему вбудовані інструментальні засоби власної розробки і підтримується до 100 і більше робочих місць. До складу «Евересту» входять наступні функціональні підсистеми: «Оперативний облік», «Бухгалтерський облік», «Фінансовий контролінг», «Управління персоналом».

ПЗ «БЕСТ-ПРО» (розробка фірми «Інтелект-Сервіс») розроблено на основі Btrieve на мові C ++ і може працювати в наступних мережевих ОС: Novell Netware, Windows. Є версія для промислових підприємств і для оптової торгівлі, готується окрема розробка для супермаркетів. Ряд модулів є як в DOS-, так і в Windows-варіанті, деякі ж призначені тільки для роботи під DOS, наприклад, для підрозділів, де потрібно масове введення даних, який зручніше здійснювати за допомогою клавіатури. Для застосування в роздрібній торгівлі система інтегрується з широким набором касових апаратів (9 типів POS-терміналів і 11 електронних касових реєстраторів) і різними видами торгового обладнання. Зв'язок з віддаленими об'єктами може бути забезпечений за допомогою модему. До складу «БЕСТ-ПРО» входять підсистеми управління запасами, закупівлями і продажами, контролю договірних зобов'язань, фінансів, взаєморозрахунків з постачальниками і покупцями. Крім того, є блоки обліку кадрів, зарплати, основних фондів, бухгалтерського та податкового обліку. Система має відкриту архітектуру, структура БД входить в комплект поставки. Користувачі і / або партнери можуть робити власні доопрацювання системи (модулі, нові звіти, функції) [1]–[17].

Популярне в Україні (особливо, серед малих підприємств) ПЗ фірми «1С» позиціонується компанією в 2-х напрямках. По-перше, як універсальний засіб створення прикладних рішень для автоматизації економічної діяльності (платформа «1С: Підприємство») - в цьому випадку система використовується для оптової та роздрібної торгівлі, для змішаних форм підприємств, для обліку послуг і т. д. По-друге, як готові рішення, що

поставляються «1С». Існують типові конфігурації «1С: Торговлі й Складу» (готові прикладні рішення), призначені тільки для торгового обліку, і типові конфігурації для комплексного обліку в торгових організаціях, в яких присутня бухгалтерський облік та розрахунок заробітної плати. Система «1С: Торговля і Склад», крім базових можливостей, містить 3 додаткові функціональні компоненти: «Оперативний облік», «Бухгалтерський облік» і «Розрахунок», які поставляються окремо. Вся прикладна частина поставляється як готове рішення, але її можна налаштовувати відповідно до вимог користувача. «1С: Торговля і Склад» працює під управлінням ОС Windows, в якості БД можуть використовуватися MS SQL Server або БД в форматі .dbf. У систему може бути включено торгове обладнання: контрольно-касові апарати, POS-термінали та ін.

КІС «Дельфін» є корпоративною системою управління середніми і великими підприємствами як роздрібною, так і оптовою торгівлі. Вона побудована на 2-рівневій клієнт-серверній архітектурі. Вся бізнес-логіка, обробка даних, забезпечення цілісності даних, розмежування прав доступу до об'єктів БД зосереджені на сервері. Клієнтська програма займається виключно інтерфейсом користувача і не займається даними. Воно побудоване таким чином, що дозволяє робити налаштування під певні потреби кожного конкретного користувача в індивідуальному порядку. Як клієнтська ОС використовується MS Windows. Система розмежування прав доступу засновані на моделі безпеки MS SQL Server і підтримується на рівні інтерфейсу клієнта. При впровадженні «Дельфіна» можливо сполучення з різним торговельним і технологічним устаткуванням і з іншими системами. У системі реалізований графічний користувацький Windows-інтерфейс. Основою обліку в «Дельфіні» вважається подія (господарська операція), а не документ, складений за підсумками події. Виробничі модулі ПЗ «Дельфін» дозволяють створювати заявки на продукцію, детально планувати виробництво, виписувати замовлення-наряди, враховувати фактичний випуск продукції, розраховувати планово-фактичну собівартість, аналізувати

відповідність факту плану. ПЗ поставляється з набором вже готових звітів. В Дельфін для зміни звітів (або створення нових) застосовується вбудований конструктор звітів. Він дозволяє створювати документи з різними шрифтами, графічними примітивами, кольорами, відображати дані у вигляді ділової графіки тощо. Як для видобування даних, так і для їх зміни, може бути задіяний механізм збережених запитів (в системі реалізований прямий доступ до БД за допомогою збережених SQL-запитів). Через інтерфейс ODBC можливий доступ з ряду програмних продуктів (MS Word, MS Access, MS Excel) до даних системи «Дельфін». Для обміну даними можуть бути також використані вбудовані можливості системи з імпорту/експорту через текстові файли (ASCII формат) або зв'язок з OLE Automation (Excel, Word).

1.2 Загальні переваги та недоліки ERP систем

Сильною стороною західних систем є алгоритми. Однак західні КІС в тій чи іншій мірі страждають від того, що українське оподаткування, бухгалтерський облік та звітність далекі від прийнятих на заході принципів. Вітчизняні КІС більше пристосовані до українських умов, але багато аналітиків стверджують, що українське ПЗ поки програє західному з точки зору якості і повноти. Представлені на українському ринку західні КІС, як правило, розроблені провідними світовими компаніями, такими як SAP або BAAN. Вони пройшли багаторічні випробування, перш ніж потрапили на український ринок. Тому майже всі західні корпоративні системи відповідають стандартам. Вони можуть застосовуватися широким колом підприємств різного масштабу і спеціалізуються в різних областях. Ці системи побудовані з використанням сучасних технологій і дають найвищі показники в продуктивності. Відповідність систем стандарту ERP зокрема означає добре продумані алгоритми виробничого і фінансового планування та моделювання [1]–[17].

До недоліків західних КІС можна віднести високу вартість. Непридатність же використання систем на деяких українських підприємствах, що мають неорганізовані і не налагоджені процеси, недоліком не є [1]–[17].

Розробки вітчизняних розробників ближчі до української специфіки і більш дешеві, тому їх впровадження виправдано для малих і середніх підприємств для вирішення конкретних завдань. В силу невідповідності хоча б стандарту MRP, багато українських систем не можуть застосовуватись на великих підприємствах, що мають кілька типів виробничих процесів. Вітчизняні розробники КІС поки слабо пропрацювали питання виробничого планування та моделювання.

Останнім часом спостерігається тенденція підвищення якості КІС українського виробництва. Деяким великим розробникам вдалося

наблизитися до стандарту ERP за прийнятною для України вартості продукту. Важливим є процес прийняття рішення про придбання ERP-системи. Етап оцінки та відбору ERP-системи для впровадження в організації має велике значення. Особливої уваги потребує оцінка вкрай складних ERP-систем, таких, як SAP R / 3, їх функціональності і технологічності. При виборі ERP-системи необхідно знайти ретельно збалансований компроміс між двома вимогами до ERP-системи, що суперечать один одному, а саме комплексністю та гнучкістю, причому як на рівні функціональності, так на рівні технології. Виробничі компанії працюють по-різному, процес виробництва може бути найрізноманітнішим, починаючи з виготовлення на замовлення і закінчуючи поточним виробництвом. Тому в деяких випадках, щоб виключити можливість появи дефектів функціональності, компаніям варто самим розробити ту чи іншу програму, а не купувати її. Порівняльна характеристика ERP-систем представлена в табл. 1.1 [1]–[17].

Таблиця 1.1 – Порівняльна характеристика ERP-систем

ERP-система	Виробник	Сфера застосування	Термін впровадження / можливість внесення специфічних змін	Простота установки / використання	Вартість впровадження
1	2	3	4	5	6
Галактика	Галактика	Нафтогазова галузь, енергетика, металургія, підприємства ВПК, харчова промисловість	4 місяці - 1,5 року і більше / не можна вносити зміни	Проста встановці \ використанні	В середньому ліцензія обходиться в \$2 тис. За одне робоче місце. Вартість впровадження становить близько 100% цієї суми

Продовження таблиці 1.1

1	2	3	4	5	6
Парус	Парус	Машинобудування, нафтогазові компанії, підприємства енергетичної галузі	4 місяці - 1 рік і більше / не можна вносити зміни	Проста в установці \ використанні	Вартість ліцензії на одне робоче місце \$ 1-2 тис. Вартість впровадження 100-200% ціни рішення
1С: Підприємство 8.2. Управління виробничим підприємством	1С	Машинобудування, харчова промисловість та ін.	3 - 9 місяців і більше \ можна вносити зміни	Складна в установці / використанні	Ліцензія на одне робоче місце \$ 150-600. Вартість впровадження на одне робоче місце \$ 200-1000
Oracle Applications	Oracle (США)	Важка промисловість (металургія), телекомунікаційні компанії, фінансовий сектор, хімічна промисловість	1-5 років і більше \ можна вносити зміни	Складна в установці / використанні	Вартість рішення на одне робоче місце становить близько \$5 тис. повна вартість істотно залежить від необхідної функціональності і складності впровадження.
IFS Application	IFS (Швеція)	Підприємства машинобудівного комплексу, енергетика, харчова промисловість, фармацевтика, кабельна промисловість	0,8-3 роки і більше \ можна вносити зміни	Складна в установці / використанні	Повна вартість впровадження може досягати \$ 250 тис. і більше. Також істотно залежить від необхідної функціональності.
Baan ERP	SSA Global (США)	Хімічна промисловість, автомобілебудування, фармацевтика, харчова промисловість	6 місяців - 1,5 року і більше \ можна вносити зміни	Складна в установці / використанні	Вартість одного робочого місця - \$3 тис. співвідношення ціни рішення і витрат на впровадження 1: 1-3.

Продовження таблиці 1.1

1	2	3	4	5	6
iRenaissance	Ross Systems (США)	Харчова промисловість, хімічні компанії, металургійна промисловість нафтопереробні, целюлозно-паперові, фармацевтичні підприємства	4 місяців – 1,3 року і більше \ можна вносити зміни	Складна в установці / використанні	Вартість впровадження в середньому \$ 200 тис.
MBS Ахapta, Navision	Microsoft (США)	Підприємства нафтової галузі, харчової промисловості, торгіві компанії, металургія, дистрибуція, телекомунікаційна галузь	6 місяців - 2 роки і більше \ можна вносити зміни	Складна в установці / використанні	В середньому вартість рішення на одне робоче місце \$3,5 тис. вартість впровадження становить 100-250% вартості рішення.
iScala	Epicor (США)	Машинобудування, телекомунікаційна галузь, харчова промисловість	3 місяців - 1,5 року і більше \ можна вносити зміни	Складна в установці / використанні	Середня вартість iScala 2.1 становить \$ 2-5 тис. за одне робоче місце.
MFG/PRO	QAD (США)	Автомобільна, авіаційна, електронна, електротехнічна, хімічна, фармацевтична і харчова промисловість	3 місяців - 1,5 року і більше \ можна вносити зміни	Складна в установці / використанні	Вартість ліцензії на одне робоче місце \$ 2-5 тис. в залежності від конфігурації. Впровадження обходиться в 100-200% цієї суми.
J.D.Edwards OneWorld	J.D.Edwards (США)	Гірничо-добувна промисловість, будівельні організації, торгіві компанії, нафтогазовий сектор	7 місяців - 1,5 року і більше \ можна вносити зміни	Складна в установці / використанні	Вартість робочого місця OneWorld варіюється від \$ 400 до \$ 4000.

Продовження таблиці 1.1

1	2	3	4	5	6
SyteLine ERP	MAPICS (США)	Виробники вимірювального і електрообладнання, деревообробка, поліграфія, машинобудування	6-9 місяців і більше \ можна вносити зміни	Складна в установці / використанні	Вартість ліцензії на одне робоче місце \$ 2-4 тис. приблизно в стільки ж обійдеться впровадження.
SAP R/3	SAP AG (Німеччина)	Широка сфера застосування: оборонні підприємства, компанії нафтогазового комплексу, металургія, енергетика телекомунікації, банківський сектор	1-5 років і більше \ можна вносити зміни	Проста в установці \ використанні	Ліцензія на 50 робочих місць коштує близько \$350 тис. вартість впровадження може в кілька разів перевищувати вартість рішення.

Вибір ERP-рішення – вкрай складне і комплексне завдання, що вимагає серйозного дослідження організації і чіткого формулювання вимог до корпоративної інформаційної системи. З огляду на, те що система вибирається під підприємство зі складною територіально розподіленою структурою, повинна бути проста установка, а також можливість доопрацювання власних модулів [1]–[17].

Тому як базис обрано систему SAP R/3, яка має широку сферу застосування (оборонні підприємства, компанії нафтогазового комплексу, металургія, енергетика телекомунікації, банківський сектор тощо), є зручною у впровадженні та експлуатації, надає широкі можливості щодо підтримки виробничих та організаційних процесів, а також забезпечує можливості створення власних програмних модулів та впровадження їх в існуючу систему.

1.3 Дослідження ERP системи SAP R/3

Система SAP R/3 складається з набору прикладних модулів, які підтримують різні бізнес-процеси компанії і інтегровані між собою в масштабі реального часу [1]–[3], [8]–[10], [12], [15], [17].

Система R/3 призначена для обліку і планування матеріальних і фінансових ресурсів компанії в режимі реального часу. Система SAP R/3 орієнтована головним чином на великі і середні підприємства. Система SAP включає в себе декілька модулів [1]–[3], [8]–[10], [12], [15], [17].

Модуль BC (Базовий модуль) являє собою набір компонентів системи SAP R/3, включаючи керуючу систему SAP R/3 [8]–[10], [12], [15], [17]. Він виконує такі завдання:

- моніторинг системи за допомогою центральної керуючої системи з використанням моніторів продуктивності і раннього попередження;
- системне обслуговування (наприклад, контроль зареєстрованих користувачів і блокування бази даних, оновлення бази даних, підтримка журнального файлу системи і таблиць бази даних);
- керування системою розмежування доступу;
- управління виведенням інформації на друк;
- керування системою перенесення даних;
- керування клієнтами;
- архівування даних.

Модуль BC включає також ABAP/4 Development Workbench - інтегроване середовище розробки, яке містить всі інструменти (Object Browser, словарь ABAP/4, редактор ABAP/4, бібліотека функцій, дизайнери екранів і меню), необхідні для створення програм. Крім того, модуль включає в себе інтерфейси між програмної взаємодії і генерації звітів [8]–[10], [12], [15], [17].

Інструментарій для бізнес-проектування, що входить до складу модуля BC, включають процедурну модель R/3, яка використовується при

впровадженні SAP R/3 на підприємствах, керівництво по налаштуванню, що містить детальну інформацію про встановлення системи, структуру клієнтів і керівництво з впровадження, кількість посилань модель R/3 і потік бізнес-операцій SAP. Крім того, модуль BC включає засоби управління базами даних (ADABAS, Oracle, Informix і т. д.), що дозволяють встановлювати СКБД і управляти ними, а також Basis Services, SAPscript для обробки текстів і створення друкованих документів, SAP ArchiveLink і SAPoffice (електронна пошта і система зберігання офісної інформації).

Модуль FI (Модуль обліку та звітності) є центральним модулем системи SAP R / 3. Він відповідає за зовнішній облік і звітність, роботу з рахунками головної бухгалтерської книги, в якій реєструються операції, що виконуються в інших модулях, перед тим як дані про них вводяться в балансний звіт і звіт про прибутки і збитки. Ось найбільш важливі його особливості. Модуль складається з наступних частин [8]–[10], [12], [15], [17]:

- головна книга – основні записи / діаграми рахунків, проводки документів, відображення проводок і балансу, інформація про банки і т.д.;
- рахунки кредиторів – основні записи постачальників, проводки документів, відображення проводок і балансу, вирівнювання відкритих позицій, програма розрахунків і т.д.;
- рахунки дебіторів – основні записи клієнтів, проводки документів, відображення проводок і балансу, вирівнювання відкритих позицій, спадні платежі, управління кредитами і т.д.;
- облік основних засобів (раніше називався «модуль AM» або управління основними засобами) – оцінка і амортизація основних засобів;
- операції по закриттю – закриття дня / місяця / року в зовнішньому обліку та звітності, податкові звіти і т.д.;
- інформаційна система фінансів – інформаційна система дебіторів / кредиторів і головної книги.

Крім того, FI містить процедури автоматизації (автоматичні проводки, кліринг), засоби групування і спеціальні реєстри.

Модуль СО (Модуль контролінгу) призначений для ведення внутрішньої звітності [8]–[10], [12], [15], [17]. Його елементи такі:

- облік накладних витрат – облік складових вартості чистих доходів, фактичних, стандартних і граничних витрат, витрат по процесах, внутрішніх замовлень, податків, перевитрат, гнучкого ціноутворення на основі маржинальних витрат;
- калькуляція собівартості продукту – включає облік прямих витрат, розрахунок вартості продукту;
- аналіз прибутковості – включає витрати на реалізацію.

Модуль ІМ (Модуль управління інвестиціями) дозволяє планувати інвестиції в розвиток основних засобів. Зважаючи на це даний модуль тісно взаємопов'язаний з модулями FI, СО і PS (модуль проєктів).

Модуль TR (Модуль фінансового менеджменту) забезпечує виконання функцій управління фінансами і фінансовими потоками, управління бюджетом, а також управління фондами. Нижче наведено опис елементів модуля:

- управління фінансами – підтримка функцій короткострокового фінансового планування та прогнозування, тобто контроль ліквідності і фінансових потоків. Для цього використовуються показники стану поточних рахунків (щоденний фінансовий стан) і особових рахунків (прогноз ліквідності, контроль руху коштів на рахунках кредиторів і дебіторів);
- грошові потоки та управління бюджетом - забезпечує середньо- і довгостроковий контроль ліквідності. При цьому використовується інформація про операції з модуля FI. Організаційним елементом для управління грошовими потоками є область фінансового менеджменту, в яку входять одне або кілька підприємств;
- управління фондами – плани руху грошових коштів аналізуються за сферами діяльності компанії і доступні у вигляді кошторисів. Порівнюючи їх з реальними фінансовими показниками, можна виявити відхилення.

Модуль ЕС (Модуль контролінгу діяльності підприємства) складається з двох частин: інформаційної системи для менеджменту та системи обліку за місцями виникнення прибутку [8]–[10], [12], [15], [17]. Інформаційна система для менеджменту призначена для збору і надання інформації, необхідної для прийняття рішень. Система дозволяє, використовуючи параметри і контрольні значення, аналізувати дані про підприємство в формі інтерактивного пошуку, а також створювати стандартні звіти (подання даних в розгорнутій формі і графічне представлення). У той час як інформаційна система для менеджменту надає доступ до узагальнених даних про всю фірму, система обліку за місцями виникнення прибутку забезпечує можливість роботи з оперативною інформацією (облік збуту, контроль результатів). Кожне місце виникнення прибутку пов'язане з областю контролю певної організаційної одиниці.

Модуль LO (Модуль загальної логістики) включає в себе інформаційну систему логістики, яка підтримує процеси прийняття рішень на основі аналізу співвідношень між планованими і реальними даними [8]–[10], [12], [15], [17].

Інформаційна система логістики вважається частиною системи «Відкрита база даних», яка аналогічна інформаційній системі персоналу і інформаційній системі фінансів. Система «Відкрита база даних» забезпечує взаємодію оперативних систем (модулі SD, MM, PP і т.д.), що є джерелами даних, і інформаційної системи для менеджменту. База даних логістики з її інформаційними структурами є основою для аналізу різноманітних параметрів в реальному масштабі часу і порівняння їх з контрольними значеннями (Online Analytical Processing або OLAP). Є такі інформаційні системи: закупівлі; виробництво; управління матеріальними запасами; продажу; техобслуговування; керування якістю.

Крім того, модуль LO включає основний запис матеріалів, які пов'язані практично з усіма модулями. До складу модуля LO також входять:

служба змін, управління партіями, конфігурація варіантів, проведення монтажу і ведення бази даних [8]–[10], [12], [15], [17].

Модуль SD (Модуль збуту) забезпечує виконання функцій підтримки збуту, відвантаження і транспортування товарів, а також фактурування, тобто весь бізнес-процес. Відповідна статистична інформація (інфраструктури) оновлюється для кожної операції фактури. Модуль SD взаємодіє з модулем ММ для перевірки наявності матеріальних запасів і відпуску товарів і з модулем FІ в частині кредитного менеджменту (ліміти кредитування) і обліку доходів.

Модуль РР (Модуль регулювання і планування виробництва) забезпечує управління дискретним виробництвом, а також управління виробництвом з безперервним циклом (РР-РІ) (наприклад, хімічної, фармацевтичної або харчової промисловості). Типовими об'єктами в цій галузі промисловості є рецептури, дані для управління технологічними та координаційними виробничими процесами, а також технологічні замовлення [8]–[10], [12], [15], [17].

В цілому модуль РР включає такі елементи, які взаємодіють з більшістю інших модулів:

- укрупнення планування збуту та виробництва;
- довгострокове планування;
- складання виробничої програми;
- планування потреби в матеріалах;
- планування виробничих потужностей;
- замовлення на виробництво;
- створення специфікацій;
- маршрутно-технологічні карти;
- робочі місця;
- виробництво з метою створення складського запасу;
- калькуляції витрат на виріб;
- автоматизоване планування; проведення монтажу.

Модуль ММ (Модуль управління матеріальними потоками) відповідає за управління матеріальними потоками [8]–[10], [12], [15], [17]. Найбільш важливими елементами цього модуля є:

- закупівлі – придбання основних матеріалів;
- обслуговування – пропозиції постачальників (каталог послуг, робіт), розміщення замовлень, приймання наданих послуг;
- управління запасами і інвентаризація;
- контроль рахунків та оцінка матеріалу;
- управління складами;
- кероване витратами планування потреб в матеріалах;
- атестація постачальника.

Модуль РМ (Модуль техобслуговування і ремонту обладнання). За допомогою модуля РМ можна автоматизувати процеси технічного обслуговування і ремонту обладнання (ТОРО) в компанії [8]–[10], [12], [15], [17]. Об'єктами техобслуговування може бути як устаткування, що належить компанії, так і обладнання замовників, яким надаються послуги з техобслуговування. Для вирішення цих завдань задаються технічні робочі місця, одиниці обладнання, специфікації ТОРО, технологічні карти ТОРО і т.д. Більш того, є можливість реалізувати планове попереджувальне техобслуговування і ремонт обладнання. Можна також створювати повідомлення ТОРО і замовлення ТОРО автоматично за вимірюваннями і показаннями лічильників. Нарешті, система дозволяє зберігати історію ТОРО для аналізу заходів з техобслуговування та збору повної інформації по витраті запасних частин.

Система сервісного обслуговування дає можливість в рамках комплексної системи SAP R / 3 управляти заходами сервісу, такими як гарантійний сервіс, технічне обслуговування та відновлення з посиланнями або без посилань на продукти. У систему вводяться заявки на обслуговування, які потім обробляються, тобто перетворюються в замовлення на обслуговування (сервісні заходи) або в замовлення клієнтів

(матеріали). Після підтвердження замовлення виконується фактурування з урахуванням витрат. З метою отримання повної і всебічної інформації даний модуль взаємодіє з іншими компонентами системи SAP R / 3, наприклад з модулями SD, FI і CO, MM, PS.

Модуль PS (Модуль проєктів) відповідає в системі SAP R / 3 за управління проєктами [8]–[10], [12], [15], [17]. Для відображення хронологічній послідовності дій можуть створюватися структурні плани проєкту, що відображають структуру проєкту, а також індивідуальні завдання і пакети робіт, елементи структурного плану проєкту та мережеві графіки. Крім того, є можливість планування витрат, порядку денного та бюджетів, підготовки звітів, можна додавати до плану вже виконані операції. Інформаційна система проєктів надає можливість підготовки різноманітних звітів для оцінки та вивчення результатів.

Модуль QM (Модуль керування якістю) підтримує базові операції з управління якістю відповідно до стандарту ISO 9000 [8]–[10], [12], [15], [17]. Високий ступінь інтегроване системи SAP R / 3, в особливості в частині логістики (модулі MM, PP і SD), в цьому випадку є важливою перевагою. На основі специфічних для модуля QM основних даних - методів тестування, контрольних ознак, контрольних каталогів, процедур і правил динамізації, а також даних про контроль якості в основних записах матеріалів (даних про закупівлі та даних тестів) - можливе проведення повного контролю якості (планування контролю, організація випробувань якості та аналіз) процесів логістики. Крім того, існує система повідомлень контролю якості, призначена для постійного підвищення якості. У цю систему можна заносити різні повідомлення результатів контролю якості. За замовчуванням є три типи повідомлень контролю якості:

- рекламація клієнта – містить посилання на такі об'єкти: замовник (клієнт), матеріал, партія, замовлення, постачання, контрольована партія, збутова організація, канал збуту, сектор;

- рекламація постачальнику – містить посилання на такі об'єкти: матеріал, партія, контрольована партія, постачальник, надходження матеріалу, замовлення на поставку, закупівельна організація;

- внутрішня помилка – містить посилання на такі об'єкти: матеріал, партія, контрольована партія, робоче місце, внутрішній партнер, виробничий замовлення.

Є можливість створювати і інші типи повідомлень.

Модуль PD (Модуль планування персоналу) відповідає за стратегію підбору і розстановки персоналу [8]–[10], [12], [15], [17]. Для вирішення цього завдання система дозволяє змодельовати внутрішню структуру компанії, що включає підрозділи і зв'язок між ними. В рамках системи також є можливість моделювання подальшого розвитку компанії (наприклад, зростання або реорганізацію). Засоби планування витрат на утримання персоналу, кадрове регулювання, управління заходами з підвищення кваліфікації також входять до складу модуля PD.

Модуль РА (Модуль адміністрування персоналу) відповідає за виконання таких функцій:

- впорядкування інформації про співробітників (діловодство);
- набір персоналу (управління даними про кандидатів і т.д.);
- облік робочого часу (введення і аналіз даних про робочий час, наприклад про роботи за змінним графіком);
- відрядна заробітна плата (час, відрядна і преміальна оплата);
- витрати на відрядження;
- розрахунок оплати праці.

Модуль СА (Компоненти, загальні для всіх додатків) відповідає за виконання таких функцій, загальних для декількох додатків [8]–[10], [12], [15], [17]:

- систематизація класів для групування незалежних об'єктів (наприклад, матеріалів);
- керування документами (графічними, текстовими і т.д.);

- ведення календаря термінів;
- резервування приміщень;
- взаємодія із зовнішніми системами (наприклад, системою архівування, мобільного введення даних або системою електронного обміну даними);
- інтерфейс з системами автоматизованого проектування;
- інтерфейси прикладного програмування (Application Programming Interface або API) для передачі даних в систему або з системи SAP;
- управління вихідними документами (печаткою, факсимільними повідомленнями, e-mail і т.д.);
- моделювання організаційної структури підприємства;
- інтерфейс IDoc.

Крім того, до складу модуля входять так звані перехресні розділи:

- особливі умови для країни (податки, платіжні операції, розрахунок зарплати і т.д.);
- концепція розподілених додатків (Application Link Enabling або ALE);
- сценарії потоку бізнес-операцій SAP;
- SAP Archive Link - система архівації даних на оптичні носії;
- SAP @ WEB - компоненти для розробки Internet-додатків SAP R / 3;
- системи архівування та видалення прикладних даних;
- інтерфейси прикладного бізнес-програмування (Business Application Programming Interface або BAPI) - стандартні інтерфейси для доступу до бізнес-об'єктів SAP.

Таким чином, досліджено основні модулі ERP системи R / 3, визначено їх структуру та призначення.

1.4 Функціональні можливості SAP/R3

SAP R / 3 – інструмент для управління бізнес-процесами. Інтегрована база даних і орієнтована на управління процесами структура системи SAP R / 3 надає можливості для оптимізації бізнес-процесів. Інструменти моделювання процесів спрощують процедуру їх реалізації в системі [8]–[10], [12], [15], [17].

Важливим питанням є оптимізація бізнес-процесів територіально розподілених підприємств. Однак для досягнення успіху при впровадженні проекту, крім використання інструментів аналізу та моделювання системи SAP R / 3 необхідним є дотримання відповідних процедур аналізу для кожної структурної одиниці (філії) компанії.

Бізнес процеси. В даний час в багатьох компаніях існують структури, що базуються на принципах поділу праці. В результаті цього багато процесів є надлишковими і призводять до уповільнення бізнесу. Виникли комунікаційні бар'єри між різними рівнями організаційних структур. Зруйнувати ці бар'єри покликані ERP системи [8]–[10], [12], [15], [17].

Функціональна переорієнтація ділових структур і процесів може допомогти вирішити ці проблеми [8]–[10], [12], [15], [17]. Ключовими положеннями переорієнтації бізнес-процесів є:

- структурування організацій на базі повних ділових процесів, а не окремих функцій;
- відмова від поділу завдання на основі максимально можливого розподілу робіт і функцій, необхідна оптимізація бізнес-процесів в цілому.

Основним терміном у цьому контексті є бізнес-процес. Бізнес-процес - це послідовність окремих операцій, сума яких дає відчутний для користувача результат [8]–[10], [12], [15], [17]. Користувач може бути внутрішнім або реальним замовником.

Такі типи бізнес-процесів часто виходять за рамки кількох організаційних структурних одиниць, а іноді і за рамки кількох організацій.

Метою є оптимізація бізнес-процесу в цілому, а не окремих бізнес-функцій. Важливим аспектом є спрощення та уніфікація бізнес-процесу як єдиного цілого, а не окремих його частин [8]–[10], [12], [15], [17]. Це, знову ж таки, відрізняється від моделі Тейлора, яка передбачає поділ кожного технологічного процесу на якомога більшу кількість елементів (технологічних переходів).

Незважаючи на існування природних відмінностей, процеси створення доданої вартості майже в усіх областях мають багато спільного [8]–[10], [12], [15], [17]. Орієнтоване на процеси структурування завдань в рамках процесу створення доданої вартості змінює систему поділу праці, а також вимоги до персоналу і організації в цілому наступним чином:

- об'єднання завдань і локалізація процесів прийняття рішень. Завданням оптимізації процесів є їх максимальне спрощення. Як правило, спрощення досягається шляхом зменшення ступеня поділу праці, тобто об'єднанням завдань. Кількість складових частин процесу зменшується, тоді як необхідний для кожної частини рівень кваліфікації зростає. У схемі Тейлора поділ праці передбачає поділ процесів прийняття та виконання рішень, а при організації процесів потрібно їх об'єднання;

- поділ праці за рамками організації. У цьому випадку розподіл праці не обмежується рамками однієї організації. Завдяки використанню інструментальних засобів оптимізації процесів з'явилися нові можливості поділу завдань в рамках компанії;

- мінімізація проміжної ланки між процесами. Ще однією рисою менеджменту, орієнтованого на процеси, є зменшення обсягу робіт за погодженням і координації. Такі роботи, як правило, збільшують витрати часу і коштів. Поділ завдань організовується так, щоб звести до мінімуму витрати на додаткові перевірки і координацію дій.

Зв'язок SAP R / 3 з Інтернетом. Використання можливостей з підключення SAP R/3 до Інтернету розширює можливості доступу користувачів до функцій SAP R / 3. Можна отримати доступ до системи з

будь-якого комп'ютера, підключеного до Інтернету. Таким чином, з'являється можливість отримання актуальної інформації, що зберігається в системі, з будь-якого місця і в будь-який час [8]–[10], [12], [15], [17]. Інтернет-додатки, як правило, працюють з меншим набором функцій, ніж ті, які підтримується системою SAP R/3. Це є перевагою для випадкових користувачів. Проте у такий спосіб вимоги користувачів, яким необхідний весь набір функцій системи, не задовольняються. Такі користувачі потребують прямого доступу до SAP R/3. З економічної точки зору вигідним є підключення хоча б частини користувачів до системи за допомогою Інтернет-технологій. Ціна ліцензій SAP для таких користувачів нижче, ніж для звичайних. Інтернет-браузер, як правило, можна отримати безкоштовно. Крім того, він не потребує технічної підтримки.

Концепція розподілених додатків. За допомогою технології ALE можна створювати складні системи управління підприємствами, що базуються на різних програмних продуктах. Існуючі в системі SAP R/3 технології дозволяють реалізовувати численні бізнес-процеси в таких областях, як облік і звітність, логістика та управління персоналом. Недоліки керування базами даних в розподілених системах компенсуються за рахунок розвиненого контролю цілісності і несуперечності інформації.

Замовники мають можливість розробляти різноманітні сценарії обробки даних [8]–[10], [12], [15], [17]. Технологія ALE дозволяє пов'язувати системи R/3 різних версій, а також зовнішні системи. Це відповідає вимогам поставленим до інтегрованих програмних систем для забезпечення стандартних способів взаємодії різних програмних продуктів.

Система SAP R/3 и сучасні вимоги. На думку більшості аналітиків, SAP R/3 є системою управління ресурсами підприємства та найбільш повно задовольняє всім вимогам компаній. Успіх компанії SAP AG говорить сам за себе. Компанія є світовим лідером, а система R/3 має значні переваги в порівнянні з конкурентами. Крім того, сучасні розробки, що ведуть до ще більшої гнучкості та відкритості (наприклад, взаємодія з Інтернетом, об'єктна орієнтація і модульність), значно збільшують її привабливість.

У системи SAP R/3 є свої недоліки. Наприклад, не можна скидати з рахунків її «мейнфреймове» минуле. Незважаючи на використання графічного інтерфейсу, деякі екрани як і раніше нагадують екрани терміналів великих машин і потребують поліпшення [8]–[10], [12], [15], [17]. Модульність системи не настільки висока, якою вона могла б бути: базові функції систем обліку і звітності та обліку витрат, управління матеріальними потоками і збуту пов'язані між собою набагато сильніше, ніж можна було б припускати. Ця ситуація не сильно змінилася з виходом версії 4.x.

Впровадження системи часто описують як складний, дорогий процес, що вимагає часових витрат. Однак частина цих труднощів на етапі впровадження часто буває викликаною специфічними вимогами кінцевих користувачів, а не самою системою. Проте повне і ефективне відтворення ділових процесів компанії, за допомогою засобів, що інтегровані в програмному забезпеченні є досить непростим завданням [8]–[10], [12], [15], [17]. SAP AG намагається зробити витрати на впровадження системи контрольованими за допомогою розробки Business Engineer (інструменту Бізнес-Інжиніринг - конфігурації і впровадження R / 3) і методології ASAP (методологія прискореного впровадження системи SAP R / 3). Використання цих інструментів сприяє значному прискоренню впровадження системи SAP R / 3.

Незважаючи на зазначені недоліки системи SAP R / 3, її функціональні можливості, відкритість і масштабованість роблять систему відповідним стратегічним рішенням для організацій, які бажають йти в ногу з сучасними вимогами і відкривати нові можливості.

Інтеграція бізнес-процесів. SAP R / 3 сама по собі структурована як програмна система, орієнтована на управління процесами. Те, що на концептуальному рівні бізнес-процеси вже відтворені в системі, а також наявність інтегрованого сховища даних і засобів управління інформаційними потоками, означає, що при впровадженні SAP R / 3, як правило, не потрібно відтворювати бізнес-процеси «з нуля». Достатньо лише налаштувати систему під потреби конкретної організації [8]–[10], [12], [15], [17].

Варто відзначити, що для автоматизації діяльності складних територіально розподілених підприємств, діяльність яких характеризується складними бізнес-процесами, необхідно розробляти нові, а також модифікувати існуючі програмні модулі SAP R / 3 [8]–[10], [12], [15], [17].

Різні відкриті інтерфейси, такі як IDoc, ALE, EDI і Internet, дозволяють організовувати взаємодію з віддаленими підрозділами та діловими партнерами. Це дає можливість інтеграції бізнес-процесів за рамками організації.

Досліджено та проаналізовано функціональні можливості ERP системи SAP/R3. Проаналізовано можливості автоматизації та інтеграції бізнес процесів, роботи з середовищем Internet, концепцію розподілених програмних застосунків.

1.5 Висновки за розділом 1

Розглянуто сучасні корпоративні інформаційні системи та їх застосування для автоматизації діяльності складних територіально розподілених підприємств. Проаналізовано загальні переваги та недоліки відомих ERP систем. Визначено, що вибір ERP-рішення є складним і комплексним завданням, що вимагає серйозного дослідження організації і чіткого формулювання вимог до корпоративної інформаційної системи. Як базис обрано систему SAP R/3, яка має широку сферу застосування (оборонні підприємства, компанії нафтогазового комплексу, металургія, енергетика телекомунікації, банківський сектор тощо), є зручною у впровадженні та експлуатації, надає широкі можливості щодо підтримки виробничих та організаційних процесів, а також забезпечує можливості створення власних програмних модулів та впровадження їх в існуючу систему.

Досліджено основні модулі ERP системи R / 3, визначено їх структуру та призначення. Проаналізовано можливості автоматизації та інтеграції

бізнес процесів, роботи з середовищем Internet, концепцію розподілених програмних застосунків.

Визначено, що для автоматизації діяльності складних територіально розподілених підприємств, діяльність яких характеризується складними бізнес-процесами, необхідно розробляти нові, а також модифікувати існуючі програмні модулі SAP R / 3.

2 ВИБІР ТА ОБҐРУНТУВАННЯ СТРУКТУРИ СИСТЕМИ, ЯКА ПРОЄКТУЄТЬСЯ

Вибір базових програмних продуктів є складним завданням з багатьма параметрами і є одним з важливих етапів при розробці корпоративних інформаційних систем. Цей вибір повинен задовольняти як поточним, так і майбутнім потребам підприємства, при цьому слід враховувати фінансові витрати не тільки на придбання, але і на розробку системи, супровід, а також навчання персоналу.

2.1 Аналіз та вибір СКБД корпоративної інформаційної системи

Важливим елементом корпоративної інформаційної системи є система керування базою даних (СКБД). Необхідно вибрати таку СКБД, яка не тільки в повній мірі задовольняє поточним потребам, а й забезпечить можливості для подальшого розширення й інтеграції системи [1]–[3].

Найбільш простий підхід при виборі СКБД заснований на оцінці того, якою мірою існуючі системи задовольняють основним вимогам проекту інформаційної системи. Більш складним і дорогим варіантом є створення випробувального проекту на основі кількох СКБД, а потім зробити вибір на користь найбільш підходящої системи. Але і в цьому випадку слід звзвити коло можливих систем, опираючись на певні критерії відбору. Перелік вимог до СКБД може змінюватися в залежності від поставлених цілей. Проте, можна виділити кілька груп критеріїв: продуктивність, надійність, особливості розробки додатків, контроль роботи системи, вимоги до робочого середовища, особливості архітектури і функціональні можливості, моделювання даних, змішані критерії [1]–[3].

Моделювання даних включає в себе такі критерії:

– модель даних, що використовується. Існує кілька моделей даних: ієрархічна, мережева, реляційна, об'єктно-реляційна, об'єктна. Вибір моделі

здійснюється виходячи з основних потреб інформаційної системи на початковому етапі проектування;

- підтримувані типи даних. Деякі сучасні системи крім загальноприйнятих базових типів даних (integer, character) підтримують додаткові типи даних, а також можливості розширення типів даних;

- реалізація мови запитів. Практично всі реляційні СКБД сумісні з тією чи іншою специфікацією стандартного мови доступу до даних SQL. Крім того, у багатьох системах реалізовані деякі розширення даного стандарту (наприклад, свого часу таким розширення були тригери);

- засоби пошуку. Деякі сучасні системи маю вбудовані додаткові засоби для контекстного пошуку;

- тригери, процедури зберігання, обмеження і затвердження як конструкції активних середовищ. Традиційні бази даних є пасивними, тобто зміни і вибірка даних здійснюються тільки під впливом зовнішнього джерела. Однак останнім часом важливу роль набуває концепція активних баз даних, коли в базу даних закладаються можливості управління даними. У створенні активних середовищ допомагають кілька основних конструкцій бази даних: обмеження, затвердження, збережені процедури і тригери. Тригер це програма бази даних, яка автоматично виконується при деяких подіях (створення запису, редагування записів, видалення запису і ін.). Процедура зберігання це програма, яка відноситься до бази даних, а не до зовнішньої програми зберігання в системі. Обмеження і затвердження це відносно прості конструкції, що мають вигляд від специфікації зв'язків первинного та зовнішнього ключів, використовуваних в обмеженнях цілісності за посиланням, до SQL-подібних обмежень, які використовуються для перевірки приналежності значень заданому діапазону або списку значень.

Особливості архітектури і функціональні можливості:

- мережеві можливості – практично всі сучасні СКБД дозволяють використовувати широкий діапазон мережевих протоколів і служб для роботи і адміністрування;

– масштабованість – як правило, система повинна передбачати зростання інформаційної системи. Розрізняють збільшення обсягу збережених даних, числа користувачів і обсягу оброблюваної інформації;

– розподіленість – як правило, інформаційна система створюється з метою об'єднання інформації підприємства, але найбільш проста реалізація у вигляді централізації даних на одному сервері не завжди може бути застосована з міркувань підвищених вимог до відмово стійкості або сильного територіального розподілу підрозділів. Механізми створення розподілених баз даних в різних системах реалізовані по-різному.

– максимальний розмір бази даних. При виборі системи слід враховувати, чи зможе база даних вмістити плановані обсяги інформації.

До критеріїв контролю роботи системи можна віднести:

– засоби моніторингу – сучасна система повинна володіти засобами моніторингу основних параметрів бази даних для адміністрування і прийняття рішень з оптимізації комплексу;

– автоматичне налаштування – можливість автоматично підлаштовуватися для досягнення максимальної ефективності.

Критерії, що характеризують вимоги до робочого середовища:

– мінімальні вимоги до апаратного забезпечення. Оскільки більшість сучасних інформаційних систем побудовані на основі архітектури клієнт-сервер, розрізняють вимоги до обладнання клієнтської і серверної частини;

– підтримувані апаратні платформи та операційні системи. Сучасні СКБД можуть працювати на різних апаратних платформах під управлінням декількох операційних систем. Це дає можливість вибирати операційні системи і апаратну конфігурацію для досягнення максимальної надійності і продуктивності в умовах обмеженого фінансування.

До критеріїв продуктивності варто віднести такі:

– рейтинг ТРС. Існують кілька тестових рейтингів. Одним з найпопулярніших і об'єктивних є ТРС. Він являє собою комплексний аналіз СКБД і апаратури. Існує два види рейтингу ТРС що оцінюють

продуктивність та ставлення продуктивність до ціни комплексу. Крім цього рейтинг також розбитий на категорії з точки зору архітектури серверної частини (кластерні системи, не кластерні системи), виробника серверної частини (Compaq, IBM, HP, Dell і т.п.), а також області застосування (OLTP, електронна комерція, підтримка прийняття рішень, бізнес звітність і т.п.);

- можливості вбудованої оптимізації запитів. Сучасні СКБД повинні містити оптимізатор запитів, який в результаті семантичний перетворень з урахуванням статистичних даних виробляє оптимальний план виконання запиту;

- можливості паралельної архітектури. Для підвищення продуктивності в багатьох сучасних СКБД застосовується паралельна обробка. Наприклад, можлива підтримка технологій паралельної обробки послідовності запитів на декількох процесорах або використання декількох комп'ютерів об'єднаних в паралельний сервер бази даних;

- можливості перерозподілу обробки між сервером і клієнтом. Деякі СКБД дозволяють частково або повністю переносити обробку даних з клієнта на сервер. В результаті стає можливим вибір з побудови конфігурації з «товстим» або з «тонким» клієнтом. У деяких СКБД можлива побудова додатків, які в залежності від ситуації самі вирішують, яка частина обробки повинна бути перенесена на сервер або до іншого клієнта.

Критерії надійності:

- модель транзакцій. Існують різні моделі транзакцій – від найпростіших, наприклад плоских, до більш витончених, таких як вкладені або багатоланкові. Велике значення має правильний вибір однієї зі складних моделей транзакцій з урахуванням специфіки діяльності конкретної організації. Наприклад, вкладені транзакції надають виключно гнучкі можливості управління субтранзакціями, але вони складні в реалізації і не у всіх випадках ці можливості необхідні. Для побудови розподіленої архітектури СКБД повинна підтримувати розподілені транзакції;

– захист інформації від несанкціонованого доступу. Будь-яка сучасна інформаційна система повинна мати можливість розмежування доступу до інформації. Існують різні рівні захисту інформації. У більшості випадків мало простої ідентифікації користувачів при вході в систему, тому СКБД повинна підтримувати засоби, за допомогою яких для кожного користувача визначаються повноваження доступу до даних і модифікації даних;

– резервне копіювання. У деяких критичних ситуаціях відновлення втрачених даних, можливо тільки з використанням резервної копії. СКБД повинна дозволяти варіювати плани резервного копіювання в залежності від обмежень на час відновлення, вартість зберігання і можливу втрату частини інформації (наприклад, за останній день). Деякі СКБД дозволяють виконувати резервне копіювання без зупинки системи. Крім цього існують технології, що знижують вартість зберігання. Наприклад, можна привести інкриментальне резервне копіювання, при якому зберігаються тільки ті блоки бази даних, які змінилися з моменту останнього резервного копіювання;

– відновлення після збоїв. Програмні або апаратні збої можуть призвести до порушення цілісності інформації. В цьому випадку подальше функціонування системи залежить від реалізації механізмів відновлення. Наприклад, ведення журналу незавершених транзакцій дозволяє автоматично відновити цілісність бази даних після збою, який виник при відключенні електрики.

До критеріїв, що характеризують особливості розробки програмних застосунків відносять:

– спектр підтримуваних мов програмування. СКБД може підтримувати кілька мов програмування. Пропонує широкий ряд мов програмування підвищує доступність системи для розробників і дозволяє створювати оптимальні по продуктивності і функціональності додатки;

– покоління підтримуваних мов програмування. Функціональність програми, а також її продуктивність і час розробки, істотно залежать від покоління мови програмування, що використовується. Застосування

проблемно-орієнтованої мови четвертого покоління (4GL), яка оперує конкретними поняттями предметної області, істотно підвищує функціональність програми. Один оператор 4GL може відповідати десяткам операторів 3GL. 4GL містить засоби візуальної розробки, що дозволяє вести розробку в стилі RAD (Rapid Application Development - швидка розробка додатків). Якщо ж прикладна область орієнтована на Internet, то потрібно застосовувати мови 3GL, заснованих на мові Java;

- засоби розробки Web-додатків. При проектуванні Internet-орієнтованого додатку, бажано, щоб СКБД містила спеціальні засоби розробки Web-додатків;

- власні засоби розробки додатків. Багато постачальників створили спеціальні засоби розробки додатків для своїх систем, які дозволяють найкращим чином використовувати можливості СКБД. Це можуть бути засоби створення звітів, засоби розробки Web-додатків, мова 4GL;

- підтримка багатьох мов. Підтримка великої кількості національних мов, що розширюють сферу застосування СКБД і побудованих на її основі додатків.

До інших критеріїв оцінювання СКБД можна віднести такі:

- підтримка національної мови. Підтримка національної мови може істотно позначатися на вартості розробки і супроводу додатків;

- якість і повнота документації. На жаль, не всі системи мають повну і детальну документацію;

- вартість системи. При оцінці вартості системи повинні враховуватися ціна базових продуктів та вартість їх супроводу. Крім цього, слід враховувати вартість розробки програми для обраної СКБД. На вартість системи сильно впливає модель формування вартості базових продуктів. Наприклад, як правило, ціна конкретного продукту залежить від числа користувачів. Тому, в деяких випадках слід вибирати більш дорогу СКБД, яка дозволить знизити фінансові вкладення при майбутньому розширенні інформаційної системи.

На сьогоднішній день відомо багато серверних СКБД. Широкого розповсюдження набули такі: Oracle, Microsoft SQL Server, Informix, Sybase, Progress [1]–[17].

Oracle була першою комерційною реляційної СКБД, яка підтримувала мову SQL. Oracle була першою компанією, яка створила СКБД, що використовувала надані іншими серверними платформами засоби для паралельних обчислень. Oracle Parallel Server дає можливість декільком процесорам звертатися до однієї бази даних, що дозволяє забезпечити високу швидкість обробки транзакцій. Останні версії дають можливість здійснити декомпозицію операцій з великими обсягами даних з метою паралельного виконання їх на кількох процесорах. Існують версії цієї СКБД для різних серверних платформ: UNIX, VMS, Windows і ін. СКБД Oracle підтримує об'єктні розширення і відповідні типи даних, такі як вкладені таблиці, масиви, об'єкти і ін. Є можливість створення таблиць, що містять агреговані дані і можливість часткового їх поновлення при зміні даних, на підставі яких вони обчислені. Підтримуються Java, XML, C/C ++, PL / SQL.

СКБД дає можливість використовувати додаткові засоби забезпечення безпеки, наприклад шифрування даних, підтримку SSL, ролі на рівні бази даних і на рівні підприємства. Розробляючи власні засоби розробки, Oracle надає своїм користувачам можливість створювати клієнтські програми за допомогою інших засобів. Крім стандартного клієнтського API (Oracle Call Interface), клієнтська частина Oracle містить об'єктну модель (Oracle Objects for OLE), що дозволяє використовувати клієнтську частину Oracle в якості набору COM-об'єктів для доступу до даних. Крім цього клієнтська частина Oracle може містити ODBC-драйвер для доступу до даних цієї СКБД.

З готових інформаційних систем на базі Oracle слід особливо відзначити одну з найбільших систем управління підприємством SAP / R3. Також нерідко використовуються готові рішення від самої Oracle Corporation, об'єднані під загальною назвою Oracle Applications, такі як Oracle Financials, Oracle Human Resources, Oracle Market Management, Oracle Project Systems і ін.

Microsoft SQL Server – відрізняється зручністю призначеного для користувача інтерфейсу утиліт адміністрування в поєднанні з досить високою продуктивністю і відносно невисокою вартістю експлуатації. СКБД має прості утиліти адміністрування. Сервіси перетворення даних (Data Transformation Services) полегшують перенесення даних в SQL Server з інших типів СКБД. Підтримуються розподілені запити і транзакції, а також розширені функції для створення Web-додатків.

Замість Microsoft SQL Server в якості вбудованої СКБД для настільних додатків і додатків для невеликих робочих груп можна використовувати Microsoft Data Engine (MSDE). Це настільний сервер баз даних сумісний з Microsoft SQL Server і призначений для використання в настільних системах або в мережевих додатках з невеликим (до 2 Гбайт) обсягом даних і малою кількістю користувачів.

Клієнтські програми для Microsoft SQL Server і MSDE можна створювати як за допомогою засобів розробки Microsoft (Visual Basic, Visual C++, Access, Visual FoxPro), так і за допомогою засобів розробки інших виробників. Для цієї мети є ODBC-драйвер і OLE DB-провайдер, а також системи, що містять їх набір бібліотек Microsoft Data Access Components (MDAC), що дозволяє використовувати в засобах розробки об'єкти ActiveX Data Objects (ADO). На відміну від Oracle, Microsoft не виробляє засобів розробки, що використовують ту ж саму мову програмування, що і мова для створення коду тригерів і збережених процедур, однак розробляє засоби налагодження серверного коду (наприклад, SQL Server Debugger входить до складу Visual Basic і Visual C ++).

Вище було сформульовано загальні критерії вибору СКБД для побудови програмних застосунків. Якщо ж необхідно вирішити задачу вибору СКБД для корпоративної інформаційної системи, то деякі з наведених критеріїв стають більш суттєвими.

Перш за все, слід зробити попередній відбір, відкинувши СКБД, що не відповідають класу проектованої системи. Очевидно, інформаційна система

середньої та великої за величиною корпорації не можуть будуватися на основі «настільних» СКБД, таких як Microsoft Access. Якщо ж припустити розширення компанії в майбутньому, то доцільно відразу виключити їх з розгляду.

Виходячи з аналізу загальних вимог, висунутих до надійності і продуктивності, лідирують продукти Oracle, MySQL та Progress. Вони можуть забезпечити належну масштабованість майбутньої системи. Саме ці СКБД дозволяють на високому рівні створювати складні розподілені архітектури. Набагато більшу увагу слід звертати на моделювання даних. Наприклад, в тому випадку, коли необхідно працювати з великими обсягами мультимедійної інформації, слід вибирати Oracle. Для більшості додатків цілком достатньо моделі даних DB2 або MySQL. Багато з розширених можливостей (об'єктні розширення) наприклад, реалізованих визнаним лідером в розробці СКБД - корпорацією Oracle - швидше за все не будуть використані. Oracle, MySQL і Progress підтримують величезні розміри бази даних, але, як правило, реальний обсяг даних ніколи не досягне і сотої частки дозволеного [1]–[17].

Велике значення в більшості випадків відіграють особливості розробки додатків. СКБД Oracle, MySQL і Progress підтримують широкий набір мов програмування, але тільки в Progress активно використовуються мови 4GL. Застосування мови 4GL серйозно знижує витрати на розробку і дозволяє створювати більш гнучкі додатки.

Далеко не останню роль при виборі СКБД грають економічні показники. Цілком природно, коли компанія купує нехай не самий сучасний, але дешевший продукт, який може задовольнити всі потреби підприємства, як зараз, так і в доступному для огляду майбутньому. Наприклад, Progress зовсім не поступається Oracle в продуктивності для більшості завдань, які можуть виникнути при побудові КІС. При цьому Progress дешевше, з точки зору загальної вартості користування базою даних. За результатами останніх років Progress є лідером на ринку вбудованих баз даних [1]–[17].

Нижче наведено порівняльний аналіз систем керування базами даних у вигляді табл. 2.1.

Таблиця 2.1 – Порівняльний аналіз систем керування базами даних

Критерії порівняння	Система керування базами даних		
	MS Access	MySQL	Progress
Багатоплатформність	–	+	+
Швидкість розробки	±	+	+
Надійність	±	+	+
Мережеві можливості	±	+	++
Масштабованість	±	+	++
Розроблення графічного інтерфейсу користувача	+	±	+
Доступність (вимоги до розповсюдження)	±	+	±
Можливість створення складних БД для ERP-систем	–	+	++

Як видно з таблиці, в якості СКБД для створення модулів ERP системи доцільно обрати Progress як надійну систему швидкої розробки з високими можливостями масштабованості, що дозволить обробляти дані складних територіально розподілених підприємств.

2.2 Вибір мови програмування

Виконаємо порівняльний аналіз різних мов програмування, які можуть застосовуватися для створення програмних за стосунків. Результати такого аналізу наведено у табл. 2.2.

Таблиця 2.2 – Порівняльний аналіз мов програмування

Критерії порівняння	Мова програмування		
	C#	Java	ABAP
Мережеві можливості	±	++	+
Швидкість розробки	++	+	+
Розроблення графічного інтерфейсу користувача	+	+	+
Ефективність інтеграції з SAP R/3	±	+	++
Можливість створення складних програмних модулів для ERP-систем	—	+	++

Як видно з табл. 2.2 як базис для створення модулів ERP системи доцільно обрати мову ABAP (Advanced Business Application Programming), яка представляє собою пропріетарну внутрішню мову програмування високого рівня для розробки SAP систем [15]. Поряд з Java є мовою створення додатків для SAP NetWeaver Application Server. Синтаксис найбільш близький до мови програмування COBOL.

Всі програмні застосунки R/3 і навіть частини базової системи були розроблені на ABAP. Гнучкість і простота використання мови в поєднанні зі спеціальним набором команд дозволяють як невеликим, так і великомасштабним комерційним підприємствам швидко розробляти і оперативно впроваджувати системи R / 3.

ABAP має цілий ряд переваг, які роблять її надзвичайно зручною для створення таких складних систем, як R/3. Вона відноситься до розряду мов, в яких реалізовано управління подіями, тобто виконання програми координується діями користувача і системними подіями. Важливою особливістю мови є концепція застосування внутрішніх таблиць, суть якої в

відображенні постійних таблиць бази даних на об'єкти, що існують під час виконання, і навпаки. Мова ABAP/4 - це не окремо використовуваний засіб, а частина загального середовища розробки Development Workbench, в яку входить також широкий набір інструментів для планування і реалізації програм. Мова ABAP/4 підходить як для створення нових додатків в архітектурі клієнт / сервер, так і для розширення можливостей діючих модулів R/3, а такі засоби, як Remote Function Call (RFC), дають можливість розробляти розподілені додатки. Крім того, додатки, написані на ABAP/4, підтримують різні СКБД і ОС, що полегшує їх інтегрування в інформаційну систему компанії. Надійність експлуатаційних якостей мови ABAP/4 ґрунтується на різноманітті функцій і високої продуктивності системи R/3 [15].

2.3 Перенесення даних із зовнішніх джерел в SAP систему

Для перенесення даних із зовнішніх систем в SAP систему ефективним засобом є модуль LSMW [1]–[3]. Дані можуть бути введені за допомогою наступних технологій: Batch Input, Direct Input, BAPIs або IDocs. Далі інструкція як цим інструментом користуватися.

Перевагами механізму LSMW є такі:

- LSMW - частина SAP системи не залежна від платформи;
- велика кількість можливостей по перетворенню даних;
- підтримка основних технологій введення: Bath Input, Direct Input, BAPIs, IDocs;
- зручний путівник по міграції даних;
- є безкоштовним інструментом для партнерів SAP.

Основні функції:

- імпорт даних (дані з файлів, електронних таблиць);
- перетворення даних в необхідний формат;
- імпорт даних в SAP систему.

До того як застосовувати LSMW необхідно переконатися в наступному:

- SAP система остаточно налаштована і готова до введення;
- вирішити, чи має сенс використовувати інструмент, його використання виправдане на великих обсягах даних. Швидкість імпорту для режиму Direct Input і Bath Input приблизно 10000 записів на годину, величина може змінюватися в залежності від апаратної частини сервера;
- визначитися з транзакціями в SAP системі, через які буде виконувати імпорт даних. Чи створені стандартні об'єкти Direct Input, Batch Input, чи є необхідні BAPI;
- протестувати вибрані транзакції вручну, визначте, які поля є обов'язковими до заповнення;
- LSMW не одержує дані із зовнішніх систем, їх необхідно попередньо підготувати. Формат файлу повинен відповідати можливостям читання файлів LSMW. Наприклад: кодування повинне бути ASCII або IBM;
- визначитися з методом імпорту даних (BAPI, Direct Input і т.п.).

Таким чином, для перенесення даних із зовнішніх систем в SAP систему ефективним засобом є модуль LSMW.

2.4 Розробка моделі модуля SAP R/3 територіально розподіленого підприємства

Розроблювана модель модуля SAP R/3 територіально розподіленого підприємства буде складатися з набору діаграм, що характеризують особливості бізнес-процесів підприємства, та дозволяють розробляти програмне забезпечення для їх автоматизації.

Розроблений модуль забезпечує роботу декількох територіально розподілених підрозділів. У зв'язку з цим стає неможливим застосування

централізованої архітектури бази даних. Інформаційні ресурси повинні стати розподіленими, це дозволяє забезпечити такі позитивні особливості:

- локальна автономність. Локальні дані повинні знаходитися під локальним володінням і управлінням, включаючи функції безпеки, цілісності, представлення даних. Винятком може бути ситуація, коли обмеження цілісності охоплюють дані декількох вузлів і коли керування розподіленої транзакцією здійснюється деяким зовнішнім вузлом;

- безперервність функціонування. Система не повинна зупинятися в разі необхідності додавання нового вузла або видалення в розподіленого середовища деяких даних і здійснення переходу до нової версії СКБД на окремому вузлі;

- незалежність від місця розташування. Користувачі і програмні застосунки не зобов'язані знати про те, де фізично розташовуються дані;

- незалежність від фрагментації. Фрагменти бази даних обробляються таким чином, щоб користувачі і програмні застосунки могли взагалі нічого не знати про це. Більш того необхідно обходити при обробці запитів фрагменти, які не мають до них ніякого відношення;

- розподілена обробка запитів та керування розподіленими транзакціями. Необхідно використовувати механізми управління транзакціями і одночасним доступом, що дозволяють виявляти і вирішувати тупикові ситуації переривання після закінчення тимчасових інтервалів, фіксувати і відкочувати розподілені транзакції;

- незалежність від обладнання, операційних систем, мережі та локальних СКБД.

На практиці досить складно реалізувати систему, що задовольняє всім цим пунктам.

Для вирішення завдання імпорту розробляється програма пакетного введення в транзакцію FB01 «Проводка документа». Завданням програми є

зчитування інформації з файлу баз даних (* .dbf), сформованого системою обліку з / пл, і заповнення відповідних полів в таблицях SAP.

Другим завданням є взаємодія підприємства з наступними зовнішніми об'єктами: постачальники, банки, податкова інспекція, оптовий і роздрібний покупець. Для модуля Контролінг SAP необхідно також, щоб кожна проводка за витратними рахунками була прив'язана до відповідного Замовлення (CO замовлення), що дозволило б відобразити витрати по зарплаті в контролінгу коректно.

Для імпорту проводок з заробітної плати користувач, запускає транзакцію ZSIM.Програма імпортує дані з файлу БД у проміжну таблицю, звідки відповідно ознаки дані заносить у AVL таблицю, звідки після аналізу та редагування полів «Сума», «Замовлення» і «Рахунок» відбувається пакетний введення в транзакції FB01. Після перенесення даних виводиться звіт.

Програма використовує 3 таблиці: ZSIM_SETTINGS; ZSIM_T; ZSIM_TSUMM.

У першій таблиці (ZSIM_SETTINGS) зберігаються правила підміни значень при перенесенні їх у проміжну таблицю і правила винятків проводок, які не переносяться в проміжну таблицю. Таблиця ZSIM_T – проміжна таблиця, в якій зберігаються дані, вивантажені з БД. Таблиця ZSIM_TSUMM – таблиця підсумкових значень, в яку після закінчення обробки окремих позицій переносяться підсумкові значення.

Підприємство взаємодіє з такими зовнішніми об'єктами: постачальники, банки, податкова інспекція, оптовий і роздрібний покупець. В сучасних умовах економічно доцільно виділити в якості окремого підрозділу центр управління мережею структурних підприємств, який візьме на себе частину функцій по роботі з постачальниками і повністю звільнить

підприємство від взаємодії з банками і податковою інспекцією. Структура взаємодії з зовнішніми об'єктами представлена на рис.2.1.

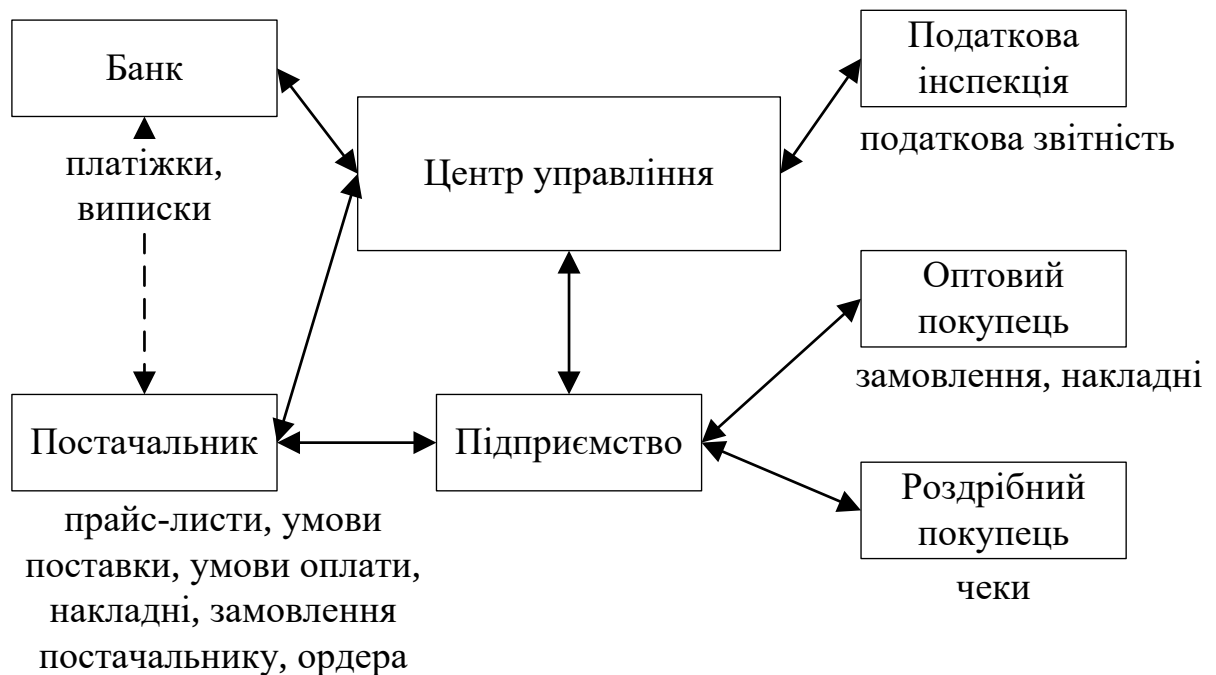


Рисунок 2.1 – Структура взаємодії з зовнішніми об'єктами

Запропонована структура може працювати тільки при повній автоматизації кожного структурного підприємства і при налагодженому оперативного обміну інформацією. Тобто, потрібна корпоративна інформаційна система здатна вирішити поставлені завдання.

Що стосується структури апаратних засобів, то вона покликана, в першу чергу, забезпечувати роботу торгового процесу підприємства. Тобто необхідно організувати ефективне функціонування пристроїв front-office, а саме: касових терміналів, електронних ваг і принтерів етикеток. Структуру апаратних засобів можна вважати ефективною, якщо вона робить можливим оперативний обмін даними і безперервну роботу пристроїв front-office. У зв'язку з цим вводиться виділений сервер управління касами, вагами і принтерами. Сервер кас, ваг і принтерів бере на себе всі функції з обміну даними з пристроями front-office, в тому числі авторизацію пластикових карт.

Запропоновану діаграму взаємодії апаратних засобів ERP системи наведено на рис.2.2.

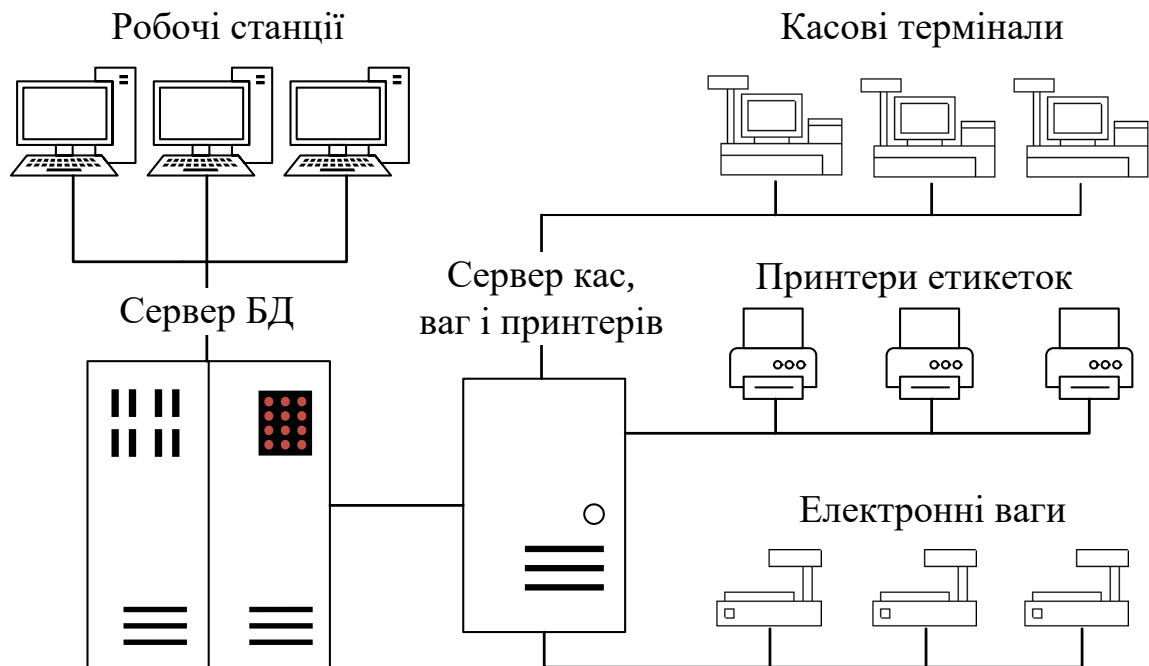


Рисунок 2.2 – Діаграма взаємодії апаратних засобів ERP системи

Розроблюваний модуль ERP системи являє собою набір інтегрованих програм, які дозволяють створити єдине середовище для автоматизації планування, обліку, контролю і аналізу всіх основних бізнес операцій в масштабі підприємства. Відповідно, програмне забезпечення повинно вирішувати завдання трьох основних груп: облік і контроль; аналіз і планування; аналіз і корекція плану.

На великих територіально розподілених підприємствах, як правило, існують кілька видів планів, що описують шляхи та напрямки діяльності підприємства. До таких планів відносяться: бізнес-план, маркетинговий план, виробничий план, основний виробничий план-графік, план закупівлі матеріалів, оперативний план виробництва. Узагальнену модель системи виробництва на основі планування наведено на рис. 2.3.

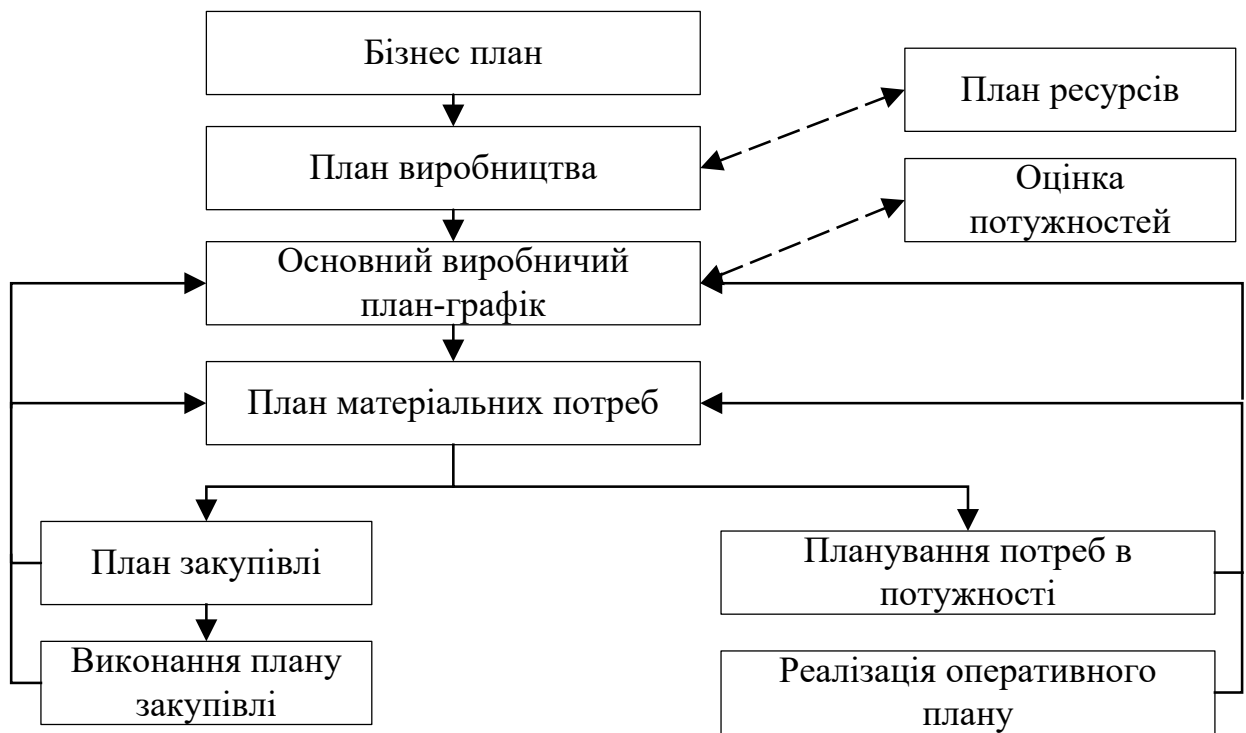


Рисунок 2.3 – Узагальнена модель системи виробництва
на основі планування

Узагальнена модель системи виробництва на основі планування дозволяє об'єднати інформацію з різних планів підприємства, що дозволить інтегрувати їх в ERP систему для підвищення ефективності бізнес-процесів підприємства.

Бізнес-план являє собою довгостроковий, стратегічний план діяльності всього підприємства і його розробка є привілеєм вищого керівництва підприємства. Пункти плану можуть включати розвиток нових ринків збуту, розвиток нових сімейств виробленої продукції, комбінацій бізнесу, реструктуризації бізнесу і т. д. Форма плану, як правило, відповідає основним фінансовим формам: баланс підприємства, звіт про фінансові результати, звіт про рух грошових коштів і виражається у фінансових показниках.

Маркетинговий план (не відображено на рисунку), є результатом бізнес-плану і повинен відповідати показникам бізнес-плану і плану

виробництва. Маркетинговий план може включати в себе методи і засоби просування продукції на ринку, організації продажів і розподілу.

План виробництва затверджується на основі бізнес-плану і може охоплювати період до декількох років. Як правило, виробничий план складається в об'ємних показниках (тони, штуки) виробництва сімейств продукції в залежності від особливостей бізнесу підприємства. План виробництва відображає загальні виробничі показники і об'єднує планові дані, що стосуються обладнання, основних інструментів, людських ресурсів, матеріалів і комплектуючих. Іноді перераховані дані становлять основу окремого так званого плану ресурсів підприємства, на який посиляється виробничий план. План виробництва і розподіл ресурсів у свою чергу ініціюють розробку фінансового плану або бюджетів, заснованих на основних нормативних показниках.

Основний виробничий план-графік (Master Production Plan - MPS) складається на основі виробничого плану, як правило, на декілька місяців або на рік в залежності від суті виробничих процесів. MPS виражається в об'ємних показниках продукції, що реально випускається і розробляється відповідно до маркетингового плану, який передбачає, що має бути вироблено і продано або на основі прогнозу, або на основі підтверджених клієнтських замовлень, або їх комбінації. Як правило, в той же час здійснюється попередня оцінка виробничих потужностей для визначення можливості виконання заданих показників виробництва з використанням доступних в даний час обладнання і трудових ресурсів.

Планування потреб в матеріалах (Material Requirement Planning – MRP). Процедура MRP перетворює графік виробництва кінцевої продукції (MRS) в графік замовлень на закупівлю комплектуючих, матеріалів і замовлень на виробництво компонент з урахуванням вимог щодо термінів і технології виробництва кінцевої продукції. Процедура даного перетворення будується на основі складу виробленого виробу, відомості матеріалів, що існують і замовлених запасів, особливостей планування і т. д.

Планування потреб у виробничих потужностях (Capacity Requirement Planning – CRP). Спільно з іншими вихідними даними процедура планування матеріальних ресурсів формує попередній графік випуску виробничих замовлень. Процедура планування потрібних виробничих потужностей здійснює аналіз згаданих виробничих замовлень з урахуванням існуючого незавершеного виробництва для оцінки можливості використання виробничих ресурсів в потрібні періоди для реалізації показників основного виробничого плану-графіка. У разі виявлення нестачі або незбалансованого завантаження устаткування необхідно приймати рішення або по ресурсам, або по планованим виробничими замовленнями, або по зміні MPS.

План закупівлі. Результатом процесу MRP планування є графік придбання матеріалів і комплектуючих, який використовується службами закупівлі для організації процесу закупівлі. Дані MRP планування є основою для організації вибору постачальника, узгодження договорів і, в кінцевому рахунку, визначають можливість задоволення запитів виробництва. Якщо виникають проблеми з придбанням всієї необхідної для виробництва номенклатури матеріалів і комплектуючих, приймається рішення про ревізію MRP запланованих виробничих замовлень або зміну основного виробничого плану-графіка.

Виконання плану закупівлі. Процедура реалізації MRP планування в частині закупівлі матеріалів і комплектуючих вимагає виконання відповідних витрат після їх затвердження відповідними повноважними особами.

Оперативний план виробництва. Формується на основі MRP для виробництва необхідного для випуску готової продукції запасу вузлів, компонент на власних потужностях.

Виконання оперативного плану виробництва. Процедура реалізації MRP плану в частині виробництва вузлів і компонентів ґрунтується на MRP графіку випуску виробничих замовлень з визначенням, яким чином виробничі замовлення випускаються, відстежуються і контролюються. Дана операція часто називається "Диспетчеризація виробництва" (PAC –

Production Activity Control). Процес диспетчеризації виробництва по зворотного зв'язку може впливати на перепланування діяльності на більш високих рівнях.

Таким чином, можна помітити, що узагальнена модель системи виробництва на основі планування описується циклічними процесами на декількох рівнях, особливо на рівні реалізації планів закупівлі та власного виробництва, що впливають на вищі рівні в зв'язку з реальними можливостями і практикою з процесам придбання і виробництва. Такого роду зворотний зв'язок часто трактується як замкнута петля в системній моделі.

Базовим об'єктом управління є замовлення. Процес замовлення забезпечує організацію ієрархічної структури системи планування та обліку, реалізованих у ERP. Діаграма потоку замовлень представлена на рис. 2.4.

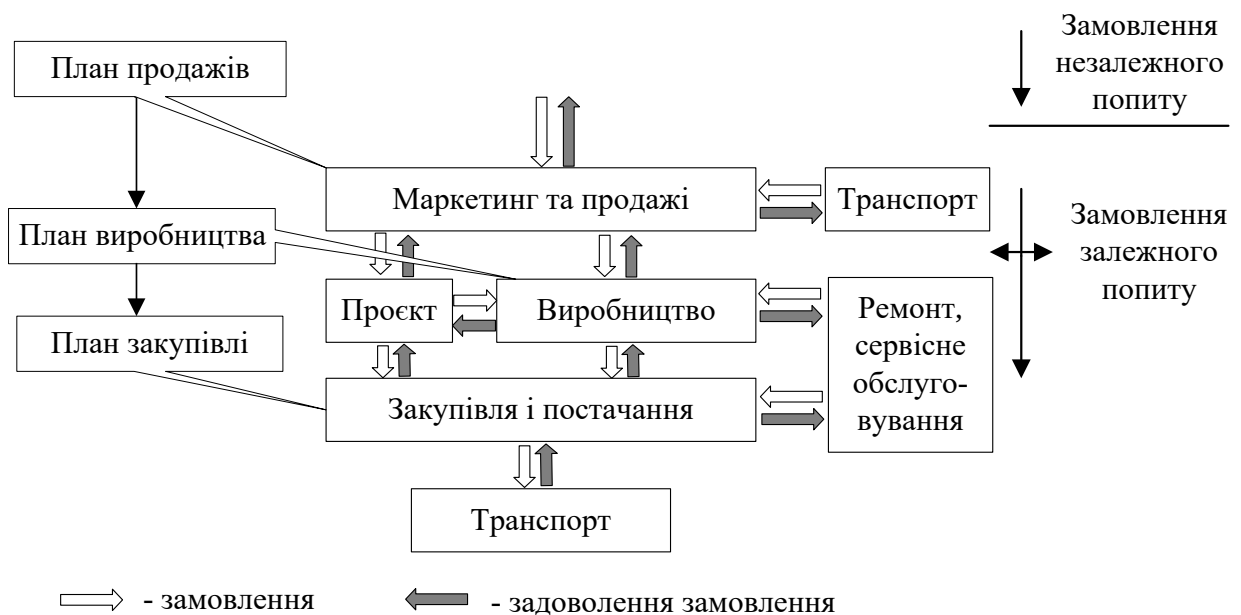


Рисунок 2.4 – Діаграма потоку замовлень

Діаграма потоку замовлень, наведена на попередньому рисунку, дозволяє автоматизувати процес забезпечення попиту на продукцію підприємства.

Замовлення, як правило, розглядаються з двох точок зору: замовлення незалежного попиту, замовлення залежного попиту.

Замовлення незалежного (ринкового) попиту надходять ззовні і, в свою чергу, мають такі основні категорії: прогнозні (передбачувані на основі аналізу потреб ринку), підтверджені (існуючий портфель замовлень). Замовлення незалежного попиту визначають показники програми збуту готової продукції або її компонент. Програма продажів є основою для формування плану виробництва і, як наслідок, плану закупівель, реалізація якого забезпечується замовленнями залежного попиту: замовлення на виробництво, замовлення на закупівлю, замовлення на внутрішнє переміщення (склад – склад, склад – виробництво / проект / профілактика або ремонт), замовлення на транспортування і інші.

В межах незалежного попиту на вироблену продукцію може бути кілька різновидів замовлень, що породжують особливості планування виробництва. Основні різновиди замовлень визначаються так званою точкою прив'язки замовлення клієнта (ТПЗК), що представляє собою найбільш далеку точку в виробничому процесі, в якій планування матеріалів, компонент, деталей і вузлів кінцевого продукту визначається замовленням клієнта. Іншими словами, це є глибина проникнення замовлення клієнта в систему планування.

У випадку виготовлення на замовлення (МТО – Make to Order), вироблені вироби виконуються на основі замовлення клієнта, плануються потреби в матеріалах і комплектуючих, а також конструктивні і технологічні особливості виробництва визначаються з специфікацією і технологією. Часто використовується термін ЕТО (Engineering To Order – Розробка на замовлення). План продажів формується на підставі портфеля замовлень. У випадку складання на замовлення, вироблені вироби виконуються на замовлення в основному зі стандартних компонент, відмінності виробів одного типу один від одного формує в результаті деякий «модельний ряд».

План продажів в основному формується на підставі портфеля замовлень і може доповнюватися прогнозними даними. У випадку виробництва на склад, є обмежена номенклатура продукції. Готова продукція виробляється на склад для подальшої доставки. План продажів в основному формується на базі прогнозу попиту плюс портфель замовлень.

Незважаючи на те, що концепція ERP розроблялася, в основному, для сфери матеріального виробництва, її методи застосовні до управління торговим підприємством.

Таким чином, запропоновано модель модуля ERP системи територіально розподіленого підприємства, яка складається з набору діаграм, що характеризують особливості бізнес-процесів підприємства, та дозволяють розробляти програмне забезпечення для їх автоматизації.

2.5 Висновки за розділом 2

Сформульовано та класифіковано основні критерії вибору СКБД для побудови корпоративної інформаційної системи. На підставі аналізу в якості основи для побудови модулів ERP системи обрано СКБД Progress. Як базис для створення модулів ERP системи було прийнято рішення обрати мову ABAP (Advanced Business Application Programming), яка представляє собою пропрієтарну внутрішню мову програмування високого рівня для розробки SAP систем.

Запропоновано модель модуля ERP системи територіально розподіленого підприємства, що надає структуру взаємодії з зовнішніми об'єктами, діаграму взаємодії апаратних засобів ERP системи, узагальнену модель системи виробництва на основі планування та діаграму потоку замовлень. Розроблена модель модуля ERP системи територіально розподіленого підприємства складається з набору діаграм, що характеризують особливості бізнес-процесів підприємства, та дозволяють розробляти програмне забезпечення для їх автоматизації.

3 ОСНОВНІ РІШЕННЯ ЩОДО РЕАЛІЗАЦІЇ КОМПОНЕНТІВ СИСТЕМИ

3.1 Функціональна структура ERP системи підприємства

Концептуальні положення, що стали основою для проектування програмної системи, визначалися завданнями, що стояли перед корпоративним територіально розподіленим підприємством. При проектуванні враховувалася обмеженість ресурсів, що виділяються на створення системи, навчання персоналу, підтримання функціонування та розвиток системи.

Будь-яке виробниче підприємство є посередником між виробником і споживачем. Тому, крім управління процесами всередині виробництва, необхідно управляти зв'язками з постачальниками і з покупцями [1]–[17]. Слід зазначити, що особлива увага повинна бути приділена зв'язкам зі споживачем. Тобто система повинна бути орієнтована на покупця. В системі застосовується натурально-вартісна система обліку, тобто облік товарів здійснюється як в кількісному, так і в вартісному вираженні. Система забезпечує ведення всієї діяльності підприємства. Закладена можливість налаштування системи при зміні бізнес-правил або податкового законодавства.

Система побудована у вигляді набору модулів, які пов'язані за допомогою ядра системи. Такий спосіб організації дозволяє конфігурувати систему для обліку потреб підприємства, а також додавати нові модулі для подальшого розширення функціональності системи.

Ядро системи є основним модулем системи та забезпечує виконання таких функціональних можливостей:

- обробка даних журналу грошових операцій;
- обробка даних журналу товарних операцій;
- обробка даних головної бухгалтерської книги;
- обробка даних книги товарів;

- налаштування бізнес-правил;
- забезпечення прав доступу;
- облік заробітної плати.

Процедура обробки даних головної бухгалтерської книги і журналу грошових операцій призначений для оброблення основної бухгалтерської інформації. Головна бухгалтерська книга являє собою план рахунків. Всі рахунки мають однакову структуру. По кожному рахунку відображається інформація про поточні залишки грошових коштів, оборотах. Кожен рахунок має рівень обліку - рівень, за яким проводиться облік даного рахунку в головній бухгалтерській книзі. План рахунків відображає поточний стан всіх фінансів підприємства. Журнал грошових операцій містить інформацію про всі рухи грошових коштів. У журналі фіксуються такі основні дані: дата і час здійснення операції; код і вид операції; номери кореспондуючих рахунків; сума операції; вид і ідентифікатор документа.

На основі журналу грошових операцій і поточного стану плану рахунків може бути отримана інформація про стан плану рахунків на будь-який момент часу.

Процедура обробки даних книги товарів і журналу товарних операцій призначена для роботи з класифікованим набором карток товарів. Картка товару містить наступну інформацію про товар: найменування; ознака ваговий / штучний; посилання на довідники виробників, країн, одиниць виміру та інші; список штрих-кодів; ціна реалізації; код товарної групи; поточні залишки; ставки ПДВ і податку з продажів.

Книга товарів відображає стан всіх товарів підприємства на поточний момент часу. Журнал товарних операцій містить інформацію про всі рухи товарів. У журналі фіксуються такі основні дані: дата і час здійснення операції; код і вид операції; посилання на картку товару; кількість операції; суми операції в облікових цінах і цінах реалізації; вид і ідентифікатор документа.

На підставі журналу товарних операцій та поточного стану книги товарів може бути отримана інформація про стан книги товарів на будь-який момент часу.

Система на рівні ядра підтримує умови безпеки. На рівні СКБД перевіряються права доступу на підключення до бази даних. СКБД підтримує також можливість обмеження прав доступу до окремих таблиць і стовпців. На рівні програми підтримуються умови безпеки звернення до тих або інших функцій системи. Крім цього, при проведенні деяких операцій ведеться реєстрація інформації про відповідального користувача.

У ядро системи закладена можливість повторного налаштування при зміні способів ведення бізнесу або поточного податкового законодавства. Таким чином, наприклад, перехід на новий план рахунків може бути проведений в найкоротші терміни і без використання послуг програміста.

Програма імпортує дані з файлу БД у проміжну таблицю, звідки дані у відповідності з ознакою заносить у AVL таблицю, звідки після аналізу та редагування полів «Сума», «Замовлення» і «Рахунок» відбувається пакетне введення в транзакції FB01. Після перенесення даних виводиться звіт. Програма використовує три таблиці: ZSIM_SETTINGS; ZSIM_T; ZSIM_TSUMM. У першій таблиці (ZSIM_SETTINGS) зберігаються правила підміни значень при перенесенні їх у проміжну таблицю і правила винятків проводок, які не переносяться в проміжну таблицю. Таблиця ZSIM_T – проміжна таблиця, в якій зберігаються дані вивантажені з БД. Таблиця ZSIM_TSUMM – таблиця підсумкових значень, в яку після закінчення обробки окремих позицій переносяться підсумкові значення.

3.2 Структура програмного забезпечення

Слід зазначити, що логічну структуру програми визначає її функціональне призначення і можливості середовища розробки. На етапі проектування логічної структури було прийнято рішення про об'єднання

окремих функціональних компонентів системи в незалежні логічні модулі. Це позитивно позначилося на можливості масштабування системи і на можливості повторного використання програмного коду сторонніми розробниками для написання додаткових модулів. Крім того, модульна структура програми значно полегшує її сервісну підтримку.

На рис. 3.1 наведено структурну схему програми.

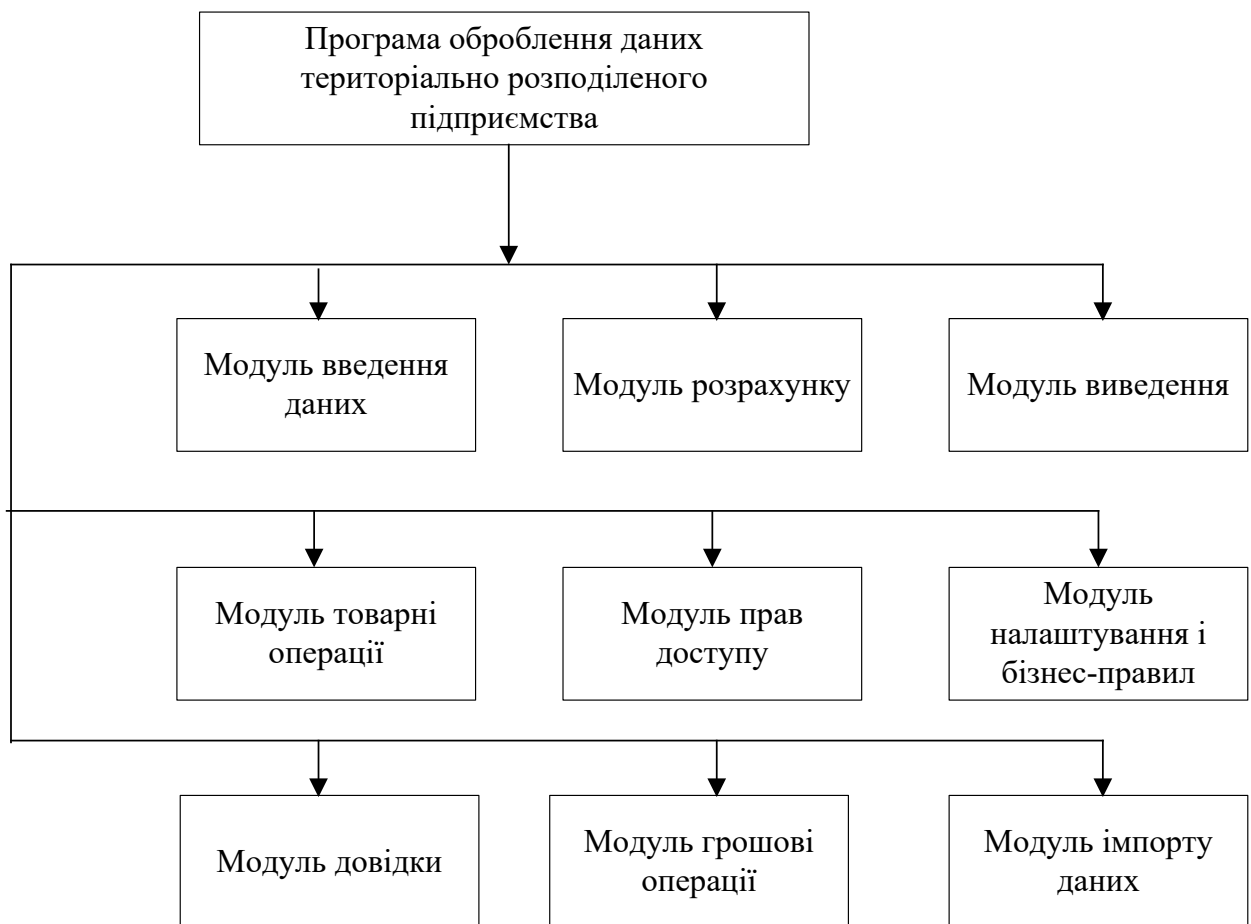


Рисунок 3.1 – Структурна схеми програми

Нижче представлено опис кожного з модулів програми.

Модуль введення даних здійснює завантаження вихідних даних:

- вибір файлу даних;
- введення даних за допомогою призначеного для користувача

інтерфейсу;

- попередній аналіз допустимості значень введених даних і, в разі виникнення помилок, виведення повідомлень про це з поясненнями і коментарями.

У розрахунковому модулі реалізовано обчислювальні алгоритми для вирішення найбільш поширених моделей розрахунку зарплати персоналу, про всі рухи грошових коштів, про поточні залишки, ставки ПДВ і податку з продажів. Розрахунковий модуль надає документований, незалежний від інших модулів інтерфейс, що дозволяє використовувати його в інших розробках і робить систему масштабованою і компонентно-незалежною.

Довідковий модуль реалізує такий набір функцій:

- надає оперативну інформацію про всі елементи призначеного для користувача інтерфейсу (контекстна допомога);
- робить повідомлення про помилки більш дружніми (за рахунок додавання можливості перегляду розширеного коментаря про помилку);
- надає детальну довідкову інформацію про всі розглянутих класах задач проведення фінансової документації.

Модуль виведення зберігає отримані в ході обчислень результати в файл звіту з можливістю подальшої роздруківки або виводить на екран в ергономічній і зручній для подальшого використання формі.

Модуль імпорту даних призначений для імпорту та передавання даних між іншими модулями.

Модуль грошові операції містить інформацію про всі рухи грошових коштів. У журналі фіксуються такі основні дані:

- дата і час здійснення операції;
- код і вид операції;
- номери кореспондуючих рахунків;
- сума операції;
- вид і ідентифікатор документа.

На основі модуля грошових операцій та поточного стану плану рахунків може бути отримана інформація про стан плану рахунків на будь-

який момент часу.

Модуль товарні операції є класифікованим набором карток товарів.

Картка товару містить наступну інформацію про товар:

- найменування;
- ознака ваговий / штучний;
- посилання на довідники виробників, країн, одиниць виміру та інші;
- список штрих-кодів;
- ціна реалізації;
- код товарної групи;
- поточні залишки;
- ставки ПДВ та податку з продажів.

Модуль товарних операцій відображає стан всіх товарів підприємства на поточний момент часу. Модуль товарних операцій містить інформацію про всі рухи товарів. У журналі товарних операцій фіксуються такі основні дані:

- дата і час здійснення операції;
- код і вид операції;
- посилання на картку товару;
- кількість операції;
- суми операції в облікових цінах і цінах реалізації;
- вид і ідентифікатор документа.

На основі журналу товарних операцій та поточного стану книги товарів може бути отримана інформація про стан книги товарів на будь-який момент часу.

Модуль прав доступу. Програма на рівні ядра підтримує умови безпеки. На рівні СКБД перевіряються права доступу на підключення до бази даних. СКБД підтримує також можливість обмеження прав доступу до окремих таблиць і стовпців. На рівні програми підтримуються умови безпеки звернення до тих або інших функцій системи. Крім цього, при

проведенні деяких операцій ведеться реєстрація інформації про відповідального користувача.

Модуль налаштування бізнес-процесів. У ядро системи закладена можливість повторного налаштування при зміні способів ведення бізнесу або поточного податкового законодавства. Таким чином, наприклад, перехід на новий план рахунків може бути проведений в найкоротші терміни і без використання послуг сторонніх організацій.

Програмний модуль реалізований таким чином, що він динамічно підключає всі необхідні стандартні бібліотеки операційної системи. Це дозволяє істотно зменшити розмір програмного модуля і зробити його переносно незалежним в межах сімейства операційних систем.

Схему взаємодії модулів програми наведено на рис. 3.2. Як видно, взаємодія між модулями відбувається за допомогою модулю імпорту даних.

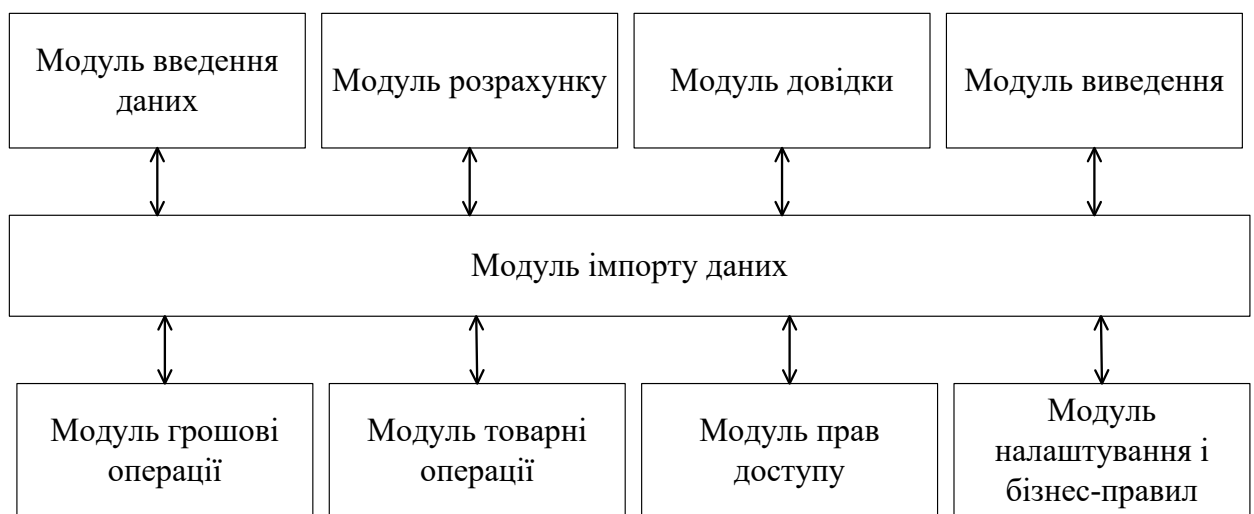


Рисунок 3.2 – Схема взаємодії модулів програми

Аналізуючи вибір засобів проектування, можна зробити висновок, що розроблювана програма є досить швидкою та не вимогливою до ресурсів, простою в зверненні, а обрана структура дозволила побудувати програму, що раціонально використовує машинні ресурси і час, і забезпечує надійну роботу.

3.3 Висновки за розділом 3

Показано, що система класу ERP дозволяє ефективно керувати ресурсами підприємства.

Спроековано структуру програмного забезпечення. Основними модулями програми є: модуль введення даних, розрахунковий модуль, довідковий модуль, модуль виведення, модуль грошових операцій, модуль товарних операцій, модуль прав доступу, модуль налаштування бізнес-процесів.

Аналізуючи вибір засобів проектування, можна зробити висновок, що розроблювана програма є досить швидкою та не вимогливою до ресурсів, простою в зверненні, а обрана структура дозволила побудувати програму, що раціонально використовує машинні ресурси і час, і забезпечує надійну роботу.

4 КЕРІВНИЦТВО ПРОГРАМІСТА

4.1 Призначення й умови застосування програми

Програмний продукт призначений для оброблення даних територіально розподіленого підприємства.

Програма забезпечує автоматизацію бізнес-процесів підприємства та взаємодію підприємства з різними зовнішніми об'єктами, зокрема, постачальниками, банками, податкової інспекцією, оптовими та роздрібними покупцями.

Програмний продукт дозволяє виконувати такі операції:

- зчитування даних з файлів бази даних;
- розрахунок заробітної плати;
- керування фінансами;
- керування замовленнями і поставками;
- керування складом;
- керування торговим залом;
- керування продажами;
- керування виробництвом;
- керування експлуатацією майна підприємства;
- можливість настройки призначеного для користувача інтерфейсу.

Інтерфейс користувача забезпечує просту і ефективну взаємодію користувача з програмним продуктом, а також передбачає можливість його адаптації до вимог конкретного користувача.

Для експлуатації програмного комплексу необхідні такі програмно-технічні засоби:

- ПК під керівництвом операційної системи;
- тактова частота процесора 2 ГГц або більше, Video 512 Мб;
- обсяг оперативної пам'яті не менше 1 Гб;
- обсяг вільного простору на диску - не менше 100 Мб;

- дозвіл екрана монітора не менше 1024x768;
- маніпулятор миша.

4.2 Характеристики програми

Час роботи програми залежить від конфігурації комп'ютера, на якому вона виконується.

У процесі роботи програма виводить інформацію у вигляді:

- форми з вхідними та вихідними даними;
- форми з результатами розрахунку заробітної плати;
- форми з імпортування даних;
- інші форми.

У процесі роботи програма використовує довідкові дані з бази даних.

Програма написана в середовищі візуального програмування АВАР/4. Програмний модуль реалізований таким чином, що він здатний динамічно підключати всі необхідні стандартні бібліотеки операційної системи. Це дозволяє істотно зменшити розмір програмного модуля і зробити його переносним незалежним в межах сімейства операційних систем.

4.3 Звернення до програми

Розроблена програма поширюється у вигляді інсталяційного пакета. Для можливості роботи з програмою необхідно розмістити файли модулів програмного забезпечення на сервері. Перед використанням програми її необхідно встановити. Для цього необхідно запустити програму `setup.exe`, яка знаходиться на установчому диску. У відповідь на запит установчої програми потрібно вказати шлях встановлення програми. Після цього програма скопіює необхідні файли, створить і помістить в головне меню системи ярлик для запуску програми і зробить необхідні зміни в файлі реєстрації. Після завершення процедури установки програму можна використовувати.

Програму можна запустити будь-якими засобами операційної системи користувача.

4.4 Вхідні і вихідні дані

Вхідними даними для програми є інформація з документів різного призначення, зокрема платіжних документів, а також інформація про кількість продукції на складі, вартість продукції, постачальників, клієнтів і т.ін.

Вихідними даними для програми є розрахунки по бізнес-процесам організації, заробітній платі, вартості відпускної продукції, управління замовленнями і поставками, управління складом, які можуть бути збережені на жорсткому диску.

4.5 Повідомлення

Реалізована програмна система гарантує її стійке функціонування. У разі виникнення помилок в програмі передбачено відображення стандартних діалогових вікон. Подібні повідомлення можуть виникати при деяких небажаних ситуаціях, до яких можна віднести: введення користувачем некоректних даних; відсутність необхідних довідкових даних у базі даних.

Приклад повідомлення наведено на рис. 4.1.

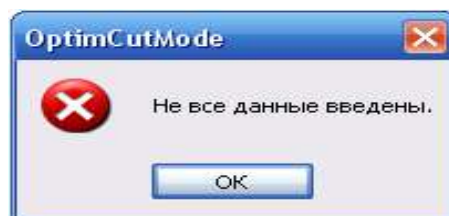


Рисунок 4.1 – Діалогове вікно з повідомленням про помилку

Можна зробити висновок, що система характеризується широкими функціональними можливостями. Програма забезпечує стійку роботу і обробку помилкових ситуацій.

5 КЕРІВНИЦТВО ОПЕРАТОРА

5.1 Призначення програми

Програмний продукт призначений для оброблення даних територіально розподіленого підприємства.

Програма забезпечує автоматизацію бізнес-процесів підприємства та взаємодію підприємства з різними зовнішніми об'єктами, зокрема, постачальниками, банками, податкової інспекцією, оптовими та роздрібними покупцями.

Програмний продукт дозволяє виконувати такі операції:

- зчитування даних з файлів бази даних;
- розрахунок заробітної плати;
- керування фінансами;
- керування замовленнями і поставками;
- керування складом;
- керування торговим залом;
- керування продажами;
- керування виробництвом;
- керування експлуатацією майна підприємства;
- можливість настройки призначеного для користувача інтерфейсу.

Інтерфейс користувача забезпечує просту і ефективну взаємодію користувача з програмним продуктом, а також передбачає можливість його адаптації до вимог конкретного користувача.

5.2 Умови виконання програми

Для експлуатації програмного комплексу необхідні такі програмно-технічні засоби:

- ПК під керівництвом операційної системи;
- тактова частота процесора 2 ГГц або більше, Video 512 Мб;

- обсяг оперативної пам'яті не менше 1 Гб;
- обсяг вільного простору на диску - не менше 100 Мб;
- дозвіл екрана монітора не менше 1024x768;
- маніпулятор миша.

Час роботи програми залежить від конфігурації комп'ютера, на якому вона виконується.

У процесі роботи програма виводить інформацію у вигляді:

- форми з вхідними та вихідними даними;
- форми з результатами розрахунку заробітної плати;
- форми з імпортування даних;
- інші форми.

5.3 Виконання програми

Розроблена програма поширюється у вигляді інсталяційного пакета. Для можливості роботи з програмою необхідно розмістити файли модулів програмного забезпечення на сервері. Перед використанням програми її необхідно встановити. Для цього необхідно запустити програму `setup.exe`, яка знаходиться на установчому диску. У відповідь на запит установчої програми потрібно вказати шлях встановлення програми. Після цього програма скопіює необхідні файли, створить і помістить в головне меню системи ярлик для запуску програми і зробить необхідні зміни в файлі реєстрації. Після завершення процедури установки програму можна використовувати. Програму можна запустити будь-якими засобами операційної системи користувача.

5.4 Приклад роботи з програмою

Після запуску програми відкривається головна форма, призначена для вибору бізнес-процесу, з яким необхідно працювати. Як правило, при роботі з програмою виникає необхідність оброблення даних з бази даних.

Наприклад, при необхідності оброблення даних для нарахування заробітної плати програма зчитує дані з файлу БД типу T_DSXXYY.BDF (де XX – місяць, YY - рік), заповнює необхідні поля і подає їх в ALV вигляді в проміжній таблиці (структурі), де можна змінити всі значення полів (табл.5.1).

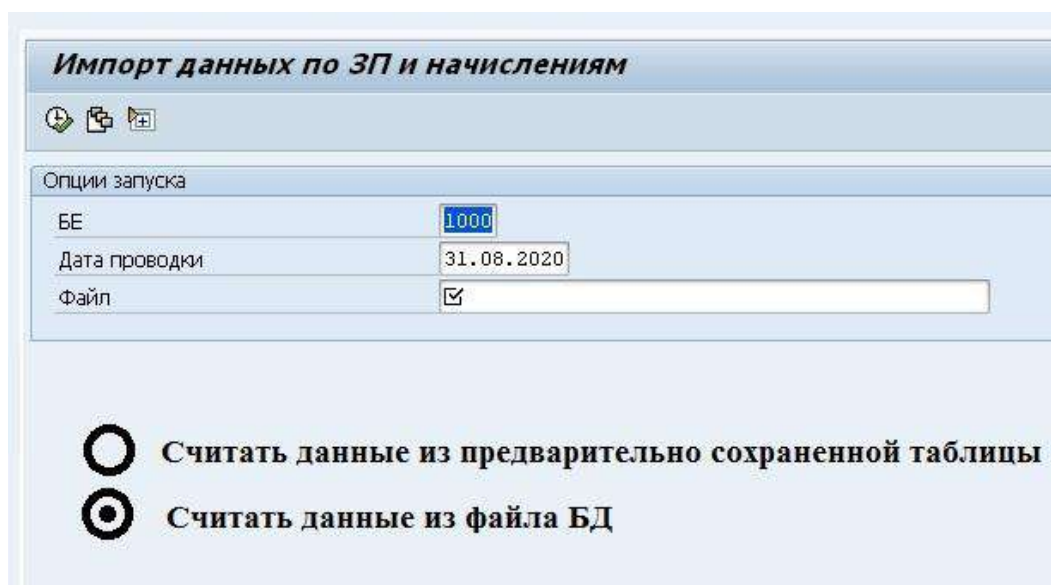
Таблиця 5.1 – Проміжна таблиця

DSS	DAS	KSS	SUMMA	TIPO	KODP	VO
92	302000	811	119328,35	1	ZAA0001	108

При натисканні кнопки Перевірка на інтерпретацію, запускається в тестовому режимі процес інтерпретації (за допомогою таблиці ZSIM_SETTINGS). Система намагається застосувати алгоритм заміни і виключення позицій. Якщо якісь позиції не проходять, алгоритм виводить ALV звіт з тими позиціями, які не змогли бути інтерпретованими або виключеними.

Оператор вносить необхідні зміни в цю таблицю (і видаляє позиції теж), натискаючи кнопку Перевірка на інтерпретацію повторно, система знову застосовує до цих позицій алгоритм інтерпретування. Якщо позиції проходять інтерпретацію, вони виключаються з цього звіту, а в першу проміжну (структурну) таблицю вносяться зміни. Коли всі позиції пройшли інтерпретацію без помилок, виводиться звіт про те, що всі позиції готові до інтерпретації.

Після цього оператор натискає кнопку Імпортувати дані, внаслідок чого з'являється форма Імпорту даних (див. рис. 5.1), програма виконує алгоритм інтерпретації і записує дані в таблицю ZSIM_T, паралельно або після закінчення внесення всіх рядків в таблицю, включається алгоритм агрегування всіх позицій і внесення їх в таблицю ZSIM_TSUMM (за вказаним вище алгоритмом). Кожен раз при внесенні змін до таблиці ZSIM_T має відбуватися оновлення таблиці ZSIM_TSUMM.



Импорт данных по ЗП и начислениям

Опции запуска

БЕ	1000
Дата проводки	31.08.2020
Файл	☒

☐ Считать данные из предварительно сохраненной таблицы
☒ Считать данные из файла БД

Рисунок 5.1 – Імпорт даних по ЗП і нарахуваннях

На екрані є можливість або зчитати дані з файлу БД (після чого відредагувати дані в ALV таблиці, зберегти їх і потім або перейти до проведення документа або вийти), або зчитати дані з уже інтерпретованої таблиці (за датою інтерпретації).

Якщо оператор намагається інтерпретувати цією ж датою ще один файл БД, виводиться повідомлення: Увага! Цією датою вже була завантажена база даних, скасувати операцію або перезаписати результат інтерпретації попереднього разу? Можна вибрати або перезаписати всю таблицю (спочатку всі позиції з таблиці ZSIM_T з цією датою видаляються, потім можна працювати з новою) або зчитати вже збережені позиції. Проводка реальних документів. Після того як всі дані були інтерпретовані, і збережені необхідно створити документи. Є можливість запустити програму в двох режимах.

Першим є тестовий режим, в якому програма пакетного введення починає зчитувати дані з проміжної таблиці ZSIM_TSUMM, програма при цьому документи не проводить, а тільки моделює, всі повідомлення типу warning і error записує в звіт з прив'язкою до документа який система намагалася провести. Після цього оператор виправляє всі помилкові позиції.

При використанні продуктивного (звичайного) режиму програма пакетного введення починає зчитувати дані з проміжної таблиці ZSIM_TSUMM через підрядник, проводить всі документи, які не містять помилки, помилкові документи пропускає і за підсумком виводить звіт про кількість успішно проведених документів і виводить ті позиції які містили помилки, з текстом помилки. Після цього оператор може виправити позиції і провести помилкові позиції повторно. У таблиці ZSIM_T ZSIM_T з'явиться індикатор, проведено/немає.

Приклади звітів з інформацією про помилкові дані наведено на рис.5.2.

Recording ZAR_AA_CO		account-account+CO
F-02 Ввод записи на ОСч		
☐	SAPMF05A	0100
☐	SAPMF05A	0300
☐	SAPLKACB	0002
☐	SAPMF05A	0300
☒	SAPLKACB	0002

Recording ZAR_AK		account-kreditor
F-02 Ввод записи на ОСч		
☐	SAPMF05A	0100
☐	SAPMF05A	0300
☐	SAPLKACB	0002
☒	SAPMF05A	0302

Рисунок 5.2 – Приклади звітів з інформацією про помилкові дані в рядках даних SAPLKACB та SAPMF05A

Після закінчення процесу на екран виводиться відомість за результатами з можливістю друку.

При натисканні кнопки «Перевірка на інтерпретацію» система інтерпретує кожен рядок відповідно до табл. 5.1, якщо при інтерпретації

помилки не виникло, повинен вивестися звіт, що можна інтерпретувати всі позиції, в разі помилки – система запропонує вручну внести зміни, для того щоб не було помилок.

Після перевірки, при натисканні кнопки «Інтерпретувати дані» система інтерпретує дані і зберігає їх в таблиці ZSIM_T. Під час інтерпретації при необхідності відбувається заміна значень, видалення значень і перевірка значень. Всім позиціям присвоюється одне з значень поля VAR (варіант), в подальшому дане поле вплине на вибір екрану заповнення.

Приклад звіту з нарахування заробітної плати наведено на рис.5.3.

О	БЕ	ДатаДокум	Текст заголовка документа	КП Дт	СчетДебет	Заказ	КП Кт	СчетКредит	Сумма
✓	1000	31.10.2020	SALARY10/11	40	82100100		50	10	4.817,79
✓	1000	31.10.2020	SALARY10/11	40	82100100		50	10	158,70
✓	1000	31.10.2020	SALARY10/11	40	82100100		50	10	7.507,67
✓	1000	31.10.2020	SALARY10/11	40	82100100		50	10	6.728,86
✓	1000	31.10.2020	SALARY10/11	40	82100100		50	10	624,60
✓	1000	31.10.2020	SALARY10/11	40	82100100		50	10	2.421,85
✓	1000	31.10.2020	SALARY10/11	40	82100100		50	10	8.663,67

Рисунок 5.3 – Приклад звіту з нарахування заробітної плати

5.5 Повідомлення оператора

Як відзначено вище, у випадку виникнення помилок в програмі передбачено відображення стандартних діалогових вікон. Подібні повідомлення можуть виникати при деяких небажаних ситуаціях, до яких можна віднести: введення користувачем некоректних даних; відсутність необхідних довідкових даних у базі даних.

Приклади повідомлень, що видаються оператору, наведено у табл. 5.2.

Таблиця 5.2 – Повідомлення оператору

Повідомлення	Дія оператора
Внимание! Этим числом уже была загружена база данных, отменить операцию или перезаписать результат интерпретации предыдущего раза?	Можна вибрати або перезаписати всю таблицю (спочатку всі позиції з таблиці ZSIM_T з цією датою видаляються, потім можна працювати з новою) або використовувати вже збережені позиції.
Внимание! Данные позиции уже проведены, исключить проведенные позиции из проводки?	Якщо оператор вибирає Так, то система пропонує проводити тільки ті позиції, які ще не проведені. Якщо Ні, то система проводить заново всі позиції.
База данных пуста	У базі даних немає інформації.
Невозможно создать файл	Можливо немає прав на запис або не вистачає вільного місця на носії.
Невозможно открыть файл	Можливо немає прав на читання або не існує запитованого файлу на носії
Ошибка при доступе к файлу	Переконайтеся, що файли, призначені для запису програми, не відкриті в операційній системі, переконайтеся що директорія, в яку проводиться запис файлу, не є захищеною від запису.
Закончить работу с программой?	Натиснути Так, якщо необхідно закінчити роботу з програмою, в іншому випадку натиснути Ні або закрити діалогове вікно

6 ЕКОНОМІКО–ОРГАНІЗАЦІЙНА ЧАСТИНА

У дипломній кваліфікаційній роботі магістра досліджено та програмно реалізовано модулі ERP-системи.

Очікується, що впровадження в експлуатацію розроблених програмних модулів на підприємстві дозволить скоротити 0,7 ставки діловода, що відповідає за обробку документів, з заробітною платою 9500 гривень на місяць.

У розробці беруть участь програміст із заробітною платою 14000 грн. в місяць і консультант з заробітною платою 11000 грн. у місяць. На розробку проєкту було витрачено 2 місяці. Розробка програми розпочалася першого вересня і повинна бути виконана до кінця жовтня 2020 року.

6.1 Планування розробки програмного продукту

Весь комплекс розробки програмного продукту можна розділити на етапи [18]. Для кожного етапу вказуються трудомісткість, кількість виконавців і тривалість робіт. Інформація з характеристики робіт по розробці програмного забезпечення зведена у таблицю 6.1.

Таблиця 6.1 – Характеристика робіт по розробці програми

Найменування робіт	Трудомісткість		Виконавці	Тривалість, днів
	люд-дн	%		
Аналіз ПЗ	6	8,9	Програміст	3
			Консультант	3
Визначення вимог до ПП	12	17,9	Програміст	6
			Консультант	6
Проектування структури програми	26	38,8	Програміст	13
			Консультант	13
Розробка схеми функціонування програми	3	4,5	Програміст	3
Створення програмного коду	9	13,4	Програміст	9
Тестування і налагодження програмного продукту	7	10,4	Програміст	7
Складання програмної документації	4	5,97	Програміст	4
Разом	67	100	Програміст	45
			Консультант	22

6.2 Визначення витрат на розробку програми

Для визначення витрат на розробку програми складають калькуляцію кошторисної вартості робіт, що включає статті:

– основна заробітна плата;

- додаткова заробітна плата;
- відрахування на єдиний соціальний внесок (ЄСВ);
- матеріали та комплектуючі вироби;
- витрати на спеціальне обладнання (на оплату машинного часу ЕОМ для написання і налагодження програмних засобів);
- накладні витрати.

Витрати за статтею «Основна заробітна плата» складаються з планового фонду зарплати всіх категорій працівників, зайнятих в розробці програми. Розрахунок зарплати на підставі даних про трудомісткість, представлений в таблиці 6.2.

Таблиця 6.2 – Основна заробітна плата

Посада виконавця	Чисельність, чол	місячний оклад	Кількість місяців роботи	Сума ЗП, грн
Програміст	1	14000	2	28000
Консультант	1	11000	1	11000
Разом	2			39000

Додаткову заробітну плату приймають рівною 10% від основної зарплати. Вона розраховується за формулою (6.1):

$$ЗП_{\text{дод}} = ЗП_{\text{осн}} \cdot 10\%, \quad (6.1)$$

де $ЗП_{\text{дод}}$ – додаткова заробітна плата;

$ЗП_{\text{осн}}$ – основна заробітна плата.

Підставивши величину основної заробітної плати в формулу (6.1) отримаємо: $ЗП_{\text{дод}} = 39000 \cdot 0,1 = 3900$ грн.

Відрахування на єдиний соціальний внесок (ЄСВ) визначають у відсотковому відношенні від сум основної та додаткової зарплати з урахуванням премій і доплат. Відрахування вираховуються за формулою (6.2):

$$\text{ВІД} = (\text{ЗП}_{\text{осн}} + \text{ЗП}_{\text{дод}}) \cdot 0,22, \quad (6.2)$$

де ВІД – відрахування на єдиний соціальний внесок;

$\text{ЗП}_{\text{осн}}$ – додаткова заробітна плата;

$\text{ЗП}_{\text{дод}}$ – основна заробітна плата.

Підставивши величини основної та додаткової заробітної плати в формулу (6.2) отримаємо:

$$\text{ВІД} = (39000 + 3900) \cdot 0,22 = 9438 \text{ грн.}$$

Використовується два найменування матеріалів: картридж – 900 грн, папір – 300 грн. Витрати на матеріали розраховують за формулою (6.3):

$$M = \sum_{i=1}^n (\text{Ц}_i \cdot N_i \cdot (1 + K_{\text{тз}}) - \text{Ц}_{i\text{в}} \cdot N_{i\text{в}}), \quad (6.3)$$

де M – витрати на матеріали, покупні напівфабрикати і комплектуючі вироби, грн;

Ц_i – ціна i -го найменування матеріалу, напівфабрикату і комплектуючого, грн.;

N_i – потреба в i -му матеріалі, напівфабрикаті і комплектуючому;

$K_{\text{тз}}$ – коефіцієнт, що враховує транспортно-заготівельні витрати;

$\text{Ц}_{i\text{в}}$ – ціна зворотних відходів i -го найменування матеріалу, грн;

$N_{i\text{в}}$ – кількість зворотних відходів i -го найменування;

n – кількість найменувань матеріалів, напівфабрикатів і комплектуючих.

Підставивши ціни матеріалів та величини $\Pi_{iB} = 0$; $N_{iB} = 0$; $K_{T3} = 0,05$ у формулу (6.3), отримаємо:

$$M = (1 + 0,05) \cdot (900 + 300) = 1260 \text{ грн.}$$

Таким чином, разом витрати на матеріали та комплектуючі вироби становлять 1260 гривень.

У статтю «Витрати на спеціальне обладнання» входять витрати на придбання, транспортування, монтаж і налагодження нестандартного обладнання. Практично, в даному випадку, в цій статті враховуються витрати на оплату машинного часу ЕОМ для написання і налагодження програмних засобів. Для цього необхідно скласти кошторис «витрат на утримання і експлуатацію устаткування», виходячи з якого визначиться вартість одного машино-години роботи ПК, після множення якої на машинний час пішло на написання і налагодження програми отримаємо витрати на оплату машинного часу.

Амортизаційні відрахування визначають за формулою (6.4):

$$A = БВ \cdot \frac{H_a}{100}, \quad (6.4)$$

де A – амортизаційні відрахування;

$БВ$ – балансова вартість обчислювальної техніки, грн;

H_a – норма амортизаційних відрахувань на повне оновлення обчислювальної техніки, %. Приймаємо, що техніка повинна працювати 4 роки, тому $H_a = 25\%$.

Балансова вартість обчислювальної техніки становить 28000 грн. Тому амортизаційні відрахування за формулою (6.4) становлять:

$$A = 28000 \cdot 0,25 = 7000 \text{ грн.}$$

Статтю "Експлуатація обладнання" розраховують підсумовуванням витрат на силову електроенергію і допоміжні матеріали. Витрати на електроенергію визначають за формулою (6.5):

$$C_e = N_n \cdot \Phi_{\text{еф}} \cdot K_{\text{зв}} \cdot K_{\text{зп}} \cdot C_e, \quad (6.5)$$

де C_e – витрати на силову електроенергію;

N_n – номінальна потужність ЕОМ, кВт;

$\Phi_{\text{еф}}$ – річний ефективний фонд часу роботи ЕОМ, машино-год;

$K_{\text{зв}}$ – середній коефіцієнт завантаження за часом;

$K_{\text{зп}}$ – коефіцієнт завантаження по потужності;

C_e – ціна одного кВт·год електроенергії, грн / (кВт·год).

Номінальна Потужність ЕОМ – 0,2 кВт. Річний ефективний фонд часу роботи ЕОМ ставити 1800 годин. Середні коефіцієнти завантаження за часом та за потужністю рівні відповідно 0,9 та 0,6. Ціна однієї кіловат–години електроенергії ставить 2,11 грн. З формули (6.5) отримуємо:

$$C_e = 0,2 \cdot 1800 \cdot 0,9 \cdot 0,6 \cdot 2,11 = 410,18 \text{ грн.}$$

Приймаємо, що $C_e = 410$ грн.

Витрати за статтею "Заробітна плата обслуговуючих робітників та відрахування на соціальне страхування в інші фонди" визначають за формулою (6.6):

$$ЗП_{\text{обсл}} = \Phi ЗП_{\text{р}} \cdot (1 + K_{\text{відрах}}) \cdot \frac{t_{\text{обсл}}}{\Phi_{\text{еф.обсл}}}, \quad (6.6)$$

де $\Phi ЗП_p$ – річний фонд заробітної плати (основної та додаткової) обслуговуючих робітників, грн.;

$K_{\text{відрах}}$ – коефіцієнт, що враховує відрахування на соціальне страхування і в інші фонди;

$t_{\text{обсл}}$ – час протягом року, необхідне технічне обслуговування ЕОМ, годин / рік. Приймаємо $t_{\text{обсл}} = 24$ год;

$\Phi_{\text{еф.обсл}}$ – річний ефективний фонд часу обслуговуючого персоналу, годин / рік. Приймаємо, що річний ефективний фонд часу обслуговуючого персоналу становить 1800 годин.

Заробітна плата обслуговуючого персоналу = 9000 грн/місяць, тому

$$\Phi ЗП_p = 9000 \cdot 12 = 108000 \text{ грн.}$$

Підставивши всі величини у формулу (6.6), отримаємо:

$$ЗП_{\text{обсл}} = 108000 \cdot (1 + 0,22) \cdot \frac{24}{1800} = 1757 \text{ грн.}$$

Суму витрат за статтею "Поточний ремонт обладнання" обчислимо як 3% від балансової вартості обладнання, оскільки неможливо заздалегідь визначити, скільки коштів необхідно витратити на придбання запасних частин наявного обладнання. Балансова вартість обчислювальної техніки становить 28000 грн. Тому сума витрат за статтею "Поточний ремонт обладнання" становить: $28000 \cdot 0,03 = 840$ грн.

Інших витрат за проектом не передбачено, тому сума витрат за цією статтею дорівнює нуль гривень.

Кошторис витрат на утримання і експлуатацію устаткування наведений в таблиці 6.3.

Таблиця 6.3 – Кошторис витрат на утримання і експлуатацію устаткування

Найменування статей витрат	Сума, грн.
Амортизація обладнання	7000
Заробітна плата основна і додаткова обслуговуючих робітників з відрахуваннями на соціальні заходи	1757
Експлуатація обладнання (крім витрат на поточний ремонт)	410
Поточний ремонт обладнання	840
Інші витрати	0
Разом	10007

Витрати на оплату машинного часу ЕОМ для написання і налагодження програмних засобів визначаються за формулою (6.7):

$$C_{\text{мо}} = B_{\text{екс}} \cdot t_{\text{мо}}, \quad (6.7)$$

де $C_{\text{мо}}$ – витрати на оплату машинного часу, грн;

$B_{\text{екс}}$ – експлуатаційні витрати на одну годину машинного часу цієї цифрової ЕОМ, грн. / машино–год.;

$t_{\text{мо}}$ – машинний час цифрової ЕОМ для написання і налагодження даного програмного продукту, машино–год.

$$B_{\text{екс}} = 10007 / 1800 = 5,56 \text{ грн/машино-год.}$$

Для написання і налагодження даного програмного продукту ЕОМ експлуатується протягом 45 днів в одну зміну по 8 годин, що складає в сумі

360 годин. Таким чином, за формулою (6.7), отримуємо суму за статтею «Витрати на спеціальне обладнання»:

$$C_{\text{мо}} = 5,56 \cdot 360 = 2001 \text{ грн.}$$

До накладних витрат відносяться витрати на загальне управління і загальногосподарські потреби (заробітна плата апарату управління, канцелярські витрати і т.д.), утримання та експлуатацію будівель. Накладні витрати включаються до вартості розробки програми непрямым шляхом – у відсотках до основної заробітної плати розробників. Для обчислення цієї статті видатків у дипломному проєкті приймемо, що накладні видатки становлять 30% від основної заробітної плати розробників, що складає 39000 грн. Тому сума накладних видатків становить: $39000 \cdot 0,25 = 11700$ грн.

Калькуляція кошторисної вартості робіт з розробки програми наведена в таблиці 6.4.

Таблиця 6.4 – Калькуляція кошторисної вартості робіт з розробки програми

Найменування статей витрат	Сума, грн.
Основна заробітна плата	39000
Додаткова заробітна плата	3900
ЄСВ	9438
Матеріали і комплектуючі	1260
Витрати на спеціальне обладнання	2001
Накладні витрати	11700
Разом	67299

6.3 Розрахунок економічної ефективності програмного продукту

Річну економію від впровадження програми, що дозволяє знизити витрати машинного часу ЕОМ для рішення певного завдання в порівнянні із програмою, що застосовувалася раніше, визначають за формулою (6.8):

$$E = (t_{M_1} - t_{M_2}) \cdot B_{\text{екс.КТЗ}} + E_n^a \cdot \frac{\Phi_{\text{б.КТЗ}} \cdot (t_{M_1} - t_{M_2})}{\Phi_{\text{еф.КТЗ}}} - \frac{S_{\text{рп}}}{T_c} + E_{\text{зп}}, \quad (6.8)$$

де t_{M_1} , t_{M_2} – машинний час, необхідний для розв’язання поставлених завдань, відповідно в старому та у новому варіантах, машино–год./рік;

$B_{\text{екс.КТЗ}}$ – експлуатаційні витрати, що доводяться на 1 год машинного часу ЕОМ, грн/ машино–год.;

E_n^a – нормативний коефіцієнт ефективності капітальних вкладень у засоби автоматизації;

$\Phi_{\text{б.КТЗ}}$ – балансова вартість ЕОМ (КТЗ), грн.;

$\Phi_{\text{еф.КТЗ}}$ – річний ефективний фонд часу роботи ЕОМ, машино–год.;

$S_{\text{рп}}$ – сумарні витрати на розробку програми, грн.;

T_c – термін служби впроваджуваної програми до її морального зношування, років. Приймаємо цей термін 5 років;

$E_{\text{зп}}$ – економія на заробітній платі.

До впровадження даного ПЗ дані функції виконувались співробітником, що відповідає за обробку рукописних символів, 1800 годин на рік, а після 540 годин на рік (економія складає 0,7 ставки діловода, що відповідає за обробку документів, з заробітною платою 9500 гривень на місяць).

Підставивши всі величини у формулу (6.8), отримаємо:

$$E = (1800 - 540) \cdot 5,56 + 0,2 \cdot (28000 \cdot (1800 - 540)) / 1800 - 67299 / 5 + \\ + 9500 \cdot 0,7 \cdot (1 + 0,22) \cdot 12 = 96233 \text{ грн}$$

Коефіцієнт ефективності капітальних вкладень $E\phi$ и строк їхньої окупності $T_{ок}$ розраховуються за формулами (6.9) і (6.10):

$$E\phi = \frac{E}{S_{pp}}, \quad (6.9)$$

$$T_{ок} = \frac{S_{pp}}{E}. \quad (6.10)$$

Підставивши всі величини у формули (6.9) і (6.10), отримаємо:

$$E\phi = 96233 / 67299 + 1,43,$$

$$T_{ок} = 67299 / 96233 = 0,7 \text{ року.}$$

Зведемо отримані техніко–економічні показники в таблицю 6.5.

Таблиця 6.5 – Техніко–економічні показники

Показники	Сума	Одиниця виміру
Витрати на розробку програми	67299	грн.
Річна економія	96233	грн.
Строк окупності програми	0,7	рік

Таким чином, впровадження в експлуатацію розробленого програмного забезпечення на виробничому підприємстві дозволить скоротити 0,7 ставки діловода, що відповідає за обробку документів, з

заробітною платою 9500 гривень на місяць. Економічні розрахунки показали, що строк окупності програми 0,7 року, що значно менше часу її нормативного строку окупності (5 років). Крім того, річна економія при використанні програми становить 96233 грн. Виходячи з виконаних розрахунків, можна зробити висновок, що розробка даного програмного продукту є економічно доцільною.

7 ОХОРОНА ПРАЦІ ТА БЕЗПЕКА У НАДЗВИЧАЙНИХ СИТУАЦІЯХ

7.1 Аналіз потенційних небезпек

Сучасний розвиток науки та техніки привносить принципові нововведення у всі сфери матеріального виробництва, докорінно змінюючи знаряддя та предмети праці, технологію, методи обробки інформації. Разом з тим, захопившись вдосконаленням засобів праці їх творці залишили поза увагою проблеми людини в рамках своєї комп'ютерної революції.

Методологічною основою охорони праці є науковий аналіз умов праці, технологічних процесів, виробничого обладнання, робочих місць, трудових операцій, організації виробництва з метою виявлення шкідливих і небезпечних виробничих факторів, їх властивостей, особливостей впливу на організм людини.

Дослідження та програмна реалізація модулів ERP системи передбачає роботу в обчислювальному центрі невеликої команди програмістів. Розміри приміщення 9х4х3,5м.

Площа приміщення становить $9 \times 4 = 36 \text{ м}^2$, об'єм – $9 \times 4 \times 3,5 \text{ м} = 126 \text{ м}^3$.

Небезпечні та шкідливі виробничі фактори за природою виникнення поділяються на такі групи: фізичні (підвищена і знижена температура повітря, надмірна запиленість і загазованість повітря та ін.); хімічні (вміст у робочій зоні шкідливих речовин та ін.); психофізіологічні (нервово-емоційні перевантаження, розумове напруження та ін.); біологічні (підвищений вміст у повітрі мікроорганізмів та ін.).

У даному розділі розглянуто наступні питання: мікроклімат та склад повітря робочої зони; виробниче освітлення; шум; виробничі випромінювання; безпека щодо організації робочих місць; безпечність технологічного обладнання та процесу; електробезпека; пожежна безпека.

7.2 Заходи щодо забезпечення безпеки

При роботі в обчислювальному центрі головним небезпечним фактором є електрика.

Згідно ПУЕ за ступенем небезпеки ураження електричним струмом, приміщення обчислювального центру відноситься до приміщень без підвищеної небезпеки.

Питання правильного електроживлення при об'єднанні комп'ютерів в локальні мережі мають важливе значення як для забезпечення електробезпеки користувачів, так і для безаварійної і безперебійної роботи обладнання [20].

Заходи щодо забезпечення електробезпеки розроблено відповідно до ДСТУ Б В.2.5-82:2016 Електробезпека в будівлях і спорудах. Вимоги до захисних заходів від ураження електричним струмом [21].

Приміщення живиться електричною енергією від централізованого трансформатора, який знаходиться на поверсі поза межами приміщення і понижує напругу до 220 В, змінним струмом частотою 50 Гц. Комп'ютери мають власні блоки живлення, що перетворюють змінний струм мережі у постійний (5 В, 12 В). Потужність даних блоків живлення: 400 – 450 Вт. ПК у приміщенні відносяться до категорії I захисту від ураження електричним струмом, оскільки усі прилади при підключенні до електричної мережі яка заземляються, використовуючи систему TN. Тобто система, у якій живильні мережі має глухе заземлення однієї точки струмопровідних частин джерела живлення, а електроприймачі і відкриті провідні частини електроустановки приєднуються до цієї точки за допомогою відповідно нейтрального і захисного провідників. Корпуса всіх електричних пристроїв виготовляються з неструмопровідних матеріалів, а живлення здійснюється спеціальним кабелем, так щоб виключити ураження людини електричним струмом. Зазначені міри електробезпеки відповідають вимогам, пропонованим до побутових приладів відповідно до міжнародного стандарту CE. У

приміщенні електропроводка трифазна, чотирьохжильна, з подвійною ізоляцією. Всі кабелі сховані в коробах. Штепсельні розетки встановлені на висоті одного та 0,2 метри від підлоги. Вимикачі на стінах розташовані на висоті 1,2 метра від підлоги.

Електрична мережа вмикається і вимикається за допомогою пускової апаратури (рубильником).

Для запобігання ураження людини струмом служить захисне заземлення. Сучасні ПК включаються в розетку за допомогою триполюсної вилки, один з полюсів якої використовується для заземлення. З ним електрично з'єднані металеві неструмоведучі частини ПК, які можуть опинитися під напругою. В обчислювальному центрі передбачається використання тільки триполюсних розеток, у яких відповідний полюс з'єднаний з шиною заземлення.

Кожен блок живлення комп'ютера або периферійного пристрою має мережевий фільтр. Конденсатори цього фільтра призначені для шунтування високочастотних перешкод живлячої мережі на землю через дріт захисного заземлення [22].

При з'єднанні двох пристроїв (комп'ютера та принтера) інтерфейсним кабелем, загальний провід інтерфейсів послідовних і паралельних портів пов'язаний зі «схемної землею» і корпусом пристрою. Якщо пристрої, які з'єднуються, надійно заземлені через окремий провід на загальний контур, проблеми різниці потенціалів не виникає.

Згідно з Правилами улаштування електроустановок опір захисного заземлення в обчислювальному центрі становить 3 Ом (за стандартом – не більше 4 Ом) [20]. В приміщенні встановлено автоматичні вимикачі, пристрій заземлення контуру.

7.3 Заходи щодо виробничої санітарії та гігієни праці

Заходи щодо забезпечення виробничої санітарії та гігієни праці для обчислювального центру обладнаного ПК з ВДТ розроблені відповідно до вимог Державних санітарних норм та правил «Гігієнічна класифікація праці за показниками шкідливості та небезпечності факторів виробничого середовища, важкості та напруженості трудового процесу», МІОУ 06.05.2014 р. за № 472/25249, ДСанПіН 3.3.2.007-98 «Державні стандартні правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин» [23].

Тип роботи програмістів в обчислювальному центрі належить до операторської роботи, даний тип роботи відносить до категорії 1а, тобто до неї належать роботи, що виконуються сидячи і не потребують фізичного напруження, робоче місце є постійним, робота проводиться у холодний та теплий періоди року, тому рекомендується підтримувати оптимальні показники мікроклімату.

Роботи, що виконуються сидячи і не потребують фізичного напруження, при яких витрати енергії складають до 139 Вт. Для забезпечення оптимального рівня параметрів повітряного виробничого середовища використовуємо ДСН 3.3.6-042-99 «Санітарні норми мікроклімату виробничих приміщень» (табл. 7.1) [24]

Таблиця 7.1 – Норми мікроклімату в приміщенні

Період Року	Категорія робіт	Температура повітря, °C	Відносна вологість повітря, %	Швидкість руху повітря, м/с
Холодний	Легка	22–24	40–60	0,1
Теплий	Легка	23–25	40–60	0,1-0,2

В обчислювальному центрі передбачено: устрій системи водяного опалення приміщення для забезпечення необхідної температури повітря в холодний період року відповідно ДБН В.2.5-67:2013 «Опалення, вентиляція та кондиціювання» [25].

Проведені дослідження мікрокліматичних умов на комп'ютеризованих робочих місцях показали, що зимою значення відносної вологості повітря часто можуть нижчими за встановлені норми і становити в середньому 30–40%. Це призводить не лише до надмірного висихання слизових оболонок очей, носа, горла, а й до нагромадження зарядів статичної електрики, що утворюються в процесі роботи комп'ютера. Для усунення таких проблем, в приміщенні передбачена наявність 3 зволожувачів повітря, які допомагають підняти вологість до 60%. Кількість зволожувачів зумовлена тим, що один такий апарат розраховано на приміщення площею до 15 м².

У теплий період року для підтримки нормальної температури повітря в приміщенні використовуються кондиціонери типу «спліт-система». Два кондиціонера розташовані у протилежних частинах обчислювального центру дозволяють підтримувати температуру повітря не вище 24°C. Кількість кондиціонерів зумовлена тим, що один такий апарат розраховано на кондиціювання приміщення площею до 18 м².

Вимоги до освітлення регламентовані ДБН В.2.5-28-2018 «Природне та штучне освітлення» [26], згідно з якими освітлення у приміщеннях з ЕОМ суміщене, при якому недостатнє за нормами природне освітлення доповнене штучним. Природне освітлення бічне, одностороннє.

В обчислювальному центрі коефіцієнт природної освітленості (КПО) має становити в середньому 1,1% (1,2% – у сонячний день та 1% – у хмарну погоду), це відповідає до вимогам ДБН В.2.5-28-2018. Для забезпечення відносної постійності природного освітлення вікна обладнанні сонцезахисними металевими жалюзіями з системою регулювання. В таких жалюзіях коефіцієнт відбиття становить $K_b = 0,7$. Робочі місця розташовано так, щоб у поле зору не потрапляли вікна або світлі поверхні світильників.

Найбільш прийнятними для приміщень, де працюють ПК, є люмінесцентні лампи ЛБ (білого світла) і ЛТБ (тепло-білого світла) потужністю 20, 40 або 80 Вт.

Розрахуємо кількість ламп необхідних для освітлення приміщення розміром $a \times b \times c$, в нашому випадку $9\text{м} \times 4\text{м} \times 3,5\text{м}$, освітленістю $E_n = 400\text{лк}$. Коефіцієнт відбиття стелі 70% і стін 50%. Для освітлення використовуються люмінесцентні лампи типу ЛБ в світильниках ЛПО. Знаходимо індекс приміщення за формулою (7.1)

$$i = \frac{a \cdot b}{(c - K_v) \cdot (a + b)}, \quad (7.1)$$

де a – довжина приміщення;

b – ширина приміщення;

c – висота приміщення;

K_v – коефіцієнт відбиття.

Підставивши всі величини у формулу (7.1), отримаємо:

$$i = \frac{9 \cdot 4}{(3,5 - 0,7) \cdot (9 + 4)} = 0,98.$$

Приймаємо коефіцієнт запасу $k_3 = 1,6$ і коефіцієнт нерівномірності освітлення $z = 1,1$. При індексі приміщення $i = 0,98$ отримуємо $\eta = 43\%$.

Світильники розміщуємо в два ряди ($N_p = 2$).

Визначаємо необхідний світловий потік ламп в кожному ряду за формулою (7.3):

$$\Phi_\Sigma = \frac{E_n \cdot S \cdot k_3 \cdot z}{N_p \cdot \eta}, \quad (7.2)$$

де E_n – освітленість приміщення;

S – площа приміщення;

k_3 – коефіцієнт запасу;

z – коефіцієнт нерівномірності освітлення;

N_p – число рядів світильників.

З формули (7.2) маємо:

$$\Phi_{\Sigma} = \frac{400 \cdot 36 \cdot 1,6 \cdot 1,1}{2 \cdot 43} = 34267 \text{ лм}$$

Якщо в світильнику встановити по дві лампи ЛБ потужністю 80 Вт і світловим потоком $\Phi_{\text{л}} = 5400$ лм, то необхідне число світильників в ряду складе

$$N_{\text{св.р}} = \frac{\Phi_{\Sigma}}{n \cdot \Phi_{\text{л}}} = \frac{34267}{2 \cdot 5400} = 3,17 = 4 \text{ шт.} \quad (7.3)$$

Для виключення засвічення екранів дисплеїв прямими світловими потоками світильники загального освітлення мають в своєму розпорядженні збоку від робочого місця, паралельно лінії зору програміста і стіні з вікнами.

Під впливом шуму знижується концентрація уваги, порушуються фізіологічні функції, з'являється стомленість у зв'язку з підвищеними енергетичними витратами і нервово-психічною напругою, погіршується мовна комутація.

Згідно із технічним паспортом обладнання (ПК, монітори, принтери, клавіатури (при натисканні), маніпулятори типу «миша» (при натисканні)), що використовується для роботи, рівень шуму складає: 43 дБА. Згідно із ДСН 3.3.6.037-99 «Санітарні норми виробничого шуму, ультразвуку та інфразвук» нормоване значення рівня шуму більше, ніж фактичне, тож додаткових робіт для усунення надмірного шуму не проводилося. Роботи в

даному приміщенні з таким обладнанням відповідають нормативним вимогам [27].

На програміста, в період роботи за дисплеєм, здійснюється інтенсивне електромагнітне випромінювання, яке може стати причиною професійних захворювань. Комп'ютери потрапляють за визначенням в 5-й діапазон (низькі частоти).

Основним джерелом електромагнітного випромінювання, у тому числі рентгенівського, в приміщенні є електронно-променеві трубки (ЕПТ) старих моніторів. Через це, для зниження потужності експозиційної дози рентгенівського випромінювання для роботи використовуються монітори з рідкокристалічними дисплеями.

Обладнання і організація робочого місця в обчислювальному центрі мають забезпечувати відповідність конструкції всіх елементів робочого місця та їх взаємного розташування ергономічним вимогам з урахуванням характеру і особливостей трудової діяльності.

Площа на одне робоче місце з ПК складає по 6 м^2 , а об'єм – по 20 м^3 . Приміщення має природне та штучне освітлення. Робочі місця розташовано так, щоб природне світло падало збоку.

Робочі місця та взаємне розташування всіх його елементів відповідати антропометричним, фізичним і психологічним вимогам.

Робочі стільці мають підйомно-повторним механізми та регулюються по висоті та куту нахилу сидіння і спинки, а також по відстані спинки від переднього краю сидіння. Регулювання кожного параметра незалежне, легко здійснюється та має надійну фіксацію. Екран монітору повинний знаходитися на оптимальній відстані 700 мм.

Робота з комп'ютером характеризується значною розумовою напругою і нервово-емоційним навантаженням операторів, високою напруженістю зорової роботи і достатньо великим навантаженням на м'язи рук при роботі з клавіатурою ЕОМ. Велике значення має раціональна

конструкція і розташовує елементів робочого місця, що важливе для підтримки оптимальної робочої пози людини-оператора [23].

В приміщенні обчислювального центру наявні 2 медичні аптечки першої допомоги.

Також приміщення обладнано шафами для зберігання документів, магнітних дисків, полицями, стелажми та тумбами.

Колір приміщень і меблів сприяє створенню комфортних умов для зорового сприйняття.

Загальний план обчислювального центру та робочих місць наведено на рисунку 7.1

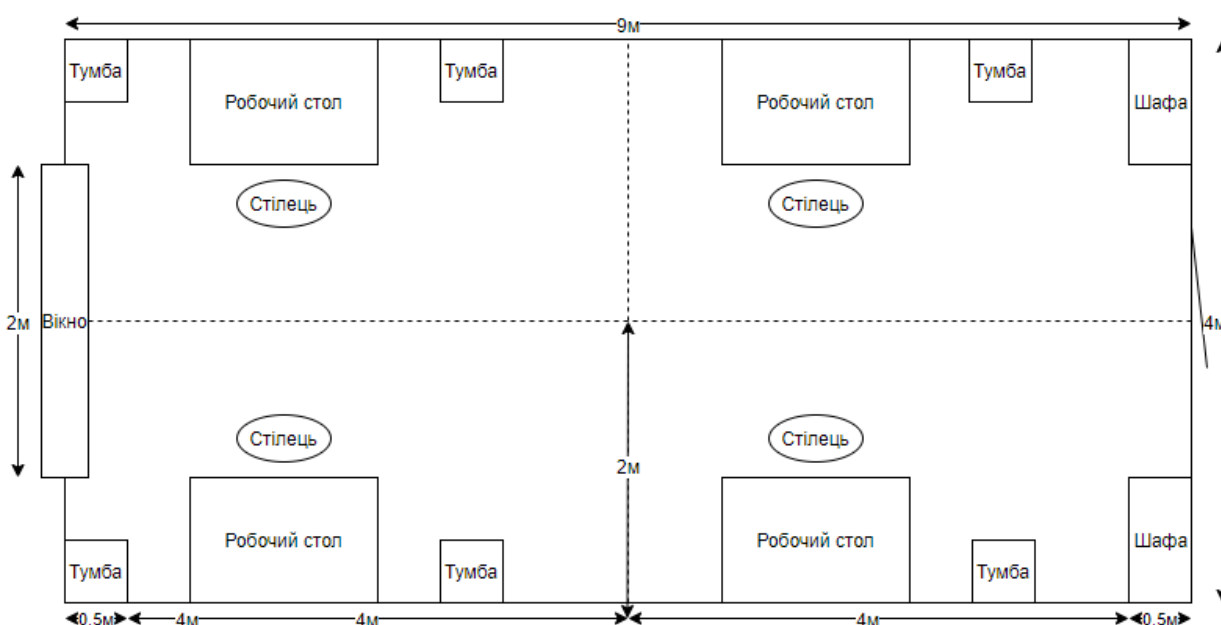


Рисунок 7.1 – Загальна схема обчислювального центру

7.4 Заходи з пожежної безпеки

Пожежі в обчислювальному центрі становлять особливу небезпеку, тому що пов'язані з великими матеріальними втратами. Як відомо, пожежа може виникнути при взаємодії горючих речовин, окислення і джерел

запалювання. У приміщеннях ВЦ присутні всі три основні чинники, необхідні для виникнення пожежі.

Для відводу теплоти від ПК у обчислювальному центрі постійно діє потужна система кондиціонування. Тому кисень, як окислювач процесів горіння, є в будь-якій точці обчислювального центру.

Обчислювальний центр, згідно ДСТУ Б В.1.1-36:2016 «Визначення категорій приміщень, будинків та зовнішніх установок за вибухопожежною та пожежною небезпекою» [28] відноситься до категорії «П-Па», а клас можливої пожежі, згідно ДБН В.1.1-7:2016 «Пожежна безпека об'єктів будівництва. Загальні вимоги» [29], визначається, як «А».

До засобів гасіння пожежі, призначених для локалізації невеликих загорянь, відносяться пожежні стовбури, внутрішні пожежні водопроводи, вогнегасники, сухий пісок, азбестові ковдри і т. п.

Оскільки приміщення (дослідницької лабораторії, конструкторського бюро, тощо) що обладнане ПК з ВДТ має площу 36 м², тому відповідно до вимог п. 3.8 розділу «Типові норми належності вогнегасників» ДСТУ 4297:2004 «Пожежна техніка. Технічне обслуговування вогнегасників. Загальні технічні вимоги» [30] для гасіння електроустановок, що знаходяться під напругою, передбачені вуглекислотні вогнегасники типу ВВК-3,5 у кількості 2 штук (з розрахунку один вогнегасник с величиною заряду вогнегасної речовини 3 кг. і більше, на 20 м² площі приміщення). Відстань між вогнегасниками та місцями можливих загорянь не перевищує 10 м.

Усі працівники обчислювального центру під час прийняття на роботу і в процесі праці повинні проходити протипожежний інструктаж, перевірку знань з питань пожежної безпеки. Евакуаційні виходи і проходи утримуються постійно вільними та нічим не захащуються.

Відповідальний за протипожежний стан приміщень обчислювального центру повинен періодично (не рідше одного разу на місяць) перевіряти працездатність пристрою, який забезпечує автоматичне вимкнення системи вентиляції у разі пожежі, а також вогнедимозатримувальних пристроїв. Під

час ремонту слід не допускати руйнування або порушення цілісності негорючих діафрагм у фальш-підлоговому просторі.

Фільтри припливно-витяжної вентиляції слід чистити згідно з затвердженим графіком, але не рідше двох разів на рік. Роботи з ремонту вузлів (блоків) ЕОМ, як правило, проводити не дозволяється. Їх необхідно проводити в окремому приміщенні (майстерні). У разі необхідності проведення ремонту або технічного обслуговування ЕОМ безпосередньо в залі допускається мати не більше 0,5 л легкозаймистих рідин у небиткій, щільно закритій тарі. Для промивання деталей необхідно застосовувати, як правило, негорючі миючі препарати. Чарунки та інші знімні пристрої допускається промивати горючими рідинами тільки в спеціальному приміщенні, обладнаному припливно-витяжною вентиляцією.

Відповідальний за пожежну безпеку приміщень обчислювального центру зобов'язаний негайно сповістити технічну службу у разі виявлення незадовільного технічного стану димових та теплових сповіщувачів автоматичної пожежної сигналізації, а також про несправності автоматичної установки об'ємного (газового) пожежогасіння.

Усі працівники обчислювального центру повинні вміти привести в дію (в разі потреби) первинні засоби пожежогасіння. Регулярно, але не рідше одного разу на квартал, слід очищати від пилу ЕОМ, контрольновимірювальну апаратуру, кабельні траншеї та фальшпідлоговий простір за допомогою пиловсмоктувача.

7.5 Заходи щодо забезпечення безпеки в надзвичайних ситуаціях

Інформування, оповіщення і дії робітників та службовців промислового об'єкта, при загрозі виникнення надзвичайної ситуації.

Головним і невід'ємним елементом всієї системи захисту населення і територій від надзвичайних ситуацій техногенного та природного характеру є інформація та оповіщення.

Зміст інформації мають становити відомості про надзвичайні ситуації, що прогножуються або вже виникли, з визначенням їхньої класифікації, меж поширення і наслідків, а також заходи реагування на них.

Оперативну і достовірну інформацію про стан захисту населення і територій від надзвичайних ситуацій техногенного та природного характеру, методи та способи їх захисту, заходи безпеки зобов'язані надавати населенню через засоби масової інформації центральні та місцеві органи виконавчої влади та виконавчі органи рад.

Оповіщення про загрозу виникнення надзвичайних ситуацій і постійне інформування про них населення забезпечується шляхом:

- завчасного створення, підтримання в постійній готовності загальнодержавної і територіальних автоматизованих систем централізованого оповіщення населення;

- організаційно-технічного з'єднання територіальних систем централізованого оповіщення і систем оповіщення на об'єктах господарювання;

- завчасного створення та організаційно-технічного з'єднання із системами спостереження і контролю постійно діючих локальних систем оповіщення та інформування населення в зонах можливого катастрофічного затоплення, районах розміщення радіаційних і хімічних підприємств та інших об'єктів підвищеної небезпеки;

- централізованого використання загальнодержавних і галузевих систем зв'язку, радіопровідного, телевізійного оповіщення, радіотрансляційних мереж та інших технічних засобів передавання інформації.

Оповіщення населення про загрозу і виникнення надзвичайної ситуації у мирний, в особливий період та постійне інформування його про наявну обстановку – одне з важливих завдань цивільного захисту України. Для цього створюється система, організаційно-технічне об'єднання засобів для передачі сигналів і розпоряджень органів управління цивільного захисту.

Система оповіщення та інформативного забезпечення створюється завчасно в усіх ланках пунктів управління.

Основу системи оповіщення утворюють автоматизована система централізованого оповіщення мережі зв'язку та радіомовлення, а також спеціальні засоби.

Автоматизована система оповіщення створюється завчасно на базі загальнодержавної мережі зв'язку та радіомовлення і поділяється на державну та територіальну. Вона може забезпечити оповіщення населення, поєднавши місцеву телефонну мережу для подачі сигналу «Увага всім!» та повну інформацію за допомогою засобів радіомовлення й телебачення.

Оповіщення підпорядкованих штабів, органів управління цивільного захисту і населення організовується і забезпечується старшими органами управління.

Додатково в Україні створюється локальна державна система оповіщення про загрозу катастрофічного затоплення в разі руйнування гідроспоруд на р. Дніпро.

Відповідальним за організацію оповіщення про загрозу і виникнення надзвичайної ситуації і постійне інформування населення про становище є органи управління цивільного захисту відповідної адміністративно-територіальної одиниці.

Сигнали передаються каналами зв'язку, радіотрансляційними мережами і засобами телебачення. Одночасно з інформацією про надзвичайну ситуацію передаються вказівки про порядок дій формувань цивільного захисту і населення. Сигнали, подані вищим органом управління цивільного захисту, мають дублюватися всіма підпорядкованими підсистемами. Дублювати сигнали на об'єктах і в населених пунктах можна за допомогою місцевого радіотрансляційного вузла, гудків підприємств, сирен транспорту, ударами в рейку, дзвони.

Для своєчасного попередження населення введені сигнали попередження населення у мирний і воєнний час.

Сигнал «Увага всім!» повідомляє населення про надзвичайну обстановку в мирний час і на випадок загрози нападу противника у воєнний час. Сигнал подається органами цивільного захисту за допомогою сирени і виробничих гудків. Тривалі гудки означають попереджувальний сигнал.

Почувши їх, необхідно включити радіо, телевізор і прослухати текст інформації про дії населення після одержання сигналу. Якщо немає радіо, телевізора або вони не працюють, слід з'ясувати значення і зміст інформації у сусідів або інших людей, які знають про неї.

Після одержання інформації необхідно виконати всі вказівки тексту інформації сигналу [31].

Сигнали і варіанти оповіщення населення в мирний час такі.

«Аварія на атомній електростанції». Повідомляються місце, час, масштаби аварії, інформація про радіаційну обстановку та дії населення. Якщо є загроза забруднення радіоактивними речовинами, необхідно провести герметизацію житлових, виробничих і складських приміщень. Провести заходи захисту від радіоактивних речовин сільськогосподарських тварин, кормів, урожаю, продуктів харчування та води. Прийняти йодні препарати. Надалі діяти відповідно до вказівок штабу органів цивільного захисту.

«Аварія на хімічно небезпечному об'єкті». Повідомляються місце, час, масштаби аварії, інформація про можливе хімічне зараження території, напрямки та швидкість можливого руху зараженого повітря, райони, яким загрожує небезпека. Дається інформація про поведінку населення. Залежно від обставин: залишатися на місці, у закритих житлових приміщеннях, на робочих місцях чи залишати їх і, застосувавши засоби індивідуального захисту, вирушити на місця збору для евакуації або в захисні споруди. Надалі діяти відповідно до вказівок штабу органів управління цивільного захисту.

«Землетрус». Подається повідомлення про загрозу землетрусу або його початок. Населення попереджається про необхідність відключити газ, воду, електроенергію, погасити вогонь у печах; повідомити сусідів про

одержану інформацію; взяти необхідний одяг, документи, продукти харчування, вийти на вулицю і розміститися на відкритій місцевості на безпечній відстані від будинків, споруд, ліній електропередачі.

«Затоплення». Повідомляється район, в якому очікується затоплення в результаті підйому рівня води в річці чи аварії дамби.

Населення, яке проживає в даному районі, повинне взяти необхідні речі, документи, продукти харчування, воду, виключити електроенергію, відключити газ і зібратись у вказаному місці для евакуації. Повідомити сусідів про стихійне лихо і надалі слухати інформацію штабу органів управління цивільного захисту.

«Штормове попередження». Подається інформація для населення про посилення вітру. Населенню необхідно зачинити вікна, двері. Закрити в приміщеннях сільськогосподарських тварин. Повідомити сусідів. Населенню, по можливості, перейти в підвали, погреби.

Сигнали оповіщення населення у воєнний час такі.

Сигнал «Повітряна тривога» подається для всього населення.

Попереджається про небезпеку ураження противником даного району. По радіо передається текст: «Увага! Увага! Повітряна тривога! Повітряна тривога!» Одночасно сигнал дублюється сиренами, гудками підприємств і транспорту. Тривалість сигналу 2–3 хв.

При цьому сигналі об'єкти припиняють роботу, транспорт зупиняється і все населення укривається в захисних спорудах. Робітники і службовці припиняють роботу відповідно до інструкції і вказівок адміністрації. Там, де неможливо через технологічний процес або через вимоги безпеки зупинити виробництво, залишаються чергові, для яких мають бути захисні споруди.

Сигнал може застати у будь-якому місці й будь-який час. В усіх випадках необхідно діяти швидко, але спокійно, впевнено, без паніки.

Суворо дотримуватися правил поведінки, вказівок органів цивільного захисту.

Сигнал «Відбій повітряної тривоги». Органами цивільного захисту через радіотрансляційну мережу передається текст: «Увага! Увага! Громадяни! Відбій повітряної тривоги!». За цим сигналом населення залишає захисні споруди і повертається на свої робочі місця і в житла.

Сигнал «Радіаційна небезпека» подається в населених пунктах і в районах, в напрямку яких рухається радіоактивна хмара, що утворилася від вибуху ядерного боєприпасу.

Почувши цей сигнал, необхідно з індивідуальної аптечки АІ-2 прийняти 6 таблеток радіозахисного препарату № 1 із гнізда 4, надіти респіратор, протипилову пов'язку, ватно-марлеву маску або протигаз, взяти запас продуктів, документи, медикаменти, предмети першої потреби і направитися у сховище або ПРУ.

Сигнал «Хімічна тривога» подається у разі загрози або безпосереднього виявлення хімічного або бактеріологічного нападу (зараження). При цьому сигналі необхідно прийняти з індивідуальної аптечки АІ-2 одну таблетку препарату при отруєнні фосфорорганічними речовинами з пенала з гнізда 2 або 5 таблеток протибактеріального препарату № 1 із гнізда 5, швидко надіти протигаз, а за необхідності – і засоби захисту шкіри, якщо можливо, та укритися в захисних спорудах. Якщо таких поблизу немає, то від ураження аерозолями отруйних речовин і бактеріальних засобів можна сховатися в житлових чи виробничих приміщеннях.

При застосуванні противником біологічної зброї населенню буде подана інформація про наступні дії.

Успіх захисту населення залежатиме від дисциплінованості, своєчасної і правильної поведінки, суворого дотримання рекомендацій і вимог органів цивільного захисту [32].

ВИСНОВКИ

У ході виконання дипломної кваліфікаційної роботи магістра було проведено огляд та аналіз корпоративних інформаційних ERP систем. Виконано дослідження ERP системи SAP R/3. Відібрані матеріали були систематизовані, проаналізовані, узагальнені і представлені в першому розділі пояснювальної записки.

Сформульовано та класифіковано основні критерії вибору СКБД для побудови корпоративної інформаційної системи. На підставі аналізу в якості основи для побудови модулів ERP системи обрано СКБД Progress. Як базис для створення модулів ERP системи було прийнято рішення обрати мову АВАР, яка представляє собою пропріетарну внутрішню мову програмування високого рівня для розробки SAP систем.

Складено технічне завдання, в якому зазначена мета розробки та її призначення, основні вимоги до програми, порядок її функціонування.

Наукова новизна роботи полягає у тому, що запропоновано модель модуля ERP системи територіально розподіленого підприємства, що надає структуру взаємодії з зовнішніми об'єктами, діаграму взаємодії апаратних засобів ERP системи, узагальнену модель системи виробництва на основі планування та діаграму потоку замовлень. Розроблена модель модуля ERP системи територіально розподіленого підприємства складається з набору діаграм, що характеризують особливості бізнес-процесів підприємства, та дозволяють розробляти програмне забезпечення для їх автоматизації.

Практичне значення роботи полягає у тому, що розроблено програмні модулі ERP системи територіально розподіленого підприємства, що забезпечує підтримку внутрішніх бізнес-процесів підприємства та що надає можливості для взаємодії з зовнішніми об'єктами. Основними модулями програми є: модуль введення даних, розрахунковий модуль, довідковий модуль, модуль виведення, модуль грошових операцій, модуль товарних операцій, модуль прав доступу, модуль налаштування бізнес-процесів.

Написано комплект програмної документації, що включає в себе опис програми, керівництво програміста, керівництво оператора.

Проведені експерименти засвідчили, що програма має достатню працездатність і може використовуватися за призначенням.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Bradford M. Modern ERP: Select, Implement, and Use Today's Advanced Business Systems / M. Bradford. – Cary, NC : SAS Institute Inc. – 2015. – 266 p.
2. Iinuma B. Enterprise Resource Planning Systems: 12 Frequently Asked Questions About Enterprise Resource Planning Systems / B. Iinuma, E. Klauss. – NY: Strategic Systems Group Inc. – 2017. – 118 p.
3. Wagner B. Concepts in Enterprise Resource Planning / B. Wagner, E. Monk. – NY : Course Technology, Cengage Learning. – 2020. – 254 p.
4. Елашкин М. SAP Business One. Строим эффективный бизнес / М. Елашкин. – М. : КУДИЦ-Пресс, 2007. – 238 с.
5. Рыбников А. И. Система управления предприятием типа ERP / А. И. Рыбников. – М. : Азроконсалт, 1999. – 214 с.
6. Андерсон Д. SAP за 24 часа / Д. Андерсон, Д. Ларокка. – К. : Баланс Бизнес Букс, 2007. – 432 с.
7. Беккер Й. Менеджмент процессов / Й. Беккер, М. Кугелер, М. Роземанн. – М. : Эксмо, 2007. – 324 с.
8. Кречмер Р. Разработка приложений SAP R/3 на языке ABAP/4 / Р. Кречмер В. Вейс. – М. : Лори, 1998. – 368 с.
9. Кале В. SAP R/3. Руководство для менеджеров и инженеров / В. Кале. – М. : Компания АйТи, 2006. – 511 с.
10. Вилл Л. Системное администрирование SAP R/3. Официальное руководство SAP / Л. Вилл. – М. : Лори, 1999. – 416 с.
11. Построение эффективной системы управления. Официальное руководство SAP. – К. : Альпина Бизнес Букс, 2008. – 144 с.
12. Хагеман С. SAP R/3. Системное администрирование / С. Хагеман, Л. Вилл. – М. : Лори, 2007. – 480 с.
13. 1С: Предприятие 8.1. Конфигурирование и администрирование. – М. : ЗАО «1С», 2007. – 1110 с.

14. Гладкий А. А. 1С: Предприятие 8.0 / А. А. Гладкий. – СПб. : Тритон, 2005. – 256 с.
15. Вольфганг В. Developing SAR's R/3. Applications with ABAP/4 / В. Вольфганг, К. Рюдигер. – М. : Лори, 2007. – 384 с.
16. Шульц О. Using SAP: A Guide for Beginners and End Users / О. Шульц. – М.: Эксперт РП, 2012. – 416 с.
17. Кале В. SAP ERP. Построение эффективной системы управления / В. Кале. – М.: Издательство: Альпина Бизнес Букс, 2012. – 154 с.
18. Экономика и организация производства программных продуктов [Электронний ресурс]. – 2019. – Режим доступу: <http://megaobzor.net/e-book/book.html>.
19. Депутат О. П. Цивільна оборона. Навчальний посібник. / О. П. Депутат, І. В. Коваленко, І. С. Мужик. — Львів.: Афіша, 2001. — 336 с.
20. Правила улаштування електроустановок [Текст] : ПУЕ-2017. – На заміну ПУЕ-86 ; чинний з 2017-08-21. – К. : Міненерговугілля України, 2017. – 617 с.
21. . ДСТУ Б В.2.5-82:2016 Електробезпека в будівлях і спорудах. Вимоги до захисних заходів від ураження електричним струмом. - Київ, ДП "УкрНДНЦ", 2016. – 109 с.
22. Охорона праці [З.М. Яремко, С.В. Тимошук, С.В. Писаревська та ін.]; за ред. З.М. Яремка. – Львів: ЛНУ імені Івана Франка, 2018. – 430 с.
23. Державні санітарні правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин [Електронний ресурс] : ДСанПіН 3.3.2.007-98. – Чинний від 1998-12-10. – К. : МОЗ України, 1998. – URL: <http://mozdocs.kiev.ua/view.php?id=2445>. – (Державні санітарні правила та норми).
24. ДСН 3.3.6.042-99 Санітарні норми мікроклімату виробничих приміщень. / Харків: Форт, 2000 – 24 с.
25. Основи професійної безпеки та здоров'я людини : підручник / В. В. Березуцький [та ін.] ; під ред. проф. В. В. Березуцького. – Харків : НТУ

“ХПІ”, 2018. – 553 с

26. ДБН В.2.5-28:2018. Природне та штучне освітлення. Посібник: проблеми природного і штучного освітлення. / К: Етін, 2019. – 180 с.

27. ДСН 3.3.6.037-99 «Санітарні норми виробничого шуму, ультразвуку та інфразвуку / МОЗ України, 1999 – 24 с.

28. Визначення категорій приміщень, будинків та зовнішніх установок за вибухопожежною та пожежною небезпекою [Текст] : ДСТУ Б В.1.1-36:2016. – На заміну НАПБ Б.03.002-2007 ; чинний від 2017-01-01. – К. : Мінрегіонбуд України, 2016. – 66 с. – (Державний Стандарт України)

29. Пожежна безпека об'єктів будівництва. Загальні вимоги [Текст] : ДБН В.1.1-7:2016. – На заміну ДБН В.1.1.7-2002 ; чинний від 2017-06-01. – К. : Мінрегіон України, 2017. – 47 с. – (Державні будівельні норми).

30. Жидецький В. Ц. Основи охорони праці [Текст] : підручник / В. Ц. Жидецький. – 5-те вид., доп. – К. : Знання, 2014. – 373 с. + 1 ел. опт. диск (CD-ROM). – ISBN 978-617-07-0134-3

31. Шоботов В. М. Цивільна оборона [Текст] : Навчальний посібник / В. М. Шоботов. – Вид. 2-ге, перероб. – К. : Центр навчальної літератури, 2006. – 438 с.

32. Цивільний захист: Конспект лекцій для студентів всіх спеціальностей і форм навчання / В.О. Васійчук, В.Є. Гончарук, О.С. Дацько та др. / За ред. В.Є. Гончарука. – Л: ВЛп, 2011. – 208 с.

ДОДАТОК А
Технічне завдання

Вступ

Програма призначена для оброблення даних територіально розподіленого підприємства.

A.1 Підстави для розробки

Підставою для розробки є завдання на дипломну роботу на тему «Дослідження та програмна реалізація модулів ERP системи», затверджене наказом по Національному університету «Запорізька політехніка».

A.2 Призначення розробки

Програма призначена для оброблення даних територіально розподіленого підприємства.

A.3 Вимоги до програми

A.3.1 Вимоги до функціональних характеристик

Система повинна забезпечувати підтримку наступних можливостей:

- зчитування даних з файлів бази даних;
- розрахунок заробітної плати;
- керування фінансами;
- керування замовленнями і поставками;
- керування складом;
- керування торговим залом;
- керування продажами;
- керування виробництвом;
- керування експлуатацією майна підприємства;
- можливість настройки призначеного для користувача інтерфейсу.

А.3.2 Вимоги до надійності

Програма повинна забезпечувати коректну обробку помилок у наступних ситуаціях: введення користувачем некоректних даних; відсутність необхідних довідкових даних у базі даних.

А.3.3 Вимоги до експлуатації

Для роботи з програмою користувач повинен володіти достатніми знаннями для коректної роботи з програмно-апаратним комплексом.

А.3.4 Вимоги до складу й параметрів технічних засобів

Для роботи з програмним продуктом рекомендується використовувати наступне апаратне забезпечення:

- ПК під керівництвом операційної системи;
- тактова частота процесора 2 ГГц або більше, Video 512 Мб;
- обсяг оперативної пам'яті не менше 1 Гб;
- обсяг вільного простору на диску - не менше 100 Мб;
- дозвіл екрана монітора не менше 1024x768;
- маніпулятор миша.

А.3.5 Вимоги до інформаційної й програмної сумісності

Для роботи програми необхідна операційна система.

А.3.6 Вимоги до транспортування і зберігання

Програма повинна зберігатися на жорсткому або гнучкому магнітному диску і транспортуватися в спеціальних боксах.

A.3.7 Вимоги до програмної документації

Програма повинна поставлятися з технічним завданням, керівництвом оператора, керівництвом програміста.

A.4 Стадії та етапи розробки

Програмний продукт повинен бути виконаний у 16-ти тижневий строк з моменту отримання завдання.

ДОДАТОК Б
Опис програми

Б.1 Загальні відомості

Програма розроблена за допомогою середовища візуального програмування АВАР.

Б.2 Функціональне призначення

Програма призначена для оброблення даних територіально розподіленого підприємства.

Б.3 Опис логічної структури

Програма представляє собою набір функціональних модулів, основними з яких є: модуль введення даних, розрахунковий модуль, довідковий модуль, модуль виведення, модуль грошових операцій, модуль товарних операцій, модуль прав доступу, модуль налаштування бізнес-процесів.

Б.4 Використані технічні засоби

Для роботи з програмою рекомендується використовувати наступне апаратне забезпечення:

- ПК під керівництвом операційної системи;
- тактова частота процесора 2 ГГц або більше, Video 512 Мб;
- обсяг оперативної пам'яті не менше 1 Гб;
- обсяг вільного простору на диску - не менше 100 Мб;
- дозвіл екрана монітора не менше 1024x768;
- маніпулятор миша.

Б.5 Виклик і завантаження

Розроблена програма поширюється у вигляді інсталяційного пакета. Для можливості роботи з програмою необхідно розмістити файли модулів програмного забезпечення на сервері. Перед використанням програми її необхідно встановити. Для цього необхідно запустити програму `setup.exe`, яка знаходиться на установчому диску. У відповідь на запит установчої програми потрібно вказати шлях встановлення програми. Після цього програма скопіює необхідні файли, створить і помістить в головне меню системи ярлик для запуску програми і зробить необхідні зміни в файлі реєстрації. Після завершення процедури установки програму можна використовувати. Програму можна запустити будь-якими засобами операційної системи користувача.

Б.6 Вхідні дані

Вхідними даними для програми є інформація з документів різного призначення, зокрема платіжних документів, а також інформація про кількість продукції на складі, вартість продукції, постачальників, клієнтів і т.ін.

Б.7 Вихідні дані

Вихідними даними для програми є розрахунки по бізнес-процесам організації, заробітній платі, вартості відпускнуї продукції, управління замовленнями і поставками, управління складом, які можуть бути збережені на жорсткому диску.

ДОДАТОК В
Текст програми

```
*&-----*
*& Include      ZUE_FI001_FORMS
*&-----*
```

form prepare_for_selection.

```
data: l_NN_BLART like BKPF-BLART,
      l_NK_BLART like BKPF-BLART,
      l_UMSKZ type UMSKZ.
```

```
clear:      t_tab_AVANSPLUS,      t_tab_AVANSPLUS[],      t_tab_AVANS,
t_tab_AVANS[].
```

```
CLEAR: J_HKONT, r_BLART[], r_UMSKZ[].
```

```
select SINGLE HKONT NN_BLART NK_BLART
into (J_HKONT, l_NN_BLART, l_NK_BLART)
from J_1UF_NAKL_MAIN
WHERE VERSION EQ 'PUR'
and BUKRS EQ pBUKRS
and SHKZG EQ 'H'.
```

```
*clear: r_BLART, r_BLART[].
```

```
*r_BLART-low = 'NN'.
```

```
*r_BLART-sign = 'I'.
```

```
*r_BLART-option = 'EQ'.
```

```
*append r_BLART.
```

```
*r_BLART-low = 'NK'.
```

```
*r_BLART-sign = 'I'.
```

```
*r_BLART-option = 'EQ'.
```

```
*append r_BLART.
```

```
if NOT l_NN_BLART is INITIAL.
```

```
  r_BLART-low = l_NN_BLART.
```

```
  r_BLART-sign = 'I'.
```

```
  r_BLART-option = 'EQ'.
```

```
  append r_BLART.
```

```
endif.
```

```
if NOT l_NK_BLART is INITIAL.
```

```
  r_BLART-low = l_NK_BLART.
```

```
  r_BLART-sign = 'I'.
```

```
  r_BLART-option = 'EQ'.
```

```
  append r_BLART.
```

```
endif.
```

```
if NOT pBLART_d is INITIAL.
```

```
  r_BLART-low = pBLART_d.
```

```
  r_BLART-sign = 'I'.
```

```
  r_BLART-option = 'EQ'.
```

```
  append r_BLART.
```

```
endif.
```

```

SELECT UMSKZ
  INTO l_UMSKZ
  from T074U
    WHERE KOART EQ 'K'
      and UMSKS EQ 'A'.
r_UMSKZ-low = l_UMSKZ.
r_UMSKZ-sign = 'T'.
r_UMSKZ-option = 'EQ'.
append r_UMSKZ.
ENDSELECT.

```

```

clear: t_lfa, t_lfa[].
SELECT LIFNR NAME1 STCD1 STCD2
  FROM LFA1
  into TABLE t_lfa
* WHERE

```

```

  if s_LIFNR IS INITIAL
  and s_EBEL2 IS INITIAL
  and s_EBELN IS INITIAL.
    g_Variant = 1.
  else.
    g_Variant = 2.
  ENDIF.

```

```
endform.
```

```
FORM get_yy CHANGING g_OK TYPE I.
```

```
clear: t_tab_h.
```

```
case g_Variant.
  when 1.

```

```
  if tabNN1-BLART EQ pBLART_d.
```

```

select SINGLE YYEBELN YYEBEL2 YYLIFNR
  into (t_tab_h-YYEBELN, t_tab_h-YYEBEL2, t_tab_h-LIFNR)
  from BSIS
    WHERE BUKRS EQ tabNN1-BUKRS
      and BELNR EQ tabNN1-BELNR
      and GJAHR EQ tabNN1-GJAHR
*    and SHKZG EQ 'H'

```

```

* select SINGLE YYEBELN YYEBEL2 YYLIFNR
*   into (t_tab_h-YYEBELN, t_tab_h-YYEBEL2, t_tab_h-LIFNR)

```

```

* from BSIS
*   WHERE BUKRS EQ tabNN1-BUKRS
*     and HKONT NE space
*     and AUGDT EQ space
*     and AUGBL EQ space
*     and ZUONR NE space
*     and GJAHR EQ tabNN1-GJAHR
*     and BELNR EQ tabNN1-BELNR
*     "and BUZEI EQ '001'
***   and SHKZG EQ 'H'

g_OK = sy-subrc.

else. " tabNN1-BLART NE pBLART_d.

select SINGLE YYEBELN YYEBEL2 YYLIFNR
into (t_tab_h-YYEBELN, t_tab_h-YYEBEL2, t_tab_h-LIFNR)
from BSIS
  WHERE BUKRS EQ tabNN1-BUKRS
    and BELNR EQ tabNN1-BELNR
    and GJAHR EQ tabNN1-GJAHR
    and HKONT EQ J_HKONT

* select SINGLE YYEBELN YYEBEL2 YYLIFNR
*   into (t_tab_h-YYEBELN, t_tab_h-YYEBEL2, t_tab_h-LIFNR)
*   from BSIS
*     WHERE BUKRS EQ tabNN1-BUKRS
*       and HKONT EQ J_HKONT
*       and AUGDT EQ space
*       and AUGBL EQ space
*       and ZUONR NE space
*       and GJAHR EQ tabNN1-GJAHR
*       and BELNR EQ tabNN1-BELNR
*       "and BUZEI EQ '001'
***     and SHKZG EQ 'H'
*
*   g_OK = sy-subrc.

endif.

when 2.

if tabNN1-BLART EQ pBLART_d.

select SINGLE YYEBELN YYEBEL2 YYLIFNR
into (t_tab_h-YYEBELN, t_tab_h-YYEBEL2, t_tab_h-LIFNR)
from BSIS

```

```

WHERE BUKRS EQ tabNN1-BUKRS
  and BELNR EQ tabNN1-BELNR
  and GJAHR EQ tabNN1-GJAHR
  and YYEBELN IN s_EBELN
  and YYEBEL2 IN s_EBEL2
  and YYLIFNR IN s_LIFNR
*   and SHKZG EQ 'H'

* select SINGLE YYEBELN YYEBEL2 YYLIFNR
*   into (t_tab_h-YYEBELN, t_tab_h-YYEBEL2, t_tab_h-LIFNR)
*   from BSIS
*     WHERE BUKRS EQ tabNN1-BUKRS
*       and HKONT NE space
*       and AUGDT EQ space
*       and AUGBL EQ space
*       and ZUONR NE space
*       and GJAHR EQ tabNN1-GJAHR
*       and BELNR EQ tabNN1-BELNR
*       and YYEBELN IN s_EBELN
*       and YYEBEL2 IN s_EBEL2
*       and YYLIFNR IN s_LIFNR
*       "and BUZEI EQ '001'
***   and SHKZG EQ 'H'

g_OK = sy-subrc.

else. " tabNN1-BLART NE pBLART_d.

select SINGLE YYEBELN YYEBEL2 YYLIFNR
into (t_tab_h-YYEBELN, t_tab_h-YYEBEL2, t_tab_h-LIFNR)
from BSIS
  WHERE BUKRS EQ tabNN1-BUKRS
    and BELNR EQ tabNN1-BELNR
    and GJAHR EQ tabNN1-GJAHR
    and HKONT EQ J_HKONT
    and YYEBELN IN s_EBELN
    and YYEBEL2 IN s_EBEL2
    and YYLIFNR IN s_LIFNR
*   and SHKZG EQ 'H'
*

* select SINGLE YYEBELN YYEBEL2 YYLIFNR
*   into (t_tab_h-YYEBELN, t_tab_h-YYEBEL2, t_tab_h-LIFNR)
*   from BSIS
*     WHERE BUKRS EQ tabNN1-BUKRS
*       and HKONT EQ J_HKONT
*       and AUGDT EQ space

```

```

*      and AUGBL EQ space
*      and ZUONR NE space
*      and GJAHR EQ tabNN1-GJAHR
*      and BELNR EQ tabNN1-BELNR
*      and YYEBELN IN s_EBELN
*      and YYEBEL2 IN s_EBEL2
*      and YYLIFNR IN s_LIFNR
      "and BUZEI EQ '001'
**      and SHKZG EQ 'H'

```

```

      g_OK = sy-subrc.

```

```

      endif.

```

```

    endcase.

```

```

endform.

```

```

FORM count_NN_value USING x_BUKRS type BUKRS
      x_BELNR type BELNR_D
      x_GJAHR type GJAHR
      x_BLART type BLART
      x_LIFNR type LIFNR
      CHANGING x_DMBTR type DMBTR
      x_HWBAS type HWBAS
      x_MWSTS type MWSTS
      x_NAME1 type NAME1_GP
      x_STCD1 type STCD1
      x_STCD2 type STCD2

```

```

clear: x_DMBTR ,
      x_HWBAS,
      x_MWSTS,
      x_NAME1,
      x_STCD1,
      x_STCD2.

```

```

*** if x_BLART EQ pBLART_d.
***   select sum( DMBTR )
***     into x_DMBTR
***     from BSAK
***       WHERE BUKRS EQ tabNN1-BUKRS
***         and BELNR EQ tabNN1-BELNR
***         and GJAHR EQ tabNN1-GJAHR
***         and SHKZG EQ 'S'
***

```

```

***  x_HWBAS = x_DMBTR.
***  x_MWSTS = x_DMBTR.
***  else.
*   select sum( HWBAS ) sum( HWSTE )
*   into (x_HWBAS, x_MWSTS)
*   from BSET
*   WHERE BUKRS EQ tabNN1-BUKRS
*   and BELNR EQ tabNN1-BELNR
*   and GJAHR EQ tabNN1-GJAHR
*   and SHKZG EQ 'S'
*
*
*   x_DMBTR = x_HWBAS + x_MWSTS.

clear: t_tab_x, t_tab_x[].
select HWBAS HWSTE
into TABLE t_tab_x
from BSET
WHERE BUKRS EQ x_BUKRS
and BELNR EQ x_BELNR
and GJAHR EQ x_GJAHR
and SHKZG EQ 'S'

loop at t_tab_x.
  x_HWBAS = x_HWBAS + t_tab_x-HWBAS.
  x_MWSTS = x_MWSTS + t_tab_x-HWSTE.
ENDLOOP.

x_DMBTR = x_HWBAS + x_MWSTS.

***  endif.
clear: t_LFA.
READ TABLE t_LFA WITH KEY LIFNR = x_LIFNR.
x_NAME1 = t_LFA-NAME1
x_STCD1 = t_LFA-STCD1
x_STCD2 = t_LFA-STCD2

endform.

FORM count_node_value USING x_BUKRS type BUKRS
                        x_BELNR type BELNR_D
                        x_GJAHR type GJAHR
                        "x_BLART type BLART
                        x_LIFNR type LIFNR
CHANGING x_DMBTR type DMBTR
        x_HWBAS type HWBAS
        x_MWSTS type MWSTS
        x_NAME1 type NAME1_GP

```

x_STCD1 type STCD1
 x_STCD2 type STCD2

```
clear: x_DMBTR ,
      x_HWBAS,
      x_MWSTS,
      x_NAME1,
      x_STCD1,
      x_STCD2.
```

```
* select sum( HWBAS ) sum( HWSTE )
* into (x_HWBAS, x_MWSTS)
* from BSET
* WHERE BUKRS EQ tabNN1-BUKRS
* and BELNR EQ tabNN1-BELNR
* and GJAHR EQ tabNN1-GJAHR
* and SHKZG EQ 'S'
*
```

```
clear: t_tab_x, t_tab_x[].
select HWBAS HWSTE
into TABLE t_tab_x
from BSET
WHERE BUKRS EQ x_BUKRS
and BELNR EQ x_BELNR
and GJAHR EQ x_GJAHR
and SHKZG EQ 'S'
```

```
loop at t_tab_x.
  x_HWBAS = x_HWBAS + t_tab_x-HWBAS.
  x_MWSTS = x_MWSTS + t_tab_x-HWSTE.
ENDLOOP.
```

```
x_DMBTR = x_HWBAS + x_MWSTS.
```

```
clear: t_LFA.
READ TABLE t_LFA WITH KEY LIFNR = x_LIFNR.
x_NAME1 = t_LFA-NAME1
x_STCD1 = t_LFA-STCD1
x_STCD2 = t_LFA-STCD2
```

```
** select sum( DMBTR ) sum( MWSTS )
** into (x_DMBTR, x_MWSTS )
** from BSAK
** WHERE BUKRS EQ tabNN1-BUKRS
** and BELNR EQ tabNN1-BELNR
** and GJAHR EQ tabNN1-GJAHR
** and SHKZG EQ 'S'
```

```

**
**  x_HWBAS = x_DMBTR - x_MWSTS.
*
*
** select sum( HWBAS ) sum( HWSTE )
**   into (x_HWBAS, x_MWSTS)
**   from BSET
**    WHERE BUKRS EQ tabNN1-BUKRS
**      and BELNR EQ tabNN1-BELNR
**      and GJAHR EQ tabNN1-GJAHR
**      and SHKZG EQ 'S'
**
**
**  x_DMBTR = x_HWBAS + x_MWSTS.
*
*  READ TABLE t_LFA WITH KEY LIFNR = x_LIFNR.
*  x_NAME1 = t_LFA-NAME1
*  x_STCD1 = t_LFA-STCD1
*  x_STCD2 = t_LFA-STCD2

endform.

FORM create_tab_h.

  CLEAR: t_tab_h, t_tab_h[].

  * *pBUKRS
  * *pGJAHR
  * *pBUDAT1
  * *pBUDAT2
  * *BLART = NN + NK + pBLART_d
  * *s_BELNR
  * *s_BLDAT
  * *s_CPUDT
  * *s_BKTXT
  **s_LIFNR
  **s_EBEL2
  **s_EBELN

  if pBUDAT_N EQ 'X'.

    CALL FUNCTION 'SAPGUI_PROGRESS_INDICATOR'
    EXPORTING
      PERCENTAGE      = 0

    CALL FUNCTION 'TH_REDISPATCH'
    EXPORTING

```

check_runtime = 0.

```

select BUKRS
      BELNR
      GJAHR
      BLART
      BLDAT
      BUDAT
      MONAT
      BKTXT
into TABLE tabNN1
from BKPF
where BUKRS EQ pBUKRS
      and BELNR IN s_BELNR
      and BLART IN r_BLART
      and BLDAT IN s_BLDAT
      and BUDAT BETWEEN pBUDAT1 and pBUDAT2
      and CPUPT IN s_CPUPT
      and STBLG EQ space
      and BKTXT IN s_BKTXT
*   and XREVERSAL EQ space

loop at tabNN1.
  PERFORM get_yy CHANGING g_OK.
  if g_OK EQ 0. "sy-subrc EQ 0.
    PERFORM count_NN_value USING tabNN1-BUKRS
                                tabNN1-BELNR
                                tabNN1-GJAHR
                                tabNN1-BLART
                                t_tab_h-LIFNR
                                CHANGING t_tab_h-DMBTR
                                t_tab_h-HWBAS
                                t_tab_h-MWSTS
                                t_tab_h-NAME1
                                t_tab_h-STCD1
                                t_tab_h-STCD2

    t_tab_h-BELNR_NN = tabNN1-BELNR.
    t_tab_h-BUDAT = tabNN1-BUDAT.
    t_tab_h-BLDAT = tabNN1-BLDAT.
    t_tab_h-MONAT = tabNN1-MONAT.
    t_tab_h-GJAHR = tabNN1-GJAHR.
    t_tab_h-BLART = tabNN1-BLART.
    if tabNN1-BLART EQ pBLART_d.
      t_tab_h-BELNR_OSOV = tabNN1-BELNR.
      t_tab_h-GJAHR2 = tabNN1-GJAHR.
    else.

```

```

        t_tab_h-BELNR_OSOV = tabNN1-BKTXT(10).
        t_tab_h-GJAHR2 = tabNN1-BKTXT+10(4).
    endif.
    append t_tab_h.
ENDIF.
ENDLOOP.

clear: tabNN1, tabNN1[].

    CALL FUNCTION 'SAPGUI_PROGRESS_INDICATOR'
    EXPORTING
        PERCENTAGE      = 100

endif.

*pBUKRS
*pBUDAT1
*pBUDAT2
*
*pBUKRS
**pGJAHR
*BLART = NN + NK + pBLART_d
*s_BELNR
*s_BLDAT
*s_CPUDT
*s_BKTXT
*
*s_LIFNR
*s_EBEL2
*s_EBELN

if pBUDAT_A EQ 'X'.

    CALL FUNCTION 'SAPGUI_PROGRESS_INDICATOR'
    EXPORTING
        PERCENTAGE      = 0

    CALL FUNCTION 'TH_REDISPATCH'
    EXPORTING
        check_runtime = 0.

select BUKRS
        BELNR
        GJAHR
        BLART
        BLDAT

```

```

      BUDAT
      MONAT
      BKTXT
into TABLE tabNN1
from BKPF
where BUKRS EQ pBUKRS
      and BELNR IN s_BELNR
      and BLART IN r_BLART
      and BLDAT IN s_BLDAT
      and BUDAT LT pBUDAT1 "LE pBUDAT2
      and CPUDT IN s_CPUDT
      and STBLG EQ space
      and BKTXT IN s_BKTXT

```

loop at tabNN1.

```

      PERFORM get_yy CHANGING g_OK.
      if g_OK EQ 0.      "sy-subrc EQ 0.

```

```

      clear: l_FI001_SUMM.
      select SUM( ZUE_FI001_SUMM )
      into l_FI001_SUMM
      from ZUE_FI001_TAB1
      where BUKRS EQ tabNN1-BUKRS
            and BELNR EQ tabNN1-BELNR

```

```

      PERFORM count_NN_value USING tabNN1-BUKRS
                                tabNN1-BELNR
                                tabNN1-GJAHR
                                tabNN1-BLART
                                t_tab_h-LIFNR
                                CHANGING t_tab_h-DMBTR
                                t_tab_h-HWBAS
                                t_tab_h-MWSTS
                                t_tab_h-NAME1
                                t_tab_h-STCD1
                                t_tab_h-STCD2

```

```

      if t_tab_h-MWSTS GT l_FI001_SUMM.
      t_tab_h-BELNR_NN = tabNN1-BELNR.
      t_tab_h-BUDAT = tabNN1-BUDAT.
      t_tab_h-BLDAT = tabNN1-BLDAT.
      t_tab_h-MONAT = tabNN1-MONAT.
      t_tab_h-GJAHR = tabNN1-GJAHR.
      t_tab_h-BLART = tabNN1-BLART.
      if tabNN1-BLART EQ pBLART_d.

```

```

        t_tab_h-BELNR_OSOV = tabNN1-BELNR.
        t_tab_h-GJAHR2 = tabNN1-GJAHR.
    else.
        t_tab_h-BELNR_OSOV = tabNN1-BKTXT(10).
        t_tab_h-GJAHR2 = tabNN1-BKTXT+10(4).
    endif.
    append t_tab_h.
endif.

ENDIF.
ENDLOOP.

clear: tabNN1, tabNN1[].

CALL FUNCTION 'SAPGUI_PROGRESS_INDICATOR'
EXPORTING
    PERCENTAGE      = 100

endif.

endform.

FORM calculate_H.
sort t_tab_h by BUDAT.
sort t_tab_l by ITEM_KEY_FIELD BUDAT.

CALL FUNCTION 'SAPGUI_PROGRESS_INDICATOR'
EXPORTING
    PERCENTAGE      = 0

CALL FUNCTION 'TH_REDISPATCH'
EXPORTING
    check_runtime = 0.

loop at t_tab_h.

SELECT sum( ZUE_FI001_SUMM )
into t_tab_h-ZUE_FI001_SUMM1
FROM ZUE_FI001_TAB1
WHERE BUKRS EQ pBUKRS
and BELNR EQ t_tab_h-BELNR_NN
and GJAHR LE t_tab_h-GJAHR
and ZUE_FI001_DATE LE pBUDAT2 "LT pBUDAT1

SELECT sum( ZUE_FI001_SUMM )

```

```

into t_tab_h-ZUE_FI001_SUMM2
FROM ZUE_FI001_TAB2
WHERE BUKRS EQ pBUKRS
and BELNR EQ t_tab_h-BELNR_NN
and GJAHR LE t_tab_h-GJAHR
and ZUE_FI001_DATE LE pBUDAT2 "LT pBUDAT1

```

t_tab_h-ZUE_FI001_SUMM3 = t_tab_h-MWSTS - t_tab_h-MWSTSSUM.

if p_but1 EQ 'X'.

```

SELECT sum( ZUE_FI001_SUMM )
into t_tab_h-ZUE_FI001_SUMM4
FROM ZUE_FI001_TAB1
WHERE BUKRS EQ pBUKRS
and BELNR EQ t_tab_h-BELNR_NN
and GJAHR EQ t_tab_h-GJAHR "pGJAHR
and ZUE_FI001_DATE GE pBUDAT1
and ZUE_FI001_DATE LE pBUDAT2

```

* t_tab_h-ZUE_FI001_SUMM5 = t_tab_h-MWSTSSUM - t_tab_h-ZUE_FI001_SUMM1 - t_tab_h-ZUE_FI001_SUMM2 - t_tab_h-ZUE_FI001_SUMM4.

"t_tab_h-ZUE_FI001_SUMM5 = t_tab_h-MWSTSSUM - t_tab_h-ZUE_FI001_SUMM1 - t_tab_h-ZUE_FI001_SUMM2.

t_tab_h-ZUE_FI001_SUMM5 = t_tab_h-MWSTSSUM - t_tab_h-ZUE_FI001_SUMM1 - t_tab_h-ZUE_FI001_SUMM2.

t_tab_h-ZUE_FI001_SUMM5 = t_tab_h-MWSTS - t_tab_h-ZUE_FI001_SUMM1 - t_tab_h-ZUE_FI001_SUMM2.

endif.

else.

```

SELECT sum( ZUE_FI001_SUMM )
into t_tab_h-ZUE_FI001_SUMM4
FROM ZUE_FI001_TAB2
WHERE BUKRS EQ pBUKRS
and BELNR EQ t_tab_h-BELNR_NN
and GJAHR EQ t_tab_h-GJAHR "pGJAHR
and ZUE_FI001_DATE GE pBUDAT1
and ZUE_FI001_DATE LE pBUDAT2

```

* t_tab_h-ZUE_FI001_SUMM5 = t_tab_h-MWSTS - t_tab_h-ZUE_FI001_SUMM1 - t_tab_h-ZUE_FI001_SUMM2 - t_tab_h-ZUE_FI001_SUMM4.

t_tab_h-ZUE_FI001_SUMM5 = t_tab_h-MWSTS - t_tab_h-ZUE_FI001_SUMM1 - t_tab_h-ZUE_FI001_SUMM2.

ENDIF.

*** loop at t_tab_AVANSPLUS
 *** WHERE HEADER_KEY_FIELD = t_tab_h-HEADER_KEY_FIELD.
 *** t_tab_h-ZUE_FI001_SUMM5 = 0.
 *** endloop.

t_tab_h-ZUE_FI001_limit = t_tab_h-ZUE_FI001_SUMM5.

endif.
 t_tab_h-CHEKER = 'X'.
 endif.

MODIFY t_tab_h.
 ENDLOOP.

CALL FUNCTION 'SAPGUI_PROGRESS_INDICATOR'
 EXPORTING
 PERCENTAGE = 100

CALL FUNCTION 'TH_REDISPATCH'
 EXPORTING
 check_runtime = 0.

ENDFORM.

FORM create_tab_l.
 data: l_line_number TYPE I,
 l_REDISPATCH_counter(2) TYPE N,
 l_line_total TYPE I,
 l_increment TYPE I,
 l_check_value TYPE I,
 l_percent(3) Type N,
 l_incr_percent(3) Type N.
 CLEAR: t_tab_l, t_tab_l[].
 l_line_number = 0.
 l_REDISPATCH_counter = 0.

CALL FUNCTION 'TH_REDISPATCH'
 EXPORTING
 check_runtime = 0.

CALL FUNCTION 'SAPGUI_PROGRESS_INDICATOR'

```
EXPORTING
  PERCENTAGE    = 0
```

```
DESCRIBE TABLE t_tab_h LINES l_line_total.
```

```
l_percent = 0.
```

```
l_check_value = 0.
```

```
if l_line_total GE 1000.
```

```
  l_increment = l_line_total / 100 .
```

```
  l_incr_percent = 1.
```

```
elseif l_line_total GE 100.
```

```
  l_increment = l_line_total / 10.
```

```
  l_incr_percent = 10.
```

```
elseif l_line_total GE 10.
```

```
  l_increment = l_line_total / 2.
```

```
  l_incr_percent = 50.
```

```
else.
```

```
  l_increment = l_line_total.
```

```
  l_incr_percent = 0.
```

```
ENDIF.
```

```
  l_percent = l_percent + l_incr_percent.
```

```
  l_check_value = l_check_value + l_increment.
```

```
loop at t_tab_h
```

```
  CLEAR: t_node, t_node[].
```

```
  l_line_number = l_line_number + 1.
```

```
  l_REDISPATCH_counter = l_REDISPATCH_counter + 1.
```

```
  if l_REDISPATCH_counter EQ 0.
```

```
    CALL FUNCTION 'TH_REDISPATCH'
```

```
      EXPORTING
```

```
        check_runtime = 0.
```

```
  endif.
```

```
  if l_line_number EQ l_check_value.
```

```
    CALL FUNCTION 'SAPGUI_PROGRESS_INDICATOR'
```

```
      EXPORTING
```

```
        PERCENTAGE    = l_percent
```

```
  l_check_value = l_check_value + l_increment.
```

```
  l_percent = l_percent + l_incr_percent.
```

```
endif.
```

```
select SINGLE BELNR GJAHR BLART BLDAT BUDAT
```

```
  INTO (t_node-BELNR, t_node-GJAHR, t_node-BLART, t_node-BLDAT,  
t_node-BUDAT)
```

```
  from BKPF
```

```
  where BUKRS EQ pBUKRS
```

```

and BELNR EQ t_tab_h-BELNR_OSOV
and GJAHR EQ t_tab_h-GJAHR2
and STBLG EQ space
*   and XREVERSAL EQ space
continue.
endif.

```

```

select SINGLE BSIS~YYEBELN BSIS~YYEBEL2 BSIS~YYLIFNR
INTO (t_node-YYEBELN, t_node-YYEBEL2, t_node-LIFNR)
from BSIS
INNER JOIN SKB1
on   BSIS~BUKRS = SKB1~BUKRS
and BSIS~HKONT = SKB1~SAKNR
where BSIS~BUKRS EQ pBUKRS
and BSIS~BELNR EQ t_tab_h-BELNR_OSOV
and BSIS~GJAHR EQ t_tab_h-GJAHR2
and SKB1~XGKON EQ 'X'.

```

```

PERFORM count_node_value USING pBUKRS
t_node-BELNR
t_node-GJAHR
"t_node-BLART
t_node-LIFNR
CHANGING t_tab_l-DMBTR
t_tab_l-HWBAS
t_tab_l-MWSTS
t_tab_l-NAME1
t_tab_l-STCD1
t_tab_l-STCD2

```

```

t_tab_l-BELNR_PLAT = t_node-BELNR.
t_tab_l-BUDAT = t_node-BUDAT.
t_tab_l-BLDAT = t_node-BLDAT.
t_tab_l-GJAHR = t_node-GJAHR.
t_tab_l-YYEBELN = t_node-YYEBELN.
t_tab_l-YYEBEL2 = t_node-YYEBEL2.
t_tab_l-LIFNR = t_node-LIFNR.
CONCATENATE t_tab_h-BELNR_NN(10) t_tab_h-GJAHR(4) INTO t_tab_l-
ITEM_KEY_FIELD.
t_tab_h-HEADER_KEY_FIELD = t_tab_l-ITEM_KEY_FIELD.
CLEAR: l_DMBTR_V, l_HWBAS_V, l_MWSTS_V.
clear: t_tab_x, t_tab_x[].
select HWBAS HWSTE
into TABLE t_tab_x
from BSET

```

```

WHERE BUKRS EQ pBUKRS
  and BELNR EQ t_tab_h-BELNR_OSOV
  and GJAHR EQ t_tab_h-GJAHR2
  and SHKZG EQ 'H'

loop at t_tab_x.
  l_HWBAS_V = l_HWBAS_V + t_tab_x-HWBAS.
  l_MWSTS_V = l_MWSTS_V + t_tab_x-HWSTE.
ENDLOOP.
l_DMBTR_V = l_HWBAS_V + l_MWSTS_V.

PERFORM longprocess2.
endif.

if t_tab_h-HEADER_KEY_FIELD NE space.
  MODIFY t_tab_h.
endif.
endloop.

loop at t_tab_h
  where BLART EQ pBLART_d.

  l_line_number = l_line_number + 1.
  l_REDISPATCH_counter = l_REDISPATCH_counter + 1.

  if l_REDISPATCH_counter EQ 0.
    CALL FUNCTION 'TH_REDISPATCH'
      EXPORTING
        check_runtime = 0.
  endif.
  if l_line_number EQ l_check_value.
    CALL FUNCTION 'SAPGUI_PROGRESS_INDICATOR'
      EXPORTING
        PERCENTAGE      = l_percent

    l_check_value = l_check_value + l_increment.
    l_percent = l_percent + l_incr_percent.
  endif.

PERFORM processVD.

if t_tab_h-HEADER_KEY_FIELD NE space.
  MODIFY t_tab_h.
endif.

endloop. "at t_tab_h

CALL FUNCTION 'SAPGUI_PROGRESS_INDICATOR'

```

EXPORTING
PERCENTAGE = 100

```

**select single HKONT      "SAKNR
**      YYEBELN
**      YYEBEL2
**      YYLIFNR
** from BSEG              " BSAK S?
** where BUKRS EQ pBUKRS
** and BELNR EQ BELNR_node
** and GJAHR EQ GJAHR_node
** and UMSKZ IN r_UMSKZ
** and BSCHL EQ '29'
** and KOART EQ 'K'
** and UMSKS EQ 'A'
**
**
**select SINGLE XGKON
** from SKB1
** where BUKRS EQ pBUKRS
** and SAKNR EQ BSEG-SAKNR
** and XGKON EQ 'X'.
endform.

```

CLEAR: t_node, t_node[], l_AUGDT1, l_AUGDT2.

```

CONCATENATE t_tab_h-GJAHR2(4) '01' '01' INTO l_AUGDT1.
t_tab_h-GJAHR2 = t_tab_h-GJAHR2 + 1.
CONCATENATE t_tab_h-GJAHR2(4) '01' '01' INTO l_AUGDT2.
l_AUGDT2 = l_AUGDT2 - 1.
t_tab_h-GJAHR2 = t_tab_h-GJAHR2 - 1.

```

```

select  BSAK~LIFNR  BSAK~BELNR  BSAK~GJAHR  BSAK~YYEBELN
BSAK~YYEBEL2
into CORRESPONDING FIELDS OF TABLE t_node
from BSAK              "BSIK\BSAK\BSIS\BSAS
INNER JOIN BKPF
on  BSAK~BUKRS EQ BKPF~BUKRS
and BSAK~BELNR EQ BKPF~BELNR
and BSAK~GJAHR EQ BKPF~GJAHR
where BSAK~BUKRS EQ pBUKRS
and BSAK~LIFNR EQ t_tab_h-LIFNR
and BSAK~UMSKS EQ 'A'
and BSAK~UMSKZ IN r_UMSKZ
and BSAK~AUGDT Between l_AUGDT1 and l_AUGDT2
and BSAK~AUGBL EQ t_tab_h-BELNR_OSOV
and BSAK~BSCHL EQ '29'

```

and BKPF~STBLG EQ space

SORT t_node by GJAHR BELNR.

clear: l_GJAHR, l_BELNR.

loop at t_node.

if t_node-GJAHR EQ l_GJAHR and t_node-BELNR EQ l_BELNR .

delete t_node.

else.

l_GJAHR = t_node-GJAHR.

l_BELNR = t_node-BELNR.

select SINGLE BSIS~HKONT

into t_node-HKONT

from BSIS

INNER JOIN SKB1

on BSIS~BUKRS EQ SKB1~BUKRS

and BSIS~HKONT EQ SKB1~SAKNR

WHERE BSIS~BUKRS EQ pBUKRS

and BSIS~BELNR EQ t_node-BELNR

and BSIS~GJAHR EQ t_node-GJAHR

and SKB1~XGKON EQ 'X'

if sy-subrc <> 0.

delete t_node.

endif.

endif.

endloop.

loop at t_node.

t_tab_l-DMBTR = l_DMBTR_V.

t_tab_l-HWBAS = l_HWBAS_V.

t_tab_l-MWSTS = l_MWSTS_V.

clear: t_LFA.

READ TABLE t_LFA WITH KEY LIFNR = t_node-LIFNR.

t_tab_l-NAME1 = t_LFA-NAME1 .

t_tab_l-STCD1 = t_LFA-STCD1 .

t_tab_l-STCD2 = t_LFA-STCD2 .

t_tab_l-BELNR_PLAT = t_node-BELNR.

t_tab_l-BUDAT = t_node-BUDAT.

t_tab_l-BLDAT = t_node-BLDAT.

t_tab_l-GJAHR = t_node-GJAHR.

t_tab_l-YYEBELN = t_node-YYEBELN.

t_tab_l-YYEBEL2 = t_node-YYEBEL2.

t_tab_l-LIFNR = t_node-LIFNR.

CONCATENATE t_tab_h-BELNR_NN(10) t_tab_h-GJAHR(4) INTO
t_tab_l-ITEM_KEY_FIELD.

t_tab_h-HEADER_KEY_FIELD = t_tab_l-ITEM_KEY_FIELD.

```

        append t_tab_l.
    endloop.

endform.

FORM shortprocess2.
    data: l_found type I.

*       CLEAR: l_DMBTR_V, l_HWBAS_V, l_MWSTS_V.
*       clear: t_tab_x, t_tab_x[].
*       select HWBAS HWSTE
*           into TABLE t_tab_x
*           from BSET
*           WHERE BUKRS EQ pBUKRS
*           and BELNR EQ t_tab_h-BELNR_OSOV
*           and GJAHR EQ t_tab_h-GJAHR2
*           and SHKZG EQ 'S'
*
*       loop at t_tab_x.
*           l_HWBAS_V = l_HWBAS_V + t_tab_x-HWBAS.
*           l_MWSTS_V = l_MWSTS_V + t_tab_x-HWSTE.
*       ENDLOOP.
*
*       l_DMBTR_V = l_HWBAS_V + l_MWSTS_V.

CLEAR: t_node2, t_node2[].

select REBZG REBZJ
    into TABLE t_node2
    from BSAK
    where BUKRS EQ pBUKRS
    and BELNR EQ t_tab_h-BELNR_OSOV
    and GJAHR EQ t_tab_h-GJAHR2
    and REBZG NE space

select REBZG REBZJ
    APPENDING TABLE t_node2
    from BSIK
    where BUKRS EQ pBUKRS
    and BELNR EQ t_tab_h-BELNR_OSOV
    and GJAHR EQ t_tab_h-GJAHR2
    and REBZG NE space

SORT t_node2 by GJAHR BELNR.
clear: l_GJAHR, l_BELNR.
CLEAR: t_node, t_node[].

```

loop at t_node2.

if NOT (t_node2-GJAHR EQ l_GJAHR and t_node2-BELNR EQ l_BELNR

).

l_GJAHR = t_node2-GJAHR.

l_BELNR = t_node2-BELNR.

l_found = 0.

CLEAR: t_node.

select single BELNR GJAHR LIFNR YYEBELN YYEBEL2

into CORRESPONDING FIELDS OF t_node

from BSAK "BSEG

where BUKRS EQ pBUKRS

and BELNR EQ t_node2-BELNR

and GJAHR EQ t_node2-GJAHR

and UMSKS EQ 'A'

and UMSKZ IN r_UMSKZ

and BSCHL EQ '29'.

if sy-subrc <> 0.

select single BELNR GJAHR LIFNR YYEBELN YYEBEL2

into CORRESPONDING FIELDS OF t_node

from BSIK "BSEG

where BUKRS EQ pBUKRS

and BELNR EQ t_node2-BELNR

and GJAHR EQ t_node2-GJAHR

and UMSKS EQ 'A'

and UMSKZ IN r_UMSKZ

and BSCHL EQ '29'.

if sy-subrc EQ 0.

l_found = 2.

endif.

else.

l_found = 1.

endif.

if l_found GT 0.

select SINGLE BSIS~HKONT

into t_node-HKONT

from BSIS

INNER JOIN SKB1

on BSIS~BUKRS EQ SKB1~BUKRS

and BSIS~HKONT EQ SKB1~SAKNR

WHERE BSIS~BUKRS EQ pBUKRS

and BSIS~BELNR EQ t_node-BELNR

and BSIS~GJAHR EQ t_node-GJAHR

and SKB1~XGKON EQ 'X'

if sy-subrc EQ 0.

```

select SINGLE BUDAT BLDAT
into CORRESPONDING FIELDS OF t_node
from BKPF
where BUKRS EQ pBUKRS
and BELNR EQ t_node2-BELNR
and GJAHR EQ t_node2-GJAHR
and STBLG EQ space

if sy-subrc EQ 0.
  APPEND t_node.
endif.
endif.
endif.
endif.
endloop.

loop at t_node.
  t_tab_l-DMBTR = l_DMBTR_V.
  t_tab_l-HWBAS = l_HWBAS_V.
  t_tab_l-MWSTS = l_MWSTS_V.
  clear: t_LFA.
  READ TABLE t_LFA WITH KEY LIFNR = t_node-LIFNR.
  t_tab_l-NAME1 = t_LFA-NAME1 .
  t_tab_l-STCD1 = t_LFA-STCD1 .
  t_tab_l-STCD2 = t_LFA-STCD2 .
  t_tab_l-BELNR_PLAT = t_node-BELNR.
  t_tab_l-BUDAT = t_node-BUDAT.
  t_tab_l-BLDAT = t_node-BLDAT.
  t_tab_l-GJAHR = t_node-GJAHR.
  t_tab_l-YYEBELN = t_node-YYEBELN.
  t_tab_l-YYEBEL2 = t_node-YYEBEL2.
  t_tab_l-LIFNR = t_node-LIFNR.
  CONCATENATE t_tab_h-BELNR_NN(10) t_tab_h-GJAHR(4) INTO
t_tab_l-ITEM_KEY_FIELD.
  t_tab_h-HEADER_KEY_FIELD = t_tab_l-ITEM_KEY_FIELD.
  append t_tab_l.
endloop.

endform.

FORM longprocess1.

CLEAR: t_node_V, t_node_V[].
select BELNR GJAHR
into TABLE t_node_V
from BSAK
where BUKRS EQ pBUKRS
and REBZG EQ t_tab_h-BELNR_OSOV

```

and REBZJ EQ t_tab_h-GJAHR2

```
select BELNR GJAHR
APPENDING TABLE t_node_V
from BSIK
where BUKRS EQ pBUKRS
and REBZG EQ t_tab_h-BELNR_OSOV
and REBZJ EQ t_tab_h-GJAHR2
```

```
SORT t_node_V by GJAHR BELNR.
clear: l_GJAHR_v, l_BELNR_v.
loop AT t_node_V.
  if NOT ( t_node_V-GJAHR EQ l_GJAHR_v and t_node_V-BELNR EQ
l_BELNR_v ).
    l_GJAHR_v = t_node_v-GJAHR.
    l_BELNR_v = t_node_v-BELNR.
```

```
CLEAR: l_DMBTR_V, l_HWBAS_V, l_MWSTS_V.
clear: t_tab_x, t_tab_x[].
select HWBAS HWSTE
into TABLE t_tab_x
from BSET
WHERE BUKRS EQ pBUKRS
and BELNR EQ l_BELNR_v
and GJAHR EQ l_GJAHR_v
and SHKZG EQ 'H'
```

```
loop at t_tab_x.
  l_HWBAS_V = l_HWBAS_V + t_tab_x-HWBAS.
  l_MWSTS_V = l_MWSTS_V + t_tab_x-HWSTE.
ENDLOOP.
```

$l_DMBTR_V = l_HWBAS_V + l_MWSTS_V.$

CLEAR: t_node, t_node[], l_AUGDT1, l_AUGDT2.

```
CONCATENATE l_GJAHR_v(4) '01' '01' INTO l_AUGDT1.
l_GJAHR_v = l_GJAHR_v + 1.
CONCATENATE l_GJAHR_v(4) '01' '01' INTO l_AUGDT2.
l_AUGDT2 = l_AUGDT2 - 1.
l_GJAHR_v = l_GJAHR_v - 1.
```

```
select BSAK~LIFNR BSAK~BELNR BSAK~GJAHR BSAK~YYEBELN
BSAK~YYEBEL2
into CORRESPONDING FIELDS OF TABLE t_node
```

```

from BSAK
  INNER JOIN BKPF
    on BSAK~BUKRS EQ BKPF~BUKRS
    and BSAK~BELNR EQ BKPF~BELNR
    and BSAK~GJAHR EQ BKPF~GJAHR
where BSAK~BUKRS EQ pBUKRS
  and BSAK~LIFNR EQ t_tab_h-LIFNR
  and BSAK~UMSKS EQ 'A'
  and BSAK~UMSKZ IN r_UMSKZ
  and BSAK~AUGDT Between l_AUGDT1 and l_AUGDT2
  and BSAK~AUGBL EQ l_BELNR_v
  and BSAK~BSCHL EQ '29'
  and BKPF~STBLG EQ space

```

```

SORT t_node by GJAHR BELNR.
clear: l_GJAHR, l_BELNR.
loop at t_node.
  if t_node-GJAHR EQ l_GJAHR and t_node-BELNR EQ l_BELNR .
    delete t_node.
  else.
    l_GJAHR = t_node-GJAHR.
    l_BELNR = t_node-BELNR.

```

```

select SINGLE BSIS~HKONT
into t_node-HKONT
from BSIS
  INNER JOIN SKB1
    on BSIS~BUKRS EQ SKB1~BUKRS
    and BSIS~HKONT EQ SKB1~SAKNR
WHERE BSIS~BUKRS EQ pBUKRS
  and BSIS~BELNR EQ t_node-BELNR
  and BSIS~GJAHR EQ t_node-GJAHR
  and SKB1~XGKON EQ 'X'

```

```

if sy-subrc <> 0.
  delete t_node.
endif.
endif.
endloop.

```

```

loop at t_node.
  t_tab_l-DMBTR = l_DMBTR_v.
  t_tab_l-HWBAS = l_HWBAS_v.
  t_tab_l-MWSTS = l_MWSTS_v.
  clear: t_LFA.
  READ TABLE t_LFA WITH KEY LIFNR = t_node-LIFNR.
  t_tab_l-NAME1 = t_LFA-NAME1 .

```

```

t_tab_l-STCD1 = t_LFA-STCD1 .
t_tab_l-STCD2 = t_LFA-STCD2 .
t_tab_l-BELNR_PLAT = t_node-BELNR.
t_tab_l-BUDAT = t_node-BUDAT.
t_tab_l-BLDAT = t_node-BLDAT.
t_tab_l-GJAHR = t_node-GJAHR.
t_tab_l-YYEBELN = t_node-YYEBELN.
t_tab_l-YYEBEL2 = t_node-YYEBEL2.
t_tab_l-LIFNR = t_node-LIFNR.
CONCATENATE t_tab_h-BELNR_NN(10) t_tab_h-GJAHR(4) INTO
t_tab_l-ITEM_KEY_FIELD.
t_tab_h-HEADER_KEY_FIELD = t_tab_l-ITEM_KEY_FIELD.
append t_tab_l.
endloop.

endif.
endloop.
endform.

```

ДОДАТОК Д
Слайди презентації

Національний університет «Запорізька політехніка»

Кафедра програмних засобів

Дипломна кваліфікаційна робота

**Дослідження та програмна реалізація
модулів ERP системи**

Виконав

ст. гр. КНТ-219м

О. І. Музика

Керівник

к. т. н., доцент

А. О. Олійник

Рисунок Д.1 – Слайд № 1

Об’єктом дослідження є процес програмної реалізації складних систем.

Предмет дослідження – програмне забезпечення ERP систем.

Мета роботи – дослідження ERP систем та програмна реалізація модулів ERP системи корпоративного підприємства.

Рисунок Д.2 – Слайд № 2

Задачі роботи:

- аналіз корпоративних інформаційних систем;
- розробка технічного завдання для реалізації програми;
- вибір засобів розробки програмного забезпечення;
- розробка моделі модуля ERP системи територіально розподіленого підприємства
- розробка структурної схеми програми;
- конструювання програмного забезпечення;
- тестування розробленого програмного забезпечення.

3

Рисунок Д.3 – Слайд № 3

Порівняльна характеристика ERP-систем

ERP-система	Виробник	Сфера застосування	Термін впровадження / можливість внесення специфічних змін	Простота установки / використання	Вартість впровадження
Oracle Applications	Oracle (США)	Важка промисловість (металургія), телекомунікаційні компанії, фінансовий сектор, хімічна промисловість	1-5 років і більше \ можна вносити зміни	Складна в установці / використанні	Вартість рішення на одне робоче місце становить близько \$5 тис. повна вартість істотно залежить від необхідної функціональності і складності впровадження.
1С: Підприємство 8.2. Управління виробничим підприємством	1С	Машинобудування, харчова промисловість та ін.	3 - 9 місяців і більше \ можна вносити зміни	Складна в установці / використанні	Ліцензія на одне робоче місце \$ 150-600. Вартість впровадження на одне робоче місце \$ 200-1000
IFS Application	IFS (Швеція)	Підприємства машинобудівного комплексу, енергетика, харчова промисловість, фармацевтика, кабельна промисловість	0,8-3 роки і більше \ можна вносити зміни	Складна в установці / використанні	Повна вартість впровадження може досягати \$ 250 тис. і більше. Також істотно залежить від необхідної функціональності.
Baan ERP	SSA Global (США)	Хімічна промисловість, автомобілебудування, фармацевтика, харчова промисловість	6 місяців - 1,5 року і більше \ можна вносити зміни	Складна в установці / використанні	Вартість одного робочого місця - \$3 тис. співвідношення ціни рішення і витрат на впровадження 1: 1-1: 3.
iRenaissance	Ross Systems (США)	Харчова промисловість, хімічні компанії, металургійна промисловість, нафтопереробні, целюлозно-паперові, фармацевтичні підприємства	4 місяців – 1,3 року і більше \ можна вносити зміни	Складна в установці / використанні	Вартість впровадження в середньому \$ 200 тис.
SAP R/3	SAP AG (Німеччина)	Широка сфера застосування: оборонні підприємства, компанії нафтогазового комплексу, металургія, енергетика, телекомунікації, банківський сектор	1-5 років і більше \ можна вносити зміни	Проста в установці / використанні	Ліцензія на 50 робочих місць коштує близько \$350 тис. вартість впровадження може в кілька разів перевищувати вартість рішення.

4

Рисунок Д.4 – Слайд № 4

Вибір системи керування базами даних

Критерії порівняння	Система керування базами даних		
	MS Access	MySQL	Progress
Багатоплатформність	–	+	+
Швидкість розробки	±	+	+
Надійність	±	+	+
Мережеві можливості	±	+	++
Масштабованість	±	+	++
Розроблення графічного інтерфейсу користувача	+	±	+
Доступність (вимоги до розповсюдження)	±	+	±
Можливість створення складних БД для ERP-систем	–	+	++

5

Рисунок Д.5 – Слайд № 5

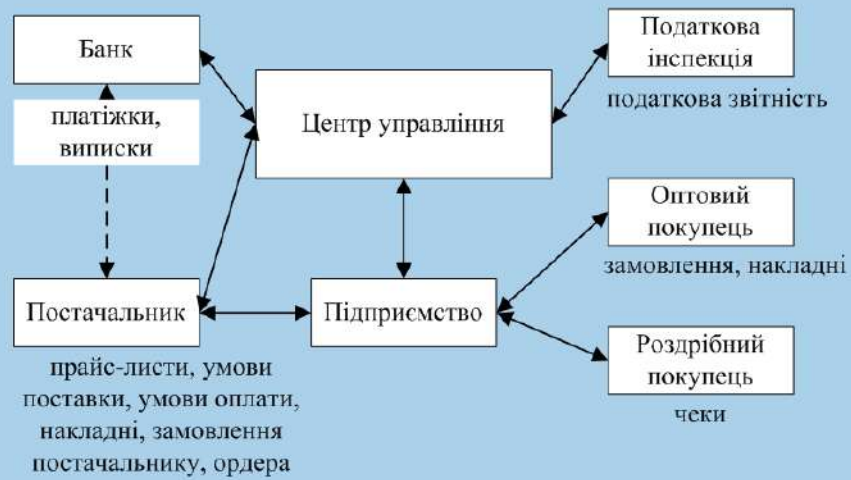
Порівняльний аналіз мов програмування

Критерії порівняння	Мова програмування		
	C#	Java	ABAP
Мережеві можливості	±	++	+
Швидкість розробки	++	+	+
Розроблення графічного інтерфейсу користувача	+	+	+
Ефективність інтеграції з SAP R/3	±	+	++
Можливість створення складних програмних модулів для ERP-систем	–	+	++

6

Рисунок Д.6 – Слайд № 6

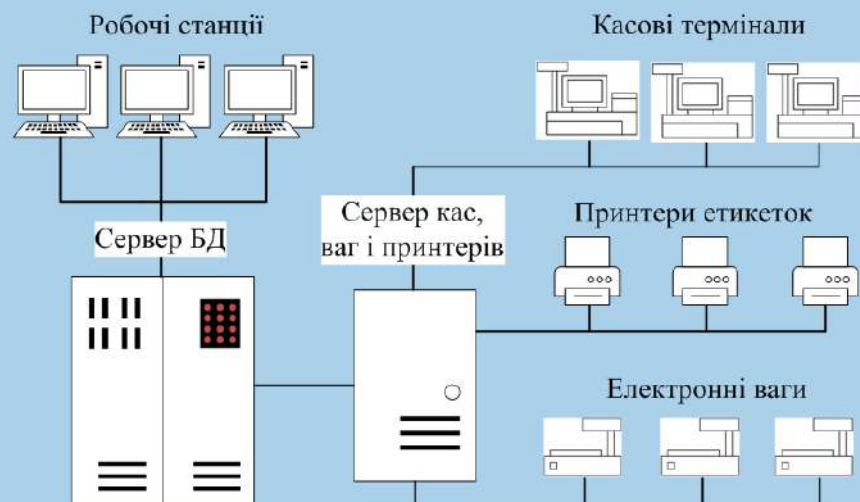
Структура взаємодії з зовнішніми об'єктами



7

Рисунок Д.7 – Слайд № 7

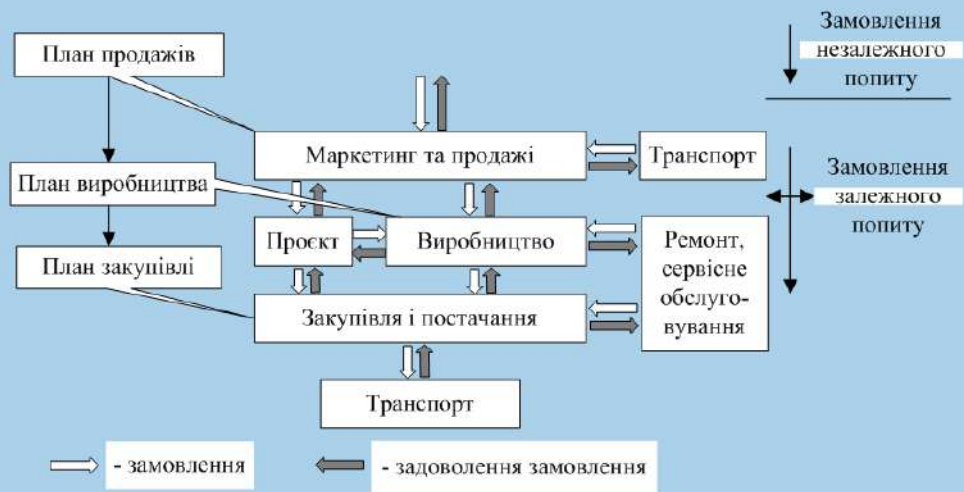
Діаграма взаємодії апаратних засобів ERP системи



8

Рисунок Д.8 – Слайд № 8

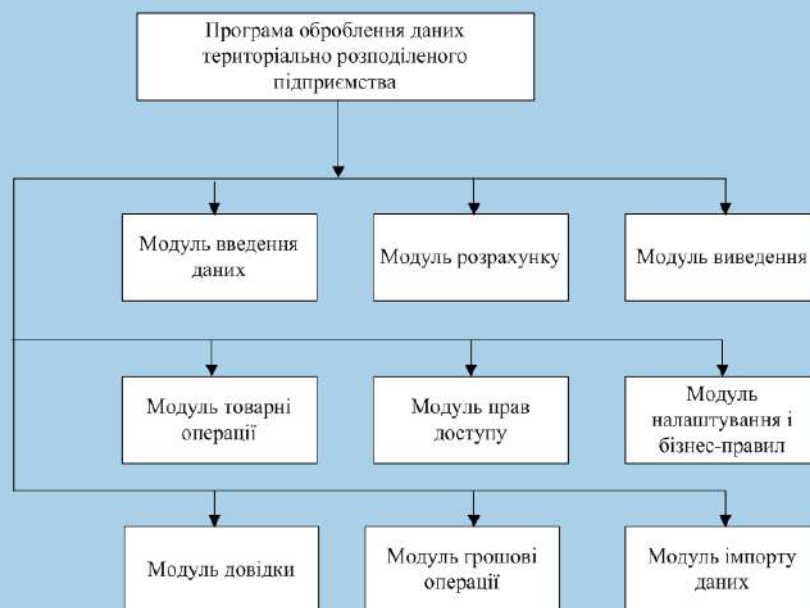
Діаграма потоку замовлень



9

Рисунок Д.9 – Слайд № 9

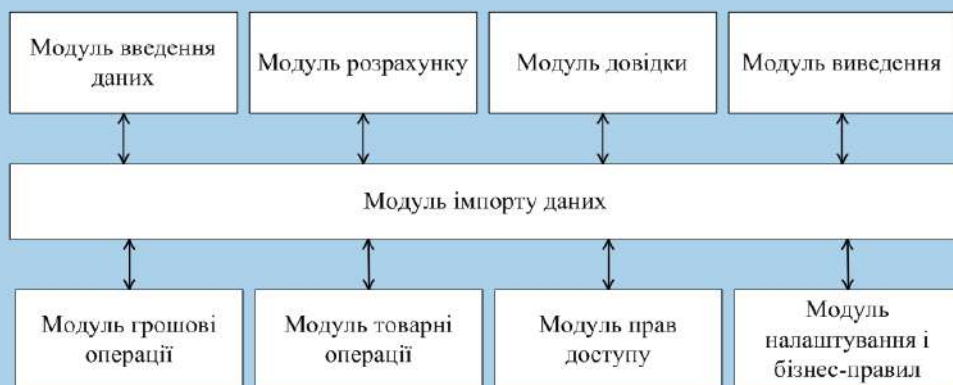
Структурна схема програми



10

Рисунок Д.10 – Слайд № 10

Схема взаємодії модулів програми



11

Рисунок Д.11 – Слайд № 11

Виконання програми

Импорт данных по ЗП и начислениям

Опции запуска

БЕ: 1000

Дата проводки: 31.08.2020

Файл: ☒

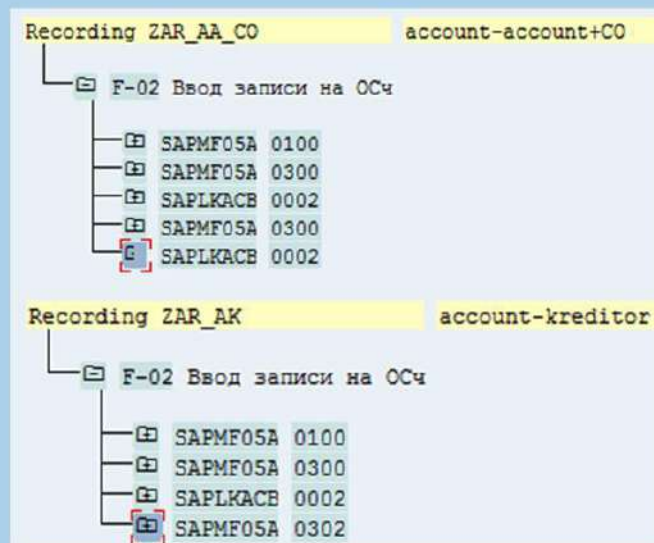
☐ Считать данные из предварительно сохраненной таблицы

☒ Считать данные из файла БД

12

Рисунок Д.12 – Слайд № 12

Виконання програми



13

Рисунок Д.13 – Слайд № 13

Виконання програми

Данные по ЗП. Файл C:\USERS\USER\DESKTOP\РАБОЧИЙ СТОЛ 2\SALARY3110.TXT

Моделирование Проводка

	О	БЕ	ДатаДокум	Текст заголовка документа	КП	Дт	СчетДебет	Заказ	КП	Кт	СчетКредит	Сумма
✓	1000	31.10.2020	SALARY10/11	40	82100100		50	10				4.817,79
✓	1000	31.10.2020	SALARY10/11	40	82100100		50	10				158,70
✓	1000	31.10.2020	SALARY10/11	40	82100100		50	10				7.507,67
✓	1000	31.10.2020	SALARY10/11	40	82100100		50	10				6.728,86
✓	1000	31.10.2020	SALARY10/11	40	82100100		50	10				624,60
✓	1000	31.10.2020	SALARY10/11	40	82100100		50	10				2.421,85
✓	1000	31.10.2020	SALARY10/11	40	82100100		50	10				8.663,67

14

Рисунок Д.14 – Слайд № 14

Висновки

У ході виконання дипломної кваліфікаційної роботи магістра було проведено огляд та аналіз корпоративних інформаційних ERP систем. Виконано дослідження ERP системи SAP R/3. Відібрані матеріали були систематизовані, проаналізовані, узагальнені і представлені в першому розділі пояснювальної записки.

В процесі розробки вирішено такі завдання:

- аналіз корпоративних інформаційних систем;
- розробка технічного завдання для реалізації програми;
- вибір засобів розробки програмного забезпечення;
- розробка моделі модуля ERP системи територіально розподіленого підприємства
- розробка структурної схеми програми;
- конструювання програмного забезпечення;
- тестування розробленого програмного забезпечення.

15

Рисунок Д.15 – Слайд № 15

Висновки

Наукова новизна роботи полягає у тому, що запропоновано модель модуля ERP системи територіально розподіленого підприємства, що надає структуру взаємодії з зовнішніми об'єктами, діаграму взаємодії апаратних засобів ERP системи, узагальнену модель системи виробництва на основі планування та діаграму потоку замовлень.

Практичне значення роботи полягає у тому, що розроблено програмні модулі ERP системи територіально розподіленого підприємства, що забезпечує підтримку внутрішніх бізнес-процесів підприємства та що надає можливості для взаємодії з зовнішніми об'єктами.

Проведені експерименти засвідчили, що програма має достатню працездатність і може використовуватися за призначенням.

16

Рисунок Д.16 – Слайд № 16